PROGRAMAÇÃO II – JAVA
Introdução

Cleyton Caetano de Souza

IFPB – Campus Monteiro

Roteiro

- História
- Características de Java
- Máquina Virtual Java
- Paradigma Orientado a Objetos (POO)
- Nossa primeira classe
- O “Main”
- Sintaxe Básica (overview)

História de Java

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- 1991
  - Green Project – grupo para o desenvolvimento de tecnologias “futurísticas”
  - 7 – controle remoto com interface touchscreen
  - Oak – o pai do Java
- 1995
  - Java – versão atualizada do Oak para internet
  - Desenvolvimento de aplicações corporativas e páginas dinâmicas – recebeu o suporte de grande companhias como a IBM

História

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Desde seu lançamento, em maio de 1995, a plataforma Java foi adotada mais rapidamente do que qualquer outra linguagem de programação na história da computação.
- Atualmente, Java é uma das linguagens de programação mais utilizadas no mundo.
  - Número 1 no Ranque da IEEE
  - Número 2 no Ranque do TIOBE

Características do Java

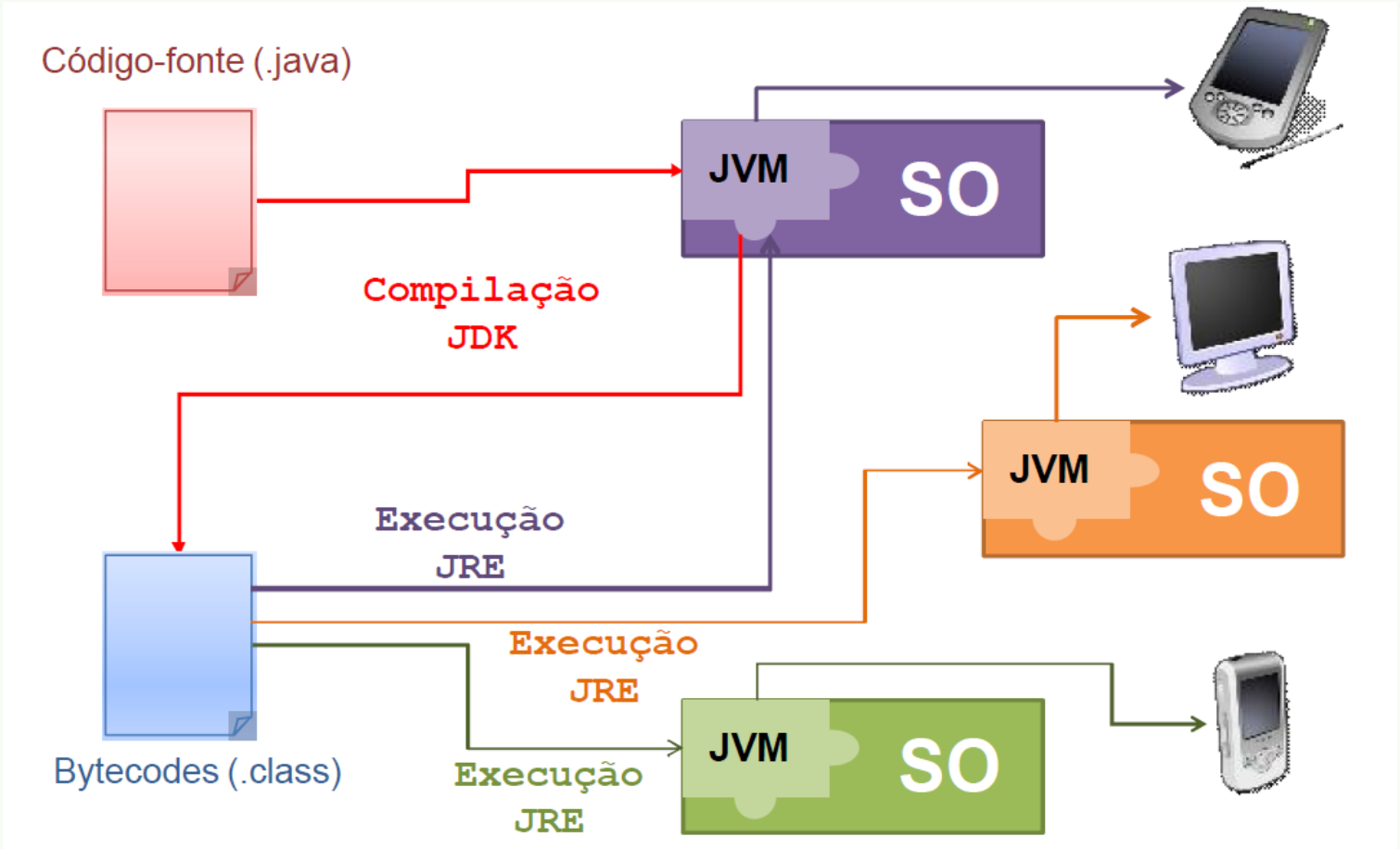
- É free!
- Possui uma vasta biblioteca, além de uma comunidade bastante ativa
- Utiliza o paradigma orientado a objetos
- Robusta (~fortemente tipada)
- Distribuída
- Portável

