Métodos

Contenidos

Introducción

al indice

Son funciones que podemos definir dentro de las clases. Estas funciones cambiarán el estado de algún atributo o realizarán cálculos que nos sirvan de output. Un ejemplo sencillo puede ser, un método de la clase coche que saque la potencia en kilovatios, en vez de en caballos. O si tiene un estado de mantenimiento (ITV pasada o no), que modifique ese estado.

El constructor es un tipo de método. La diferencia con el resto de métodos radica en su nombre, `__init__`. La sintaxis para definir los métodos es como si fuese una función. Y luego para llamar al método se utiliza `objeto.metodo(argumentos_metodo)`. Esto ya lo hemos usado anteriormente, cuando haciamos un `string.lower()`, simplemente llamábamos al método `lower()`, que no requería de argumentos, de la clase *string*.

In [13]:
```
class Coche:
    ruedas = 4

    def __init__(self, marca_coche, num_puertas = 4):
        self.marca = marca_coche
        self.puertas = num_puertas
    def caracteristicas(self):
        return "Marca: " + self.marca +", Puertas:" + str(self.puertas)
```

In [14]:
```
ford_ka = Coche("Ford")
ford_ka.caracteristicas()
```

Out[14]:
```
'Marca: Ford, Puertas:4'
```

Fíjate que para llamar a las ruedas se usa `self`, a pesar de que no lo habíamos metido en el constructor. Así evitamos llamar a otra variable del programa que se llame *ruedas*. Nos aseguramos que son las ruedas de ese coche con el `self`.

In [23]:
```
class Coche:
    ruedas = 4

    def __init__(self, marca_coche, precio_coche, num_puertas = 4):
        self.marca = marca_coche
        self.puertas = num_puertas
        self.precio = precio_coche

    def caracteristicas(self):
        return "Marca: " + self.marca +", Puertas:" + str(self.puertas)

    def precio_actual(self, agnos):
        if agnos <= 5:
            return self.precio * 0.7
        elif agnos > 5 and agnos < 10:
            return self.precio * 0.5
        else:
            return self.precio * 0.3
```

In [24]:
```
jeep_cherokee = Coche("Jeep", 35000)
jeep_cherokee.precio_actual(8)
```

Out[24]:
```
17500.0
```

In [ ]:
```

```