

# Back-end Spring Boot (API REST)

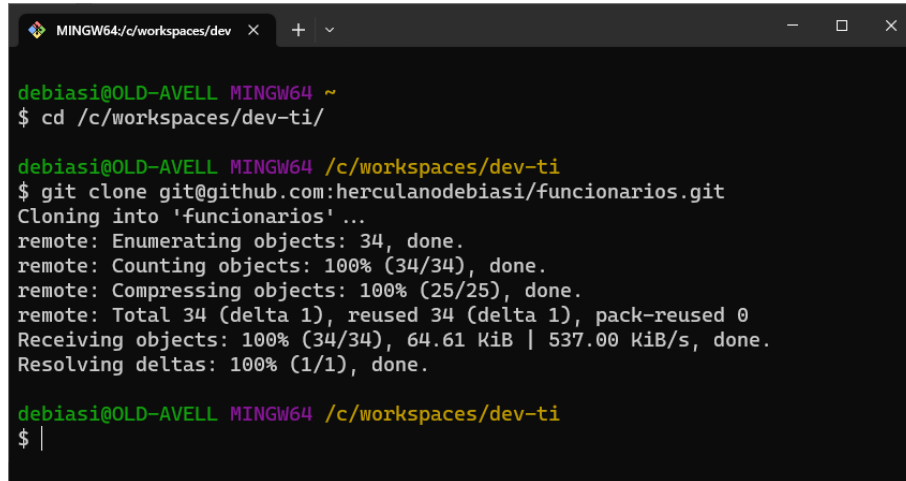
## 1 Importando o projeto

Duas formas, clonando o projeto do GitHub ou importando o .zip.

### 1.1 Clonando do GitHub

No *prompt* de comando, dentro da pasta do *workspace* execute:

➤ `git clone git@github.com:herculanodebiasi/funcionarios.git`

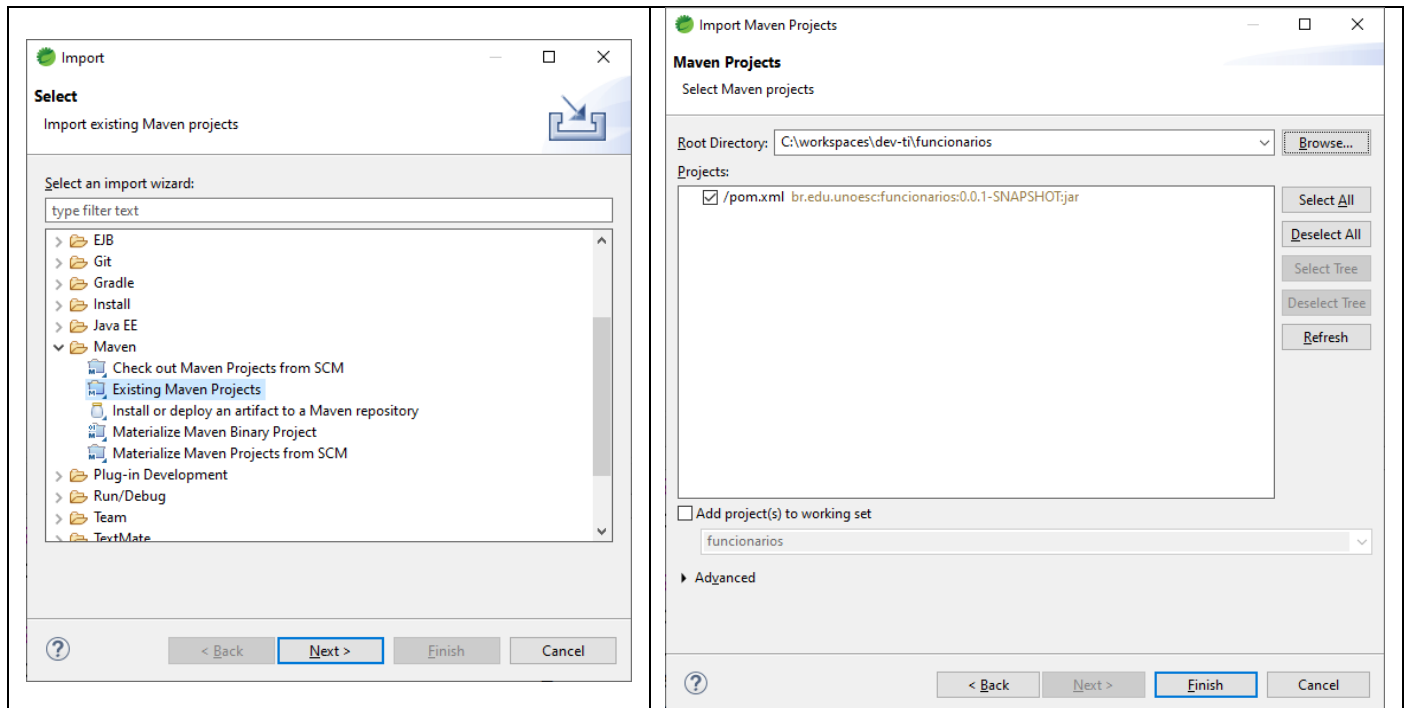


```
MINGW64/c:/workspaces/dev X + -
debiasi@OLD-AVELL MINGW64 ~
$ cd /c:/workspaces/dev-ti/

debiasi@OLD-AVELL MINGW64 /c:/workspaces/dev-ti
$ git clone git@github.com:herculanodebiasi/funcionarios.git
Cloning into 'funcionarios' ...
remote: Enumerating objects: 34, done.
remote: Counting objects: 100% (34/34), done.
remote: Compressing objects: 100% (25/25), done.
remote: Total 34 (delta 1), reused 34 (delta 1), pack-reused 0
Receiving objects: 100% (34/34), 64.61 KiB | 537.00 KiB/s, done.
Resolving deltas: 100% (1/1), done.

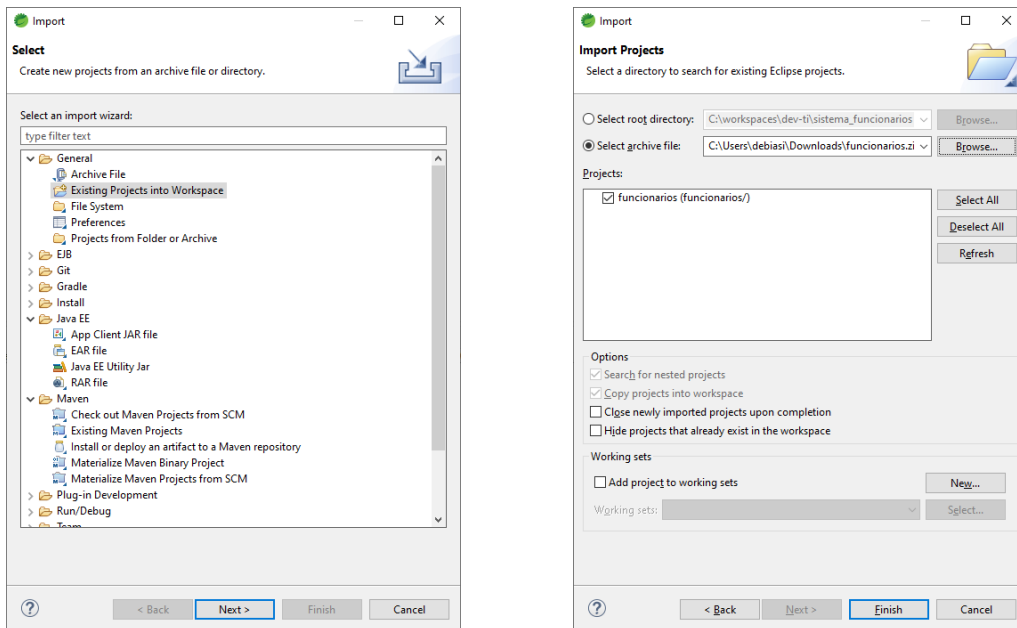
debiasi@OLD-AVELL MINGW64 /c:/workspaces/dev-ti
$ |
```

A importação do projeto para dentro do STS não é necessária caso se queira executar a aplicação (.jar) somente via de comando (seção 2.2). Para importar o projeto no STS vá no menu *File* → *Import* → *Existing Maven Projects*. Clique em *Browse* e escolha a pasta *funcionarios*. Clique em *Finish*.



