



## Guia Exemplo de Integração com API fSense

# Sumário:

## Guia Exemplo de Integração

Introdução

Principais Funcionalidades

Requisitos

Configuração do Ambiente

Primeiro Teste

## Configurações da Aplicação

Configuração do arquivo Properties

Localização do Arquivo

Configuração das Credenciais da Organização

Obtendo as Credenciais

Configurando as Credenciais

Definir a URL de domínio

Configurando a URL da API

Definir o Serviço

Definir o Método

Execução e Resultados

## Serviço de Autenticação

Processo de Autenticação

Obter Token de Acesso

Definição do Serviço

Método do Serviço

Execução da Aplicação

## Serviço de Atividades

Exportar Atividades - Exemplo de Implementação CSV

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

Exportar Atividades - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Serviço de Eventos

### Exportar Eventos - Exemplo de Implementação CSV

Configuração do Arquivo Properties.

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

### Exportar Eventos - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties.

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Serviço de Históricos de Login

### Exportar Histórico de Login - Exemplo de Implementação CSV

Configuração do Arquivo Properties.

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

### Exportar Histórico de Login - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties.

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Serviço de Equipes

### Exportar Equipes - Exemplo de Implementação CSV

Configuração do Arquivo Properties.

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Exportar Equipes - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties.

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Importar Equipes

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Informar o Caminho do Arquivo

Exemplo de Arquivo de Importação de Equipe

Execução da Aplicação

Verificação da Operação

## Atualizar Equipes por ID

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Informar o Caminho do Arquivo

Execução da Aplicação

Verificação da Operação

## Deletar Equipes por ID

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Definição de Parâmetros de Remoção

Execução da Aplicação

Verificação da Operação

## Exportar Detalhes de Equipes

Configuração do Arquivo Properties

Definição do serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Adicionar Membros

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Remove Membros

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Adicionar Líderes

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Remove Líderes

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

### Serviço de Pessoas

#### Exportar Pessoas - Exemplo de Implementação CSV

Configuração do Arquivo Properties.  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar Pessoas - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties  
Definição do Serviço

Método do Serviço  
Tipo de Operação  
Execução da Aplicação  
Verificação da Operação

#### Importar Pessoas

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Informar o Caminho do Arquivo  
Execução da Aplicação  
Verificação da Operação

#### Atualizar Pessoas

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Informar o Caminho do Arquivo  
Execução da Aplicação  
Verificação da Operação

#### Exportar Líderes de Pessoas

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Informar o Caminho do Arquivo  
Execução da Aplicação  
Verificação da Operação

### Serviço de Categorias

#### Exportar Categorias - Exemplo de Implementação CSV

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar Categorias - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação

Execução da Aplicação  
Verificação da Operação

#### Exportar Categorias Gerais - Exemplo de Implementação de CSV

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar Categorias Gerais - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar Categorias por Equipe - Exemplo de Implementação CSV

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar Categorias por Equipe - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

### Serviço de Jornadas

#### Exportar Jornadas - Exemplo de Implementação CSV

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

## Exportar Jornadas - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Importar Jornadas

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

## Informar o Caminho do Arquivo

Execução da Aplicação

Verificação da Operação

## Exportar Jornadas por ID

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Atualizar Jornadas por ID

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Informar o Caminho do Arquivo

Execução da Aplicação

Verificação da Operação

## Deletar Jornadas por ID

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Definição de Parâmetros

Verificação da Operação

## Associar Pessoas

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Desassociar Pessoas

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Associar Equipes

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Desassociar Equipes

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar pessoas associadas a uma jornada por ID

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar equipes associadas a uma jornada por ID

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação

Verificação da Operação

## Trocas de Jornada

### Exportar Trocas de Jornada - Exemplo de Implementação CSV

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Execução

Execução da Aplicação

Verificação da Operação

### Exportar Trocas de Jornada - Exemplo de Implementação de Banco de dados

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

### Importar Trocas de Jornada

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Informar o Caminho do Arquivo

Execução da Aplicação

Verificação da Operação

### Deletar Trocas de Jornada

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Definição de Parâmetros de Remoção

Execução da Aplicação

Verificação da Operação

## Acréscimos de Jornada

### Exportar Acréscimos de Jornada - Exemplo de Implementação CSV

Configuração do Arquivo Properties

Definição do Serviço

Método do Serviço

Tipo de Operação

Parâmetros de Exportação

Execução da Aplicação

Verificação da Operação

## Exportar Acréscimos de Jornada - Exemplo de Implementação de Banco de Dados

- Configuração do Arquivo Properties
- Definição do Serviço
- Método do Serviço
- Tipo de Operação
- Parâmetros de Exportação
- Execução da Aplicação
- Verificação da Operação

## Importar Acréscimos de Jornada

- Configuração do Arquivo Properties
- Definição do Serviço
- Método do Serviço
- Tipo de Operação
- Informar o Caminho do Arquivo
- Execução da Aplicação
- Verificação da Operação

## Deletar Acréscimos de Jornada

- Configuração do Arquivo Properties
- Definição do Serviço
- Método do Serviço
- Tipo de Operação
- Definição de Parâmetros de Remoção
- Execução da Aplicação
- Verificação da Operação

## Ferriados

### Exportar Ferriados - Exemplo de Implementação CSV

- Configuração do Arquivo Properties
- Definição do Serviço
- Método do Serviço
- Tipo de Operação
- Parâmetros de Exportação
- Execução da Aplicação
- Verificação da Operação

### Exportar Ferriados - Exemplo de Implementação de Banco de Dados

- Definição do Serviço
- Método do Serviço
- Tipo de Operação
- Parâmetros de Exportação
- Execução da Aplicação
- Verificação da Operação

### Importar Ferriados

- Configuração do Arquivo Properties
- Definição do Serviço

Método do Serviço  
Tipo de Operação  
Informar o Caminho do Arquivo  
Execução da Aplicação  
Verificação da Exportação

#### Atualizar Feriados

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Informar o Caminho do Arquivo  
Execução da Aplicação  
Verificação da Exportação

#### Exportar Feriados por ID

Configuração do Arquivo Properties  
Definição do Serviço

#### Método do Serviço

Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Deletar Feriados por ID

Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Definição de Parâmetros de Remoção  
Execução da Aplicação  
Verificação da Operação

### Serviço de Teclas/Mouse

#### Exportar Teclas/Mouse - Exemplo de Implementação CSV

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação  
Execução da Aplicação  
Verificação da Operação

#### Exportar Teclas/Mouse - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo Properties  
Definição do Serviço  
Método do Serviço  
Tipo de Operação  
Parâmetros de Exportação

Execução da Aplicação  
Verificação da Operação

## Serviço de Calendário

### Exportar Calendários - Exemplo de Implementação CSV

Configuração do Arquivo fsense.properties.  
Execução da Aplicação  
Verificação da Operação

### Exportar Calendários - Exemplo de Implementação de Banco de Dados

Configuração do Arquivo fsense.properties.  
Execução da Aplicação  
Verificação da Operação

### Exportar Calendário por Código Externo

Configuração do Arquivo fsense.properties.  
Execução da Aplicação  
Verificação da Operação

### Exportar Calendário por Id

Configuração do Arquivo fsense.properties.  
Execução da Aplicação  
Verificação da Operação

### Importar Calendário

Execute a aplicação  
Verificação da Operação  
Modelo de Arquivo de CSV para importação

### Atualizar Calendário

Execute a aplicação  
Verificação da Operação  
Modelo de Arquivo de CSV para importação

### Remover Calendário

Execução da Aplicação  
Verificação da Operação

### Exportar Associações de Calendário

Configuração do Arquivo fsense.properties.  
Execução da Aplicação  
Verificação da Operação

## Integração com Banco de Dados

Criação das Tabelas  
Credenciais do Usuário  
String de Conexão  
Teste de Conexão

## Importação de Arquivos CSV

# Guia Exemplo de Integração

O Guia exemplo de Integração oferece um guia passo a passo com exemplos de como integrar cada funcionalidade do fSense. Consulte o guia agora e acelere o processo de integração em seus projetos.

## Introdução

Esta aplicação de console desenvolvida em Java, oferece uma demonstração prática do processo de consumo da API de Integração do fSense. O objetivo principal é demonstrar chamadas HTTP à API, processar os dados de resposta e fornecer uma interface eficiente para diversas interações.

A aplicação permite internalização de equipes, gestão de pessoas, configuração de jornadas de trabalho, alterações nas jornadas, cadastro de feriados, entre outros. Além disso, é possível a exportação de relatórios no formato CSV e suporta a integração com banco de dados.

Ao executar a aplicação, é possível visualizar informações relevantes no console, incluindo detalhes sobre o estado da execução, logs e feedbacks provenientes da API de Integração do fSense.

## Principais Funcionalidades

- **Chamadas HTTP à API fSense:** A aplicação faz chamadas HTTP para a API de Integração do fSense facilitando a comunicação e troca de informações.
- **Processamento de Dados de Resposta:** Os dados retornados pela API são processados pela aplicação para garantir uma integração eficiente e apropriada com o sistema fSense.
- **Internalização de dados:** Suporte para Internalização de Equipes, Pessoas, Jornadas, alterações de jornadas, acréscimos de jornada, cadastros de feriados, etc.
- **Exportação de Relatórios em CSV:** A Aplicação permite a geração de relatórios no formato CSV, simplificando o compartilhamento e análise de dados.

- **Integração com Banco de Dados:** Alguns métodos contém exemplos práticos de integração com bancos de dados. Para mais informações sobre essa integração, consulte o link fornecido no tópico específico



#### DICA

Para obter Detalhes sobre como configurar a Integração com bancos de dados, consulte o link a seguir: [Integração com Banco de Dados](#)

Para Fazer o Download do código fonte da aplicação, basta clicar no botão abaixo:

Fazer Download

Após ter feito o Download, descompacte o arquivo para um diretório da sua preferência.

## Requisitos

Antes de executar o aplicativo, assegure-se de que os seguinte requisitos estejam devidamente instalados em seu ambiente:

- **Java JDK 11 ou superior (preferencial JDK 11)** - O aplicativo foi desenvolvido para ser executado em um ambiente Java, garantindo a compatibilidade com JDK 11 ou versões mais recentes. Recomendamos a utilização do JDK 11 para uma experiência ideal.
- **IDE de sua preferencia (Eclipse, IntelliJ, Netbeans, etc...)** - Escolha a IDE que melhor se alinha com suas preferências de desenvolvimento, como Eclipse, IntelliJ, Netbeans entre outras. Certifique-se de que esla esteja devidamente instalada e configurada.
- **Servidor de Banco de Dados** - Para integração com Banco de Dados, é necessário um servidor de banco de dados. O aplicativos suporta servidores como o MySQL Server, ou o PostgreSQL ou SQLServer. Certifique-se de que o servidor de bancos de dados esteja em execução.
- **Cliente SGBD(Sistema de Gerenciamento de Banco de Dados)** - Um cliente SGBD é essencial para facilitar a administração e interação com o banco de dados. Exemplos de clientes incluem o MySQL Workbench, pgAdmin, Dbeaver.



#### CONFIGURAÇÃO ADICIONAL

Se a integração com o banco de dados for necessária, consulte o tópico específico sobre integração de banco de dados para obter informações detalhadas e orientações de configuração.

### **Integração com Banco de Dados**

Com estes requisitos atendidos, você estará pronto para iniciar a execução do aplicativo de forma eficaz.

## **Configuração do Ambiente**

Para garantir uma configuração adequada do ambiente para execução do projeto, siga os passos abaixo:

- **Abra o projeto em sua IDE** - Abra o projeto utilizando a sua IDE de preferência.
- **Exemplo usando o Eclipse** - Este guia utiliza o Eclipse como exemplo para demonstração, mas os passos geralmente são semelhantes em outras IDEs.
- **Baixar as dependências do Maven** - Certifique-se de baixar as dependências do Maven para garantir que o projeto esteja configurado corretamente.
- **Baixar Dependências manualmente** - Se a IDE não baixar as dependências automaticamente, clique com botão direito na pasta principal do projeto, escolha a opção **Maven**.

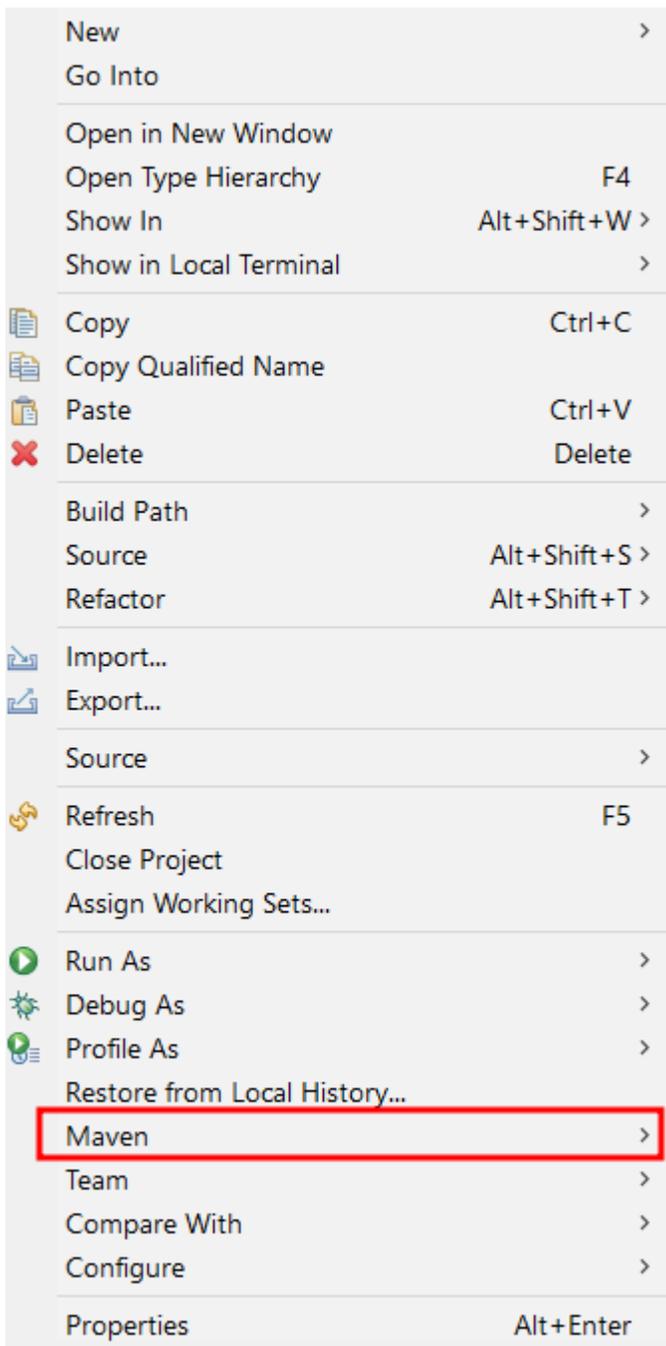
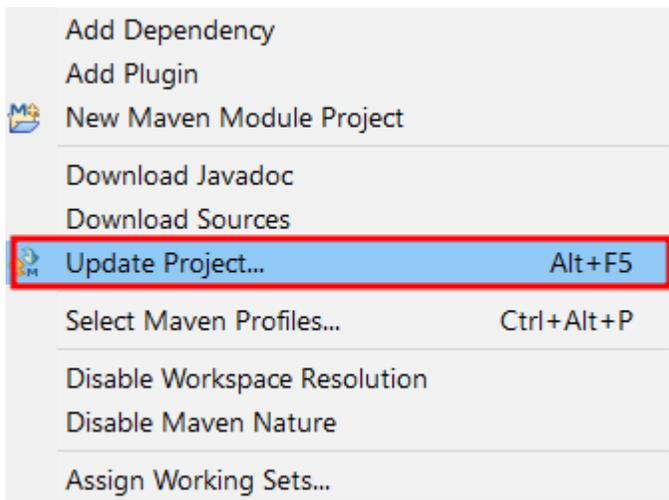


Figura 1 - Eclipse - Project Maven

- **Atualizar Projeto** - Selecione a opção **Update Project**.



*Figura 2 - Eclipse - Update Project*

- **Forçar Atualizações de Snapshots/Releases** - Marque a opção **Force Update of Snapshots/Releases**, clique em Ok.

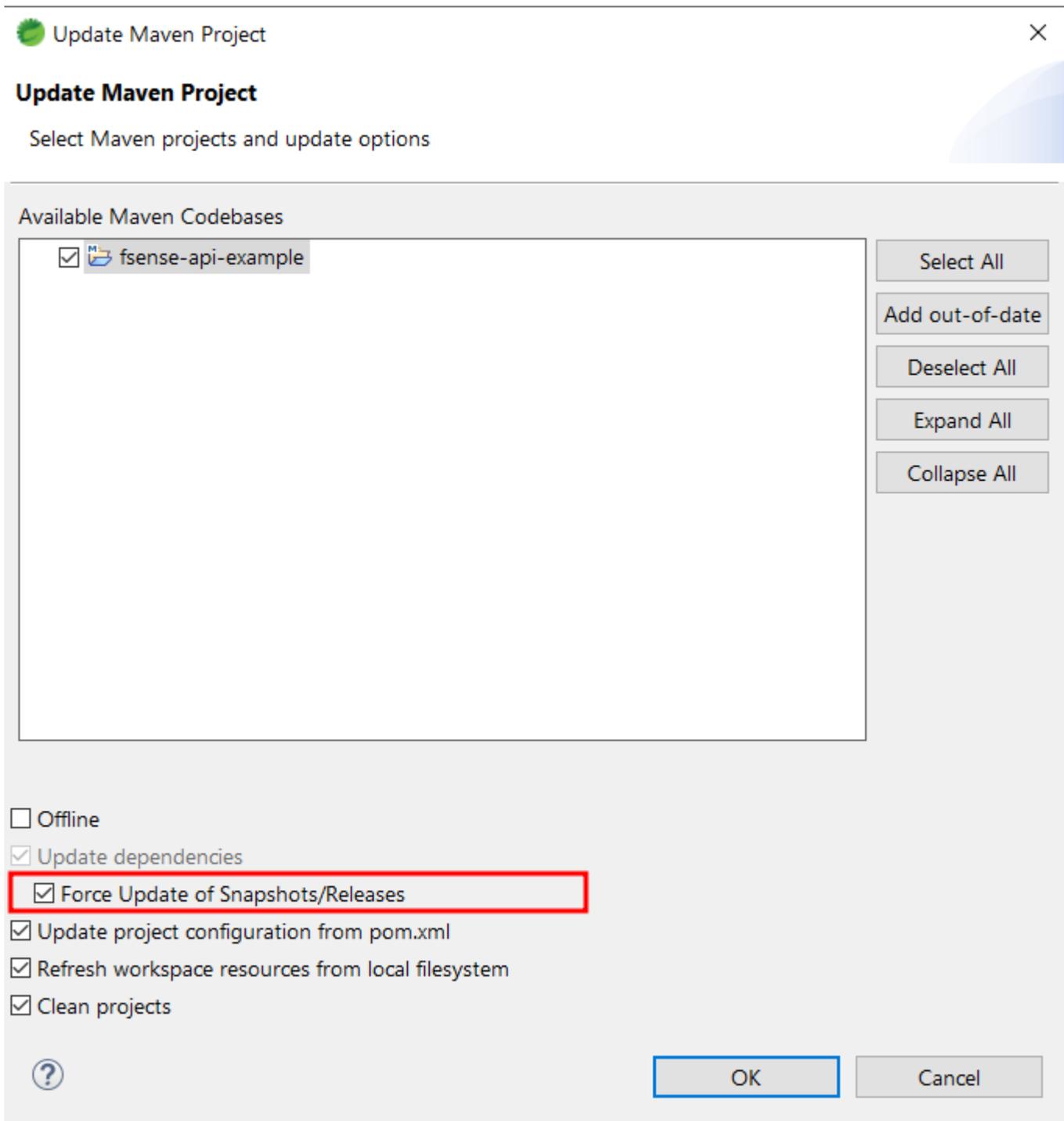


Figura 3 - Eclipse - Force Update of Snapshots/Releases

- **Configurar a Execução** - Clique novamente na pasta principal do projeto, escolha a Opção **Run as** e depois em **Run configurations...**

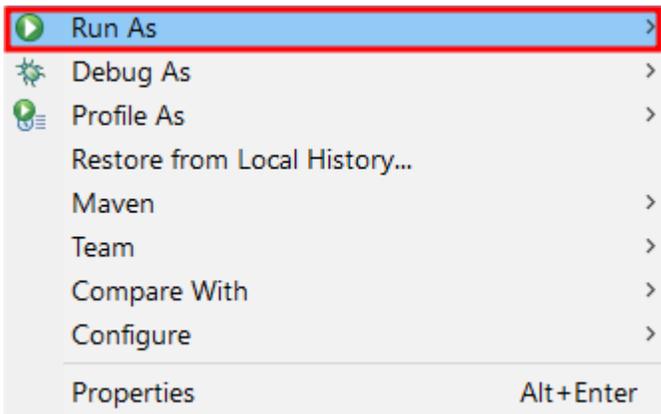


Figura 4 - Eclipse - Run As

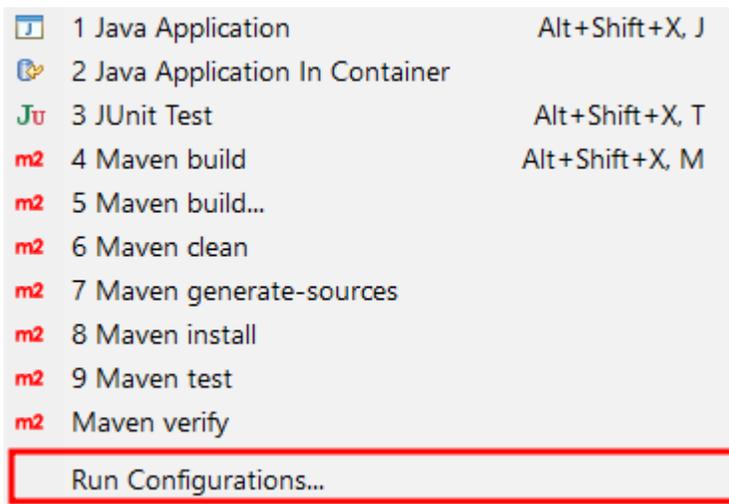


Figura 5 - Eclipse - Project Run Configurations

- **Selecionar a Versão do Java** - Na aba JRE, selecione **Java-SE11 (jre)**.

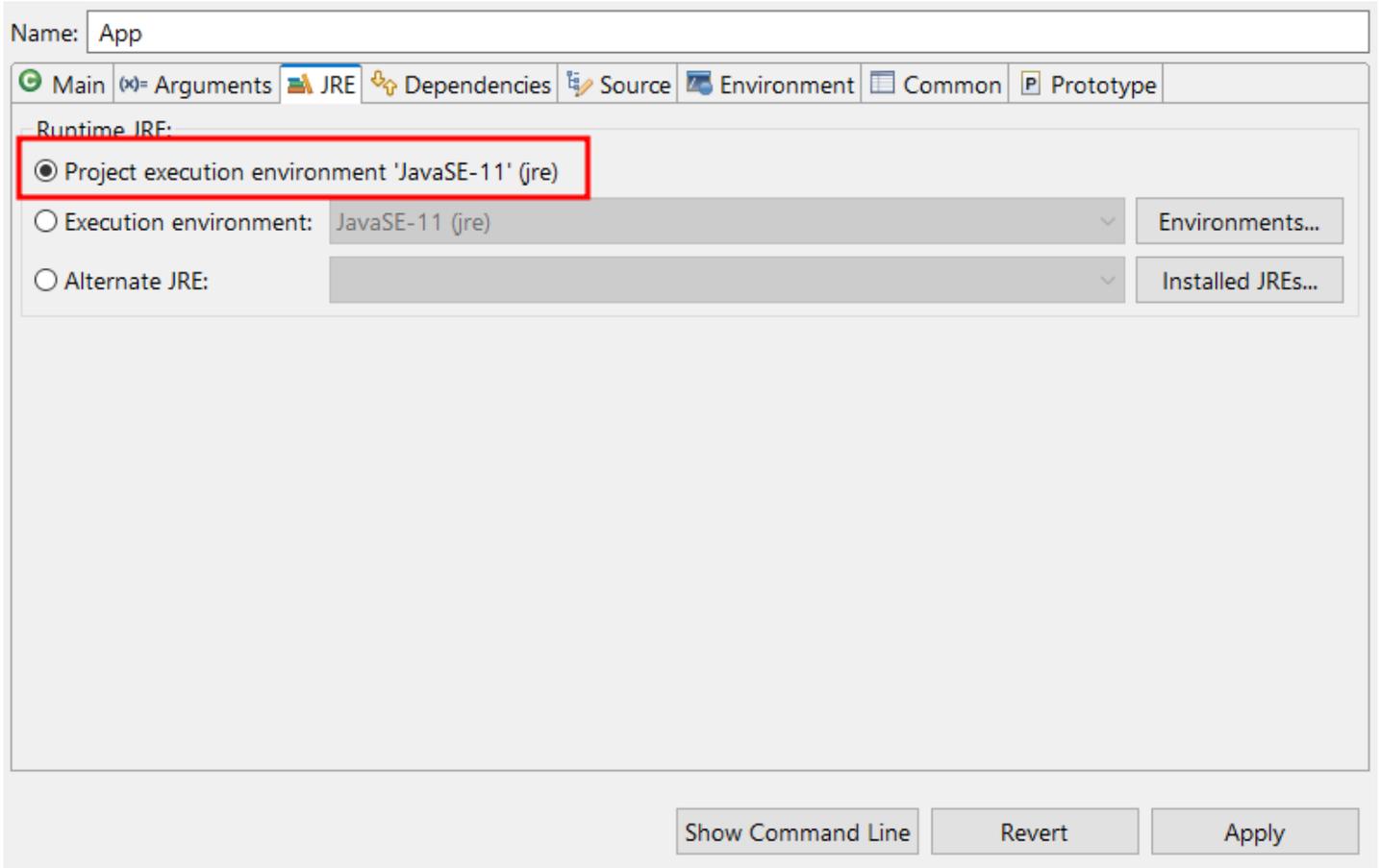


Figura 6 - Eclipse - JRE (SE) 11

- **Seleção Manual da Versão do Java** - Se caso estiver selecionado uma versão superior do java, você pode selecionar manualmente marcando a opção **Execution Environments**.

