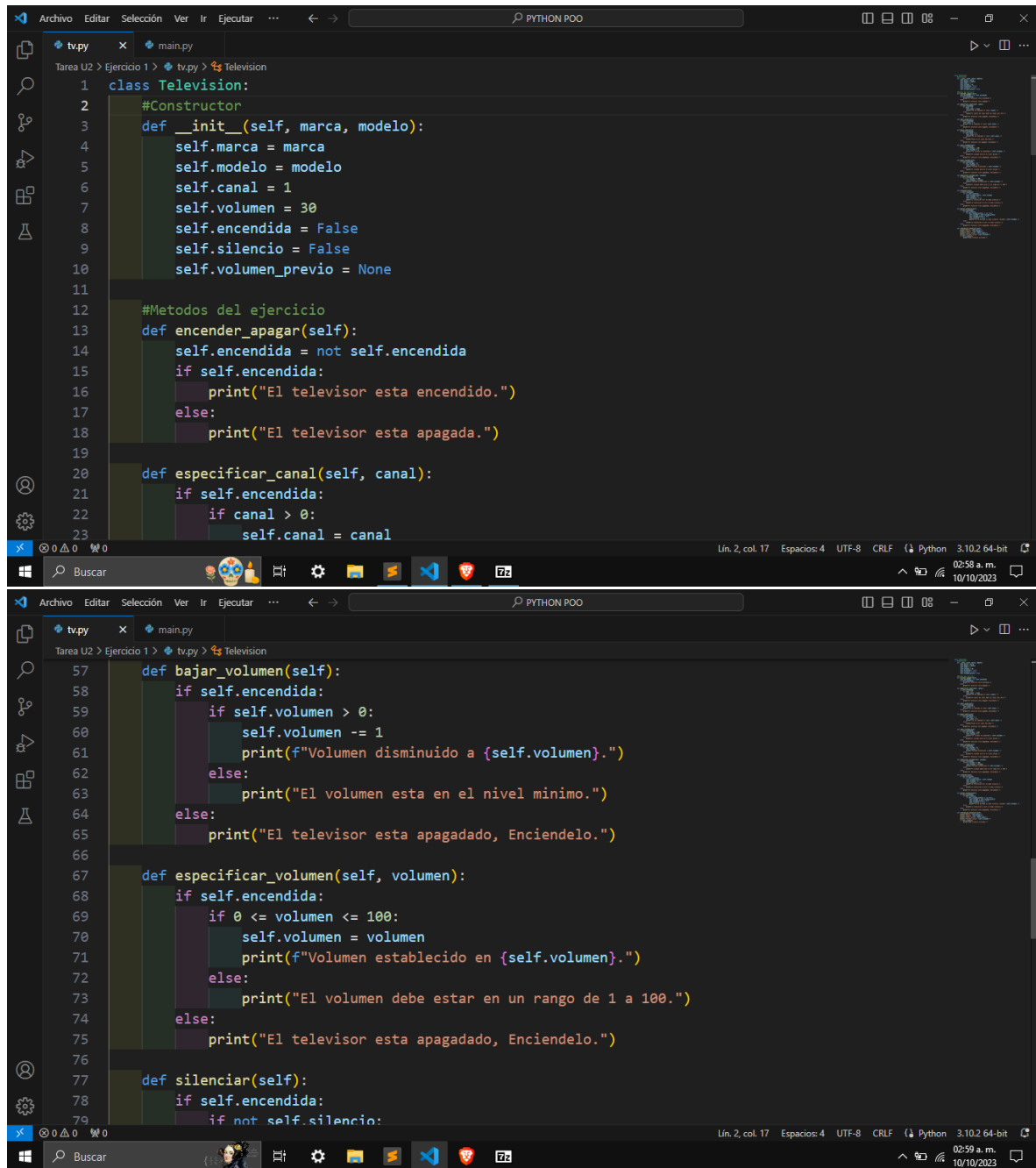


## EJERCICIO 1



The image displays two screenshots of a Python IDE (likely Visual Studio Code) showing the implementation of a `Television` class. The IDE interface includes a menu bar (Archivo, Editar, Selección, Ver, Ir, Ejecutar), a toolbar, and a status bar at the bottom. The file explorer on the left shows the project structure: `Tarea U2 > Ejercicio 1 > tv.py`.