Portabilidade

- Java é uma linguagem interpretada
  - Não é compilada
  - O que isso significa?

Execução de um Programa em Java

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA



Bytecodes

- É o formato “compilado” dos programas em Java
- Consiste em um código intermediário (ajuda na portabilidade)
- A Máquina Virtual Java (JVM) “interpreta” os bytecodes
  - Qualquer dispositivo com uma JVM é capaz de executar um programa escrito em Java
    - “*Write once, run anywhere*”

Java Virtual Machine

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Máquina virtual Java (do inglês Java Virtual Machine - JVM) é um programa, instalado no SO, que carrega e executa aplicativos escritos em Java pré-compilados (bytecodes)
- A JVM funciona como uma camada intermediária entre o SO e os bytecodes
- Graças à JVM, os programas escritos em Java podem funcionar em qualquer plataforma de hardware e software, que possua uma versão da JVM, tornando essas aplicações independentes de plataforma.

Java Virtual Machine

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARAÍBA

- JVM (Máquina Virtual Java)
  - Executa aplicações Java
- JRE (Ambiente de Execução Java)
  - Bibliotecas (APIs) + JVM
- JDK (Kit de Desenvolvimento Java)
  - Permite criar software em Java

Paradigma Orientado a Objetos

- Paradigma Estruturado
- Paradigma Orientado a objetos
  - É um paradigma de programação que “imita” como os objetos interagem no mundo real;
  - O “mundo real” é organizado através de objetos que se comunicam através da troca de mensagens, fornecendo e/ou consumindo serviços.

Paradigma Orientado a Objetos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

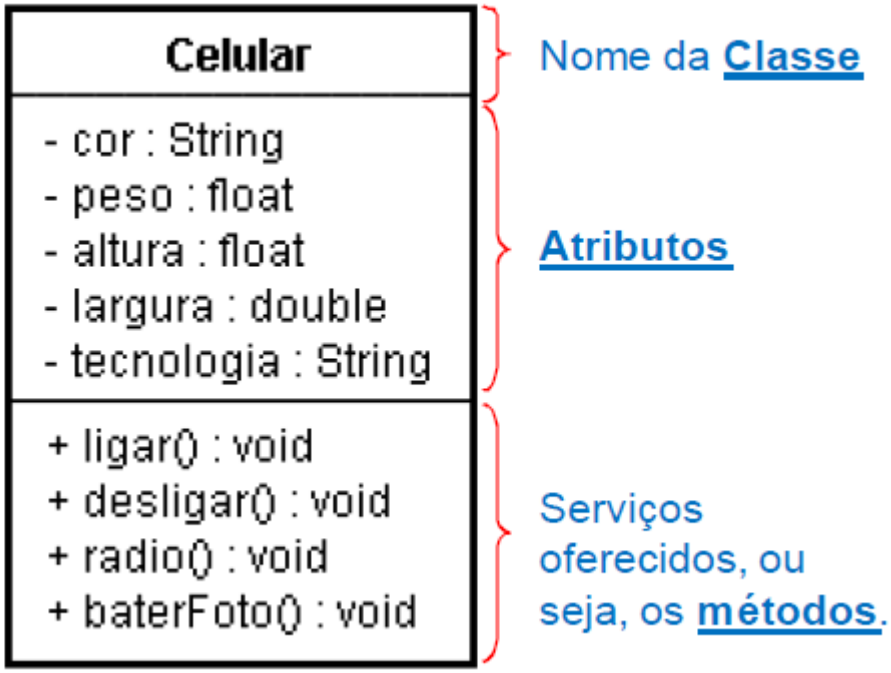
- Analogia
  - Manual de Instruções e Celular

