

TI-220 Java Orientado a Objetos

ANTONIO CARVALHO - TREINAMENTOS

A solid blue horizontal bar spanning the width of the slide, located at the bottom.

Java Orientação a Objetos

Interface

Interface

Permite a abstração de comportamentos específicos que podem ser aplicados a diferentes classes.

Como exemplo:

- As classes **Carro** e **Cachorro** possuem o comportamento **mover()** porém elas não possuem um herança comum
- As classes **GotaDeChuva** e **BolaBasquete** possuem o comportamento **cair()** , porém elas também não possuem uma herança comum

Interface

A interface é um contrato que diz a uma classe o que ela deve fazer e a classe se encarrega de fazer da maneira que achar melhor.

Os nomes das interfaces normalmente são ***adjetivos***.

Ao criar uma interface chamada **Imprimivel** contendo o método **print()**

É possível usar este contrato em qualquer classe, portanto uma classe chamada **Livro** pode implementa-la, assim como uma classe chamada **Tatuagem** também pode implementar a interface **Imprimivel**

Ambas as classes imprimem, mas cada uma ao seu modo

Interface

As interfaces diferem das classes abstratas pois não é possível ter métodos **não abstratos** em uma interface

As outras regras são :

- Todos os métodos da interface são implicitamente **public** e **abstract**
- A interface não pode conter variáveis, apenas constantes, ou seja as variáveis precisam ser **public, static** e **final**.

Nota : Todas as variáveis criadas na interface são automaticamente **public, static** e **final**

- Não podem existir métodos **static** na interface
- Os métodos na interface não podem ser marcados como **native, strictfp** ou **final**
- A interface pode **estender** quantas outras interfaces quiser.

Nota : Uma interface não pode herdar de uma classe

- Uma interface não pode **implementar** outra interface

Interface

A declaração da interface segue o mesmo princípio de declaração de uma classe porém deve ser feita usando a palavra chave ***interface***.

Sintaxe:

```
[ modificadores ] interface <nome> [ extends <interface1>, <interface2> ...  
<interfaceN> ] {  
}
```

Interface

```
package ocjp.java.certification;  
public interface Imprimivel {  
    public int linhas = 66;  
    public void print();  
}
```

Variável linhas é final portanto
não pode ser mudada



```
package ocjp.java.certification;  
public class Livro implements Imprimivel {  
    @Override  
    public void print() {  
        System.out.println("Imprimindo");  
    }  
}
```

Compile success



```
package ocjp.java.certification;  
public class Livro implements Imprimivel {  
    @Override  
    public void print() {  
        linhas = 80;  
        System.out.println("Imprimindo");  
    }  
}
```

The final field Imprimivel.linhas cannot be
assigned

Dúvidas

