

POO – Classe - Objeto

Jose.wellington@uniceub.br

Calendário

◀ agosto de 2013 ▶

D	S	T	Q	Q	S	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
-	-	-	-	-	-	-

◀ setembro de 2013 ▶

D	S	T	Q	Q	S	S
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

◀ outubro de 2013 ▶

D	S	T	Q	Q	S	S
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2

◀ novembro de 2013 ▶

D	S	T	Q	Q	S	S
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

◀ dezembro de 2013 ▶

D	S	T	Q	Q	S	S
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Agenda

- **Classe**
- **Objeto**
- **Exercício**

Classe

Objeto

- Um objeto é capaz de armazenar estados através de seus atributos.
- Exemplo de objetos da classe Humanos: João, José, Maria

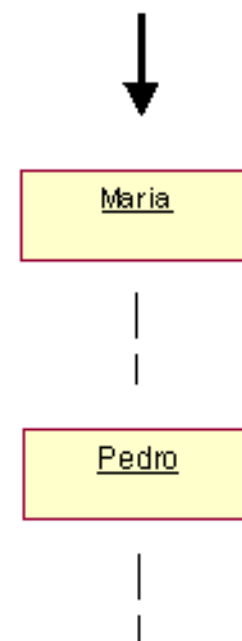
Classe



A
T
R
I
B
U
T
O
S

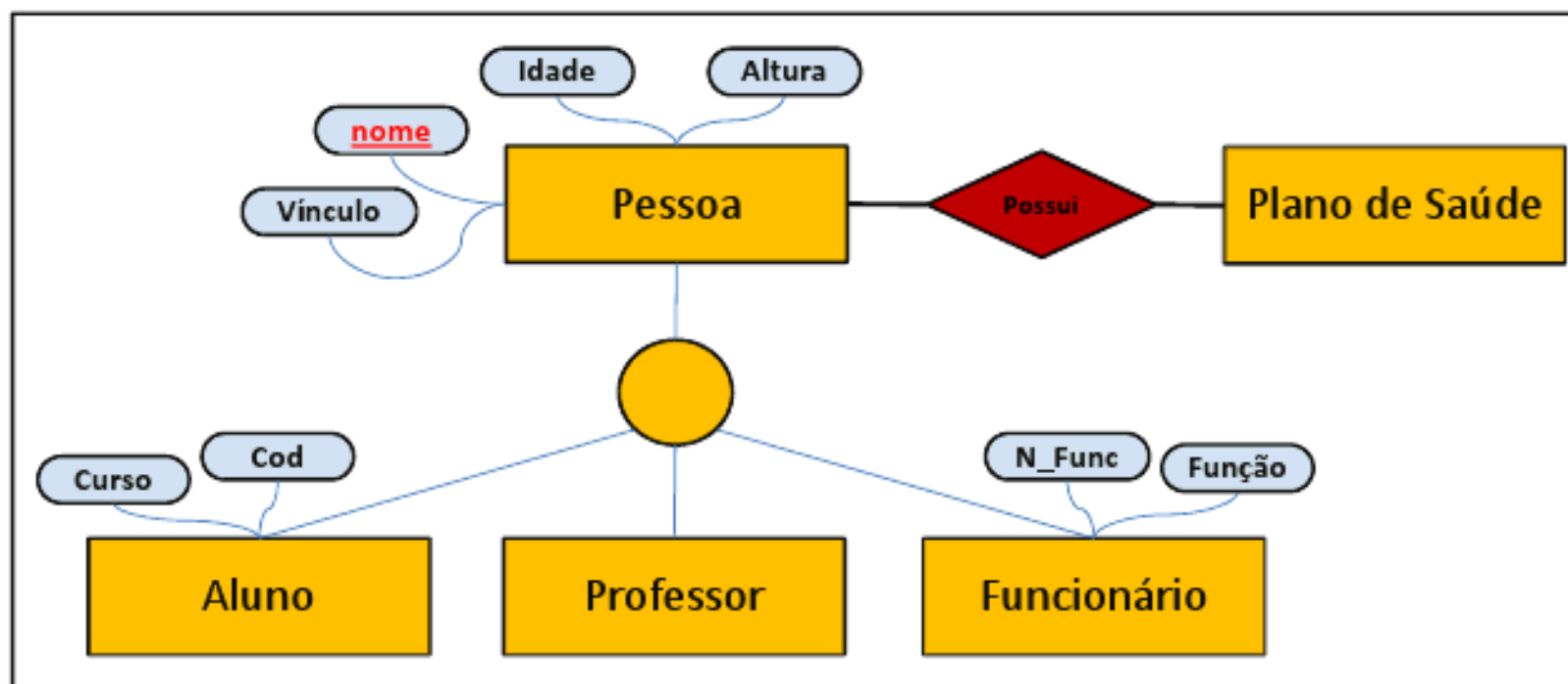
M
É
T
O
D
O
S

Objetos



Herança

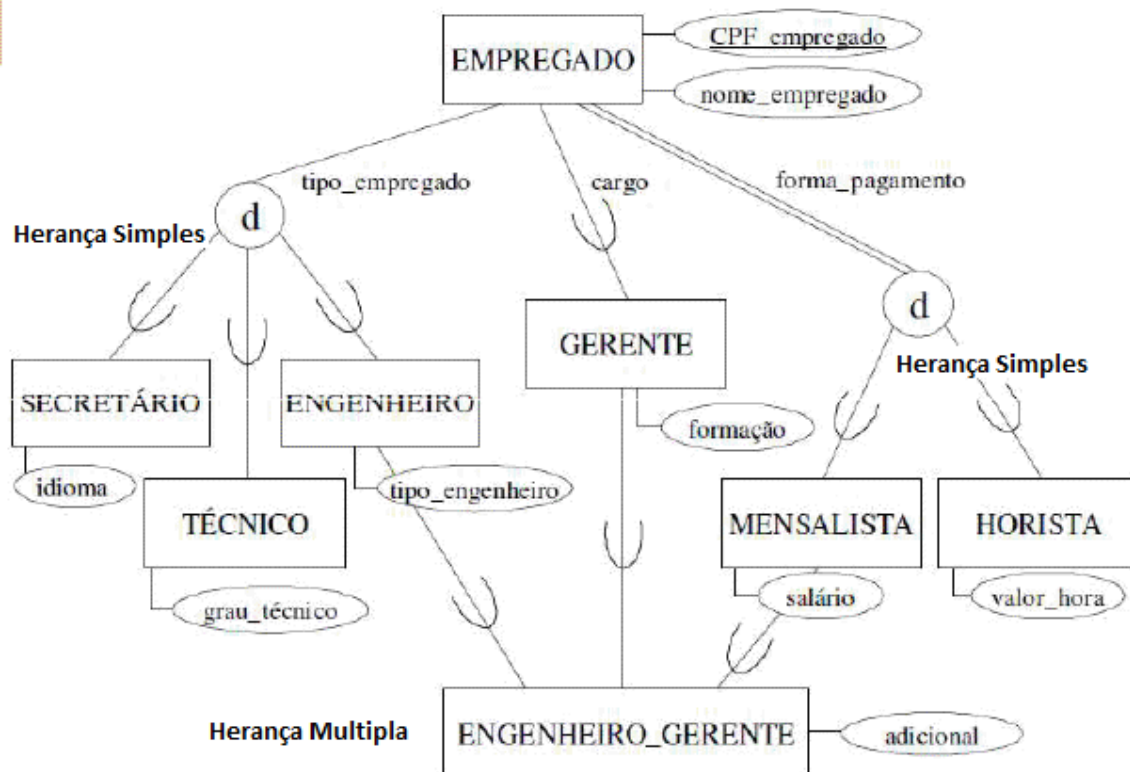
- Conjuntos de entidades mais específicos herdam **todos os atributos** do conjunto de entidades mais genérico;



Herança

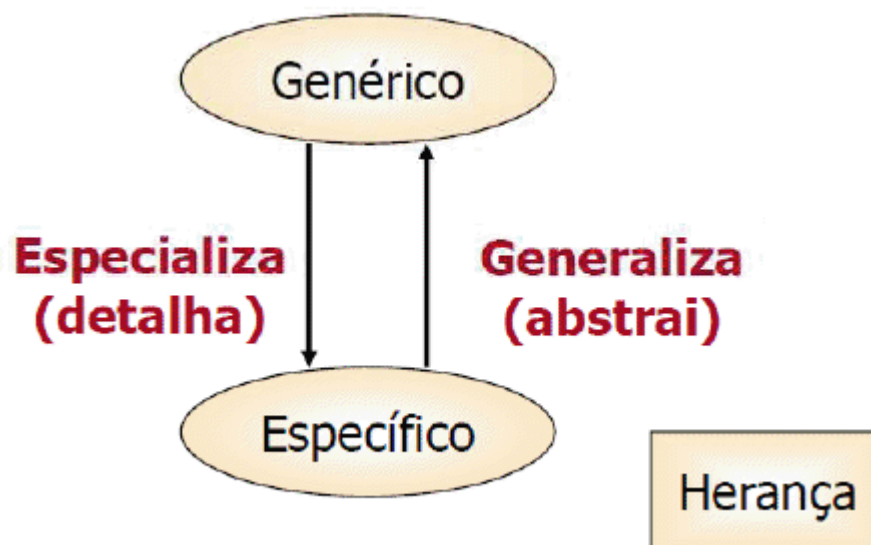
■ Herança simples

- cada **subclasse** herda de apenas informações de uma **Superclasse**.