## 1.2 Importando do Arquivo .zip

Menu *File* → *Import* → *General* → *Existing Projects into Workspace*.

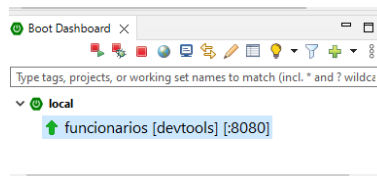


## 2 Executando o projeto

Duas formas, pela GUI ou pelo terminal. Por padrão o projeto sobe um servidor que fica ouvindo na porta 8080.

### 2.1 Pela GUI do STS

Após importar o projeto no STS basta abri-lo e então clicar no botão *Start* no *Boot Dashboard*.



### 2.2 Pelo terminal

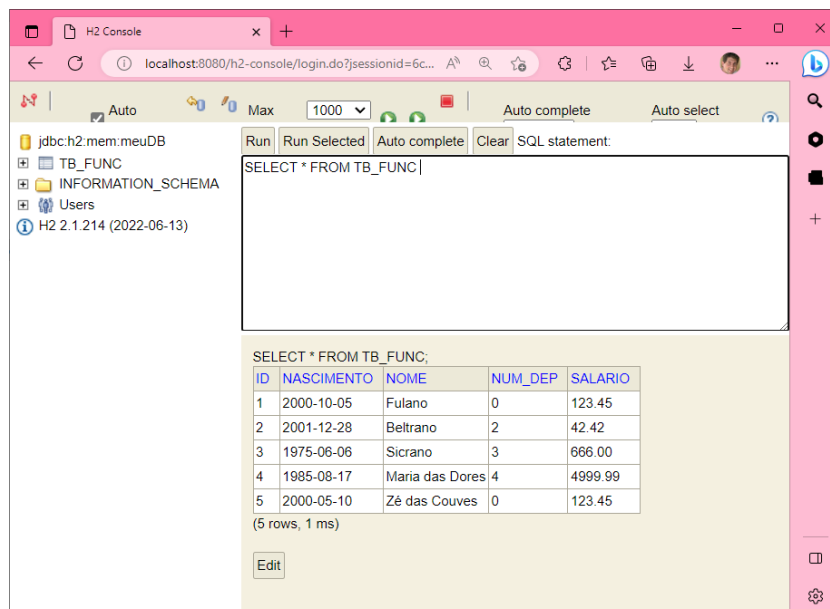
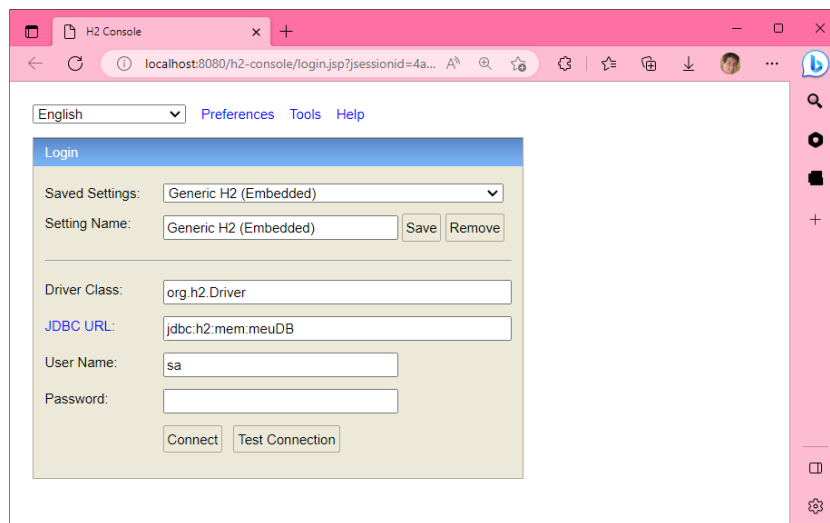
Abra um terminal e entre dentro da pasta *target* do projeto.

