



Linguagem de Programação
JAVA

Profa. Joyce Miranda

Introdução



▶ Tópicos

- ▶ responder o que é Java;
- ▶ Principais características da linguagem;
- ▶ compilar e executar um programa simples.

Introdução

► Breve Histórico

- **1992** - Sun Microsystems (Atual Oracle) - James Gosling
- **Ideia Original**
 - Interpretador multiplataforma para sistemas embarcados
- **Grande Ideia**
 - Adicionar conteúdo dinâmico às páginas da Web.





► Apresentação da Plataforma JAVA

SIGLA	SIGNIFICADO	UTILIZAÇÃO
JSE	<i>Java Standard Edition</i>	É a base da plataforma. Inclui o ambiente de execução e as bibliotecas comuns.
JEE	<i>Java Enterprise Edition</i>	Desenvolvimento de aplicações corporativas e para internet.
JME	<i>Java Micro Edition</i>	Desenvolvimento de aplicações para dispositivos móveis e embarcados.



Introdução

▶ JSE – Java Standard Edition

- ▶ Ambiente completo para o desenvolvimento de aplicações baseadas no modelo cliente-servidor.
- ▶ JSE é distribuída nas formas de:

SDK

Software
Development Kit

JRE

Java Runtime
Environment



Introdução

- ▶ **JSE – Java Standard Edition**
 - ▶ Compilação
 - ▶ Debugging
 - ▶ Geração de documentação (javadoc)
 - ▶ Empacotador de componentes (jar)
 - ▶ JRE, que contém a JVM (Java Virtual Machine).



Introdução

► Resumo de definições

SIGLA	DEFINIÇÃO
JVM = Java Virtual Machine	Identifica o interpretador, esse download não existe.
JRE = Java Runtime Environment	Ambiente de execução Java, formado pela JVM e bibliotecas, tudo que você precisa para executar uma aplicação Java.
JDK = Java Development Kit	(Nós) Desenvolvedores, faremos o download do JDK do Java SE (Standard Edition).

