

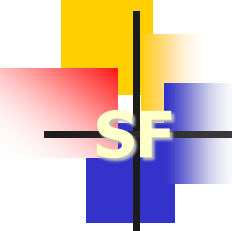
Programação Orientada a Objetos

Semana 01

Apresentação da Disciplina

Programação Orientada a Objetos

Prof. Dr. Sérgio Furgeri: sergio.furgeri@fatec.sp.gov.br



| Semana | Conteúdo |
|--------|-------------------------------|
| 1 | Apresentação da Disciplina |
| 2 | Classes Math e String |
| 3 | Estruturas de repetição |
| 4 | Manipulação de datas |
| 5 | Classes e Objetos |
| 6 | Encapsulamento e Serialização |
| 7 | Herança e Polimorfismo |
| 8 | Relacionamento entre Classes |
| 9 | Interface Gráfica |
| 10 | Manipulação de Arquivos |
| 11 | Banco de Dados |
| 12 | Padrão MVC, JPA |
| 13 | Acesso a Stored Procedures |
| 14 | Gráficos, PDF e Relatórios |
| 15 | XML e JSON |

Linguagens Orientadas a Objeto



[https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))



<http://www.techribbon.com/category/c/>



<http://viladosilicio.com.br/conhecendo-a-linguagem-python-parte-1/>



<http://www.technewsable.com/learn-programming-with-c-introduction/>

Linguagens de programação no Brasil na década passada

| Linguagem | 2003 | | 2004 | |
|----------------------------|--------------------|------|--------------------|------|
| | Nº. de Indicações* | %** | Nº. de Indicações* | %** |
| Delphi | 431 | 37,3 | 542 | 39,0 |
| Visual Basic | 251 | 21,8 | 435 | 31,3 |
| C++/Visual C++/C++ Builder | 206 | 17,9 | 317 | 22,8 |
| Java | 131 | 11,4 | 212 | 15,3 |
| SQL | 133 | 11,5 | 195 | 14,0 |
| HTML/DHTML | 108 | 9,4 | 181 | 13,0 |
| Clipper | 124 | 10,7 | 167 | 12,0 |
| XML | 73 | 6,3 | 142 | 10,2 |
| C/Visual C | 80 | 6,9 | 111 | 8,0 |
| ASP | 50 | 4,3 | 104 | 7,5 |
| Cobol | 73 | 6,3 | 86 | 6,2 |
| FoxPro/Visual FoxPro | 65 | 5,6 | 81 | 5,8 |
| Pascal | 55 | 4,8 | 78 | 5,6 |
| Java Script | 12 | 1,0 | 77 | 5,5 |
| Progress | 39 | 3,4 | 73 | 5,3 |
| Power Builder | 13 | 1,1 | 63 | 4,5 |
| Assembly | 38 | 3,3 | 48 | 3,5 |
| WebServices/.Net | 9 | 0,8 | 43 | 3,1 |
| Toolbook | 22 | 1,9 | 38 | 2,7 |
| Paradox | 29 | 2,5 | 34 | 2,4 |
| JSP/Servlet | 6 | 0,5 | 30 | 2,2 |
| PHP | 3 | 0,3 | 27 | 1,9 |
| Visual Objects | 15 | 1,3 | 23 | 1,7 |
| Informix | 5 | 0,4 | 17 | 1,2 |
| Lotus Notes/Lotus Script | 12 | 1,0 | 12 | 0,9 |

Fonte: Bancos de dados do IEES

Ranking das linguagens de programação

| Dec 2016 | Dec 2015 | Change | Programming Language | Ratings | Change |
|----------|----------|--------|----------------------|---------|--------|
| 1 | 1 | | Java | 17.856% | -3.12% |
| 2 | 2 | | C | 8.726% | -7.73% |
| 3 | 3 | | C++ | 5.335% | -0.61% |
| 4 | 4 | | Python | 4.239% | -0.19% |
| 5 | 7 | ▲ | Visual Basic .NET | 3.302% | +0.91% |
| 6 | 5 | ▼ | C# | 3.171% | -0.94% |
| 7 | 6 | ▼ | PHP | 2.919% | +0.13% |
| 8 | 8 | | JavaScript | 2.862% | +0.50% |
| 9 | 11 | ▲ | Assembly language | 2.539% | +0.61% |
| 10 | 9 | ▼ | Perl | 2.338% | +0.13% |

