



03

## Criando Ambiente

### Transcrição

Nesta primeira aula, aprenderemos como preparar o ambiente: o que precisamos instalar e configurar para acompanhar as aulas. Deveremos ter na máquina três itens: o Java propriamente, JRE ou JDK — a segunda opção possui mais documentação, mas para o nosso objetivo neste curso o JRE é o suficiente.

Usaremos a versão Java 10, mas as versões 8 ou 9 também são suficientes. Talvez, quando você realizar esse curso, já existam versões mais recentes, como 11 ou 12. Não há problema algum em utilizá-las pois o Java trabalha com servidor e as versões recentes sempre funcionam.

Observe a versão instalada na linha de comando:

```
C:\Users\Alura>java -version
java version "10.0.2" 2018-07-17
Java(TM) SE Runtime Environment 18.3 (build 10.0.2+13)
Java HotSpot(TM) 64-Bit Server VM 18.3 (build 10.0.2+12,mixed
mode)
```

[COPIAR CÓDIGO](#)

Não usaremos o Java na linha de comando, então não há necessidade de utilizar quaisquer comandos por essa via. Nosso trabalho se dará no Eclipse, portanto é necessário que ele esteja instalado.

Diferentemente dos cursos básicos de Java, a [versão do Eclipse](https://www.eclipse.org/) (<https://www.eclipse.org/>) necessária para esse curso mais avançado apresenta alguns recursos e ferramentas a mais.

Atualmente, quando realizamos o download do Eclipse, ele vem com um instalador a ser executado. A diferença dessa versão para as anteriores é que precisamos instalar o Eclipse IDE for Java EE Developers, um pacote de ferramentas para criação de aplicações web, como EE, JPA, JSF, Mylyn, entre outras.

Em seguida, iremos configurar o workspace, isto é, a pasta em que armazenaremos nossos projetos.

O último item que precisamos instalar é o servidor, que será necessário para trabalhar com Servlet. Nesse caso, usaremos o [Tomcat](http://tomcat.apache.org/whichversion.html) (<http://tomcat.apache.org/whichversion.html>), de longe o servidor mais famoso no mundo Java e que foi criado pela Apache, que é uma organização *open source*.

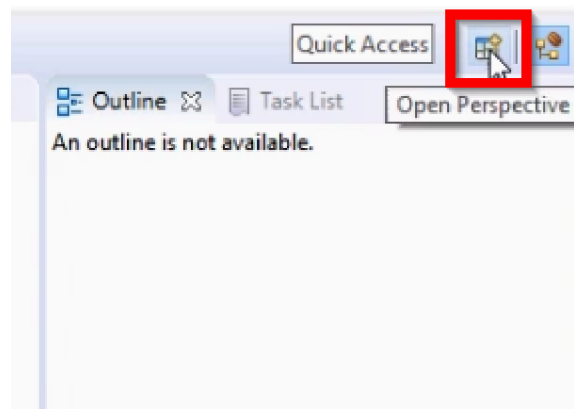
Usaremos a versão 9.0.11, que até o momento é a mais recente, mas você poderia utilizar a versão 7 ou 8 sem problemas. O que importa, de fato, é que a versão possa dar suporte para o Servlet 3.0 em diante.

Lembrando que o ideal é que você utilize as mesmas versões apresentadas no curso, para que tenhamos o máximo de sincronia e possamos desenvolver o projeto sem problemas.