Introdução

- ▶ **Criação de Programas em JAVA**
 - ▶ 1º Etapa – Definição do Código Fonte



Introdução

► Criação de Programas em JAVA

► 2º Etapa – Compilação



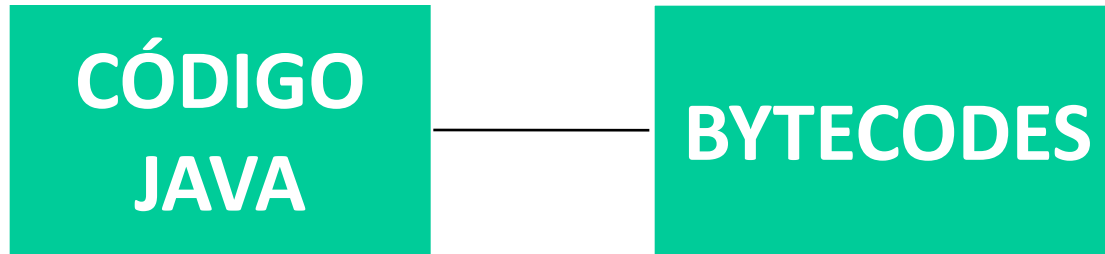
► Compilador: **javac**



Introdução

► Criação de Programas em JAVA

► 2º Etapa – Compilação



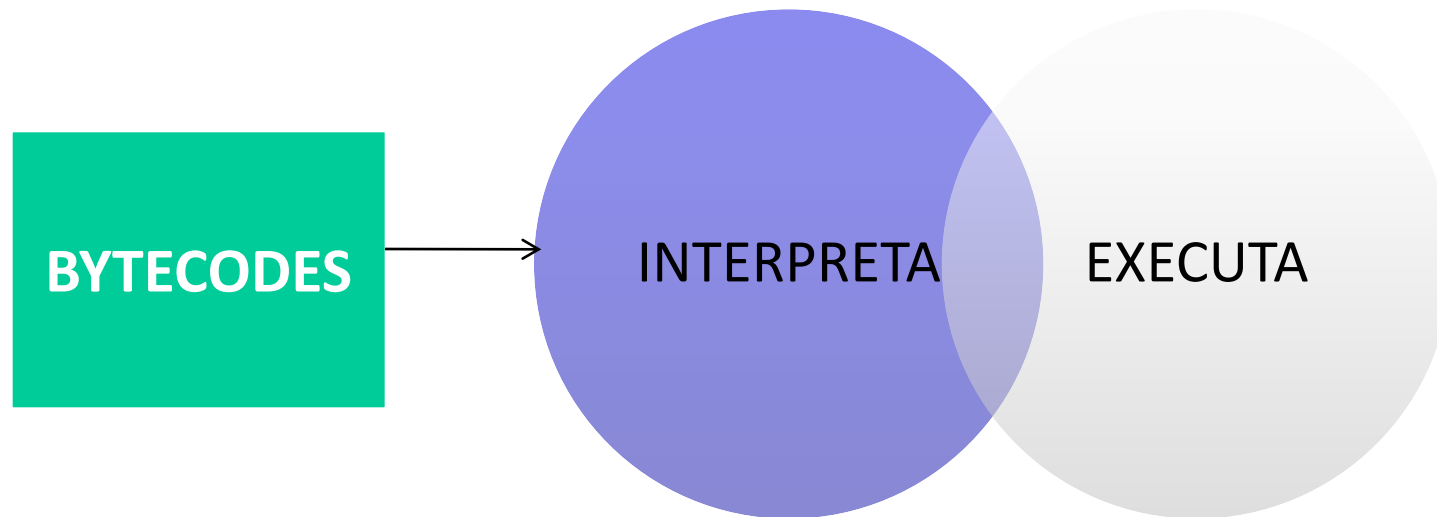
- *São os bytecodes que são independentes de plataforma.*

