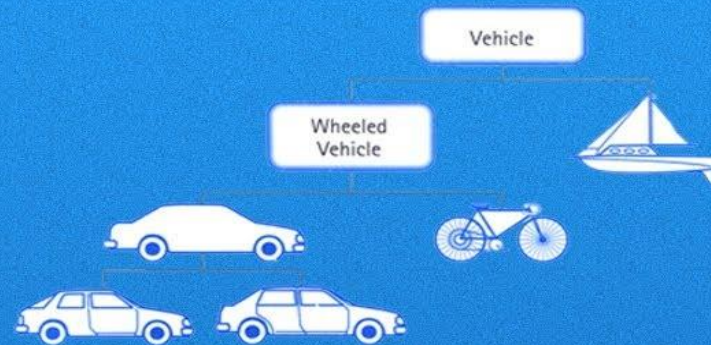


Programação I

PROGRAMAÇÃO ORIENTADA A OBJETOS



P OO

Conceito de P OO – cont.

Definição da POO

➤ Linguagens orientadas a objetos:

- Ao invés de passar dados a procedimentos, requisita-se que **objetos realizem operações** neles próprios.

O que são objetos?

Objetos são **instâncias** de classes, que determinam qual informação um objeto contém e como ele deve manipulá-la.

O que são objetos?

Objeto é o **conceito abstrato** que deve se parecer com o **artefato real** que estamos implementando, tanto nas suas características (**atributos**), quanto na sua utilidade (**seus métodos**).

O que são objetos?

Todo objeto tem um **tipo** ou seja **modelo**.
Objeto é uma instância e uma classe, onde classe é um sinônimo de tipo.

O que são objetos?

Objetos são como uma única **entidade autônoma** que combina a representação da **informação** (estrutura de dados) e sua **manipulação** (procedimentos), uma vez que possuem capacidade de processamento e armazenam um estado local.

O que são classes?

Classe é um **modelo de objeto** onde são implementadas **características** e **funcionalidades** desse objeto.

O que são classes?

Classes interagem entre si para formarem componentes, objetos maiores, bibliotecas, pacotes, etc.

O que são classes?

Cada objeto criado a partir de uma classe é chamado de ***instância*** dessa classe.

Uma classe provê toda a informação necessária para construir e utilizar objetos de um tipo particular.

O que são classes?

Em termos de programação, **para criarmos um objeto**, definindo seus **atributos** e **operações**, precisamos de **uma entidade** para reunir essas características do objeto.

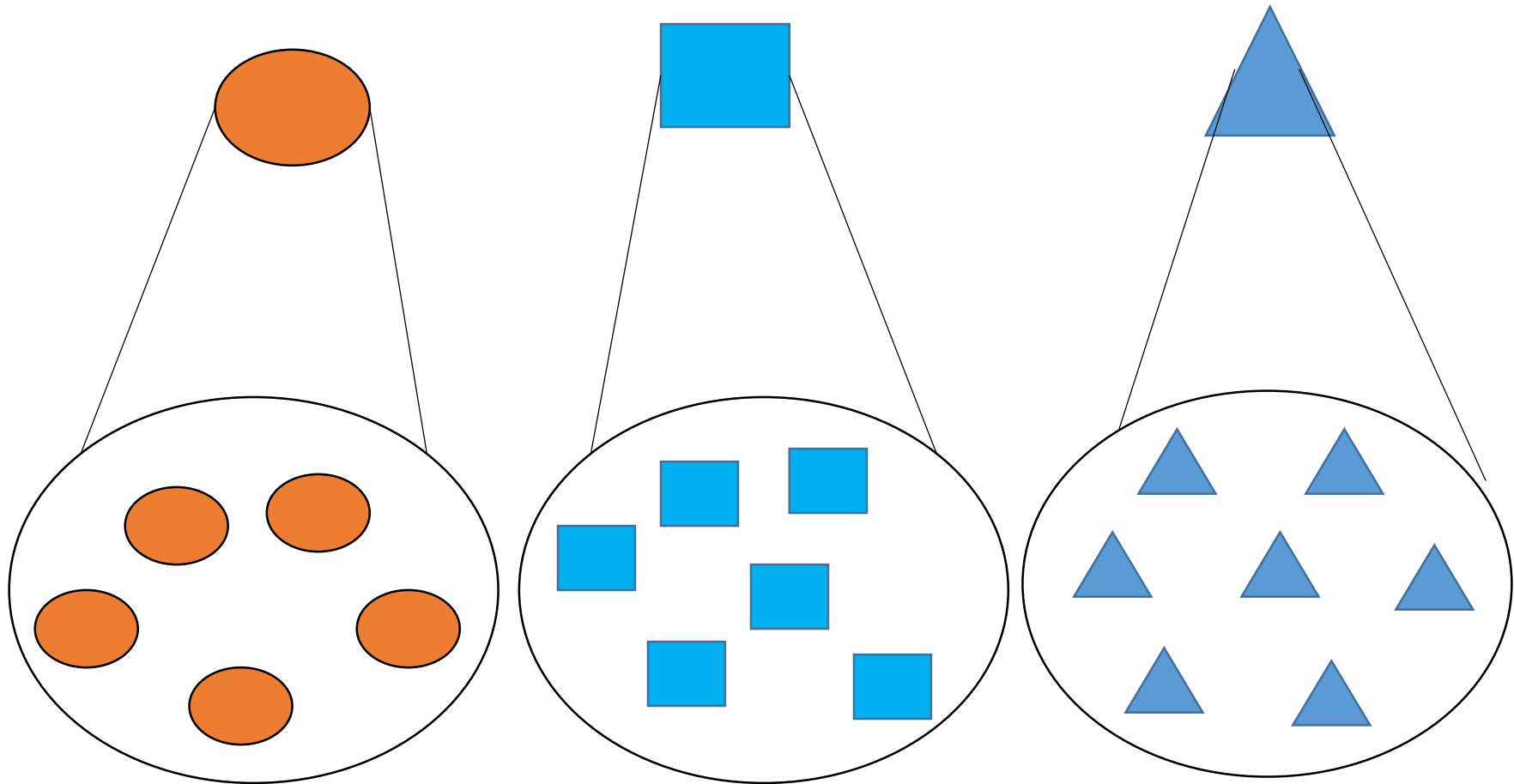
O que são classes?