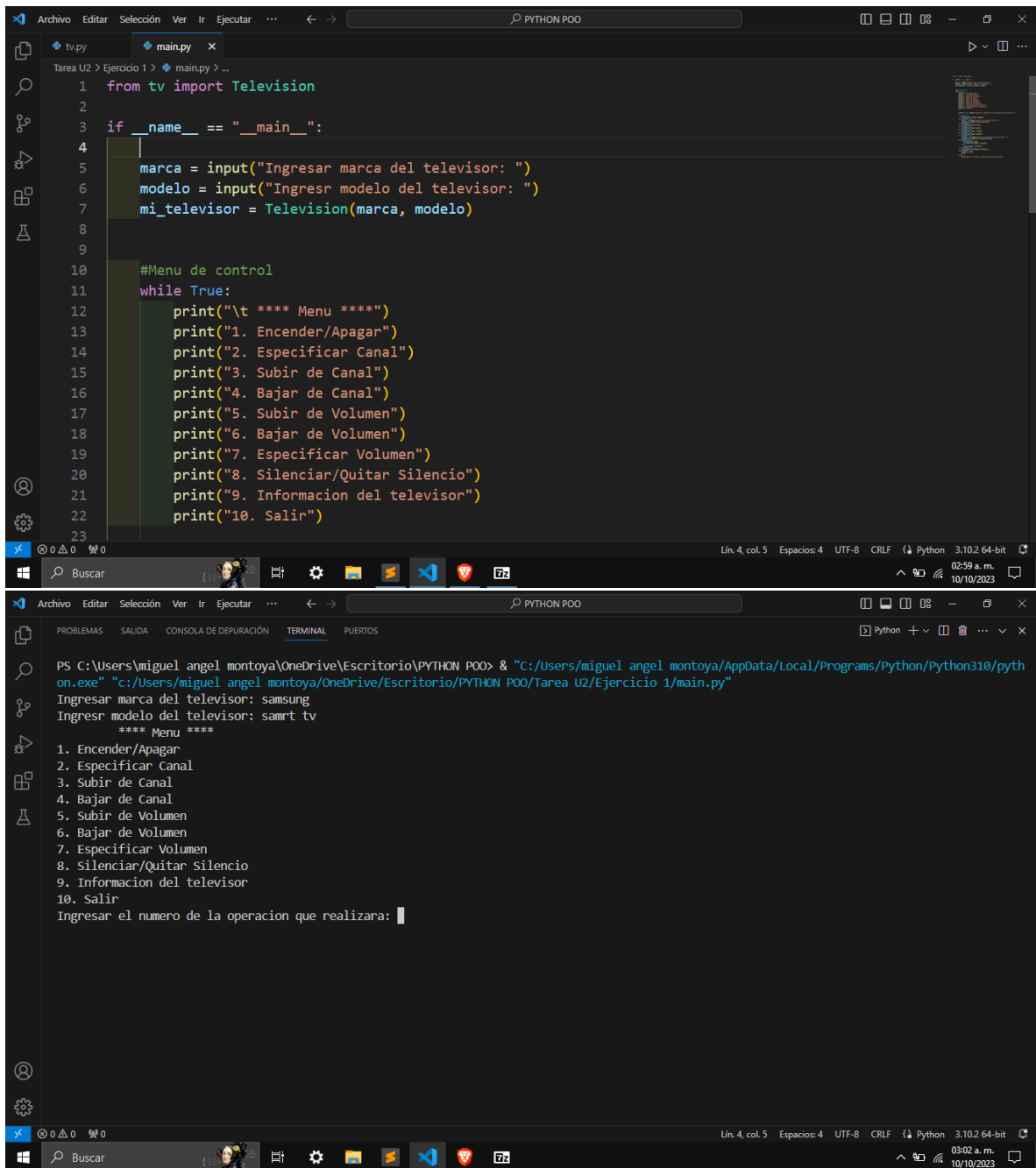
**Top Screenshot:** Shows the initial part of the `Television` class implementation in `tv.py`.

```
1 class Television:
2     #Constructor
3     def __init__(self, marca, modelo):
4         self.marca = marca
5         self.modelo = modelo
6         self.canal = 1
7         self.volumen = 30
8         self.encendida = False
9         self.silencio = False
10        self.volumen_previo = None
11
12    #Metodos del ejercicio
13    def encender_apagar(self):
14        self.encendida = not self.encendida
15        if self.encendida:
16            print("El televisor esta encendido.")
17        else:
18            print("El televisor esta apagada.")
19
20    def especificar_canal(self, canal):
21        if self.encendida:
22            if canal > 0:
23                self.canal = canal
```

