

Aula 01: Conceitos Avançados de Orientação a Objetos

USJT 2019 – Arquitetura e Desenvolvimento de Sistemas Multicamadas
Prof. Sergio Bonato

- Classes Abstratas
- Interfaces
- Polimorfismo
- Enumeradores
- Anotações

Classe Abstrata

- Não pode ser instanciada
- Pode ou não ter métodos concretos (com implementação)
- Pode ou não ter métodos abstratos
- Um método abstrato é aquele que é declarado como abstrato e não possui implementação, isto é, o corpo do método está ausente (*method body*)

```
public abstract double salario();
```

```
public abstract class Empregado{  
    private String nome;  
  
    public Empregado(String nome){  
        this.nome = nome;  
    }  
  
    public String getName(){  
        return nome;  
    }  
  
    public void setName(String nome){  
        this.nome = nome;  
    }  
  
    public abstract double salario();  
}
```

Interfaces

- É parecida com uma classe abstrata, mas não pode ter métodos concretos, isto é, com implementação.
- Só pode ter métodos sem implementação, mas não é necessária declará-los como abstratos.

```
public interface Bonus {  
    public double bonus();  
}
```

Como usar Interfaces e Classes Abstratas

- Uma classe concreta que estende uma classe abstrata precisa fazer a sobreposição dos métodos abstratos herdados.
- Uma classe concreta que implementa uma interface precisa fazer a sobreposição dos métodos abstratos herdados.

```
public class Gerente extends Empregado implements Bonus {  
  
    private double salario;  
  
    public Gerente(String nome, double salario) {  
        super(nome);  
        this.salario = salario;  
    }  
  
    public void setSalario(double salario) {  
        this.salario = salario;  
    }  
  
    @Override  
    public double bonus() {  
        return salario * 0.5;  
    }  
  
    @Override  
    public double salario() {  
        return salario;  
    }  
}
```

Quando usar classe abstrata e quando usar interface?

- Classe abstrata:
 - Para compartilhar código entre classes bem parecidas
- Interface
 - As classes que irão implementar a interface não tem muito uma a ver com a outra
 - Você quer implementar um tipo de comportamento independente de quem for adota-lo
 - Classes que já são subclasses de outra precisam também se comportar como a interface (herança múltipla)

Exemplo de Código Polimórfico
usando Classes Abstratas e
Interfaces

Anotações

- São um tipo de metadado, fornecendo dados sobre o programa que não são parte do programa propriamente dito.
- Usos comuns:
 - **Informações para o compilador**
 - **Processamento em tempo de compilação e de implantação**
 - **Processamento em tempo de execução**
- Usados intensivamente em frameworks, como Spring e Hibernate

Anotações

- Básicas

- @Override

- o compilador verifica se o método está sobrecarregando um método da superclasse

- @Suppress Warnings

- desliga os avisos do compilador (e do Eclipse)

- @WebServlet

- informa o endereço de um servlet

- @WebFilter

- informa qual servlet será filtrado por um filtro