



Variáveis de ambiente

Sumário

[Configurando variáveis de ambiente](#)

[Introdução](#)

[Spring Tool Suite \(STS\)](#)

[Visual Studio Code](#)

[Intellij](#)

Configurando variáveis de ambiente

Introdução

A configuração das variáveis de ambiente impede que variáveis sensíveis como chaves ou senhas, por exemplo, sejam expostas a qualquer pessoa com acesso ao projeto, além de torná-las personalizáveis para cada colaborador, de acordo com suas necessidades.

Exemplo de sintaxe:

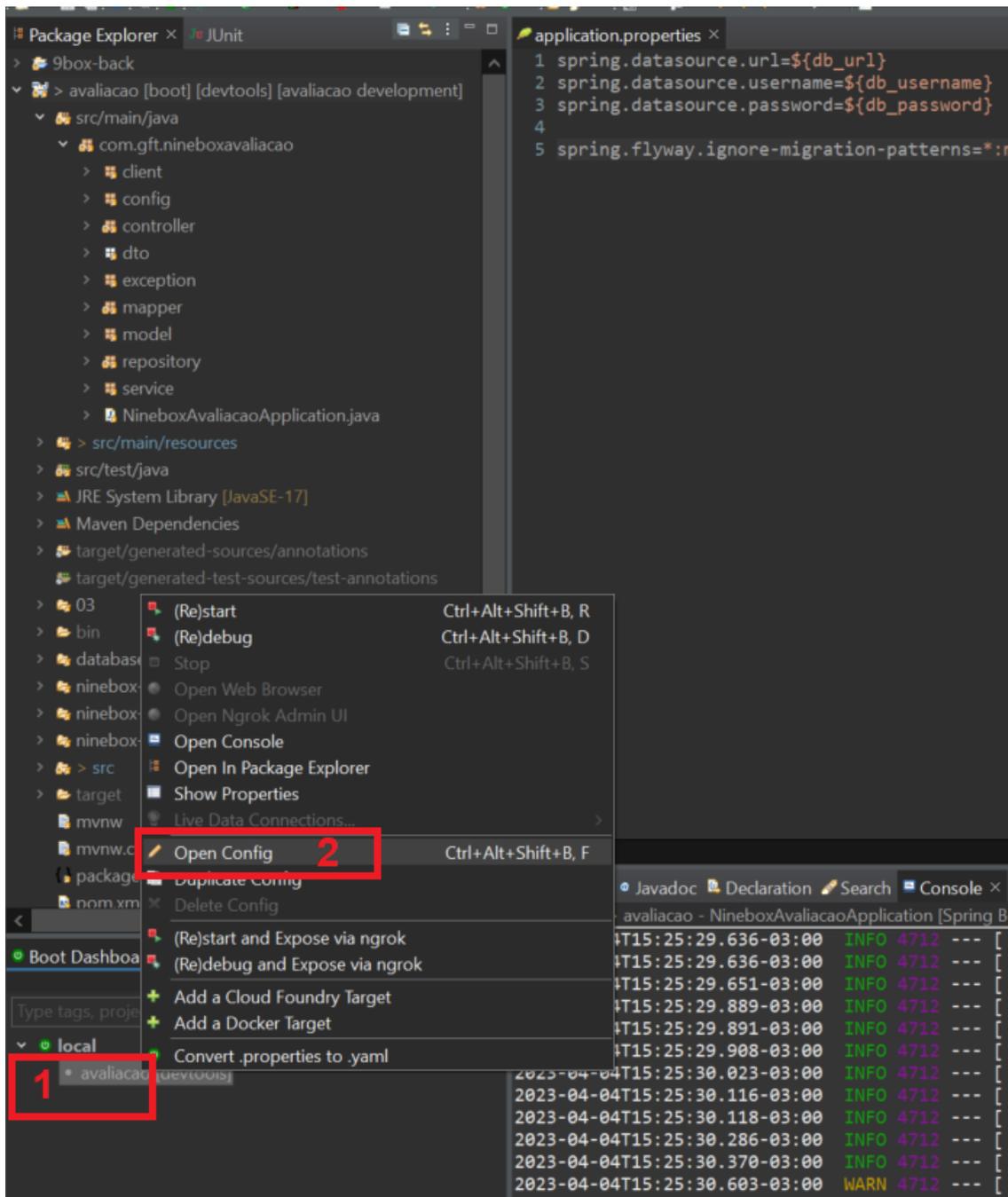
```
spring.datasource.url=${db_url}
spring.datasource.username=${db_username}
spring.datasource.password=${db_password}
```

Nos próximos tópicos há instruções para esta configuração, de acordo com a IDE utilizada.