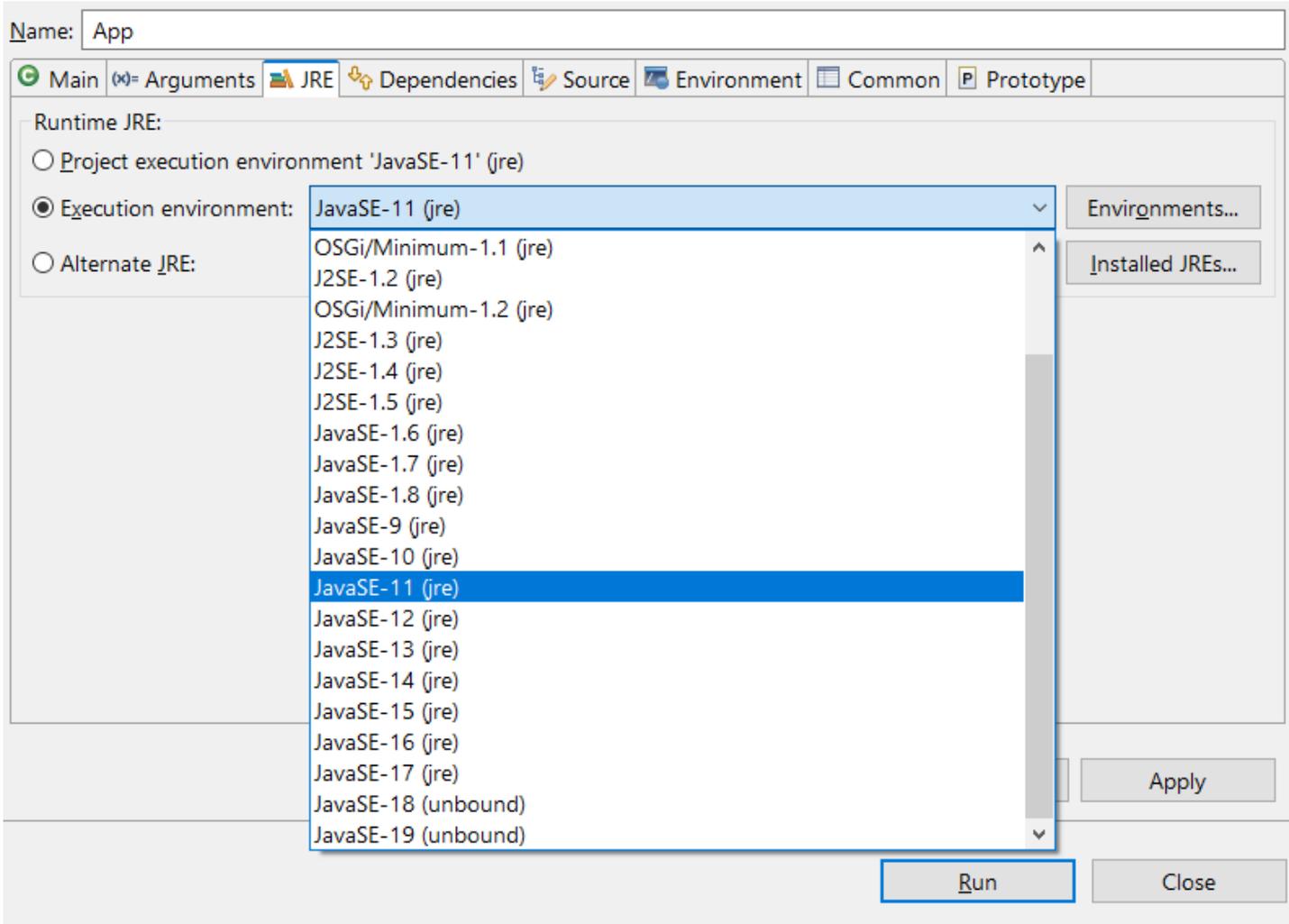


Figura 7 - Eclipse - JRE personalizado

- **Especificar o caminho do arquivo fsense.properties** - Na aba Arguments, selecione o caminho para o arquivo **fsense.properties** que é o argumento de linha de comando necessário pra inicialização do programa, como no exemplo abaixo.

```
C:\Users\%USER%\projects\Suporte\fsense-api-example\src\main\resources\fsense.properties
```

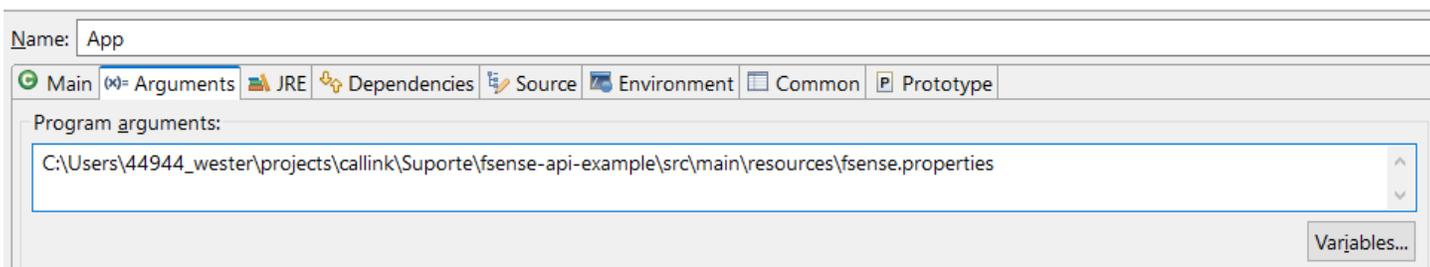


Figura 8 - Eclipse - Argumentos

Ao seguir estes passos, você garantirá uma configuração adequada do ambiente para a execução do projeto. Consulte as informações anteriores ou posteriores conforme necessário.

## Primeiro Teste

- **Execução Inicial** - Agora vamos executar o programa pra verificar se o ambiente foi configurado corretamente.
- **Selecionar o Arquivo App.java** - Navegue até a pasta **src/main/java** selecione o arquivo **App.java** clique com botão direito e selecione a opção **Run as** seguido de **Java application**.

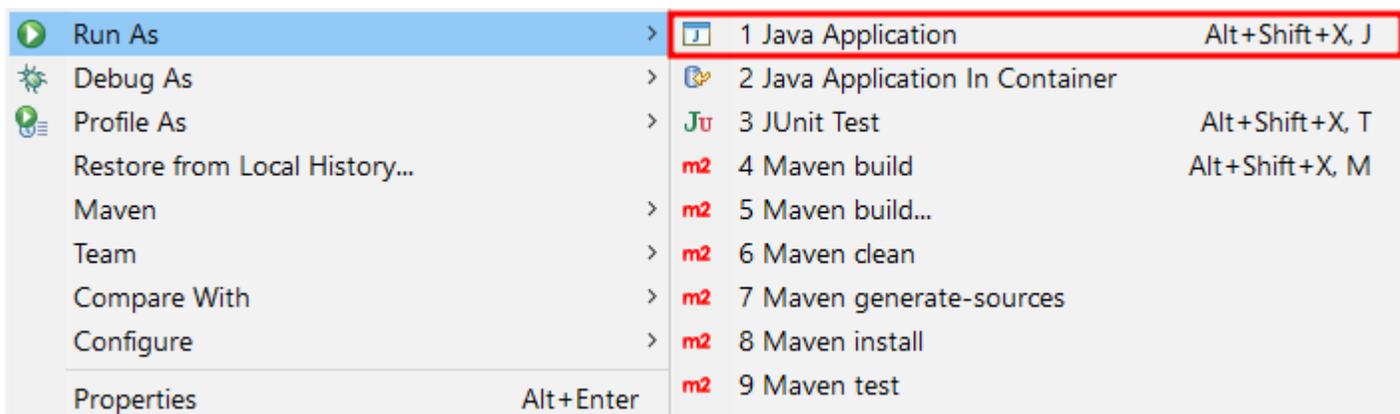


Figura 9 - Eclipse - Executa Aplicação

- **Verificação de Mensagem no Console** - Confirme se a mensagem **No service found for informed name** foi exibida no console. Isso indica que o ambiente está configurado corretamente. Agora avançaremos para a configuração do arquivo **fsense properties**.

```
=====
SELECTED SERVICE []
=====
No service found for informed name:
=====
Available services:
[LOGIN_HISTORY_SERVICE]
```

```
[PERSON_SERVICE]
[ACTIVITY_SERVICE]
[EVENT_SERVICE]
[WORKDAY_SERVICE]
[CALENDAR_SERVICE]
[CATEGORY_SERVICE]
[DATABASE_TEST_SERVICE]
[TEAM_SERVICE]
[WORKDAY_OVERRIDE_SERVICE]
[WORKDAY_PROROGATION_SERVICE]
[HOLIDAY_SERVICE]
[KEYSTROKES_SERVICE]
[AUTH_SERVICE]
=====
```

- **Mensagem Não Apareceu** - Caso a mensagem não tenha aparecido, por favor, refaça os passos anteriores, garantindo que seu ambiente esteja configurado corretamente.

 **AVISO**

Ao observar a mensagem mencionada acima, você pode prosseguir para a próxima etapa da configuração. Caso contrário, repita os passos ou revise a configuração do seu ambiente para garantir que esteja devidamente configurado.

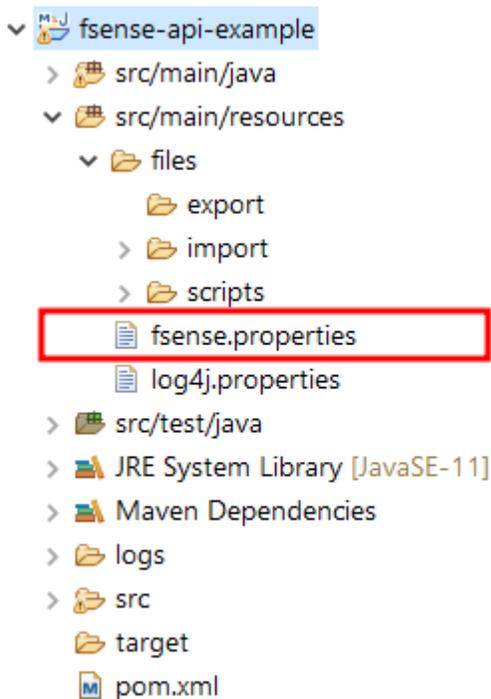
# Configurações da Aplicação

## Configuração do arquivo Properties

A configuração correta do arquivo `fsense.properties` é essencial para o funcionamento da aplicação exemplo de integração.

## Localização do Arquivo

O arquivo `fsense.properties` pode ser localizado no diretório `src/main/resources` da aplicação como no exemplo abaixo:



*Eclipse - Arquivo fsense.properties*

## Configuração das Credenciais da Organização

### Obtendo as Credenciais

- acesse o site <http://app.fsense.com> e faça login na sua conta.

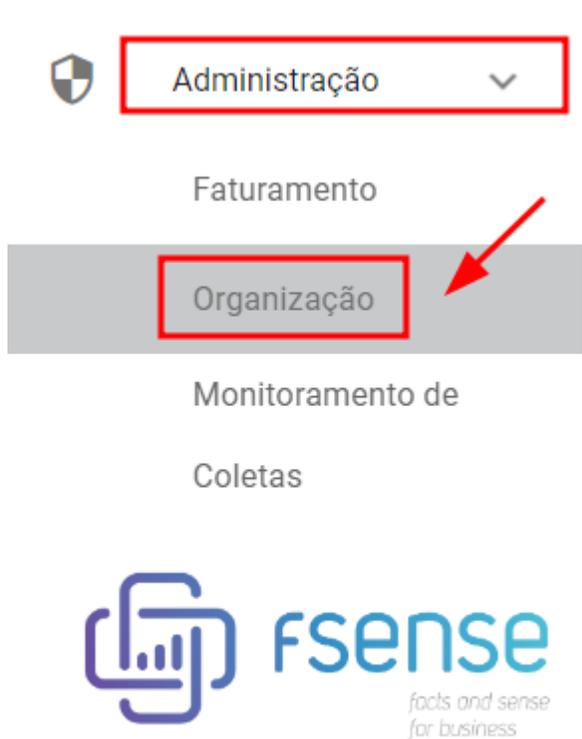


Figura 1 - fSense - Administração - Organização

- No menu lateral, vá para **Administração > Organização**. Na aba Organização está localizado o card Integrações onde estão as credenciais da organização.

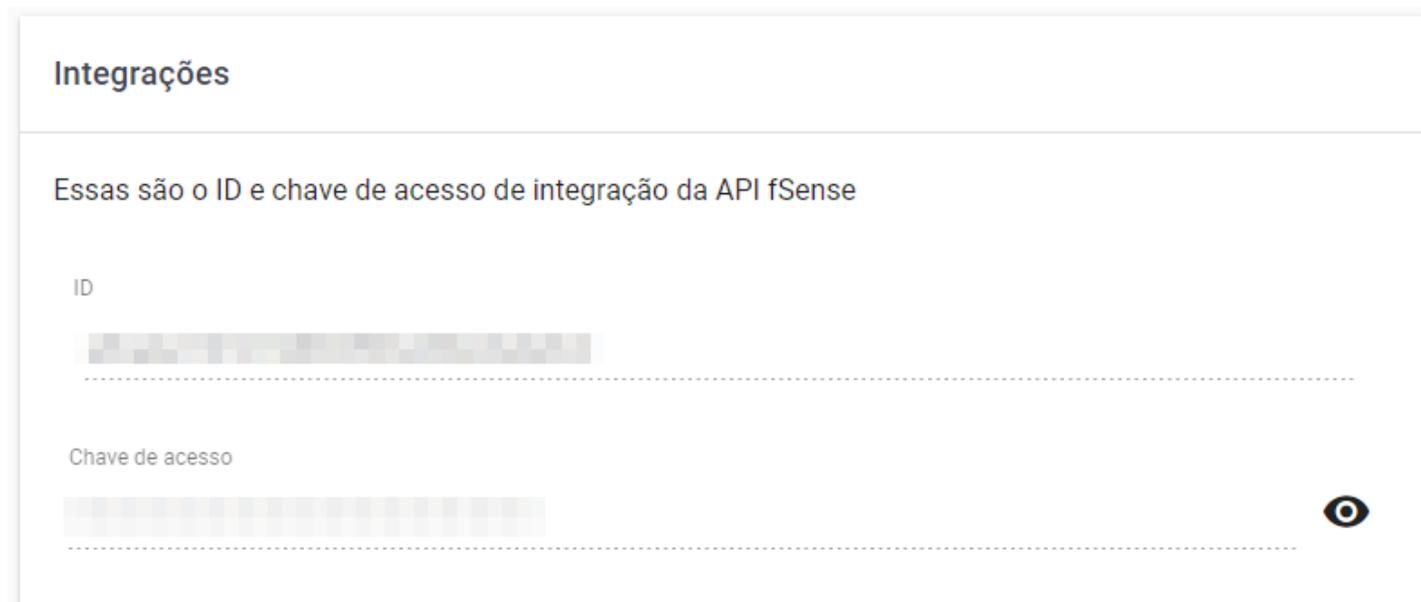


Figura 2 - fSense - Administração - Organização - Credenciais

## Configurando as Credenciais

Substitua o valores `<you-client-id>` e `<you-client-secret>` pelas as credenciais da sua organização obtidas no painel de Gestão Online fSense. Informe os valores sem aspas e sem espaços.

```
#CREDENCIAIS DA ORGANIZAÇÃO
client.id=<your-client-id>
client.secret=<your-client-secret>
```

## Definir a URL de domínio

Em seguida, é preciso definir a URL base da API de integração e a versão da API que será utilizada.

## Configurando a URL da API

- Acesse o site [API de Integração do fsense](#), para verificar a URL e a versão atual da documentação.



Figura 3 - API de Integração do fSense

A configuração é feita com base neste modelo, a `URL base`, seguido da `versão da API`.



Figura 4 - API de Integração do fSense - URL - versão da API

- Preencha as propriedades do arquivo `fsense.properties` como a referência abaixo:

```
#URL DA API DE INTEGRAÇÃO - [https://api.fsense.io, https://api.fsense.com]
api.base.url= http://api.fsense.com
#versão da api - ["v1.0", "v2.0", "v3.0"]
api.version=v3.0
```

## Definir o Serviço

Para que a aplicação funcione corretamente, é necessário definir o serviço a ser executado.

Se executarmos a aplicação, uma lista de serviços disponíveis será exibida se nenhum serviço for informado.

```
=====
SELECTED SERVICE []
=====
No service found for informed name:
=====
Available services:
[LOGIN_HISTORY_SERVICE]
[WORKDAY_OVERRIDE_SERVICE]
[PERSON_SERVICE]
[ACTIVITY_SERVICE]
[WORKDAY_SERVICE]
[CATEGORY_SERVICE]
[EVENT_SERVICE]
[DATABASE_TEST_SERVICE]
[TEAM_SERVICE]
[WORKDAY_PROROGATION_SERVICE]
[HOLIDAY_SERVICE]
[AUTH_SERVICE]
=====
```

Por exemplo, para selecionar o serviço `AUTH_SERVICE`, que acessa o endpoint `oauth/token` para obter o token de acesso, adicione a seguinte configuração na propriedade `service.name`:

```
#SERVICIO A SER EXECUTADO
```

```
service.name=AUTH_SERVICE
```

Ao definir o serviço, você deve definir o método a ser utilizado;

```
=====
SELECTED SERVICE [AUTH_SERVICE]
=====
SELECTED METHOD []
No method has been selected.
=====
Available methods:
[GET_ACCESS_TOKEN]
=====
End of execution.
```

## Definir o Método

Para obter o token de acesso, defina o método `GET_ACCESS_TOKEN`:

```
#SERVICO A SER EXECUTADO
service.name=AUTH_SERVICE

#METODO DO SERVIÇO A SER EXECUTADO
service.method=GET_ACCESS_TOKEN
```

## Execução e Resultados

O Serviço e o método estão definidos, então, se tudo estiver configurado corretamente, o console da aplicação exibirá a resposta da API.

- **RESPOSTA COM SUCESSO**

```
=====
SELECTED SERVICE [AUTH_SERVICE]
=====
SELECTED METHOD [GET_ACCESS_TOKEN]
checking if token is valid?: false
=====
```

```
Starting authentication process
Getting access token
Token received.
1a1974a5-16ab-41a8-ad03-d25b1ff4d435
=====
AUTHENTICATION SUCCESS
=====
End of execution.
```

- **RESPOSTA COM ERRO**

Caso as credenciais foram inseridas incorretas, o console exibirá uma mensagem vinda da API com o erro.

```
=====
SELECTED SERVICE [AUTH_SERVICE]
=====
SELECTED METHOD [GET_ACCESS_TOKEN]
checking if token is valid?: false
=====
Starting authentication process
error: invalid_client
error description: Bad client credentials
Invalid credentials
=====
AUTHENTICATION FAILED
=====
End of execution.
```

Certifique-se seguir cada etapa e validar as configurações para que a aplicação funcione conforme o esperado.

# Serviço de Autenticação

O Serviço de Autenticação é responsável por garantir que o usuário da aplicação esteja autenticado corretamente na API de integração do fSense. Este processo é crucial pois assegura que toda a comunicação com a API seja feita de forma segura.

## Processo de Autenticação

Antes da execução de qualquer outro serviço, o Serviço de Autenticação é executado internamente para autenticar o usuário e obter os recursos necessários para a integração com a API. Dessa forma, todas as chamadas subsequentes sejam feitas com um token de acesso válido.

## Obter Token de Acesso

### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `AUTH_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=AUTH_SERVICE
```

### Método do Serviço

Configure o método do serviço para `GET_ACCESS_TOKEN` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=GET_ACCESS_TOKEN
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará um requisição HTTP à API fSense para obter o token de acesso. Este método é somente para demonstração, pois ele é executado internamente na chamada de todos os demais serviços.

```
=====
SELECTED SERVICE [AUTH_SERVICE]
=====
SELECTED METHOD [GET_ACCESS_TOKEN]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1a1974a5-16ab-41a8-ad03-d25b1ff4d435
=====
AUTHENTICATION SUCCESS
=====
End of execution.
```

Caso as credenciais foram inseridas incorretas, o console exibirá uma mensagem vinda da API com o erro.

```
=====
SELECTED SERVICE [AUTH_SERVICE]
=====
SELECTED METHOD [GET_ACCESS_TOKEN]
checking if token is valid?: false
=====
Starting authentication process
error: invalid_client
error description: Bad client credentials
Invalid credentials
=====
AUTHENTICATION FAILED
=====
End of execution.
```

# Serviço de Atividades

Serviço que permite a busca de atividades do usuário através dos dados coletados pelo fSense nas estações de trabalho.

## Exportar Atividades - Exemplo de Implementação CSV

Para exportar as atividades no formato **.csv**, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `ACTIVITY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=ACTIVITY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_ACTIVITIES` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_ACTIVITIES
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

É necessário definir um período de data para busca de atividades. Estes parâmetros podem ser definidos no arquivo `fsense.properties` através das propriedades `activity.initialDate` e `activity.finalDate`. Veja o exemplo abaixo para referência:

```
# INTERVALO DE DATAS [dd/MM/yyyy] PARA SERVIÇO DE ATIVIDADES
activity.initialDate = 04/12/2023
activity.finalDate = 07/12/2023
```

Também é possível opcionalmente especificar a hora inicial e hora final para a busca de atividades através das propriedades `activity.initialHour` e `activity.finalHour`.

```
# INTERVALO DE HORAS PARA SERVIÇO DE ATIVIDADES
# FAVOR PREENCHER NESTE FORMATO [HH:mm:ss]
# SE VAZIO SE CONSIDERA ENTRE 00:00:00 e 23:59:59
activity.initialHour = HH:mm:ss
activity.finalHour = HH:mm:ss
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter as atividades e gerará um arquivo CSV como nome `calendars_{ANO_MES_DIA}.csv`.

```
=====
SELECTED SERVICE [ACTIVITY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_ACTIVITIES]
checking if token is valid?: false
=====
Starting authentication process
```

```

Getting access token
Token received.
5ad5b6fc-76ad-4012-b8e6-da40f710a51a
=====
AUTHENTICATION SUCCESS
=====
Getting activities of day: 06-09-2024
Nº of records found: 63
Nº pages : 7
Getting activities 1/7
Getting activities 2/7
Getting activities 3/7
Getting activities 4/7
Getting activities 5/7
Getting activities 6/7
Getting activities 7/7
=====
GENERATING CSV FILE
=====
File generated: activities_2024-09-06.csv
Status: saved successfully
File saved in path: ./src/main/resources/files/export/activities_2024-09-
06.csv
=====
EXPORT ACTIVITIES FINISHED SUCCESSFULLY
=====
End of execution.

```

Os arquivos são armazenados no diretório **src/main/resources/files/export/**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as atividades exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```

'date','application','login','processName','evtDurationMilliseconds','host','cate
'2024-09-06','Google Chrome','maria_silva','chrome','4441909','localhost:3000','B
'2024-09-06','Google
Chrome','maria_silva','chrome','1678839','cogna.lms.ecoedu.com.br','U','Unclassif
'2024-09-06','Google Chrome','maria_silva','chrome','781753','bitbucket.org','B',
'2024-09-06','Google Chrome','maria_silva','chrome','121723','app.fsense.io','B',
'2024-09-06','Google Chrome','maria_silva','chrome','34988','_SEARCHING_','B','Bu
'2024-09-06','Google Chrome','maria_silva','chrome','214976','api.fsense.com','B'

```

```
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '159727', 'login.live.com', 'B',  
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '129588', 'app.fsense.com', 'B',  
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '70800', 'instagram.com', 'A', 'I',  
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '12891', 'login.microsoftonline.com', 'Use',  
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '77169', 'portal.azure.com', 'B',  
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '144426', 'github.com', 'B', 'Bu',  
'2024-09-06', 'Google Chrome', 'maria_silva', 'chrome', '18598', 'id.atlassian.com', 'B
```

## Exportar Atividades - Exemplo de Implementação de Banco de Dados

Para exportar as atividades na organização, fazendo a persistência no banco de dados, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `ACTIVITY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=ACTIVITY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_ACTIVITIES` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_ACTIVITIES
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO  
operation.type=database
```

### DICA

Certifique-se de que o banco de dados está devidamente configurado e a tabela previamente criada. Certifique-se de que o banco de dados está devidamente configurado e a tabela previamente criada.

Se tiver dúvidas, consulte a seção [Integração com Banco de Dados](#).

## Parâmetros de Exportação

É necessário definir um período de data para busca de atividades. Estes parâmetros podem ser definidos no arquivo `fsense.properties` através das propriedades `activity.initialDate` e `activity.finalDate`. Veja o exemplo abaixo para referência:

```
# INTERVALO DE DATAS [dd/MM/yyyy] PARA SERVIÇO DE ATIVIDADES  
activity.initialDate = 04/12/2023  
activity.finalDate = 07/12/2023
```

Também é possível opcionalmente especificar a hora inicial e hora final para a busca de atividades através das propriedades `activity.initialHour` e `activity.finalHour`.

```
# INTERVALO DE HORAS PARA SERVIÇO DE ATIVIDADES  
# FAVOR PREENCHER NESTE FORMATO [HH:mm:ss]  
# SE VAZIO SE CONSIDERA ENTRE 00:00:00 e 23:59:59  
activity.initialHour = HH:mm:ss  
activity.finalHour = HH:mm:ss
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter as atividades e em seguida, fará a persistência dos dados no banco.

```
=====
SELECTED SERVICE [ACTIVITY_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_ACTIVITIES]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5ad5b6fc-76ad-4012-b8e6-da40f710a51a
=====
AUTHENTICATION SUCCESS
=====
Getting activities of day: 06-09-2024
Nº of records found: 63
Nº pages : 7
Getting activities 1/7
Getting activities 2/7
Getting activities 3/7
Getting activities 4/7
Getting activities 5/7
Getting activities 6/7
Getting activities 7/7
=====
Open database connection
Saving on tb_activity
=====
Update Count: 63
Database connection close
=====
SAVING DATABASE ACTIVITIES FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_activity**.

```
SELECT * FROM tb_activity;
```

A tabela terá uma estrutura semelhante à esta:

date	application	login	processName	evtDurationMilliseconds	host
2024-09-06	Google Chrome	maria_silva	chrome	4441909	localho:
2024-09-06	Google Chrome	maria_silva	chrome	1678839	cogna.lr
2024-09-06	Google Chrome	maria_silva	chrome	781753	bitbuck
2024-09-06	Google Chrome	maria_silva	chrome	121723	app.fsei
2024-09-06	Google Chrome	maria_silva	chrome	34988	SEARCH/
2024-09-06	Google Chrome	maria_silva	chrome	214976	api.fsen
2024-09-	Google Chrome	maria_silva	chrome	159727	login.liv

<b>date</b>	<b>application</b>	<b>login</b>	<b>processName</b>	<b>evtDurationMilliseconds</b>	<b>host</b>
06					
2024-09-06	Google Chrome	maria_silva	chrome	129588	app.fse
2024-09-06	Google Chrome	maria_silva	chrome	70800	instagra
2024-09-06	Google Chrome	maria_silva	chrome	12891	login.mi
2024-09-06	Google Chrome	maria_silva	chrome	77169	portal.a
2024-09-06	Google Chrome	maria_silva	chrome	144426	github.c
2024-09-06	Google Chrome	maria_silva	chrome	18598	id.atlas:

# Serviço de Eventos

Serviço que permite a busca dos eventos coletados pelo fSense nas estações de trabalho.

## Exportar Eventos - Exemplo de Implementação CSV

Para exportar os eventos no formato **.csv**, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `EVENT_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=EVENT_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_EVENTS` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_EVENTS
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
```

```
operation.type=csv
```

## Parâmetros de Exportação

É necessário definir um período de data para busca de eventos. Estes parâmetros podem ser definidos no arquivo `fsense.properties` através das propriedades `event.initialDate` e `event.finalDate`. Veja o exemplo abaixo para referência:

```
# INTERVALO DE DATAS [dd/MM/yyyy] PARA SERVIÇO DE EVENTOS
event.initialDate = 04/12/2023
event.finalDate = 07/12/2023
```

Também é possível opcionalmente especificar a hora inicial e hora final para a busca de eventos através das propriedades `event.initialHour` e `event.finalHour`.

```
# INTERVALO DE HORAS PARA SERVIÇO DE EVENTOS
# FAVOR PREENCHER NESTE FORMATO [HH:mm:ss]
# SE VAZIO SE CONSIDERA ENTRE 00:00:00 e 23:59:59
event.initialHour =
event.finalHour =
```

É possível buscar eventos com base no horário em que ocorreram na máquina do usuário, ou com base que foi dada entrada no sistema. Se desejar que os eventos sejam buscados com base no momento que ocorreram na máquina do usuário, defina a propriedade `event.findByCreationDate` como `true`.

```
# SE BUSCA POR DATA DE CRIAÇÃO SELECIONE [true/false]
# POR PADRAO FALSE
event.findByCreationDate=false
```

Também é possível filtrar por um usuário específico. Para isso basta informar na propriedade `event.login` o usuário que deseja filtrar.

```
#FILTRA POR LOGIN DO USUÁRIO. Exemplo: event.login=maria_silva
event.login=
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter os eventos e gerará um arquivo CSV como nome `events_{ANO_MES_DIA}.csv`.

```
=====
SELECTED SERVICE [EVENT_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_EVENTS]
Getting events of day: 04-12-2023
find by creation date: false
token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
ba88697a-1c96-4a31-93b7-93cfae727ccb
=====
AUTHENTICATION SUCCESSFULLY
=====
Nº of records found: 1662
Nº pages found: 1
Get events page 1/1
=====
File generated: events_2023-12-04.csv
Status: successfully saved
File saved in path: ./src/main/resources/files/export/events_2023-12-04.csv
=====
EXPORT EVENTS FINISHED SUCCESSFULLY
=====
End of execution.
```

Os arquivos são armazenados no diretório **`src/main/resources/files/export/`**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar os eventos exportados. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'application','login','processName','evtDurationMilliseconds','host','url','machin
'Lock Time','maria_silva','Lock Time','47089016','_SEARCHING_','','N-TEC-MARIA',':
'Lock Time','maria_silva','Lock Time','1197831','_SEARCHING_','','N-TEC-MARIA','1
'Google Chrome','maria_silva','chrome','1399','_SEARCHING_','','N-TEC-MARIA','172
'Java(TM) Platform SE binary','maria_silva','jabswitch','67','','','N-TEC-MARIA',
'Google Chrome','maria_silva','chrome','3964','_SEARCHING_','','N-TEC-MARIA','172
'Microsoft Teams','maria_silva','ms-teams','369994','','','N-TEC-MARIA','172.25.1:
'Microsoft Edge WebView2','maria_silva','msedgewebview2','147','','','N-TEC-MARIA
'Microsoft Teams','maria_silva','ms-teams','548484','','','N-TEC-MARIA','172.25.1:
'Windows Explorer','maria_silva','explorer','416','','','N-TEC-MARIA','172.25.128
'Search application','maria_silva','SearchApp','4947','','','N-TEC-MARIA','172.25
05T16:40:41.668+00:00','2.0214271275520516','4.042854255104103','2','1'
'Microsoft Excel','maria_silva','EXCEL','15553','','','N-TEC-MARIA','172.25.128.1
'Idle Time','maria_silva','Idle Time','837122','','','N-TEC-MARIA','172.25.128.1'
```

## Exportar Eventos - Exemplo de Implementação de Banco de Dados

Para exportar os eventos fazendo a persistência no banco de dados, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `EVENT_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=EVENT_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_EVENTS` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_EVENTS
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=database
```

## Parâmetros de Exportação

É necessário definir um período de data para busca de eventos. Estes parâmetros podem ser definidos no arquivo `fsense.properties` através das propriedades `event.initialDate` e `event.finalDate`. Veja o exemplo abaixo para referência:

```
# INTERVALO DE DATAS [dd/MM/yyyy] PARA SERVIÇO DE EVENTOS
event.initialDate = 04/12/2023
event.finalDate = 07/12/2023
```

Também é possível opcionalmente especificar a hora inicial e hora final para a busca de eventos através das propriedades `event.initialHour` e `event.finalHour`.

```
# INTERVALO DE HORAS PARA SERVIÇO DE EVENTOS
# FAVOR PREENCHER NESTE FORMATO [HH:mm:ss]
# SE VAZIO SE CONSIDERA ENTRE 00:00:00 e 23:59:59
event.initialHour =
event.finalHour =
```

É possível buscar eventos com base no horário em que ocorreram na máquina do usuário, ou com base que foi dada entrada no sistema. Se desejar que os eventos sejam buscados com base no momento que ocorreram na máquina do usuário, marque esta opção como `true`.

```
# SE BUSCA POR DATA DE CRIAÇÃO SELECIONE [true/false]
# POR PADRÃO FALSE
event.findByCreationDate=false
```

Também é possível filtrar por um usuário específico. Para isso basta informar no parâmetro `event.login` o usuário que deseja filtrar.

```
#FILTRA POR LOGIN DO USUÁRIO. Exemplo: event.login=44944_wester
event.login=
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter os dados de eventos e em seguida fará a persistência dos dados no banco.

