

Introdução

Apache Maven, ou Maven, é uma ferramenta de automação de compilação utilizada primariamente em projetos Java.



Para quê serve ?

Mesmo tendo uma das principais funções como o gerenciamento de dependências, o maven também é capaz de auxiliar na compilação, build e deploy de nossos projetos desenvolvidos em Java.

Toda a configuração do projeto fica centralizada em arquivo demonidade de `pom.xml`

```
<!Estrutura simples do arquivo pom.xml>

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven
    <modelVersion>4.0.0</modelVersion>

    <groupId>com.suaempresa.ou.seuprojeto</groupId>
    <artifactId>projeto-maven</artifactId>
    <version>1.0</version>

    <name>projeto-maven</name>

    <properties>
      <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
      <maven.compiler.source>1.7</maven.compiler.source>
      <maven.compiler.target>1.7</maven.compiler.target>

      <sua.propriedade> ... </sua.propriedade>
    </properties>

    <dependencies>
      <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.11</version>
      </dependency>

      <dependency>
        ...
      </dependency>
    </dependencies>

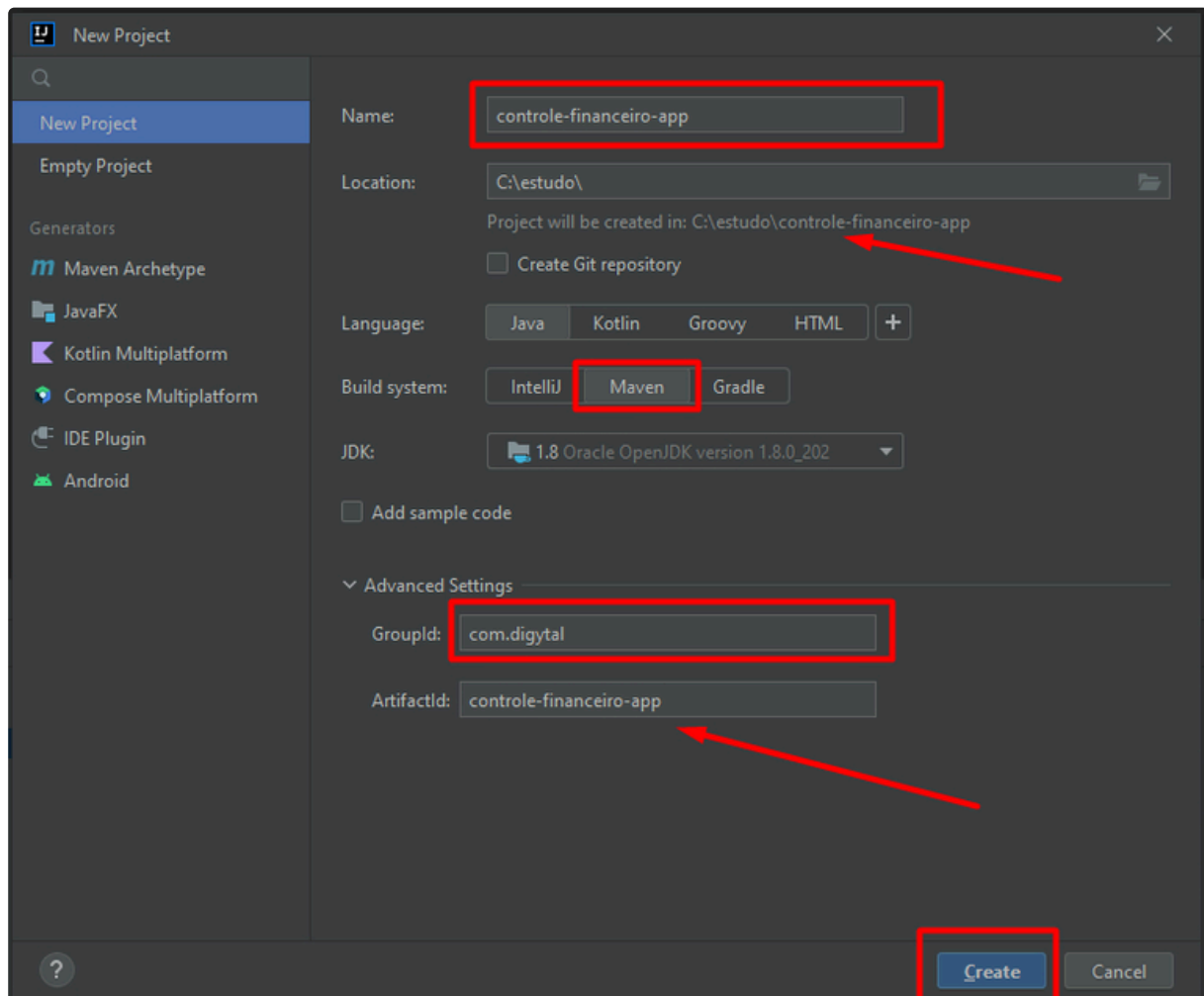
    <build>
      ...
    </build>
  </project>
```

