SESIÓN 21 · MÓDULO 2

Herencia múltiple, MRO e introspección

Saber qué método se ejecuta cuando hay varias clases madre, y preguntar a un objeto por sus atributos y su clase.

Antonio Chanes Espigares · CC BY-NC-ND 4.0

Hoy

0:00	Corrección de la tarea 20.
0:15	Situación: preguntar una duda: primero al compañero de mesa, luego al profesor de la materia, luego al tutor.
0:25	**Predice**. El MRO se dibuja como un rombo.
0:45	Construcción: herencia múltiple, `__mro__`, `isinstance`, `issubclass`, `hasattr`, `getattr`, `is` frente a…
1:20	Ejercicios del capítulo.
1:45	Preguntas tipo examen y tarea 21.

Para empezar

Preguntar una duda: primero al compañero de mesa, luego al profesor de la materia, luego al tutor. Siempre en el mismo orden.

Predice

Escribe en papel qué crees que pasará. Después, lo ejecutamos.

```python
class A:
    def quien(self):
        return "A"

class B(A):
    def quien(self):
        return "B"

class C(A):
    def quien(self):
        return "C"

class D(B, C):
    pass

print(D().quien())
print([k.__name__ for k in D.__mro__])
print(isinstance(D(), C), issubclass(C, B))
```

Predice

Escribe en papel qué crees que pasará. Después, lo ejecutamos.

```python
class A:
    def quien(self):
        return "A"

class B(A):
    def quien(self):
        return "B"

class C(A):
    def quien(self):
        return "C"

class D(B, C):
    pass

print(D().quien())
print([k.__name__ for k in D.__mro__])
print(isinstance(D(), C), issubclass(C, B))
```

LO QUE SALE

`B` · `['D', 'B', 'C', 'A', 'object']` ·

`True False`

El MRO recorre de izquierda a derecha y deja `A` para el final.

Herencia múltiple y MRO

Clases a fondo · 4

```
class A:
    def quien(self):
        return "A"

class B(A):
    def quien(self):
        return "B"

class C(A):
    def quien(self):
        return "C"

class D(B, C):
…
```

Introspección: el objeto se describe a sí mismo

Clases a fondo · 5

```
p = Perro("Rex")
print(type(p).__name__, Perro.__bases__, Perro.__module__)
print(isinstance(p, Animal), issubclass(Perro, Animal), issubclass(Perro, Gato))
setattr(p, "edad", 3)
print(p.__dict__, getattr(p, "edad"))
```

Construir objetos con cuidado

Clases a fondo · 6

```
class Cuenta:
    def __init__(self, titular, saldo=0, movimientos=None):
        if saldo < 0:
            raise ValueError("el saldo inicial no puede ser negativo")
        self.titular = titular
        self.saldo = saldo
        self.movimientos = [] if movimientos is None else movimientos

c = Cuenta("Ana", saldo=50)        # argumento por nombre: se lee solo
print(c.titular, c.saldo, c.movimientos)
try:
    Cuenta("Biel", -10)
except ValueError as e:
…
```

Identidad e igualdad: `is` y `==`

Clases a fondo · 7

```
x = [1, 2]
y = [1, 2]
z = x
print(x == y, x is y, x is z)
print(x is not y)
```

`__str__`, `__repr__` y propiedades

Clases a fondo · 8

```
class Punto:
    def __init__(self, x, y):
        self.__x, self.__y = x, y

    @property
    def x(self):
        return self.__x

    def __str__(self):
        return f"({self.__x}, {self.__y})"

    def __repr__(self):
        return f"Punto({self.__x}, {self.__y})"
…
```

Manos a la obra

Abre el cuaderno de la sesión en Colab, con los ejemplos de hoy y estos ejercicios:

1. ¿Qué imprime?
2. Desfiguración. Para class Caja: con self.__secreto = 7 en su…
3. Polimorfismo. Añade una clase Vaca(Animal) que diga «¡Muu!» y métela…
4. MRO. Con las clases A, B, C del apartado 4, ¿qué imprime…
5. Introspección. Escribe una expresión que diga si un objeto p tiene un…



Cuaderno en Colab

Tarea 21 · El rombo y las órdenes por nombre

1. Crea cuatro clases en rombo: `Base`, `Izq(Base)`, `Der(Base)` y `Abajo(Izq, Der)`.
2. **Predice** en qué orden salen los cuatro nombres al crear un `Abajo()`, y después compruébalo con `Abajo.__mro__`.
3. Crea una clase `Robot` con los métodos `saluda()` y `avanza()`.
4. Con una lista de órdenes en texto, `["saluda", "baila", "avanza"]`, ejecuta cada una con `getattr` si el robot la tiene…

COMPROBACIÓN `Base` se imprime una sola vez, la última.

Antes de irte

Las **preguntas tipo examen** del capítulo, en directo: se corrigen al momento.



Preguntas del capítulo

LA PRÓXIMA SESIÓN

Sesión 22 · Comprensiones y el `else` de los bucles