

08

Mão na massa: javadoc

Sabemos que um código produzido por um desenvolvedor Java pode ser utilizado por outros desenvolvedores. Nesse sentido, documentar o código e facilitar seu uso por outros desenvolvedores é uma boa prática a ser seguida.

Sobre a geração de documentação, podemos usar o Javadoc, já para distribuição podemos empacotar a aplicação em um jar . Vamos começar pelo Javadoc.

Exercício Javadoc

O Javadoc só estará disponível para instalações de JDK's e não para JRE's. Certifique-se de estar usando uma JDK.

1 - Quando criamos nossas classes, nada mais justo do que adicionarmos a informação sobre o autor. Nesse sentido, vamos alterar a classe `Cliente.java` e nela adicionar a meta informação sobre o autor e sua versão.

```
/**
 * Classe que representa um cliente no ByteBank
 *
 * @author Nico Steppat
 * @version 0.1
 */
public class Cliente {
    /* código omitido */
}
```

Lembre-se que através do atalho `/**` o Eclipse automaticamente adicionar a documentação com `@author` para você. Vale a pena se habituar com esse atalho.

2 - Faça a mesma coisa para a classe `Conta.java` .

3 - Sabemos que uma documentação pode ir além do nome do autor e versão, por exemplo, documentando métodos e construtores. Utilizando o atalho `/**` , documente o construtor da classe `Conta.java` . O resultado da documentação deve ficar assim:

```
/**
 * @param agencia
 * @param numero
 */
public Conta(init agencia, int numero) {
    /* código omitido */
}
```

O atalho `/**` apenas adiciona a informação sobre os parâmetros do método, sendo de responsabilidade do desenvolvedor completar a documentação.

```
* @param agencia
* @param numero
*/
public Conta(init agencia, int numero) {
    /* código omitido */
}
```

4 - Ainda na classe `Conta.java`, documente o método `saca`. Utilize mais uma vez o atalho `/**`:

```
/**
 * Valor precisa ser maior que o saldo
 * @param valor
 * @throws SaldoInsuficienteException
*/
public void saca(double valor) throws SaldoInsuficienteException {
    /* código omitido */
}
```

5 - Através do *Quick Access*, procure por **Javadoc** em `View`. Selezionando a opção, será exibida a aba "Javadoc". Experimente clicar nas classes que você acabou de documentar. Na aba "javadoc" será exibida a documentação da classe.

6 - Além de exibirmos a documentação clicando em cada classe documentada, podemos gerar a documentação como arquivo para ser consultado. Para isso, acesse o menu `Project -> Generate Javadoc`.

Na lista de projetos exibidos, não esqueça de selecionar o projeto `bytebank-herdando-conta`. Certifique-se que em `Javadoc command` esta preenchido com o caminho da ferramenta `javadoc`. Utilize na opção `User standard docket` o caminho padrão projeto já preenchido, sem qualquer alteração. Por fim, clique no botão "Finish" para gerar a documentação.

7 - Verifique a pasta `doc` gerada com a documentação. Acesse o arquivo `doc/resources/index.html`. Essa página é o ponto de entrada para a documentação. Passeie pelas classes documentadas verificando o resultado.