

Construtores

O método construtor é responsável por inicializar o novo objeto.

Ex: `Soma s1 = new Soma();`, onde Soma é uma classe e s1, um atributo do tipo dessa classe ao qual é atribuído um objeto da classe Soma.

Classes provêem métodos construtores para se inicializar objetos daquele tipo.

Construtores

Construtores são métodos cujo nome é o mesmo nome da classe.

```
Ex: public class Soma{  
    public Soma () {}  
  
}
```

O construtor deve ter um modificador.

```
Ex: public Soma () {}
```



Construtores



Não tem tipo de retorno, pois este é o próprio tipo da classe.

São utilizados para realizar a atribuição de valores a atributos da classe, podendo haver diferença na execução de seus métodos em função disso.

A maioria das classes tem pelo menos o construtor default. Caso não o tenha, o próprio compilador criará um.

Construtores

Construtor default não recebe qualquer parâmetro e não executa nenhuma instrução.

Ex: `public Soma() { }`

não recebe parâmetros

não executa nenhuma instrução

Construtores

Classes podem ter múltiplos construtores, todas com o mesmo nome mas com número ou tipos de argumentos diferentes.

Ex:

```
public class Pessoa{  
    private String nome;  
    private int idade;  
    public Pessoa() {}  
    public Pessoa(String novoNome) {  
        setNome(novoNome);  
    }  
    public Pessoa(String novoNome, String novaIdade) {  
        setNome(novoNome);  
        setIdade(novaIdade);  
    }  
}
```



Destrutores



Coleta automática de lixo: conceito de destrutores diferente de seus equivalentes em outras linguagens orientadas a objetos

Em Java: destrutores são métodos chamados pelo sistema quando a memória de um objeto está para ser liberada pelo coletor automático de lixo (não quando está para ser coletada)



Destrutores



Sintaxe:

```
protected void finalize() {  
    //codigo para arrumar a casa, antes que o  
    objeto seja apagado  
}
```



Destrutores



finalize - caracteriza o destrutor

Sem argumentos

Valor de retorno: void



Destrutores

Aplicação:

Objeto tem que avisar a outros que será inutilizado

Nó de lista ligada: faz somente o anterior pontar para o posterior