**Bottom Screenshot:** Shows the continuation of the `Television` class implementation.

```
57    def bajar_volumen(self):
58        if self.encendida:
59            if self.volumen > 0:
60                self.volumen -= 1
61                print(f"Volumen disminuido a {self.volumen}.")
62            else:
63                print("El volumen esta en el nivel minimo.")
64        else:
65            print("El televisor esta apagado, Enciendolo.")
66
67    def especificar_volumen(self, volumen):
68        if self.encendida:
69            if 0 <= volumen <= 100:
70                self.volumen = volumen
71                print(f"Volumen establecido en {self.volumen}.")
72            else:
73                print("El volumen debe estar en un rango de 1 a 100.")
74        else:
75            print("El televisor esta apagado, Enciendolo.")
76
77    def silenciar(self):
78        if self.encendida:
79            if not self.silencio:
```

The status bar at the bottom of both screenshots indicates the current line and column (Lín. 2, col. 17), the number of spaces (Espacios: 4), the encoding (UTF-8), the line ending (CRLF), the Python version (Python 3.10.2 64-bit), and the date and time (02:58 a. m. 10/10/2023).



The image displays a Visual Studio Code (VS Code) environment with two panels. The top panel shows a Python script named `main.py` with the following code:

```
1 from tv import Television
2
3 if __name__ == "__main__":
4
5     marca = input("Ingresar marca del televisor: ")
6     modelo = input("Ingresar modelo del televisor: ")
7     mi_televisor = Television(marca, modelo)
8
9
10    #Menu de control
11    while True:
12        print("\t **** Menu ****")
13        print("1. Encender/Apagar")
14        print("2. Especificar Canal")
15        print("3. Subir de Canal")
16        print("4. Bajar de Canal")
17        print("5. Subir de Volumen")
18        print("6. Bajar de Volumen")
19        print("7. Especificar Volumen")
20        print("8. Silenciar/Quitar Silencio")
21        print("9. Informacion del televisor")
22        print("10. Salir")
23
```

The bottom panel shows the terminal output of the script execution. The command prompt shows the file path and the execution of `python.exe`. The output shows the user inputting `samsung` for the brand and `samrt tv` for the model. The program then displays a menu with 10 options. The user is prompted to enter the number of the operation to perform.

```
PS C:\Users\miguel angel montoya\OneDrive\Escritorio\PYTHON POO> & "C:/Users/miguel angel montoya/AppData/Local/Programs/Python/Python310/python.exe" "c:/Users/miguel angel montoya/OneDrive/Escritorio/PYTHON POO/Tarea U2/Ejercicio 1/main.py"
Ingresar marca del televisor: samsung
Ingresar modelo del televisor: samrt tv
**** Menu ****
1. Encender/Apagar
2. Especificar Canal
3. Subir de Canal
4. Bajar de Canal
5. Subir de Volumen
6. Bajar de Volumen
7. Especificar Volumen
8. Silenciar/Quitar Silencio
9. Informacion del televisor
10. Salir
Ingresar el numero de la operacion que realizara: 
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  ...  PYTHON POO
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS
Python + -

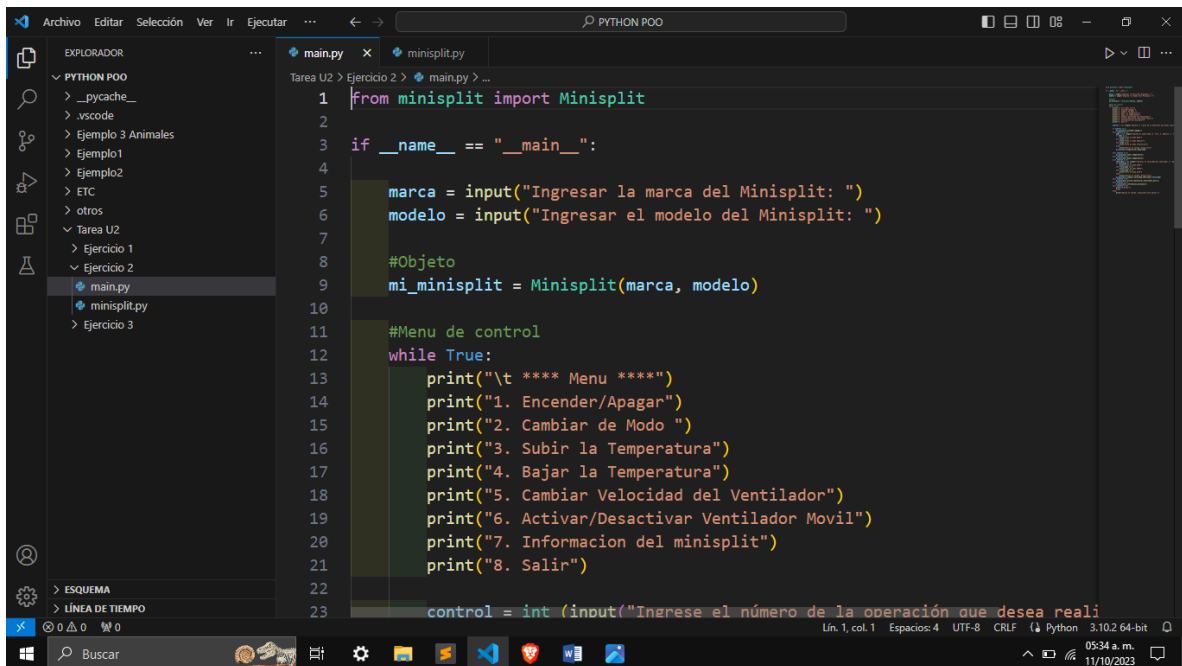
**** Menu ****
1. Encender/Apagar
2. Especificar Canal
3. Subir de Canal
4. Bajar de Canal
5. Subir de Volumen
6. Bajar de Volumen
7. Especificar Volumen
8. Silenciar/Quitar Silencio
9. Informacion del televisor
10. Salir
Ingresar el numero de la operacion que realizara: 9
Marca: samsung
Modelo: samrt tv
Canal actual: 10
Volumen actual: 34
**** Menu ****
1. Encender/Apagar
2. Especificar Canal
3. Subir de Canal
4. Bajar de Canal
5. Subir de Volumen
6. Bajar de Volumen
7. Especificar Volumen
8. Silenciar/Quitar Silencio
9. Informacion del televisor
10. Salir
Ingresar el numero de la operacion que realizara:

Lin. 4, col. 5  Espacios: 4  UTF-8  CRLF  Python 3.10.2 64-bit
03:03 a. m.
10/10/2023
```