Paradigma Orientado a Objetos

- Analogia
  - Classe (Manual de Instruções)
    - A Classe contém as informações sobre as características do objeto (atributos) e as ações que ele realiza (métodos)
  - Objeto (Celular)
    - Possui características e realiza ações!
- Quais são as características e ações de um celular?

Classe Celular

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA



Objeto Celular

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Celular
- cor : amarelo - peso : 200 - altura : 18,5 - largura : 14,2 - tecnologia : Android
+ ligar() : void + desligar() : void + radio() : void + baterFoto() : void

Nome da **Classe**

Atributos

Serviços oferecidos, ou seja, os **métodos**.

Nossa Primeira Classe

- Vamos Implementar a Classe Celular, na lousa!

Nossa Primeira Classe

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

```
class celular {

    String cor;
    float peso;
    float altura;
    float largura;
    String tecnologia;

    void ligar(){

    }
    void desligar(){

    }
    void radio() {

    }
    void bater_foto() {

    }

}
```

Convenções dos Programadores

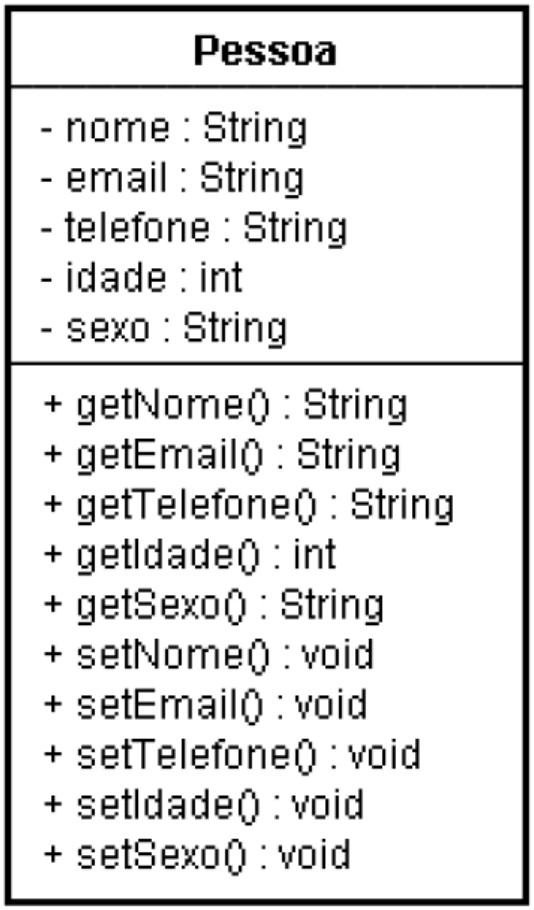
- Nomenclatura das Classes
  - Primeira Letra em Maiúsculo

  Ex: Carro, Celular, Pessoa, Aluno;
- Nomenclatura dos Métodos
  - Camel Case

  Ex: curtir, baterFoto, enviarMensagemBoasVindas;
- Modificadores de Acesso
  - Público, Privado;
- Encapsulamento
  - Gets e Sets

Getters e Setters

- A convenção é (1) **deixar os atributos de uma classe como privados** & (2) **definir métodos públicos para retornar** (get) **e alterar** (set) **o valor dos atributos**
- Há um par de métodos get e set para cada atributo da classe



Nossa Classe “Melhorada”

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

```
public class Celular {

    private String cor;
    private String tecnologia;
    private float peso;
    private float altura;
    private float largura;

    public void ligar(int numero) {

    }

    public int baterFoto() {
        return -1;
    }

    public String getCor() {
        return cor;
    }

    public void setCor(String novaCor) {
        cor = novaCor;
    }

    //demais métodos

}
```

O “main”

