



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS ADMINISTRATIVAS
CARRERA GESTIÓN DE LA INFORMACIÓN GERENCIAL



ASIGNATURA:

Programación Orientada A Objetos

TEMA:

Tarea de Python - Herencia - Asíncrono

DOCENTE:

Ing. Guillermo Enrique Valarezo Guzmán

NOMBRES:

Quiñonez Ronquillo Odalys Raquel

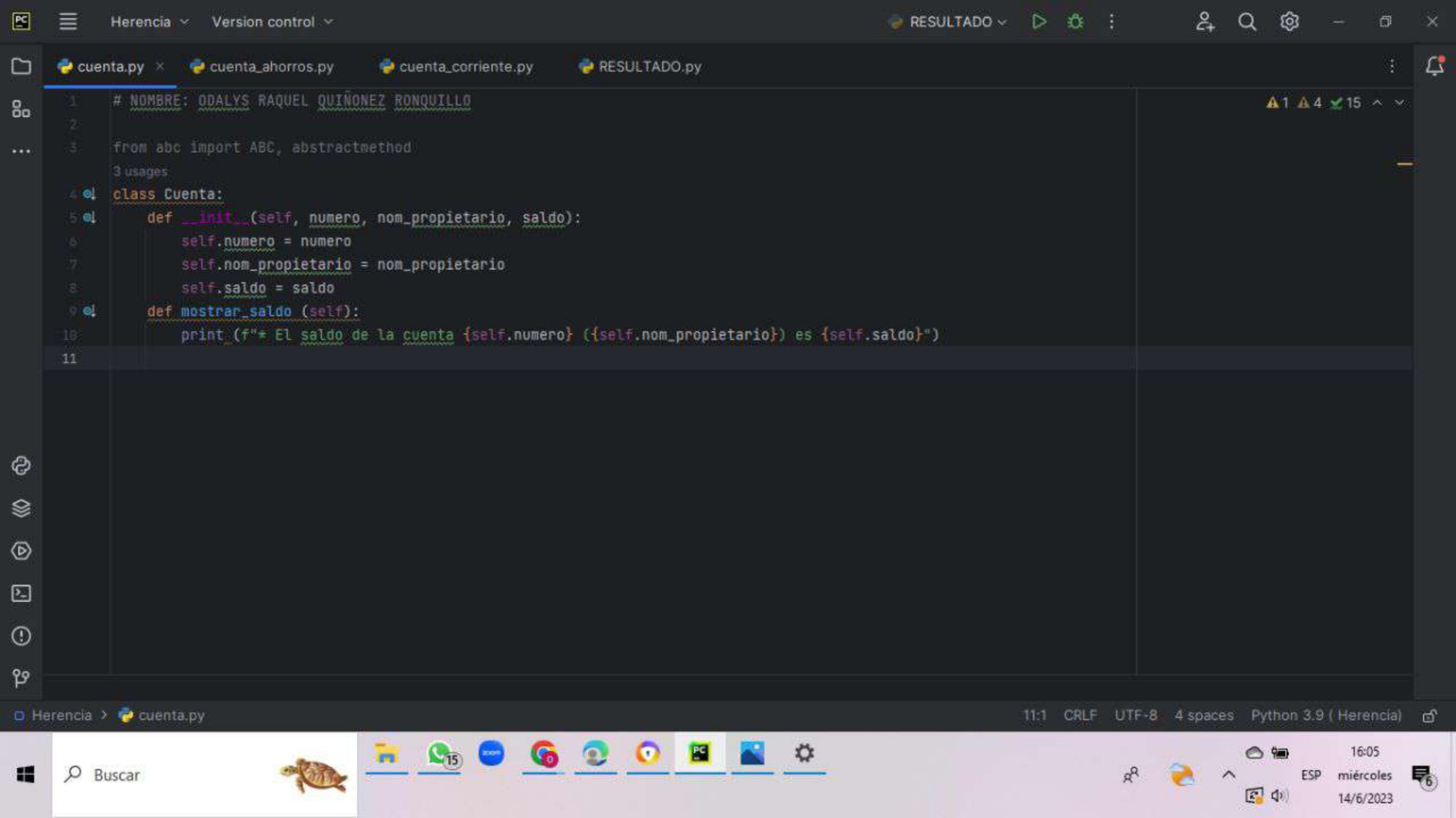
CURSO:

GIG-SNO-3-4

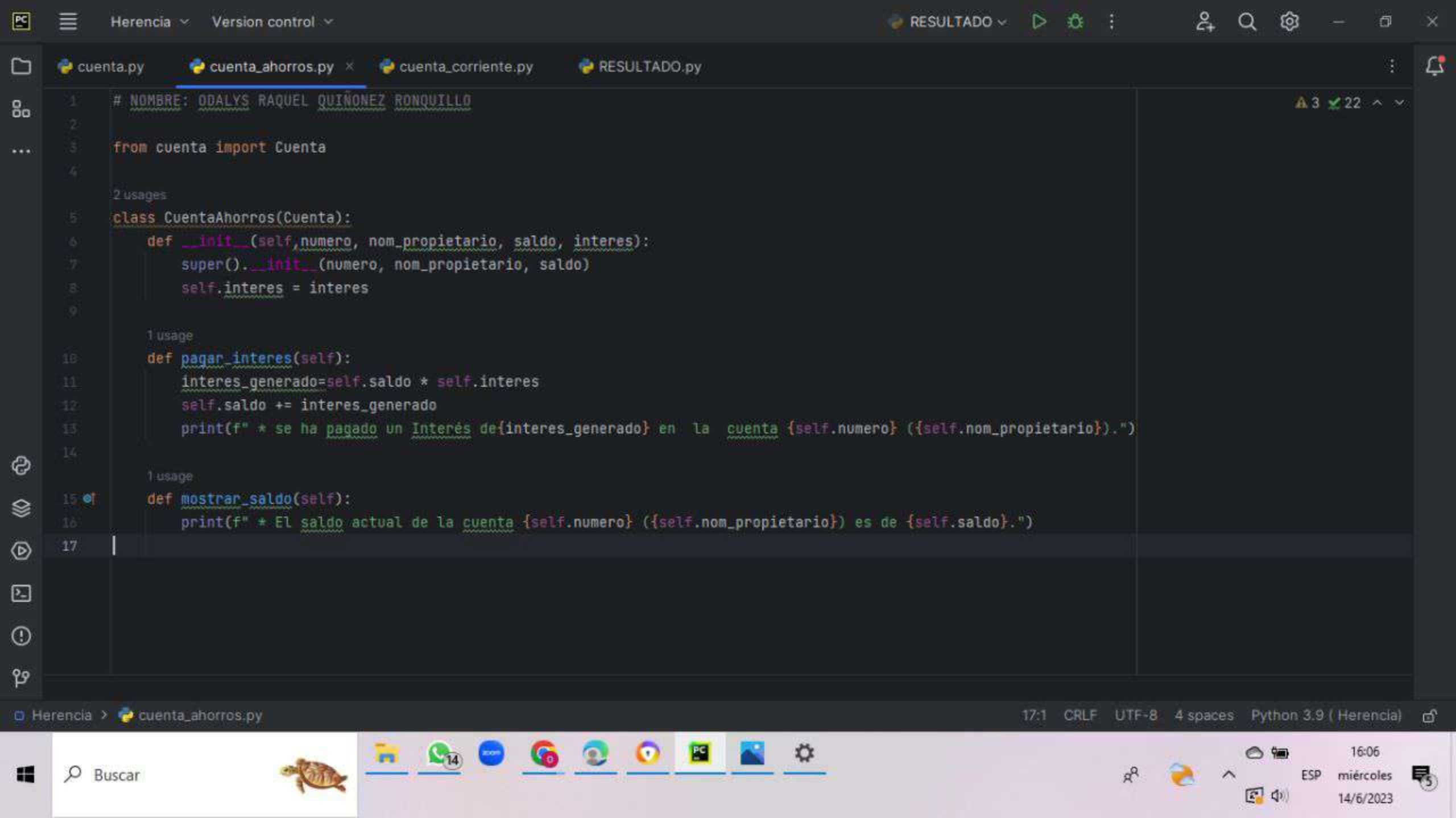
AÑO:

2022-2023 CII

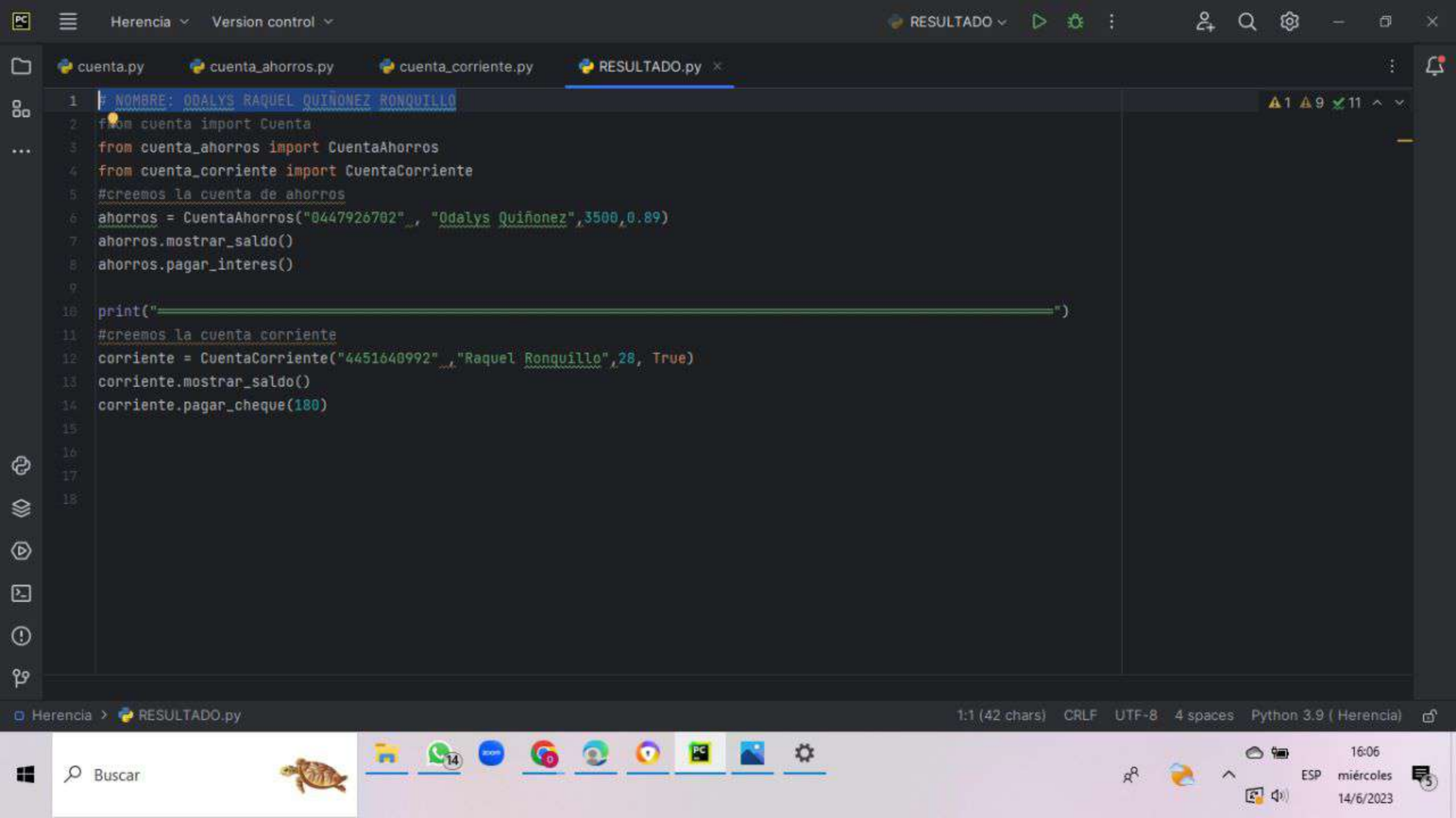
Guayaquil, 14 de Junio del 2023



```
1 # NOMBRE: ODALYS RAQUEL QUINONEZ RONQUILLO
2
3 from abc import ABC, abstractmethod
4
5 class Cuenta:
6     def __init__(self, numero, nom_propietario, saldo):
7         self.numero = numero
8         self.nom_propietario = nom_propietario
9         self.saldo = saldo
10    def mostrar_saldo (self):
11        print (f" El saldo de la cuenta {self.numero} ({self.nom_propietario}) es {self.saldo}")
```



```
1 #NOMBRE: ODALYS RAQUEL QUINONEZ RONQUILLO
2 from cuenta import Cuenta
3 class CuentaCorriente(Cuenta):
4     def __init__(self, numero, nom_propietario, saldo, tiene_chequera):
5         super().__init__(numero, nom_propietario, saldo)
6         self.tiene_chequera = tiene_chequera
7
8
9     def pagar_cheque(self, valor):
10         if self.tiene_chequera:
11             if self.saldo - valor >= 0:
12                 self.saldo -= valor
13                 print(f" * se ha pagado un cheque por {valor} desde la cuenta {self.numero} ({self.nom_propietario}).")
14             else:
15                 print(" * No se puede realizar la operación. saldo insuficiente.")
16         else:
17             print(" * No se puede realizar la operación. la cuenta no tiene chequera.")
18
19     def mostrar_saldo(self):
20         print(f" * El saldo actual de la cuenta {self.numero} ({self.nom_propietario}) es de {self.saldo}.")
```



```
1 # NOMBRE: ODALYS RAQUEL QUIÑONEZ RONQUILLO
2 from cuenta import Cuenta
3 from cuenta_ahorros import CuentaAhorros
4 from cuenta_corriente import CuentaCorriente
5 #creemos la cuenta de ahorros
6 ahorros = CuentaAhorros("0447926762", "Odalys Quiñonez", 3500, 0.89)
7 ahorros.mostrar_saldo()
8 ahorros.pagar_interes()
9
10 print("-----")
11 #creemos la cuenta corriente
12 corriente = CuentaCorriente("4451648992", "Raquel Ronquillo", 28, True)
13 corriente.mostrar_saldo()
14 corriente.pagar_cheque(180)
15
16
17
18
```

1 9 11

Herencia > RESULTADO.py

1:1 (42 chars) CRLF UTF-8 4 spaces Python 3.9 (Herencia)

Buscar

