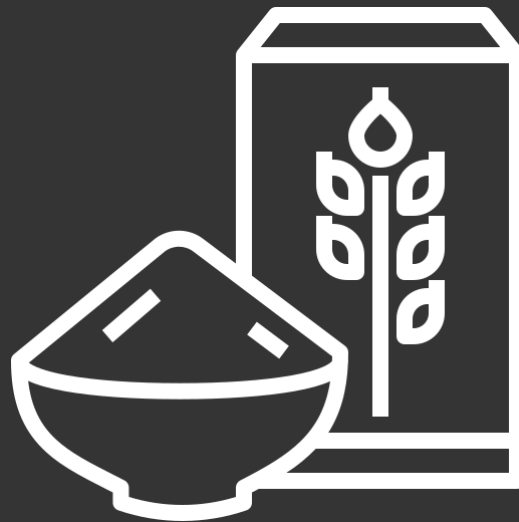


Classes e  
objetos

# Conteúdo

- Classes e objetos
- Instanciação
- Representação na memória
- Propriedades (ou atributos)
- Métodos





Tipos primitivos  
Formatos de  
dados embutidos  
na linguagem

```
int a;
```

```
a = 1;
```

```
boolean y = true;
```

declaração  
de variável

`int a;`

tipo

identificador

atribuição  
de valor

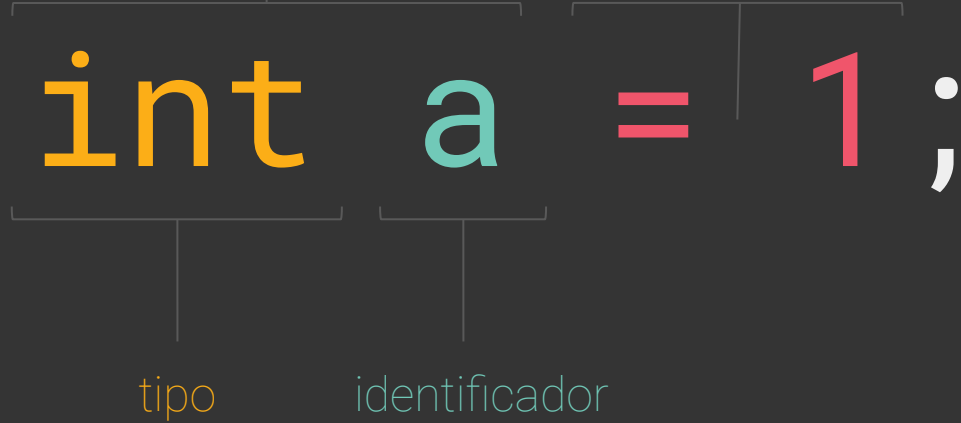


a = 1 ;

The diagram illustrates the assignment statement 'a = 1;'. A light blue 'a' is followed by a red '=', a red '1', and a white semicolon ';'. A thin grey line with a vertical segment and horizontal brackets is positioned above the right-hand side of the assignment, pointing to the text 'atribuição de valor' above it.

declaração  
de variável

atribuição  
de valor





Tipos complexos  
Formatos não  
existentes na  
linguagem

Compostos pelos tipos primitivos

**Pão** p ;

p = 🍞 ;

**Bola** b = 🏀 ;

# Classes

Formatos complexos não  
existentes na linguagem  
originalmente

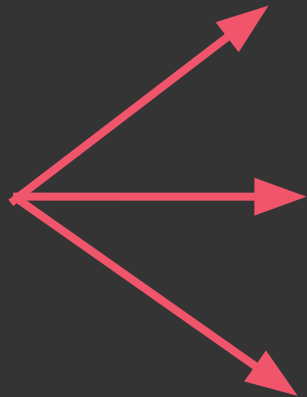






Receita  
Ingredientes, quantidades, modo  
de preparo, etc.

Pão  
Resultado, materialização  
da receita





Classe

Definição de dados e  
funções.