<http://www.tiobe.com/tiobe-index/>

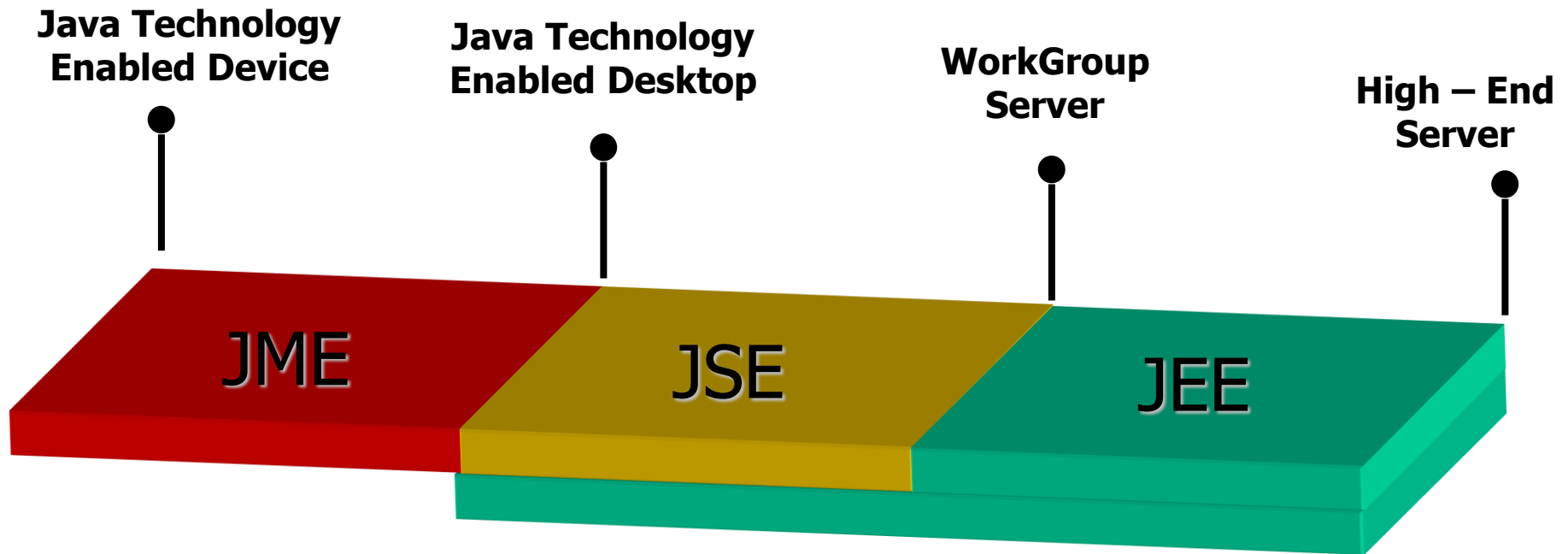
Histórico

- Desenvolvida pela Sun Microsystems;
- Originalmente, gerada para utilização em pequenos dispositivos eletrônicos;
- 1995 – deixa de ser apenas uma linguagem de programação para ser uma nova plataforma;
- Utilização na Web – criação de páginas interativas e dinâmicas;
- Expansão para dispositivos móveis.

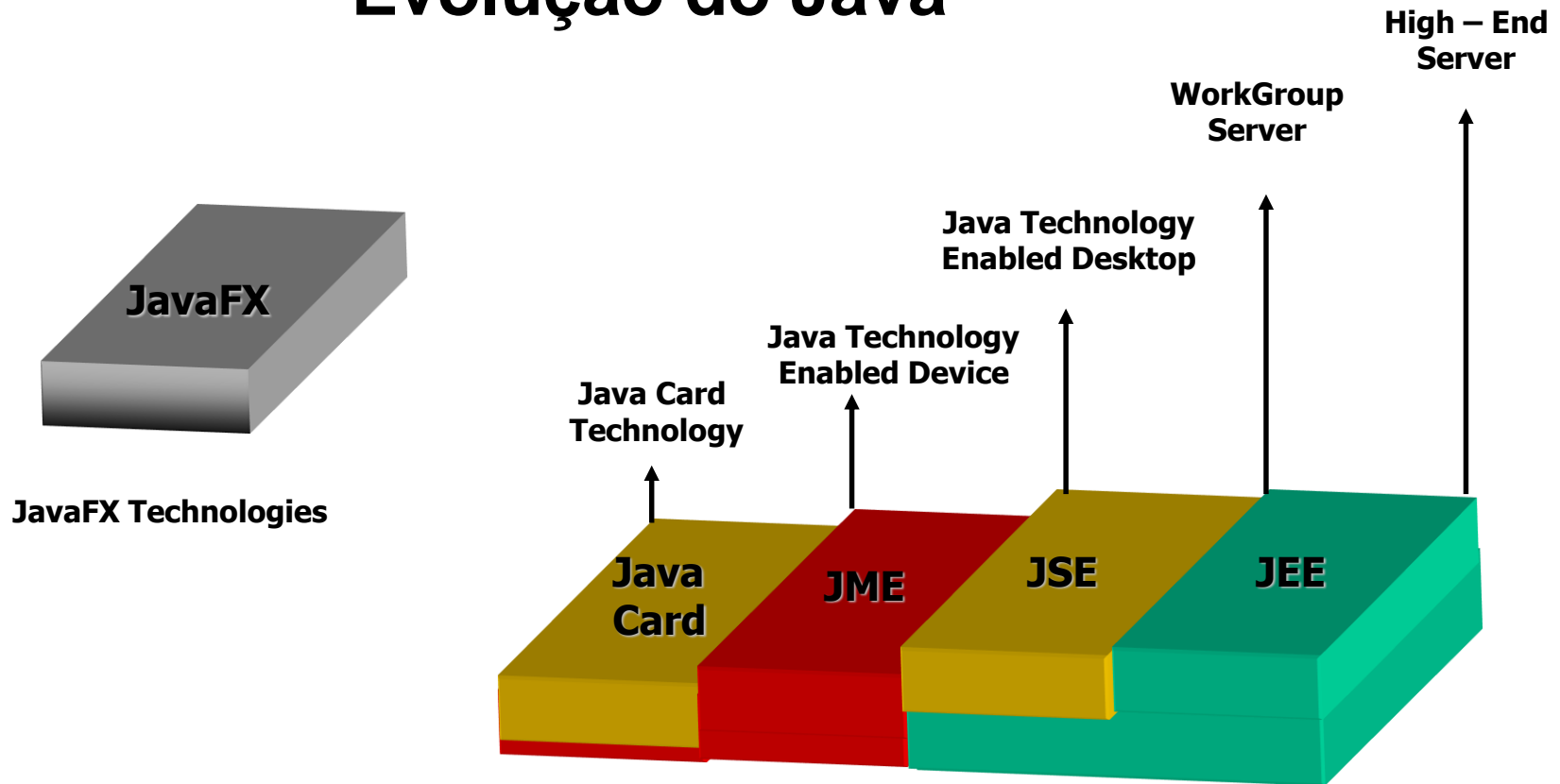
Algumas características de Java

- Sintaxe parecida com C / C++;
- Precisa de uma JVM (*Java Virtual Machine*);
- Voltada para a orientação a objetos;
- Portável – de pequenos dispositivos a mainframes;
- Multithreading – execução simultânea de vários eventos em um programa;
- Suporta aplicações cliente / servidor;
- Sem acesso a ponteiros do sistema operacional;

Evolução do Java



Evolução do Java



Fonte:

<http://www.javabuilding.com/academy/java-platform/os-alicerces-da-plataforma-java.html>