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- O main é um método especial que é chamado quando nosso programa em JAVA é executado
- O nosso projeto em Java é composto por várias classes
  - A JVM “procura” o método main para executar o projeto!
- As duas finalidades do Main
  - Testar sua classe
  - Acionar/Iniciar seu aplicativo Java
- Vamos testar nossa classe celular

Ambiente de Desenvolvimento

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Eclipse
  - https://www.eclipse.org/downloads/
- JDK e JRE
  - http://www.oracle.com/technetwork/pt/java/javase/downloads/index.html

Conhecendo o Eclipse

- Visões
- Console
- Janela dos Problemas
- Novo Projeto
- Nova Classe
- Executar
- Modo Debug

Testando nossa classe Celular

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

```
public class Principal {

    public static void main(String[] args) {

        Celular celular = new Celular();

        celular.setCor("Amarelo");

    }

}
```

Classes e Objetos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Uma aplicação Java pode possuir dezenas, centenas ou até milhares de classes
- A classe é a menor unidade de código Java – não é possível escrever um programa em Java sem construir uma classe
- **COROLÁRIO PARA GUIAR O PROJETO DE CLASSES:** uma classe, um objetivo

Objetos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Uma aplicação OO pode ser compreendida como a interação entre objetos de diferentes classes
- Os objetos executam ações (interagem) por meio da chamada de métodos
- **COROLÁRIO PARA GUIAR A IMPLEMENTAÇÃO DE MÉTODOS:** um método, um objetivo
  - Evite criar um método que faça “tudo”. Sempre que possível, distribua sua lógica.

Sintaxe Básica de Java

- A sintaxe de Java é baseada na sintaxe de C
- Java é $case\ sensitive$

Sintaxe Básica de Java

- Delimitadores de bloco e de comando
- Comentários
- Declarando variáveis locais
- Entrada e Saída de dados pelo console (mais simples)
- Tipos primitivos de dados
- Operadores aritméticos
- Estruturas de Controle: condicionais e laços

Sintaxe

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARAÍBA

- { – inicia um novo bloco de comando
- } – fecha um bloco de comando
- ; – o final de todas as instruções contém um ponto e vírgula
- // – comentário de uma linha
- /**/ – comentário de múltiplas linhas

Entrada e Saída (pelo console)

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Entrada

```
Scanner leitor = new Scanner(System.in);
String entrada = leitor.nextLine();
```

- Saída

```
System.out.println("Olá mundo!");
```

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Faça um programa que leia seu nome e imprima a mensagem “Olá ” seguida do seu nome.
- Resposta
  - https://repl.it/DhLQ/1

Tipos Primitivos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Tipo	Tamanho	Faixa de Valores
boolean		true ou false
char	2 bytes	0 a 65.535
byte	Inteiro com 1 byte	-128 a 127
short	Inteiro com 2 bytes	-32.768 a 32.767
int	Inteiro com 4 bytes	-2.147.483.648 a 2.147.483.647
long	Inteiro com 8 bytes	-2^{63} a $2^{63}-1$
float	Ponto flutuante com 4 bytes	$1.40239846e^{-46}$ a $3.40282347e^{+38}$
double	Ponto flutuante com 8 bytes	$4.9406564584124654e^{-324}$ a $1.7976931348623157e^{+308}$

Operadores de Atribuição

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Operador	Símbolo
=	Atribuição normal
+=	Soma
-=	Subtração
*=	Multiplicação
/=	Divisão
%=	Resto

Operadores Aritméticos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Operador	Símbolo	Exemplo	Resultado
Adição	+	10 + 4	14
Subtração	–	10 – 4	6
Multiplicação	*	10 * 4	40
Divisão	/	10 / 4	2 (int) e 2.5 (float)
Resto	%	10 % 4	2
Incremento	++	++x ou x++	-
Decremento	--	--x ou x--	-

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Faça um programa que leia dois números inteiros e exiba a soma, produto, diferença, média, razão e o resto entre eles.
- Resposta
  - https://repl.it/Do1y/1

Comandos Condicionais

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- SE

```
if (condição)
    ação();
```

- Exemplo (ação com uma linha)

```
if (2>3)
    System.out.println("2 é maior que 3");
```

- Exemplo (ação com mais de uma linha)

```
if (x==3) {
    System.out.println("x é igual a 3");
    System.out.println("x é igual a 3");
}
```