Configuração

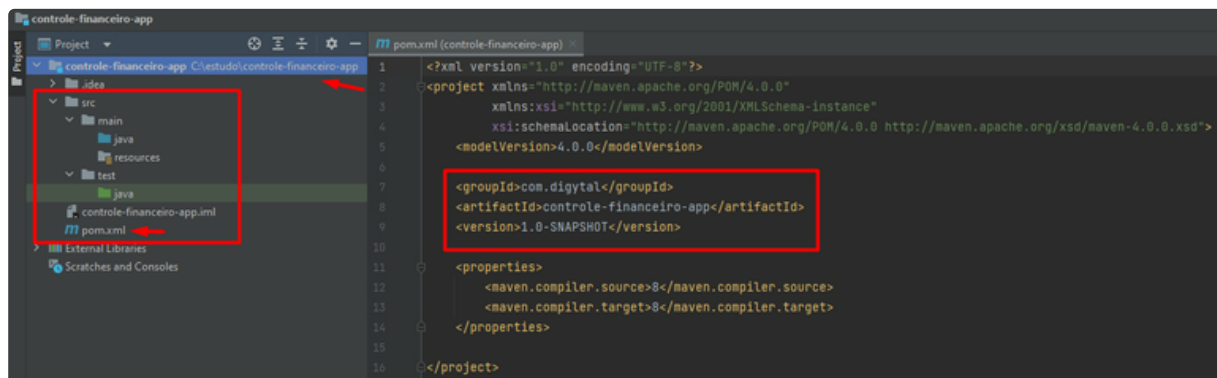
É mais conveniente em um ambiente de desenvolvimento utilizar o Maven através de uma IDE como Eclipse, IntelliJ e VSCode para criar o novo projeto, o que precisamos considerar inicialmente são as propriedades de identificação deste novo projeto:

- **groupId:** O nome da organização ao qual pertence esse projeto, possui o mesmo padrão de nomenclatura de pacotes, por exemplo `com.digytal`
- **artifactId:** O nome do projeto em questão, exemplo `controle-financeiro-app`
- **version:** A versão do nosso projeto, caso não seja colocado nenhum valor o Maven irá utilizar o valor padrão que é `1.0-SNAPSHOT`
- **package:** O nome do pacote principal do nosso projeto, é boa pratica colocar a junção do groupId com o artifactId, no nosso caso ficará `com.digytal.controlefinanceiro`, caso não seja passado nenhum valor, o Maven irá assumir o mesmo valor colocado no groupId

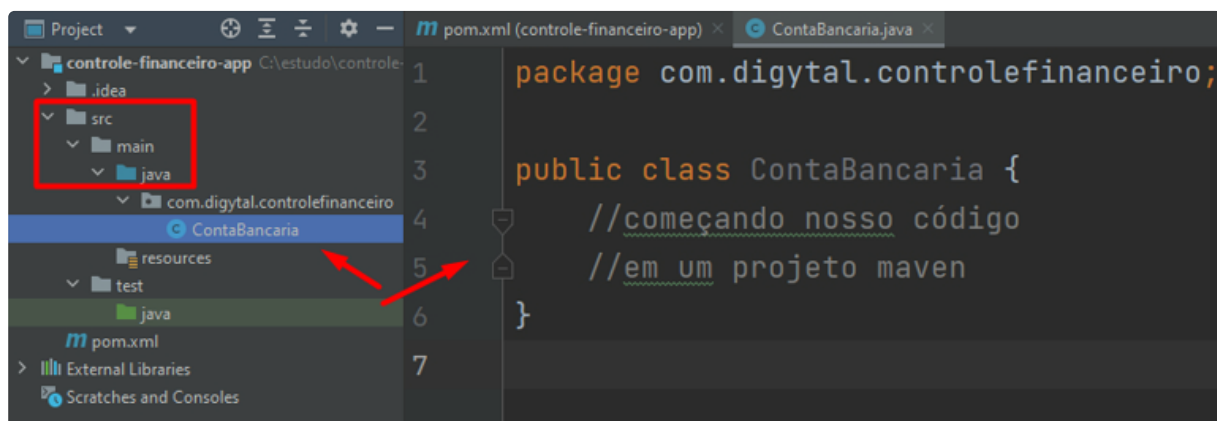
Abaixo temos uma ilustração de um projeto Maven sendo criado pelo assistente da IDE IntelliJ



Este é o resultado esperado



Agora todo o código fonte existente nos arquivos .java deverão existir no diretório `src/main/java`



Previous
Interface

Last updated 1 year ago