

Orientação a objetos

Programação orientada a objetos

- Conceitos da OO:
- Em orientação a objetos, uma classe é uma estrutura que abstrai um conjunto de objetos com características similares.
- A classe define o comportamento dos objetos através de métodos e atributos
 - Métodos = funções na linguagem estruturada
 - Atributos = variáveis na linguagem estruturada

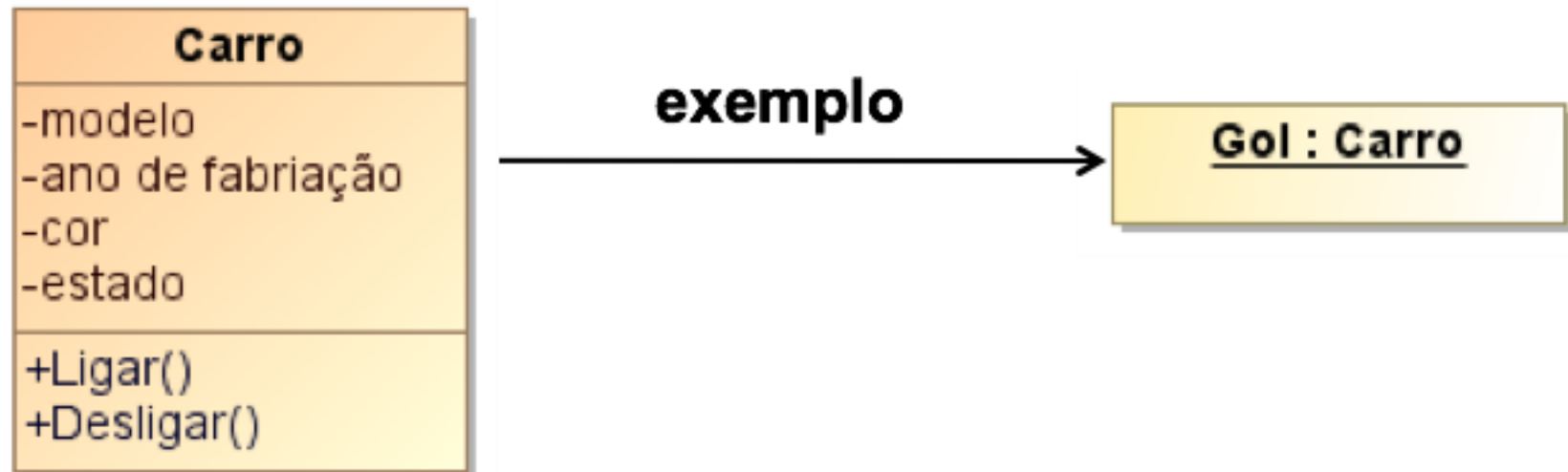
Programação orientada a objetos

- A classe descreve as características e funcionalidades dos objetos



Programação orientada a objetos

- Conceitos da OO:
- O objeto é a instância de uma classe
- Ex: Classe Carro e Objeto Gol



Programação orientada a objetos

- Conceitos da OO:
- É possível ter vários objetos a partir de uma classe

Celta : Carro

Corsa : Carro

Civic : Carro

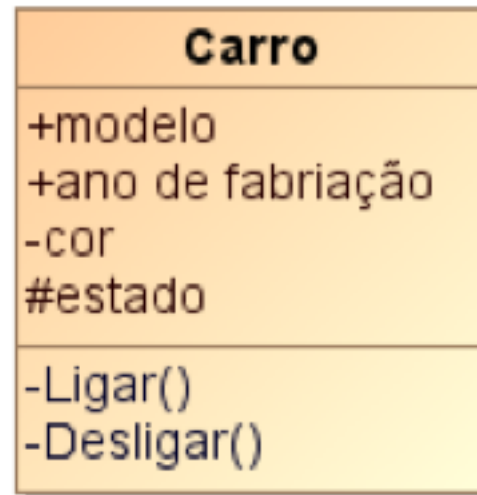
Meriva : Carro

Programação orientada a objetos

- Estado e Comportamento
 - O estado de um objeto é representado por seus atributos
 - O comportamento é representado pelos seus métodos
- Exemplo:
 - Classe Carro
 - Objeto Gol
 - Atributos: modelo, ano, cor, estado
 - Métodos: ligar, desligar

