



Metaprogramación

Manipulación de una clase

Índice

- Añadir miembros de forma dinámica
- Ejemplo
- Creación de clases con type

Añadir miembros de forma dinámica

- A través de la función `setattr` se pueden añadir nuevos atributos y métodos a una clase de forma dinámica
- La sintaxis de la función es:

`setattr(clase,nombre_elemento,valor)`

- Donde:
 - `clase`. Nombre de la clase a la que se va a añadir el elemento
 - `nombre_elemento`. Nombre del atributo o método
 - `valor`. Valor del atributo o referencia a la función en caso de método

Ejemplo

- Dada la siguiente clase

```
class Test:  
    pass
```

- Se añaden de forma dinámica un atributo y un método:

```
setattr(Test, "dato", 100)  
def func(s):  
    print("metodo nuevo")  
setattr(Test, "metodo", func)  
obj=Test()  
print(obj.dato)#100  
obj.metodo()# imprime metodo nuevo  
print(Test.__dict__)
```

```
100  
metodo nuevo  
{'__module__': '__main__', '__dict__': <attribute '__dict__' of 'Test' objects>, '__weakref__': <attribute '__weakref__' of 'Test' objects>,  
  '__doc__': None, 'dato': 100, 'metodo': <function func at 0x000002223FD28488>}
```


Creación de clases con type

- Además de obtener el tipo de un objeto, la función type sirve para crear clases de forma dinámica
- La sintaxis de la función es:

`type(clase, bases, dict)`



- Donde:
 - clase. Nombre de la clase que se va a crear
 - bases. Tupla con las clases que hereda la nueva clase
 - dict. Diccionario con los miembros de la clase