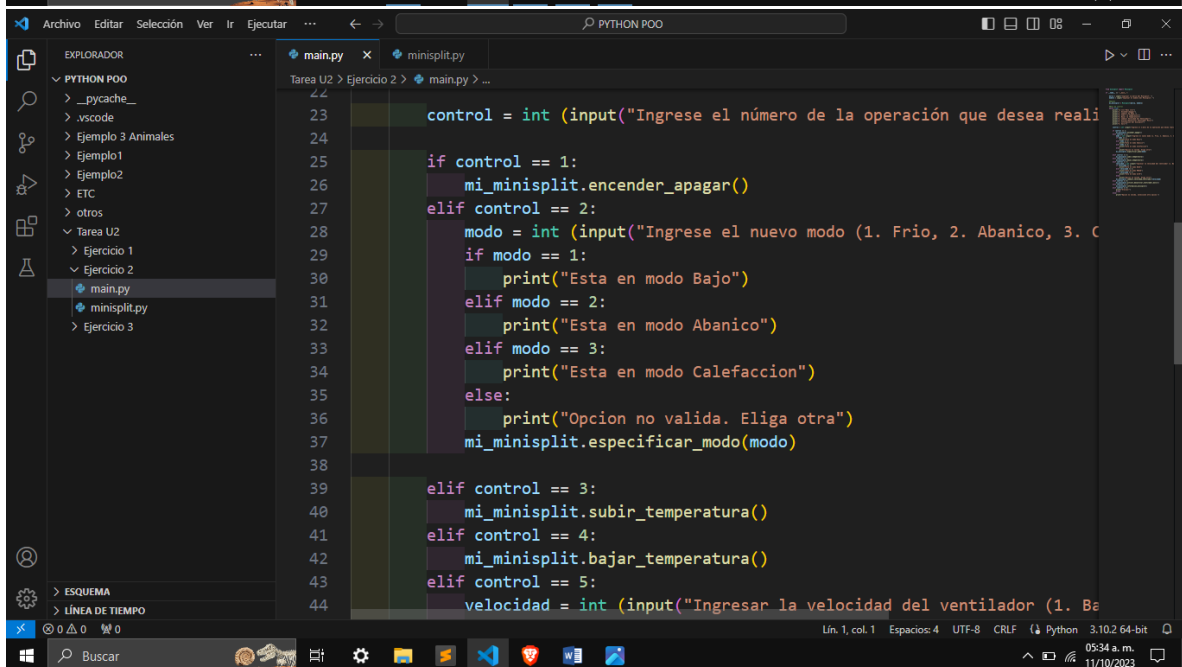
## EJERCICIO 2

```
1 class Minisplit:
2
3     #Constructor
4     def __init__(self, marca, modelo):
5         self.marca = marca
6         self.modelo = modelo
7         self.encendido = False
8         self.modo = "Frio"
9         self.temperatura = 25
10        self.velocidad_ventilador = "Bajo"
11        self.ventilador_movil_activated = False
12
13    #Metodos del ejercicio
14    def encender_apagar(self):
15        self.encendido = not self.encendido
16        if self.encendido:
17            print(f"{self.marca} {self.modelo} esta encendido.")
18        else:
19            print(f"{self.marca} {self.modelo} esta apagado.")
20
21
22    def especificar_modo(self, modo):
23        if self.encendido:
```

```
21
22    def especificar_modo(self, modo):
23        if self.encendido:
24            modos_validos = [1, "Frio", 2, "Abanico", 3, "Calefacción"]
25            if modo in modos_validos:
26                self.modo = modo
27                print(f"Modo cambiado a {modo}.")
28            else:
29                print("Modo no valido. Los modos validos son: Frio, Abanico, Ca")
30        else:
31            print("El Minisplit esta apagado. Enciendolo primero.")
32
33    def subir_temperatura(self):
34        if self.encendido:
35            if self.temperatura < 35:
36                self.temperatura += 1
37                print(f"Temperatura aumentada a {self.temperatura} grados Celsi")
38            else:
39                print("La temperatura se encuentra en el nivel maximo.")
40        else:
41            print("El Minisplit esta apagado. Enciendolo primero.")
42
43    def bajar_temperatura(self):
```



```
1 from minisplit import Minisplit
2
3 if __name__ == "__main__":
4
5     marca = input("Ingresar la marca del Minisplit: ")
6     modelo = input("Ingresar el modelo del Minisplit: ")
7
8     #Objeto
9     mi_minisplit = Minisplit(marca, modelo)
10
11     #Menu de control
12     while True:
13         print("\t **** Menu ****")
14         print("1. Encender/Apagar")
15         print("2. Cambiar de Modo ")
16         print("3. Subir la Temperatura")
17         print("4. Bajar la Temperatura")
18         print("5. Cambiar Velocidad del Ventilador")
19         print("6. Activar/Desactivar Ventilador Movil")
20         print("7. Informacion del minisplit")
21         print("8. Salir")
22
23     control = int (input("Ingrese el número de la operación que desea reali
```



```
23     control = int (input("Ingrese el número de la operación que desea reali
24
25     if control == 1:
26         mi_minisplit.encender_apagar()
27     elif control == 2:
28         modo = int (input("Ingrese el nuevo modo (1. Frio, 2. Abanico, 3. C
29         if modo == 1:
30             print("Esta en modo Bajo")
31         elif modo == 2:
32             print("Esta en modo Abanico")
33         elif modo == 3:
34             print("Esta en modo Calefaccion")
35         else:
36             print("Opcion no valida. Eliga otra")
37         mi_minisplit.especificar_mod(modo)
38
39     elif control == 3:
40         mi_minisplit.subir_temperatura()
41     elif control == 4:
42         mi_minisplit.bajar_temperatura()
43     elif control == 5:
44         velocidad = int (input("Ingresar la velocidad del ventilador (1. Ba
```

