

CURSOS  
INTERSEMESTRALES



PROTECO

POO

Programación  
Orientada a  
Objetos



#### Atributos:

Nombre: Mamut  
Pelo café y lacio.  
Altura: 25 cm  
Longitud: 35 cm  
Extras: Chalequito

#### Métodos:

Comer  
Dormir  
Ladrar



PROTECO



#### Atributos:

Nombre: Jett  
Pelo: Blanco y chino.  
Altura: 20 cm  
Longitud: 90 cm  
Extras: Le gusta la sopa

#### Métodos:

Comer  
Dormir  
Ladrar



Atributos:

Nombre: Perro generico

Pelo:

Altura:

Longitud:

Extras:

Métodos:

Comer

Dormir

Ladrar

Cazar



Atributos:

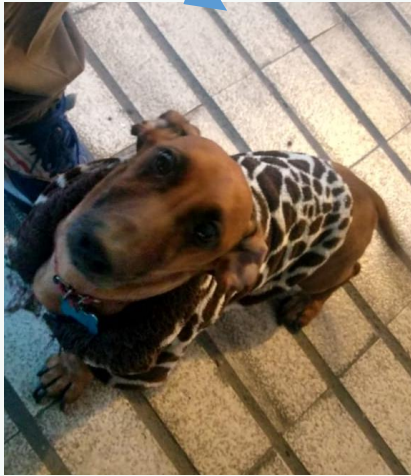
Pelo: Multicolor.  
Altura: 81 cm  
Longitud: 2 m  
Extras: Amigo de los Stark

Métodos:

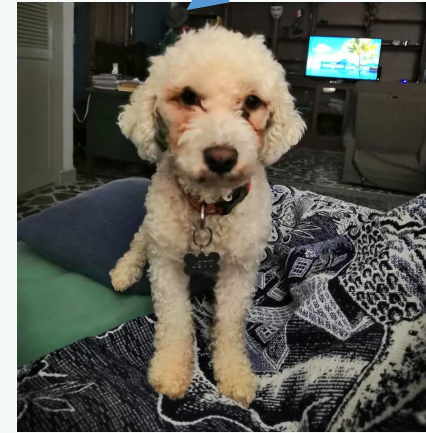
Comer  
Dormir  
Ladran  
Cazar



## Clase: Perro

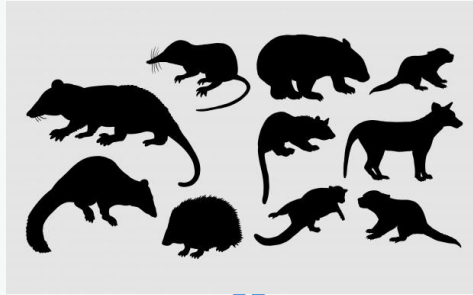


Objeto: Mamut  
Objeto: Mamut



Objeto: Jett

Clase: Mamífero



Clase: Perro



Clase: Lobo



PROTECO

# ¿Qué es?

Inicio en los 60's.

Entre las características principales se encuentra la representación de unidades mínimas usando entidades o objetos(abstracción del mundo real).

Los cuales se comunican entre si para solucionar los problemas.



PROTECO

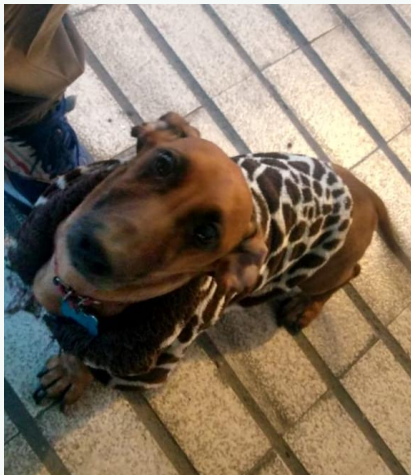




# Objetos:

Objetos: Instancia de una clase. Tiene un conjunto de acciones y un conjunto de características que lo diferencian con los demás

En otras palabras: Estado + comportamiento



Objeto: Mamut



Objeto: Jeet

# Clases:

Clases: Es una plantilla o plano para la creación de objetos genérico, esta nos dice cuales deben ser las características y acciones que deben de tener cada objeto..



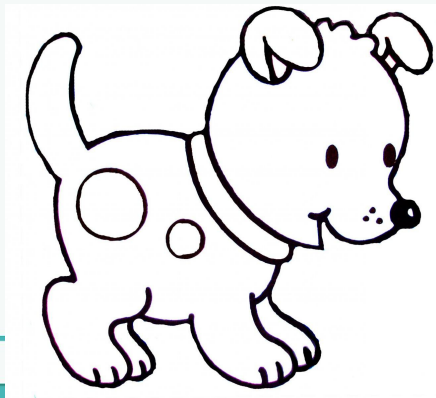
Clase: Perro

# Atributos::

Atributos: Características de un objeto, pueden pertenecer a la clase o al objeto en cuestión.

Ejemplo:

Número de patas de los perros: Pertenece a la clase perro, es general.  
Color de pelo: Depende del objeto.



# Métodos:

Acciones que puede realizar el objeto, son funciones definidas para cierto objeto. Existen de clase (Estáticos) y de instancia.



Método dormir



Método ladrar

# ¿Método o función?

Una diferencia muy clara entre un método en un objeto y una función cualquiera es que el método pertenece a cierta clase u objeto.



PROTECO



# ¿Y en Python?

La syntaxis de python nos permite abstraer de manera mas sencilla nuestras ideas a Código.

Clases: Molde del objeto `class ClassName(object):`

Atributos: Caracteristicas del objeto, pueden ser diferentes tipos de dato

```
class ClassName(object):  
    Atributo_1 = |  
    Atributo_2 =
```

Métodos: Acciones de la clase

```
class ClassName(object):  
    Atributo_1 =  
    Atributo_2 =  
    def metodo_1():
```



Método Constructor: Es el primero que se ejecuta

```
class ClassName(object):  
    Atributo_1 =  
    Atributo_2 =  
    def metodo_1():  
        LO_QUE_VAYA_DENTRO  
  
    def __init__(self, arg):  
        super(ClassName, self).__init__()  
        self.arg = arg
```



# "Self"

El uso de "self" en python es una convención, es una buena práctica y algunos libros la toman como regla (aunque no necesariamente)

¿Que significa?

El parámetro self se refiere al objeto instanciado de esa clase sobre el cual se está invocando dicho método. Es decir, el objeto que usaste para llamar al método