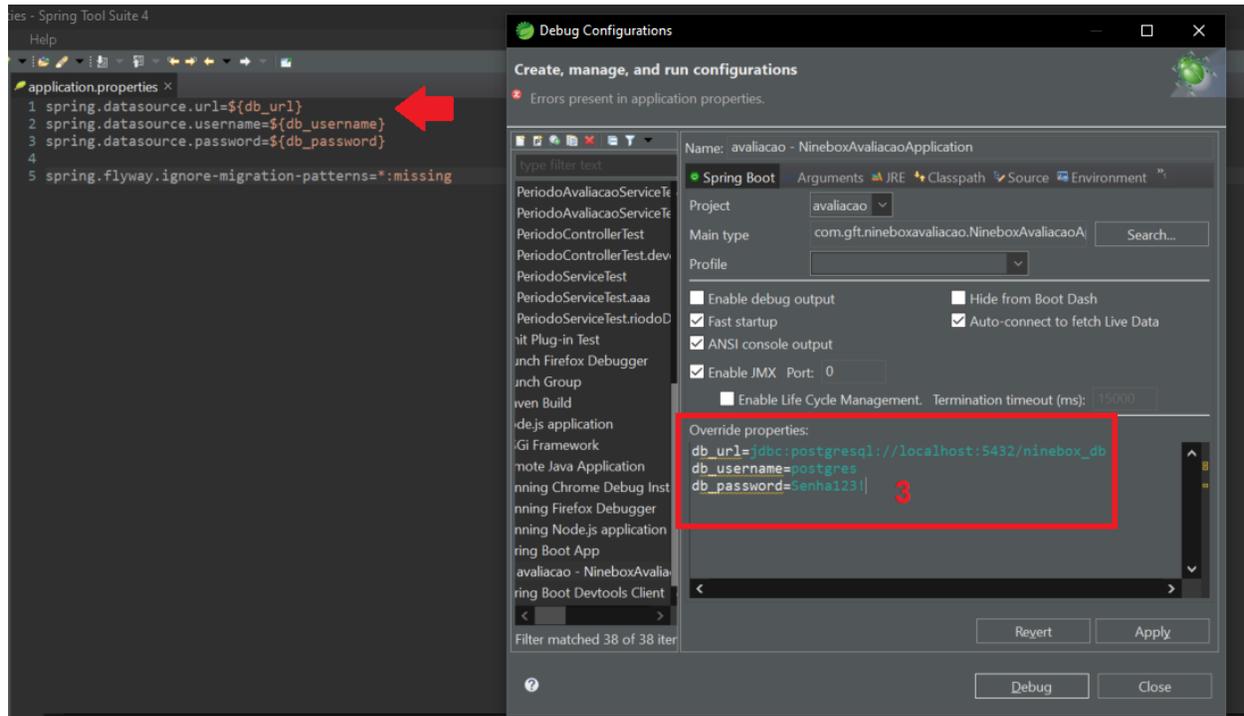
Spring Tool Suite (STS)

1 - No *Boot Dashboard*, clicar com botão direito no nome do projeto que se quer configurar variáveis;

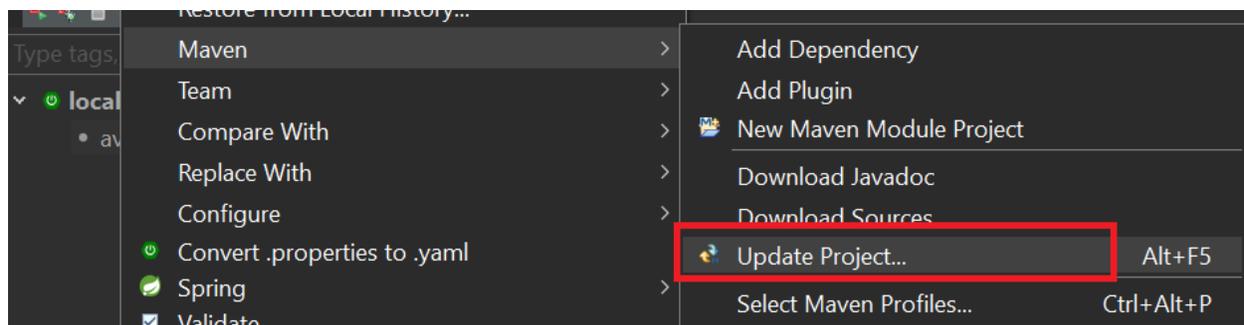
2 - Opção “*Open Config*”;