```
=====
SELECTED SERVICE [EVENT_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_EVENTS]
Getting events of day: 04-12-2023
find by creation date: false
token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
ba88697a-1c96-4a31-93b7-93cfae727ccb
=====
AUTHENTICATION SUCCESSFULLY
=====
Nº of records found: 1662
Nº pages found: 1
Get events page 1/1
=====
SAVING ON DATABASE
=====
Open database connection
```

```
Saving on tb_event
Update Count: 1662
Database connection close
```

```
=====
SAVING DATABASE EVENTS FINISHED SUCCESSFULLY
=====
```

```
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_event**.

```
SELECT * FROM tb_event;
```

A tabela terá uma estrutura semelhante à esta:

application	login	processName	evtDurationMilliseconds	host
Lock Time	maria_silva	Lock Time	47089016	SEARCHING
Lock Time	maria_silva	Lock Time	1197831	SEARCHING
Google Chrome	maria_silva	chrome	1399	SEARCHING
Java(TM) Platform SE binary	maria_silva	jabswitch	67	
Windows Explorer	maria_silva	explorer	1	

<b>application</b>	<b>login</b>	<b>processName</b>	<b>evtDurationMilliseconds</b>	<b>host</b>
Microsoft Teams	maria_silva	ms-teams	369994	
Microsoft Edge WebView2	maria_silva	msedgewebview2	147	
Microsoft Teams	maria_silva	ms-teams	548484	

# Serviço de Históricos de Login

Serviço responsável pela busca de logins no sistema fSense web da organização.

## Exportar Histórico de Login - Exemplo de Implementação CSV

Para exportar o histórico de login no formato CSV, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `LOGIN_HISTORY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=LOGIN_HISTORY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_LOGIN_HISTORY` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_LOGIN_HISTORY
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

É possível informar parâmetros opcionais, como o ID de um pessoa cadastrada ou um período de datas que deseja buscar o histórico de logins efetuados no painel de Gestão Online fSense. Preencha esses campos conforme necessário.

```
# PARÂMETROS OPCIONAIS PARA EXPORTAÇÃO DE LOGIN
login.history.person.id=
#[dd/MM/yyyy]
login.history.start=
#[dd/MM/yyyy]
login.history.end=
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter o histórico de login e gerará um arquivo CSV como nome `login_history.csv`.

```
=====
SELECTED SERVICE [LOGIN_HISTORY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_LOGIN_HISTORY]
=====
Getting Login History
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
ba88697a-1c96-4a31-93b7-93cfae727ccb
=====
AUTHENTICATION SUCCESS
=====
```

```
Nº pages found : 20
Getting Login history page 1/20
Getting Login history page 2/20
Getting Login history page 3/20
Getting Login history page 4/20
Getting Login history page 5/20
Getting Login history page 6/20
Getting Login history page 7/20
Getting Login history page 8/20
Getting Login history page 9/20
Getting Login history page 10/20
Getting Login history page 11/20
Getting Login history page 12/20
Getting Login history page 13/20
Getting Login history page 14/20
Getting Login history page 15/20
Getting Login history page 16/20
Getting Login history page 17/20
Getting Login history page 18/20
Getting Login history page 19/20
Getting Login history page 20/20
Nº records found : 1000
=====
GENERATING CSV
=====
File Generated: login_history.csv
Status: saved successfully
File Saved in path: ./src/main/resources/files/export/login_history.csv
=====
EXPORT LOGIN HISTORY FINISHED SUCCESSFULLY
=====
End of execution.
```

Os arquivos são armazenados no diretório **src/main/resources/files/export/**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar o histórico de login exportado. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'name','email','date'
'ROOT','maria@gmail.com','2024-09-09T20:18:00.196+00:00'
'ROOT','maria@gmail.com','2024-09-09T20:17:43.622+00:00'
```

```
'ROOT','maria@gmail.com','2024-09-09T14:11:16.628+00:00'  
'ROOT','maria@gmail.com','2024-09-09T12:37:11.826+00:00'  
'ROOT','maria@gmail.com','2024-09-06T20:53:21.679+00:00'  
'ROOT','maria@gmail.com','2024-09-06T18:08:02.944+00:00'
```

## Exportar Histórico de Login - Exemplo de Implementação de Banco de Dados

Para exportar o histórico de login fazendo a persistência no banco de dados, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `LOGIN_HISTORY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=LOGIN_HISTORY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_LOGIN_HISTORY` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_HISTORY_SERVICE
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=DATABASE
```

## Parâmetros de Exportação

É possível informar parâmetros opcionais, como o ID de um pessoa cadastrada ou um período de datas que deseja buscar o histórico de logins efetuados no painel de Gestão Online fSense. Preencha esses campos conforme necessário.

```
# PARÂMETROS OPCIONAIS PARA EXPORTAÇÃO DE LOGIN
login.history.person.id=
#[dd/MM/yyyy]
login.history.start=
#[dd/MM/yyyy]
login.history.end=
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter o histórico de Login e em seguida fará a persistência dos dados no banco.

```
=====
SELECTED SERVICE [LOGIN_HISTORY_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_LOGIN_HISTORY]
=====
Getting Login History
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
ba88697a-1c96-4a31-93b7-93cfae727ccb
=====
AUTHENTICATION SUCCESS
=====
```

```
Nº pages found : 20
Getting Login history page 1/20
Getting Login history page 2/20
Getting Login history page 3/20
Getting Login history page 4/20
Getting Login history page 5/20
Getting Login history page 6/20
Getting Login history page 7/20
Getting Login history page 8/20
Getting Login history page 9/20
Getting Login history page 10/20
Getting Login history page 11/20
Getting Login history page 12/20
Getting Login history page 13/20
Getting Login history page 14/20
Getting Login history page 15/20
Getting Login history page 16/20
Getting Login history page 17/20
Getting Login history page 18/20
Getting Login history page 19/20
Getting Login history page 20/20
Nº records found : 1000
=====
SAVING ON DATABASE
=====
Open database connection
Saving on tb_login_history
Update Count: 1000
Database connection close
=====
SAVING DATABASE LOGIN HISTORY FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_login\_history**.

```
SELECT * FROM tb_login_history;
```

A tabela terá uma estrutura semelhante à esta:

<b>name</b>	<b>email</b>	<b>date</b>
ROOT	maria@gmail.com	2024-09-09T20:18:00.196+00:00
ROOT	maria@gmail.com	2024-09-09T20:17:43.622+00:00
ROOT	maria@gmail.com	2024-09-09T14:11:16.628+00:00
ROOT	maria@gmail.com	2024-09-09T12:37:11.826+00:00
ROOT	maria@gmail.com	2024-09-06T20:53:21.679+00:00
ROOT	maria@gmail.com	2024-09-06T18:08:02.944+00:00

# Serviço de Equipes

Serviço que permite internalizar a estrutura de equipes da sua organização.

## Exportar Equipes - Exemplo de Implementação CSV

Para exportar o as equipes cadastradas no fSense no formato CSV, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=TEAM_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_TEAMS` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_TEAMS
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
```

```
operation.type=csv
```

## Parâmetros de Exportação

É possível definir parâmetros opcionais na exportação de equipes como nome da equipe, código externo, identificador único. É possível definir através das respectivas propriedades `team.name`, `team.external.code` e `team.id`.

```
#PARÂMETROS PARA SERVIÇO DE BUSCA DE EQUIPES
#Exemplo:[Equipe Tecnologia da Informação]
team.name=
#Exemplo:[3b45fc4e-0ce6-42b5-acag-f35b6nmd0088]
team.external.code=
#PARÂMETRO PARA DEMAIS SERVIÇOS DE EQUIPES
#Exemplo:[4056]
team.id=
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará requisições HTTP à API fSense para obter os dados de equipes e gerará um arquivo CSV como nome `teams.csv`.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_TEAMS]
Export teams
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
8255196a-cec5-4843-b1f4-ca7cd6e28de8
=====
AUTHENTICATION SUCCESS
=====
TEAM DATA
```

```
=====
Nº teams found: 5
Nº pages found: 1
=====
GENERATING CSV
=====
Generated file: teams.csv
Status: save successfully.
File saved in path: ./src/main/resources/files/export/teams.csv
=====
EXPORT TEAMS FINISHED SUCCESSFULLY
=====
```

Os arquivos são armazenados no diretório **src/main/resources/files/export**

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as equipes exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','name','workdayId','teamParentId','teamParentName'
'4079','Tecnologia da Informação','2657','',''
'5799','fSense DEV','','4077','Tecnologia da Informação'
'18187','Comercial','2544','',''
'18960','Service Desk','3407','',''
'19482','Recursos Humanos','2757','',''
```

## Exportar Equipes - Exemplo de Implementação de Banco de Dados

Para exportar as equipes no fSense fazendo a persistência no banco de dados, podemos fazer da seguinte maneira.

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=TEAM_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_TEAMS` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_TEAMS
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=DATABASE
```

## Parâmetros de Exportação

É possível definir parâmetros opcionais na exportação de equipes como nome da equipe, código externo, identificador único. É possível definir através das respectivas propriedades `team.name`, `team.external.code` e `team.id`.

```
#PARÂMETROS PARA SERVIÇO DE BUSCA DE EQUIPES
#Exemplo:[Equipe Tecnologia da Informação]
team.name=
#Exemplo:[3b45fc4e-0ce6-42b5-acag-f35b6nmd0088]
team.external.code=
#PARÂMETRO PARA DEMAIS SERVIÇOS DE EQUIPES
#Exemplo:[4056]
team.id=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_TEAMS]
Export teams
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
8255196a-cec5-4843-b1f4-ca7cd6e28de8
=====
AUTHENTICATION SUCCESS
=====
TEAM DATA
=====
Nº teams found: 5
Nº pages found: 1
=====
SAVING ON DATABASE
=====
Open database connection
Saving on tb_team
Update Count: 5
Database connection close
=====
SAVING DATABASE TEAMS FINISHED SUCCESSFULLY
=====
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_team**.

```
SELECT * FROM tb_team;
```

A tabela terá uma estrutura semelhante à esta:

id	name	workdayId	teamParentId	teamParentName
4079	Tecnologia da Informação	2657		
5799	fSense DEV		4077	Tecnologia da Informação
18187	Comercial	2544		
18960	Service Desk	3407		
19482	Recursos Humanos	2757		

## Importar Equipes

Para importar equipes para o fSense a partir de um arquivo CSV, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=TEAM_SERVICE
```

### Método do Serviço

Configure o método do serviço para `IMPORT_TEAM` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=IMPORT_TEAM
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário especificar o caminho do arquivo CSV contendo as informações das equipes a serem importadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório `src/main/resources/files/import/team_service_import_team.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações das equipes a serem importadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = team_service_import_team.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO  
csv.delimiter  
csv.delimiter = ,  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO
```

```
csv.text.qualifier
csv.text.qualifier = '
```

### **NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Exemplo de Arquivo de Importação de Equipe

```
'id','name','workdayId','teamParentId','teamParentName'
','Nova Equipe 1','2751','',''
','Nova Equipe 2','','',''
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
SELECTED METHOD [IMPORT_TEAM]
=====
OPERATION TYPE: CSV
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1b7749ad-2cbf-44c3-9138-766b101f9979
=====
AUTHENTICATION SUCCESS
=====
Team Data:
Team name: Teste Nova Equipe 38
Trying insert a new Team
team imported successfully
Team : Teste Nova Equipe 38 imported successfully
=====
```

```
Team Data:
Team name: Teste Nova Equipe 39
Trying insert a new Team
team imported successfully
Team : Teste Nova Equipe 39 imported successfully
=====
Team Data:
Team name: Teste Nova Equipe 40
Trying insert a new Team
team imported successfully
Team : Teste Nova Equipe 40 imported successfully
=====
RESULT OF IMPORT OPERATION
=====
Total of requests made 3
Total of requests successfully: 3
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará as requisições Http para a API fSense e internalizar as informações das equipes para o fSense.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipes** para verificar se as equipes foram inseridas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Atualizar Equipes por ID

Para atualizar equipes cadastradas no fSense a partir de um arquivo CSV, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=TEAM_SERVICE
```

## Método do Serviço

Configure o método do serviço para `UPDATE_TEAM_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=UPDATE_TEAM_BY_ID
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

Para atualizar uma equipes no fSense, o processo é semelhante ao de inserir.

É necessário especificar o caminho do arquivo CSV contendo as informações das equipes a serem importadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório `src/main/resources/files/import/team_service_update_team.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações dos calendários a serem importados.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = team_service_update_team.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO  
csv.delimiter  
csv.delimiter = ,  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO  
csv.text.qualifier  
csv.text.qualifier = '
```

### **NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====  
SELECTED SERVICE [TEAM_SERVICE]  
=====  
SELECTED METHOD [IMPORT_TEAM]  
=====  
OPERATION TYPE: CSV  
=====  
checking if token is valid?: false  
=====
```

```
Starting authentication process
Getting access token
Token received.
1b7749ad-2cbf-44c3-9138-766b101f9979
=====
AUTHENTICATION SUCCESS
=====
[Teste Nova Equipe 38 Atualizada, Teste Nova Equipe 39 Atualizada, Teste
Nova Equipe 40 Atualizada]
=====
Team Data:
Team name: Teste Nova Equipe 38 Atualizada
Team id: 19448 updated successfully
Team : Teste Nova Equipe 38 Atualizada updated successfully
=====
Team Data:
Team name: Teste Nova Equipe 39 Atualizada
Team id: 19449 updated successfully
Team : Teste Nova Equipe 39 Atualizada updated successfully
=====
Team Data:
Team name: Teste Nova Equipe 40 Atualizada
Team id: 19450 updated successfully
Team : Teste Nova Equipe 40 Atualizada updated successfully
=====
RESULT OF UPDATE OPERATION
=====
Total of requests made 3
Total of requests successfully: 3
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará as requisições Http para a API fSense para atualizar as informações das equipes no fSense.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipes** para verificar se as equipes foram atualizadas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Deletar Equipes por ID

Para remover uma equipe previamente cadastrada no fSense, podemos fazer os seguintes passos:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=TEAM_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `DELETE_TEAM_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=DELETE_TEAM_BY_ID
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Definição de Parâmetros de Remoção

Para remover uma ou mais equipes, é necessário informar o identificador único da equipe que deseja remover através da propriedade `team.ids`. É possível informar um ou mais ids, basta informá-los separados por vírgula.

```
##ID DAS EQUIPES PARA REMOVER/EXPORTAR DETALHES
team.ids=19448, 19449, 19450
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [DELETE_TEAM_BY_ID]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
Teams selected to delete:
[19448, 19449, 19450]
TEAM ID TO BE REMOVED: 19448 - 1/3
Team id 19448 delete successfully
=====
TEAM ID TO BE REMOVED: 19449 - 2/3
Team id 19449 delete successfully
=====
TEAM ID TO BE REMOVED: 19450 - 3/3
Team id 19450 delete successfully
=====
RESULT OF DELETE OPERATION
=====
```

```
Total of requests made 3
Total of requests successfully: 3
Total of requests failed: 0
```

```
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipes** para verificar se as equipes foram removidas corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Exportar Detalhes de Equipes

Para exportar informações detalhadas com informações de líderes e membros de uma equipe cadastrada no fSense podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=TEAM_SERVICE
```

- Definir o método **EXPORT\_TEAM\_DETAILS\_BY\_ID**

### Método do Serviço

Configure o método do serviço para `UPDATE_TEAM_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_TEAM_DETAILS_BY_ID
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

Para exportar equipes com informações detalhadas no fSense, é necessário informar o identificador que deseja remover através da propriedade `team.ids`. É possível informar um ou mais ids, basta informá-los separados por vírgula.

```
# ID DAS EQUIPES PARA REMOVER/EXPORTAR DETALHES
team.ids=19446
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_TEAM_DETAILS_BY_ID]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
```

```
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
GENERATING CSV
=====
Generated file: team_details_id_19146.csv
Status: save successfully.
File saved in path:
./src/main/resources/files/export/team_details_id_19146.csv
=====
EXPORT TEAM DETAILS BY ID FINISHED SUCCESSFULLY
=====
```

A aplicação fará requisições para a API fSense para obter as informações detalhadas das Equipes informadas. Posteriormente, serão gerados arquivos CSV.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar a exportação das equipes com informações detalhadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'teamId','teamName','id','name','email','deviceIdentifier','isLeader'
'19146','Tecnologia da
Informação','6755523','usuario','usuario@fsense.com','usuario@fsense.com','false'
'19146','Tecnologia da
Informação','6702343','maria_silva','maria.silva@fsense','','false'
'19146','','6665143','ROOT','root@fsense.com','','true'
```

## Adicionar Membros

Para adicionar membros em uma equipe, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=TEAM_SERVICE
```

## Método do Serviço

Configure o método do serviço para `MEMBERSHIP_ADDITION` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=MEMBERSHIP_ADDITION
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

É necessário definir o id da Equipe que deseja adicionar membros, através da propriedade `team.id`, em seguida é necessário definir o id dos pessoas que deseja adicionar à equipe. Se necessário, execute antes o serviço `PERSON_SERVICE` junto com o método `EXPORT_PERSONS`, para obter os ids das pessoas que deseja adicionar na equipe.

Como exemplo, vamos utilizar esta equipe

```
#PARÂMETRO PARA DEMAIS SERVIÇOS DE EQUIPES
team.id=18787
```

```
#ID MEMBROS PARA ADICIONAR/REMOVER DA EQUIPE
```

```
team.member.ids=6757359, 6757360, 6757361, 6757362, 6757363
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [MEMBERSHIP_ADDITION]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
ADD MEMBERSHIP ON TEAM ID: 18187
ID OF MEMBERS SELECTED TO ADD INTO TEAM:
=====
[6757359, 6757360, 6757361, 6757362, 6757363]
=====
Member added into team successfully
=====
End of execution.
```

A aplicação realizará requisições Http para a API fSense e adicionar os membros à equipe no sistema.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipe** para verificar se as pessoas foram adicionadas à equipe corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Remover Membros

Para remover membros de uma equipe, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=TEAM_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `MEMBERSHIP_REMOVAL` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=MEMBERSHIP_REMOVAL
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

#### Parâmetros de Exportação

É necessário definir o id da Equipe que deseja adicionar membros, através da propriedade `team.id`, em seguida é necessário definir o id dos pessoas que deseja adicionar à equipe. Se necessário, execute antes o serviço `PERSON_SERVICE` junto com o método `EXPORT_PERSONS`, para obter os ids das pessoas que deseja adicionar na equipe.

Como exemplo, vamos utilizar esta equipe

```
#PARÂMETRO PARA DEMAIS SERVIÇOS DE EQUIPES  
team.id=18787
```

```
#ID MEMBROS PARA ADICIONAR/REMOVER DA EQUIPE  
team.member.ids=6757359, 6757360, 6757361, 6757362, 6757363
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [MEMBERSHIP_REMOVAL]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
REMOVE MEMBERSHIPS ON TEAM ID: 18187
ID OF MEMBERS SELECTED TO REMOVE INTO TEAM:
=====
[6757359, 6757360, 6757361, 6757362, 6757363]
=====
Members removed from team successfully
```

```
=====  
End of execution.
```

A aplicação realizará requisições Http para a API fSense e remover os membros à equipe no sistema.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipe** para verificar se as pessoas foram removidas à equipe corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Adicionar Líderes

Para adicionar líderes em uma equipe, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=TEAM_SERVICE
```

### Método do Serviço

Configure o método do serviço para `LEADER_ADDITION` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=LEADER_ADDITION
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

É necessário definir o id da Equipe que deseja adicionar líderes, através da propriedade `team.id`, em seguida é necessário definir o id dos pessoas que deseja adicionar à equipe como líder. Se necessário, execute antes o serviço `PERSON_SERVICE` junto com o método `EXPORT_PERSONS`, para obter os ids das pessoas que deseja adicionar como líder na equipe.

Como exemplo, vamos utilizar esta equipe

```
#PARÂMETRO PARA DEMAIS SERVIÇOS DE EQUIPES
team.id=18787
```

```
#ID MEMBROS PARA ADICIONAR/REMOVER DA EQUIPE
team.leader.ids=6757359, 6757360, 6757361, 6757362, 6757363
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [LEADER_ADDITION]
=====
checking if token is valid?: false
=====
```

```
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
ADD LEADER ON TEAM ID: 18187
ID OF LEADERS SELECTED TO ADD INTO TEAM:
=====
[6757359, 6757360, 6757361, 6757362, 6757363]
=====
Leaders added into team successfully
=====
End of execution.
```

A aplicação realizará requisições Http para a API fSense e adicionar os líderes à equipe no sistema.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipe** para verificar se as pessoas foram adicionadas à equipe como líderes corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Remover Líderes

Para remover líderes de uma equipe, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `TEAM_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=TEAM_SERVICE
```

## Método do Serviço

Configure o método do serviço para `LEADER_REMOVAL` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=LEADER_REMOVAL
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

É necessário definir o id da Equipe que deseja remover líderes, através da propriedade `team.id`, em seguida é necessário definir o id dos líderes que deseja remover da equipe. Se necessário, execute antes o serviço `PERSON_SERVICE` junto com o método `EXPORT_PERSONS`, para obter os ids dos líderes que deseja remover da equipe.

Como exemplo, vamos utilizar esta equipe

```
#PARÂMETRO PARA DEMAIS SERVIÇOS DE EQUIPES
team.id=18787
```

```
#ID MEMBROS PARA ADICIONAR/REMOVER DA EQUIPE
team.member.ids=6757359, 6757360, 6757361, 6757362, 6757363
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [TEAM_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [LEADER_REMOVAL]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
REMOVE LEADERS ON TEAM ID: 18187
ID OF LEADERS SELECTED TO ADD INTO TEAM:
=====
[6757359, 6757360, 6757361, 6757362, 6757363]
=====
Leaders removed into team successfully
=====
End of execution.
```

A aplicação realizará requisições Http para a API fSense e remover os líderes à equipe no sistema.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipe** para verificar se os líderes foram removidos da equipe corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

# Serviço de Pessoas

Serviço responsável por fazer a busca de categorias de uma aplicação por pessoa ou por equipe.

## Exportar Pessoas - Exemplo de Implementação CSV

Para exportar as pessoas cadastradas na organização em formato CSV, podemos fazer da seguinte forma:

### Configuração do Arquivo Properties.

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `PERSON_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=PERSON_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_PERSONS` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_PERSONS
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

Opcionalmente, você pode definir os seguintes parâmetros:

- **name:** Nome da pessoa
- **email:** Email da pessoa
- **externalCode:** Código da pessoa fornecido pela organização
- **deviceIdentifier:** Identificador utilizado para identificar uma pessoa no fSense (por exemplo, nome de usuário)
- **teamId:** Identificador único gerado pelo fSense para uma equipe

```
# PARÂMETROS PARA SERVIÇO DE BUSCA PERSONALIZADA POR PESSOAS  
# Exemplo: [ROOT]  
person.name=  
# Exemplo: [person@fsense.com]  
person.email=  
# Exemplo: [3b45fc4e-0ce6-42b5-acag-f35b6nmd0088]  
person.externalCode=  
# Exemplo: [person.fsense]  
person.deviceIdentifier=  
# Exemplo: [4056]  
person.teamId=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====  
SELECTED SERVICE [PERSON_SERVICE]  
=====  
OPERATION TYPE: CSV  
=====  
SELECTED METHOD [EXPORT_PERSONS]  
Exporting persons  
checking if token is valid?: false
```

```

=====
Starting authentication process
Getting access token
Token received.
5839dd80-1444-4a8e-a2c2-71d5f18ba8d0
=====
AUTHENTICATION SUCCESS
=====
Nº pages: 23
Nº peoples found: 221
Getting peoples page: 1/23
Getting peoples page: 2/23
...
Getting peoples page: 23/23
=====
GENERATING CSV
=====
Generated file: persons.csv
Status: saved successfully
File saved in path: ./src/main/resources/files/export/persons.csv
=====
EXPORT PERSONS FINISHED SUCCESSFULLY
=====

```

Os arquivos são armazenados no diretório **src/main/resources/files/export**

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as equipes exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```

'id','name','email','deviceIdentifier','teamId','teamName','workdayId','canDelete
'6665124','ROOT','root@fsense.com','','','','false','false',''
'6665126','Maria','maria.silva@fsense.com.br','maria_silva','','','2658','false',
'6719848','Lider Equipe','','lider_equipe','','','true','false',''
'6744454','Wester','','44944_wester','19482','Equipe fSense DEV','2751','false',
'6755534','usuario','usuario@fsense.com','usuario@fsense.com','4079','Tecnologia
Informação','74','false','false',''

```

# Exportar Pessoas - Exemplo de Implementação de Banco de Dados

Para exportar as pessoas cadastradas na organização fazendo a persistência de dados no banco, podemos fazer da seguinte forma:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `PERSON_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=PERSON_SERVICE
```

### Método do Serviço

Configure o método do serviço para `EXPORT_PERSONS` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_PERSONS
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO  
operation.type=database
```

## Execução da Aplicação

Após definidos os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [PERSON_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_PERSONS]
Exporting persons
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5839dd80-1444-4a8e-a2c2-71d5f18ba8d0
=====
AUTHENTICATION SUCCESS
=====
Nº pages: 23
Nº peoples found: 221
Getting peoples page: 1/23
Getting peoples page: 2/23
...
Getting peoples page: 23/23
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_person
UpdateCount: 221
Database connection close
=====
SAVING DATABASE PERSONS FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_person**.

```
SELECT * FROM tb_person;
```

A tabela terá uma estrutura semelhante à esta:

id	name	email	deviceIdentifier	teamId	teamName
6665124	ROOT	root@fsense.com			
6665126	Maria	maria.silva@fsense.com.br	maria_silva		
6719848	Lider Equipe		lider_equipe		
6744454	Wester		44944_wester	19482	Equipe fSense DEV
6755534	usuario	usuario@fsense.com	usuario@fsense.com	4079	Tecnologia da Informação

## Importar Pessoas

Para importar pessoas para o fSense partir de um arquivo CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo as propriedades `service.name` para `PERSON_SERVICE`.