Comandos Condicionais

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- SE... SENÂO

```
if (condição)
     ação1();
else
     ação2();
```

- Exemplo

```
if (2>3)
    System.out.println("2 é maior que 3");
else
    System.out.println("2 não é maior que 3");
```

Comandos Condicionais

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- SE... SENÂO SE...

```
if (condição)
    ação1();
else if (outraCondição)
    ação2();
```

- Exemplo

```
if (x>3)
    System.out.println("x é maior que 3");
else if (x==3)
    System.out.println("x é igual 3");
else
    System.out.println("x é menor 3");
```

Operadores Relacionais*

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Operador	Símbolo
>	Maior
>=	Maior igual
<	Menor
<=	Menor igual
==	Igual
!=	Diferente

*só servem para tipos primitivos

Operadores Lógicos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Operador	Símbolo
&&	E
||	Ou
!	Não

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Faça um programa que leia um número e imprima se o número é par ou impar.
  - Resposta: https://repl.it/DhMk/11
- Faça um programa que leia três números, calcule a média entre eles e exiba a mensagem se a média é maior, menor ou igual a 7.
  - Resposta: https://repl.it/Dn1X/3

Comandos Condicionais

- Switch Case
  - Também pode ser utilizado com Char, Strings...

```
int var=0;

switch (var) {
case 0:
        ação1();
    break;
case 1:
        ação2();
    break;
default:
        ação2();
    break;
}
```

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Escreva um programa que leia um dia da semana em formato de número (domingo=0, segunda=1, ...) e exiba uma mensagem “Bom” seguido do nome do dia.
  - Resposta: https://repl.it/Dn2C/1
- Escreva um programa que leia um dia da semana em formato de String e exiba uma mensagem “Bom dia” se o dia for útil (de segunda a sexta) ou “Mal dia” se for não útil (sábado e domingo).
  - Resposta: https://repl.it/Dn2O/1

Comandos de Repetição

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Enquanto

```
while (condição)
    ação();
```

- Faça... Enquanto

```
do
    ação();
while (condição);
```

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Escreva um programa utilizando ***while*** que calcule o fatorial de um número.
  - Resposta https://repl.it/DzVz/1
- Escreva um programa com ***do while*** que leia inteiros até que seja lido o número -1 e ao final informe o total de números lidos.
  - Resposta https://repl.it/DzWo/1

Comandos de Repetição

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Java possui dois tipos de laços do tipo For
  - FOR “normal”

```
for(int contador=0;condição;contador++)
    ação();
```

  - Exemplo

```
for(int i = 0 ; i <= 10; i++) {
    System.out.println(i);
}
```

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Escreva um programa, utilizando for, que calcule o fatorial de um número.
  - Resposta https://repl.it/DzWZ/latest
- Escreva um programa, utilizando for, que leia 10 números e calcule a média deles e o maior deles.
  - Resposta https://repl.it/DzWK/2

Comandos de Repetição

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Java possui dois tipos de laços do tipo For
  - FOR “each” – serve para percorrer um conjunto de elementos

```
for(int elemento: coleção)
    ação();
```

Arrays em Java

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Um Array é um tipo de estrutura de tamanho fixo para armazenar dados
  - Também chamado de matriz ou vetor
- Em Java, arrays são objetos, por isso precisam ser criados com o operador **new**
  - Como o tamanho do Array é fixo, é preciso definir em sua criação qual seu tamanho máximo

