

Programação II

Aula 01

Evandro J.R. Silva¹

¹Bacharelado em Ciência da Computação
Estácio Teresina

Sumário

1 JDK x JVM x JRE

2 Programação Orientada a Objetos
■ Classes

3 FIM

JDK x JVM x JRE

- Um dos trunfos do Java é ser **multiplataforma**: o mesmo código pode ser executado em qualquer máquina!

JDK x JVM x JRE

- Isso é possível através do **JVM** (*Java Virtual Machine*). Quando compilamos o código a JVM o traduz para **bytecodes**, os quais podem ser interpretados e executados em qualquer máquina. Para programar e continuar programando em Java não é necessário entender a fundo como isso funciona. Grosso modo, é análogo ao *Assembly* para C e C++.

JDK x JVM x JRE

- A JVM executa os programas compilados em bytecode. Isso significa que programas criados em outras linguagens podem também ser executadas se tiverem sido *traduzidas* para bytecodes: Scala, JRuby, Jython, Clojure, etc.

JDK x JVM x JRE

- Como a JVM executa o seu código?
- O código compilado é lido junto com as bibliotecas padrões do Java, que estão no **JRE** (*Java Runtime Environment*, ou Ambiente de Execução Java)
 - Na verdade a JVM vem no JRE, além das bibliotecas padrão. É o JRE que provê o executável **java.exe** a qual executa a classe que tem o método **main(String args[])**.

JDK x JVM x JRE

- E como eu posso programar em Java?
 - É aqui que aparece o **JDK** (*Java Development Kit*, ou Kit de Desenvolvimento Java). Quando você baixa o JDK ele vai com todo o básico necessário para se programar em Java, incluindo JRE.
 - A partir daqui, você pode abrir um arquivo de texto e começar a programar! Basta salvar com a extensão **.java**

JDK x JVM x JRE

- Quanto maior e mais complexo o sistema a ser desenvolvido, mais difícil fica para lidarmos com o código. Programar no bloco de notas vai se tornando cada vez mais um desafio!

JDK x JVM x JRE

- Neste ponto entram as IDEs (*Integrated Development Environment*, ou Ambiente de Desenvolvimento Integrado). São programas que te ajudam a fazer programas!

JDK x JVM x JRE

- Cada IDE tem suas características e funcionalidades, algumas mais completas (auto completar, assistência de documentação, verificação do código antes da compilação, destaque de sintaxe (*syntaxe highlight*), GUI, etc.

JDK x JVM x JRE

- Cada IDE tem suas características e funcionalidades, algumas mais completas (auto completar, assistência de documentação, verificação do código antes da compilação, destaque de sintaxe (*syntaxe highlight*), GUI, etc.
 - Eclipse;
 - Netbeans;
 - IntelliJ IDEA;
 - BlueJ;
 - etc.

Programação Orientada a Objetos

- O que é? Onde vive? O que come?

Programação Orientada a Objetos

- O que é? Onde vive? O que come?
 - **Paradigma de Linguagens de Programação** baseado no conceito de objetos.

Programação Orientada a Objetos

- O que é Paradigma de Linguagens de Programação?

Programação Orientada a Objetos

- O que é Paradigma de Linguagens de Programação?
 - Uma maneira de classificar as linguagens de programação de acordo com suas características.
 - Mais detalhes na disciplina Paradigmas de Linguagens de Programação.

Programação Orientada a Objetos

- E o que é um objeto?

Programação Orientada a Objetos

- E o que é um objeto?
 - Combinação de variáveis, funções e estruturas de dados.
 - Em Java, essa combinação é feita através do **instanciamento de classes**.

Programação Orientada a Objetos

- E o que é uma classe?

Classes

- Combinação de atributos (variáveis e estruturas de dados) e métodos (funções).
- A descrição de *alguma coisa* e de como ela pode ser manipulada.
- Exemplo: Conta Bancária
 - **Atributos:** Número da conta, nome do titular, saldo.
 - **Métodos:** Sacar, depositar, transferir.

Classes

■ Vantagens

- Descrição clara dos elementos que vão fazer parte do projeto.
- Modularidade/reusabilidade (tem impacto no tempo de desenvolvimento)

Classes

- Ok, entendi o que é uma classe, mas ainda não entendi direito o que é um objeto
...

Classes

- Ok, entendi o que é uma classe, mas ainda não entendi direito o que é um objeto
 - ...
 - Enquanto a classe descreve *algo*, o objeto é esse *algo*, ou seja, uma **instância** dessa classe.
 - criar uma instância de uma classe é reservar um espaço na memória que possa conter seus atributos, cujos valores podem ou não ser modificados.

FIM

- Levando em consideração que vocês já sabem Java, essa aula foi apenas uma pequena revisão do Java.
- **Tarefa de casa:** Capítulos 2, 3, 4 e 5 do livro **Java: Como Programar** do Deitel, oitava edição. Disponível nas bibliotecas virtuais (creio que na biblioteca física também).
- Os capítulos citados são apenas para revisar o Java!

Até a próxima!