

Introdução à Orientação a Objetos I

Rafael Silva Guimarães

Instituto Federal do Espírito Santo

rafaelg@ifes.edu.br

<http://rafaelguimaraes.net>

<https://github.com/rafaelsilvag/ifesJava/>

22 de Julho de 2015



INSTITUTO FEDERAL
ESPÍRITO SANTO

Parte 1

Histórico

A Linguagem Java

Ambiente de Desenvolvimento

Primeiro Contato

Parte 2

Tipos de Dados

Operadores

Referências



- ▶ Em 1995, a Sun Microsystem® anunciou o Java; não era apenas mais uma linguagem de programação, mas uma plataforma de desenvolvimento.
- ▶ Atualmente o Java está na versão 8 com algumas mudanças interessantes a caminho na versão 9, obviamente, de domínio da Oracle. Em 2007 a Sun Microsystem abriu o projeto Java para a comunidade através do projeto OpenJDK. Apesar disso, atualmente as VMs Oracle®, através da aquisição da Sun, e OpenJDK, não conseguem ter 100% de compatibilidade.



► Características da Linguagem Java:

- **Orientação a objetos:** é uma prática de programação já sólida no mercado e na maioria das linguagens de hoje.
- **Portabilidade:** Java é uma linguagem multiplataforma, ou seja, uma mesma aplicação pode ser executada em diferentes tipos de plataforma sem a necessidade de adaptação do código. Vamos para um exemplo prático!
- **Multithreading:** é o meio pelo qual se consegue fazer com que mais de um evento aconteça simultaneamente em um programa.

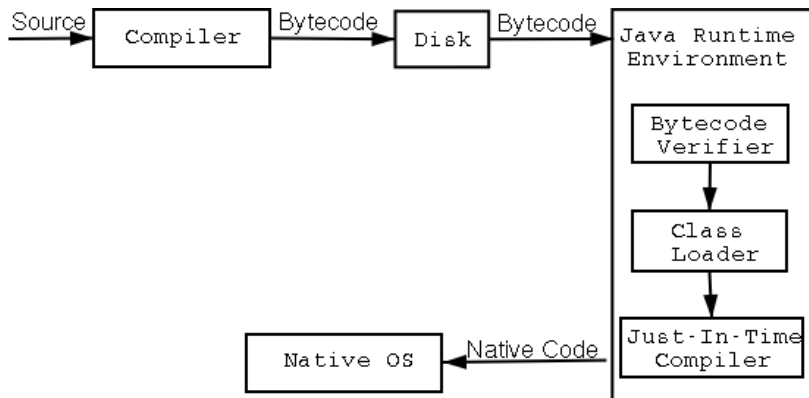


► Características da Linguagem Java:

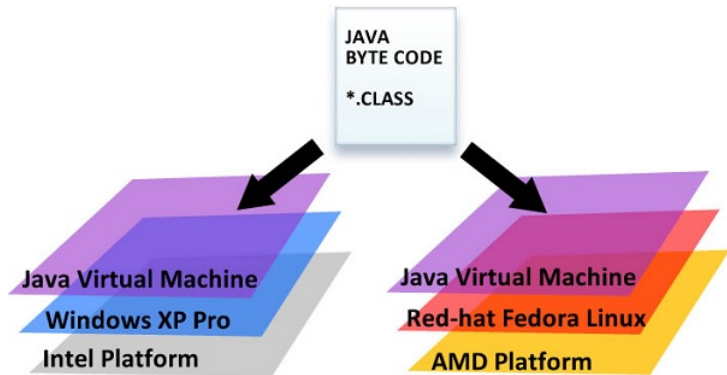
- **Suporte à comunicação:** Java detém de um grande conjunto de classes(bibliotecas) com funcionalidades específicas, ou seja, um alto nível de manipulação já estão encapsuladas em classes prontas, acelerando o desenvolvimento. Neste contexto, temos classes que já suportam a manipulação de tecnologias de comunicação, tais como: TCP/IP, HTTP, FTP, etc.
- **Acesso remoto a um banco de dados:** No Java utiliza-se classes prontas, criadas pela Sun Microsystem, para realizar a manipulação de SGBD.