<http://www.oracle.com/technetwork/java/javafx/tech/index.html>

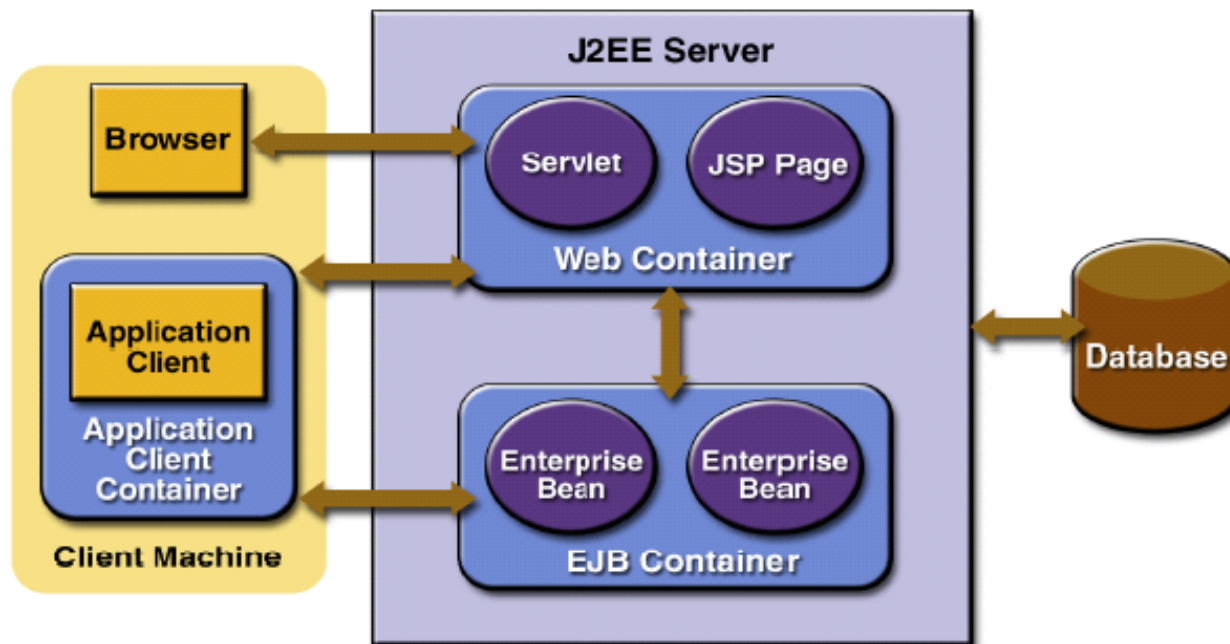
<http://www.oracle.com/technetwork/java/javacard/overview/index.html>

JSE (Java Standard Edition)

- Edição Standard da plataforma Java;
- Desenvolvimento de aplicações em geral;
- Base da JEE e Java Web Services;
- Dividida em dois grupos conceituais: Core Java e Desktop Java.
- JDK (Java Development Kit)

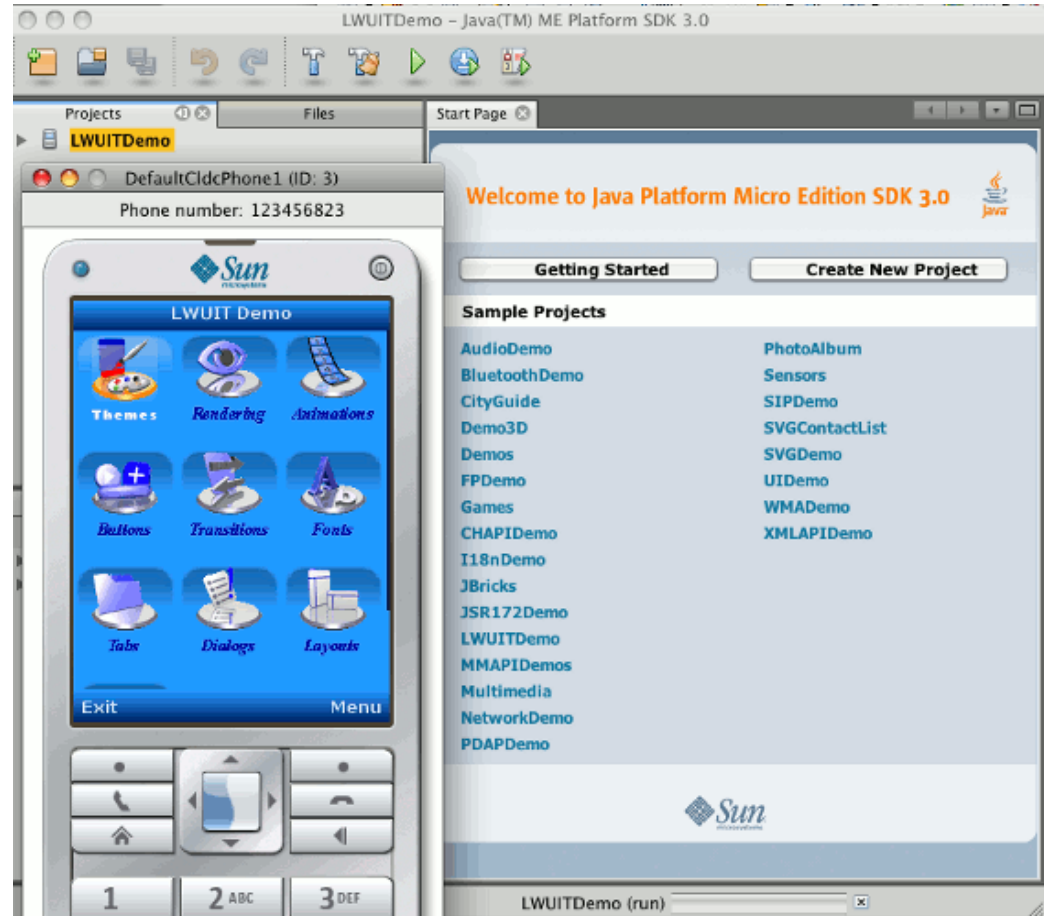
JEE (Java Enterprise Edition)

- Conceito / especificação definida pela Sun;
- Voltada ao desenvolvimento de grandes aplicações;
- Simplificação de aplicações multi-camadas;
- Baseados em componentes padronizados, modulares e reusáveis (EJB):



JME (Java Micro Edition)

- Edição Micro da linguagem Java;
- Utilizado para o desenvolvimento de aplicações para dispositivos móveis como celulares e PDA's;
- MIDlets – aplicações geradas no JME MIDP;



