

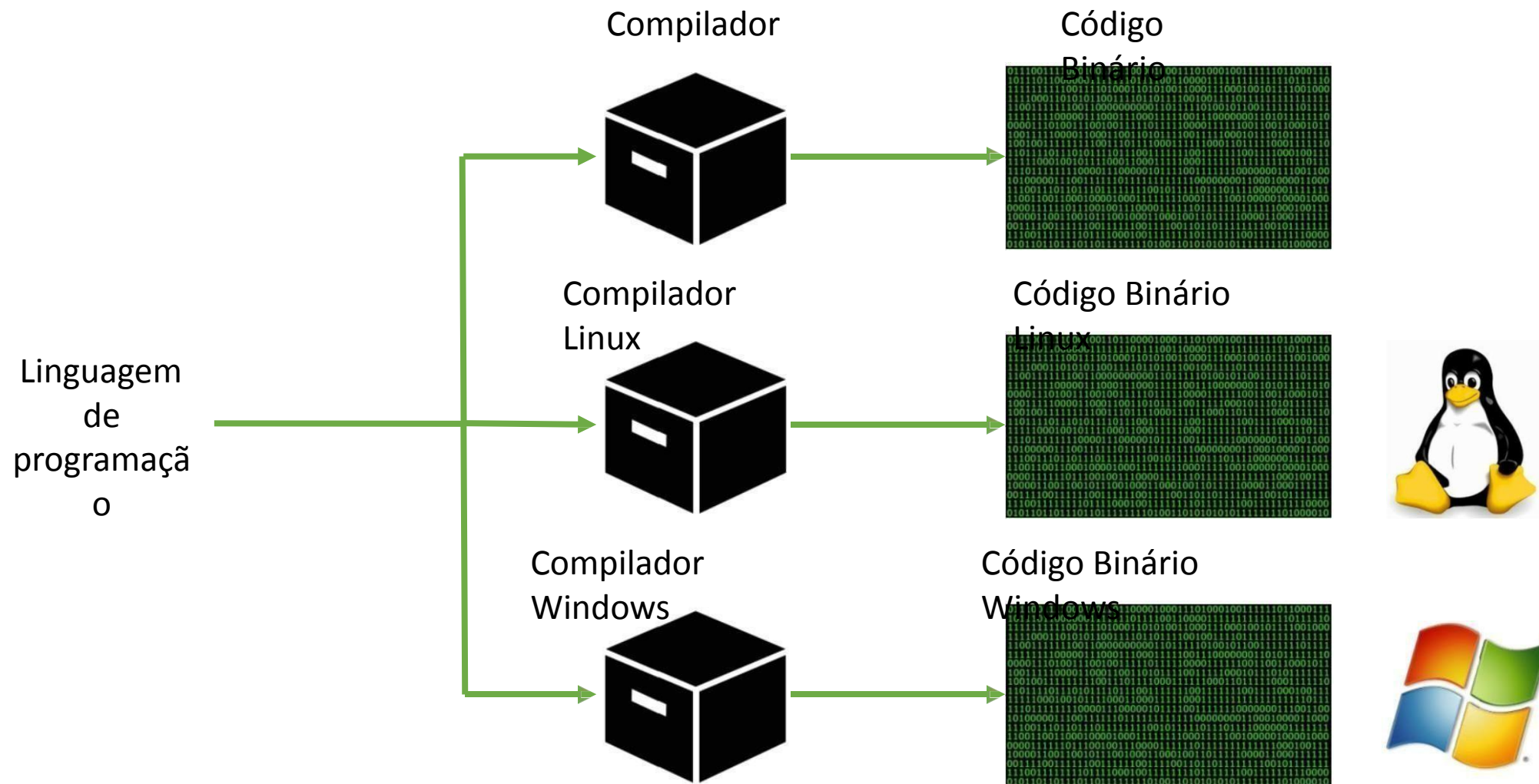
# Residência de Software 2024.2

# Aula 1

# História do Java

- Criada pela antiga Sun Microsystems no início da década de 1990 e comprada pela Oracle em 2009;
- Idealizado como um interpretador para pequenos dispositivos(vídeocassete, televisão, TV a cabo)
  - Falhou após tentativa de parcerias com fabricantes.
- Sun percebeu que poderia rodar pequenas aplicações dentro do browser
  - Lançamento em 1994 como Applets (pequeno software que executa uma atividade específica);
  - Grande quantidade de sistemas operacionais e browsers;
  - Programar numa única linguagem, independente da plataforma;
  - Fazer o browser realizar operações avançadas, e não apenas renderizar html.
- Java nasceu com um objetivo em mente (interpretador);
- Foi lançado com outro (pequenas aplicações);
- Decolou mesmo no desenvolvimento de aplicações do lado do servidor!