- `cd funcionarios/target`
- `java -jar funcionarios-0.0.1-SNAPSHOT.jar`

```
MINGW64/c:/workspaces/dev x + v
debiassi@OLD-AVELL MINGW64 /c:/workspaces/dev-ti/funcionarios/target (main)
$ java -jar funcionarios-0.0.1-SNAPSHOT.jar
2023-03-14 17:42:36.522 DEBUG 38340 --- [main] o.s.jdbc.datasource.DataSourceUtils : Fetching JDBC Connection from DataSource
2023-03-14 17:42:36.881 DEBUG 38340 --- [main] o.s.jdbc.datasource.DataSourceUtils : Fetching JDBC Connection from DataSource
2023-03-14 17:42:36.882 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : Executing SQL script from URL [jar:file:/C:/workspac
es/dev-ti/funcionarios/target/funcionarios-0.0.1-SNAPSHOT.jar!/BOOT-INF/classes!/schema.sql]
2023-03-14 17:42:36.901 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : 0 returned as update count for SQL: CREATE TABLE tb_
func ( id LONG AUTO_INCREMENT PRIMARY KEY, nome VARCHAR(255) NOT NULL, num_dep INT NOT NULL, salario NUMERIC(10, 2) NOT NULL, nascimento DATE NOT NULL )
2023-03-14 17:42:36.901 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : Executed SQL script from URL [jar:file:/C:/workspac
es/dev-ti/funcionarios/target/funcionarios-0.0.1-SNAPSHOT.jar!/BOOT-INF/classes!/schema.sql] in 19 ms.
2023-03-14 17:42:36.903 DEBUG 38340 --- [main] o.s.jdbc.datasource.DataSourceUtils : Fetching JDBC Connection from DataSource
2023-03-14 17:42:36.903 DEBUG 38340 --- [main] o.s.jdbc.datasource.DataSourceUtils : Fetching JDBC Connection from DataSource
2023-03-14 17:42:36.903 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : Executing SQL script from URL [jar:file:/C:/workspac
es/dev-ti/funcionarios/target/funcionarios-0.0.1-SNAPSHOT.jar!/BOOT-INF/classes!/data.sql]
2023-03-14 17:42:36.908 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : 1 returned as update count for SQL: INSERT INTO tb_f
unc (nome, num_dep, salario, nascimento) VALUES ('Fulano', 2, 1234.56, '1990-12-31')
2023-03-14 17:42:36.908 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : 1 returned as update count for SQL: INSERT INTO tb_f
unc (nome, num_dep, salario, nascimento) VALUES ('Beltrano', 0, 4999.99, '1975-06-06')
2023-03-14 17:42:36.908 DEBUG 38340 --- [main] o.s.jdbc.datasource.init.ScriptUtils : Executed SQL script from URL [jar:file:/C:/workspac
es/dev-ti/funcionarios/target/funcionarios-0.0.1-SNAPSHOT.jar!/BOOT-INF/classes!/data.sql] in 5 ms.
Hibernate:
    drop table if exists tb_func CASCADE
Hibernate:
    create table tb_func (
      id bigint generated by default as identity,
      nascimento date,
      nome varchar(255),
      num_dep integer,
      salario numeric(19,2),
      primary key (id)
    )
```

### 3 Console H2

Os dados são armazenados em um SGBD H2 somente em memória e a tabela é recriada e populada a cada vez que o sistema é reiniciado. O console do H2 está disponível no seguinte *link*: <http://localhost:8080/h2-console>



### 4 Documentação e Testes da API

Endereço: <http://localhost:8080/swagger-ui/index.html>