Archivo Editar Selección Ver Ir Ejecutar ... PYTHON POO

EXPLORADOR

- PYTHON POO
  - \_\_pycache\_\_
  - .vscode
  - Ejemplo 3 Animales
  - Ejemplo1
  - Ejemplo2
  - ETC
  - otros
  - Tarea U2
    - Ejercicio 1
    - Ejercicio 2
    - \_\_pycache\_\_
    - main.py
    - minisplit.py
    - Ejercicio 3
- ESQUEMA
- LÍNEA DE TIEMPO

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

```
PS C:\Users\miguel angel montoya\OneDrive\Escritorio\PYTHON POO> & "C:/Users/miguel angel montoya/AppData/Local/Programs/Python/Python310/python.exe" "c:/Users/miguel angel montoya/OneDrive/Escritorio/PYTHON POO/Tarea U2/Ejercicio 2/main.py"
Ingresar la marca del Minisplit: YORK
Ingresar el modelo del Minisplit: WR
**** Menu ****
1. Encender/Apagar
2. Cambiar de Modo
3. Subir la Temperatura
4. Bajar la Temperatura
5. Cambiar Velocidad del Ventilador
6. Activar/Desactivar Ventilador Movil
7. Informacion del minisplit
8. Salir
Ingrese el número de la operación que desea realizar: 
```

Lin. 34, col. 50 Espacios: 4 UTF-8 CRLF Python 3.10.2 64-bit 05:42 a. m. 11/10/2023

