

LTP 1 – Java Básico

Aula 1

Tópicos

- 1 - Introdução ao Java
- 2 - Bibliotecas de classe do Java
- 3 - Fase de um programa em Java
- 4 - Primeiro Programa em Java (Exemplo.java)
 - 4.1 - Comentários
 - 4.2 - Declaração da Classe
 - 4.3 - Método main
 - 4.4 - Objeto de Saída Padrão
 - 4.5 - Ambiente de desenvolvimento
 - 4.6 - Onde buscar ajuda?
- 5 - Exercícios

1 - Introdução ao Java

Java é uma linguagem de programação orientada a objeto desenvolvida na década de 90 pela Sun Micro systems.

Diferentemente das linguagens convencionais, que são compiladas para código nativo. O código de um programa de computador escrito na linguagem Java é compilado para uma forma intermediária de código denominada bytecode, que é interpretada pelas Máquinas Virtuais Java (JVMs). É essa característica que faz com que os programas Java sejam independentes de plataforma, executando em qualquer sistema que possua uma JVM.

A linguagem Java foi projetada tendo em vista os seguintes objetivos:

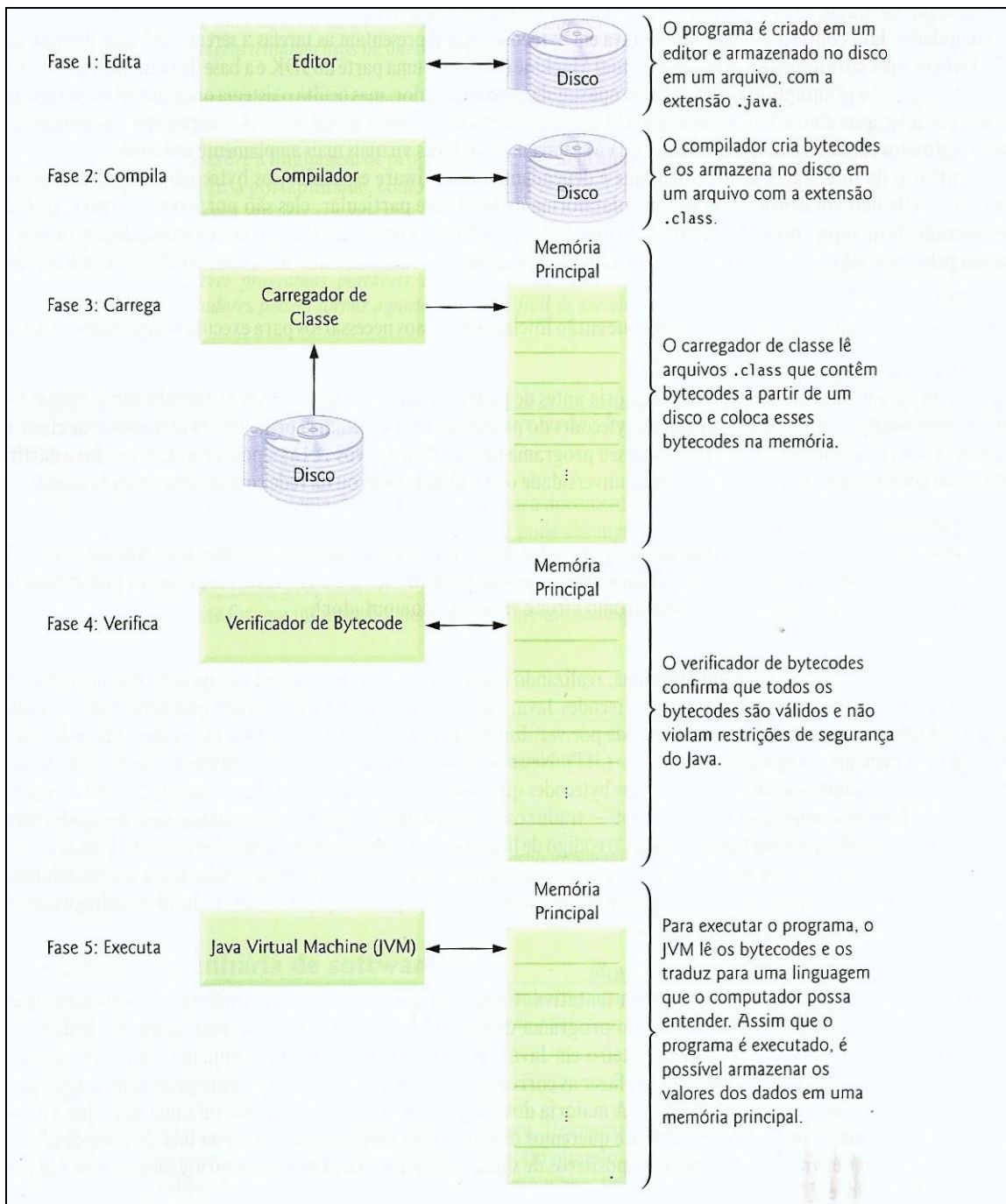
- Orientação a objetos - Baseado no modelo de Simula67;
- Portabilidade - Independência de plataforma - "escreva uma vez, execute em qualquer lugar" ("write once, run anywhere");
- Recursos de Rede - Possui extensa biblioteca de rotinas que facilitam a cooperação com protocolos TCP/IP, como HTTP e FTP;
- Segurança - Pode executar programas via rede com restrições de execução;
- # Sintaxe similar a C/C++.