```
# SERVICIO A SER EXECUTADO
```

```
service.name=PERSON_SERVICE
```

## Método do Serviço

Configure o método do serviço para `IMPORT_PERSON` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=IMPORT_PERSON
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário especificar o caminho do arquivo CSV contendo as informações das pessoas a serem importadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório `src/main/resources/files/import/person_service_import_person.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações das pessoas a serem importadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]
file.name = person_service_import_person.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO
csv.delimiter
csv.delimiter = ,
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO
csv.text.qualifier
csv.text.qualifier = '
```

### **NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [PERSON_SERVICE]
=====
SELECTED METHOD [UPDATE_PERSON]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1b7749ad-2cbf-44c3-9138-766b101f9979
=====
AUTHENTICATION SUCCESS
=====
PERSON DATA
=====
Name: worker 1
```

```
Email: worker1@orgsemcountry.com
Team id: 18187
DeviceIdentifier worker_1
Team Name: Equipe teste jornada
Starting import persons
person import successfully
Person: worker 1 import successfully
=====
PERSON DATA
=====
Name: worker 2
Email: worker1@orgsemcountry.com
Team id: 18187
DeviceIdentifier worker_2
Team Name: Equipe teste jornada
Starting import persons
person import successfully
Person: worker 2 import successfully
=====
PERSON DATA
=====
Name: worker 5
Email: worker1@orgsemcountry.com
Team id: 18187
DeviceIdentifier worker_5
Team Name: Equipe teste jornada
Starting import persons
person import successfully
Person: worker 5 import successfully
=====
RESULT OF IMPORT OPERATION
=====
Total of requests made 5
Total of requests successfully: 5
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará as requisições Http para a API fSense e internalizar as informações das pessoas para o fSense.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Pessoas** para verificar se as pessoas foram inseridas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Atualizar Pessoas

Para atualizados dados de um pessoa cadastrada no fSense partir de um arquivo CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedades `service.name` para `PERSON_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=PERSON_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `UPDATE_PERSON` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=UPDATE_PERSON
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário especificar o caminho do arquivo CSV contendo as informações das pessoas a serem atualizadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório

`src/main/resources/files/import/person_service_update_person.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações das pessoas a serem importadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = person_service_update_person.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO  
csv.delimiter  
csv.delimiter = ,  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO  
csv.text.qualifier  
csv.text.qualifier = '
```



**NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [PERSON_SERVICE]
=====
SELECTED METHOD [IMPORT_PERSON]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1b7749ad-2cbf-44c3-9138-766b101f9979
=====
AUTHENTICATION SUCCESS
=====
PERSON DATA
=====
Name: worker 1 updated
Email: worker1@orgsemcountry.com
Team id: 18187
DeviceIdentifier worker_1
Team Name: Equipe teste jornada
Starting update persons
person update successfully
Person: worker 1 updated successfully
=====
PERSON DATA
=====
Name: worker 2 updated
Email: worker1@orgsemcountry.com
Team id: 18187
DeviceIdentifier worker_2
Team Name: Equipe teste jornada
Starting update persons
person update successfully
Person: worker 2 updated successfully
=====
```

## PERSON DATA

```
=====
Name: worker 5 updated
Email: worker1@orgsemcountry.com
Team id: 18187
DeviceIdentifier worker_5
Team Name: Equipe teste jornada
Starting update persons
person update successfully
Person: worker 5 updated successfully
=====
```

## RESULT OF UPDATE OPERATION

```
=====
Total of requests made 5
Total of requests successfully: 5
Total of requests failed: 0
=====
```

**End** of execution.

A aplicação realizará as requisições Http para a API fSense e atualizar as informações das pessoas para o fSense.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Pessoas** para verificar se as pessoas foram atualizadas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Exportar Líderes de Pessoas

Para exportar os líderes de Pessoas no formato CSV, podemos fazer da seguinte forma:

## Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, configure o serviço a ser executado através da propriedade `service.name` para `PERSON_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=PERSON_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_LEADER_PERSON` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_LEADER_PERSON
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

Para exportar os líderes de pessoas no fSense, é necessário informar o caminho do arquivo CSV contendo as informações das pessoas para exportar os seus respectivos líderes. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório **`src/main/resources/files/import/person_service_export_leader_person.csv`**. Preencha as informações conforme as colunas indicadas no modelo.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]
path.import = ./src/main/resources/files/import

#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]
file.name = person_service_export_leader_person.csv
```

```
#CONFIGURAÇÕES DO CSV
```

```
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARAMETRO
```

```
csv.delimiter
```

```
csv.delimiter = ,
```

```
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARAMETRO
```

```
csv.text.qualifier
```

```
csv.text.qualifier = '
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

A aplicação exportará os arquivos com nome `leader_of_person{ person_id }.csv`.

```
=====
SELECTED SERVICE [PERSON_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT__LEADER_PERSON]
Exporting persons
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5839dd80-1444-4a8e-a2c2-71d5f18ba8d0
=====
AUTHENTICATION SUCCESS
=====
GENERATING CSV
=====
Generating file leader_persons.csv
File: leader_persons.csv saved successfully
Path to file saved: ./src/main/resources/files/export/leader_persons.csv
=====
EXPORT LEADER PERSON FINISHED SUCCESSFULLY
=====
```

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as equipes exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'personId';'leaderId';'name';'email';'deviceIdentifier';'teamId';'teamName';'root
'6702360';'6665124';'ROOT';'root@fsense.com';'''''';'true'
'6744643';'6769489';'Gestor de equipe';'gestor@fsense.io';'''''';'false'
'6744654';'6755534';'usuario';'usuario@fsense.com';'usuario@fsense.com';'4079';'T
da Informação';'false'
'6755544';'6665124';'ROOT';'maria@fsense.com';'''''';'true'
```

# Serviço de Categorias

Serviço é responsável por fazer a busca de categorias de aplicação por pessoa ou por equipe.

## Exportar Categorias - Exemplo de Implementação CSV

Para exportar as categorias da organização no formato CSV podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `CATEGORY_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=CATEGORY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_CATEGORIES` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_CATEGORIES
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

Para utilizar este método, é preciso definir o email da pessoa que será buscada as categorias, o parâmetro a ser definido no arquivo fsense.properties, como no exemplo abaixo:

```
#USUARIO PARA SERVIÇO DE BUSCA DE CATEGORIAS
category.username=usuario@fsense.com
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação. Podemos ver que a implementação selecionada é de CSV.

```
=====
SELECTED SERVICE [CATEGORY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_CATEGORIES]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5e7b9bee-bddf-40e3-a6be-c3de6af46694
=====
AUTHENTICATION SUCCESS
=====
Nº categories found: 390
=====
GENERATING CSV
=====
Generated file categories.csv
Status: saved successfully
File saved in path: ./src/main/resources/files/export/categories.csv
```

```
=====
EXPORT CATEGORIES FINISHED SUCCESSFULLY
=====
```

End of execution.

Os arquivos são armazenados no diretório `src/main/resources/files/export`

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar se as categorias foram exportadas. O arquivo terá uma estrutura semelhante a do exemplo abaixo:

```
'applicationName','category','categoryName','username'
'cursos.alura.com.br','B','Business Use','fsense_user'
'Gerenciador de Limpeza de Espaço em Disco para Windows','B','Business
Use','fsense_user'
'FortiClient System Tray Controller','B','Business Use','fsense_user'
'login.live.com','B','Business Use','fsense_user'
'bitbucket.org','B','Business Use','fsense_user'
'Adobe Acrobat','A','Personal Use','fsense_user'
'srvseniormdw01.callink.com.br:8080','B','Business Use','fsense_user'
'drive.google.com','A','Personal Use','fsense_user'
'agent.fsense.com','D','Unapproved Personal Use','fsense_user'
'Microsoft Word','B','Business Use','fsense_user'
'Microsoft Excel','B','Business Use','fsense_user'
```

## Exportar Categorias - Exemplo de Implementação de Banco de Dados

Para exportar as categorias da organização, fazendo a persistência no banco de dados, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `CATEGORY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=CATEGORY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_CATEGORIES` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_CATEGORIES
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=database
```

## Parâmetros de Exportação

Para utilizar este método, é preciso definir o email da pessoa que será buscada as categorias, o parâmetro a ser definido no arquivo `fsense.properties`, como no exemplo abaixo:

```
#USUARIO PARA SERVIÇO DE BUSCA DE CATEGORIAS
category.username=usuario@fsense.com
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [CATEGORY_SERVICE]
```

```

=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_CATEGORIES]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5e7b9bee-bddf-40e3-a6be-c3de6af46694
=====
AUTHENTICATION SUCCESSFULLY
=====
Nº categories found: 390
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_categories
UpdateCount: 390
Database connection close
=====
SAVING DATABASE CATEGORIES FINISHED SUCCESSFULLY
=====
End of execution.

```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_category**.

```
SELECT * FROM tb_category
```

A tabela terá uma estrutura semelhante à esta:

applicationName	category	categoryName	username
cursos.alura.com.br	B	Business Use	fsense_user

<b>applicationName</b>	<b>category</b>	<b>categoryName</b>	<b>username</b>
Gerenciador de Limpeza de Espaço em Disco para Windows	B	Business Use	fsense_user
FortiClient System Tray Controller	B	Business Use	fsense_user
login.live.com	B	Business Use	fsense_user
bitbucket.org	B	Business Use	fsense_user
Adobe Acrobat	A	Personal Use	fsense_user
srvseniormdw01.callink.com.br:8080	B	Business Use	fsense_user
drive.google.com	A	Personal Use	fsense_user
agent.fsense.com	D	Unapproved Personal Use	fsense_user
Microsoft Word	B	Business Use	fsense_user
Microsoft Excel	B	Business Use	fsense_user

## **Exportar Categorias Gerais - Exemplo de Implementação de CSV**

Para exportar as categorias gerais da organização no formato CSV podemos fazer da seguinte maneira:

### **Configuração do Arquivo Properties**

### **Definição do Serviço**

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `CATEGORY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=CATEGORY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_GENERAL_CATEGORIES` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_GENERAL_CATEGORIES
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

Opcionalmente pode-se definir como parâmetro as opções como:

- **LastCollectedData** - Data mais recente que algum usuário acessou a aplicação/site,
- **LastModification** - Data mais recente que alguma categoria foi alterada no sistema e também
- **OnlyCategorized** - Busca apenas as aplicações já categorizadas.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE CATEGORIAS GERAIS
#[dd/MM/yyyy]
last.collected.date = dd/MM/yyyy
```

```
last.modification.date = dd/MM/yyyy
#SE ESTE PARAMETRO NAO ESTIVER MARCADO, O PROGRAMA ENTENDERÁ COMO [false]
#[true/false]
only.categorized = false
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [CATEGORY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_GENERAL_CATEGORIES]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5e7b9bee-bddf-40e3-a6be-c3de6af46694
=====
AUTHENTICATION SUCCESS
=====
Nº categories found: 2600
  Nº of pages found: 53
Getting general categories page 1/53
Getting general categories page 2/53
Getting general categories page 3/53
---
Getting general categories page 53/53
=====
GENERATING CSV
=====
Generated file general_categories.csv
Status: saved successfully
File saved in path:
./src/main/resources/files/export/general_categories.csv
=====
EXPORT GENERAL CATEGORIES FINISHED SUCCESSFULLY
```

```
=====  
End of execution.
```

Os arquivos são armazenados no diretório `src/main/resources/files/export`

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar se as categorias gerais foram exportadas. O arquivo terá uma estrutura semelhante a do exemplo abaixo:

```
'applicationName', 'category', 'categoryName', 'username'  
'cursos.alura.com.br', 'B', 'Business Use', 'fsense_user'  
'Gerenciador de Limpeza de Espaço em Disco para Windows', 'B', 'Business  
Use', 'fsense_user'  
'FortiClient System Tray Controller', 'B', 'Business Use', 'fsense_user'  
'login.live.com', 'B', 'Business Use', 'fsense_user'  
'bitbucket.org', 'B', 'Business Use', 'fsense_user'  
'Adobe Acrobat', 'A', 'Personal Use', 'fsense_user'  
'srvseniormdw01.callink.com.br:8080', 'B', 'Business Use', 'fsense_user'  
'drive.google.com', 'A', 'Personal Use', 'fsense_user'  
'agent.fsense.com', 'D', 'Unapproved Personal Use', 'fsense_user'  
'Microsoft Word', 'B', 'Business Use', 'fsense_user'  
'Microsoft Excel', 'B', 'Business Use', 'fsense_user'
```

## Exportar Categorias Gerais - Exemplo de Implementação de Banco de Dados

Para exportar as categorias gerais da organização, fazendo a persistência no banco de dados, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `CATEGORY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=CATEGORY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_GENERAL_CATEGORIES` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_GENERAL_CATEGORIES
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=database
```

## Parâmetros de Exportação

Opcionalmente pode-se definir como parâmetro as opções como:

- **LastCollectedData** - Data mais recente que algum usuário acessou a aplicação/site,
- **LastModification** - Data mais recente que alguma categoria foi alterada no sistema e também
- **OnlyCategorized** - Busca apenas as aplicações já categorizadas.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE CATEGORIAS GERAIS
#[dd/MM/yyyy]
last.collected.date = dd/MM/yyyy
last.modification.date = dd/MM/yyyy
#SE ESTE PARAMETRO NAO ESTIVER MARCADO, O PROGRAMA ENTENDERÁ COMO [false]
```

```
#[true/false]
only.categorized = false
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [CATEGORY_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_CATEGORIES]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5e7b9bee-bddf-40e3-a6be-c3de6af46694
=====
AUTHENTICATION SUCCESSFULLY
=====
Nº categories found: 2600
Nº of pages found: 53
Getting general categories page 1/53
Getting general categories page 2/53
Getting general categories page 3/53
---
Getting general categories page 53/53
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_categories
UpdateCount: 2600
Database connection close
=====
SAVING DATABASE GENERAL_CATEGORIES FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_general\_category**.

```
SELECT * FROM tb_category
```

A tabela terá uma estrutura semelhante à esta:

<b>applicationName</b>	<b>category</b>	<b>categoryName</b>	<b>username</b>
cursos.alura.com.br	B	Business Use	fsense_user
Gerenciador de Limpeza de Espaço em Disco para Windows	B	Business Use	fsense_user
FortiClient System Tray Controller	B	Business Use	fsense_user
login.live.com	B	Business Use	fsense_user
bitbucket.org	B	Business Use	fsense_user
Adobe Acrobat	A	Personal Use	fsense_user
srvseniormdw01.callink.com.br:8080	B	Business Use	fsense_user
drive.google.com	A	Personal Use	fsense_user
agent.fsense.com	D	Unapproved Personal Use	fsense_user
Microsoft Word	B	Business Use	fsense_user
Microsoft Excel	B	Business Use	fsense_user

# Exportar Categorias por Equipe - Exemplo de Implementação CSV

Para exportar as categorias por Equipe da organização no formato CSV podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `CATEGORY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=CATEGORY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `EXPORT_CATEGORIES_BY_TEAM` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_CATEGORIES_BY_TEAM
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

### Parâmetros de Exportação

Opcionalmente pode-se definir como parâmetro as opções como:

- **LastCollectedData** - Data mais recente que algum usuário acessou a aplicação/site,
- **LastModification** - Data mais recente que alguma categoria foi alterada no sistema e também
- **OnlyCategorized** - Busca apenas as aplicações já categorizadas.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE CATEGORIAS GERAIS
#[dd/MM/yyyy]
last.collected.date = dd/MM/yyyy
last.modification.date = dd/MM/yyyy
#SE ESTE PARAMETRO NAO ESTIVER MARCADO, O PROGRAMA ENTENDERÁ COMO [false]
#[true/false]
only.categorized = false
```

Também podemos selecionar uma equipe específica para buscar as categorias informando o id da equipe neste parâmetro

```
#ID EQUIPE PARA SERVIÇO DE BUSCA DE CATEGORIAS POR EQUIPE
category.team.id=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [CATEGORY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_CATEGORIES_BY_TEAM]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5e7b9bee-bddf-40e3-a6be-c3de6af46694
=====
```

```
AUTHENTICATION SUCCESS
```

```
=====
```

```
Nº categories found: 1958
```

```
Nº of pages found: 48
```

```
Getting categories by team page 1/48
```

```
Getting categories by team page 2/48
```

```
Getting categories by team page 3/48
```

```
---
```

```
Getting categories by team page 48/48
```

```
=====
```

```
GENERATING CSV
```

```
=====
```

```
Generated file categories_team.csv
```

```
Status: saved successfully
```

```
File saved in path: ./src/main/resources/files/export/categories_team.csv
```

```
=====
```

```
EXPORT GENERAL CATEGORIES BY TEAM FINISHED SUCCESSFULLY
```

```
=====
```

```
End of execution.
```

Os arquivos são armazenados no diretório `src/main/resources/files/export`

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar se as categorias por equipe foram exportadas. O arquivo terá uma estrutura semelhante a do exemplo abaixo:

```
'applicationName','category','categoryName','teamId','teamName'  
'console.cloud.google.com','','','6499','Comercial'  
'revistacrescer.globo.com','','','6499','Comercial'  
'www.imovelweb.com.br','','','6499','Comercial'  
'globoesporte.globo.com','','','6499','Comercial'  
'economia.uol.com.br','','','6499','Comercial'  
'www.globo.com','','','6499','Comercial'  
'centauro.com.br','','','6499','Comercial'  
'revistacasaedjardim.globo.com','','','6499','Comercial'  
'meet.google.com','','','6499','Comercial'  
'gfl.sinclog.com.br','','','6499','Comercial'  
'gshow.globo.com','','','6499','Comercial'
```

# Exportar Categorias por Equipe - Exemplo de Implementação de Banco de Dados

Para exportar as categorias por equipe da organização, fazendo a persistência no banco de dados, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `CATEGORY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=CATEGORY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `EXPORT_CATEGORIES_BY_TEAM` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_CATEGORIES_BY_TEAM
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO  
operation.type=database
```

### Parâmetros de Exportação

Opcionalmente pode-se definir como parâmetro as opções como:

- **LastCollectedData** - Data mais recente que algum usuário acessou a aplicação/site,
- **LastModification** - Data mais recente que alguma categoria foi alterada no sistema e também
- **OnlyCategorized** - Busca apenas as aplicações já categorizadas.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE CATEGORIAS GERAIS
#[dd/MM/yyyy]
last.collected.date = dd/MM/yyyy
last.modification.date = dd/MM/yyyy
#SE ESTE PARAMETRO NAO ESTIVER MARCADO, O PROGRAMA ENTENDERÁ COMO [false]
#[true/false]
only.categorized = false
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [CATEGORY_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_CATEGORIES_BY_TEAM]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
5e7b9bee-bddf-40e3-a6be-c3de6af46694
=====
AUTHENTICATION SUCCESSFULLY
=====
Nº categories found: 1958
Nº of pages found: 48
Getting categories by team page 1/48
Getting categories by team page 2/48
Getting categories by team page 3/48
---
```

```
Getting categories by team page 48/48
```

```
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_categories
UpdateCount: 1958
Database connection close
=====
SAVING DATABASE CATEGORIES_BY_TEAM FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_categories\_by\_team**.

```
SELECT * FROM tb_categories_by_team
```

A tabela terá uma estrutura semelhante à esta:

applicationName	category	categoryName	teamId	teamName
console.cloud.google.com			6499	Comercial
revistacrescer.globo.com			6499	Comercial
<a href="http://www.imovelweb.com.br">www.imovelweb.com.br</a>			6499	Comercial
globoesporte.globo.com			6499	Comercial
economia.uol.com.br			6499	Comercial
<a href="http://www.globo.com">www.globo.com</a>			6499	Comercial
centauro.com.br			6499	Comercial

<b>applicationName</b>	<b>category</b>	<b>categoryName</b>	<b>teamId</b>	<b>teamName</b>
revistacasaejardim.globo.com			6499	Comercial
meet.google.com			6499	Comercial
gfl.sinclog.com.br			6499	Comercial
gshow.globo.com			6499	Comercial

# Serviço de Jornadas

Este serviço permite internalizar a estrutura de jornadas da sua organização.

## Exportar Jornadas - Exemplo de Implementação CSV

Para exportar as jornadas cadastrados na sua organização no formato CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado ajustando a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=WORKDAY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_WORKDAYS` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_WORKDAYS
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

É possível definir parâmetros opcionais na exportação de jornadas como:

- **name** - Filtra pelo nome da jornada.

```
#PARÂMETROS PARA SERVIÇO DE IMPORTAÇÃO/EXPORTAÇÃO DE JORNADA  
#Exemplo:[Jornada Padrão]  
workday.name=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_WORKDAYS]
=====
export workday
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f18d2c7-d0ce-4e3a-8844-6c6b748bdd83
=====
AUTHENTICATION SUCCESS
=====
Nº of records found: 11
Nº of pages found: 2
Getting workdays page: 1/2
Getting workdays page: 2/2
=====
GENERATING CSV
```

```
=====
Generated file: workdays.csv
Status saved successfully.
File saved in path: ./src/main/resources/files/export/workdays.csv
=====
EXPORT WORKDAYS FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as jornadas exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','name','isFlexible','ignoreNonFocusLongerThan','workNationalHolidays','workR
'74','Geral','false','','false','false','MONDAY','','08:00:00','12:00:00',''
'74','Geral','false','','false','false','MONDAY','','14:00:00','18:00:00',''
'74','Geral','false','','false','false','TUESDAY','','08:00:00','12:00:00',''
'74','Geral','false','','false','false','TUESDAY','','14:00:00','18:00:00',''
'74','Geral','false','','false','false','WEDNESDAY','','08:00:00','12:00:00',''
'74','Geral','false','','false','false','WEDNESDAY','','14:00:00','18:00:00',''
'74','Geral','false','','false','false','THURSDAY','','08:00:00','12:00:00',''
'74','Geral','false','','false','false','THURSDAY','','14:00:00','18:00:00',''
'74','Geral','false','','false','false','FRIDAY','','08:00:00','12:00:00',''
'74','Geral','false','','false','false','FRIDAY','','14:00:00','18:00:00',''
'3051','Jornada Flexível Exemplo','false','','false','false','TUESDAY','07:00:00'
'3051','Jornada Flexível Exemplo','false','','false','false','MONDAY','07:00:00',
'3051','Jornada Flexível Exemplo','false','','false','false','FRIDAY','07:00:00',
'3051','Jornada Flexível Exemplo','false','','false','false','THURSDAY','07:00:00
'3051','Jornada Flexível Exemplo','false','','false','false','WEDNESDAY','07:00:00
'3052','Jornada Fixa Exemplo','false','','true','false','MONDAY','','07:30:00','1:
'3052','Jornada Fixa Exemplo','false','','true','false','MONDAY','','13:12:00','1:
'3052','Jornada Fixa Exemplo','false','','true','false','TUESDAY','','07:30:00','1:
'3052','Jornada Fixa Exemplo','false','','true','false','TUESDAY','','13:12:00','1:
'3052','Jornada Fixa Exemplo','false','','true','false','WEDNESDAY','','07:30:00'
'3052','Jornada Fixa Exemplo','false','','true','false','WEDNESDAY','','13:12:00'
'3052','Jornada Fixa Exemplo','false','','true','false','THURSDAY','','07:30:00',
'3052','Jornada Fixa Exemplo','false','','true','false','THURSDAY','','13:12:00',
'3052','Jornada Fixa Exemplo','false','','true','false','FRIDAY','','07:30:00','1:
'3052','Jornada Fixa Exemplo','false','','true','false','FRIDAY','','13:12:00','1:
```

# Exportar Jornadas - Exemplo de Implementação de Banco de Dados

Para exportar as jornadas cadastrados na sua organização, fazendo a persistência dos dados no banco, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado ajustando a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=WORKDAY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `EXPORT_WORKDAYS` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_WORKDAYS
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO  
operation.type=DATABASE
```

### Parâmetros de Exportação

É possível definir parâmetros opcionais na exportação de jornadas como:

- **name** - Filtra pelo nome da jornada.

```
#PARÂMETROS PARA SERVIÇO DE IMPORTAÇÃO/EXPORTAÇÃO DE JORNADA
#Exemplo:[Jornada Padrão]
workday.name=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_WORKDAYS]
=====
export workday
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f18d2c7-d0ce-4e3a-8844-6c6b748bdd83
=====
AUTHENTICATION SUCCESS
=====
Nº of records found: 11
Nº of pages found: 2
Getting workdays page: 1/2
Getting workdays page: 2/2
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_workday
UpdateCount: 11
Database connection close
=====
SAVING DATABASE WORKDAYS FINISHED SUCCESSFULLY
```

```
=====
End of execution.
```

## Verificação da Operação

Após a execução, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela `tb_workday`.

```
SELECT * FROM tb_workday;
```

A tabela **tb\_workday** terá uma estrutura semelhante à esta:

id	name	work_national_holidays	work_regional_holidays	is_flexible	ig
74	Geral	0	0	0	
3051	Jornada Flexível Exemplo	0	0	0	
3052	Jornada Fixa Exemplo	1	0	0	
3053	Jornada Comercial	1	1	0	
3054	jornada flexivel	0	0	0	

Para visualizar os dias trabalhados faça uma consulta na tabela `tb_workday_schedule`.

```
SELECT * FROM tb_workday_schedule;
```

<b>weekday</b>	<b>start_time</b>	<b>end_time</b>	<b>max_flexible_duration</b>	<b>workday_id</b>
MONDAY	08:00:00	12:00:00		74
MONDAY	14:00:00	18:00:00		74
TUESDAY	08:00:00	12:00:00		74
TUESDAY	14:00:00	18:00:00		74
WEDNESDAY	08:00:00	12:00:00		74
WEDNESDAY	14:00:00	18:00:00		74
THURSDAY	08:00:00	12:00:00		74
THURSDAY	14:00:00	18:00:00		74
FRIDAY	08:00:00	12:00:00		74
FRIDAY	14:00:00	18:00:00		74
TUESDAY	06:00:00	22:00:00	07:00:00	3051
MONDAY	06:00:00	22:00:00	07:00:00	3051
FRIDAY	06:00:00	22:00:00	07:00:00	3051
THURSDAY	06:00:00	22:00:00	07:00:00	3051
WEDNESDAY	06:00:00	22:00:00	07:00:00	3051
MONDAY	07:30:00	12:00:00		3052
MONDAY	13:12:00	17:30:00		3052

weekday	start_time	end_time	max_flexible_duration	workday_id
TUESDAY	07:30:00	12:00:00		3052
TUESDAY	13:12:00	17:30:00		3052
WEDNESDAY	07:30:00	12:00:00		3052
WEDNESDAY	13:12:00	17:30:00		3052
THURSDAY	07:30:00	12:00:00		3052
THURSDAY	13:12:00	17:30:00		3052
FRIDAY	07:30:00	12:00:00		3052
FRIDAY	13:12:00	17:30:00		3052

## Importar Jornadas

Para importar jornadas, a partir de um arquivo CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço que será executado ajustando a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=WORKDAY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `IMPORT_WORKDAY` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=IMPORT_WORKDAY
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações das jornadas a serem importadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório `src/main/resources/import/workday_service_import_workday.csv`. Preencha as informações conforme as colunas indicadas no modelo.

```
# EXEMPLO DE JORNADAS
```

```
'id','name','isFlexible','ignoreNonFocusLongerThan','workNationalHolidays','workR
','Jornada Padrão Exemplo','false','','false','false','MONDAY','','07:30:00','11
','Jornada Padrão Exemplo','false','','false','false','MONDAY','','12:20:00','17
','Jornada Padrão Exemplo','false','','false','false','TUESDAY','','07:30:00','1:
','Jornada Padrão Exemplo','false','','false','false','TUESDAY','','12:20:00','1:
','Jornada Padrão Exemplo','false','','false','false','WEDNESDAY','','07:30:00',
','Jornada Padrão Exemplo','false','','false','false','WEDNESDAY','','12:20:00',
','Jornada Padrão Exemplo','false','','false','false','THURSDAY','','07:30:00',':
','Jornada Padrão Exemplo','false','','false','false','THURSDAY','','12:20:00',':
','Jornada Padrão Exemplo','false','','false','false','FRIDAY','','07:30:00','11
','Jornada Padrão Exemplo','false','','false','false','FRIDAY','','12:20:00','17
','Jornada Flexível Exemplo','true','','false','false','MONDAY','','07:00:00',':
','Jornada Flexível Exemplo','true','','false','false','TUESDAY','','07:00:00',
```

```
','Jornada Flexível Exemplo', 'true','','false','false','WEDNESDAY','','07:00:00'  
','Jornada Flexível Exemplo', 'true','','false','false','THURSDAY','','07:00:00'  
','Jornada Flexível Exemplo', 'true','','false','false','FRIDAY','','07:00:00',':  
','Jornada Flexível Exemplo com Restrição de Horário', 'true','','false','false'  
','Jornada Flexível Exemplo com Restrição de Horário', 'true','','false','false'
```

No exemplo acima, vamos tres jornadas diferentes, uma jornada padrão, uma jornada flexível sem restrição de horário, e uma flexível com restrição de horário.

