**Aula 05 - Classe e Atributo**

**1. Classe**

Podemos definir uma classe como a estrutura de um objeto. Uma classe pode possuir atributos (características) e métodos (ações). É a partir de uma classe que são instanciados (criados) objetos.

Exemplo:

Classe Pessoa

package classe\_atributo;

public class Pessoa {

public *int* id;

public *String* nome;

}

Para que a classe possa ser utilizada e estar visível em outras classes do projeto Java, é necessário a inclusão da palavra reservada public. Public, neste caso, é um especificador de acesso.

**2. Atributo**

Atributo são as propriedades ou características de uma classe. Quando a classe for utilizada para instanciar um objeto, os atributos, serão os “campos” de um objeto.

Exemplo:

Neste caso os atributos são id e nome:

package classe\_atributo;

public class Pessoa {

public *int* id;

public *String* nome;

}

Para que os atributos possam ser utilizados e estejam visíveis em outras classes do projeto Java, é necessário a inclusão da palavra reservada public. Public, neste caso, é um especificador de acesso.

**3. Objeto**

Objeto é a instância (criação) de uma classe. Ou seja, um objeto, e a criação de uma instância na memória do computador com toda a estrutura de uma classe (atributos e métodos).

Exemplo:

Neste caso o objeto é a variável **p**:

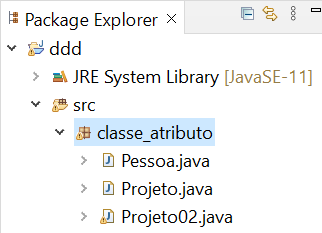
*Pessoa* p = **new** Pessoa();

Podemos fazer uma analogia sobre a relação de Classe e Objeto, comparando com:

- Receita de Bolo (Classe) e a partir da receita, posso fazer (instanciar) vários bolos (Objeto)

- Planta de uma casa (Classe) e a partir de uma planta feita pelo Engenheiro Civil, posso construir (instanciar) várias casas (Objeto)

**4. Projeto Java utilizando Classe, Atributos e um programa para consumir as classes**

****

Pessoa.java

package classe\_atributo;

public class Pessoa {

public *int* id;

public *String* nome;

}

Projeto.java

package classe\_atributo;

public class Projeto {

public static *void* main(*String*[] *args*) {

// Cria um array contendo 10 posições para armazenar objetos

*Pessoa*[] listaPessoas = **new** *Pessoa*[10];

// Instancia (cria) o objeto

*Pessoa* a = **new** Pessoa();

// Popula o objeto (preenche os atributos do objetos)

a.id = 1;

a.nome = "João";

// Adiciona o objeto no array "listaPessoas"

listaPessoas[0] = a;

// =====================================

// Instancia (cria) o objeto

*Pessoa* b = **new** Pessoa();

// Popula o objeto (preenche os atributos do objetos)

b.id = 2;

b.nome = "Maria";

// Adiciona o objeto no array "listaPessoas"

listaPessoas[1] = b;

// =====================================

// Instancia (cria) o objeto

*Pessoa* c = **new** Pessoa();

// Popula o objeto (preenche os atributos do objetos)

c.id = 3;

c.nome = "José";

// Adiciona o objeto no array "listaPessoas"

listaPessoas[2] = c;

// =====================================

System.out.printf("Pessoa %d criada com sucesso (%s)!", listaPessoas[0].id, listaPessoas[0].nome);

System.out.printf("\nPessoa %d criada com sucesso (%s)!", listaPessoas[1].id, listaPessoas[1].nome);

System.out.printf("\nPessoa %d criada com sucesso (%s)!", listaPessoas[2].id, listaPessoas[2].nome);

}

}

Projeto02.java

package classe\_atributo;

import java.util.Scanner;

public class Projeto02 {

public static *void* main(*String*[] *args*) {

*Scanner* ler = **new** Scanner(System.in);

// Cria um array contendo 10 posições para armazenar objetos

*Pessoa*[] listaPessoas = **new** *Pessoa*[10];

for (*int* i=0; i<=2; i++) {

// Instancia (cria) o objeto

*Pessoa* p = **new** Pessoa();

// Popula o objeto (preenche os atributos do objetos)

System.out.printf("Digite o id da pessoa: ");

p.id = ler.nextInt();

System.out.printf("Digite o nome da pessoa: ");

p.nome = ler.next();

// Adiciona o objeto no array "listaPessoas"

listaPessoas[i] = p;

}

for (*int* i=0; i<=2; i++) {

System.out.printf("\nPessoa %d criada com sucesso (%s)!", listaPessoas[i].id, listaPessoas[i].nome);

}

}

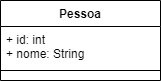
}

**5. Linguagem UML**

UML (Unified Modeling Language) é uma linguagem baseada em diagramas para documentar sistemas orientado a objetos.

O UML possui diversos diagramas e um deles é o Diagrama de Classe.

Com isso, se formos utilizar o UML para documentar a nossa classe Pessoa, faríamos isso:



Pois a nossa classe **Pessoa** é composta de apenas 2 atributos:

package classe\_atributo;

public class Pessoa {

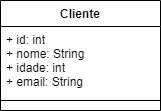
public *int* id;

public *String* nome;

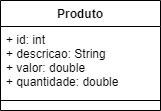
}

**6. Exercícios**

51. Crie uma classe conforme o Diagrama de Classe (UML). Crie um programa que utilize essa classe para cadastrar 5 clientes em uma lista de clientes (array). Ao final exibir apenas os clientes que têm idade superior a 18 anos.



52. Crie uma classe conforme o Diagrama de Classe (UML). Crie um programa que utilize essa classe para cadastrar 10 produtos em uma lista de produtos. Ao final exibir apenas os produtos cujo valor é menor do que 100 reais.



53. Crie uma classe conforme o Diagrama de Classe (UML). Crie um programa que utilize essa classe para cadastrar 5 alunos em uma lista de alunos. Durante o cadastro do aluno, solicitar para o aluno cadastrar a sigla/iniciais das 6 matérias que ela está cursando no momento. Ao final exibir apenas os alunos do período “noite”.