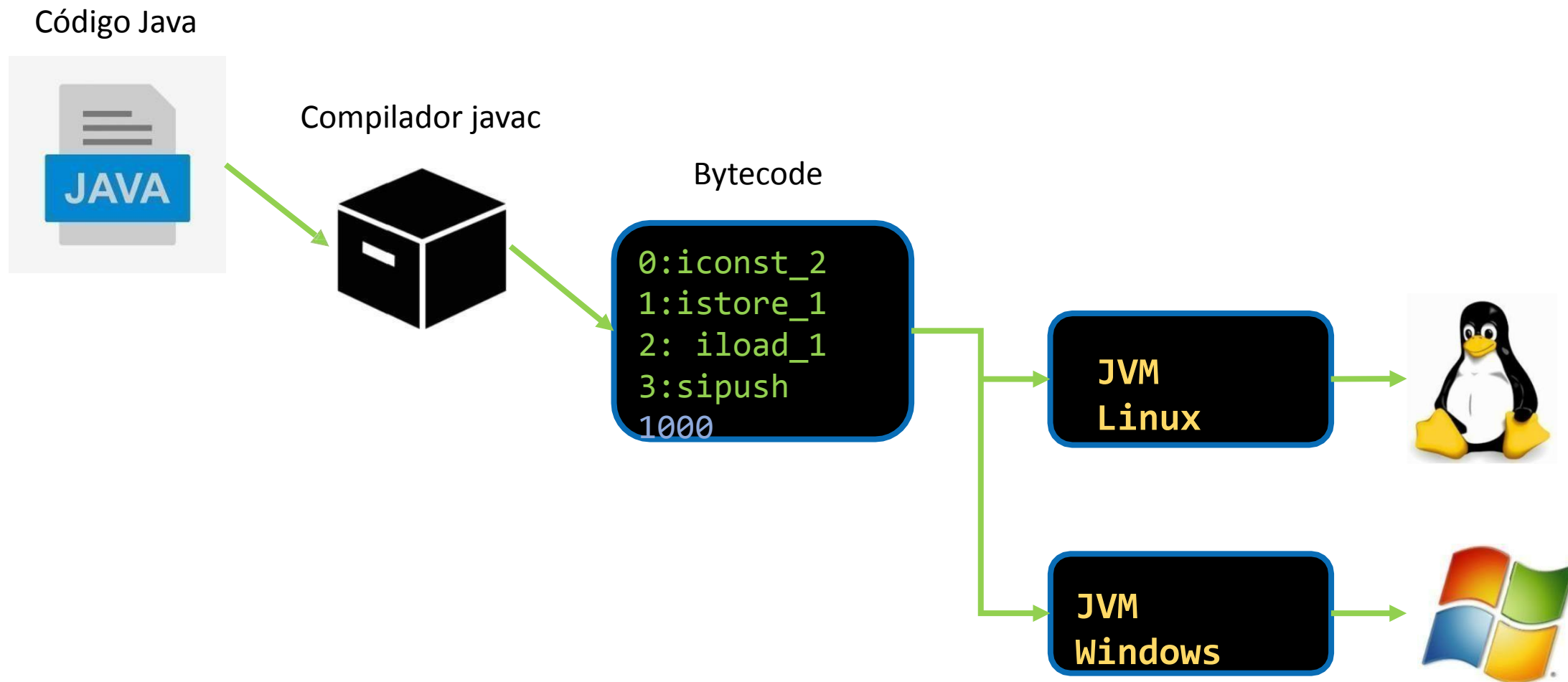
# Compilando uma



# Maquina Virtual Java

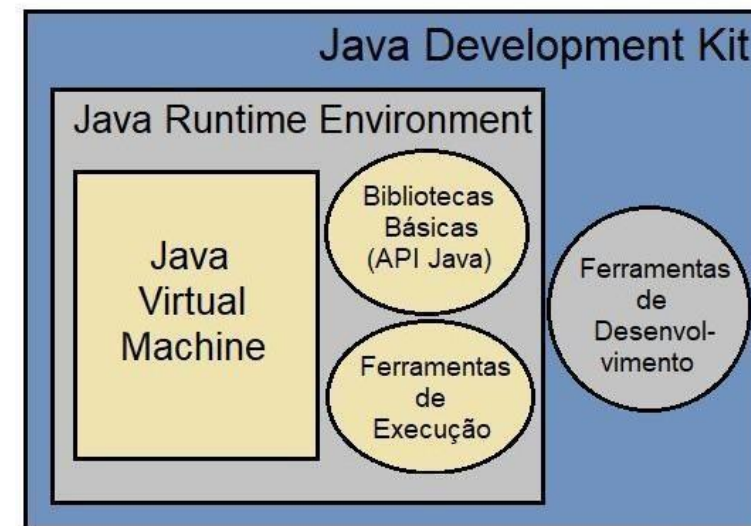
- Introduz uma camada entre o Sistema Operacional(SO) e a Aplicação;
- Isola totalmente a Aplicação do SO;
- O compilador *javac* transforma a linguagem de programação em Bytecode;
- A JVM interpreta o Bytecode;
- O mesmo código roda em diversos SOs;
- A JVM não entende código java, apenas o Bytecode (código intermediário entre o código fonte e o código de máquina);
- Mais do que um interpretador, a máquina virtual gerencia memória, threads, a pilha de execução, etc.