### ATENÇÃO

Os dias da Semana do Arquivo modelo a ser preenchido devem ser preenchido em **INGLÊS**.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações das jornadas a serem importadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = workday_service_import_workday.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARAMETRO  
csv.delimiter  
csv.delimiter = ;  
  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARAMETRO
```

```
csv.text.qualifier
csv.text.qualifier = '
```

### NOTA

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [IMPORT_WORKDAY]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
90d824df-6691-4c12-8c3c-9a29a5c8f4aa
=====
Authentication done successfully
=====
Jornada Padrão Exemplo - Flexible: false
Jornada Flexível Exemplo - Flexible: true
Jornada Flexível Exemplo com Restrição de Horário - Flexible: true
=====
Jornada Padrão Exemplo imported successfully!
=====
Jornada Flexível Exemplo imported successfully!
=====
Jornada Flexível Exemplo com Restrição de Horário imported successfully!
=====
RESULT OF IMPORT OPERATION
=====
Total of requests made 3
Total of requests successfully: 3
```

```
Total of requests failed: 0
```

```
=====
```

```
End of execution.
```

A aplicação realizará as requisições para importar as jornadas no fSense.

## Verificação da Operação

Abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Jornada** para verificar se as jornadas foram inseridas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Exportar Jornadas por ID

Para exportar uma jornada, buscando pelo ID no formato CSV, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço a ser executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=WORKDAY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `EXPORT_WORKDAY_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_WORKDAY_BY_ID
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

Para essa operação é necessário definir o identificador único da jornada a qual se deseja exportar, através do parâmetro `workday.id` no arquivo `fsense.properties`.

```
#Exemplo=[82]  
workday.id=
```

Como exemplo, vamos buscar pelo id abaixo:

```
#Exemplo=[82]  
workday.id=3051
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_WORKDAY_BY_ID]
=====
workday id to be exported: [3051]
checking if token is valid?: false
=====
Starting authentication process
```

```
Getting access token
Token received.
e5467634-2d3b-41af-8ec4-86bba853a4b1
=====
Authentication done successfully
=====
export workday by id
Workday founded: [3051]
=====
GENERATING CSV
=====
Generated file: workday_3051.csv
Status: saved successfully.
Path to saved file: ./src/main/resources/files/export/workday_3051.csv
=====
EXPORT WORKDAY BY ID FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar a jornada exportada. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','name','isFlexible','ignoreNonFocusLongerThan','workNationalHolidays','workR
'3051','Jornada Flexível Exemplo','false','','false','false','TUESDAY','07:00:00',
'3051','Jornada Flexível Exemplo','false','','false','false','MONDAY','07:00:00',
'3051','Jornada Flexível Exemplo','false','','false','false','FRIDAY','07:00:00',
'3051','Jornada Flexível Exemplo','false','','false','false','THURSDAY','07:00:00
'3051','Jornada Flexível Exemplo','false','','false','false','WEDNESDAY','07:00:00'
```

## Atualizar Jornadas por ID

Para atualizar jornadas cadastradas no fSense a partir de um arquivo CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.name=WORKDAY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `UPDATE_WORKDAY_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=UPDATE_WORKDAY_BY_ID
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

Para atualizar uma jornada cadastrada no fSense, é preciso especificar o caminho do arquivo CSV contendo as informações das jornadas a serem atualizadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório `src/main/resources/import/workday_service_update_workday.csv`. Preencha as informações conforme as colunas indicadas no modelo.

### ATENÇÃO

Os dias da Semana do Arquivo modelo a ser preenchido devem ser preenchido em **INGLÊS**.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações das jornadas a serem atualizadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = workday_service_update_workday.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARAMETRO  
csv.delimiter  
csv.delimiter = ;  
  
#UTILIZE [VAZIO] [ASPAS SIMPES] OU [ASPAS DUPLAS] PARA O PARAMETRO  
csv.text.qualifier  
csv.text.qualifier = '
```

### **NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação. Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
```

```

=====
SELECTED METHOD [UPDATE_WORKDAY_BY_ID]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
90d824df-6691-4c12-8c3c-9a29a5c8f4aa
=====
Authentication done successfully
=====
Jornada Padrão Exemplo ATUALIZADA - Flexible: false
Jornada Flexível Exemplo ATUALIZADA- Flexible: true
Jornada Flexível Exemplo com Restrição de Horário ATUALIZADA - Flexible:
true
=====
Jornada Padrão Exemplo update successfully!
=====
Jornada Flexível Exemplo update successfully!
=====
Jornada Flexível Exemplo com Restrição de Horário update successfully!
=====
RESULT OF UPDATE OPERATION
=====
Total of requests made 3
Total of requests successfully: 3
Total of requests failed: 0
=====
End of execution.

```

A aplicação realizará as requisição para atualizar as jornadas informadas no fSense.

## Verificação da Operação

Após a execução da Aplicação, abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Jornada** e verifique se as jornadas foram atualizadas corretamente.

Caso haja algum erro, será exibido uma mensagem de erro no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

# Deletar Jornadas por ID

Para remover jornadas cadastradas, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.name=WORKDAY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `DELETE_WORKDAY_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=DELETE_WORKDAY_BY_ID
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

### Definição de Parâmetros

Para remover uma ou mais jornadas, é necessário informar o identificador único da jornada que deseja remover através `workday.ids`. É possível informar um ou mais ids, basta

informá-los separados por vírgula.

```
## ID DAS JORNADAS PARA REMOVER  
workday.ids=2775, 2776, 2777
```

Após definidos os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [DELETE_WORKDAY_BY_ID]
=====
Workday id selected to be delete: 2775
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
50ade397-a5c8-43c9-a6b3-729540d977aa
=====
AUTHENTICATION SUCCESS
=====
Workday id 2775 removed successfully
=====
Workday id 2776 removed successfully
=====
Workday id 2777 removed successfully
=====
RESULT OF DELETE OPERATION
=====
Total of requests made 3
Total of requests successfully: 3
Total of requests failed: 0
=====
End of execution.
```

## Verificação da Operação

Após a execução da Aplicação, abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Jornada** e verifique se as jornadas foram removidas.

Caso haja algum erro, será exibido uma mensagem de erro no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Associar Pessoas

Para associar pessoas a uma jornada, podemos fazer da seguinte forma:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=WORKDAY_SERVICE
```

#### Método do Serviço

Configure o método para `ASSOCIATE_PEOPLE` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=ASSOCIATE_PEOPLE
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
```

```
operation.type=csv
```

## Parâmetros de Exportação

É necessário definir o identificador único da Jornada que deseja associar as pessoas, juntamente com identificador único das pessoas que deseja associar à jornada. Se necessário, execute antes o serviço `PERSON_SERVICE` juntamente como o método `EXPORT_PERSONS`, para obter as pessoas que deseja associar à jornada.

Como exemplo, vamos utilizar esta jornada

```
## PARÂMETRO PARA SERVIÇOS DE JORNADA  
workday.id=2778
```

```
## ID PESSOAS PARA ASSOCIAR/DESASSOCIAR À JORNADA  
workday.person.ids=675359,6757360,6757361,6757362,6757362,6757363
```

## Execução da Aplicação

Após definido os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [ASSOCIATE_PEOPLE]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
ASSOCIATE PEOPLES ON WORKDAY ID: 18187
ID OF PEOPLES SELECTED TO ASSOCIATE INTO WORKDAY:
```

```
=====  
[6757359, 6757360, 6757361, 6757362, 6757363]  
=====
```

```
Peoples associate into workday successfully  
=====
```

```
End of execution.
```

A aplicação realizará requisições Http para a API fSense para associar as pessoas à jornada.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Pessoas** para verificar se as pessoas foram associadas à jornada corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Desassociar Pessoas

Para desassociar pessoas de uma jornada, siga as instruções abaixo:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
# DEFINIÇÃO DO SERVIÇO  
service.name=WORKDAY_SERVICE
```

### Método do Serviço

Configure o método para `DISASSOCIATE_PEOPLE` através da propriedade `service.method`.

```
# DEFINIÇÃO DO MÉTODO
service.method=DISASSOCIATE_PEOPLE
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

Defina o identificador único da Jornada da qual deseja desassociar as pessoas, juntamente com os identificadores únicos das pessoas que deseja desassociar.

Como exemplo, vamos utilizar a mesma jornada:

```
# ID DA JORNADA QUE DESEJA ASSOCIAR/DESASSOCIAR PESSOAS/EQUIPES
workday.id=2778
```

```
# ID PESSOAS PARA DESASSOCIAR À JORNADA
workday.person.ids=675359,6757360,6757361,6757362,6757363
```

## Execução da Aplicação

Após definir os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [DISASSOCIATE_PEOPLE]
=====
```

```
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
DISASSOCIATE PEOPLES FROM WORKDAY ID: 2778
ID OF PEOPLES SELECTED TO DISASSOCIATE FROM WORKDAY:
=====
[6757359, 6757360, 6757361, 6757362, 6757363]
=====
Peoples disassociated from workday successfully
=====
End of execution.
```

A aplicação realizará requisições HTTP para a API fSense para desassociar as pessoas da jornada.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Pessoas** para verificar se as pessoas foram desassociadas da jornada corretamente. Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

### INFO

As jornadas possuem uma hierarquia e podem ser aplicadas em nível de organização, em nível de Equipe, e em nível individual, caso a jornada em nível individual, se houver uma configuração de jornada definida para a Equipe da qual a pessoa pertence, automaticamente será aplicada, conforme a hierarquia da jornada.

## Associar Equipes

Para associar equipes a uma jornada, siga as instruções abaixo:

# Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
service.name=WORKDAY_SERVICE
```

## Método do Serviço

Configure o método para `ASSOCIATE_TEAMS` através da propriedade `service.method`.

```
service.method=ASSOCIATE_TEAMS
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

Defina o identificador único da Jornada à qual deseja associar as equipes, juntamente com os identificadores únicos das equipes que deseja associar.

Como exemplo, vamos utilizar a mesma jornada:

```
workday.id=2778
```

```
# ID EQUIPES PARA ASSOCIAR À JORNADA
```

```
workday.team.ids=123456,123457,123458,123459,123460
```

## Execução da Aplicação

Após definir os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [ASSOCIATE_TEAMS]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
ASSOCIATE TEAMS ON WORKDAY ID: 2778
ID OF TEAMS SELECTED TO ASSOCIATE INTO WORKDAY:
=====
[123456, 123457, 123458, 123459, 123460]
=====
Teams associated into workday successfully
=====
End of execution.
```

A aplicação realizará requisições HTTP para a API fSense para associar as equipes à jornada.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipes** para verificar se as equipes foram associadas à jornada corretamente. Caso haja algum erro durante a operação, uma

mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

### ! INFO

As jornadas possuem uma hierarquia e podem ser aplicadas em nível de organização, em nível de Equipe, e em nível individual, caso a jornada em nível individual, se houver uma configuração de jornada definida para a Equipe da qual a pessoa pertence, e ser aplicada uma jornada individual, a jornada individual vai sobrescrever a definição em nível de equipe, e em nível de organização.

## Desassociar Equipes

Para desassociar equipes de uma jornada, siga as instruções abaixo:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
service.name=WORKDAY_SERVICE
```

#### Método do Serviço

Configure o método para `DISASSOCIATE_TEAMS` através da propriedade `service.method`.

```
service.method=DISASSOCIATE_TEAMS
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

Defina o identificador único da Jornada da qual deseja desassociar as equipes, juntamente com os identificadores únicos das equipes que deseja desassociar.

```
# Como exemplo, vamos utilizar a mesma jornada:
workday.id=2778
```

```
# ID EQUIPES PARA DESASSOCIAR DA JORNADA
workday.team.ids=123456,123457,123458,123459,123460
```

## Execução da Aplicação

Após definir os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [DISASSOCIATE_TEAMS]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
DISASSOCIATE TEAMS FROM WORKDAY ID: 2778
ID OF TEAMS SELECTED TO DISASSOCIATE FROM WORKDAY:
=====
[123456, 123457, 123458, 123459, 123460]
=====
```

```
Teams disassociated from workday successfully
```

```
=====  
End of execution.
```

A aplicação realizará requisições HTTP para a API fSense para desassociar as equipes da jornada.

## Verificação da Operação

Após a execução da aplicação, abra o painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Monitorados > Equipes** para verificar se as equipes foram desassociadas da jornada corretamente. Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

### ! INFO

As jornadas possuem uma hierarquia e podem ser aplicadas em nível de organização, em nível de Equipe, e em nível individual, caso a jornada em nível individual, se houver uma configuração de jornada definida para a Equipe da qual a pessoa pertence, e ser aplicada uma jornada individual, a jornada individual vai sobrescrever a definição em nível de equipe, e em nível de organização.

## Exportar pessoas associadas a uma jornada por ID

Para exportar pessoas associadas a uma jornada, buscando pelo ID no formato CSV, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço a ser executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
service.name=WORKDAY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_PERSON_ASSOCIATE_ON_WORKDAY_BY_ID` através da propriedade `service.method`.

```
service.method=EXPORT_PERSON_ASSOCIATE_ON_WORKDAY_BY_ID
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

Para essa operação, é necessário definir o identificador único da jornada da qual se deseja exportar as pessoas, através do parâmetro `workday.id` no arquivo `fsense.properties`.

Como exemplo, vamos buscar pelo ID abaixo:

```
# Exemplo=[82]  
workday.id=3051
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_PERSON_ASSOCIATE_ON_WORKDAY_BY_ID]
```

```
=====
workday id to be exported: [3051]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
e5467634-2d3b-41af-8ec4-86bba853a4b1
=====
Authentication done successfully
=====
Exporting persons associated with workday
Workday ID found: [3051]
=====
GENERATING CSV
=====
Generated file: persons_workday_3051.csv
Status: saved successfully.
Path to saved file: ./src/main/resources/files/export/persons_workday_3051.csv
=====
EXPORT PERSONS ASSOCIATED WITH WORKDAY FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as pessoas exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','name','email','workdayId','status'
'6757359','João Silva','joao.silva@example.com','3051','active'
'6757360','Maria Oliveira','maria.oliveira@example.com','3051','active'
'6757361','Pedro Santos','pedro.santos@example.com','3051','inactive'
'6757362','Ana Costa','ana.costa@example.com','3051','active'
'6757363','Lucas Pereira','lucas.pereira@example.com','3051','active'
```

## Exportar equipes associadas a uma jornada por ID

Para exportar equipes associadas a uma jornada, buscando pelo ID no formato CSV, podemos fazer da seguinte maneira:

# Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, configure o serviço a ser executado definindo a propriedade `service.name` para `WORKDAY_SERVICE`.

```
service.name=WORKDAY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_TEAM_ASSOCIATE_ON_WORKDAY_BY_ID` através da propriedade `service.method`.

```
service.method=EXPORT_TEAM_ASSOCIATE_ON_WORKDAY_BY_ID
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

Para essa operação, é necessário definir o identificador único da jornada da qual se deseja exportar as equipes, através do parâmetro `workday.id` no arquivo `fsense.properties`.

```
# Como exemplo, vamos buscar pelo ID abaixo:  
# Exemplo=[82]  
workday.id=3051
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_TEAM_ASSOCIATE_ON_WORKDAY_BY_ID]
=====
workday id to be exported: [3051]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
e5467634-2d3b-41af-8ec4-86bba853a4b1
=====
Authentication done successfully
=====
Exporting teams associated with workday
Workday ID found: [3051]
=====
GENERATING CSV
=====
Generated file: teams_workday_3051.csv
Status: saved successfully.
Path to saved file: ./src/main/resources/files/export/teams_workday_3051.csv
=====
EXPORT TEAMS ASSOCIATED WITH WORKDAY FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as equipes exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','teamName','teamLead','workdayId','status'
'123456','Equipe A','Carlos Mendes','3051','active'
'123457','Equipe B','Fernanda Lima','3051','active'
'123458','Equipe C','Roberto Costa','3051','inactive'
```

'123459', 'Equipe D', 'Patricia Santos', '3051', 'active'

'123460', 'Equipe E', 'Eduardo Lima', '3051', 'active'

# Trocas de Jornada

Este Serviço permite internalizar a estrutura de trocas jornadas da sua organização.

## Exportar Trocas de Jornada - Exemplo de Implementação CSV

Para exportar as trocas de jornada cadastrados na organização, podemos fazer os seguintes passos:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_OVERRIDE_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=WORKDAY_OVERRIDE_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_WORKDAY_OVERRIDE` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_WORKDAY_OVERRIDE
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Execução

Opcionalmente é possível definir um filtro de exportação, como o identificador único da pessoa cadastrada no fSense. Dessa forma, só serão exportados as trocas de jornada da pessoa informada. Também é possível definir um intervalo de data que deseja fazer a exportação

- **personId** - Identificador único de uma pessoa cadastrada no fSense.
- **startDate** - Data inicial para busca de trocas de jornada.
- **endDate** - Data final para busca de trocas de jornada.

```
#PARÂMETROS PARA SERVIÇO DE TROCAS DE JORNADA
#Exemplo:[67445454]
workday.override.person.id=
##[dd/MM/yyyy]
workday.override.start.date=
##[dd/MM/yyyy]
workday.override.end.date=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_OVERRIDE_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_WORKDAY_OVERRIDE]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
```

```
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
AUTHENTICATION SUCCESS
=====
Getting workday overrides
Nº records found: 2
Nº pages found: 1
=====
GENERATING CSV
=====
Generated file: workday_overrides.csv
Status: saved successfully
File saved in path: ./src/main/resources/files/export/workday_overrides.csv
=====
EXPORT WORKDAY OVERRIDES FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as trocas de jornada exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','personId','personName','personDeviceIdentifier','personEmail','date','start'
'303561','6755534','usuario','usuario@fsense.com','usuario@fsense.com','2023-11-2:
'303274','6744454','Maria','maria_silva','mariasilva@gmail.com','2023-09-16','06::
```

## Exportar Trocas de Jornada - Exemplo de Implementação de Banco de dados

Para exportar as trocas de jornadas cadastradas na organização, fazendo a persistência de dados no banco, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `WORKDAY_OVERRIDE_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=WORKDAY_OVERRIDE_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_WORKDAY_OVERRIDE` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_WORKDAY_OVERRIDE
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=database
```



### DICA

Certifique-se de que o banco de dados está devidamente configurado e a tabela previamente criada. Certifique-se de que o banco de dados está devidamente configurado e a tabela previamente criada.

Se tiver dúvidas, consulte a seção [Integração com Banco de Dados](#).

## Parâmetros de Exportação

Opcionalmente é possível definir um filtro de exportação, como o identificador único da pessoa cadastrada no fSense. Dessa forma, só serão exportados as trocas de jornada da pessoa informada. Também é possível definir um intervalo de data que deseja fazer a exportação

- **personId** - Identificador único de uma pessoa cadastrada no fSense.
- **startDate** - Data inicial para busca de trocas de jornada.
- **endDate** - Data final para busca de trocas de jornada.

```
#PARÂMETROS PARA SERVIÇO DE TROCAS DE JORNADA
#Exemplo:[67445454]
workday.override.person.id=
##[dd/MM/yyyy]
workday.override.start.date=
##[dd/MM/yyyy]
workday.override.end.date=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_OVERRIDE_SERVICE]
=====
EXPORTATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_WORKDAY_OVERRIDE]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
AUTHENTICATION SUCCESS
=====
Getting workday overrides
Nº records found: 2
Nº pages found: 1
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_workday_override
Update Count: 2
```

```
Database connection Close
```

```
=====
SAVING DATABASE WORKDAY OVERRIDES FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela `tb_workday_override`.

```
SELECT * FROM tb_workday_override;
```

A tabela terá uma estrutura semelhante à esta:

id	personId	personName	personDeviceIdentifier	personEmail
303561	6755534	usuario	usuario@fsense.com	usuario@fsense.com
303274	6744454	Maria	maria_silva	mariasilva@gmail.com

## Importar Trocas de Jornada

Para importar trocas de jornada para o fSense, a partir de um arquivo CSV, podemos fazer da seguinte forma:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `WORKDAY_OVERRIDE_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=WORKDAY_OVERRIDE_SERVICE
```

## Método do Serviço

Configure o método do serviço para `IMPORT_WORKDAY_OVERRIDE` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=IMPORT_WORKDAY_OVERRIDE
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações das trocas de jornadas a serem importadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório

```
src/main/resources/import/workday_override_service_import_workday_override.csv.
```

Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações das trocas de jornada a serem importadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]
file.name = workday_override_service_import_workday_override.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV

#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARAMETRO
csv.delimiter
csv.delimiter = ;

#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARAMETRO
csv.text.qualifier
csv.text.qualifier = '
```

### NOTA

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_OVERRIDE_SERVICE]
=====
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [IMPORT_WORKDAY_OVERRIDE]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
AUTHENTICATION SUCCESS
```

```
=====
personId=6757362,
date=05/01/2024,
startTime=08:00:00,
intervalStartTime=11:00:00,
endTime=18:08:00,
intervalEndTime=12:12:00,
message=Troca de jornada teste1
=====
personId=6757362,
date=06/01/2024,
startTime=08:00:00,
intervalStartTime=11:00:00,
endTime=18:08:00,
intervalEndTime=12:12:00,
message=Troca de jornada teste2
=====
personId=6757362,
date=07/01/2024,
startTime=08:00:00,
intervalStartTime=11:00:00,
endTime=18:08:00,
intervalEndTime=12:12:00,
message=Troca de jornada teste3
=====
personId=6757362,
date=08/01/2024,
startTime=08:00:00,
intervalStartTime=11:00:00,
endTime=18:08:00,
intervalEndTime=12:12:00,
message=Troca de jornada teste4
=====
starting create workday overrides
Workday override create successfully
2024-01-05 Troca de jornada teste1 worker4@orgsemcountry.com
=====
starting create workday overrides
Workday override create successfully
2024-01-06 Troca de jornada teste2 worker4@orgsemcountry.com
=====
starting create workday overrides
Workday override create successfully
2024-01-07 Troca de jornada teste3 worker4@orgsemcountry.com
=====
```

```
starting create workday overrides
Workday override create successfully
2024-01-08 Troca de jornada teste4 worker4@orgsemcountry.com
=====
RESULT OF IMPORT OPERATION
=====
Finish of operation:
Total of requests made 4
Total of requests successfully: 4
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará as requisições à API fSense para importar as trocas de jornada no fSense.

Caso haja algum erro, será exibido uma mensagem de erro no console da aplicação. Certifique-se de que as informações estejam corretas e correspondam aos dados do arquivo CSV.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Trocas de Jornada** para verificar se as trocas de jornada foram inseridas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Deletar Trocas de Jornada

Para deletar uma troca de jornada cadastrada no fSense, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, configure o serviço a ser executado definindo a propriedade `service.name` para `WORKDAY_OVERRIDE_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=WORKDAY_OVERRIDE_SERVICE
```

## Método do Serviço

Configure o método do serviço para `DELETE_WORKDAY_OVERRIDE` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=DELETE_WORKDAY_OVERRIDE
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Definição de Parâmetros de Remoção

Para remover uma ou mais trocas de jornada, é necessário informar o identificador único da troca de jornada que deseja remover através da propriedade `workday.override.ids`. É possível informar um ou mais ids, basta informá-los separados por vírgula.

```
#### IDS TROCAS DE JORNADA PARA REMOVER
workday.override.ids=
```

## Execução da Aplicação

Definidos os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_OVERRIDE_SERVICE]
=====
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [DELETE_WORKDAY_OVERRIDE]
=====
workday overrides Id selected to deleted:
[305407, 305406, 305405, 305404, 305403, 305402, 305401, 305400, 305399,
305398, 305397, 305396, 305395]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
Authentication done successfully
=====
Try delete workday override id: 305406 - 1/13
Removing workday overrides
Workday Override removed successfully
=====
Try delete workday override id: 305405 - 2/13
Removing workday overrides
Workday Override removed successfully
=====
Try delete workday override id: 305404 - 3/13
Removing workday overrides
Workday Override removed successfully
=====
...
=====
Try delete workday override id: 305395 - 13/13
Removing workday overrides
Workday Override removed successfully
=====
RESULT OF DELETE OPERATION
=====
Total of requests made 13
Total of requests successfully: 13
Total of requests failed: 0
```

```
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Trocas de Jornada** para verificar se as trocas de jornada foram removidas corretamente.

Caso haja algum erro durante a operação, fique atento às mensagens exibidas no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

# Acréscimos de Jornada

Este Serviço internalizar a estrutura de acréscimos de jornada da sua organização.

## Exportar Acréscimos de Jornada - Exemplo de Implementação CSV

Para exportar os acréscimos de jornada cadastrados na organização no formato CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `WORKDAY_PROROGATION_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=WORKDAY_PROROGATION_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_WORKDAY_PROROGATION` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_WORKDAY_PROROGATION
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

É possível filtrar a exportação de acréscimos de jornada definir parâmetros opcionais como:

**personId** - Identificador único de uma pessoa cadastrada no fSense. **date** - Data Específica do acréscimo de Jornada. **status** - Status da solicitação de acréscimo de jornada.

```
#PARAMETROS PARA SERVIÇO DE PRORROGAÇÕES DE JORNADA  
workday.prorogation.person.id=  
##[dd/MM/yyyy]  
workday.prorogation.date=  
#[APPROVED, DENIED, PENDING, CANCELLED, EXPIRED]  
workday.prorogation.status=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====  
SELECTED SERVICE [WORKDAY_PROROGATION_SERVICE]  
=====  
OPERATION TYPE: CSV  
=====  
SELECTED METHOD [EXPORT_WORKDAY_PROROGATION]  
=====  
checking if token is valid?: false  
=====  
Starting authentication process  
Getting access token  
Token received.  
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7  
=====  
AUTHENTICATION SUCCESS  
=====
```

```
Nº records found: 13
Nº pages found: 1
Getting workday prorogations page 1/1
=====
GENERATING CSV
=====
Generated file: workday_prorogations.csv
Status: saved successfully.
File saved in path:
./src/main/resources/files/export/workday_prorogations.csv
=====
EXPORT WORKDAY PROROGATION FINISHED SUCCESSFULLY
=====
End of execution.
```

A aplicação fará requisições à API fSense para obter os acréscimos de jornadas, em seguida, serão gerados arquivos CSV correspondentes.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar os acréscimos de jornada exportados. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','source','personId','personName','personDeviceIdentifier','deviceId','deviceI
'3852','LEADER','6782700','maria','maria silva','','','','','2024-09-26','08:00:00
'3851','PERSON_REQUEST','6782700','maria','maria silva','73139','N-TEC-maria','','
'3850','PERSON_REQUEST','6782700','maria','maria silva','73139','N-TEC-maria','','
'3849','PERSON_REQUEST','6782700','maria','maria silva','73139','N-TEC-maria','','
'3848','LEADER','6782700','maria','maria silva','','','','','2024-09-25','10:00:00
'3847','LEADER','6782700','maria','maria silva','','','','','2024-09-25','10:00:00
```

## Exportar Acréscimos de Jornada - Exemplo de Implementação de Banco de Dados

Para exportar os acréscimos de jornada cadastrados na organização, fazendo a persistência de dados no banco, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `WORKDAY_PROROGATION_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=WORKDAY_PROROGATION_SERVICE
```

## Método do Serviço

Configure o método do serviço para `IMPORT_WORKDAY_PROROGATION` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=IMPORT_WORKDAY_PROROGATION
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=DATABASE
```

## Parâmetros de Exportação

É possível filtrar a exportação de acréscimos de jornada definindo parâmetros opcionais como:

- **personId** - Identificador único de uma pessoa cadastrada no fSense.
- **date** - Data Específica do acréscimo de Jornada.
- **status** - Status da solicitação de acréscimo de jornada.

```
#PARAMETROS PARA SERVIÇO DE PRORROGAÇÕES DE JORNADA
workday.prorogation.person.id=
##[dd/MM/yyyy]
workday.prorogation.date=
#[APPROVED, DENIED, PENDING, CANCELLED, EXPIRED]
workday.prorogation.status=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_PROROGATION_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_WORKDAY_PROROGATION]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
AUTHENTICATION SUCCESS
=====
Nº records found: 13
Nº pages found: 1
Getting workday prorogations page 1/1
=====
SAVING ON DATABASE
=====
Open database Connection
Saving on tb_workday_override
Update count: 13
Database connection Close
=====
SAVING DATABASE WORKDAY PROROGATIONS FINISHED SUCCESSFULLY
=====
End of execution.
```

## Verificação da Operação

Após a execução, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela `tb_workday_prorogation`.

```
SELECT * FROM tb_workday_prorogation;
```

A tabela terá uma estrutura semelhante à esta:

id	source	personId	personName	personDeviceIdentifier	device
3852	LEADER	6782700	maria	maria silva	
3851	PERSON_REQUEST	6782700	maria	maria silva	73139
3850	PERSON_REQUEST	6782700	maria	maria silva	73139
3849	PERSON_REQUEST	6782700	maria	maria silva	73139
3848	LEADER	6782700	maria	maria silva	
3847	LEADER	6782700	maria	maria silva	

## Importar Acréscimos de Jornada

Para importar acréscimos de jornada para o fSense, a partir de um arquivo CSV, podemos fazer da seguinte forma:

# Configuração do Arquivo Properties

## Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `WORKDAY_PROROGATION_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=WORKDAY_PROROGATION_SERVICE
```

## Método do Serviço

Configure o método do serviço para `IMPORT_WORKDAY_PROROGATION` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=IMPORT_WORKDAY_PROROGATION
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações dos acréscimos de jornadas a serem importadas através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório

`src/main/resources/import/workday_prorogation_service_import_workday_prorogation.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações dos acréscimos de jornada a serem importadas.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = workday_override_service_import_workday_prorogation.csv
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO  
csv.delimiter  
csv.delimiter = ;  
  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO  
csv.text.qualifier  
csv.text.qualifier = '
```

### **NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====  
SELECTED SERVICE [WORKDAY_PROROGATION_SERVICE]  
=====  
OPERATION TYPE: CSV  
=====  
SELECTED METHOD [IMPORT_WORKDAY_PROROGATION]  
=====
```

checking **if** token is valid?: false

=====

Starting authentication **process**

Getting access token

Token received.

844ccb65-8311-4e72-a5f2-6669862f6c71

=====

AUTHENTICATION SUCCESS

=====

Date: 2024-01-02

PersonId: 6757362

startTime: 07:30:00

EndTime: 17:30:00

ExtraTime: 01:00:00

Starting import workday prorogation

Workday Prorogation **for** date 2024-01-02 imported successfully

=====

Date: 2024-01-03

PersonId: 6757362

StartTime: 07:30:00

EndTime: 17:30:00

ExtraTime: 01:00:00

Starting import workday prorogation

Workday Prorogation **for** date 2024-01-03 imported successfully

=====

Date: 2024-01-04

PersonId: 6757362

StartTime: 07:30:00

EndTime: 17:30:00

ExtraTime: 01:00:00

Starting import workday prorogation

Workday Prorogation **for** date 2024-01-04 imported successfully

=====

Date: 2024-01-05

PersonId: 6757362

StartTime: 07:30:00

EndTime: 17:30:00

ExtraTime: 01:00:00

Starting import workday prorogation

Workday Prorogation **for** date 2024-01-05 imported successfully

=====

RESULT OF IMPORT OPERATION

=====

Total of requests made 4

Total of requests successfully: 4

```
Total of requests failed: 0
```

```
=====
```

```
End of execution.
```

A aplicação realizará as requisições à API fSense para importar os acréscimos de jornada no fSense.

Caso haja algum erro, será exibido uma mensagem de erro no console da aplicação. Certifique-se de que as informações estejam corretas e correspondam aos dados do arquivo CSV.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Acréscimos de Jornada** para verificar se os acréscimos de jornada foram inseridas corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Deletar Acréscimos de Jornada

Para deletar um acréscimo de jornada cadastrada no fSense, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço a ser executado definindo a propriedade `service.name` para `WORKDAY_PROROGATION_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=WORKDAY_PROROGATION_SERVICE
```

## Método do Serviço

Configure o método do serviço para `DELETE_WORKDAY_PROROGATION` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=DELETE_WORKDAY_PROROGATION
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Definição de Parâmetros de Remoção

Para remover uma ou mais acréscimos de jornada, é necessário informar o identificador único do acréscimo de jornada que deseja remover através da propriedade `workday.prorogation.ids`. É possível informar um ou mais ids, basta informá-los separados por vírgula.

```
##### IDS TROCAS DE JORNADA PARA REMOVER
workday.prorogation.ids=
```

## Execução da Aplicação

Definidos os parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [WORKDAY_PROROGATION_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [DELETE_WORKDAY_PROROGATION]
=====
```

```
workday proration Ids selected to deleted:
[305407, 305406, 305405, 305404, 305403, 305402, 305401, 305400, 305399,
305398, 305397, 305396, 305395]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
Authentication done successfully
=====
Try delete workday proration id: 305406 - 1/13
Removing workday prorogations
Workday proration removed successfully
=====
Try delete workday proration id: 305405 - 2/13
Removing workday prorogations
Workday proration removed successfully
=====
Try delete workday proration id: 305404 - 3/13
Removing workday prorogations
Workday proration removed successfully
=====
...
=====
Try delete workday proration id: 305395 - 13/13
Removing workday prorogations
Workday proration removed successfully
=====
RESULT OF DELETE OPERATION
=====
Total of requests made 13
Total of requests successfully: 13
Total of requests failed: 0
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online do fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Acréscimos de Jornada** para verificar se os

acréscimos de jornada foram removidas corretamente.

