

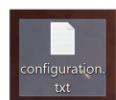


Externalizar configurações da aplicação em arquivos

O **application properties** do **Spring** contém diversas informações relevantes para a aplicação, como *dados para se conectar em um database, segredos de autenticação e zas e zas....*

Existem diversas formas de externalizar esses dados sensíveis, eu gosto de usar **variáveis de ambiente**, mas também é possível simplesmente externalizar essas configurações em **outro arquivo** e pedir para o **Spring** ler desse **arquivo externalizado**.

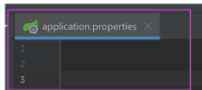
Para fazer isso basta criar um arquivo no diretório que você quiser e usar as informações desejadas dentro desse arquivo, igualzinho ao que você faria no **application properties**:



```
OrOddStepConfig.java  application.properties
spring.datasource.url=${DB_URL}
spring.datasource.username=${DB_USERNAME}
spring.datasource.password=${DB_PASSWORD}
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true
spring.batch.jdbc.initialize-schema=always
```

```
configuration.txt - Bloco de Notas
Arquivo Editar Formatar Exibir Ajuda
spring.datasource.url=jdbc:mysql://localhost:3306/batch_process
spring.datasource.username=root
spring.datasource.password=root
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true
spring.batch.jdbc.initialize-schema=always
```

Com o **arquivo criado** basta dizer ao Spring o que ele deve usar como fonte de configuração, basta fazer isso injetando um **Bean específico**, esse Bean é o **PropertySourcesPlaceholderConfigurer**. Dizemos nesse Bean aonde está o arquivo de configuração e deixamos limpo o **application.properties**, o Spring então vai usar esse **Bean** para procurar a fonte da configuração:



```
@Configuration
public class PropertiesConfig {

    @Bean
    public PropertySourcesPlaceholderConfigurer config() {
        PropertySourcesPlaceholderConfigurer propertyConfig = new PropertySourcesPlaceholderConfigurer();
        propertyConfig.setLocation(new FileSystemResource("path: /c:/users/lucas/desktop/configuration.txt"));
        return propertyConfig;
    }
}
```

