Roteiro de Atividade Prática

Nome: Turma: .

**Atividade: Configurando *builds* automatizados**

**Tempo estimado:** 30 minutos.

**Lista de materiais:**

* Computador com internet;
* Caderno ou folhas de sulfite;​
* Caneta para anotação.​

**Objetivo:**

Configurar um processo de *build* automatizado no Jenkins para gerar artefatos de *build* para uma aplicação Java.

**Enunciado:**

Seu próximo passo é configurar o Jenkins para compilar o código e gerar artefatos de *build* para uma aplicação Java. Para isso, segue o passo a passo usando um projeto Maven como exemplo.

**Passo a passo:**

**1. Adicione um projeto Maven no repositório:**

* Adicione um diretório chamado java-app no repositório.
* Crie um arquivo pom.xml com a configuração básica do Maven para um projeto Java. Um exemplo básico de pom.xml é:

|  |
| --- |
| <project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  <modelVersion>4.0.0</modelVersion>  <groupId>com.exemplo</groupId>  <artifactId>java-app</artifactId>  <version>1.0-SNAPSHOT</version>  <build>  <plugins>  <plugin>  <groupId>org.apache.maven.plugins</groupId>  <artifactId>maven-compiler-plugin</artifactId>  <version>3.8.1</version>  <configuration>  <source>1.8</source>  <target>1.8</target>  </configuration>  </plugin>  </plugins>  </build>  </project> |

**2. Configure o Jenkins para executar o *build* Maven:**

* No Jenkins, crie um novo *job* do tipo “Maven Project”.
* Dê um nome ao projeto, como “Build-Java-App”.
* Na seção “Source Code Management”, selecione “Git” e adicione a URL do repositório do GitHub.
* Na seção “Build”, adicione a tarefa ‘mvn clean install’.

**3. Configure o *webhook* no GitHub:**

* No repositório do GitHub, vá para “Settings” > “Webhooks” > “Add webhook”.
* Adicione a URL do Jenkins (por exemplo, http://seu-jenkins-url/github-webhook/) e selecione “Just the push event”.
* Salve o *webhook* para que o GitHub envie notificações ao Jenkins sempre que houver um *push* no repositório.

**Procedimento experimental:**

1. Após a configuração e *push* no repositório, o Jenkins executa o comando ‘mvc clean install’. Explique qual é o propósito deste comando. Sua resposta deve ser enviada no AVA em forma de explicação textual.

|  |
| --- |
|  |
|  |
|  |
|  |
|  |