3 - Ao Abrir o modal *Debug Configurations*, inserir valores para as variáveis indicadas no arquivo *application.properties*. Os valores devem ser correspondentes ao que você utiliza e não os indicados abaixo (exemplo: se sua senha de conexão com o banco for *root*, alterar *Senha123!* para *root*);



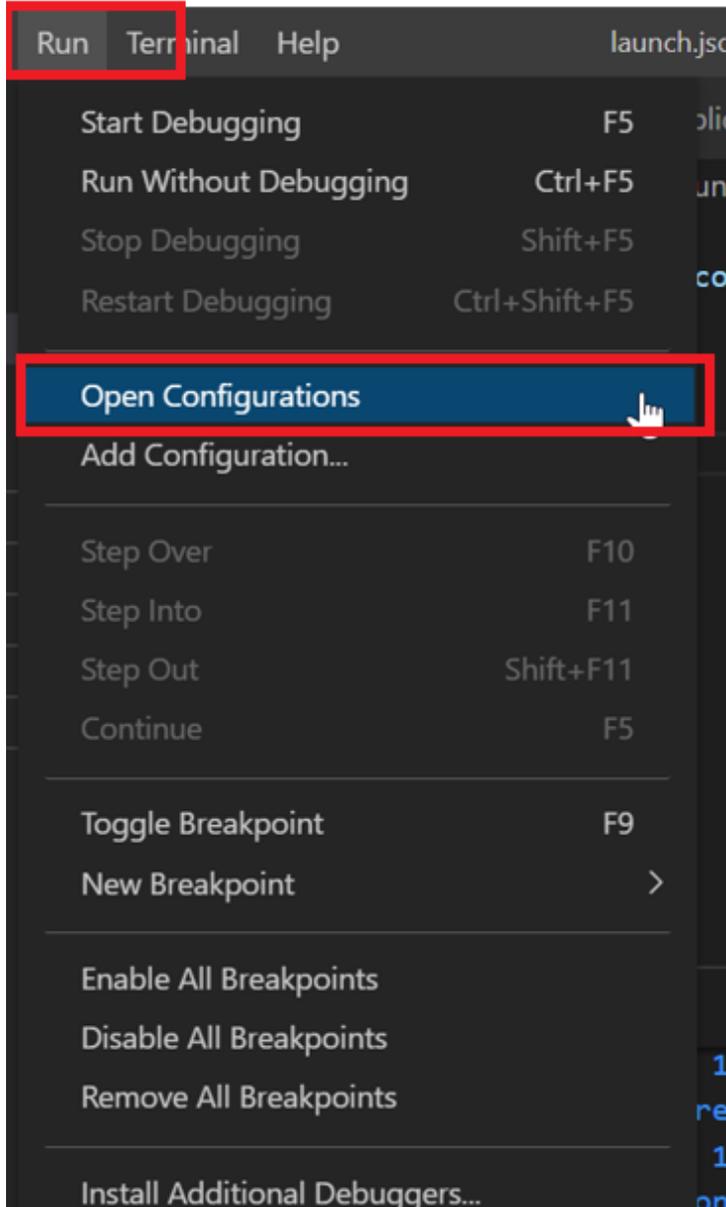
4 - Fazer update com o maven para novo build do projeto (*botão direito sobre o projeto > Maven > Update Project...*);



5 - Ao rodar o projeto, deve funcionar!

Visual Studio Code

1 - No menu superior da IDE, clicar em *Run* e, depois, na opção “*Open Configurations*”;



2 - Será aberto o arquivo *launch.json*, que deverá ser configurado conforme abaixo:

Obs.: para “*db_url*”, “*db_username*” e “*db_password*” inserir valores para as variáveis indicadas no arquivo *application.properties*. Os valores devem ser correspondentes ao

que você utiliza e não os indicados abaixo (exemplo: se sua senha de conexão com o banco for *root*, alterar *Senha123!* para *root*);