Introdução

► Criação de Programas em JAVA

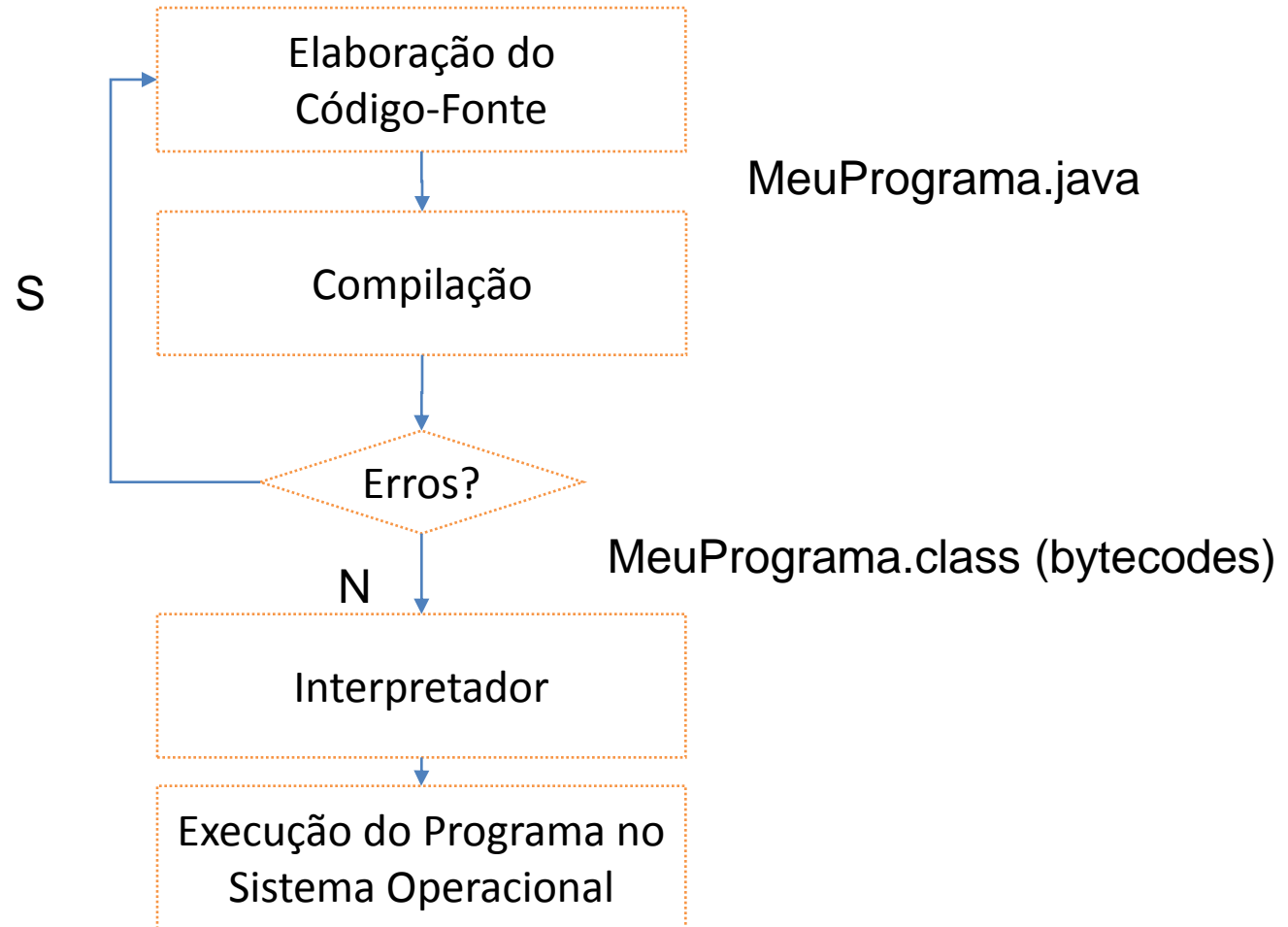
► 3º Etapa - Execução

► Interpretador





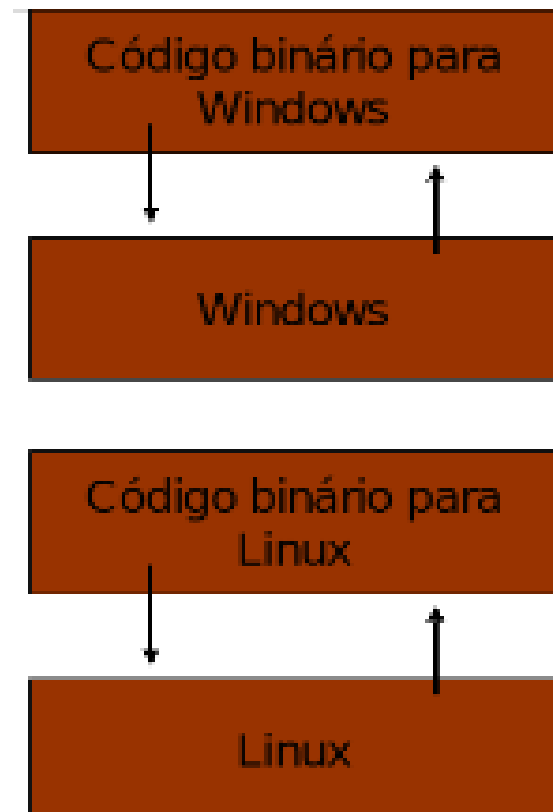
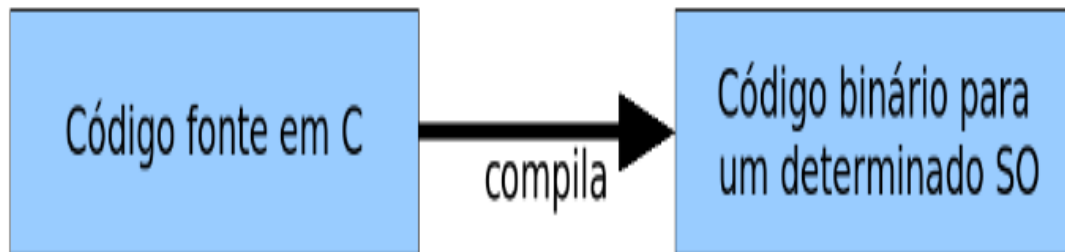
► Criação de Programas em JAVA



