



El futuro digital  
es de todos

MinTIC

so  
bre  
carga

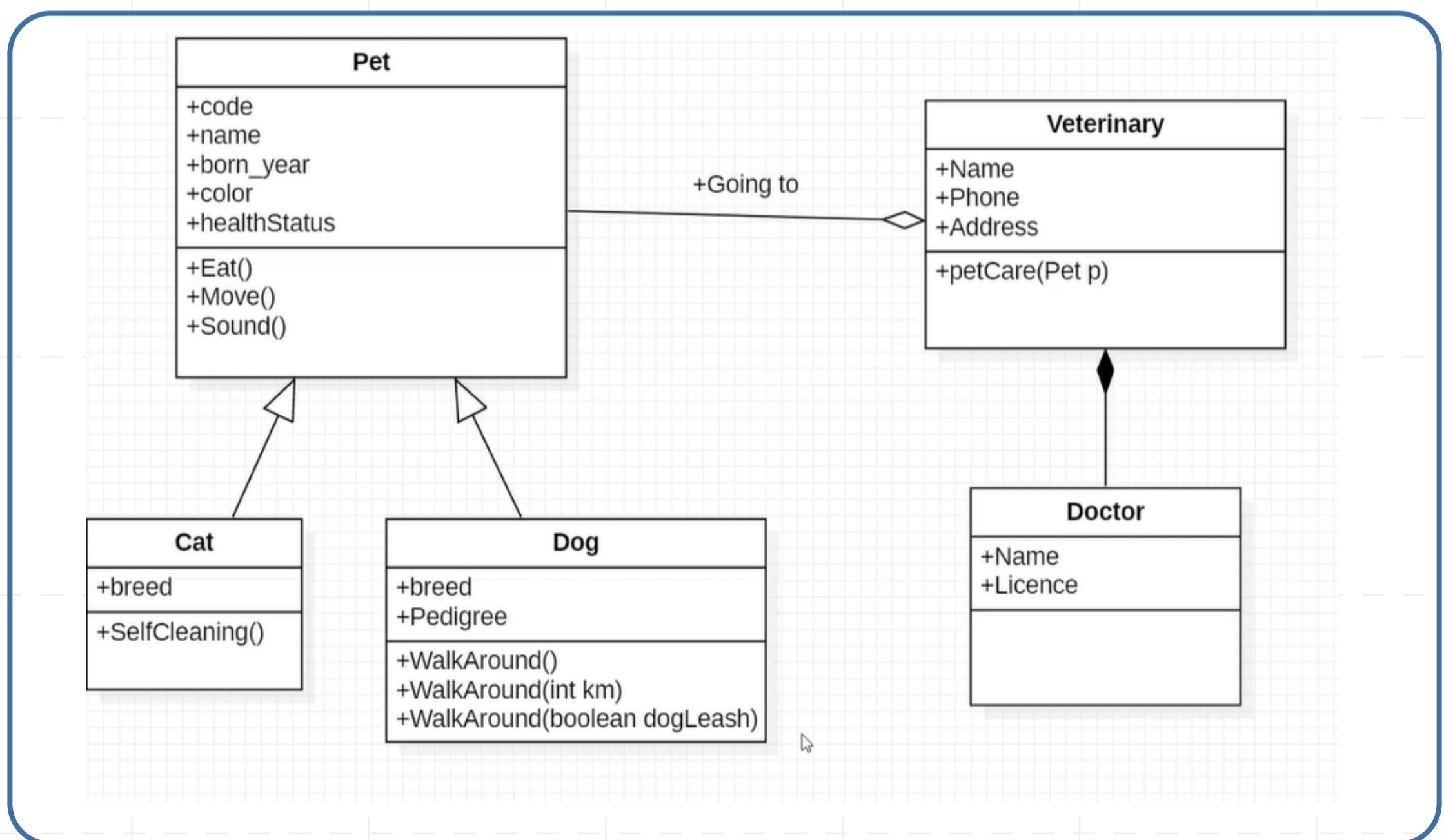


Universidad de Caldas

# Hola:

El **polimorfismo de sobrecarga** es un conjunto de métodos que tienen el mismo nombre pero se diferencian en el número y/o tipo de parámetros que recibe. Vamos a verlo de manera práctica para aclarar este concepto.

Video con el ejemplo de polimorfismo por sobrecarga (ejemplo con el método pasear recibiendo o no parámetros, por ejemplo, collar, km, corriendo o caminando)



```

20
21
22
23
24     public void WalkAround(){
25         System.out.println("El perro " + super.getName() + " está caminando.");
26     }
27
28     public void WalkAround(int km){
29         System.out.println("El perro " + super.getName() + " está caminando " + km + " km");
30     }
31
32     public void WalkAround(boolean dogLeash){
33         String hasLeash = dogLeash ? "con correa" : "sin correa";
34         System.out.println("El perro " + super.getName() + " está caminando " + hasLeash);
35     }
36
37

```

**WalkAround - Navigator**

- Members
- dsDog :: dsPet
  - dsDog(String breed, boolean pedigree, Sound[] cdPet)
  - WalkAround()
  - WalkAround(int km)
  - WalkAround(boolean dogLeash)
  - getBreed(): String
  - isPedigree(): boolean
  - setBreed(String breed)
  - setPedigree(boolean pedigree)
  - breed : String
  - pedigree : boolean

**Output - PetManager (run)**

```

7
8 package petmanager;
9
10 import Classes.clsDog;
11
12 /**
13  * @author USUARIO
14 */
15 public class Polymorphism3 {
16     public static void main(String[] args) {
17         // Instancias de las clases hijas
18         clsDog dog = new clsDog("criollo", false, "001", "Firulais", 2013, "Negro", "Sano");
19
20         dog.WalkAround();
21         dog.WalkAround(5);
22         dog.WalkAround(true);
23     }
24

```

**main - Navigator**

- Members
- Polymorphism3
  - main(String[] args)

**Output - PetManager (run)**

```

run:
El perro Firulais está caminando.
El perro Firulais está caminando 5 km
El perro Firulais está caminando con correa
BUILD SUCCESSFUL (total time: 0 seconds)

```

Una vez comprendidos los diferentes tipos de **polimorfismo** que se encuentran en la programación orientada a objetos, los invito a practicar desarrollando el reto de la semana con base en el conocimiento adquirido. También pueden profundizar en esta temática revisando el material adicional de la semana.



Universidad de Caldas