Programação orientada a objetos

- Encapsulamento:
 - É o empacotamento dos atributos e métodos numa classe
 - Proteção dos dados:
 - Público (+)
 - Privado (-)
 - Protegido (#)



Programação orientada a objetos

- Em resumo:
 - Classe: representa um conjunto de objetos
 - Subclasse: classe filha que herda os atributos e os métodos da classe mãe (Herança)
 - Objeto: instância de uma classe
 - Atributo: características do objeto
 - Método: funcionalidades do objeto

Programação orientada a objetos

- E veremos futuramente:
 - Subclasse: classe filha que herda os atributos e os métodos da classe mãe (Herança)
 - Mensagem: troca de informação entre os objetos
 - Associação: utilização de recursos entre objetos
 - Abstração: classe não instanciável
 - Polimorfismo: métodos com muitas formas: sobrecarga, sobrescrita

Programação orientada a objetos

- Modelagem Orientada a Objetos
- Linguagem UML
 - - Padrão para modelagem de sistemas
 - - Amplamente difundida e consolidada
 - - Largamente utilizada na indústria de software
- Mais informações em: <http://www.uml.org>

Programação orientada a objetos

- Perguntas e Respostas:
 - O que é uma Classe?
 - O que é um Objeto?
 - O que é um Método?
 - O que é um Atributo?

Classes e atributos

- Primeira classe em Java

```
3 public class Pessoa {  
4     public String nome;  
5     public int idade;  
6 }
```

- - Qual o nome da classe?
- - Qual o nome do arquivo .java?
- - A classe pode ter um nome diferente do nome do arquivo?
- - A classe possui atributos e métodos? Quais?
- - O que faz a classe Pessoa?
- - Ela é executável?

Classes e atributos

- Instanciando a classe Pessoa

```
3 public class Pessoa {  
4     public String nome;  
5     public int idade;  
6 }
```

Pessoa.java

```
3 public class Principal {  
4  
5     public static void main(String[] args) {  
6         Pessoa p = new Pessoa();  
7         System.out.println("A pessoa foi instanciada!");  
8     }  
9 }
```

Principal.java

- - Instanciar = criar um objeto
- - Qual o objeto criado?
- - Qual o resultado da execução da classe Principal?

Classes e atributos

- Classe Pessoa com valores nos atributos

```
3 public class Pessoa {  
4     public String nome = "Ricardo";  
5     public int idade = 21;  
6 }
```

- - Atributos definidos na classe com valores padrão (default);

Classes e atributos

- Instanciando a classe Pessoa novamente e obtendo o valor dos atributos

```
3 public class Principal {  
4  
5     public static void main(String[] args) {  
6         Pessoa p = new Pessoa();  
7         System.out.println("O nome da pessoa é: "+p.nome);  
8         System.out.println("E tem "+p.idade+" anos!");  
9     }  
10 }
```

- Qual o resultado da execução da classe Principal?
- Acesso direto aos atributos e aos valores dos atributos !!! Como isso é possível ? Pelo fato de os atributos estarem definidos como public.
 - Se o(s) atributo(s) estivesse(m) com visibilidade private, apenas a própria classe poderia acessar o(s) atributo(s).

Classes e atributos

- Pergunta: mas se a Classe deve ser um “molde”, todos os objetos desse molde sairão com os valores default quando instanciados? Isso é correto? É o que espera-se?
- Normalmente, cria-se um objeto e, após, define-se os valores dos atributos.
- A ideia é: cada objeto tenha seus atributos valorados de acordo com a necessidade, após a instanciação, e não a classe já passe esses valores no momento da instanciação.
- Salvo exceções !