instanciação

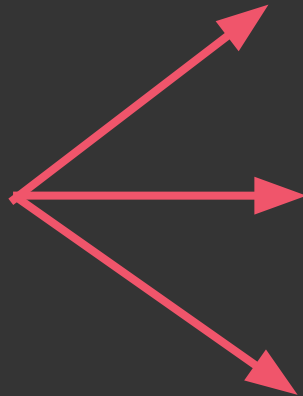


Objeto

Resultado, materialização  
da classe



Classe



objeto 1

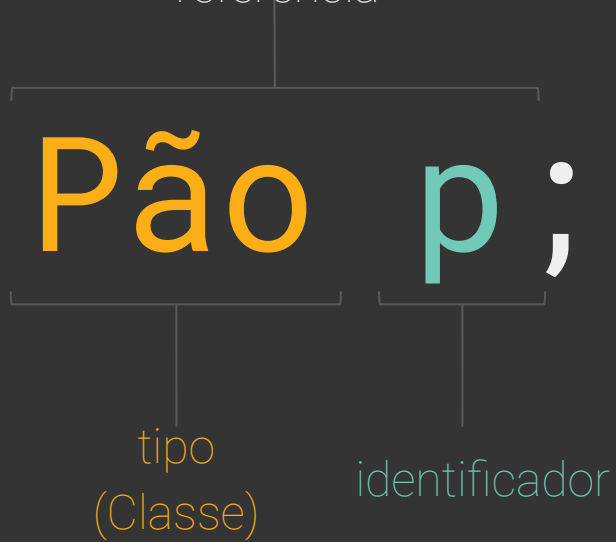


objeto 2

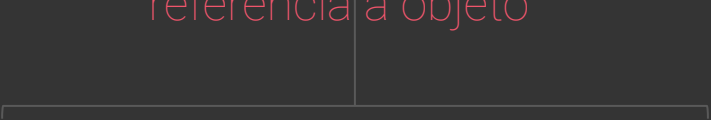


objeto 3

declaração de  
variável de  
referência



**instanciação** e  
atribuição de  
referência a objeto



```
p = new Pão();
```

declaração de  
variável de  
referência

**instanciação** e  
atribuição de  
referência a objeto

```
Pão p = new Pão();
```

tipo  
(Classe)

identificador

## Classe

Teste.java

```
class Teste {  
  
  
  
  
  
  
  
  
  
}
```

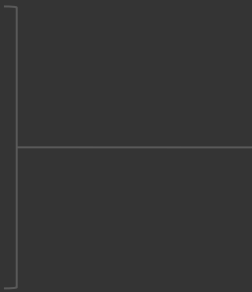
## Objeto(s)

Meu programa

```
Teste a = new Teste();  
  
Teste b, c, d;  
  
b = new Teste();  
c = new Teste();  
d = new Teste();
```



```
class Pessoa {  
    int idade;  
    double peso;  
}
```



Propriedades ou  
atributos

Pessoa.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

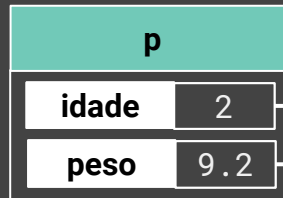
Meu programa

```
int a;  
a = 2;  
Pessoa p = new Pessoa();  
p = 2;  
p.idade = 2;  
p.peso = 9.2;
```



Memória do meu programa

a	2
---	---



?

Pessoa.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

Meu programa

```
Pessoa h = new Pessoa();  
Pessoa z = new Pessoa();  
  
h.idade = 2;  
z.idade = 70;  
h.peso = 9.2;  
z.peso = 58.1;
```



Memória do meu programa

h	
idade	2
peso	9.2

z	
idade	70
peso	58.1

```
class Retangulo {
```

```
    int base;
```

```
    int altura;
```

```
    int calculaArea() {...}
```

```
    void imprimeDimensoes() {...}
```

```
}
```

Propriedades ou  
atributos

Métodos

```
class Retangulo {  
    int base;  
    int altura;  
  
    int calculaArea() {  
        return base * altura;  
    }  
  
    void imprimeDimensoes() {  
        System.out.println("Base: " + base);  
        System.out.println("Altura: " + altura);  
    }  
}
```

Retangulo.java

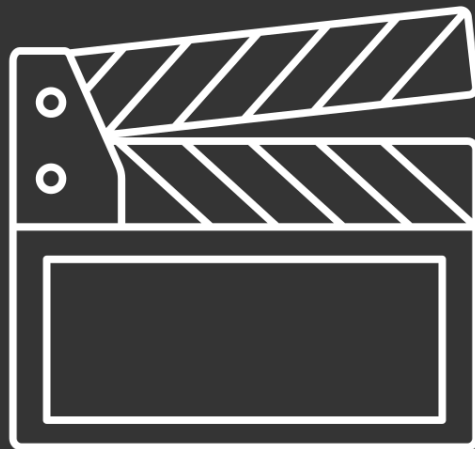
```
class Retangulo {  
    int base;  
    int altura;  
  
    int calculaArea() {  
        return base * altura;  
    }  
  
    void imprimeDimensoes() {  
        System.out.println("Base: " +  
            base);  
        System.out.println("Altura: " +  
            altura);  
    }  
}
```

Meu programa

```
Retangulo r = new Retangulo();  
r.base = 3;  
r.altura = 5;  
  
int area = r.calculaArea();  
System.out.println("Area: " +  
    area);  
  
r.imprimeDimensoes();
```

Terminal

```
Area: 15  
Base: 3  
Altura: 5
```



Objetos em  
ação na prática

Teste.java

```
class Pessoa {  
    int idade;  
    double peso;  
}  
  
class Teste {  
    public static void main(String args[]) {  
          
    }  
}
```