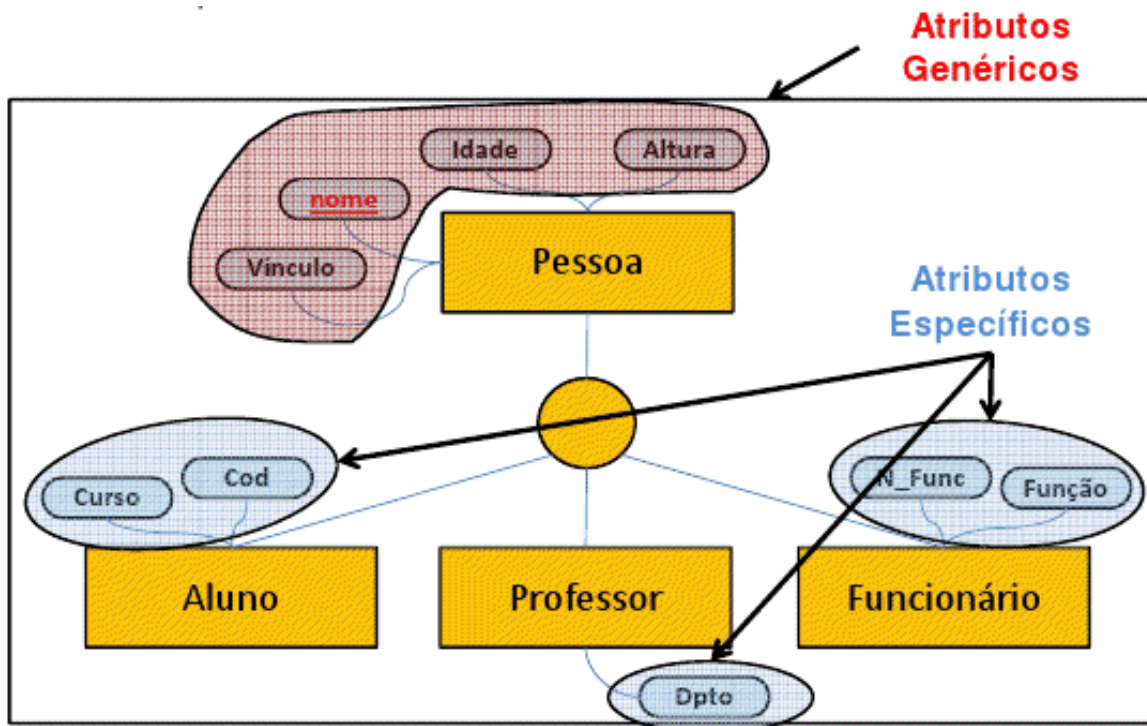
Generalização e Especialização

Dado que é associado
a cada ocorrência de
uma entidade ou de
um relacionamento



Especialização / Generalização

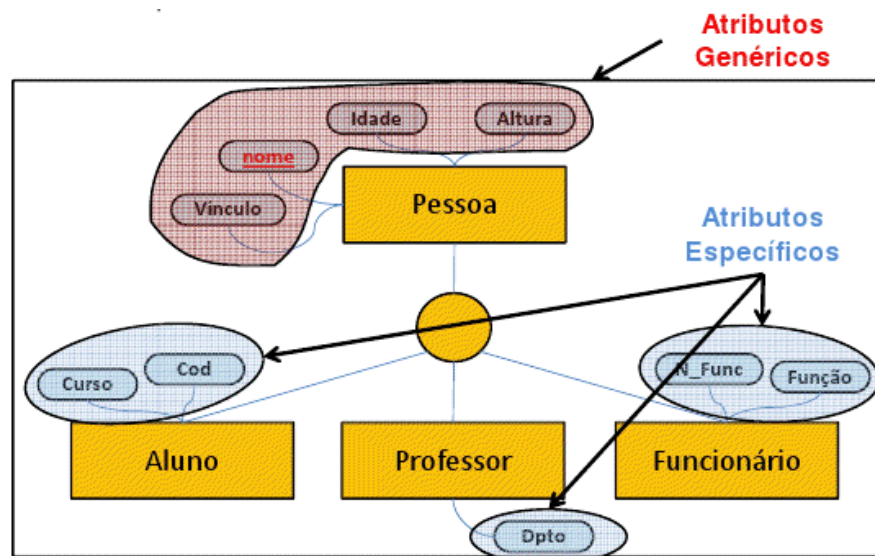
- **Especialização:**
- Resultado da separação de um tipo-entidade de **nível mais alto (superclasse)**, formando vários **tipos-entidade de nível mais baixo (subclasse)**.