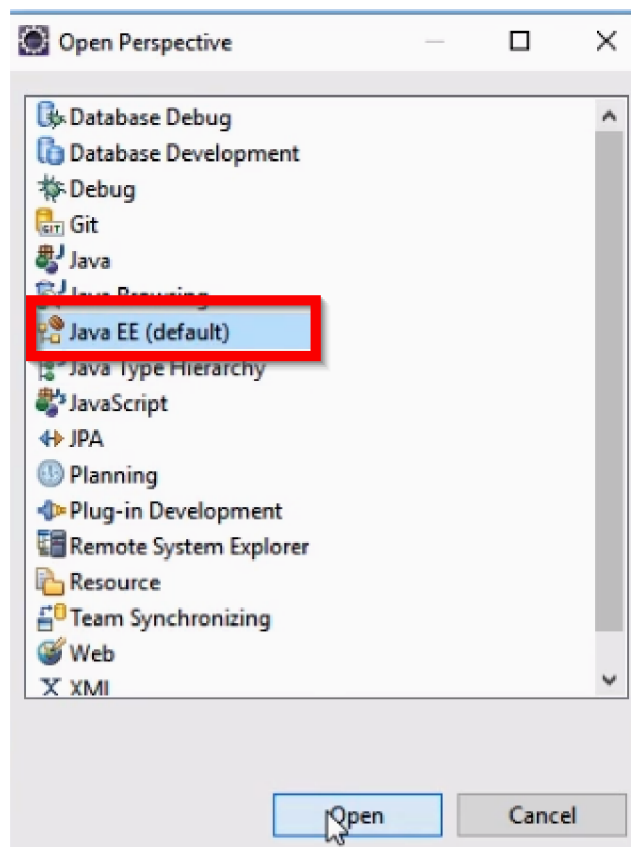
No site do [Tomcat](http://tomcat.apache.org/whichversion.html) (<http://tomcat.apache.org/whichversion.html>) você encontrará a versão para ser baixada em zip, e para os usuários de Linux está disponível o arquivo `tar.gz`. Trata-se de um arquivo pequeno, cerca de 10.268 KB. Depois de realizar o download do arquivo, basta extraí-lo.

Em seguida, precisaremos associar o Eclipse ao Tomcat, afinal não queremos inicializá-lo via linha de comando, pois no dia a dia nós inicializamos o servidor a partir do ambiente de desenvolvimento.

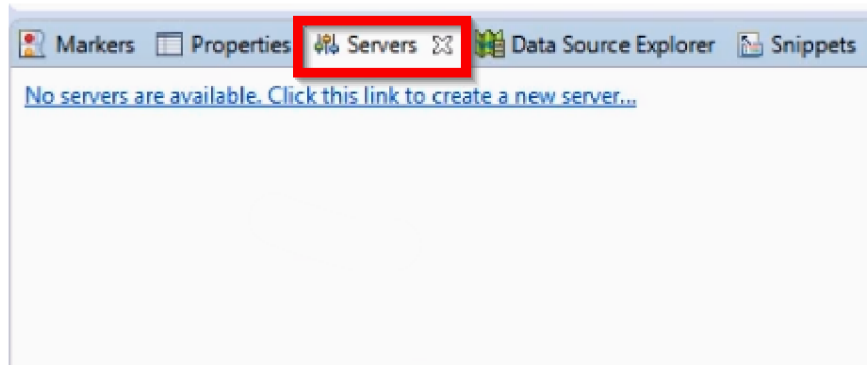
Com o Eclipse aberto, selecionaremos o ícone "Open Perspective", na parte superior direita da tela.



Na nova janela, selecionaremos a opção "Java EE (default)", que é a opção padrão.



A próxima etapa é configurarmos o Tomcat. Por meio do "Quick Access" — buscador na parte superior direita da tela — no Eclipse, digitaremos a palavra "Servers". Essa opção também pode ser acessada via aba "Servers".



Clicaremos no link disponível na área "Servers" e começaremos a realizar a configuração.

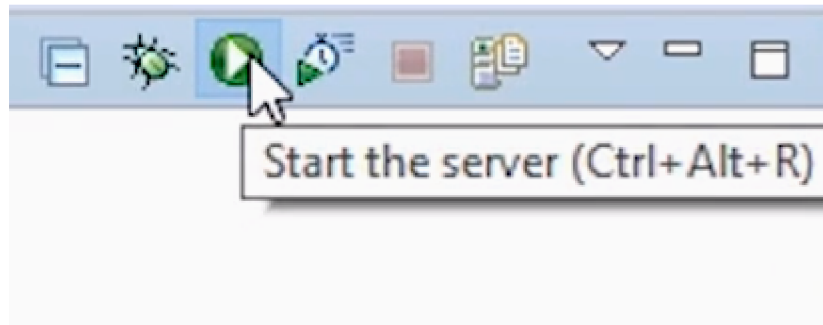
Na nova janela de diálogo "*Define a New Server*", clicaremos sobre "Apache > Tomcat v9.0 Server". Existem algumas opções editáveis, como "*Server's host name*", que manteremos com o modelo padrão. Em seguida, pressionaremos o botão "*Next*".