Códigos  
Variáveis, objetos, etc.



```
class Pessoa {  
    int idade;  
    double peso;  
}  
  
class Teste {  
    public static void main(String args[]) {  
        Pessoa a = new Pessoa();  
  
        a.idade = 10;  
        a.peso = 23.5;  
  
        System.out.println("Idade: " + a.idade);  
    }  
}
```

Teste.java

```
class Teste {  
    public static void main(String args[]) {  
        System.out.println("Hello, world!");  
    }  
}
```

Terminal

```
cd pasta_ou_esta_o_arquivo_java  
javac Teste.java  
java Teste  
Hello, World!  
|
```

javac Teste.java

Teste.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

java Abobrinha

```
class Abobrinha {  
    public static void main(String args[]) {  
        System.out.println("Hello, world!");  
    }  
}
```

Pessoa.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

Meu programa

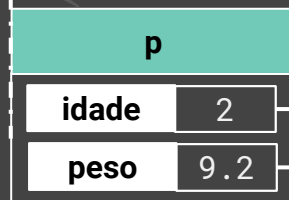
```
int a;  
a = 2;  
  
Pessoa p;  
p = new Pessoa();  
p.idade = 2;  
p.peso = 9.2;
```

Não é exatamente  
assim que as coisas  
funcionam...



Memória do meu programa

a	2
---	---



Pessoa.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

Meu programa

```
int a;  
a = 2;
```



Memória do meu programa

a

2

Pessoa.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

Meu programa

```
int a;  
a = 2;  
  
Pessoa p;  
p = new Pessoa();  
p.idade = 2;  
p.peso = 9.2;
```



Memória do meu programa

a	2
---	---

p	→
---	---



Pessoa.java

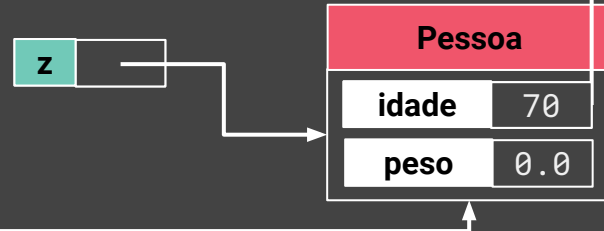
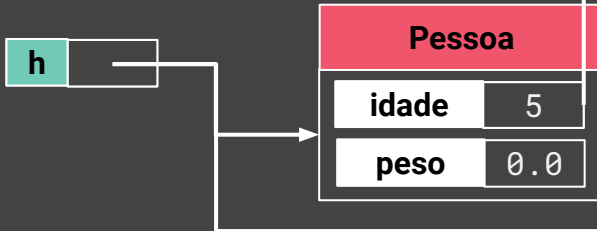
```
class Pessoa {  
    int idade;  
    double peso;  
}
```

Meu programa

```
Pessoa h = new Pessoa();  
Pessoa z = new Pessoa();  
  
h.idade = 5;  
z.idade = 70;  
  
h = z;  
// h.idade == 70
```



Memória do meu programa



Pessoa.java

```
class Pessoa {  
    int idade;  
    double peso;  
}
```

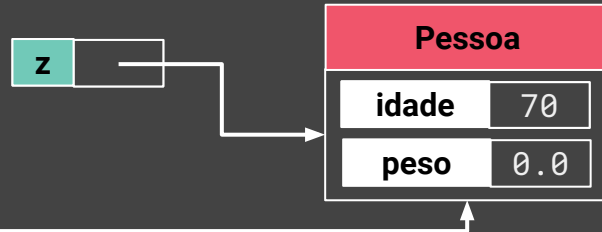
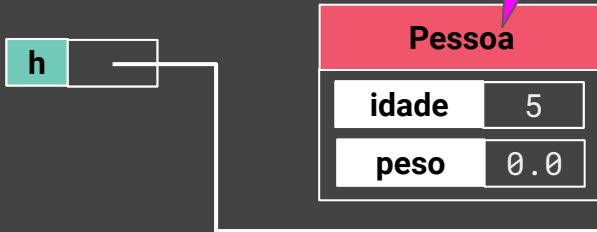
Nenhuma variável  
referencia este objeto!

Meu programa

```
Pessoa h = new Pessoa();  
Pessoa z = new Pessoa();  
  
h.idade = 5;  
z.idade = 70;  
  
h = z;  
// h.idade == 70
```



Memória do meu programa





Quadrado.java

```
class Quadrado {  
    int lado;  
    int x;  
    int y;  
}
```

Meu programa

```
Quadrado q1, q2;
```



Memória do meu programa

q1

?

q2

?