Classes e atributos

- Criando objetos (instanciando classes) em somente depois definindo valores aos atributos;

```
3 public class Pessoa {
4     public String nome;
5     public int idade;
6 }

3 public class Principal {
4
5     public static void main(String[] args) {
6         Pessoa p1 = new Pessoa();
7         p1.nome = "João";
8         Pessoa p2 = new Pessoa();
9         p2.nome = "Maria";
10        System.out.println("O nome da pessoa 1 é: "+p1.nome);
11        System.out.println("O nome da pessoa 2 é: "+p2.nome);
12    }
13 }
```

Classes e atributos

- Jogo rápido: exibir apenas a idade de três pessoas, ou seja: criar três objetos do tipo Pessoa;

Classes e atributos

```
3 public class Principal {  
4  
5     public static void main(String[] args) {  
6         Pessoa p1 = new Pessoa();  
7         p1.idade = 20;  
8         Pessoa p2 = new Pessoa();  
9         p2.idade = 22;  
10        Pessoa p3 = new Pessoa();  
11        p3.idade = 24;  
12        System.out.println("A idade da pessoa 1 é: "+p1.idade);  
13        System.out.println("A idade da pessoa 2 é: "+p2.idade);  
14        System.out.println("A idade da pessoa 3 é: "+p3.idade);  
15    }  
16 }
```

Classes e atributos

- Jogo rápido (2): Faça a leitura pelo teclado dos dados das três pessoas e apresente na tela.

```
5 public class Principal {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         Pessoa p1 = new Pessoa();
9         System.out.println("Digite o nome da Pessoa 1: ");
10        p1.nome = sc.next();
11        System.out.println("Digite a idade da Pessoa 1: ");
12        p1.idade = sc.nextInt();
13        Pessoa p2 = new Pessoa();
14        System.out.println("Digite o nome da Pessoa 2: ");
15        p2.nome = sc.next();
16        System.out.println("Digite a idade da Pessoa 2: ");
17        p2.idade = sc.nextInt();
18        Pessoa p3 = new Pessoa();
19        System.out.println("Digite o nome da Pessoa 3: ");
20        p3.nome = sc.next();
21        System.out.println("Digite a idade da Pessoa 3: ");
22        p3.idade = sc.nextInt();
23        System.out.println("Pessoa 1: "+p1.nome+" Idade: "+p1.idade);
24        System.out.println("Pessoa 2: "+p2.nome+" Idade: "+p2.idade);
25        System.out.println("Pessoa 3: "+p3.nome+" Idade: "+p3.idade);
26    }
27 }
```

Classes e atributos

- Jogo rápido (3): crie uma classe chamada Professor que contenha um atributo público chamado nome, do tipo String. Crie também uma classe chamada Laboratorio, que contenha um atributo público, chamado local, do tipo String.
- Além disso, crie uma classe executável chamada Disciplina, responsável por instanciar as classes Professor e Laboratorio, definindo valor aos atributos e exibindo na tela o resultado da criação desses objetos.
- Por exemplo: o resultado da execução da classe Disciplina deve ser algo similar com:
 - O nome do professor é: Ricardo da Silva
 - O local da aula é: Sala 108

Classes e atributos

```
3 public class Professor {  
4     public String nome;  
5 }
```

```
3 public class Laboratorio {  
4     public String local;  
5 }
```

```
3 public class Disciplina {  
4     public static void main(String[] args) {  
5         Professor p = new Professor();  
6         p.nome = "Ricardo da Silva";  
7  
8         Laboratorio lab = new Laboratorio();  
9         lab.local = "Sala 108";  
10  
11         System.out.println("O nome do professor é: "+p.nome);  
12         System.out.println("O local é: "+lab.local);  
13     }  
14 }
```

Exercícios

- 1 - Crie uma classe chamada Carro que possua 3 atributos: marca, modelo e anoFabricacao. Além disso, crie uma classe executável para criar 2 objetos do tipo Carro e exibir na tela os valores dos atributos criados.
- Esses valores devem ser solicitados ao usuário, por linha de execução.
- 2 - Crie uma classe chama Computador que possua 4 atributos: marca, modelo, tipo (notebook, netbook, tablet, etc) e preço.
- Essa classe Computador também deverá ser executável, entretanto crie uma outra classe executável que irá instanciar a classe Computador, criando 2 objetos. O primeiro objeto deverá ser criado através da solicitação dos valores ao usuário por linha de execução. Já, os valores do segundo objeto, deverão ser definidos no código-fonte da classe que o criou. Exiba na tela os resultados.

Exercícios

- 3 - Crie uma classe Pessoa com os atributos nome, idade e gênero. Imprima as informações da pessoa na tela.
- 4 - Crie uma classe Livro com os atributos título, autor e ano de publicação. Imprima as informações do livro na tela.