

POO

Se trata de crear una clase con el nombre CuentaJoven que construya cuentas corrientes bancarias heredando de la clase CuentaCorriente creada en el ejercicio del vídeo 33. Puedes crear la clase en el mismo fichero del ejercicio 33 (a continuación de la clase CuentaCorriente) o en un fichero aparte.

La clase CuentaJoven tendrá un atributo o propiedad propia con el nombre de bonus_promocion. El atributo bonus_promocion permitirá, de forma opcional, añadir un bonus que se añadirá al saldo de la cuenta. El establecimiento del bonus y el incremento del saldo se harán desde el constructor.

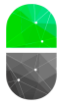
Además, la clase CuentaJoven contará con los siguientes métodos:

- getBonus() encargado de devolver el importe del bonus
- ingresar() que permitirá ingresar dinero a la cuenta. Este método se heredaré de CuentaCorriente
- retirar() que permitirá retirar dinero de la cuenta. Este método se heredaré de CuentaCorriente
- getDatos() que permitirá ver los datos de la Cuenta Joven: nº de cuenta, titular, saldo y bonus. Este método se heredaré de CuentaCorriente

Prueba el programa creando un objeto de tipo CuentaJoven, ingresa y retira dinero de la cuenta y finalmente muestra los datos de la cuenta.

Posible solución en la siguiente página:





POSIBLE SOLUCIÓN:

```
23 class CuentaJoven(CuentaCorriente):
24
25     def __init__(self, numero, titular, saldo, bonus_promocion=0):
26         super().__init__(numero, titular, saldo)
27         self.bonus_promocion=bonus_promocion
28         self.saldo+=bonus_promocion
29
30
31     def getBonus(self):
32
33         return self.bonus_promocion
34
35     def getDatos(self):
36
37         return super().getDatos() + " Bonus: " + str(self.bonus_promocion)
38
39     def ingresar(self, cantidad):
40
41         super().ingresar(cantidad)
42
43     def retirar(self, cantidad):
44
45         super().retirar(cantidad)
46
47
48 c1=CuentaJoven("0500", "María", 37000, 1500)
49
50 c1.ingresar(5000)
51
52 c1.retirar(350)
53
54 print (c1.getDatos())
```