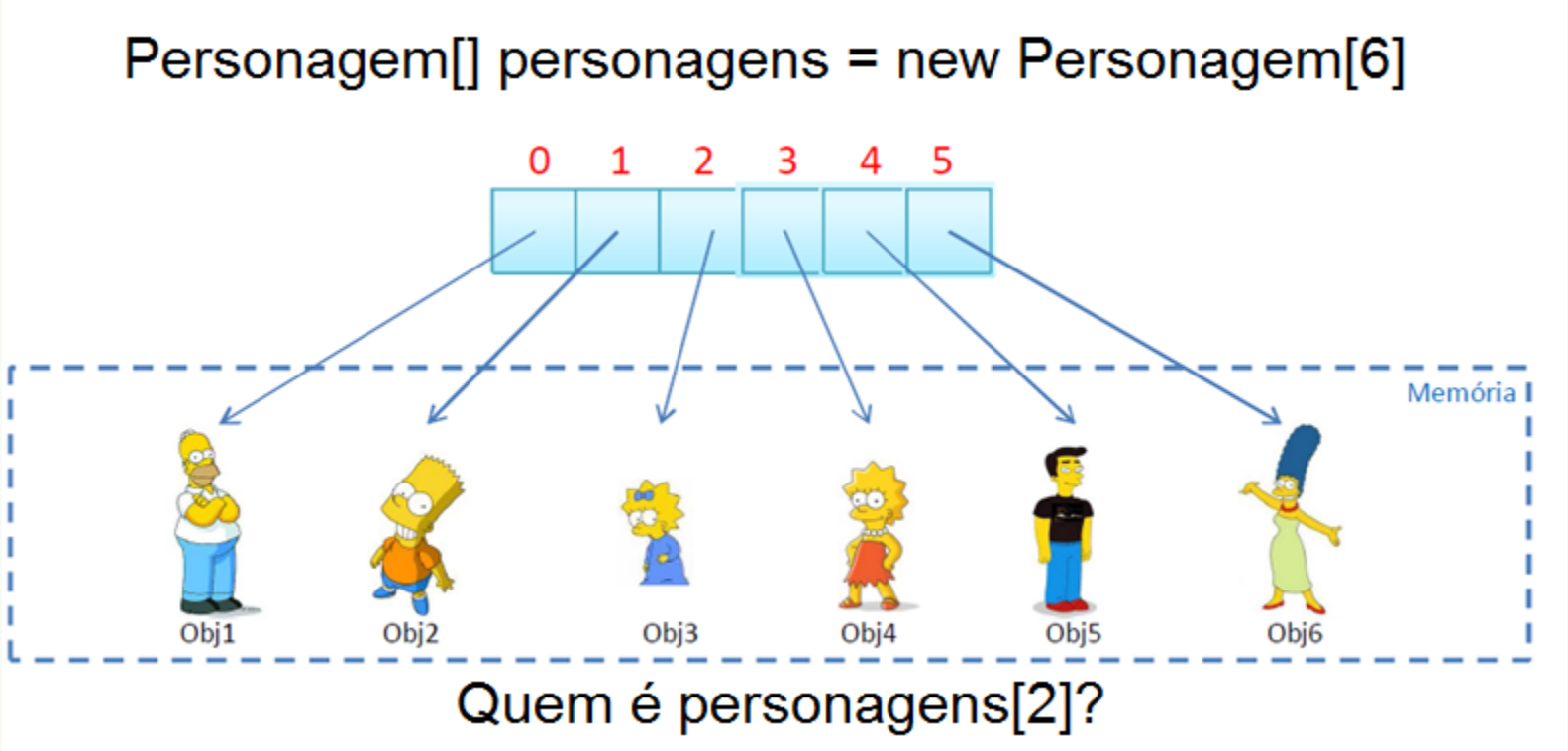
Arrays


```
boolean[] arrayDeBooleans;
arrayDeBooleans = new boolean[8];

int[] arrayDeInteiros;
arrayDeInteiros = new int[10];

String[] arrayDeStrings;
arrayDeStrings = new String[15];

Celular[] arrayDeCelular;
arrayDeCelular = new Celular[999];
```

Arrays

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Os elementos do Array são acessados por meio da posição em que eles se encontram (índice)



Array

- Como um vetor é um objeto ele possui propriedades e métodos
  - Uma propriedade útil:
    - length – diz o tamanho do vetor
- Como percorrer todos os elementos de um vetor?
  - Do primeiro índice até o último!