Quadrado.java

```
class Quadrado {  
    int lado;  
    int x;  
    int y;  
}
```

Meu programa

```
Quadrado q1, q2;  
q1 = new Quadrado();  
q1.x = 8;  
q1.y = 5;
```



Memória do meu programa

q1

**Quadrado**

lado 0

x 8

y 5

q2

?

Quadrado.java

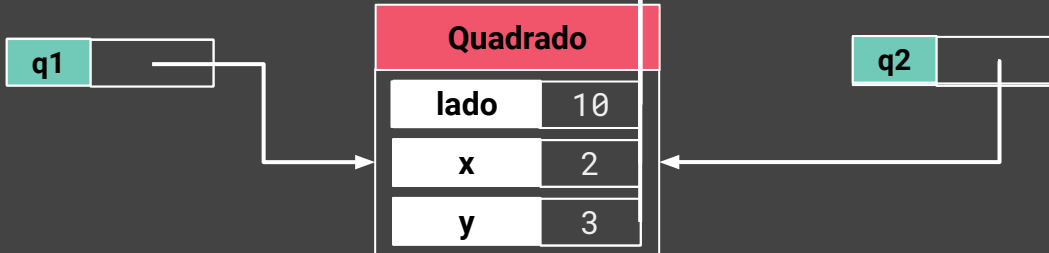
```
class Quadrado {  
    int lado;  
    int x;  
    int y;  
}
```

Meu programa

```
Quadrado q1, q2;  
q1 = new Quadrado();  
q1.x = 8;  
q1.y = 5;  
q2 = q1;  
q1.lado = 10;  
q2.x = 2;  
q2.y = 3;
```



Memória do meu programa



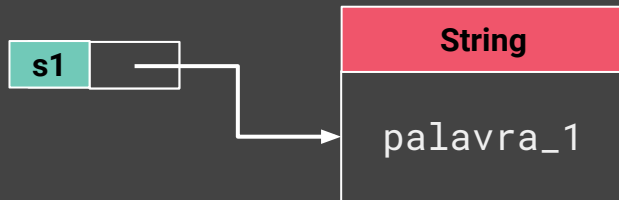
Meu programa

Terminal

```
int a = 1;  
String s1 = "palavra_" + a;  
// Forma simplificada de:  
// String s1 = new String("palavra_" + a);
```



Memória do meu programa



Meu programa

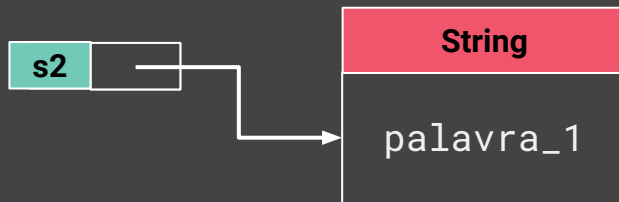
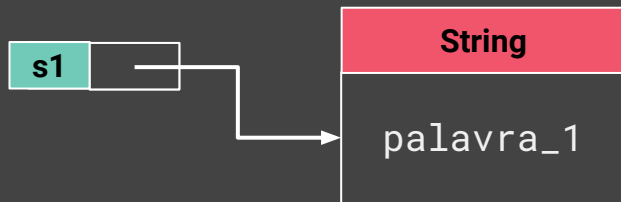
```
int a = 1;  
String s1 = "palavra_" + a;  
String s2 = "palavra_" + a;  
  
System.out.println("s1: " + s1);  
System.out.println("s2: " + s2);  
  
System.out.println("s1 == s2: " + (s1 == s2));  
System.out.println("s1.equals(s2): " + s1.equals(s2));
```

Terminal

```
s1: palavra_1  
s2: palavra_1  
s1 == s2: false  
s1.equals(s2):  
true
```



Memória do meu programa



# Representação gráfica de classes

Nome da classe

Carro

Cachorro

Atributos

marca  
modelo  
cor  
placa

raça  
nome  
idade

Métodos

frear()  
acelerar()  
trocarMarcha()  
darPartida()  
desligar()

latir()  
rolar()  
dormir()

# Créditos

Criado a partir de um conteúdo produzido por

Fernando Bevilacqua  
[fernando.bevilacqua@uffs.edu.br](mailto:fernando.bevilacqua@uffs.edu.br)

## Ícones

DinosoftLabs  
[flaticon.com/authors/dinosoftlabs](https://flaticon.com/authors/dinosoftlabs)

mangsaabguru  
[flaticon.com/authors/mangsaabguru](https://flaticon.com/authors/mangsaabguru)

Freepik  
[flaticon.com/authors/freepik](https://flaticon.com/authors/freepik)

Smashicons  
[flaticon.com/authors/smashicons](https://flaticon.com/authors/smashicons)

Vitaly Gorbachev  
[flaticon.com/authors/vitaly-gorbachev](https://flaticon.com/authors/vitaly-gorbachev)