Caso haja algum erro durante a operação, fique atento às mensagens exibidas no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

# Ferriados

Serviço que permite internalizar a estrutura de jornadas da sua organização.

## Exportar Ferriados - Exemplo de Implementação CSV

Para exportar os feriados cadastradas na organização no formato CSV, podemos fazer da seguinte forma:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `HOLIDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=HOLIDAY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_HOLIDAY` através da propriedade `EXPORT_HOLIDAYS`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_HOLIDAY
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Parâmetros de Exportação

Para exportar os feriados, é preciso definir o ano que deseja realizar a exportação através da propriedade `holiday.year`. Opcionalmente é possível exportar determinado feriado filtrando pelo nome através da propriedade `holiday.name`.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE FERIADOS
holiday.name=Natal
holiday.year=2023
```

## Execução da Aplicação

Após a definição dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [HOLIDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_HOLIDAY]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
Authentication done successfully
=====
Nº records founded: 3
Nº pages found: 1
Getting holidays page: 1/1
=====
GENERATING CSV
=====
Generated file holidays.csv
Status: saved successfully
```

```
Path to saved file: ./src/main/resources/files/export/holidays.csv
```

```
=====
```

```
EXPORT HOLIDAYS FINISHED SUCCESSFULLY
```

```
=====
```

```
End of execution.
```

A aplicação fará requisições Http à API fSense para obter os feriados. Posteriormente, serão gerados arquivos CSV correspondentes. Os arquivos são armazenados no diretório **src/main/resources/files/export**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar se os feriados foram exportadas. O arquivo terá uma estrutura semelhante a do exemplo abaixo:

```
'id','name','type','date','canRepeat','year','month','day','calendarId'  
'317','Aniversario de Uberlândia','REGIONAL','2024-08-30  
21:00:00','true','2024','8','31','341'  
'1','Dia da Confraternização Universal','NATIONAL','2023-12-31  
21:00:00','true','2024','1','1','341'  
'313','Finados','NATIONAL','2024-11-01 21:00:00','false','2024','11','2','341'  
'312','Independência do Brasil','NATIONAL','2024-09-06  
21:00:00','true','2024','9','7','341'  
'316','Natal','NATIONAL','2024-12-24 21:00:00','true','2024','12','25','341'  
'315','NSa Aparecida','NATIONAL','2024-10-11  
21:00:00','true','2024','10','12','341'  
'314','Proclamação da República','NATIONAL','2024-11-14  
21:00:00','true','2024','11','15','341'  
'211','Tiradentes','NATIONAL','2024-04-19  
21:00:00','true','2024','4','20','341'
```

## Exportar Feriados - Exemplo de Implementação de Banco de Dados

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `HOLIDAY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=HOLIDAY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `EXPORT_HOLIDAY` através da propriedade `EXPORT_HOLIDAYS`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_HOLIDAY
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=database
```

## Parâmetros de Exportação

Para exportar os feriados, é preciso definir o ano que deseja realizar a exportação através da propriedade `holiday.year`. Opcionalmente é possível exportar determinado feriado filtrando pelo nome através da propriedade `holiday.name`.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE FERIADOS
holiday.name=Natal
holiday.year=2023
```

## Execução da Aplicação

Após a definição dos parâmetros, execute a aplicação.

```

=====
SELECTED SERVICE [HOLIDAY_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_HOLIDAY]
checking if token is valid?: false
=====

Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====

Authentication done successfully
=====

Nº records founded: 8
Nº pages found: 1
Getting holidays page: 1/1
=====

SAVING ON DATABASE
=====

Open database Connection
Saving on tb_holiday
UpdateCount: 8
Database connection close
=====

SAVING DATABASE HOLIDAYS FINISHED SUCCESSFULLY
=====

End of execution.

```

A aplicação fará requisições Http à API fSense para obter os feriados e em seguida, serão gerados persistidos no banco. Os arquivos são armazenados no diretório **src/main/resources/files/export**.

## Verificação da Operação

Abra a ferramenta de banco de dados da sua preferência e faça uma consulta na tabela **tb\_holiday**.

```
SELECT * FROM tb_holiday;
```

id	name	type	date	canRepeat	year	month	day
317	Aniversario de Uberlândia	REGIONAL	2024-08-30 21:00:00	true	2024	8	31
1	Dia da Confraternização Universal	NATIONAL	2023-12-31 21:00:00	true	2024	1	1
313	Finados	NATIONAL	2024-11-01 21:00:00	false	2024	11	2
312	Independência do Brasil	NATIONAL	2024-09-06 21:00:00	true	2024	9	7
316	Natal	NATIONAL	2024-12-24 21:00:00	true	2024	12	25
315	NSa Aparecida	NATIONAL	2024-10-11 21:00:00	true	2024	10	12
314	Proclamação da República	NATIONAL	2024-11-14 21:00:00	true	2024	11	15
211	Tiradentes	NATIONAL	2024-04-19 21:00:00	true	2024	4	20

## Importar Feriados

Para cadastrar um novo feriado no fSense, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `HOLIDAY_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=HOLIDAY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `IMPORT_HOLIDAY` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=IMPORT_HOLIDAY
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

### Informar o Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações dos feriados a serem importados através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório

`src/main/resources/files/import/holiday_service_import_holiday.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações dos feriados a serem importados.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = holiday_service_import_holiday.csv  
#CONFIGURAÇÕES DO CSV
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARAMETRO  
csv.delimiter  
csv.delimiter = ;  
  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARAMETRO  
csv.text.qualifier  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARAMETRO  
csv.text.qualifier  
csv.text.qualifier = '
```

### **NOTA**

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [HOLIDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [IMPORT_HOLIDAY]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
AUTHENTICATION SUCCESS
=====
holiday to be imported:
Name: Finados
Date: 2024-11-02T00:00:00Z
Type: NATIONAL
Can Repeat: false
holiday Finados import successfully
=====
HOLIDAY DATA
=====
Name: Proclamação da República
Date: 2024-11-15T00:00:00Z
Type: NATIONAL
Can Repeat: true
holiday Proclamação da República import successfully
=====
HOLIDAY DATA
=====
Name: NSa Aparecida
Date: 2024-10-12T00:00:00Z
Type: NATIONAL
Can Repeat: true
holiday NSa Aparecida import successfully
=====
HOLIDAY DATA
=====
Name: Natal
Date: 2024-12-25T00:00:00Z
Type: NATIONAL
Can Repeat: true
holiday Natal import successfully
```

```
=====
RESULT OF IMPORT OPERATION
=====
Finish of operation:
Total of requests made 5
Total of requests successfully: 5
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará requisições Http para a API fSense e importar os feriados para o fSense.

## Verificação da Exportação

Após a execução da Aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Feriados** e verifique se os feriados foram inseridos corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Atualizar Feriados

Para atualizar um feriado cadastrado no fSense, podemos fazer da seguinte maneira:

## Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `HOLIDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=HOLIDAY_SERVICE
```

## Método do Serviço

Configure o método do serviço para `UPDATE_HOLIDAY` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO
service.method=UPDATE_HOLIDAY
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO
operation.type=csv
```

## Informar o Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações dos feriados a serem atualizados através da propriedade `file.name`. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório

`src/main/resources/files/import/holiday_service_update_holiday.csv`. Preencha as informações conforme as colunas indicadas no modelo.

Configure o caminho para o diretório onde o arquivo CSV está localizado

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]
path.import = ./src/main/resources/files/import
```

Defina o nome do arquivo CSV que contém as informações dos feriados a serem atualizados.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]
file.name = holiday_service_update_holiday.csv
#CONFIGURAÇÕES DO CSV
```

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARAMETRO
csv.delimiter
csv.delimiter = ;

#UTILIZE [VAZIO] [ASPAS SIMPES] OU [ASPAS DUPLAS] PARA O PARAMETRO
csv.text.qualifier
csv.text.qualifier = '
```

### NOTA

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [HOLIDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [IMPORT_HOLIDAY]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
df1c90a6-6c2c-4f4f-a5bc-99e79e4ad3d7
=====
AUTHENTICATION SUCCESS
=====
holiday to be imported:
Name: Finados
Date: 2024-11-02T00:00:00Z
Type: NATIONAL
Can Repeat: false
holiday Finados import successfully
=====
```

```
HOLIDAY DATA
=====
Name: Proclamação da República
Date: 2024-11-15T00:00:00Z
Type: NATIONAL
Can Repeat: true
holiday Proclamação da República import successfully
=====
HOLIDAY DATA
=====
Name: NSa Aparecida
Date: 2024-10-12T00:00:00Z
Type: NATIONAL
Can Repeat: true
holiday NSa Aparecida import successfully
=====
HOLIDAY DATA
=====
ce.impl.csv.HolidayServiceCSVwiter:80 - Name: Natal
Date: 2024-12-25T00:00:00Z
Type: NATIONAL
Can Repeat: true
holiday Natal import successfully
=====
RESULT OF IMPORT OPERATION
=====
Finish of operation:
Total of requests made 5
Total of requests successfully: 5
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará requisições Http para a API fSense e atualizar os feriados para o fSense.

## Verificação da Exportação

Após a execução da Aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Feriados** e verifique se os feriados foram atualizados corretamente.

Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Exportar Feriados por ID

Para exportar um novo feriado pelo ID, para um arquivo CSV, podemos fazer da seguinte maneira:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `HOLIDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=HOLIDAY_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_HOLIDAY_BY_ID` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_HOLIDAY_BY_ID
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

É preciso definir o id do feriado a ser exportado no arquivo **fsense.properties**.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE FERIADOS
holiday.id=
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [HOLIDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_HOLIDAY_BY_ID]
=====
holiday id selected to export [213]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
8232e9dd-9ee3-4a94-b571-2e56c338eb76
=====
AUTHENTICATION SUCCESS
=====
GENERATING CSV
=====
Generating file holiday_id_213.csv
File: holiday_by_id_213.csv saved successfully
Path to saved file: ./src/main/resources/files/export/
Exporting holiday by id 213 completed successfully
=====
EXPORT HOLIDAY BY ID FINISHED SUCCESSFULLY
=====
End of execution.
```

A aplicação fará requisição para a API fSense para obter o feriado por ID. Posteriormente, será gerado o arquivo CSV correspondente.

A aplicação exportará os arquivos com nome **holiday\_[HOLIDAY\_ID].csv** .

Os arquivos são armazenados no diretório **src/main/resources/files/export**

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','name','type','date','canRepeat','year','month','day','calendarId'  
'317','Aniversario de Uberlândia','REGIONAL','0024-08-30  
21:00:00','true','24','8','31','341'
```

## Deletar Feriados por ID

Para deletar um feriado cadastrada no fSense, podemos fazer a seguinte maneira:

### Configuração do Arquivo Properties

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `HOLIDAY_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=HOLIDAY_SERVICE
```

### Método do Serviço

Configure o método do serviço para `DELETE_HOLIDAY` através da propriedade `service.method`.

```
# METODO DO SERVIÇO A SER EXECUTADO  
service.method=DELETE_HOLIDAY
```

## Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Definição de Parâmetros de Remoção

Para remover um ou mais feriados, é necessário informar o identificador único dos feriados que deseja remover através da propriedade `team.ids`. É possível informar um ou mais ids, basta informá-los separados por vírgula.

```
##ID DOS FERIADOS PARA REMOVER  
holiday.ids=213, 214, 215, 216, 217
```

## Execução da Aplicação

Após a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [HOLIDAY_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [DELETE_HOLIDAY_BY_ID]
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
37dcabc9-19c1-45e7-923a-7b091da89d30
=====
AUTHENTICATION SUCCESS
=====
Id holidays selected to remove:
```

```
[213, 214, 215, 216, 217]
Removing holiday id: 213
starting delete holiday by id
Holiday id 213 deleted successfully
=====
Removing holiday id: 214
starting delete holiday by id
Holiday id 214 deleted successfully
=====
Removing holiday id: 215
starting delete holiday by id
Holiday id 215 deleted successfully
=====
Removing holiday id: 216
starting delete holiday by id
Holiday id 216 deleted successfully
=====
Removing holiday id: 217
starting delete holiday by id
Holiday id 217 deleted successfully
=====
RESULT OF OPERATION
=====
Total of requests made 0
Total of requests successfully: 5
Total of requests failed: 0
=====
End of execution.
```

A aplicação realizará as requisições Http para a API fSense para remover os feriados no fSense.

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Feriados** para verificar se os feriados foram removidos corretamente.

Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

# Serviço de Teclas/Mouse

Serviço que permite a busca dos dados capturados de pressionamentos de teclas e cliques de mouse sendo possível o filtro por data ou por usuários.

## Exportar Teclas/Mouse - Exemplo de Implementação CSV

### Configuração do Arquivo Properties

Para exportar as categorias da organização no formato CSV podemos fazer da seguinte forma

### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado ajustando a propriedade `service.name` para `KEYSTROKES_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=KEYSTROKES_SERVICE
```

### Método do Serviço

Configure o método do serviço para `EXPORT_KEYSTROKES` através da propriedade `service.method`.

```
# SERVICIO A SER EXECUTADO
service.method=EXPORT_KEYSTROKES
```

### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE OPERAÇÃO  
operation.type=csv
```

## Parâmetros de Exportação

É possível definir um período de data para a busca dos dados de Teclas/Mouse através da propriedades `keystrokes.startDate` e `keystrokes.endDate`.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE TECLAS/MOUSE (KEYSTROKES)  
#[dd/MM/yyyy]  
keystrokes.startDate=01/06/2024  
#[dd/MM/yyyy]  
keystrokes.endDate=13/06/2024
```

Também é possível filtrar por um ou mais usuários como filtro para busca dos dados de Teclas/Mouse. Basta informar o nome do usuário que deseja filtrar no parâmetro `keystrokes.usernames`. Se desejar mais de um usuário, basta passar os nomes separados por vírgula, como no exemplo abaixo:

```
#[EXEMPLO: John, Bob, Charlie]  
keystrokes.usernames=
```

## Execução da Aplicação

Após realizada a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [KEYSTROKES_SERVICE]
=====
OPERATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_KEYSTROKES]
Executing export Keystrokes
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
```

```
Token received.
87caa16b-f418-487d-bb43-1832dc93e2e0
=====
AUTHENTICATION SUCCESSFULLY
=====
KEYSTROKES DATA
=====
Getting keystrokes for day: 2024-06-03
Nº of records found: 502
Nº of pages: 6
Get keystrokes 1/6
Get keystrokes 2/6
Get keystrokes 3/6
Get keystrokes 4/6
Get keystrokes 5/6
Get keystrokes 6/6
=====
GENERATING CSV
=====
File generated: keystrokes_2024_06_04.csv
Status: saved successfully.
File saved in path:
./src/main/resources/files/export/keystrokes_2024_06_03.csv
Export keystrokes of day 2024-06-03 completed successfully.
=====
EXPORT KEYSTROKES FINISHED SUCCESSFULLY
=====
End of execution.
```

Os arquivos são armazenados no diretório `scr/main/resources/files/export/`.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar se as captura de teclas/mouse foram exportadas. O arquivo terá uma estrutura semelhante a do exemplo abaixo:

```
'machineName','userName','creationDate','personId','deviceId','keystrokesCount','l
'N-TEC-FSENSE_USER','fsense_user','2024-09-09T20:26:09.736Z','6744454','54285','0
'N-TEC-FSENSE_USER','fsense_user','2024-09-09T20:26:10.381Z','6744454','54285','4:
09T11:00:59.999Z','57'
'N-TEC-FSENSE_USER','fsense_user','2024-09-09T20:26:29.205Z','6744454','54285','3:
```

```
09T11:01:59.999Z', '59'  
'N-TEC-FSENSE_USER', 'fsense_user', '2024-09-09T20:26:29.205Z', '6744454', '54285', '0  
'N-TEC-FSENSE_USER', 'fsense_user', '2024-09-09T20:26:49.193Z', '6744454', '54285', '0  
'N-TEC-FSENSE_USER', 'fsense_user', '2024-09-09T20:26:49.193Z', '6744454', '54285', '3:  
09T11:04:59.999Z', '59'  
'N-TEC-FSENSE_USER', 'fsense_user', '2024-09-09T20:27:09.271Z', '6744454', '54285', '0
```

## Exportar Teclas/Mouse - Exemplo de Implementação de Banco de Dados

Para exportar os dados de Teclas/Mouse, fazendo a persistência das informações no banco de dados, podemos fazer da seguinte forma:

### Configuração do Arquivo Properties

#### Definição do Serviço

No arquivo `fsense.properties`, configure o serviço que será executado definindo a propriedade `service.name` para `KEYSTROKES_SERVICE`.

```
# SERVICIO A SER EXECUTADO  
service.name=KEYSTROKES_SERVICE
```

#### Método do Serviço

Configure o método do serviço para `EXPORT_KEYSTROKES` através da propriedade `service.method`.

```
# SERVICIO A SER EXECUTADO  
service.method=EXPORT_KEYSTROKES
```

#### Tipo de Operação

Especifique o tipo de operação desejado através da propriedade `operation.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE OPERAÇÃO
operation.type=database
```

## Parâmetros de Exportação

É possível definir um período de data para a busca dos dados de Teclas/Mouse através da propriedades `keystrokes.startDate` e `keystrokes.endDate`.

```
#PARAMETROS PARA SERVIÇO DE EXPORTAÇÃO DE TECLAS/MOUSE (KEYSTROKES)
#[dd/MM/yyyy]
keystrokes.startDate=01/06/2024
#[dd/MM/yyyy]
keystrokes.endDate=13/06/2024
```

Também é possível filtrar por um ou mais usuários como filtro para busca dos dados de Teclas/Mouse. Basta informar o nome do usuário que deseja filtrar no parâmetro `keystrokes.username`. Se desejar mais de um usuário, basta passar os nomes separados por vírgula, como no exemplo abaixo:

```
#[EXEMPLO: John, Bob, Charlie]
keystrokes.username=
```

## Execução da Aplicação

Após realizada a configuração dos parâmetros, execute a aplicação.

```
=====
SELECTED SERVICE [KEYSTROKES_SERVICE]
=====
OPERATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_KEYSTROKES]
Executing export Keystrokes
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
```

```
Token received.
87caa16b-f418-487d-bb43-1832dc93e2e0
=====
AUTHENTICATION SUCCESSFULLY
=====
KEYSTROKES DATA
=====
Getting keystrokes for day: 2024-06-03
Nº of records found: 502
Nº of pages: 6
Get keystrokes 1/6
Get keystrokes 2/6
Get keystrokes 3/6
Get keystrokes 4/6
Get keystrokes 5/6
Get keystrokes 6/6
=====
SAVING ON DATABASE
=====
Open database Connection
persisting on tb_keystrokes
=====
Update Count: 502
Database connection close
=====
SAVING DATABASE KEYSTROKES FINISHED SUCCESSFULLY
=====
End of execution.
```

A aplicação fará requisições para a API fSense para obter os dados. Posteriormente, os dados serão persistidos no banco.

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_keystrokes**.

```
SELECT * FROM tb_keystrokes
```

A tabela terá uma estrutura semelhante à esta:

<b>machineName</b>	<b>userName</b>	<b>creationDate</b>	<b>personId</b>	<b>deviceId</b>	<b>keystrokesC</b>
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:26:09.736Z	6744454	54285	0
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:26:10.381Z	6744454	54285	42
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:26:29.205Z	6744454	54285	32
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:26:29.205Z	6744454	54285	0
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:26:49.193Z	6744454	54285	0
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:26:49.193Z	6744454	54285	32
N-TEC-FSENSE_USER	fsense_user	2024-09-09T20:27:09.271Z	6744454	54285	0

# Serviço de Calendário

Este guia fornece exemplos de como internalizar a estrutura de calendário da sua organização, bem como exportar os calendários cadastrados no formato **CSV**.

## Exportar Calendários - Exemplo de Implementação CSV

Para exportar os calendários cadastrados na organização no formato **.csv**, podemos fazer da seguinte maneira.

### Configuração do Arquivo `fsense.properties`.

#### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=CALENDAR_SERVICE
```

#### 2. Método do Serviço

Configure o método do serviço para `EXPORT_CALENDARS` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_CALENDARS
```

#### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO
export.type=csv
```

#### 4. Parâmetros de Exportação

É possível aplicar um filtro opcional por nome do calendário na exportação de calendários. Defina o parâmetro `calendar.name` como nome ou parte do nome do calendário que deseja exportar.

```
# PARÂMETRO PARA EXPORTAÇÃO DE CALENDÁRIO
calendar.name=Calendário Padrão
```

### Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará um requisição HTTP à API fSense para obter os calendários e gerará um arquivo CSV como nome `calendars.csv`.

```
=====
SELECTED SERVICE [CALENDAR_SERVICE]
=====
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_CALENDARS]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1cdb6c76-0dab-4912-a1fa-771da67a1c55
=====
AUTHENTICATION SUCCESS
=====
Nº calendars found: 6
Nº Pages found: 1
Getting calendars page: 1/1
=====
GENERATING CSV FILE
=====
File generated: calendars.csv
```

```
Status: saved successfully
File saved in path: ./src/main/resources/files/export/calendars.csv
=====
EXPORT CALENDARS FINISHED SUCCESSFULLY
=====
End of execution.
```

Os arquivos são armazenados no diretório **src/main/resources/files/export/**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verificar as calendários exportadas. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id','externalCode','name'
'342','','Padrão'
'1016','','Calendário Alternativo'
'1045','','Novo Calendário'
'1048','a9cabe7401d1b83b9f63a906e5aff','Calendário com Código Externo'
```

## Exportar Calendários - Exemplo de Implementação de Banco de Dados

Para exportar os calendários cadastrados na organização, fazendo a persistência no banco de dados, podemos fazer da seguinte maneira.

### Configuração do Arquivo `fsense.properties`.

#### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=CALENDAR_SERVICE
```

#### 2. Método do Serviço

Configure o método do serviço para `EXPORT_CALENDARS` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_CALENDARS
```

### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `DATABASE`.

```
# TIPO DE EXPORTAÇÃO
export.type=database
```



#### DICA

Certifique-se que o banco de dados está devidamente configurado e a tabela previamente criada.

Se tiver dúvidas, consulte a seção [Integração com Banco de Dados](#).

### 4. Parâmetros de Exportação

É possível aplicar um filtro opcional por nome do calendário na exportação de calendários. Defina o parâmetro `calendar.name` como nome ou parte do nome do calendário que deseja exportar.

```
# PARÂMETRO PARA EXPORTAÇÃO DE CALENDÁRIO
calendar.name=Calendário Padrão
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará um requisição HTTP à API fSense para obter os calendários e em seguida, fará a persistência dos dados no banco.

```

=====
SELECTED SERVICE [CALENDAR_SERVICE]
=====
EXPORTATION TYPE: DATABASE
=====
SELECTED METHOD [EXPORT_CALENDARS]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
f109f26d-0f50-4e32-a33c-7772e1ee5014
=====
AUTHENTICATION SUCCESS
=====
Nº calendars found: 3
Nº Pages found: 1
Getting calendars page: 1/1
Open database connection
Saving on tb_calendars
=====
Update Count: 3
Database connection close
=====
End of execution.

```

## Verificação da Operação

Após a execução da aplicação, abra a ferramenta de banco de dados de sua preferência e faça uma consulta na tabela **tb\_calendar**.

```
SELECT * FROM tb_calendar;
```

A tabela terá uma estrutura semelhante à esta:

id	name	external_code
342	Padrão	-

id	name	external_code
1016	Calendário Alternativo	-
1051	Calendário Teste Importação	-

## Exportar Calendário por Código Externo

Para fazer a exportação de um calendário por código externo, podemos fazer da seguinte maneira.

### Configuração do Arquivo `fsense.properties`.

#### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=CALENDAR_SERVICE
```

#### 2. Método do Serviço

Configure o método do serviço para `EXPORT_CALENDAR_BY_EXTERNAL_CODE` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_CALENDAR_BY_EXTERNAL_CODE
```

#### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO
export.type=csv
```

#### 4. Parâmetros de Exportação

Para exportar um calendário, é preciso preencher o código do externo do calendário que deseja exportar. Defina o parâmetro `calendar.externalcode` com o valor correspondente ao código externo do calendário.

```
# PARÂMETRO PARA EXPORTAÇÃO DE CALENDÁRIO
calendar.externalcode=a9cabe7401d1b83b9faaaa906e5afff
```

#### NOTA

Este campo é obrigatório e deve ser preenchido com o valor do código externo do calendário que você deseja exportar. Se o código externo não for especificado, a exportação não será realizada.

### Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará uma requisição HTTP à API fSense para obter os calendário pelo código externo e gerará um arquivo CSV como nome `calendar_external_code_{codigo externo}.csv`. Durante a execução, a aplicação gerará um log semelhante a este.

