

Entendendo a linguagem de programação Java



JAVA?



- A Sun criou um time para desenvolver inovações tecnológicas em 1992.
- Esse time foi liderado por James Gosling.
- O time voltou com a ideia de criar um interpretador para pequenos dispositivos, como vídeo cassete, televisão e aparelhos de TV a cabo.

- Até então o nome adotado era Oak, que quer dizer *carvalho*, uma árvore que ele podia observar quando olhava através da sua janela.
- Então em 1995 foi lançada uma nova versão onde foi rebatizada para Java.
- Com o anuncio de que a IBM iria dar suporte ao Java, cresceu o número de desenvolvedores.

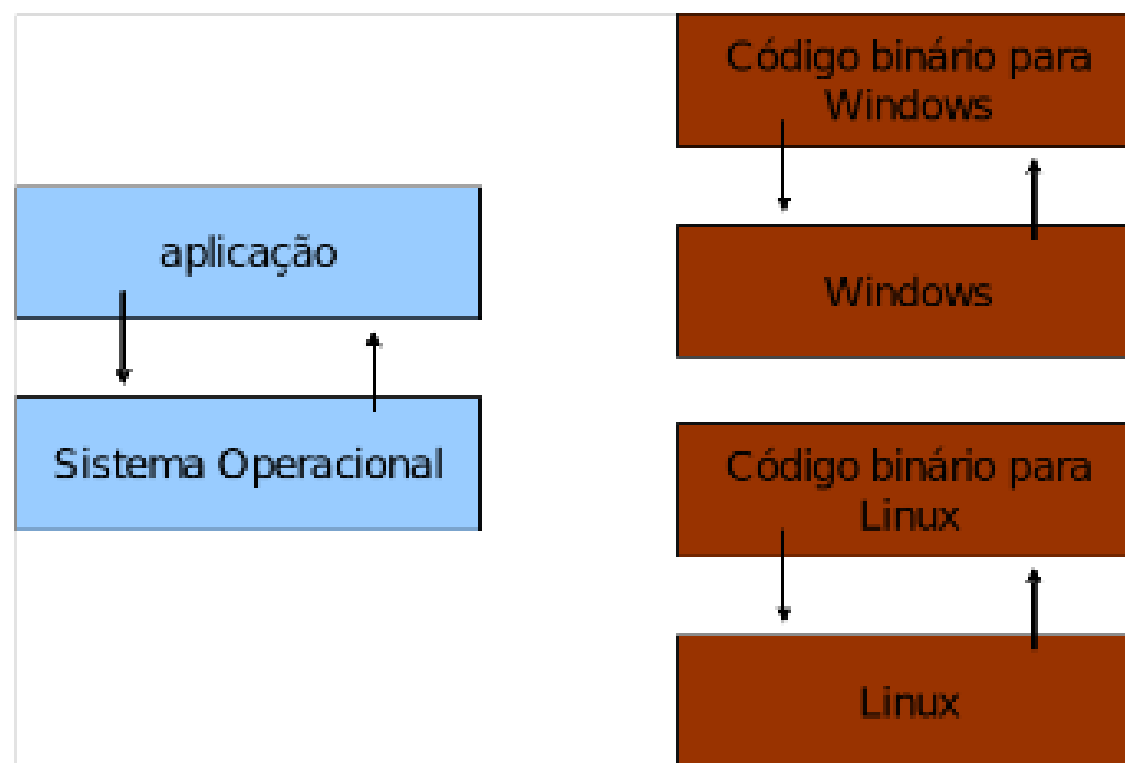
- Ok, mas, qual a diferença do JAVA para as demais linguagens de programação?