# Maquina Virtual Java (JVM) - Esquema



# JVM, JRE e JDK?

- **JVM** - Executa o bytecode Java. Esse download não existe, ela sempre vem acompanhada.
- **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: ambiente de desenvolvimento Java, o JDK Java SE (Standard Edition) é formado pela JRE somado a ferramentas de desenvolvimento, como o compilador javac.



<https://dicasdejava.com.br/>



# JVM, JRE e JDK?

## JDK

Development Tools : java, javac, javadoc, jar

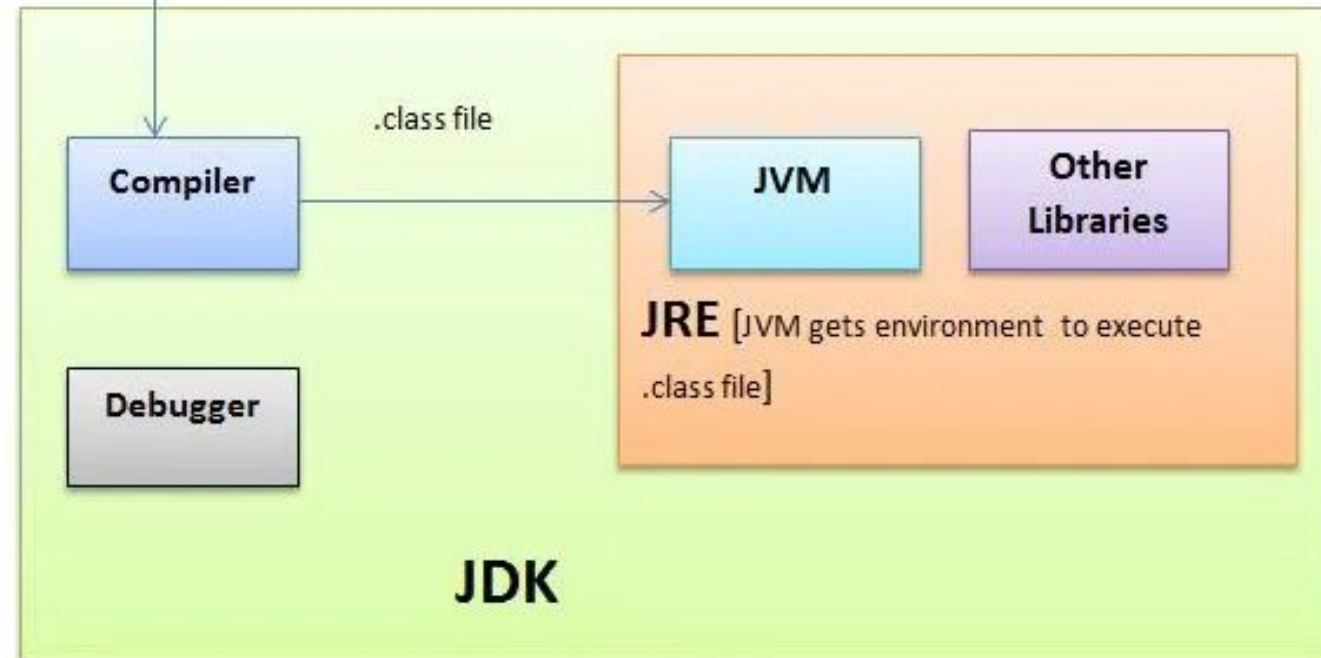
## JRE

Runtime Libraries, Byte Code Verifier, Libraries

## JVM

Java Interpreter, JIT, Garbage Collector

Source files (.java files)





# Ferramentas de trabalho

1. Java JDK versão 17 - <https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>
2. Eclipse - <https://www.eclipse.org/downloads/>  
\*Pode ser necessário instalar o JDK e executar os passos de configuração das variáveis de ambiente antes de iniciar o Eclipse.