Archivo Editar Selección Ver Ir Ejecutar ... PYTHON POO

EXPLORADOR

- PYTHON POO
  - \_\_pycache\_\_
  - .vscode
  - Ejemplo 3 Animales
  - Ejemplo1
  - Ejemplo2
  - ETC
  - otros
  - Tarea U2
    - Ejercicio 1
    - Ejercicio 2
    - \_\_pycache\_\_
    - main.py
    - minisplit.py
    - Ejercicio 3
- ESQUEMA
- LÍNEA DE TIEMPO

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

```
PS C:\Users\miguel angel montoya\OneDrive\Escritorio\PYTHON POO> & "C:/Users/miguel angel montoya/AppData/Local/Programs/Python/Python310/python.exe" "c:/Users/miguel angel montoya/OneDrive/Escritorio/PYTHON POO/Tarea U2/Ejercicio 2/main.py"
Ingresar la marca del Minisplit: YORK
Ingresar el modelo del Minisplit: WR
**** Menu ****
1. Encender/Apagar
2. Cambiar de Modo
3. Subir la Temperatura
4. Bajar la Temperatura
5. Cambiar Velocidad del Ventilador
6. Activar/Desactivar Ventilador Movil
7. Informacion del minisplit
8. Salir
Ingrese el número de la operación que desea realizar: 1
YORK WR esta encendido.
**** Menu ****
1. Encender/Apagar
2. Cambiar de Modo
3. Subir la Temperatura
4. Bajar la Temperatura
5. Cambiar Velocidad del Ventilador
6. Activar/Desactivar Ventilador Movil
7. Informacion del minisplit
8. Salir
Ingrese el número de la operación que desea realizar: 
```

Lin. 34, col. 50 Espacios: 4 UTF-8 CRLF Python 3.10.2 64-bit 05:42 a. m. 11/10/2023

Archivo Editar Selección Ver Ir Ejecutar ... PYTHON POO

EXPLORADOR

- PYTHON POO
  - \_pycache\_
  - .vscode
  - Ejemplo 3 Animales
  - Ejemplo1
  - Ejemplo2
  - ETC
  - otros
  - Tarea U2
    - Ejercicio 1
    - Ejercicio 2
    - \_pycache\_
    - main.py
    - minisplit.py
    - Ejercicio 3
- ESQUEMA
- LÍNEA DE TIEMPO

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

```
8. Salir
Ingrese el número de la operación que desea realizar: 1
YORK WR esta encendido.
**** Menu ****

1. Encender/Apagar
2. Cambiar de Modo
3. Subir la Temperatura
4. Bajar la Temperatura
5. Cambiar Velocidad del Ventilador
6. Activar/Desactivar Ventilador Movil
7. Informacion del minisplit
8. Salir
Ingrese el número de la operación que desea realizar: 7
YORK WR - Estado: encendido
Modo: Frio
Temperatura actual: 25 grados
Velocidad del ventilador: Bajo
El ventilador movil esta desactivado.
**** Menu ****

1. Encender/Apagar
2. Cambiar de Modo
3. Subir la Temperatura
4. Bajar la Temperatura
5. Cambiar Velocidad del Ventilador
6. Activar/Desactivar Ventilador Movil
7. Informacion del minisplit
8. Salir
Ingrese el número de la operación que desea realizar: 
```