JAVAFX

- Especializada em criar interfaces gráficas para interação humana, com efeitos especiais e animações.
- Permite a criação para desktop, browser e celulares.
- Na web, é a concorrente da tecnologia Flash.
- Permite:
 - ✓ Criar interfaces gráficas
 - ✓ Animações
 - ✓ Desenhar na tela
 - ✓ Efeitos
 - ✓ Gráficos
 - ✓ Programar arrastando e soltando
 - ✓ Tocar vídeo e áudio



JAVAFX



Site que com vários artigos sobre JavaFX em português...muito bom!
<http://javafree.uol.com.br/catartigos/67/Tutoriais-JavaFX.html>

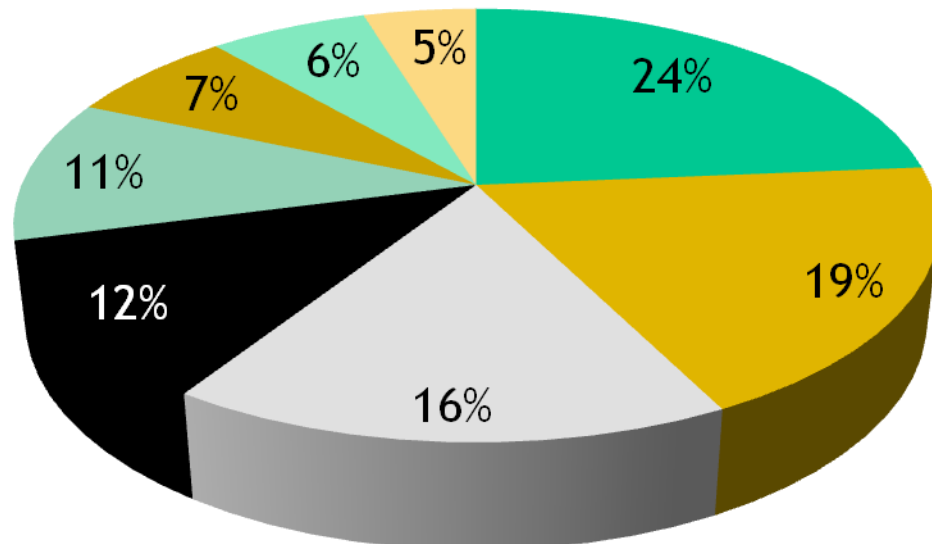
Ferramentas para desenvolvimento

- JDK (Kit Desenvolvimento Java);
- Bloco de Notas;