Essa entidade deverá **garantir** que naquele ***objeto*** só estão sendo inseridos **atributos** e **operações** que realmente fazem parte daquele objeto.

O que são classes?

Após sua finalização, esta entidade será o **modelo do objeto** e poderá ser utilizada quantas vezes aqueles **atributos** ou **operações** do objeto forem necessários.

Representação do relacionamento entre classes e objetos



Objetos da Classe Círculo

Objetos da Classe Quadrado

Objetos da Classe Triângulo

Classes

➤ No exemplo de uma **empresa**, podemos criar classes para **várias entidades**, como: empregados, departamentos, filiais, produtos, documentos, etc.

Classes

- Na definição de uma **Classe *Empregado***, são especificados as suas propriedades e comportamento.
- Quando um **novo empregado** é admitido na empresa, ele é criado como uma **instância da Classe *Empregado***, e automaticamente possui todas as características da classe.

Vantagens da POO

A POO tem alcançado muita popularidade devido às suas vantagens, como:

Reusabilidade de código;

Escalabilidade de Aplicações;

Mantenabilidade;

Vantagens da POO

Reusabilidade

Permite que programas sejam escritos mais rapidamente;

Demanda maior, necessidade de desenvolvimento de sistemas mais rápido.

Vantagens da POO

Reusabilidade

Possibilita o desenvolvimento de novos sistemas utilizando muito código já existente.

A herança é o ponto principal para a reusabilidade de código.

Vantagens da POO

Escalabilidade

Capacidade de uma aplicação crescer facilmente sem acréscimo demasiado de complexidade ou comprometimento no seu desempenho.

Vantagens da POO

Escalabilidade

A POO é adequada para o desenvolvimento de grandes sistemas, visto que pode-se construir e ampliar um sistema agrupando objetos e fazendo-os trocar mensagens entre si.

Vantagens da POO

Encapsulamento

Proporciona ocultamento e proteção da informação.

Os acessos à objetos somente podem ser realizados através das mensagens que ele está *habilitado* a receber.

Vantagens da POO

Polimorfismo

O polimorfismo torna o programa mais enxuto, claro e fácil de compreender.

Sem a utilização do polimorfismo, seriam necessárias um número imenso de métodos com nomes diferentes mas com comportamento similar.

Em termos de manutenção, significa que o programa será mais facilmente entendido e alterado.

Vantagens da POO

Herança

A herança torna a manutenção mais fácil.

Caso a aplicação precisar de alguma funcionalidade adicional, não há a necessidade de alteração do código atual.

Vantagens da POO

Herança

Basta criar uma nova geração de uma classe, herdando o comportamento antigo e adicionando novo comportamento, ou até mesmo redefinindo o comportamento antigo.

Modificadores de Acesso

- Uma classe pode ser:
 - **public** – acessado por qualquer outra classe.
 - **nenhum modificador** (package-private) – *acessada somente dentro do seu pacote.*

Modificadores de Acesso

- **Atributos e métodos podem ser:**
 - **public** – acessados por qualquer outra classe.
 - **nenhum modificador** (package-private) – acessados somente dentro do seu pacote
 - **private** – acessados somente dentro de suas próprias classes.
 - **protected** – acessados somente dentro do seus pacotes e por suas subclasses.

Modificadores de Acesso

modificador	Classe/ Atributos ou métodos	pacote	subclasse	todos
public	Sim	Sim	Sim	Sim
protected	Sim	Sim	Sim	Não
nenhum	Sim	Sim	Não	Não
private	Sim	Não	Não	Não

E então? Agora que conhecemos um pouco sobre Orientação a objetos...



Bora trabalhar?