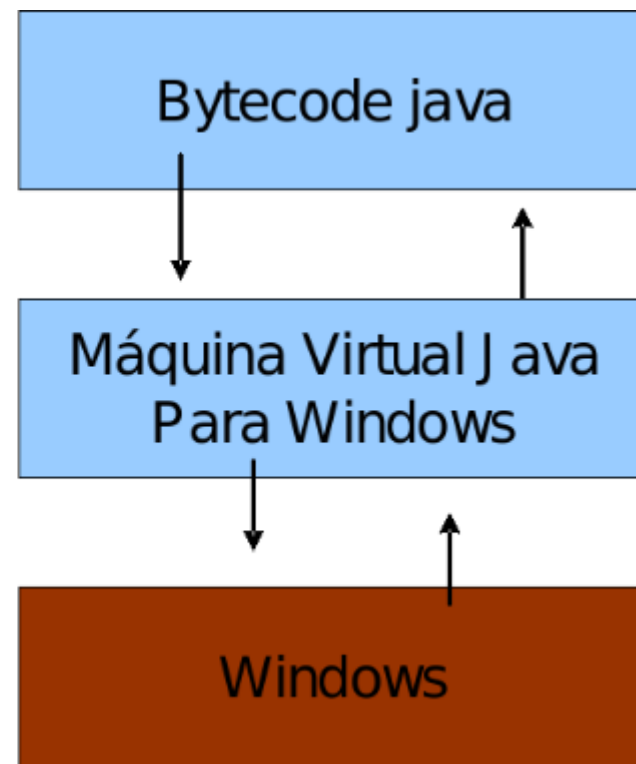
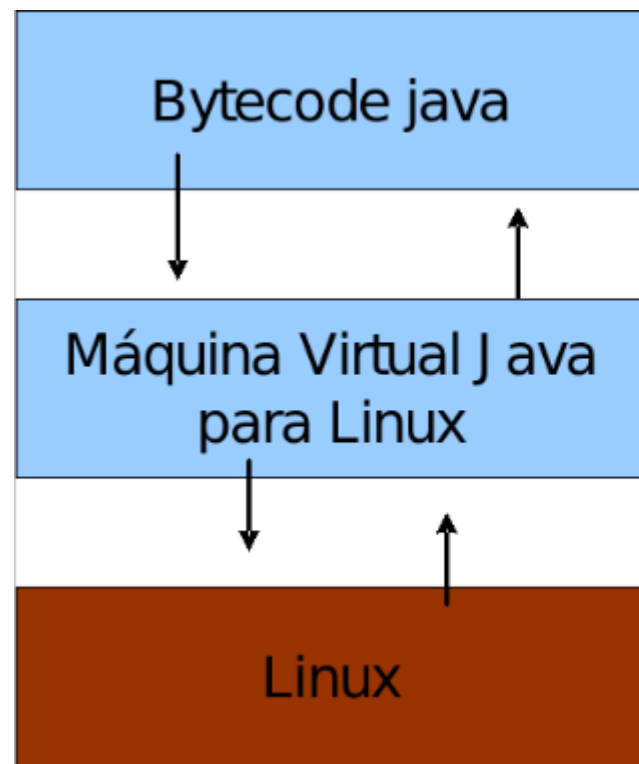
- Nas demais linguagens de programação, temos a seguinte situação quando vamos compilar um programa:



- O código fonte é compilado para código de máquina específico de uma plataforma e sistema operacional.
- Esse código executável (binário) resultante será executado pelo sistema.

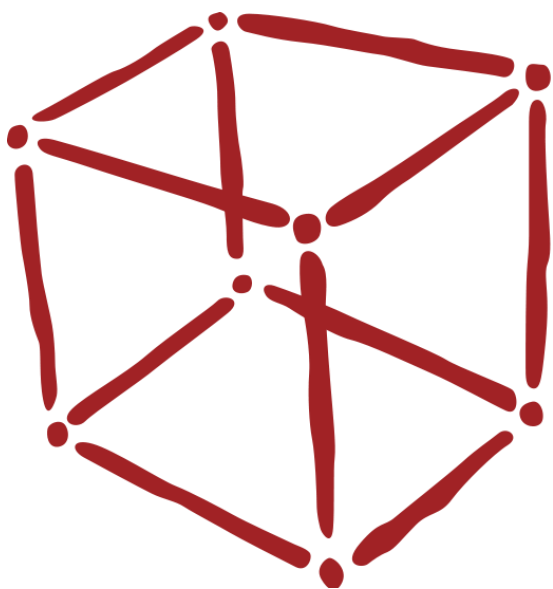


- Já o Java utiliza do conceito de máquina virtual, onde existe, entre o *sistema operacional* e a *aplicação*, uma *camada extra responsável por "traduzir"*.



Ok, agora vamos colocar a mão na massa?





NetBeans

IDE, do inglês *Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado.