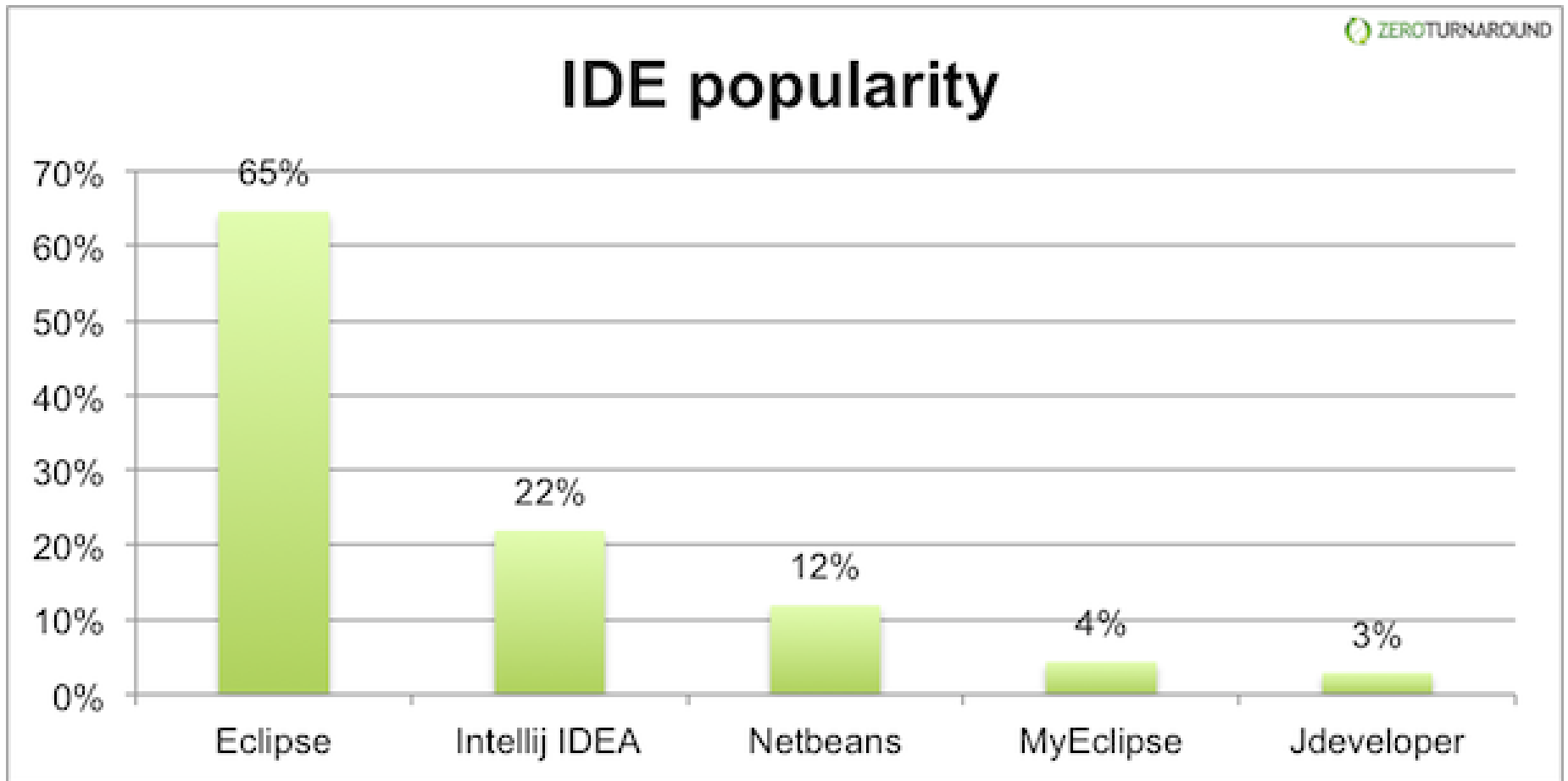


IDE's JAVA 2016

Eclipse IntelliJ IDEA NetBeans
BueJ JDveloper Dr Java
Android Studio Outros



Popularidade das IDEs



JDK - Kit de Desenvolvimento Java

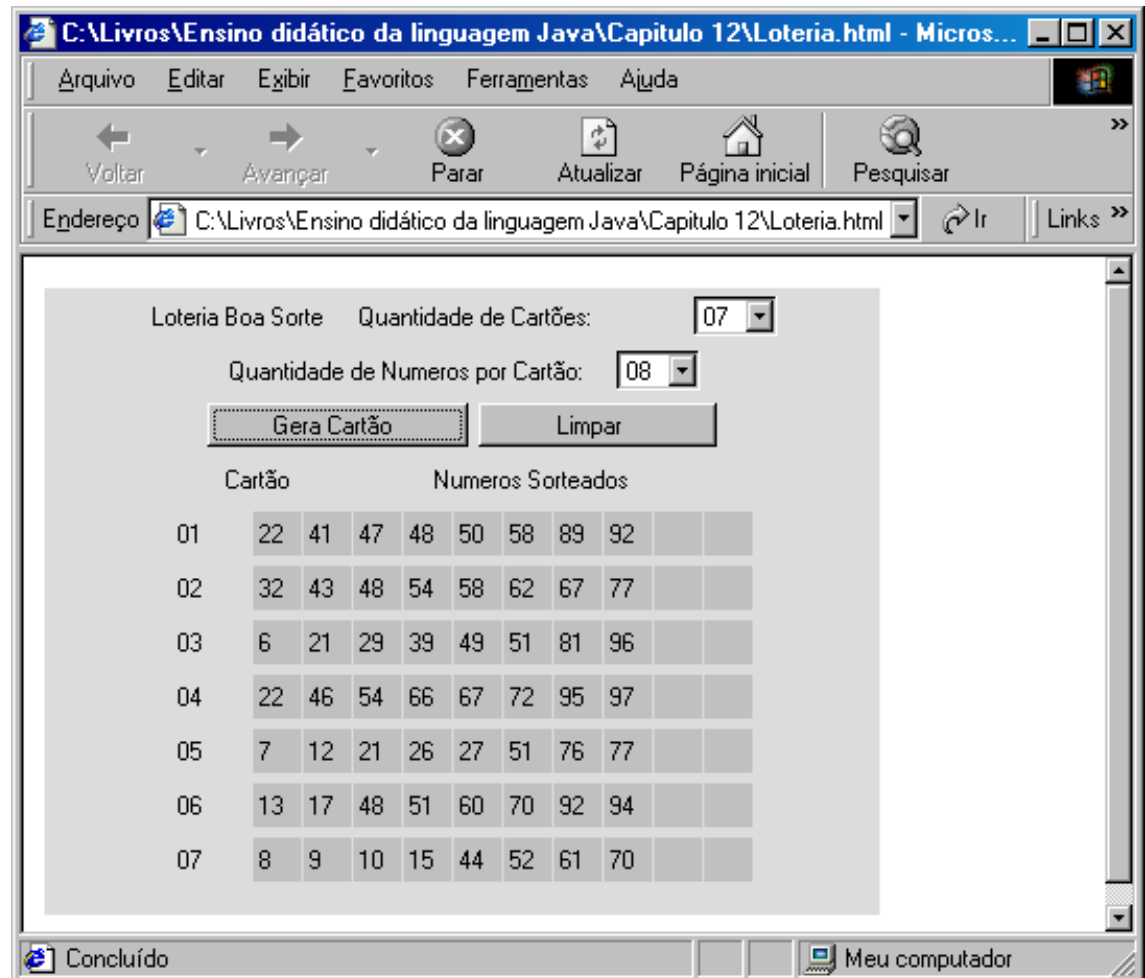
- Compilador (javac)
- Interpretador de bytecode (java)
- Visualizador de applets (appletviewer)
- Gerador de documentação (javadoc)
- Fornece pacotes standard para:
 1. Linguagem básica
 2. Sistema de janelas
 3. Construção de Applets
 4. Instruções para Entrada/Saída
 5. Comunicação em Rede

JavaScript

- Linguagem derivada do Java, mas o JavaScript não é Java;
 - Java – orientada a objetos.
 - JavaScript – baseada em objetos.
- Independente de plataforma, os scripts são interpretados pelo próprio browser;
- Não necessita de uma JVM para ser executada e não precisa ser compilada;
- Desenvolvida para ser utilizada junto com páginas HTML.

Applets

- Execução em páginas HTML visualizadas em browsers;
- AppletViewer da Sun;
- Interface mais sofisticada;
- Necessita que o java esteja ativado;



Servlets

- Aplicações Java hospedadas em servidores Web;
- Baseados em Requisições e Respostas;
- Semelhante a tecnologias como: CGI, ASP, JSP;
- Permanecem ativas durante todo o tempo;
- Necessitam ser alocadas em um container

JSP – Java Server Pages

- Tecnologia de desenvolvimento de aplicações Web;
- Criação de páginas dinâmicas, semelhante ASP;
- Parte integrante da plataforma JEE;
- Facilidade de elaboração e codificação;
- Permite realizar a separação da parte lógica (dinâmica) da parte visual (estática);
- Scriptlets – blocos de códigos inseridos na JSP;
- Suportam a utilização de JavaBeans;

JavaBeans

- Também conhecidos como Beans;
- Componentes projetados para serem unidades reutilizáveis;
- São utilizados em applets, sevlets, JSP etc.;