Teremos a opção "*Tomcat installation directory*", na qual colocaremos o diretório em que o servidor foi salvo na máquina.

Na opção "JRE", selecionaremos "jre-10.0.2" (às vezes o Eclipse já detecta que estamos utilizando o Java 10). Feito isso, pressionaremos o botão "*Finish*".

Desse modo, na aba "*Servers*", será exibida a informação `Tomcat v9.0 Server at localhost [Stopped, Republish]`. O Eclipse automaticamente criará um novo projeto para guardar as configurações do servidor, como podemos ver na área "Project Explorer", "*Servers > Tomcat v9.0 Server at localhost-config*". Não podemos apagar ou fechar o projeto, pois, caso isso seja feito, o Eclipse não conseguirá mais acessar as configurações.

Para iniciar o servidor, clicaremos sobre o ícone "*Start the server*" ou usaremos o atalho "Ctrl + Alt + R".



Ao clicarmos na aba "Console", poderemos ver a saída em vermelho. Não devemos associar essa cor com alguma mensagem de erro, já que é possível verificar na última linha que o servidor foi inicializado com sucesso:

INFORMAÇÕES: Server startup in 520 ms

Com o Tomcat rodando, poderemos acessá-lo!

Mas por que escolhemos trabalhar com este servidor? Não sabemos ainda o que são os tais "Servlets", mas vamos esclarecer alguns pontos: estamos programando em Java, queremos trabalhar na web e, no que diz respeito ao Tomcat, o mais importante é que ele utiliza o protocolo HTTP e HTML.

A [página da Alura \(https://www.alura.com.br/\)](https://www.alura.com.br/), por exemplo, é acessada por meio do navegador e utiliza o protocolo HTTP. O que o navegador exibe ao usuário é o resultado de uma página HTML, portanto precisamos aprender mais a fundo sobre como esses sistemas operam.

De volta ao Eclipse, sabemos que o nosso servidor está no ar, portanto já temos um processo Java sendo executado. Dessa forma, é possível utilizar o protocolo HTTP para acessar o servidor.

Abriremos uma nova aba no navegador Chrome, mas você pode utilizar o Firefox ou qualquer outro de sua preferência. Em algum momento na configuração inicial, foi explicitado o endereço do servidor, que é `http://localhost` — o padrão de endereço para qualquer máquina, independentemente do sistema operacional. Porém, se escrevermos somente essa informação, não poderemos acessar o servidor e seremos levados a uma página de erro, **alegando um problema de conexão**. Isso ocorreu pois, além do endereço, faltou explicitarmos a **porta específica**.

Quando não explicitamos uma porta na configuração, a porta padrão `80`, que faz parte do TCP utilizado pelo protocolo HTTP, é selecionada. Porém, o Tomcat não é executado na porta padrão, e sim na porta `8080`. Portanto, precisamos explicitá-la.

Ao acessarmos o endereço `http://localhost:8080` no navegador, teremos uma nova mensagem de erro, mas com um formato diferente: não há mais um erro de conexão, pois essa conexão com o servidor foi de fato estabelecida, observemos a mensagem:

HTTP Status 404 - Not Found

type Status Report

Message/

Description The origin server did not find a current representation for the target resource or is not willing to disclose that one exists.

Apache Tomcat/9.0.11

O Tomcat recebeu a chamada do navegador, e respondeu que não há nada a ser exibido. No mundo do protocolo HTTP, o status "404" significa que nenhum conteúdo foi encontrado. Mesmo que acessássemos essa página por outra máquina, como outro computador ou um celular (por meio do IP da máquina que está rodando o servidor), teríamos a mesma mensagem de erro.

Conseguimos instalar e executar o Tomcat, e nas próximas aulas começaremos a discutir de fato sobre Servlet e a criar nosso projeto Java.