2 - Bibliotecas de classe do Java

Programas em Java são feitos de partes chamadas CLASSES. As classes incluem partes chamadas MÉTODOS, que realizam tarefas e retornam alguma informação ao concluir a execução. Você pode criar suas próprias classes e métodos, entretanto deve-se sempre que possível utilizar as bibliotecas de classe existentes, também conhecidas como APIs (Application Programming Interfaces)

3 - Fase de um programa em Java

Os programas em Java passam por 5 fases: edição, compilação, carga, verificação e execução.



Fase 2: o programa compila o arquivo `.java` e cria um outro arquivo com a extensão `.class` com os bytecodes, que representam as tarefas a serem executadas pela JVM (Java Virtual Machine), que não depende de um hardware específico, tornando o código portátil.

Fase 3: os bytecodes (arquivo `.class`) são carregados na memória juntamente com qualquer outro arquivo `.class` que seu programa utiliza.

Fase 4: é feita a verificação para verificar se os programas Java que chegam pela rede não danificam os arquivos ou sistema, tornando o programação Java mais segura.

Fase 5: se tudo correr bem, a JVM executa os bytecodes, realizando assim as ações especificadas pelo programa. Senão, podem produzir:

- Erros de tempo de execução fatais: não termina de rodar.

- Erros de tempo de execução não-fatais: roda mas dá um resultado errado.

4 - Primeiro Programa em Java (Exemplo.java)

```
/**
 * testando exemplos
 */
public class Exemplo
{
    public static void main(String args[ ])
    {
        System.out.println("Primeiro Programa!");
    } //fim do método principal
} //fim da classe exemplo1
```

4.1 - Comentários

Você pode fazer de algumas formas:

// indica que o restante da linha é um comentário, que servem para documentar um programa e melhorar a legibilidade. É conhecido como comentário de fim de linha.

/* */ é um comentário que pode se estender por várias linhas. Ou seja, o compilador Java ignora tudo o que estiver entre /* e */

/** */ que são comentários no estilo javadoc que os utiliza para gerar a documentação do seu programa no formato HTML.

Observação: erro comum é esquecer um dos delimitadores de comentários.