Geração de Telas

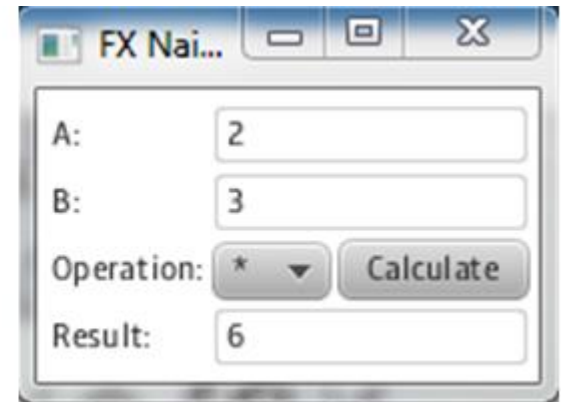
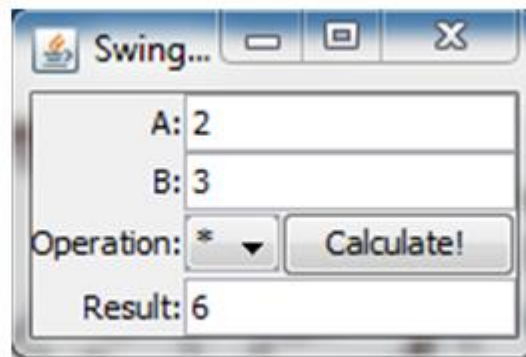
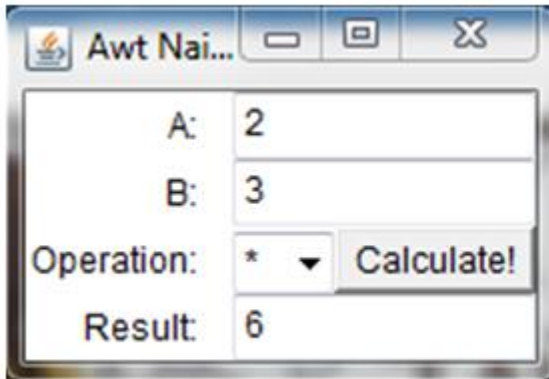
AWT



Swing

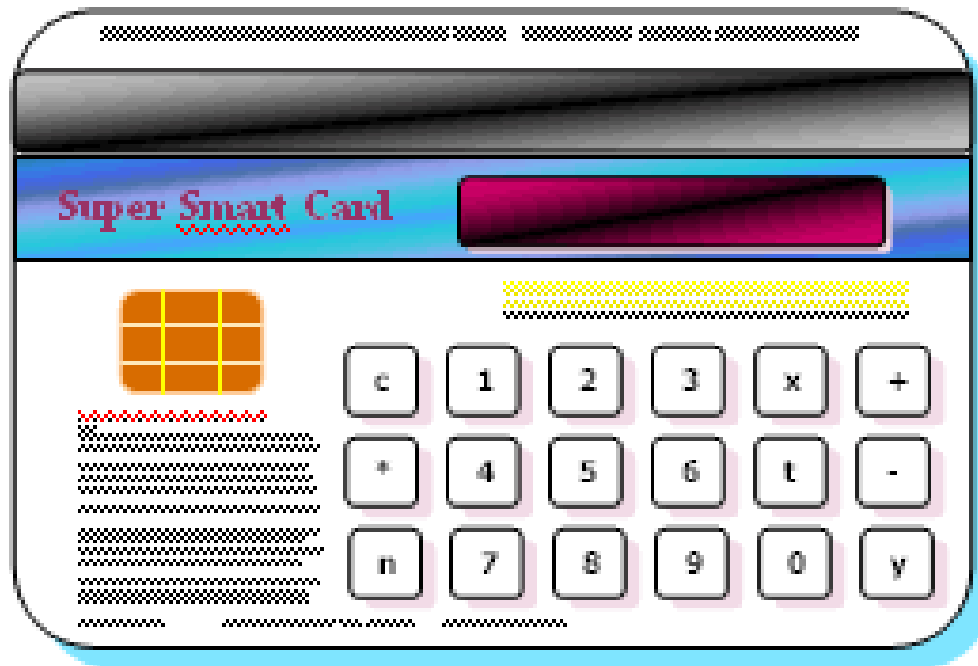


JavaFX



Java Card

- Desenvolvida pela Sun;
- Adaptação da Plataforma Java;
- Destinada ao desenvolvimento de aplicações para dispositivos como menor capacidade de processamento e de armazenamento;