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "type": "java",
      "name": "NineboxAvaliacaoApplication",
      "request": "launch",
      "mainClass": "com.gft.nineboxavaliacao.NineboxAvaliacaoApplication",
      "projectName": "ninebox-avaliacao",
      "env": {
        "db_url": "jdbc:postgresql://localhost:5432/ninebox_db",
        "db_username": "postgres",
        "db_password": "Senha123!"
      }
    },
    {
      "type": "robotframework-lsp",
      "name": "Robot Framework: Launch .robot file",
      "request": "launch",
      "cwd": "^\\\\"${workspaceFolder}\\",
      "target": "^\\\\"${file}\\",
      "terminal": "integrated",
      "args": [],
      "env": {}
    }
  ],
}
```

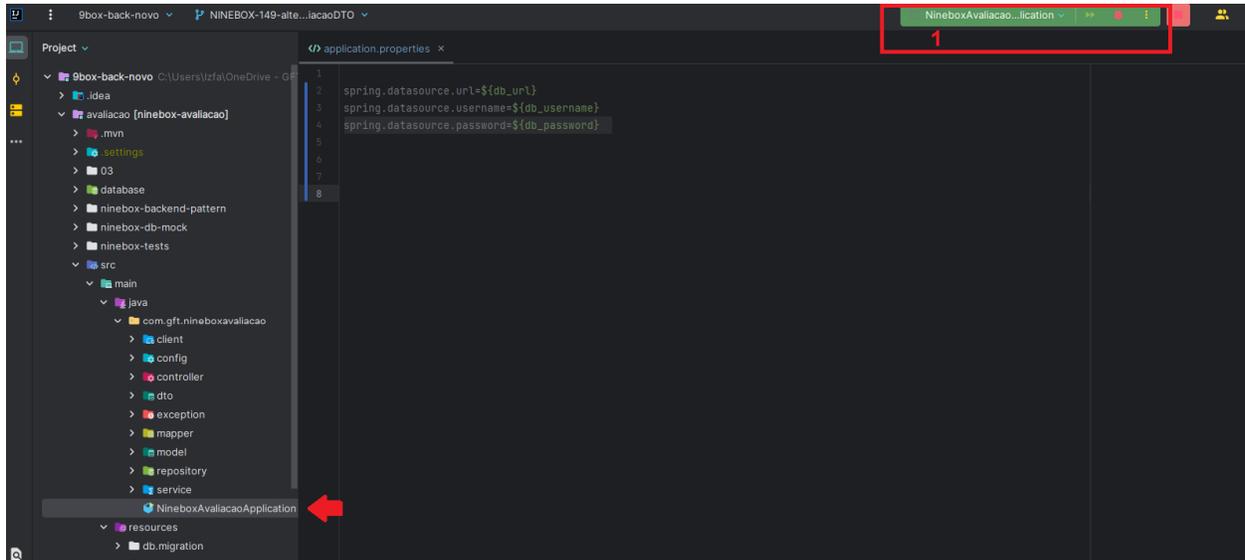
3 - É preciso fazer um update com o maven para gerar um novo build do projeto. Talvez esse link ajude, em caso de dúvidas:

<https://code.visualstudio.com/docs/java/java-build>

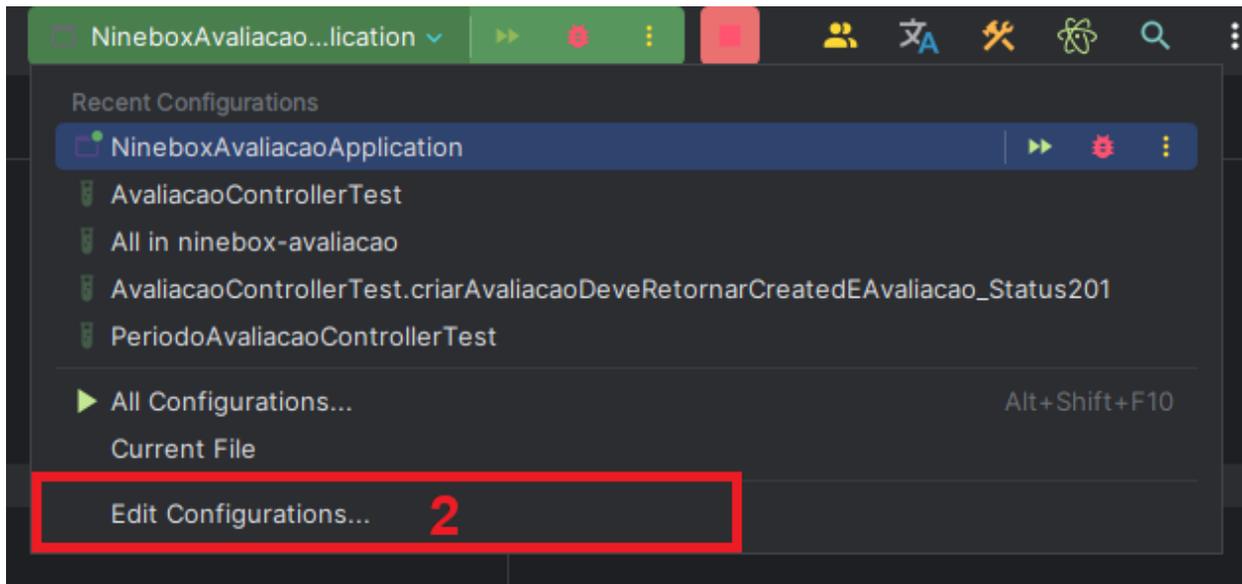