4.2 - Declaração da Classe

Cada programa em Java consiste em pelo menos uma declaração de classe que é definida por você. O nome de uma classe Java é um identificador, que deve seguir estas regras:

- Pode usar letras, dígitos, sublinhados ou cifrão;
- Não pode começar com número;
- Não pode ter espaços em branco (em nomes compostos);
- O nome da classe public deve ser igual ao nome do arquivo com a extensão .java;
- Todo nome de uma classe deve começar com letra maiúscula e as demais minúsculas;
- As chaves { e } indicam o início e o fim do corpo da classe.

4.3 - Método main

Para um aplicativo Java devemos ter um método definido exatamente da seguinte forma:

```
public static void main(String args[ ])
ou
public static void main(String[ ] args)
```

Os parênteses depois do identificador main indica que se trata de um método.

Um método em Java é equivalente a uma função, subrotina ou procedimento em outras linguagens de programação. Não existe em Java o conceito de métodos globais. Todos os métodos devem sempre ser definidos dentro de uma classe.

- As chaves { e } indicam o início e o fim do corpo do método.

4.4 - Objeto de Saída Padrão

Para imprimir uma string de caracteres, contida entre aspas duplas, podemos utilizar o objeto `System.out`

O método `System.out.println` exibe uma linha de texto e coloca o cursor no início da linha de baixo (quebra de linha), que termina com um ponto e vírgula.

O Argumento desse método é a string entre as aspas duplas.

4.5 - Ambiente de desenvolvimento

Para codificar um programa em Java basta utilizar um “bloco de notas” e salvar com uma extensão `.java`, mas existem muitos ambientes de desenvolvimento integrados (IDE's) que estão disponíveis no mercado, muitos de forma gratuita. Por exemplo: Eclipse e NetBeans.

O NetBeans IDE é um ambiente de desenvolvimento integrado (IDE) gratuito e de código aberto para desenvolvedores de software nas linguagens Java. Seu principal concorrente é o Eclipse, que é uma outra IDE desenvolvida em Java pela IBM, com código aberto, para a construção de programas de computador.

Você pode baixar o NetBeans num dos seguintes endereços:

- http://netbeans.org/downloads/start.html?platform=windows&lang=pt_BR&option=java
- <http://www.baixaki.com.br/download/netbeans-ide.htm>
- <http://www.superdownloads.com.br/download/193/netbeans-ide/>

Para rodar um programa em Java no NetBeans é preciso fazer o seguinte:

Arquivo -> Novo Projeto -> Escolha o projeto:

Em CATEGORIAS: selecione JAVA

Em PROJETOS: selecione APLICATIVO JAVA

Selecione: próximo

No campo NOME DO PROJETO: dê um nome (exemplos)

No campo CRIAR CLASSE PRINCIPAL: coloque Ex1, por exemplo

Clique em FINALIZAR

Edite o programa principal que já foi criado automaticamente pra você.

Para rodar o programa pressione Shift + F6

4.6 - Onde buscar ajuda?

<http://www.guj.com.br> ou <http://javafree.uol.com.br/>

5 - Exercícios

1. Baixe e instale o NetBeans no computador da sua casa
2. Crie um novo projeto e teste o exemplo desta aula
3. Implemente uma nova classe do mesmo projeto que mostra na tela esta frase usando o método `println()` "O primeiro programa a gente nunca esquece!".
4. Elabore um programa em Java que escreve seu nome completo na primeira linha, seu endereço na segunda, e o CEP e telefone na terceira.

5. Escrever um programa que mostre a seguinte figura na tela:

```
XXXXX
X    X
X    X
X    X
XXXXX
```

6. Escreva um programa que produza a seguinte saída na tela:

ALUNO(A)	NOTA
=====	=====
ALINE	9.0
MÁRIO	DEZ
SÉRGIO	4.5
SHIRLEY	7.0

7. Letra grande - Elabore um programa para produzir na tela a letra usando a própria letra. Se fosse 'L', seria assim:

```
L
L
L
LLLLL
```