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

- **public** = Refere-se a visibilidade desta classe. Quando dizemos que uma classe é de visibilidade “public” , estamos dizendo que esta classe poderá ser acessada por outras classes.
- **class** = Mostramos que estamos criando uma classe.
- **OlaMundo** = Refere-se ao nome da classe que estamos criando. Nesse caso , o nome da minha classe será “OlaMundo”.

- **{ }** = As chaves indicam o início e fim da classe ou método. O código que queremos inserir nesta classe deverá ser escrito dentro do espaço das chaves.
- **static** = Nos garante que somente haverá uma, e não mais que uma, referência para nosso método main, ou seja, todas as instâncias da classe irão compartilhar a mesma cópia do método main.

- **void** = Assim como um método comum, refere-se ao tipo de retorno que esse método terá. Nesse caso, como o tipo de retorno deve ser “void”, ou seja , “vazio”, esse método não retornará valor nenhum.
- **(String[] args)** = Refere-se aos argumentos que serão passados para esse método, sendo obrigatório no caso do método main.

- O objeto **System.out** é a saída padrão, que permite exibir as Strings no console (terminal) de comando quando o aplicativo de Java é executado. Dentro desse objeto existem métodos para gerar saídas de Strings, entre elas são: **println**, **print** e o **printf**.

- Para cálculos, podemos usar os operadores abaixo:

- $+$ => soma

- $-$ => subtração

- $*$ => multiplicação

- $/$ => divisão

- Ordem de precedência:
- Operadores de multiplicação, divisão e *módulo(mod)* são aplicadas primeiro. Por exemplo, quando aparecer uma expressão com várias dessas operações, elas serão aplicadas da esquerda para a direita.
- As operações de adição e subtração são aplicadas em seguida.
- ‘*’, ‘/’, ‘%’, ... ‘+’, ‘-’

/*Com base no que vimos até aqui,
vamos exercitar...*/

Declaração e atribuição de variáveis

- Faça um programa que calcule a idade de uma pessoa.
 1. Cândida nasceu em 1995.
 2. Pedro em 1945.

```
public class CalIda {  
    public static void main(String[] args) {  
        int nasc = 1995, ano = 2017;  
        int ida;  
        ida = ano - nasc;  
        System.out.println(ida);  
    }  
}
```

- Faça um programa que calcule a média de João.

1º Un => 8

2º Un => 6

3º Un => 4

4º Un => 10

```
public class CalMedia {  
    public static void main(String[] args) {  
        int un1 = 8, un2 = 6, un3 = 4, un4 = 10;  
        int med;  
        med = (un1+un2+un3+un4)/4;  
        System.out.println("med");  
    }  
}
```


1) Na empresa onde trabalhamos, há tabelas com o quanto foi gasto em cada mês. Para fechar o balanço do primeiro trimestre, precisamos somar o gasto total. Sabendo que, em Janeiro, foram gastos 15000 reais, em Fevereiro, 23000 reais e em Março, 17000 reais, faça um programa que calcule e imprima o gasto total no trimestre. Siga esses passos:

- a) Crie uma classe chamada `BalancoTrimestral` com um bloco `main`, como nos exemplos anteriores;
- b) Dentro do `main` (o miolo do programa), declare uma variável inteira chamada `gastosJaneiro` e inicialize-a com 15000;
- c) Crie também as variáveis `gastosFevereiro` e `gastosMarco`, inicializando-as com 23000 e 17000, respectivamente, utilize uma linha para cada declaração;
- d) Crie uma variável chamada `gastosTrimestre` e inicialize-a com a soma das outras 3 variáveis:

```
int gastosTrimestre = gastosJaneiro + gastosFevereiro + gastosMarco;
```

e) Imprima a variável `gastosTrimestre`.

2) Adicione código (sem alterar as linhas que já existem) na classe anterior para imprimir a média mensal de gasto, criando uma variável `mediaMensal` junto com uma mensagem. Para isso, concatene a String com o valor, usando "Valor da média mensal = "+ `mediaMensal`.

```
public class GastosTrimestre {  
    public static void main(String[] args) {  
        int gastosJaneiro = 15000, gastosFevereiro = 23000,  
gastosMarco = 17000;  
        int gastosTrimestre = gastosJaneiro + gastosFevereiro +  
gastosMarco;  
        System.out.println(gastosTrimestre);  
    }  
}
```