Lin. 34, col. 50 Espacios: 4 UTF-8 CRLF Python 3.10.2 64-bit

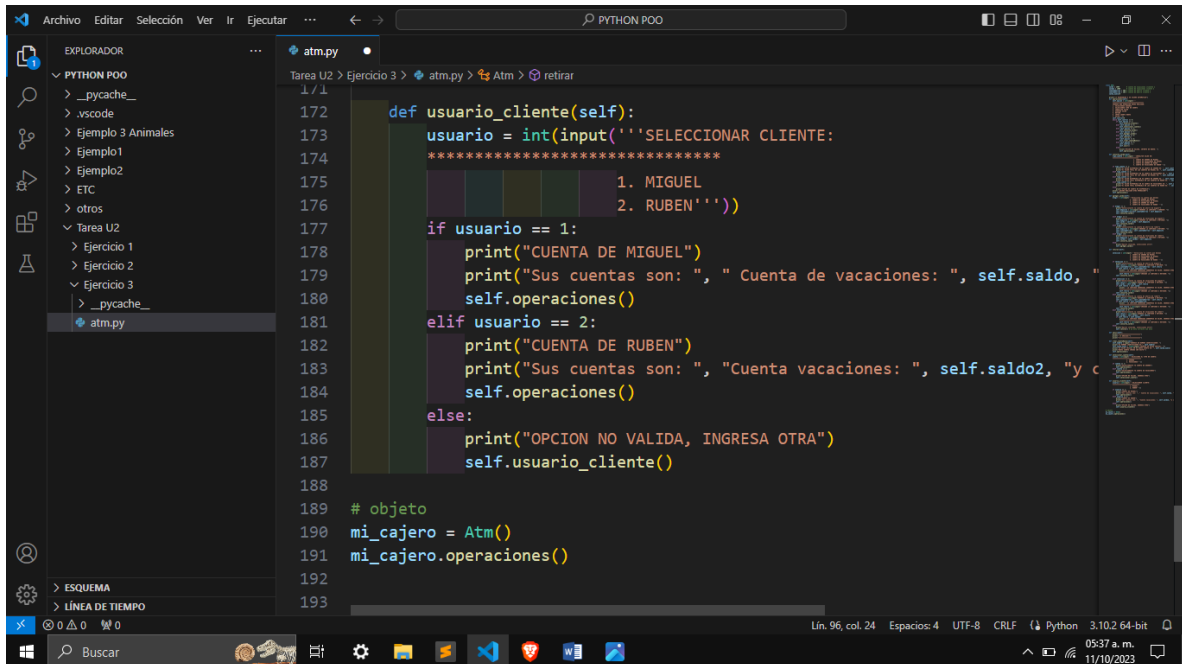
05:42 a. m. 11/10/2023

## EJERCICIO 3

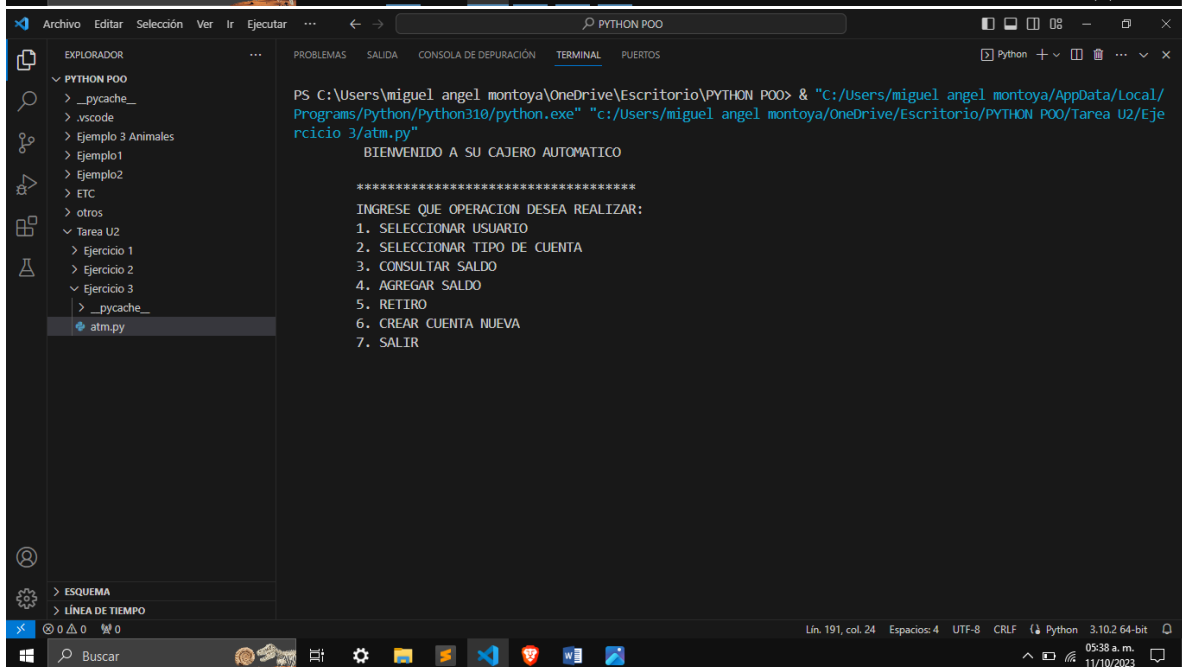
```
18 6. CREAR CUENTA NUEVA
19 7. SALIR''')
20 self.control=0
21 while self.control == 0:
22     if self.opcion == 1:
23         self.usuario_cliente()
24     elif self.opcion == 2:
25         self.seleccionar_cuenta()
26     elif self.opcion == 3:
27         self.consulta_saldo()
28     elif self.opcion == 4:
29         self.agregar_saldo()
30     elif self.opcion == 5:
31         self.retirar()
32     elif self.opcion == 6:
33         self.crear_cuentaNueva()
34     elif self.opcion == 7:
35         self.control = 1
36         self.salir()
37     else:
38         print("OPCION NO VALIDA, INTENTE DE NUEVO: ")
39         self.operaciones()
40
```

```
144 def salir(self):
145     print("*****")
146     print(" \t GRACIAS!")
147     print("*****")
148
149 def crear_cuentaNueva(self):
150     self.nombre = input("INGRESE UN NOMBRE IDENTIFICADOR: ")
151     print("SU NOMBRE IDENTIFICADOR ES:", self.nombre)
152     self.saldo_nuevo = float(input("INGRESE UN SALDO INICIAL: "))
153     print("SU SALDO INICIAL EN SU NUEVA CUENTA ES: ", self.saldo_nuevo)
154     print("NUEVA CUENTA CREADA CON EXITO!")
155     self.operaciones()
156
157 def seleccionar_cuenta(self):
158     cuenta = int(input('SELECCIONA EL TIPO DE CUENTA:
159     *****
160     1. AHORRO
161     2. VACACIONES'))
162     if cuenta == 1:
163         print("SELECCIONASTE TU CUENTA DE AHORRO")
164         self.operaciones()
165     elif cuenta == 2:
166         print("SELECCIONASTE TU CUENTA DE VACACIONES")
```

