

TI-220 Java Orientado a Objetos

ANTONIO CARVALHO - TREINAMENTOS

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

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.

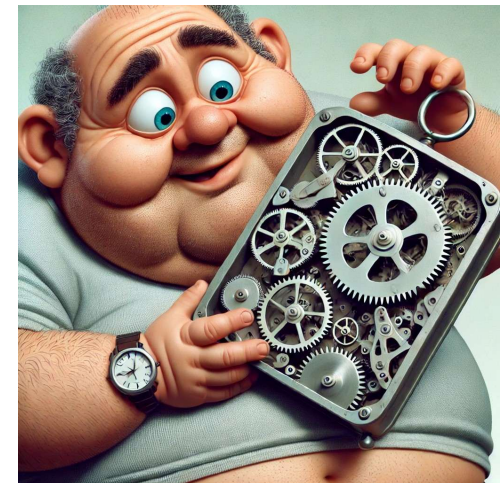
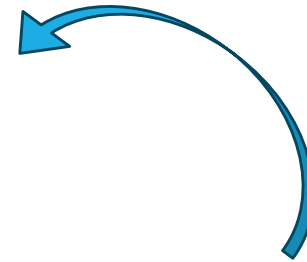
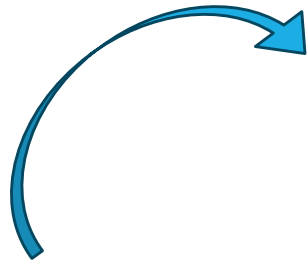
Os nomes das interfaces normalmente são ***adjetivos***.

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



Interface

As interfaces diferem das classes abstratas pois não gasta a herança. Antes do Java8 as interfaces não podiam ter métodos **não abstratos**.

A partir do Java 8 é possível criar um método concreto na interface, por meio do modificador **default**

As outras regras são :

- Todos os métodos não **default** 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**

- 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
- Uma interface não pode ser instanciada diretamente, segue a mesma regra da classe **abstract**

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

Os métodos na interface devem ser abstratos, e não devem conter implementação.


Porém a partir da versão 8 do Java as interfaces permitem métodos padrões, contendo código.

Para usar este recurso o método precisa conter o modificador **default** na sua assinatura e depois o bloco de código.

Sintaxe:

```
default <tipo do retorno> <nome do método>([[tipo1  
parâmetro1], ... [tipoN parametroN]]) {  
    ... código implementado ...  
}
```

A classe que implementar a interface pode não implementar os métodos padrões usando seus códigos.



Interface

Outra facilidade adicionada no Java 8 foram os métodos **static** nas interfaces.

O método **static** é um método sem comunicação com a instância, e portanto deve ser invocado usando a interface e não a instância.

Sintaxe:

```
static <tipo do retorno> <nome do método>([[tipo1  
parâmetro1], ... [tipoN parametroN]]) {  
    ... código implementado ...  
}
```

Os métodos **static** não podem ser sobrescritos na classe que implementa a interface, e eles também apenas podem acessar outros métodos **static**

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 Tatuagem implements Imprimivel {  
    @Override  
    public void print() {  
        linhas = 80;  
        System.out.println("Imprimindo");  
    }  
}
```

The final field Imprimivel.linhas cannot be
assigned

Dúvidas