Convenções usadas em Java

Nomes de arquivos

- Customer.java, RentalItem.java

Nomes de Classes

- Customer, RentalItem, InventorItem

Nomes de Pacotes

- semana01, br.com.fatec.poo

Nomes de Métodos

- `getCustomerName()`, `setRentalItemPrice()`

Standard para variáveis

- `customerName`, `customerCreditLimit`

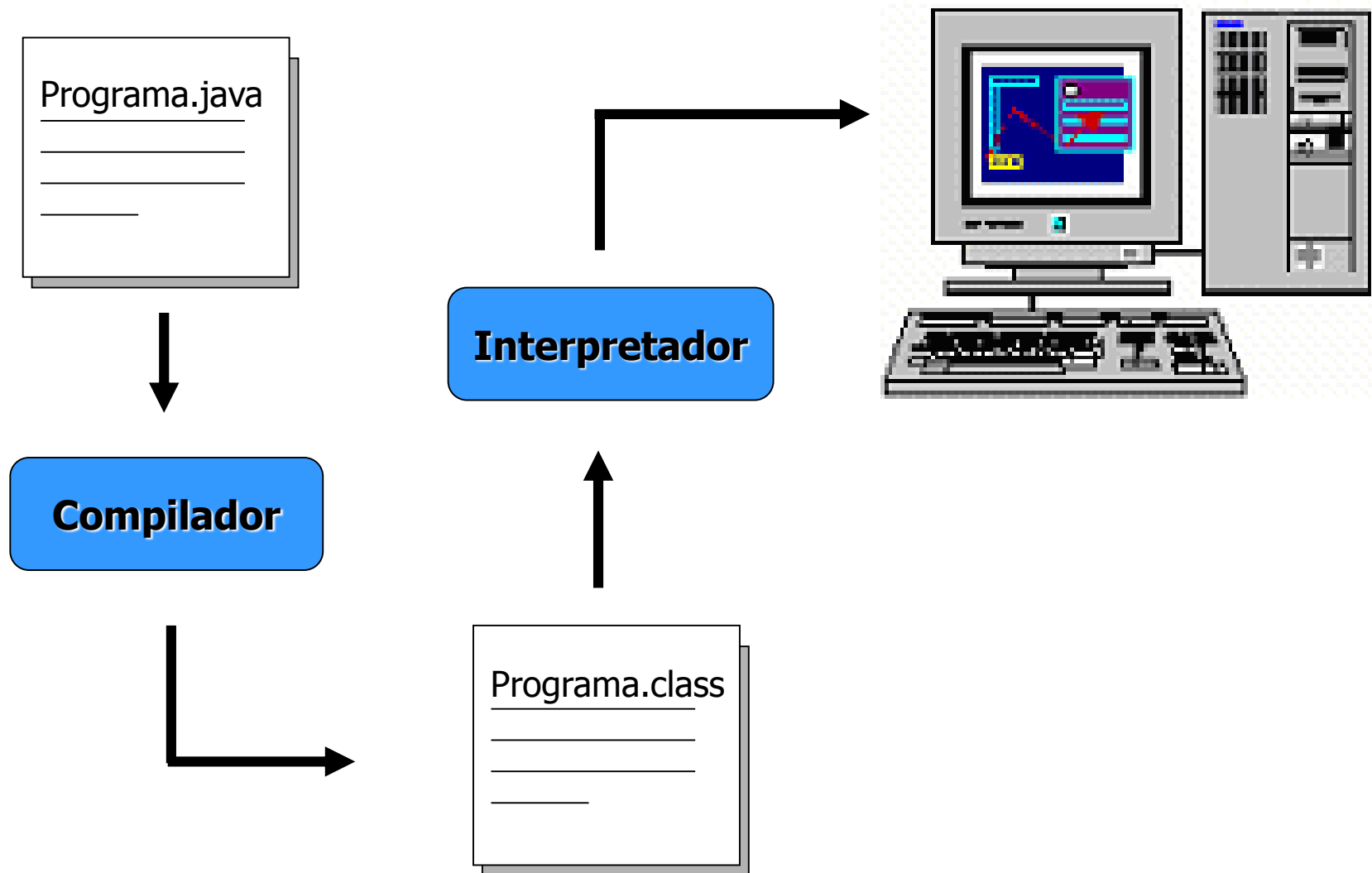
Standard para constantes

- `MIN_WIDTH`, `MAX_NUMBER_OF_ITEMS`

Desafios para os Profissionais Java

- Dominar as Técnicas de Orientação a Objetos;
- Conhecer as metodologias de engenharia de software;
- Ter fundamentos sólidos em Java (JSE);
- Saber como utilizar as classes disponíveis em Java;
- Obter as certificações técnicas;
- Dominar a Arquitetura Web e Internet;
- Conhecer Ferramentas de Desenvolvimento;
- Ficar atento na evolução do Java.

Ciclo de Desenvolvimento



Exemplo Java

```
public class Hello{  
    public static void main (String args[]){  
        System.out.println("Hello!");  
    }  
}
```

Operadores Aritméticos

| Operador | Significado | Exemplo |
|----------|---------------------------|------------|
| + | Adição | $a + b$ |
| - | Subtração | $a - b$ |
| * | Multiplicação | $a * b$ |
| / | Divisão | a / b |
| % | Resto da divisão inteira | $a \% b$ |
| - | Sinal negativo (- unário) | -a |
| + | Sinal positivo (+ unário) | +a |
| ++ | Incremento unitário | ++a ou a++ |
| -- | Decremento unitário | --a ou a-- |

Operadores Relacionais

| Operador | Significado | Exemplo |
|----------|------------------|---------|
| == | Igual | a == b |
| != | Diferente | a != b |
| > | Maior que | a > b |
| >= | Maior ou igual a | a >= b |
| < | Menor que | a < b |
| <= | Menor ou igual a | a <= b |

Operadores Lógicos

| Operador | Significado | Exemplo |
|----------|-------------------------|---------|
| && | E lógico (<i>and</i>) | a && b |
| | Ou Lógico (<i>or</i>) | a b |
| ! | Negação (<i>not</i>) | !a |

Entrada de dados via teclado

Java.io

```
import java.io.*
```

```
BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));
```

Scanner

```
import java.util.Scanner;
```

```
Scanner entrada = new Scanner (System.in);  
System.out.print("Digite seu nome: ");
```

JOptionPane

```
import javax.swing.JOptionPane;
```

```
a = JOptionPane.showInputDialog("Digite um Valor");
```

Conversores de tipo

Conversão de String:

```
int - Integer.parseInt( string )  
Float - Float.parseFloat( string )  
Double - Double.parseDouble( string )
```

Da mesma forma é possível fazer a conversão da String para os outros tipos:

```
double i = 42.0;
```

```
String str = Double.toString(i);
```


Estruturas em Java

Estrutura Condicional – Baseia a execução com base na condição verdadeira ou falsa.