Especialização / Generalização

■ Generalização

- Pode ser visto como o processo inverso à especialização;

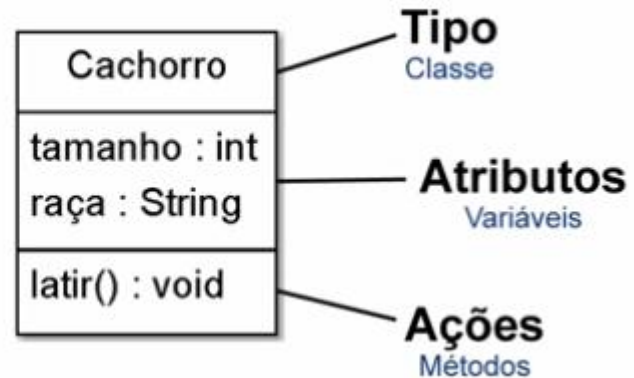


Programação Orientado a Objeto

Orientação a Objetos

- Propósitos da Programação Orientada a Objetos
 - ⇒ Prover mecanismos para visualizar a complexidade do desenvolvimento de software da mesma forma que visualizamos a complexidade do mundo real;
 - ⇒ Acelerar o desenvolvimento de softwares com base na modularidade e acoplamento;
 - ⇒ Melhorar a qualidade do software desenvolvido.

Classes - conceitos



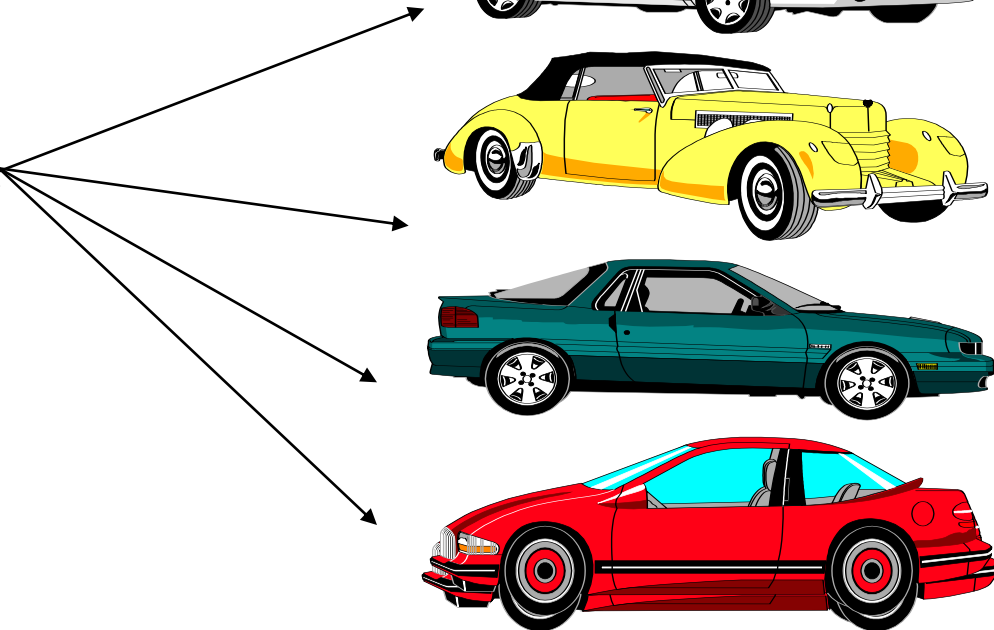
- definem a estrutura e o comportamento de um tipo de objeto;
- atuam como templates;
- permitem a instanciação de um ou mais objetos de um mesmo tipo.

Classes

Visão do mundo real:

- moldes para criação de objetos;
- especificam propriedades e ações em comum a todos seus objetos.

Classe
Automóvel



Classes

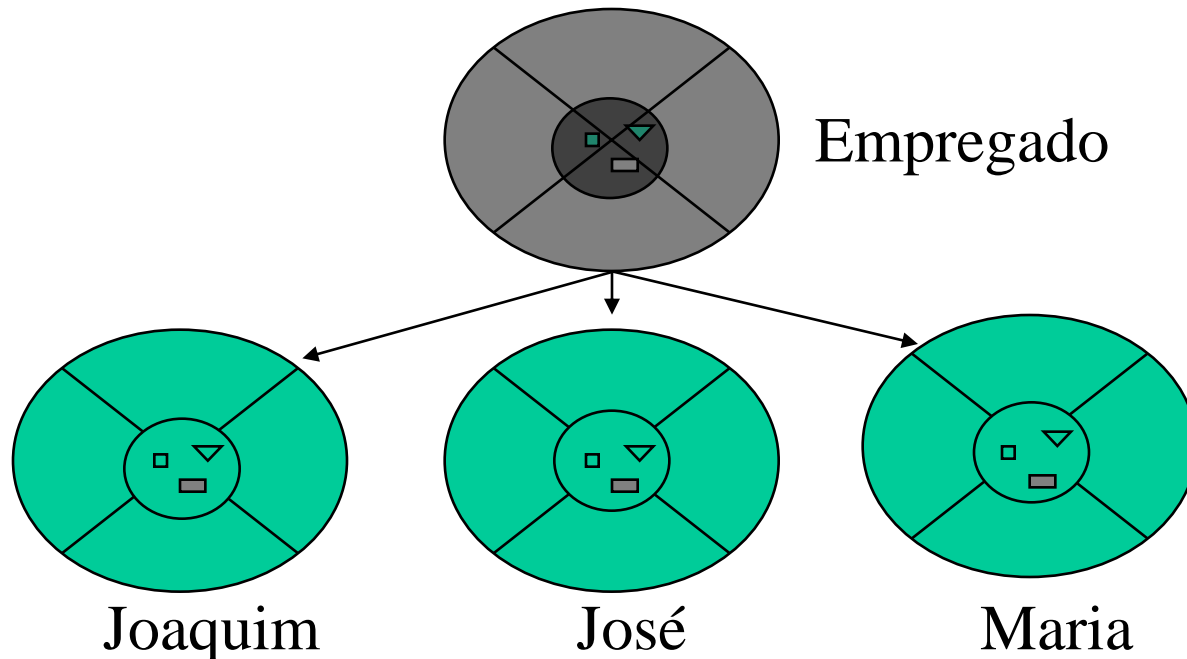
Classes - Visão em POO:

Um protótipo que define os atributos e os métodos comuns a todos objetos de um determinado tipo e da própria classe, com a finalidade de servir como molde para a criação de objetos.

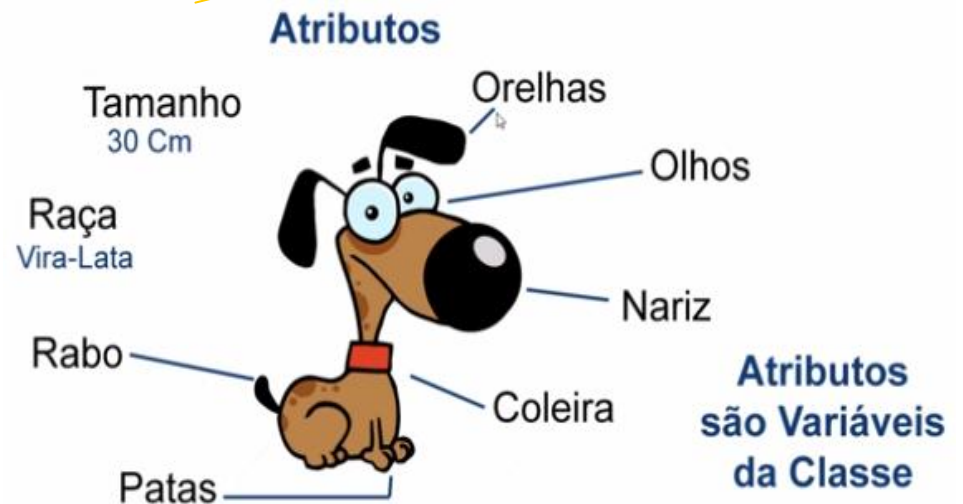
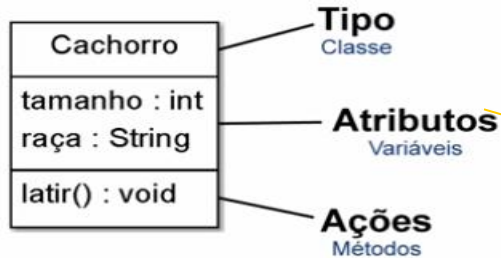
Classe

Empregado

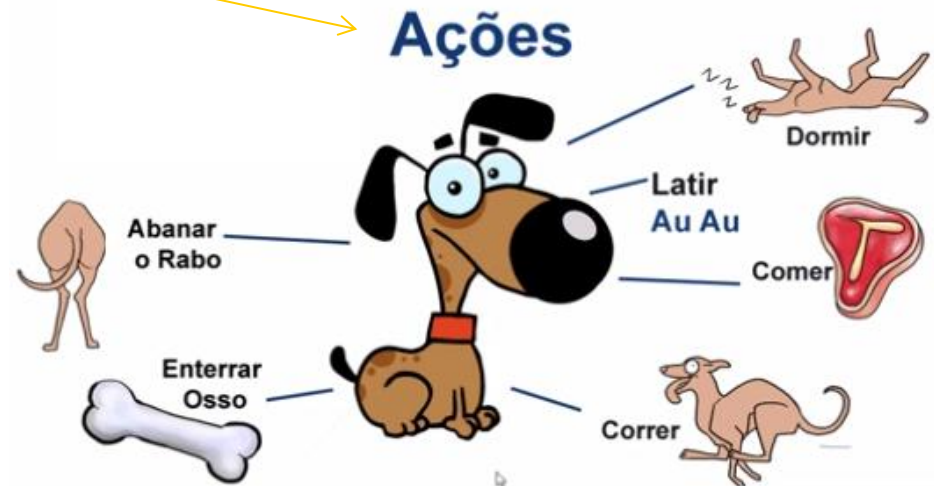
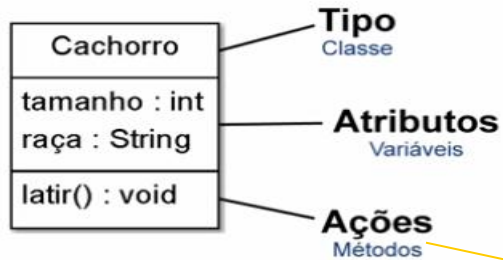
Objetos