Comandos de Repetição

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Java possui dois tipos de laços do tipo For
  - FOR “each” – serve para percorrer um conjunto de elementos

```
for(int elemento: coleção)
    ação();
```

  - Exemplo

```
String[] nomes = {"a", "b", "c"};
for(String nome: nomes) {
    System.out.println(nome);
}
```

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Crie uma classe Aluno com os atributos nome e nota. Leia os dados de 10 alunos e, após a leitura, informe, para cada aluno, se ele foi aprovado ou reprovado.
  - Resposta https://repl.it/DzWz/4
- Leia o nome de 10 pessoas e, após a leitura, informe os nomes lidos concatenados.
  - Resposta https://repl.it/DzXR/3

Palavras Reservadas

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

abstract	boolean	break	byte	case	catch
char	class	const	continue	default	do
double	else	extends	final	finally	float
for	goto	if	implements	import	instanceof
int	interface	long	native	new	package
private	protected	public	return	short	static
strictfp	super	switch	synchronized	this	throw
throws	transient	try	void	volatile	while
assert					

ADENDO

Algumas **bibliotecas** (?) muito úteis em Java, seus pacotes e como usá-las

Contextualização

- Muitos projetos tem funcionalidades em comum!
- Não precisamos sempre programar tudo!
  - Já existe muita coisa pronta!
  - Precisamos aprender a usar!

Contextualização

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- O Java tem uma ampla gama de recursos padrão
  - Desde de tocar um som até enviar um e-mail.
- Esses recursos fazem parte da API (*Application Programming Interface*) – em português: Interface de Programação de Aplicativos – padrão do Java.

Contextualização

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- A API é um grupo de programas de suporte destinados a cumprir funções específicas
  - Essas funções estão divididas em diferentes partes (pacotes) relacionadas as suas funcionalidades (o que nós denominamos bibliotecas).
- Para usá-las não precisamos saber como elas realizam tal tarefa, mas apenas como usar.
  - As vezes, descobrir como usar uma classe é intuitivo e simples, outras precisamos consultar a documentação

Documentação

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Todas as classes e métodos que vamos estudar possuem sua documentação online e que pode ser acessada pelo link abaixo
  - https://docs.oracle.com/javase/7/docs/api/
- Também é possível gerar a documentação das nossas classes, por meio de uma linguagem de marcação chamada JavaDoc

Pacotes mais usados da API Java

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Pacote API	Recurso
java.awt	Recursos Gráficos
java.io	Entrada e Saída de Dados
java.lang	**Recursos Fundamentais da Linguagem Java**
java.math	Operações Matemáticas
java.util	Miscelânea de Recursos Utilitários

Classes que vamos conhecer na aula de hoje

- java.lang.Math
- java.lang.String
- java.lang.Integer

Math

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- A classe Math do pacote **java.lang** contém uma série de métodos matemáticos bastante úteis
  - Ex. Gerar Número Aleatório, Arredondar um número, Calcular a Raiz Quadrada, etc.
- Possui definida duas constantes
  - public final static double PI
  - public final static double E
- **Todos os métodos da classe Math são static**
  - Isso significa para invocá-los não precisamos instanciar um objeto Math!

Math: métodos úteis

- Math.max
- Math.min
- Math.ceil
- Math.floor
- Math.round
- Math.random
- Math.sqrt
- Math.pow

Math

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Método	O que faz	Exemplo de utilização	Resultado
Math.max	Retorna o maior valor entre os valores fornecidos	Math.max(145, 159)	159
Math.min	Retorna o menor valor entre os valores fornecidos	Math.min(2, 3)	2

Math

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Método	O que faz	Exemplo de utilização	Resultado
Math.abs	Retorna o módulo do valor passado como parâmetro	Math.abs(-154)	154
Math.ceil	Arredonda o parâmetro para cima	Math.ceil(8.02)	9.0
Math.floor	Arredonda para baixo	Math.floor(8.8)	8.0
Math.round	Arredonda para o inteiro mais próximo	Math.round(1.5) e Math.round(1.4)	2 e 1

Math

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Método	O que faz	Exemplo de utilização	Resultado
Math.random	Retorna um número aleatório no intervalo [0, 1[	Math.random()	0.88
Math.sqrt	Retorna a raiz quadrada do número passado como parâmetro	Math.sqrt(4)	2
Math.pow	Retorna a potência do primeiro parâmetro elevado ao segundo parâmetro	Math.pow(2,3)	8

Math

- Para mais detalhes consultem a documentação da classe Math
  - https://docs.oracle.com/javase/7/docs/api/java/lang/Math.html

