

P00: Programación orientada a objetos (Parte 2)

Programación 2



Repasando conceptos

- Clases
- Objeto/Instancia/Ejemplar de clase
- Modularización
- Encapsulación

Construcción de código: clase y atributos o propiedades

```
class Auto():  
    largoChasis=250  
    anchoChasis=120  
    ruedas=4  
    enmarcha=False
```

Construcción de código: métodos o comportamientos

```
class Auto():  
    largoChasis=250  
    anchoChasis=120  
    ruedas=4  
    enmarcha=False  
  
    def arrancar(self):  
        (...)
```

Construcción de código: objetos o instancias de clase

```
class Auto():  
    largoChasis=250  
    anchoChasis=120  
    ruedas=4  
    enmarcha=False  
  
    def arrancar(self):  
        (...)
```

miAuto=Auto() -----> Instanciamos la clase Auto

Construcción de código: nomenclatura del punto

```
class Auto():
    largoChasis=250
    anchoChasis=120
    ruedas=4
    enmarcha=False

    def arrancar(self):
        (...)

miAuto=Auto()
print(miAuto.largoChasis)                # 250
print("el auto tiene ", miAuto.ruedas, " ruedas")    # El auto tiene 4 ruedas

miAuto.arrancar()
```

Construcción de código: nomenclatura del punto

```
class Auto():  
    largoChasis=250  
    anchoChasis=120  
    ruedas=4  
    enmarcha=False  
  
    def arrancar(self):  
        self.enmarcha=True
```


```
miAuto=Auto()
```

```
print(miAuto.enmarcha)           # False
```

```
miAuto.arrancar()
```

```
print(miAuto.enmarcha)           # True
```

Al ejecutarse el método self se convierte en miAuto y luego el cuerpo de la función representa lo mismo que si decimos:
miAuto.enmarcha=True



Construcción de código: nomenclatura del punto

```
class Auto():
    largoChasis=250
    anchoChasis=120
    ruedas=4
    enmarcha=False

    def arrancar(self):
        self.enmarcha=True

    def estado(self):
        if(self.enmarcha):
            return "El auto está en marcha"
        else:
            return "El auto está detenido"

miAuto=Auto()

print(miAuto.estado())
miAuto.arrancar()
print(miAuto.estado())
```

Para finalizar tenemos que tener en claro algunas cosas:

- La cantidad de propiedades o atributos que tiene nuestra clase Auto es 4.
- La cantidad de comportamientos o métodos que tiene nuestra clase Auto es 2.

El auto está detenido

El auto está en marcha