Vamos criar uma nova classe Java cujo objetivo será calcular o valor total de livros do nosso estoque na livraria. Podemos chamá-la de *CalculadoraDeEstoque*.

```
public class CalculadoraDeEstoque{  
    public static void main(String[] args) {  
        double livroJava8;  
        double livroTDD;  
    }  
}
```

```
double livroJava8;
```

```
double livroTDD;
```

```
livroJava8 = 59.90;
```

```
livroTDD = 59.90;
```

```
double soma = livroJava8 + livroTDD;
```

- Pronto, isso é tudo que precisamos por enquanto para completar a nossa CalculadoraDeEstoque. Após somar esses valores, vamos imprimir o valor do resultado no *console*:

Olá mundo!

```
public class CalculadoraDeEstoque{  
    public static void main(String[] args) {  
        double livroJava8 = 59.90;  
        double livroTDD = 59.90;  
        double soma = livroJava8 + livroTDD;  
        System.out.println("O total em estoque é: " + soma);  
    }  
}
```

Adicionando condicionais

- Nossa CalculadoraDeEstoque precisa de uma nova funcionalidade. Se o valor total de livros for menor que 150 reais, precisamos ser alertados de que nosso estoque está baixíssimo, caso contrário, devemos mostrar uma mensagem indicando de que está tudo sob controle!
- Em Java, podemos fazer essa condicional de uma forma bem comum, utilizando um if e else. Observe:


```
public class CalculadoraDeEstoque{
    public static void main(String[] args) {
        double livroJava8 = 59.90;
        double livroTDD = 59.90;
        double soma = livroJava8 + livroTDD;
        System.out.println("O total em estoque é " + soma);
        if (soma<150) {
            System.out.println("Seu estoque está muito baixo!");
        } else {
            System.out.println("Seu estoque está bom!");
        }
    }
}
```

```
public class CalculadoraDeEstoque {  
    public static void main(String[] args){  
        double livroJava8 = 59.90;  
        double livroTDD = 59.90;  
        double soma = livroJava8 + livroTDD;  
        System.out.println("O total em estoque é: " + soma);  
        if(soma<150){  
            System.out.println("Seu estoque está baixo!");  
        } else if(soma>=2000){  
            System.out.println("Seu estoque está muito alto!");  
        } else {  
            System.out.println("Seu estoque está bom!");  
        }  
    }  
}
```

```

public class CalculadoraDeEstoque {
    public static void main(String[] args) {
        double soma = 0;
        int contador = 0;
        while (contador < 35){
            double valorDoLivro = 59.90;
            soma = soma + valorDoLivro;
            contador = contador + 1; Ou (++)
        }
    }
}

```

```

System.out.println("O total em estoque é " +
soma);
        if (soma<150) {
            System.out.println("Seu estoque
está muito baixo!");
        } else if(soma >= 2000){
            System.out.println("Seu estoque
está muito alto!");
        } else {
            System.out.println("Seu estoque
está bom!");
        }
    }
}
}

```

Solicitando dados do usuário

Biblioteca

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

- `Scanner Scan = new Scanner (System.in);`

Solicitando dados do usuário

```
public class SolicitaNum{  
    public static void main(String[]args){  
        Scanner dg = new Scanner (System.in);  
        System.out.println("Digite um número");  
        int num = dg.nextInt();  
        System.out.println("O número digitado foi: " + num);  
    }  
}
```

- Crie um programa que informe quando o número for PAR ou ÍMPAR e exiba a mensagem.

```
public class SolicitaNum{  
    public static void main(String[]args){  
        Scanner dg = new Scanner (System.in);  
        System.out.println("Digite um número");  
        int num = dg.nextInt();  
        System.out.println("O número digitado foi: " + num);  
        if(num % 2 == 0){  
            System.out.println("E esse número é PAR");  
        } else {  
            System.out.println("E esse número é ÍMPAR");  
        }  
    }  
}
```

Materiais para estudar Java

- Sites:

- <http://www.devmedia.com.br/cursos/>
- <https://www.cursoemvideo.com/>
- <http://www.brasilmaisti.com.br/index.php/pt-br/>
- <https://www.ev.org.br/Paginas/Home.aspx>

- Livros

- Use a Cabeça! Java
- Java para iniciantes
- Java Como Programar
- Desbravando Java e Orientação a Objetos

Obrigado!

Estudem muuuuito!