# IDE Eclipse

## Porque usar uma IDE?

- IDE, ou ambiente de desenvolvimento integrado, é um software que combina ferramentas comuns de desenvolvimento em uma única interface gráfica do usuário (GUI), facilitando o desenvolvimento de aplicações;
- A **IDE Eclipse** é uma plataforma de software amplamente utilizada para desenvolvimento de aplicações em várias linguagens de programação, sendo mais conhecida no mundo do desenvolvimento Java. Além disso, possui seu código livre.

# Instalação das ferramentas



# “Olá Mundo” - Portugol

Lembra como imprimir um texto simples no Portugol Studio?

```
programa
```

```
{
```

```
    funcao inicio ()
```

```
    {
```

```
        escreva("Olá  
Mundo!!")
```

```
    }
```

```
}
```

# “Olá Mundo” -

No Java, a estrutura não é tão diferente assim:

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

# Novidade!?

## O que iremos aprender:

- Variáveis Primitivas e Controle de Fluxo
  - Declarando e usando Variáveis
  - Tipos Primitivos e Valores
  - Casting (conversores)
  - If e Else (se, senão)
  - While ( enquanto )
  - O For ( para... até... faça )
  - Escopo de variáveis

# Declarando e usando variáveis

No Portugal, tínhamos os seguintes tipos de variáveis:

- inteiro: Número inteiros -> 1 ; 2 ; 3
- real: Números de ponto flutuante -> 1.1 ; 3.14 ; 10.3
- cadeia: Cadeia de caracteres -> “Adoro estudar programação!”
- character: Apenas um caractere -> ‘A’
- logico: Caractere booleano: verdadeiro, falso



# Declarando e usando variáveis

No java apenas a notação para cada tipo de variável muda:

- int, **Integer**, **Long** ou long: Número inteiros -> 1 ; 2 ; 3
- float, **Float**, double, **Double** ou **BigDecimal**: Números de ponto flutuante -> 1.1 ; 3.14 ; 10.3
- String**: Conjunto de caracteres -> “Adoro estudar programação”
- char: Apenas um caractere -> ‘A’
- boolean, **Boolean**: Caractere booleano : true, false

# Atribuição de valores

A atribuição de valores não é muito diferente do que foi visto em Portugal.

Exemplos:

- `int a = 3; // a recebe o valor 3`
- `int b = 5; // b recebe o valor 5`
- `b = a + b; // b vira 8 e a continua 3`
- `String nome = "Fulano"; // nome recebe Fulano`
- `Double pi = 3,1415; // pi recebe o valor de 3,1415`

# Alguns atalhos do

- **Ctrl + 1** – Aciona o quickfix com sugestões para correção de erros;
- **Ctrl + Espaço** – Completa códigos;
- **Ctrl + 3** – Aciona modo de descoberta de menu. Experimente digitar **Ctrl + 3** e depois, **ggas** e **enter**. Ou, então, dê **Ctrl + 3** e digite new class.
- **Ctrl + F11** – Roda a última classe que você executou. É o mesmo que clicar no ícone verde que parece um botão de play, localizado na barra de ferramentas.
- **Ctrl + PgUp** e **Ctrl + PgDown** – Navegam nas abas abertas. Úteis quando estiver editando vários arquivos ao mesmo tempo.
- **Ctrl + /** – Comenta/Descomenta o bloco ou linha selecionado.
- **Ctrl + Shift + F** – Formata o código segundo as convenções do Java.
- **Ctrl + Shift + O** – Adiciona/Remove importações.
- **Ctrl + M** – Expande a View atual para a tela toda (mesmo efeito de dar dois cliques no título da View).
- **Ctrl + Shift + L** – Exibe todos os atalhos possíveis.
- **Ctrl + O** – Exibe um outline para rápida navegação.
- **Alt + Shift + X e depois J** – Roda o da classe atual. Péssimo para pressionar! Mais fácil você digitar **Ctrl + 3** e depois, Run!

Vamos praticar!!

# Exercício 1

- No Eclipse execute o código do “Olá Mundo” em java;
- Altere seu programa para imprimir duas linhas de código usando o System.out (Teste os caracteres dentro do syso: `\t` - `\n` - `\\` - `\'` - `\"`)
- Escreva um programa que receba do teclado seu nome e seu sobrenome separadamente e escreva na tela seu nome completo, como no exemplo abaixo:

Exemplo:

Entrada:

Digite seu nome: Fulano

Digite seu sobrenome: de Tal

Saída:

Meu nome é Fulano de Tal!

# Exercício 2

- Escreva um programa que receba do teclado dois números separadamente e escreva na tela a soma deles, como no exemplo abaixo:

Exemplo:

Entrada:

Digite o primeiro número: 5

Digite o segundo número: 10

Saída:

A soma dos números é igual a 15.