if/else – switch/case

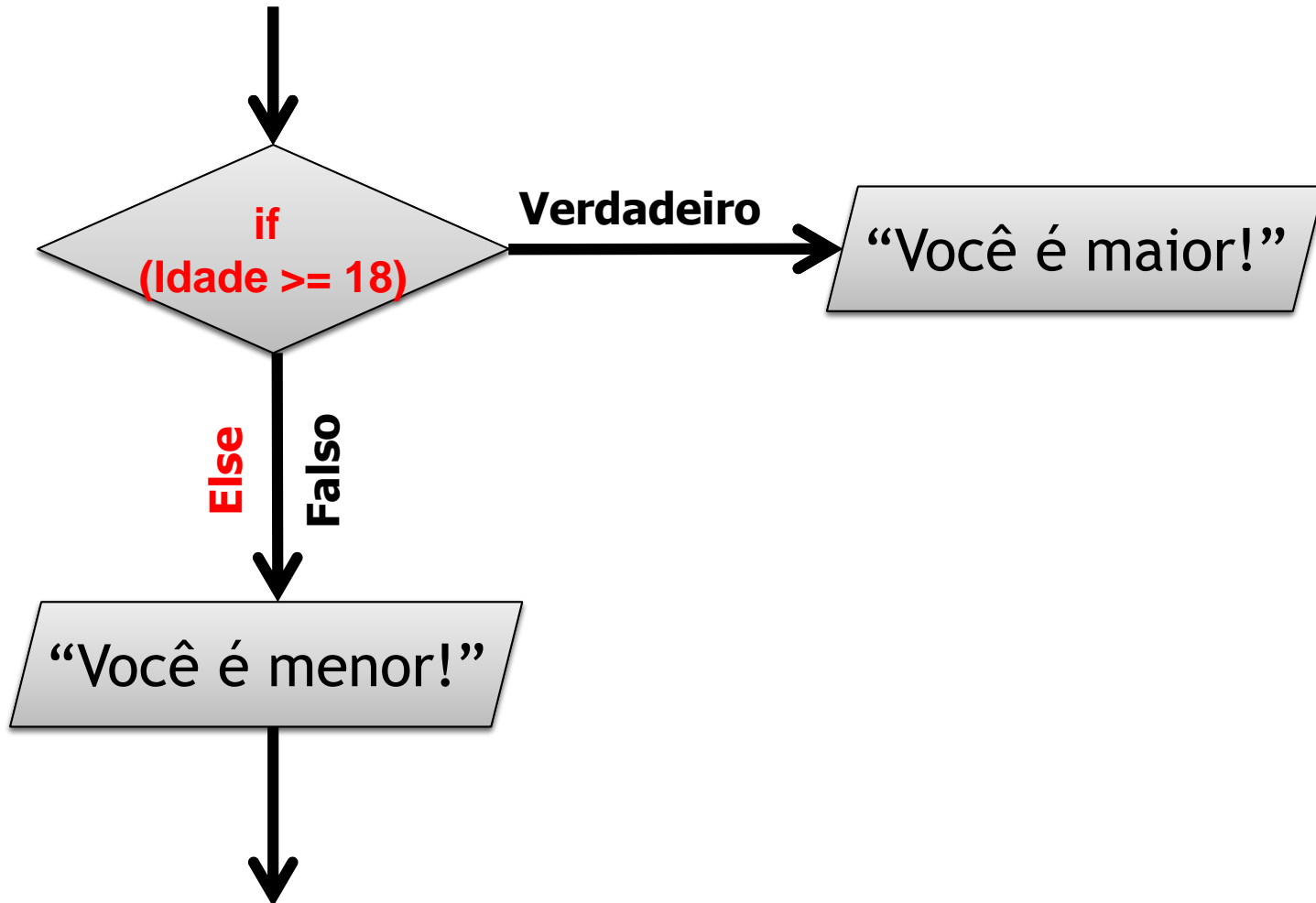
Estrutura de Repetição – Repete um bloco de código durante um laço determinado.

for – while – do...while

Estrutura condicional if

```
2 public class If {  
3     public static void main(String[] args) {  
4         int num = 3;  
5         if (num == 1) {  
6             System.out.println("Hoje é Domingo.");  
7         }  
8         if (num == 2) {  
9             System.out.println("Hoje é Segunda.");  
10        }  
11        if (num == 3) {  
12            System.out.println("Hoje é Terça.");  
13        }  
14        if (num == 4) {  
15            System.out.println("Hoje é Quarta.");  
16        }  
17    }  
18 }
```

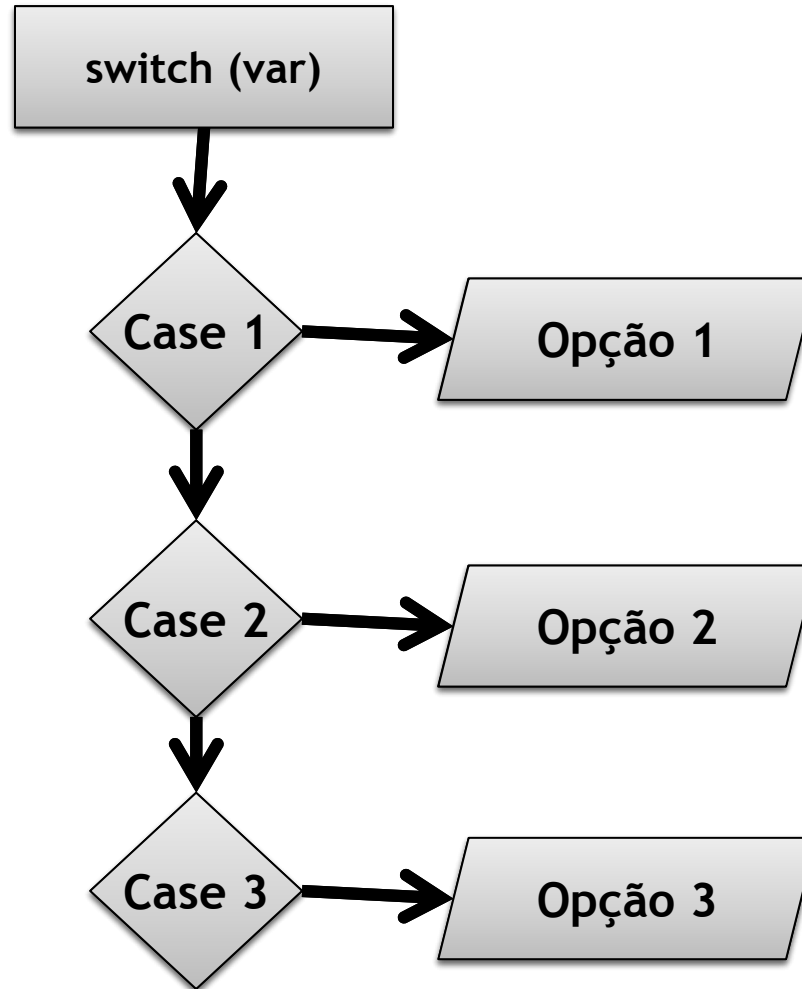
Estrutura Condicional if-else



if-else Exemplo

```
1 public class If_Else {  
2     public static void main(String[] args) {  
3         int idade = 25;  
4         if (idade >= 18) {  
5             System.out.println("Você é maior!");  
6         } else {  
7             System.out.println("Você é menor!");  
8         }  
9     }  
10 }
```

Estrutura Condicional switch-case



switch-case Exemplo

```
1 public class Switch_Case {  
2     public static void main(String[] args) {  
3         int op = 4;  
4  
5         switch (op) {  
6  
7             case 1:  
8                 System.out.println("Você escolheu a opção 1");  
9                 break;  
10            case 2:  
11                System.out.println("Você escolheu a opção 2");  
12                break;  
13            case 3:  
14                System.out.println("Você escolheu a opção 3");  
15                break;  
16            default:  
17                System.out.println("Opção Inválida!");  
18                break;  
19  
20        }  
21    }  
22 }
```