Introdução

► Linguagens Compiladas

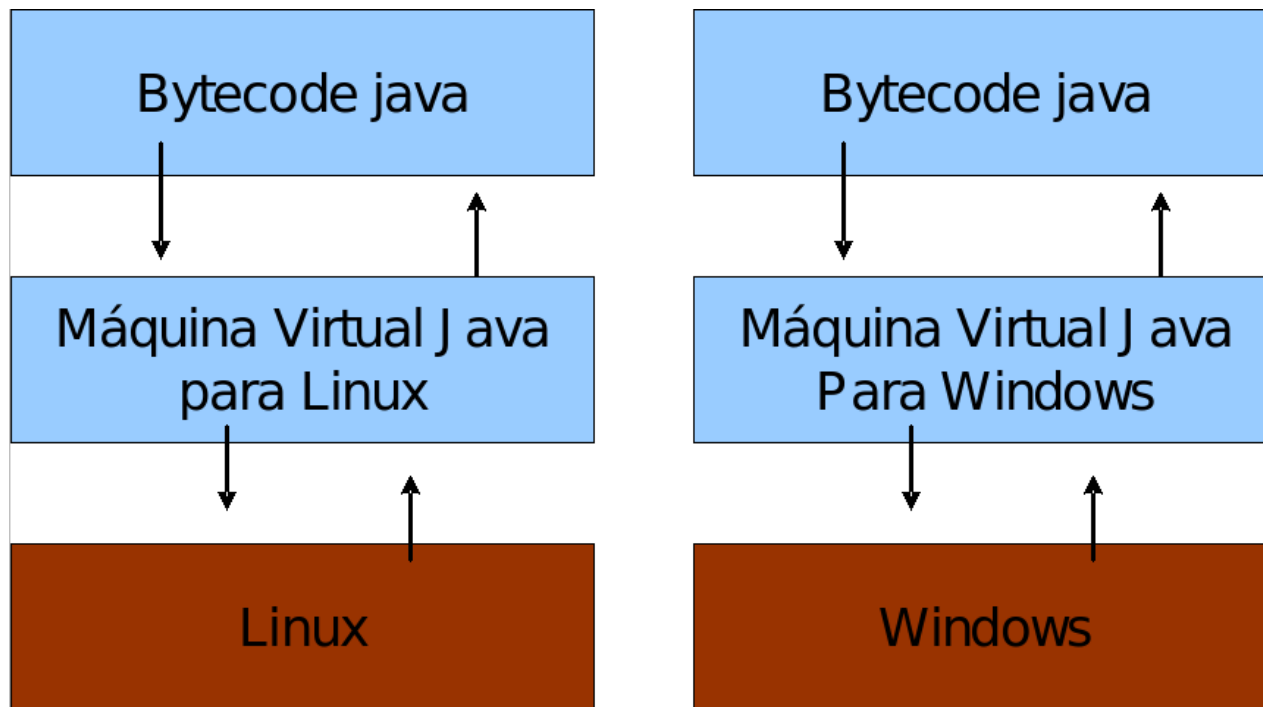
- O código resultante será executado pelo SO.



Introdução

► JAVA - Máquina Virtual

Write once, run anywhere..





```
import java.awt.*;
import java.awt.event.*;
class Party {
    public void buildInvite() {
        Frame f = new Frame();
        Label l = new
Label("Party at Tim's");
        Button b = new
Button("You bet");
        Button c = new
Button("Shoot me");
        Panel p = new Panel();
        p.add(l);
    } // mais código aqui...
}
```

Código-fonte

1

Digite seu código-fonte.
Salve como: **Party.java**

```
File Edit Window Help Plead
%javac Party.java
```

Compilador

2

Compile o arquivo **Party.java** executando o `javac` (o aplicativo do compilador). Se não houver erros, você terá um segundo documento chamado **Party.class**. O arquivo `Party.class` gerado pelo compilador é composto de **bytecodes**.

```
Method Party()
0 aload_0
1 invokespecial #1 <Method
java.lang.Object()>
4 return
Method void buildInvite()
0 new #2 <Class java.awt.Frame>
3 dup
4 invokespecial #3 <Method
java.awt.Frame()>
```

Saída (código)

3

Código compilado:
Party.class

```
File Edit Window Help Swear
%java Party
```

Máquinas virtuais

4

Execute o programa iniciando a Java Virtual Machine (JVM) com o arquivo **Party.class**. A JVM converterá o **bytecode** em algo que a plataforma subjacente entenda e executará seu programa.



Introdução

- ▶ Vamos para o nosso primeiro código!

```
1 class MeuPrograma {  
2     public static void main(String[] args) {  
3         System.out.println("Minha primeira aplicação Java!!");  
4     }  
5 }
```

- ▶ **Salve-o como MeuPrograma.java em algum diretório.**



Introdução

▶ Vamos para o nosso primeiro código!

▶ Compilando

```
"C:\Program Files\Java\jdk1.8.0_25\bin\javac" MeuPrograma.java
```

▶ Executando

```
java MeuPrograma
```



Introdução

▶ Exercícios

▶ Modificando MeuPrograma

- ▶ **1) Altere seu programa para imprimir uma mensagem diferente.**
- ▶ **2) Altere seu programa para imprimir duas mensagens distintas.**
- ▶ **3) Sabendo que os caracteres `\n` representam uma quebra de linha, imprima duas linhas de texto usando uma única linha de código `System.out.println`**
- ▶ **4) Altere o nome da classe no seu arquivo, compile e execute.**