A Plataforma Java



A Plataforma Java



Example (Primeiro Contato - PrimeiroContato01.java)

```
public class PrimeiroContato01 {  
    // Nome da Classe - Uma restrição forte do java  
    // é que devemos ter o nome do arquivo com o mesmo  
    // nome da classe  
    public static void main(String args[]){  
        // Função principal ou método principal  
        System.out.println("Testando...");  
    }  
}
```



- ▶ **Public:** É um qualificador do método ou da classe que indica que ele é acessível externamente a essa classe por outra classe.
- ▶ **Static:** Qualificador que indica que o método deve ser compartilhado por todos os objetos que são criados a partir dessa classe.
- ▶ **Void:** É o valor de retorno do método. Funcionalidade idêntica da utilizada na linguagem C. Aliás, verão que a sintaxe de muitos comandos são idênticas da linguagem C, principal causa de uma fácil transição entre estas linguagens.
- ▶ **Main:** Este é o nome do método que indica ao compilador o início do programa.



- ▶ A linguagem de programação Java não tem tipagem dinâmica, necessitando sempre mencionar o tipo de dado que será utilizado naquela determinada variável.
- ▶ Neste caso, a linguagem fornece alguns tipos primitivos da linguagem, como mostra na tabela abaixo:

Tipo	Qtd de Bits
char	16
byte	8
int	32
short	16
long	64
float	32
double	64
boolean	8 (true ou false)



Example (TiposDados01.java)

```
public class TiposDados01 {  
    // Função principal ou método principal  
    public static void main(String args[]){  
        //Declaração das variáveis  
        int x = 10, y = 20;  
        double dolar = 2.62;  
  
        System.out.println(x);  
        System.out.println(y);  
        System.out.println(dolar);  
    }  
}
```



- ▶ Os nomes de variáveis devem começar com letra, caracteres de sublinhado(_) ou cifrão (\$). **Não é permitido iniciar o nome de uma variável com número.**
- ▶ Por convenção, a linguagem Java utiliza o seguinte padrão para nomear as variáveis:
 - Quando o nome da variável for composto apenas por um caractere ou palavra, os caracteres devem ser minúsculos.
 - Quando o nome da variável tiver mais de uma palavra, a primeira letra, da segunda palavra em diante, devem ser maiúsculas. Todos os outros caracteres devem ser minúsculos.
- ▶ Exemplos: a, a1, x, tempo, nome, valorVenda, codigoFornecedor.



- ▶ Não existe constantes em Java, o que existe é um tipo de variável com comportamento semelhante a uma constante de outras linguagens. É na verdade um tipo de variável que não pode ter o seu conteúdo alterado depois de ter sido inicializada. Em Java, essa variável é chamada de final, exemplo:

```
final double pi = 3.14;  
final int miliSegundosporSegundos = 1000;
```



Operadores Aritméticos

Função	Sinal	Exemplo
Adição	+	$x + y$
Subtração	-	$x - y$
Multiplicação	*	$x * y$
Divisão	/	x / y
Resto da Divisão	%	$x \% y$
Sinal Negativo	-	-x
Sinal Positivo	+	+x
Incremento Unitário	++	++x ou x++
Decremento Unitário	--	--x ou x--

Tabela: Operadores Aritméticos



Função	Caractere(s) utilizados(s)	Exemplo
Igual	<code>==</code>	<code>x == y</code>
Diferente	<code>!=</code>	<code>x != y</code>
Maior que	<code>></code>	<code>x > y</code>
Maior ou igual a	<code>>=</code>	<code>x >= y</code>
Menor que	<code><</code>	<code>x < y</code>
Menor ou igual a	<code><=</code>	<code>x <= y</code>

Tabela: Operadores Relacionais

Operador	Ação
&&	AND (E)
	OR (OU)
!	NOT (NÃO)

Tabela: Operadores Lógicos



Cornell, G. ; Horstmann, S. C.

Core Java 2: Fundamentos (vol.1.)

Pearson Makron Books São Paulo



Cornell, G. ; Horstmann, S. C.

Core Java 2: Recursos Avançados (vol.2.)

Pearson Makron Books São Paulo



Sebesta, R.W.

Conceitos de Linguagens de Programação

Bookman São Paulo



Deitel, Paul; Deitel, Harvey

Java - Como Programar

Pearson São Paulo