Classes - Atributos

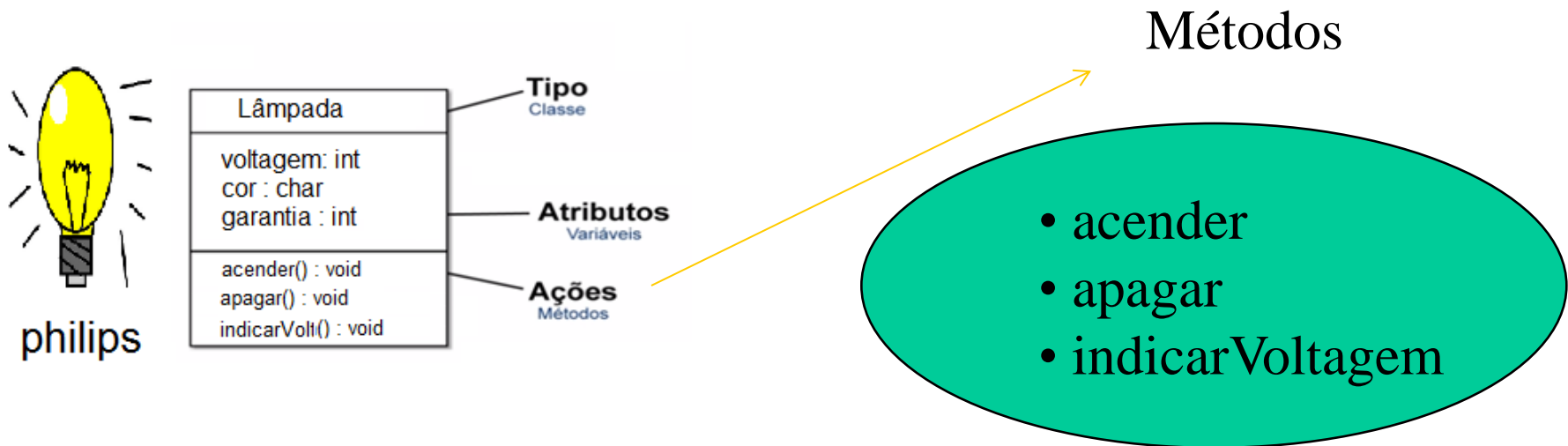


Classes - Métodos

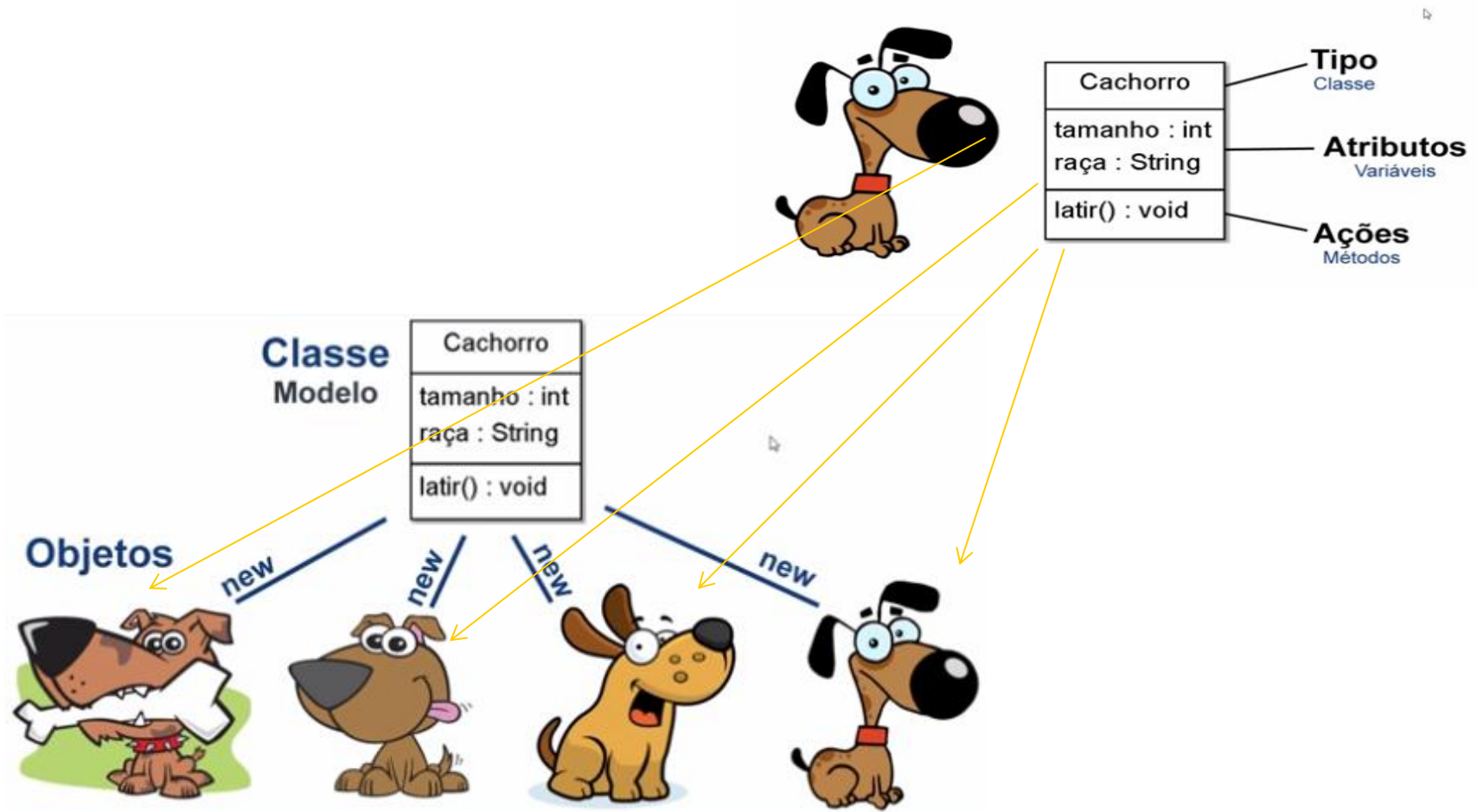


Classes - Métodos

- Operações que manipulam o estado do objeto.
- Podem fazer parte da interface do objeto ou realizar uma função interna.



Objetos – Instância de Objeto



Objetos – Instância de Objeto

Visão do mundo real:

- moldes para criação de objetos;
- especificam propriedades e ações em comum a todos seus objetos.

Objeto

Classe

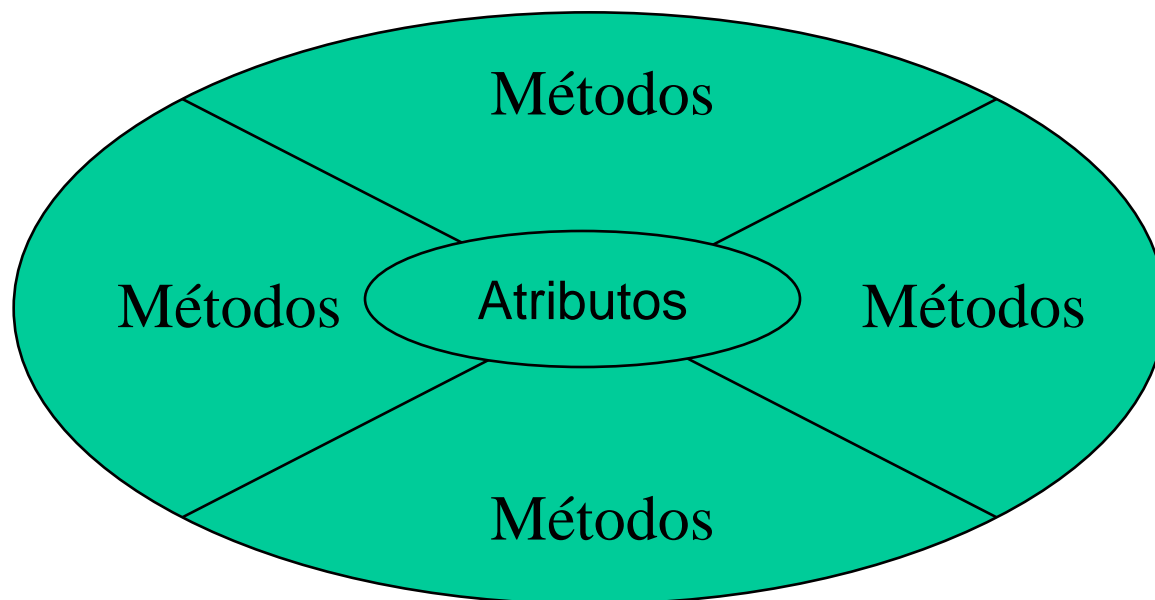
Automóvel



Objetos – Instância de Objeto

Visão em POO:

Um conjunto distinto de atributos e métodos que representam entidades do mundo real, implementados por software e gerados (instanciados) através de classes.