4 - Ao rodar o projeto, deve funcionar!

IntelliJ

1 - Na caixa de seleção indicada em **1**, selecione a opção correspondente ao arquivo principal do projeto (o que está apontado com uma seta);



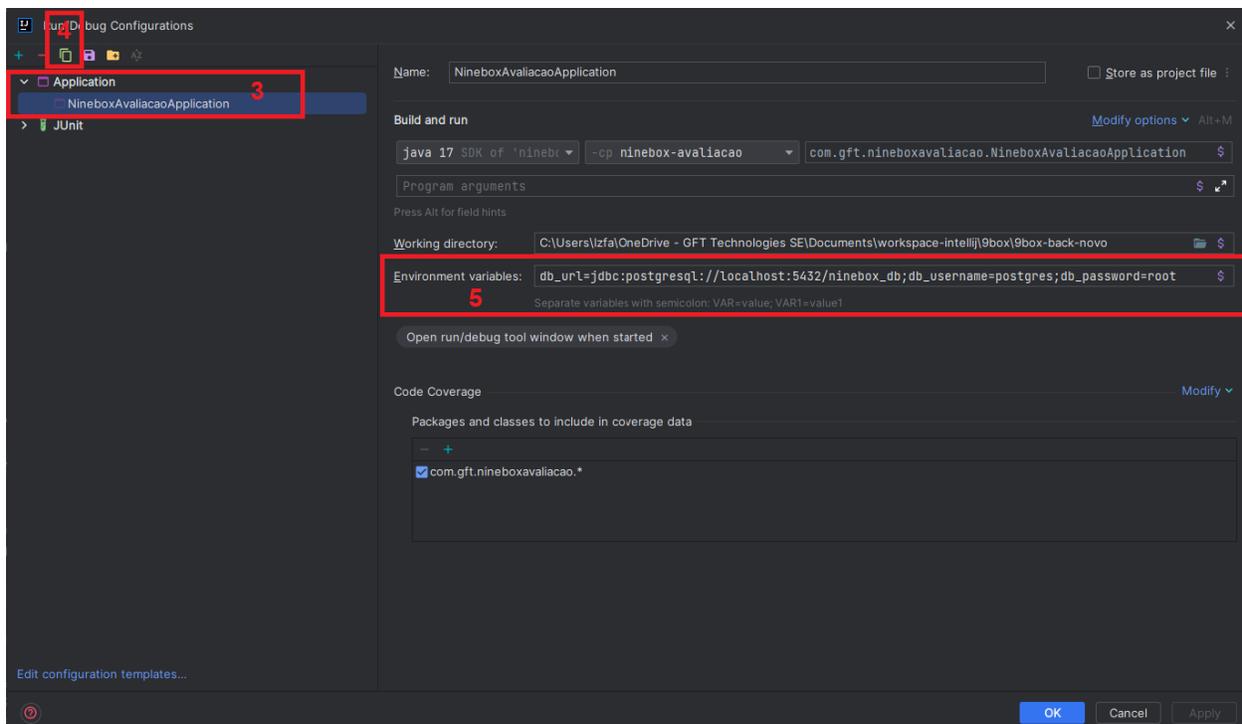
2 - Clique na opção “*Edit Configurations*”;



- 3 - Selecione novamente o arquivo principal da aplicação;
- 4 - Clique no botão verde indicado em 4 para habilitar configurações;
- 5 - Preencha o campo “*Environment variables*” conforme o exemplo abaixo;

Obs.: inserir valores para as variáveis indicadas no arquivo *application.properties*. Os valores devem ser correspondentes ao que você utiliza e não os indicados abaixo (exemplo: se sua senha de conexão com o banco for *root*, alterar *Senha123!* para *root*);

```
db_url=jdbc:postgresql://localhost:5432/ninebox_db;db_username=postgres;db_password=Senha123!
```



- 5 - Fazer um update com o maven para gerar um novo build do projeto;
- 6 - Ao rodar o projeto, deve funcionar!