



Fluxo de um Programa Constructores Ciclos

Conceitos Fundamentais de Programação Orientada a
Objectos

Conceitos

1

Fluxo de um
Programa

2

Constructores

3

Ciclos Condicionais

Fluxo de um Programa

- Um **programa Java** é composto por **uma ou mais classes**.
- Por norma, cada classe definida tem o seu próprio **ficheiro “.java”**
- Um **ponto de entrada** ou início de um programa é definido pelo método reservado **public static void main(String[] args)**.
- O **fluxo de um programa** é depois controlado por vários tipos de **condicionais**.

Exemplo

```
public class Telefone {  
    private String contacto;  
    public void toca() {  
        System.out.println("RING RING!");  
    }  
  
    public String getContacto() {  
        return this.contacto;  
    }  
}
```

```
public class Controlotelefone {  
  
    public static void main(String []args){  
        Telefone t1 = new Telefone();  
        t1.toca();  
        System.out.println(t1.getContacto());  
    }  
}
```

Resultado:

RING RING!

null

Constructores

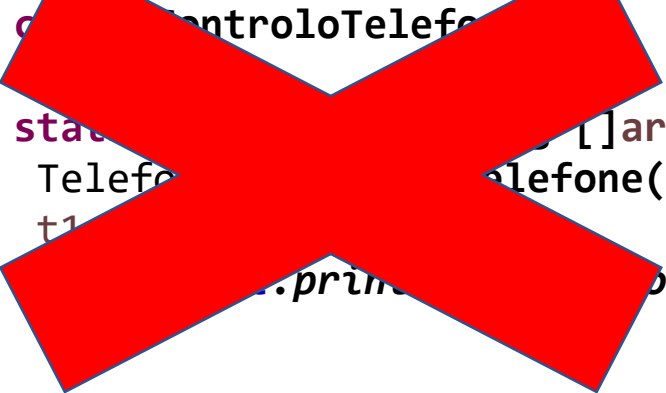
- Métodos especiais que **instanciam um objecto**.
- Todos os objectos têm, por omissão, um **constructor básico**.
- Podemos **criar o nosso próprio constructor** que define certas características de um objecto.
- **ATENÇÃO:** Se criamos o nosso próprio constructor, o construtor por omissão deixa de ser válido!

Exemplo

```
public class Telefone {  
    private String contacto;  
    public void toca() {  
        System.out.println("RING RING!");  
    }  
  
    public String getContacto() {  
        return this.contacto;  
    }  
  
    public Telefone(String contacto) {  
        this.contacto=contacto;  
    }  
}
```

Resultado:

RING RING!
917395735



```
public class ControlloTelefono {  
    public static void main(String []args){  
        Telefone t1 = new Telefone("917395735");  
        t1.toca();  
        System.out.println(t1.getContacto());  
    }  
}  
  
public class ControlloTelefono {  
  
    public static void main(String []args){  
        Telefone t1 = new Telefone("917395735");  
        t1.toca();  
        System.out.println(t1.getContacto());  
    }  
}
```

Ciclos condicionais

- Ciclos são **ferramentas de controlo de fluxo** fundamentais na programação.
- São compostos por uma **condição lógica** e um conjunto de instruções que devem ser executadas enquanto a **condição avaliar para “true”**.
- Regra geral, devem ter **condições de paragem** definidas para que o programa não corra eternamente.

Ciclos condicionais

```
while(condição) {  
    instrução1;  
    instrução2;  
}
```

```
for(inicialização; condição; iteração) {  
    instrução1;  
    instrução2;  
    instrução3;  
}
```


Exemplo

```
public class ControleTelefone {  
  
    public static void main(String []args){  
        Telefone t1 = new Telefone("917395735");  
  
        int numeroTentativas = 5;  
        while(numeroTentativas > 0) {  
            t1.toca();  
            numeroTentativas--;  
        }  
  
        for(numeroTentativas = 5; numeroTentativas > 0; numeroTentativas--) {  
            t1.toca();  
        }  
    }  
}
```

A large, dark blue, irregular ink blot with white splatters on a black background. The blot is roughly circular but has many jagged, feathered edges. Numerous small, white, teardrop-shaped and circular splatters are scattered around the main blot, particularly towards the top and right sides. The overall effect is that of a fresh ink splash or a stylized graphic element.

FIM

Exercícios

Exercício 1

- Cria uma classe “Enemy” que tem atributos que guardam um nome (String), quantidade de vida que ele tem (número inteiro) e quanto ataque ele tem (número inteiro).
- Cria um constructor para esta classe que recebe como argumento o seu nome, a vida e o ataque e os atribui ao objecto “enemy”.
- Cria um método “enemyStats” que imprime na consola todos os atributos da classe.

Exercício 2

- Cria uma classe “Game” que contém o método main.
- Dentro do main, instancia 2 objectos enemy com nomes, vidas e valores de ataque distintos à tua escolha.
- Imprime na consola os valores dos 2 objectos que criaste através do método que definiste no exercício 1.

Exercício 3

- Cria um ciclo while no método main que mostre os valores do primeiro objecto “Enemy” que criaste 3 vezes.
- Cria um ciclo for no método main que mostre os valores do segundo objecto “Enemy” que criaste 4 vezes.