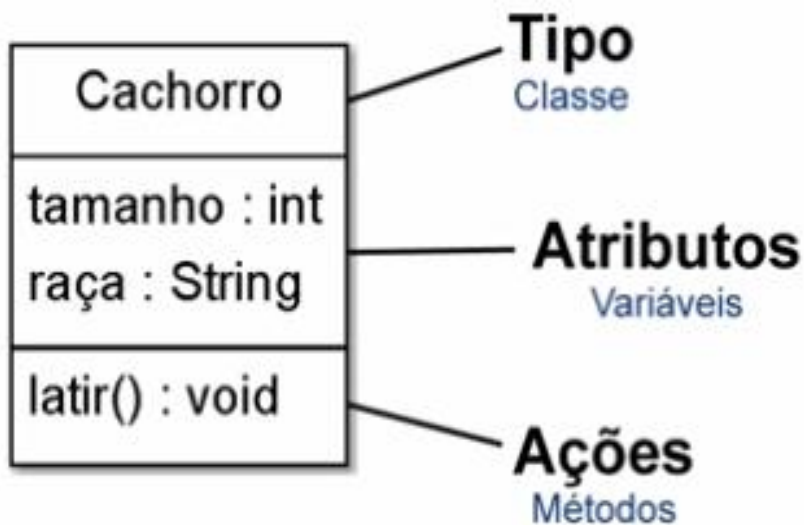


Criando Objetos em Java

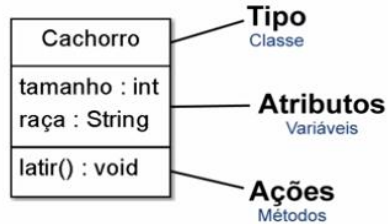
- Objetos são criados através da declaração `new`, seguida de um método construtor. Exemplo:

```
Automovel gol = new Automovel();
```
- Uma classe pode ter construtores especializados ou somente o construtor default (gerado pelo compilador).

Criando Objetos em Java



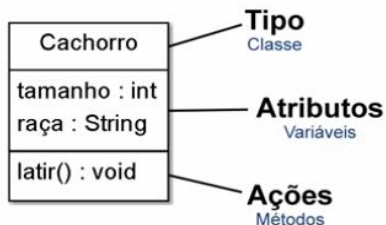
Criando Objetos em Java



```
package canil;
```

```
class Cachorro {  
    int tamanho;  
    String raca;  
  
    void latir() {  
        System.out.println("Au Au Au!!!");  
    }  
}
```


Criando Objetos em Java



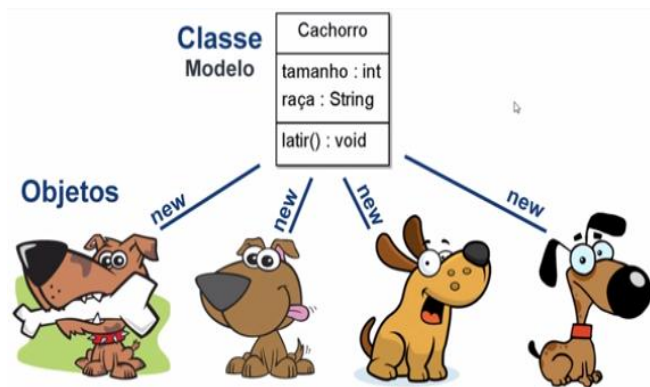
```
public class Canil {  
  
    public static void main(String[] args) {
```

```
        Cachorro pitbul = new Cachorro();
```

```
        pitbul.raça = "Pitbull";  
        pitbul.tamanho = 40;  
        pitbul.latir();
```

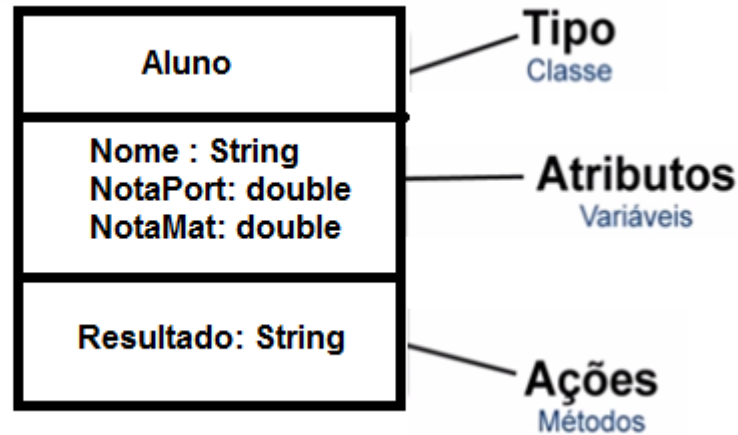
```
        Cachorro viralata = new Cachorro();  
        viralata.raça = "Viralata";  
        viralata.tamanho=50;  
        viralata.latir();
```

```
    }  
}
```



Exercício - Exemplo

Criando Objetos em Java



- Maria
- Nota Mat = 8
- Nota Port = 7

- Joao
- Nota Mat = 4
- Nota Port = 3

- Carlos
- Nota Mat = 9
- Nota Port = 8

```
Maria Aprovado 7.0  
João Reprovado 3.0  
Carlos Aprovado 8.0
```

Agenda

- **Classe**
- **Objeto**
- **Exercício**

■ Jose.wellington@uniceub.br