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Escreva um programa que leia dois números, calcule a potência do primeiro elevado ao segundo, e imprima a raiz quadrada do resultado, arredondando para cima.
  - Resposta https://repl.it/Dzkk/1
- Escreva um programa que gere dois números aleatórios, o primeiro entre 0 e 100 e o segundo entre 51 e 100, em seguida, imprima o maior desses dois números elevado ao cubo.
  - Resposta https://repl.it/Dzkn/2
- Escreva um programa que calcule a raiz cúbica de um número digitado pelo usuário.
  - Resposta https://repl.it/DzlN/1

String

- String é uma cadeia de caracteres
- Em Java
  - Não há um tipo primitivo para String
  - Strings são objetos

String

- O estudo de como as Strings funcionam em baixo nível em Java é algo relativamente avançado que exigiria bastante tempo e que é uma particularidade de Java.
- Nesse momento, o que nos interessa é
  - o que a classe String pode fazer pode nós?

String

- public **boolean** equals(String other)
  - Compara se duas String são iguais
- public **boolean** equalsIgnoreCase(String other)
  - Compara se duas Strings são iguais considerando letras maiúsculas iguais a minúsculas

```
String nome = "teste";
System.out.println(nome.equals("Teste"));
System.out.println(nome.equalsIgnoreCase("Teste"));
```

String

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- public **int** length()
  - Retorna o comprimento da String

```
String nome = "123";
System.out.println(nome.length());
```

- public **char** charAt(int i)
  - Retorna o caractere de uma posição específica

```
String nome = "abc";
System.out.println(nome.charAt(2));
```

String

- public **String** toLowerCase()
  - Retorna a String em caixa baixa

```
String nome = "Teste";
String nomeCaixaBaixa = nome.toLowerCase();
```

- public **String** toUpperCase()
  - Retorna a String em caixa alta

```
String nome = "Teste";
String nomeCaixaAlta = nome.toUpperCase();
```

- public **String** replace(char old, char new)
  - Retorna uma String com os caracteres substituídos

```
String nome = "AbAcAxi";
System.out.println(nome.replace('A', 'a'));
```

String

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Para mais detalhes consultem a documentação da classe String
  - https://docs.oracle.com/javase/7/docs/api/java/lang/String.html

Exercício

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Escreva um programa que leia uma String e substitua suas vogais por números (A por 4, E por 3, O por 0, I por 1).
  - Resposta https://repl.it/EBJO/3
- Escreva um programa leia uma String e a imprima essa String invertida.
  - Resposta https://repl.it/EBJ0/1
- Escreva um programa que leia uma String e imprima ela com as consoantes em caixa baixa e as vogais em caixa alta.
  - Resposta https://repl.it/EBJb/0

Envelopadores

- Os envelopadores (*wrappers*, em inglês) são classes em Java que servem para representar os tipos primitivos
- Cada tipo primitivo possui uma classe para representá-la

Envelopadores

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

Tipo Primitivo	Classe
boolean	Boolean
char	Char
byte	Byte
short	Short
int	**Integer**
long	Long
float	Float
double	Double

Envelopadores

- Além de representar os tipos primitivos, essas classes possuem diversos métodos bastante úteis

Exemplos

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- A classe Integer possui o método longValue que converte um inteiro para double

```
Integer i = new Integer(1);
double d = i.doubleValue();
```

- A classe Integer possui o método estático parseInt, o qual converte uma String em um número inteiro.

```
int i = Integer.parseInt(“1”);
```

- A classe Integer possui o método estático valueOf, o qual converte uma String em um objeto do tipo Integer

```
Integer i = Integer.valueOf(“1”);
```

Integer

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA

- Para mais detalhes consultem a documentação da classe Integer
  - https://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html

Exercício

- Escreva um programa que leia um número em hexadecimal e o imprima na base decimal.
- Escreva um programa que leia um número em decimal e o imprima em hexadecimal.
- Escreva um programa que leia um número na base decimal, o imprima em binário e informe a soma dos dígitos um presentes nesse número binário.
  - Pesquise que método faz isso

Pacotes

- Também podemos criar pacotes para organizar nossas classes.
- Essa é uma prática de programação aconselhada
  - Agrupar classes pela sua “semelhança”
- Criar nomes de pacotes significativos também
- Exemplo Eclipse!

Próxima Aula

Classes e Objetos