```
=====
SELECTED SERVICE [CALENDAR_SERVICE]
=====
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_CALENDAR_BY_EXTERNAL_CODE]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1cdb6c76-0dab-4912-a1fa-771da67a1c55
=====
```

```
AUTHENTICATION SUCCESS
=====
GENERATING CSV FILE
=====
File generated: calendar_external_code_a9cabe7401d1b83b9f63a906e5a6e.csv
Status: saved successfully
File saved in path:
./src/main/resources/files/export/calendar_external_code_a9cabe7401d1b83b9f63a906
=====
EXPORT CALENDAR BY EXTERNAL CODE FINISHED SUCCESSFULLY
=====
End of execution.
```

Os arquivos são armazenados no diretório **src/main/resources/files/export/**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verifica se o calendário por código externo foi exportado. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id', 'externalCode', 'name'
'1048', 'a9cabe7401d1b83b9f63a906e5a6e', 'Calendário com Código Externo'
```

## Exportar Calendário por Id

Para fazer a exportação de um calendário por id, podemos fazer da seguinte maneira.

### Configuração do Arquivo `fsense.properties`.

#### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=CALENDAR_SERVICE
```

## 2. Método do Serviço

Configure o método do serviço para `EXPORT_CALENDAR_BY_ID` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=EXPORT_CALENDAR_BY_ID
```

## 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO  
export.type=csv
```

## 4. Parâmetros de Exportação

Para exportar um calendário, é preciso preencher o id (identificador único) do calendário que deseja exportar. Defina o parâmetro `calendar.id` com o valor correspondente ao identificador único do calendário.

```
# PARÂMETRO PARA EXPORTAÇÃO DE CALENDÁRIO  
calendar.id=1016
```

### NOTA

Este campo é obrigatório e deve ser preenchido com o valor do id do calendário que você deseja exportar. Se o código externo não for especificado, a exportação não será realizada.

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará uma requisição HTTP à API fSense para obter os calendário pelo id e gerará um arquivo

CSV como nome `calendar_id_{ identificador único }.csv`. Durante a execução, a aplicação gerará um log semelhante a este.

```
=====
SELECTED SERVICE [CALENDAR_SERVICE]
=====
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [EXPORT_CALENDAR_BY_ID]
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1cdb6c76-0dab-4912-a1fa-771da67a1c55
=====
AUTHENTICATION SUCCESS
=====
GENERATING CSV FILE
=====
File generated: calendar_id_1016.csv
Status: saved successfully
File saved in path: ./src/main/resources/files/export/calendar_id_1016.csv
=====
EXPORT CALENDAR BY ID FINISHED SUCCESSFULLY
=====
End of execution.
```

Os arquivos são armazenados no diretório **src/main/resources/files/export/**.

## Verificação da Operação

Após a execução da aplicação, abra o arquivo gerado para verifica se o calendário por id foi exportado. O arquivo exportado terá uma estrutura semelhante à do exemplo abaixo:

```
'id', 'externalCode', 'name'
'1016', '', 'Calendário Alternativo'
```

## Importar Calendário

Para importar calendários no fSense, a partir de um arquivo **CSV**, podemos fazer da seguinte maneira:

### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVIÇO A SER EXECUTADO
service.name=CALENDAR_SERVICE
```

### 2. Método do Serviço

Configure o método do serviço para `IMPORT_CALENDAR` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=IMPORT_CALENDAR
```

### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO
export.type=csv
```

### 4. Definição do Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações dos feriados a serem importados. Se não houver nenhum arquivo pronto, há um modelo disponível no diretório `src/main/resources/files/import/holiday_service_import_holiday.csv`. Preencha as informações conforme as colunas indicadas no modelo. Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]
path.import = ./src/main/resources/files/import
```

## 5. Nome do Arquivo

Defina o nome do arquivo CSV que contém as informações dos calendários a serem importados.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]
file.name = holiday_service_import_holiday.csv
```

## 6. Configurações do CSV

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV

#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO csv.delimiter
csv.delimiter = ,

#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO
csv.text.qualifier
csv.text.qualifier = '
```

### NOTA

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execute a aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará a leitura do arquivo CSV, processará os dados e importará os calendários para o fSense.

```
=====
SELECTED SERVICE [CALENDAR_SERVICE]
=====
```

```
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [IMPORT_CALENDAR]
IMPORT DATA 1/1
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1cdb6c76-0dab-4912-a1fa-771da67a1c55
=====
AUTHENTICATION SUCCESS
=====
{id: 1051, externalCode: null, name: Calendário Teste Importação}
=====
RESULT OF IMPORT OPERATION
=====
Finish of operation:
Total of requests made: 1
Total of requests successfully: 1
Total of requests failed: 0
=====
```

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Calendário** para verificar se os calendários foram importados corretamente. Caso haja algum erro durante a importação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Modelo de Arquivo de CSV para importação

Se não houver um arquivo CSV pronto, utilize o modelo disponível no diretório `src/main/resources/files/import/calendar_service_import_calendar.csv`. Preencha as informações conforme as colunas indicadas no modelo. O arquivo tem uma estrutura semelhante ao exemplo abaixo:

```
'id', 'externalCode', 'name'
```

```
','','Calendário Teste Importação'
```

## Atualizar Calendário

Para atualizar calendários no fSense através do exemplo de integração, podemos fazer da seguinte maneira:

### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVIÇO A SER EXECUTADO  
service.name=CALENDAR_SERVICE
```

### 2. Método do Serviço

Configure o método do serviço para `UPDATE_CALENDAR` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=UPDATE_CALENDAR
```

### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO  
export.type=csv
```

### 4. Definição do Caminho do Arquivo

É necessário informar o caminho do arquivo CSV contendo as informações dos feriados a serem atualizados. Se não houver nenhum arquivo pronto, há um modelo disponível no

diretório **src/main/resources/files/import/holiday\_service\_update\_holiday.csv**. Preencha as informações conforme as colunas indicadas no modelo. Configure o caminho para o diretório onde o arquivo CSV está localizado.

```
#CAMINHO PARA IMPORTAR O ARQUIVO [O CAMINHO DEVE ESTAR PREVIAMENTE CRIADO]  
path.import = ./src/main/resources/files/import
```

## 5. Nome do Arquivo

Defina o nome do arquivo CSV que contém as informações dos calendários a serem atualizados.

```
#NOME DO ARQUIVO A SER IMPORTADO [file.name=team_import.csv]  
file.name = holiday_service_update_holiday.csv
```

## 6. Configurações do CSV

Configure o delimitador e o qualificador de texto utilizados no arquivo CSV.

```
#CONFIGURAÇÕES DO CSV  
  
#UTILIZE [VIRGULA] [PONTO E VIRGULA] OU [VAZIO] PARA O PARÂMETRO csv.delimiter  
csv.delimiter = ;  
  
#UTILIZE [VAZIO] [ASPAS SIMPLES] OU [ASPAS DUPLAS] PARA O PARÂMETRO  
csv.text.qualifier  
csv.text.qualifier = '
```

### NOTA

Certifique-se que o arquivo CSV esteja no formato correto e que os parâmetros de configurações correspondam ao formato do arquivo.

## Execute a aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará a leitura do arquivo CSV, processará os dados e atualizará os calendários no fSense.

```

=====
SELECTED SERVICE [CALENDAR_SERVICE]
=====
EXPORTATION TYPE: CSV
=====
SELECTED METHOD [UPDATE_CALENDAR]
IMPORT DATA 1/1
=====
checking if token is valid?: false
=====
Starting authentication process
Getting access token
Token received.
1cdb6c76-0dab-4912-a1fa-771da67a1c55
=====
AUTHENTICATION SUCCESS
=====
{id: 1051, externalCode: null, name: Calendário Teste Importação ATUALIZADO}
=====
RESULT OF UPDATE OPERATION
=====
Finish of operation:
Total of requests made: 1
Total of requests successfully: 1
Total of requests failed: 0
=====

```

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Calendário** para verificar se os calendários foram atualizados corretamente. Caso haja algum erro durante a atualização, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Modelo de Arquivo de CSV para importação

Se não houver um arquivo CSV pronto, utilize o modelo disponível no diretório `src/main/resources/files/import/calendar_service_update_calendar.csv`. Preencha as informações conforme as colunas indicadas no modelo. O arquivo tem uma estrutura semelhante ao exemplo abaixo:

```
'id', 'externalCode', 'name'  
'1051', '', 'Calendário Teste Importação ATUALIZADO'
```

## Remover Calendário

Para remover calendários no fSense pela aplicação exemplo de integração, podemos fazer da seguinte forma:

### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
#SERVIÇO A SER EXECUTADO  
service.name=CALENDAR_SERVICE
```

### 2. Método do Serviço

Configure o método do serviço para `DELETE_CALENDAR_BY_ID` através da propriedade `service.method`.

```
#MÉTODO DO SERVIÇO A SER EXECUTADO  
service.method=DELETE_CALENDAR_BY_ID
```

### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO  
export.type=csv
```

### 4. Definição de Parâmetros de Remoção

Para remover um ou mais calendários, é necessário informar o identificador único do calendário que deseja remover através da propriedade `calendar.remove.ids`. É possível

informar um ou mais ids, basta informá-los separados por vírgula.

```
#ID DOS CALENDÁRIOS PARA REMOVER  
calendar.remove.ids=1016,1045,1051
```

## Execução da Aplicação

Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará uma requisição HTTP à API fSense com os ids marcados para remoção. Durante a execução, a aplicação gerará um log semelhante a este.

```
=====  
SELECTED SERVICE [CALENDAR_SERVICE]  
=====  
EXPORTATION TYPE: CSV  
=====  
SELECTED METHOD [DELETE_CALENDAR_BY_ID]  
=====  
checking if token is valid?: false  
=====  
Starting authentication process  
Getting access token  
Token received.  
f109f26d-0f50-4e32-a33c-7772e1ee5014  
=====  
AUTHENTICATION SUCCESS  
=====  
CALENDAR ID TO BE REMOVED: 1045 - 1/5  
Calendar id 1045 delete successfully  
=====  
CALENDAR ID TO BE REMOVED: 1046 - 2/5  
=====  
Calendar id 1046 delete successfully  
=====  
CALENDAR ID TO BE REMOVED: 1047 - 3/5  
=====  
Calendar id 1047 delete successfully  
=====  
CALENDAR ID TO BE REMOVED: 1048 - 4/5  
=====  
Calendar id 1048 delete successfully  
=====
```

```
CALENDAR ID TO BE REMOVED: 1051 - 5/5
=====
Calendar id 1051 delete successfully
=====
RESULT OF DELETE OPERATION
=====
Finish of operation:
Total of requests made: 5
Total of requests successfully: 5
Total of requests failed: 0
=====
End of execution.
```

## Verificação da Operação

Após a execução da aplicação, abra o Painel de Gestão Online fSense no seu navegador e faça login. Navegue até a seção **Jornadas > Calendário** para verificar se os calendários foram removidos corretamente. Caso haja algum erro durante a operação, uma mensagem de erro será exibida no console da aplicação. Verifique as mensagens de erro para solucionar possíveis problemas.

## Exportar Associações de Calendário

Para fazer a exportação das associações de calendário, podemos fazer da seguinte maneira.

### Configuração do Arquivo `fsense.properties`.

#### 1. Definição do Serviço

No arquivo `fsense.properties`, defina o serviço a ser executado ajustando a propriedade `service.name` para `CALENDAR_SERVICE`.

```
# SERVICIO A SER EXECUTADO
service.name=CALENDAR_SERVICE
```

#### 2. Método do Serviço

Configure o método do serviço para `EXPORT_CALENDAR_ASSOCIATION` através da propriedade `service.method`.

```
# MÉTODO DO SERVIÇO A SER EXECUTADO
service.method=EXPORT_CALENDAR_ASSOCIATION
```

### 3. Tipo de Exportação

Especifique o tipo de exportação desejado através da propriedade `export.type`. Defina o como `CSV` para exportar para um arquivo CSV ou como `DATABASE` para exportar para um banco de dados. Para esta implementação, defina `CSV`.

```
# TIPO DE EXPORTAÇÃO
export.type=csv
```

### 4. Parâmetros de Exportação

Para exportar as associações de calendário, é preciso preencher o id (identificador único) do calendário que deseja exportar as associações. Defina o parâmetro `calendar.id` com o valor correspondente ao identificador único.

```
# PARÂMETRO PARA EXPORTAÇÃO DE CALENDÁRIO
calendar.id=342
```

#### NOTA

Este campo é obrigatório e deve ser preenchido com o valor do id do calendário que você deseja exportar as associações. Se o identificador único não for especificado, a exportação não será realizada.

## Execução da Aplicação

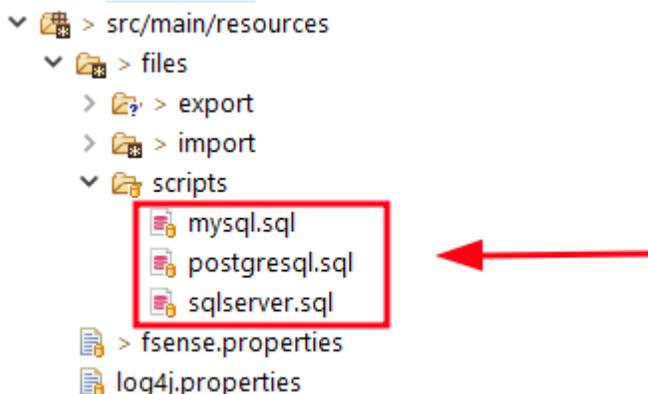
Após configurar o arquivo `fsense.properties`, execute a aplicação. A aplicação realizará uma requisição HTTP à API fSense para obter as associações de calendário e gerará um arquivo CSV como nome `calendar_id_{identificador único}_associations.csv`. Durante a execução, a aplicação gerará um log semelhante a este.



# Integração com Banco de Dados

## Criação das Tabelas

- Preparação do Ambiente Antes de iniciar é fundamental garantir que o servidor de banco de dados esteja devidamente instalado.
- Compatibilidade Esta integração é compatível com os bancos de dados MySQL, PostgreSQL e SQL Server. Para outros bancos de dados relacionais, podem ser necessárias adaptações específicas.
- Cliente de Banco de Dados Certifique-se de ter um cliente de banco de dados instalado em seu computador, como o MySQL Workbench (comumente usado com o MySQL), o pgAdmin (para PostgreSQL), ou o DBeaver, capaz de administrar diversos bancos de dados. No entanto, você pode optar pelo sistema gerenciador de banco de dados de sua preferência.
- Script SQL para Criação das tabelas Há um script SQL está disponível para a criação do banco de dados e das tabelas, adaptado para cada um dos bancos relacionais suportados mencionados acima. Este script se encontra no diretório do projeto `src/main/resources/files/scripts`



- Execução do Script Abra o script correspondente ao banco desejado em um editor de textos de sua escolha e copie o conteúdo. O conteúdo tem a estrutura semelhante ao exibido abaixo:

```

CREATE DATABASE db_fsense;

use db_fsense;

CREATE TABLE tb_team (
    id BIGINT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    team_parent_id BIGINT,
    team_parent_name VARCHAR(255),
    FOREIGN KEY (team_parent_id) REFERENCES tb_team(id)
);

CREATE INDEX idx_team_parent_id ON tb_team(team_parent_id);

CREATE TABLE tb_person (
    id BIGINT PRIMARY KEY,
    name VARCHAR(255),
    email VARCHAR(255),
    device_identifier VARCHAR(255),
    team_id BIGINT,
    team_name VARCHAR(255),
    can_delete BOOLEAN,
    root BOOLEAN,

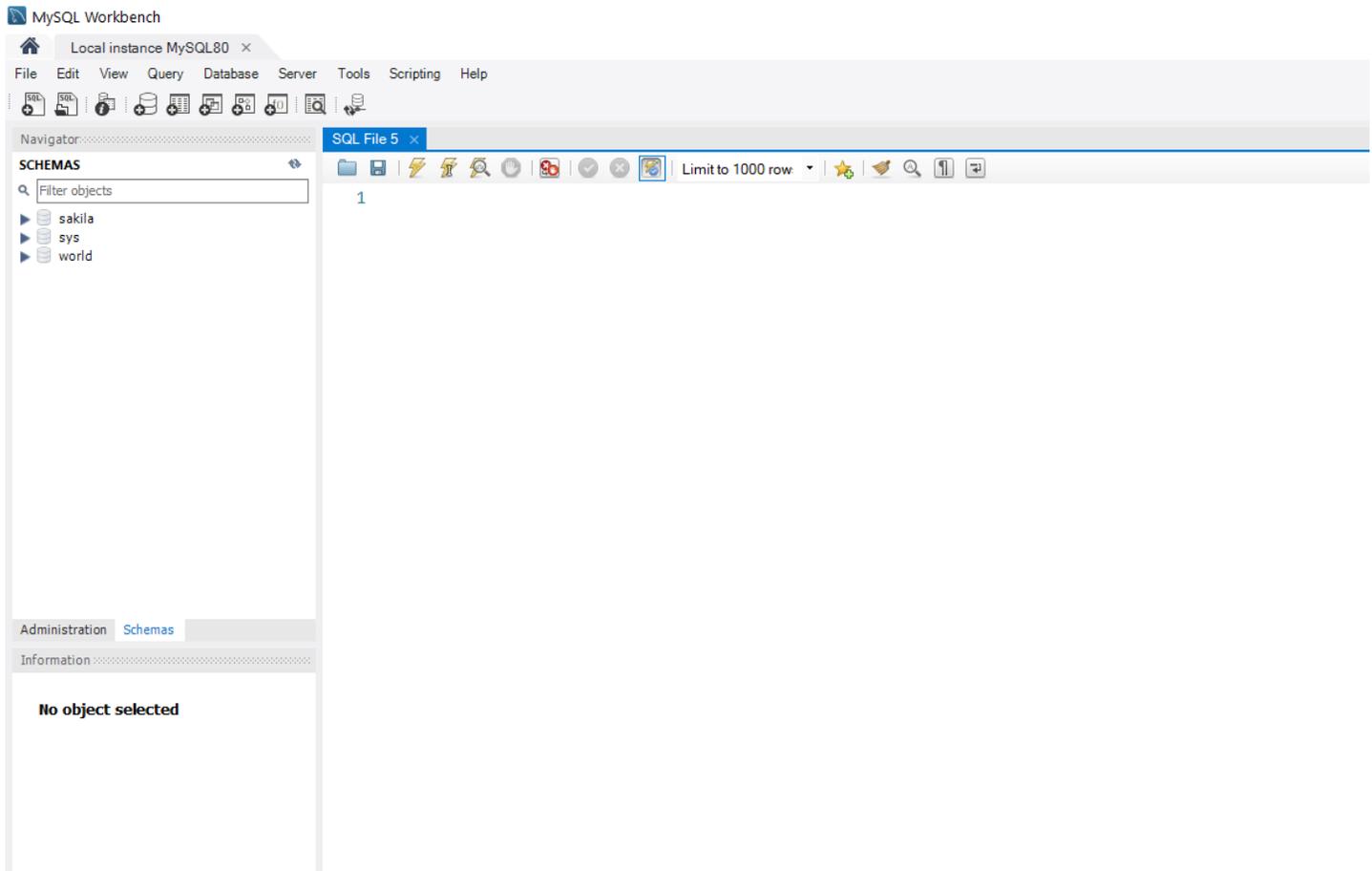
    FOREIGN KEY (team_id) REFERENCES tb_team(id)
);

CREATE TABLE tb_category (
    id BIGINT PRIMARY KEY auto_increment,
    application_name VARCHAR(255),
    category CHAR(1),
    category_name VARCHAR(255),
    username VARCHAR(255)
);

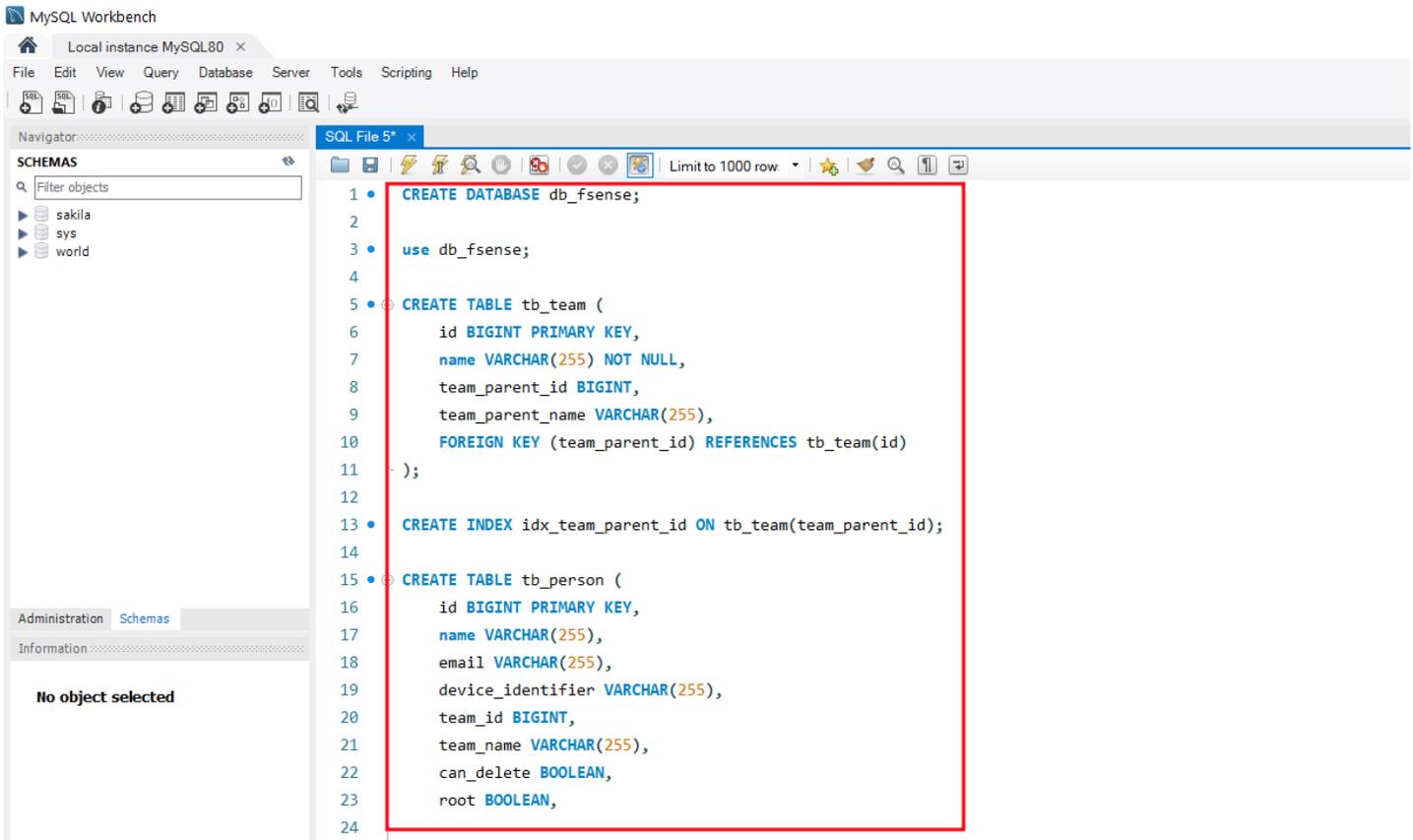
...

```

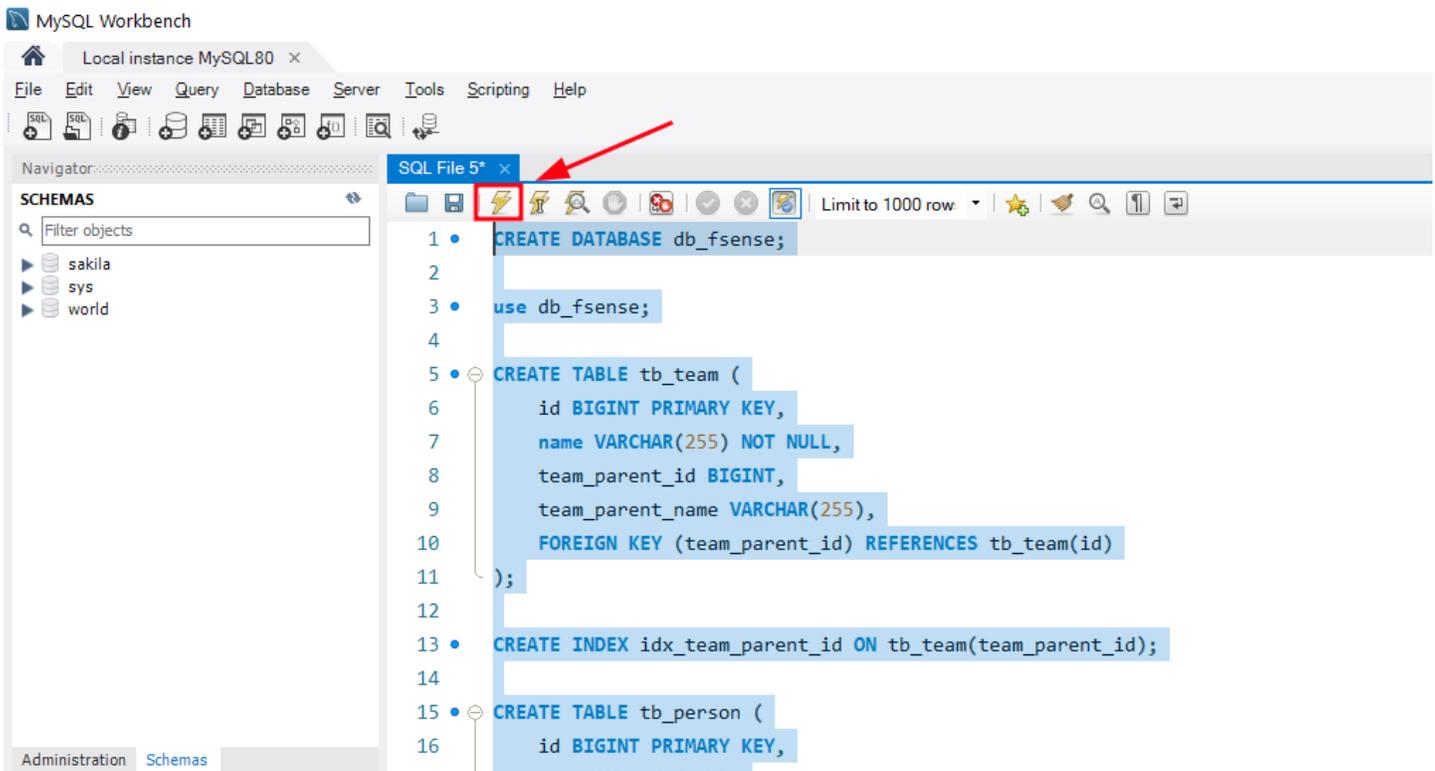
- Uso do Sistema Gerenciador de Banco de Dados Em seguida, abra o sistema gerenciador de banco de dados de sua preferência. Por exemplo, utilizaremos o MySQL Workbench neste exemplo.



- Inserção do Script no Editor Cole o conteúdo do script SQL no editor do Workbench.



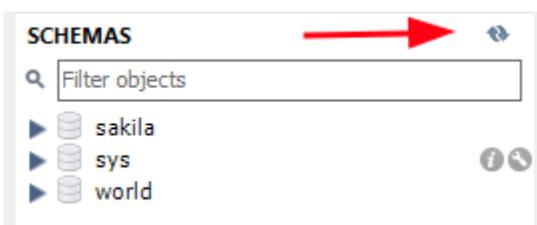
- Execução e Verificação Selecione todo o conteúdo do script no editor do Workbench utilizando as teclas CTRL + A e, em seguida, clique no botão **Executar**, representado por um ícone de raio, conforme mostrado na imagem abaixo.



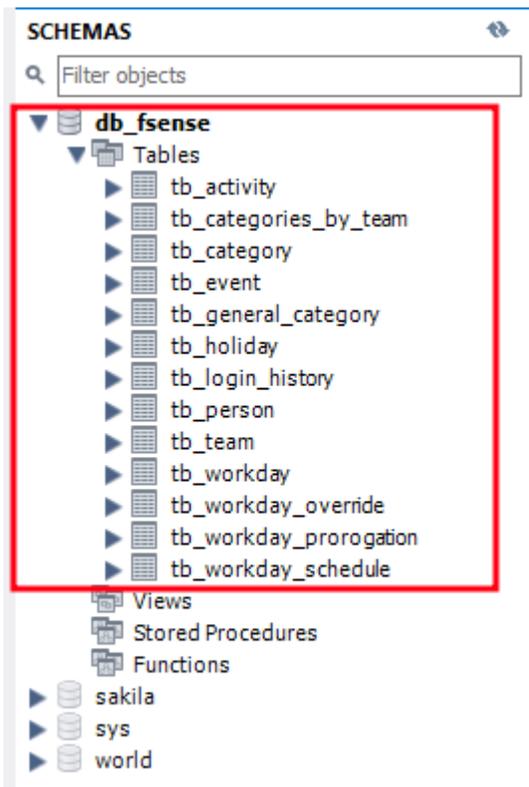
- Confirmação da Execução Se tudo ocorrer corretamente, a aba abaixo do editor exibirá mensagens confirmando que os comandos foram inseridos corretamente.

