



# Programación Orientada a Objetos



# Creación de clases



## Índice

- Clases y objetos
- Creación de una clase y objetos de la misma
- Constructores
- Destrucción
- Atributos implícitos



# Clases y objetos

- Una clase define el comportamiento de un tipo de objeto, a través de la definición de **métodos y atributos** asociados a los mismos
- Los atributos se definen mediante **variables**, accesibles directamente desde el exterior del objeto
- Los métodos se definen mediante **funciones**
- El aspecto de una clase es:

```
class NombreClase:  
    #miembros clase
```



# Creación de una clase y objetos de la misma

- Todos los métodos incluyen al menos el parámetro ***self*** para acceder a los atributos de la clase. Este parámetro se pasa implícitamente en la llamada al método:

```
class Punto:  
    x=0  
    y=0  
    dimension=2  
    def imprimir(self):  
        print(self.x,self.y)
```

- Los objetos se referencian a través de variables:

```
pt=Punto() #creación de un objeto Punto
```

- Y se utiliza el operador "." para acceder a sus miembros:

```
pt.x=10  
pt.y=5  
pt.imprimir()
```



# Constructores

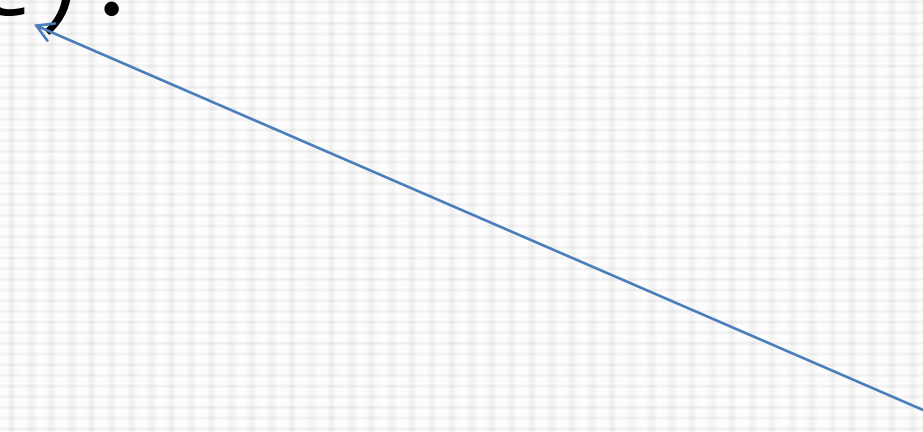
- Funciones que se ejecutan al crear un objeto de la clase
- Se definen con la función especial `__init__`, e incluye al menos el parámetro `self`:

```
def __init__(self):  
    print("constructor")
```

- En python no existe la sobrecarga de constructores, pero se puede conseguir un efecto similar definiendo parámetros opcionales:

```
class MiClase:  
    def __init__(self, a=None):  
        print("constructor")
```

```
m1=MiClase()  
m2=MiClase(10)
```



El valor por defecto del parámetro "a" es None



# Destruidores

- Funciones que se ejecutan antes de que un objeto sea destruido por el Garbage Collector
- Se definen con la función especial `__del__` y define el parámetro `self`:

```
def __del__(self):  
    print("destructor")
```

- El objeto será destruido por el GC cuando **no sea referenciado por ninguna variable**:

```
class MiClase:  
    def __del__(self):  
        print("se destruye")  
o1=MiClase()  
o2=o1  
del o1  
del o2 #imprime se destruye
```



# Atributos implícitos

- Todas las clases de Python tienen una serie de atributos implícitos:
  - `__doc__`. Documentación de la clase
  - `__dict__`. Diccionario que contiene el espacio de nombres de la clase
  - `__name__`. Nombre de la clase
  - `__module__`. Nombre del módulo en el que se encuentra la clase
  - `__base__`. Tupla con las clases base
- Además de utilizar directamente su nombre, se puede acceder a cualquier atributo de un objeto mediante las funciones *setattr* y *getattr*:

```
pt=Punto()  
setattr(pt,"x",10)  
print(getattr(pt,"__name__") #imprime Punto
```