#	Time	Action	Message
1	08:29:54	CREATE DATABASE db_fsense	1 row(s) affected
2	08:29:54	use db_fsense	0 row(s) affected
3	08:29:54	CREATE TABLE tb_team ( id BIGINT PRIMARY KEY, name VARCHAR(255) NOT NULL, team_parent_id BIGINT, ...	0 row(s) affected
4	08:29:54	CREATE INDEX idx_team_parent_id ON tb_team(team_parent_id)	0 row(s) affected R
5	08:29:54	CREATE TABLE tb_person ( id BIGINT PRIMARY KEY, name VARCHAR(255), email VARCHAR(255), device_id...	0 row(s) affected
6	08:29:54	CREATE TABLE tb_category ( id BIGINT PRIMARY KEY auto_increment, application_name VARCHAR(255), ca...	0 row(s) affected
7	08:29:54	ALTER table tb_category ADD constraint unique_category_constraint UNIQUE (application_name, category, category_...	0 row(s) affected R
8	08:29:54	CREATE TABLE tb_general_category ( id BIGINT PRIMARY KEY AUTO_INCREMENT, application_name VARC...	0 row(s) affected
9	08:29:54	ALTER TABLE tb_general_category ADD constraint unique_general_category_constraint UNIQUE (application_name, ...	0 row(s) affected R
10	08:29:54	CREATE TABLE tb_categories_by_team ( id BIGINT PRIMARY KEY AUTO_INCREMENT, application_name VARC...	0 row(s) affected
11	08:29:54	ALTER table tb_categories_by_team ADD constraint unique_categories_by_team_constraint UNIQUE (application_na...	0 row(s) affected R
12	08:29:54	CREATE TABLE tb_activity ( id BIGINT PRIMARY KEY auto_increment, date DATE, application VARCHAR(255), ...	0 row(s) affected
13	08:29:55	CREATE TABLE tb_event ( id BIGINT PRIMARY KEY auto_increment, application VARCHAR(255), login VARCHA...	0 row(s) affected
14	08:29:55	CREATE TABLE tb_workday ( id INT PRIMARY KEY, name VARCHAR(255), work_national_holidays BOOLEAN, ...	0 row(s) affected
15	08:29:55	CREATE table tb_workday_schedule ( weekday ENUM('MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRID...	0 row(s) affected
16	08:29:55	ALTER TABLE tb_workday_schedule ADD CONSTRAINT uq_weekday_start_end UNIQUE(weekday, start_time, end_...	0 row(s) affected R
17	08:29:55	ALTER TABLE tb_workday_schedule ADD CONSTRAINT uq_weekday_max_flex_duration UNIQUE(weekday, max_fl...	0 row(s) affected R
18	08:29:55	CREATE TABLE tb_workday_override ( id bigint PRIMARY KEY, person_id BIGINT, date Date, start_time TIME, ...	0 row(s) affected
19	08:29:55	CREATE TABLE tb_workday_prorogation ( id BIGINT PRIMARY KEY, source VARCHAR(255), person_id BIGINT, ...	0 row(s) affected
20	08:29:55	CREATE TABLE tb_login_history( id BIGINT primary key AUTO_INCREMENT, name VARCHAR(255), email VARC...	0 row(s) affected
21	08:29:55	create table tb_holiday ( id BIGINT PRIMARY KEY, name VARCHAR(255), date TIMESTAMP, type ENUM('NATI...	0 row(s) affected

- Atualização do Schema Na aba **Schemas**, clique no botão **Refresh**. Isso mostrará que o esquema foi criado corretamente, como demonstrado na imagem abaixo.



- Conclusão com Êxito Parabéns! Seu banco de dados foi criado com sucesso. Na próxima seção, vamos configurar a aplicação com as credenciais do seu usuário.



## Credenciais do Usuário

- Pré-Requisitos

Antes de configurar as credenciais, é crucial garantir que o banco de dados desejado esteja configurado e o esquema de dados tenha sido criado. Se houver dúvidas sobre a criação do esquema de dados, consulte o tópico [Criação das Tabelas](#).

- Localizando e Editando o Arquivo de Configuração

No arquivo `fsense.properties` da aplicação, há uma configuração predefinida para MySQL, PostgreSQL e SQL Server. Encontre a seção de configuração correspondente ao banco que você pretende usar.

```
#MySQL
#string.connection=jdbc:mysql://localhost:3306/<your-database>?useSSL=true
#database.user=<your-username>
#database.password=<your-password>

#PostgreSQL
#string.connection=jdbc:postgresql://localhost:5432/<your-database>
```

```
#database.user=<your-username>
#database.password=<your-password>
```

```
#SQLServer
```

```
#string.connection=jdbc:sqlserver://localhost:1433;databaseName=<your-
database>;encrypt=false;trustServerCertificate=false;
#database.user=<your-username>
#database.password=<your-password>
```

- Descomentar as Configurações

Remova o caractere `"#"` de cada linha do banco que deseja integrar para que a aplicação consiga ler a configuração. Como no exemplo abaixo

```
#MySQL
```

```
string.connection=jdbc:mysql://localhost:3306/<your-database>?useSSL=true
database.user=<your-username>
database.password=<your-password>
```

```
#PostgreSQL
```

```
#string.connection=jdbc:postgresql://localhost:5432/<your-database>
#database.user=<your-username>
#database.password=<your-password>
```

```
#SQLServer
```

```
#string.connection=jdbc:sqlserver://localhost:1433;databaseName=<your-
database>;encrypt=false;trustServerCertificate=false;
#database.user=<your-username>
#database.password=<your-password>
```

- Inserindo Credenciais Reais

Substitua os parâmetros `<your-username>` e `<your-password>` pelas informações reais do seu banco de dados.

-Exemplo

```
database.user=db_admin
database.password=12345678
```

## IMPORTANTE

Certifique-se de seguir estas etapas com atenção para garantir uma configuração adequada das credenciais do usuário.

## String de Conexão

Agora, vamos configurar a string de conexão com o banco de dados.

- Ativando a Configuração

Remova o caractere `"#"` da linha `string.connection` no arquivo de configuração.

Substitua o parâmetro `<your-database>` pelo nome do esquema que foi criado usando o script nos tópicos anteriores, por exemplo, `db_fsense`.

```
#MySQL
#string.connection=jdbc:mysql://localhost:3306/<your-database>?useSSL=true
database.user=db_admin
database.password=12345678
```

```
#MySQL
string.connection=jdbc:mysql://localhost:3306/db_fsense?useSSL=true
database.user=db_admin
database.password=12345678
```

- Verificação da URL e Porta

## IMPORTANTE

Certifique-se de que a `URL de conexão` e a `porta` estejam corretas. Se necessário, substitua pela porta que o seu banco esteja utilizando.

- Conclusão da Configuração

Pronto! A configuração está concluída. No próximo tópico, vamos testar a conexão com o banco de dados.

## ! DICA

Ao remover o "#" da linha `string.connection`, você está ativando essa configuração, permitindo que o programa utilize a string de conexão específica para o banco de dados configurado. Certifique-se de inserir corretamente o nome do esquema e verificar a URL e a porta para garantir uma conexão bem-sucedida.

## Teste de Conexão

Agora, vamos testar a conexão do aplicativo com o banco de dados.

- Identificação dos Serviços Disponíveis

Se o aplicativo for executado sem nenhum serviço informado como parâmetro, ele exibirá no console a lista de serviços disponíveis. Há um serviço que foi implementado para fazer o teste da aplicação Java com o banco de dados chamado `DATABASE_TEST_SERVICE`.

```
SELECTED SERVICE []  
No service found for informed name:  
Available services:  
[LOGIN_HISTORY_SERVICE]  
[WORKDAY_OVERRIDE_SERVICE]  
[PERSON_SERVICE]  
[ACTIVITY_SERVICE]  
[WORKDAY_SERVICE]  
[CATEGORY_SERVICE]  
[EVENT_SERVICE]  
[DATABASE_TEST_SERVICE]  
[TEAM_SERVICE]  
[WORKDAY_PROROGATION_SERVICE]  
[HOLIDAY_SERVICE]  
[AUTH_SERVICE]
```

- Escolha do Serviço de Teste de Conexão com o Banco

Selecione o Serviço `DATABASE_TEST_SERVICE` no campo `service.name`.

- Escolha do Método de Teste de Conexão

Em seguida, selecione o método `DATABASE_TEST_CONNECTION` no campo `service.method` e execute a aplicação.

```
#SERVICO A SER EXECUTADO  
service.name=DATABASE_TEST_SERVICE  
#METODO DO SERVIÇO A SER EXECUTADO  
service.method=DATABASE_TEST_CONNECTION
```

- Verificação do Sucesso da Conexão

Caso tudo esteja configurado corretamente, o console exibirá uma mensagem informando que a conexão foi realizada com sucesso.

```
SELECTED SERVICE [DATABASE_TEST_SERVICE]  
SELECTED METHOD [DATABASE_TEST_CONNECTION]  
Connection successfully  
End of execution.
```

- Lidando com Erros

Se algum parâmetro estiver incorreto, o console exibirá uma mensagem de erro do banco de dados. Tente revisar as configurações e execute novamente.

```
SELECTED SERVICE [DATABASE_TEST_SERVICE]  
SELECTED METHOD [DATABASE_TEST_CONNECTION]  
Access denied for user 'admin'@'localhost' (using password: YES)  
End of execution.
```

- Conclusão Bem-Sucedida

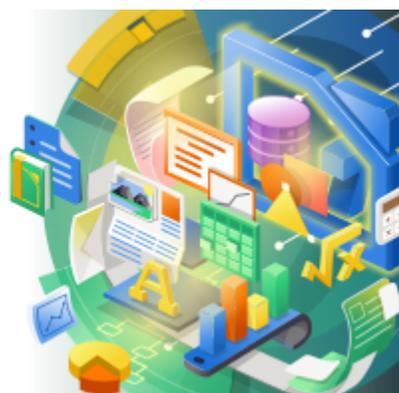
Parabéns! A configuração da aplicação com o banco de dados foi realizada. Agora, a aplicação Java está pronta para interagir com o seu banco de dados. Certifique-se de que todos os passos foram seguidos corretamente para garantir uma operação eficaz.

# Importação de Arquivos CSV

Neste tópico, fornecemos diretrizes para abrir e manipular arquivos CSV, considerando detalhes como separadores e o uso de aspas. É crucial garantir que, após salvar os dados, sua estrutura seja preservada. Como exemplo, usaremos a suíte LibreOffice Calc, mas você pode optar por um editor de texto ou uma suíte de escritório de sua preferência, fazendo as adaptações necessárias.

- Abertura do LibreOffice Calc

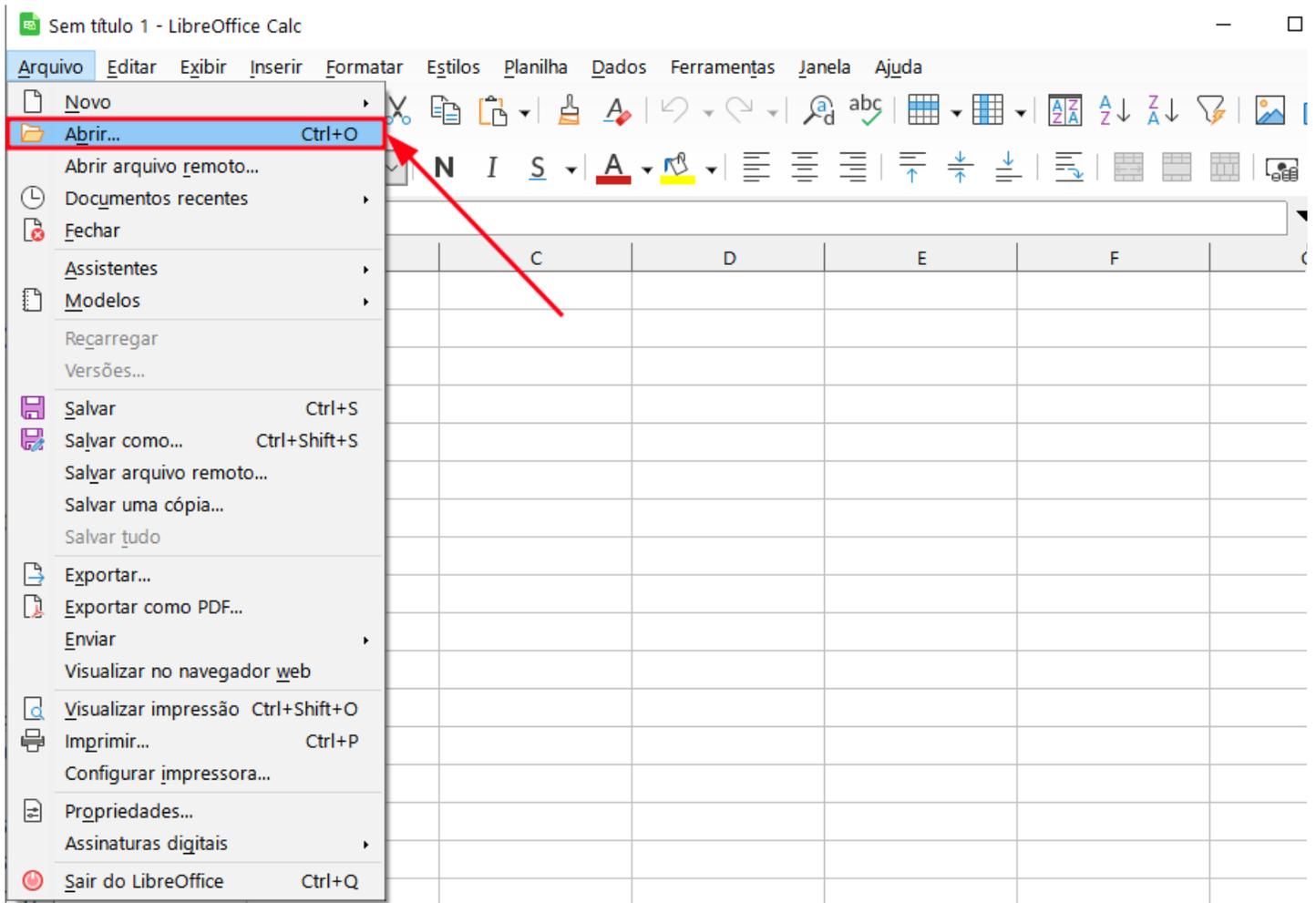
Abra o LibreOffice Calc.



- Abertura do Arquivo CSV

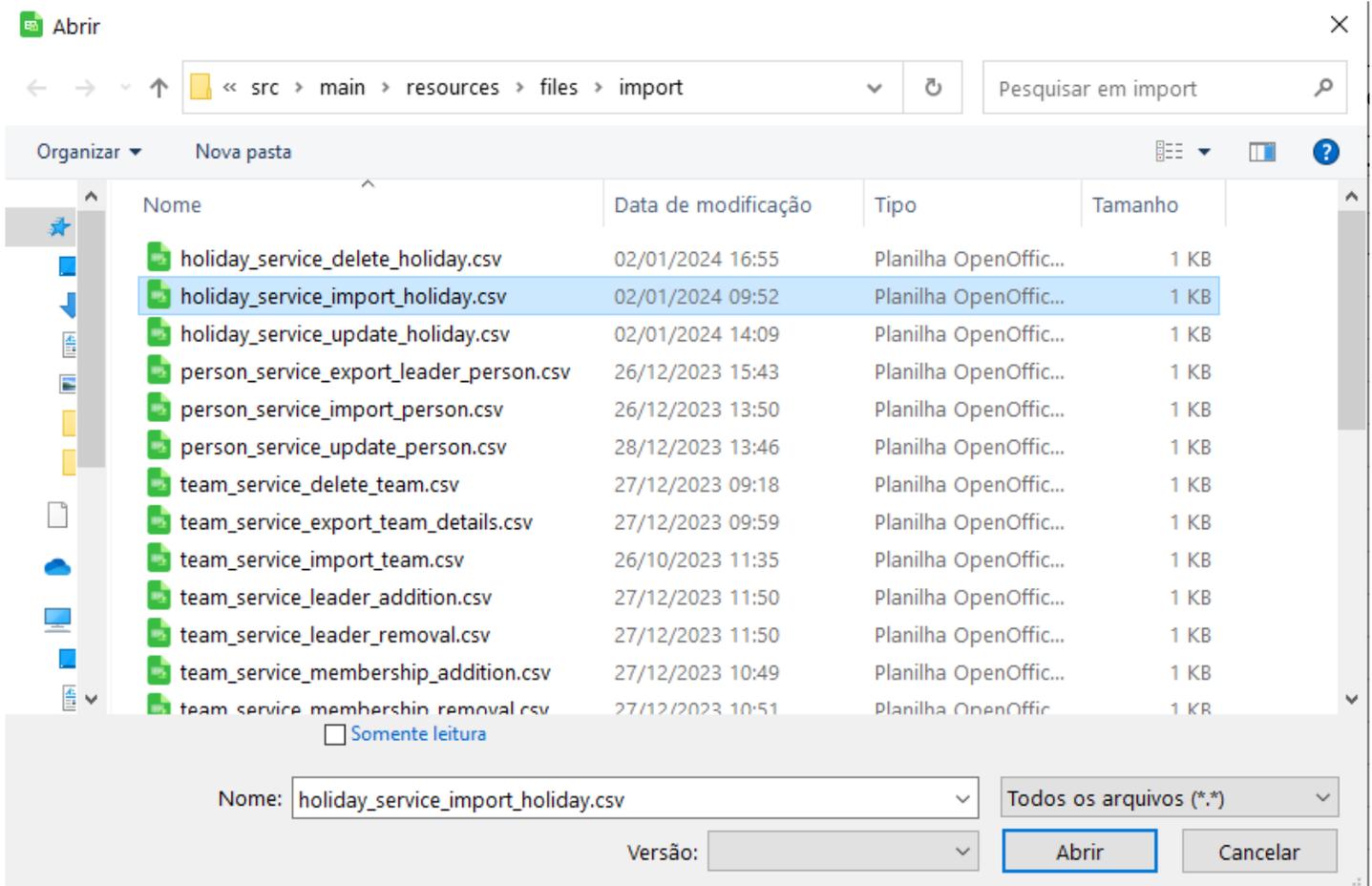
Clique em **Arquivo** no menu superior.

Selecione **Abrir** ou pressione "**Ctrl + O**" no teclado para abrir o diálogo de abertura de arquivo.



Navegue até o local onde o arquivo CSV está armazenado.

Selecione o arquivo CSV desejado e clique em **Abrir**.



- Configuração da Importação

Após abrir o arquivo, a janela do **Assistente de Importação** será exibida, permitindo a configuração da importação dos dados.

**Importar**

Conjunto de caracteres:

Localidade:

Da linha:

**Opções de separadores**

Largura fixa                       Separado por  
 Tabulação    Vírgula    Ponto-e-vírgula    Espaço    Outro     
 Mesclar delimitadores    Aparar os espaços   Delimitador de texto:

**Outras opções**

Formatar campos entre aspas como texto    Detectar números especiais  
 Avaliar fórmulas

**Campos**

Tipo de coluna:

	Padrão	Padrão	Padrão	Padrão
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Natal	NATIONAL	25/12/2024	true

Na seção **Separador**, certifique-se de que o separador selecionado seja o mesmo configurado na propriedade **csv.delimiter** do arquivo **fsense.properties**.

Marque a opção **Tabulação** caso esteja desmarcada.

Importação de texto - [holiday\_service\_import\_holiday.csv] X

**Importar**

Conjunto de caracteres: Unicode (UTF-8) v

Localidade: Padrão - Português (Brasil) v

Da linha: 1 v

**Opções de separadores**

Largura fixa                       Separado por  
 **Tabulação**    Vírgula    Ponto-e-vírgula    Espaço    Outro     
 Mesclar delimitadores    Aparar os espaços   Delimitador de texto: ' ' v

**Outras opções**

Formatar campos entre aspas como texto    Detectar números especiais  
 Avaliar fórmulas

**Campos**

Tipo de coluna:

	Padrão	Padrão	Padrão	Padrão
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Natal	NATIONAL	25/12/2024	true

<  >

Marque a opção **Mesclar delimitadores**.

Importação de texto - [holiday\_service\_import\_holiday.csv] X

**Importar**

Conjunto de caracteres: Unicode (UTF-8) v

Localidade: Padrão - Português (Brasil) v

Da linha: 1 v

**Opções de separadores**

Largura fixa  Separado por:

Tabulação  Vírgula  Ponto-e-vírgula  Espaço  Outro

Mesclar delimitadores  Aparar os espaços Delimitador de texto: ' v

**Outras opções**

Formatar campos entre aspas como texto  Detectar números especiais

Avaliar fórmulas

**Campos**

Tipo de coluna: v

	Padrão	Padrão	Padrão	Padrão
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Natal	NATIONAL	25/12/2024	true

Ajuda OK Cancelar

Na seção **Delimitador de texto**, certifique-se de que o delimitador selecionado seja o mesmo configurado na propriedade **csv.text.qualifier** do arquivo **fsense.properties**.

Importação de texto - [holiday\_service\_import\_holiday.csv] X

**Importar**

Conjunto de caracteres: Unicode (UTF-8) v

Localidade: Padrão - Português (Brasil) v

Da linha: 1

**Opções de separadores**

Largura fixa  Separado por

Tabulação  Vírgula  Ponto-e-vírgula  Espaço  Outro

Mesclar delimitadores  Aparar os espaços **Delimitador de texto: ' v**

**Outras opções**

Formatar campos entre aspas como texto  Detectar números especiais

Avaliar fórmulas

**Campos**

Tipo de coluna: v

	Padrão	Padrão	Padrão	Padrão
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Natal	NATIONAL	25/12/2024	true

Ajuda OK Cancelar

Marque a opção **Formatar campos entre aspas como texto** para garantir que todos os dados sejam tratados como texto.

**Importar**

Conjunto de caracteres: Unicode (UTF-8)

Localidade: Padrão - Português (Brasil)

Da linha: 1

**Opções de separadores**

Largura fixa                       Separado por  
 Tabulação    Vírgula    Ponto-e-vírgula    Espaço    Outro  
 Mesclar delimitadores    Aparar os espaços   Delimitador de texto: '

**Outras opções**

Formatar campos entre aspas como texto    Detectar números especiais  
 Avaliar fórmulas

**Campos**

Tipo de coluna:

	Padrão	Padrão	Padrão	Padrão
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Natal	NATIONAL	25/12/2024	true

Clique em **OK**.

Após configurar a importação de acordo com suas preferências, o arquivo será aberto no LibreOffice com os dados formatados conforme as opções escolhidas.

- Remoção da Formatação

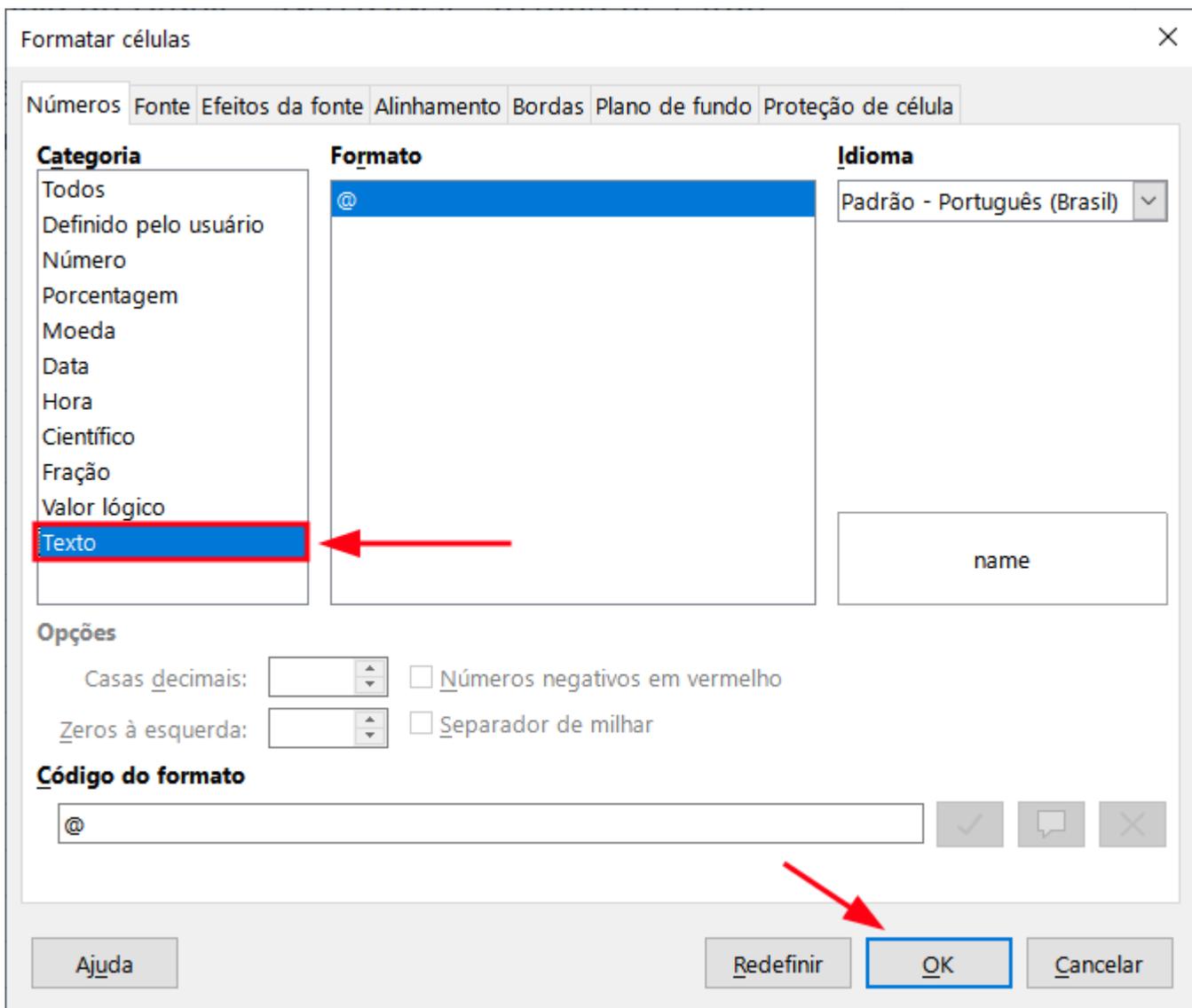
Selecione todas as células da tabela e clique com o botão direito, escolhendo a opção **Formatar células**.

	A	B	C	D
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Na	NATIONAL	25/12/2024	true
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				

-  Cortar Ctrl+X
-  Copiar Ctrl+C
-  Colar Ctrl+V
-  Colar especial ▾
- Lista de seleção Alt+Down
-  Inserir... Ctrl++
-  Excluir... Ctrl+-
-  Limpar conteúdo... Backspace
- Mesclar células
-  Clonar formatação
-  Limpar formatação direta Ctrl+M
- Estilos ▾
-  Inserir anotação Ctrl+Alt+C
- Minigráficos ▾
-  **Formatar células...** Ctrl+1



Na aba **Números**, selecione a opção **Como texto**.



Insira um novo registro e salve o arquivo.

	A	B	C	D
1	name	type	date	canRepeat
2	Independencia do Brasil	NATIONAL	07/09/2024	true
3	Finados	NATIONAL	02/11/2024	false
4	Proclamação da República	NATIONAL	15/11/2024	true
5	NSa Aparecida	NATIONAL	12/10/2024	true
6	Natal	NATIONAL	25/12/2024	true
7	<b>Novo Registro</b>	NATIONAL	01/01/24	true
8				

Ao abrir o arquivo em outro editor de texto, como o Visual Studio Code por exemplo, verifique que os dados foram salvos mantendo a mesma estrutura de separadores e

delimitadores, essencial para que a aplicação Java possa realizar a leitura do arquivo CSV corretamente e evitar possíveis erros de leitura devido delimitadores diferentes.

```
1 |'name';'type';'date';'canRepeat'  
2 |'Independencia do Brasil';'NATIONAL';'07/09/2024';'true'  
3 |'Finados';'NATIONAL';'02/11/2024';'false'  
4 |'Proclamação da República';'NATIONAL';'15/11/2024';'true'  
5 |'NSa Aparecida';'NATIONAL';'12/10/2024';'true'  
6 |'Natal';'NATIONAL';'25/12/2024';'true'  
7 |'Novo Registro';'NATIONAL';01/01/24;'true'  
8
```