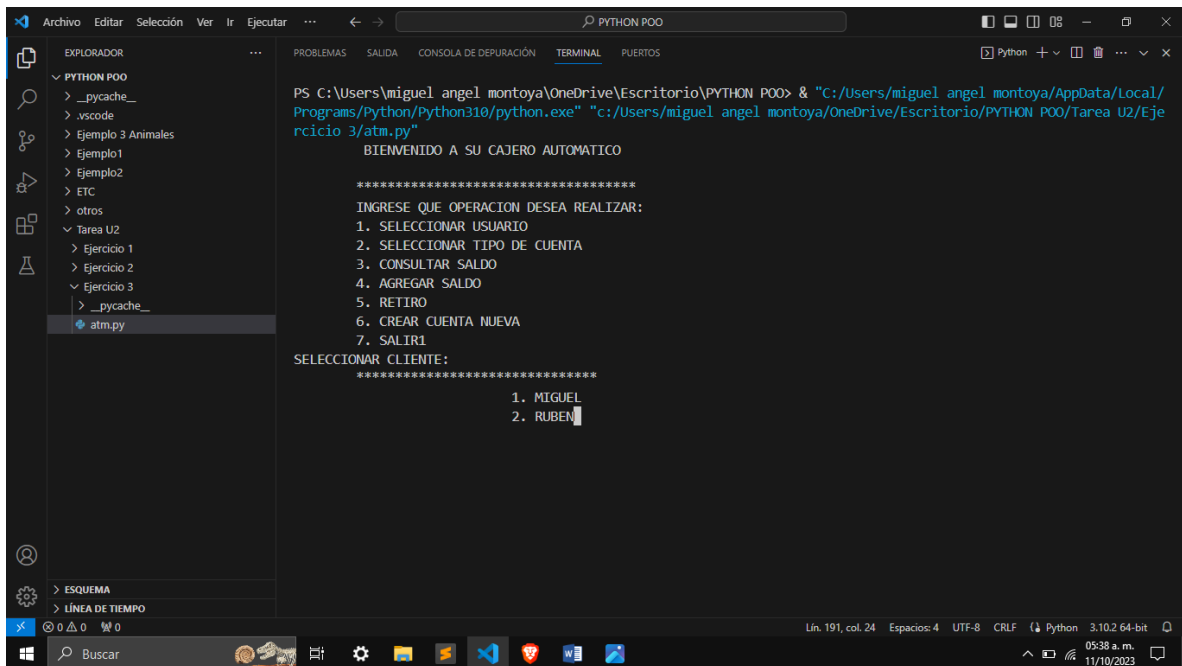




```
172 def usuario_cliente(self):
173     usuario = int(input('SELECCIONAR CLIENTE:
174     *****
175     1. MIGUEL
176     2. RUBEN'))
177
178     if usuario == 1:
179         print("CUENTA DE MIGUEL")
180         print("Sus cuentas son: ", " Cuenta de vacaciones: ", self.saldo, "
181         self.operaciones()
182     elif usuario == 2:
183         print("CUENTA DE RUBEN")
184         print("Sus cuentas son: ", "Cuenta vacaciones: ", self.saldo2, "y c
185         self.operaciones()
186     else:
187         print("OPCION NO VALIDA, INGRESA OTRA")
188         self.usuario_cliente()
189
190 # objeto
191 mi_cajero = Atm()
192 mi_cajero.operaciones()
193
```

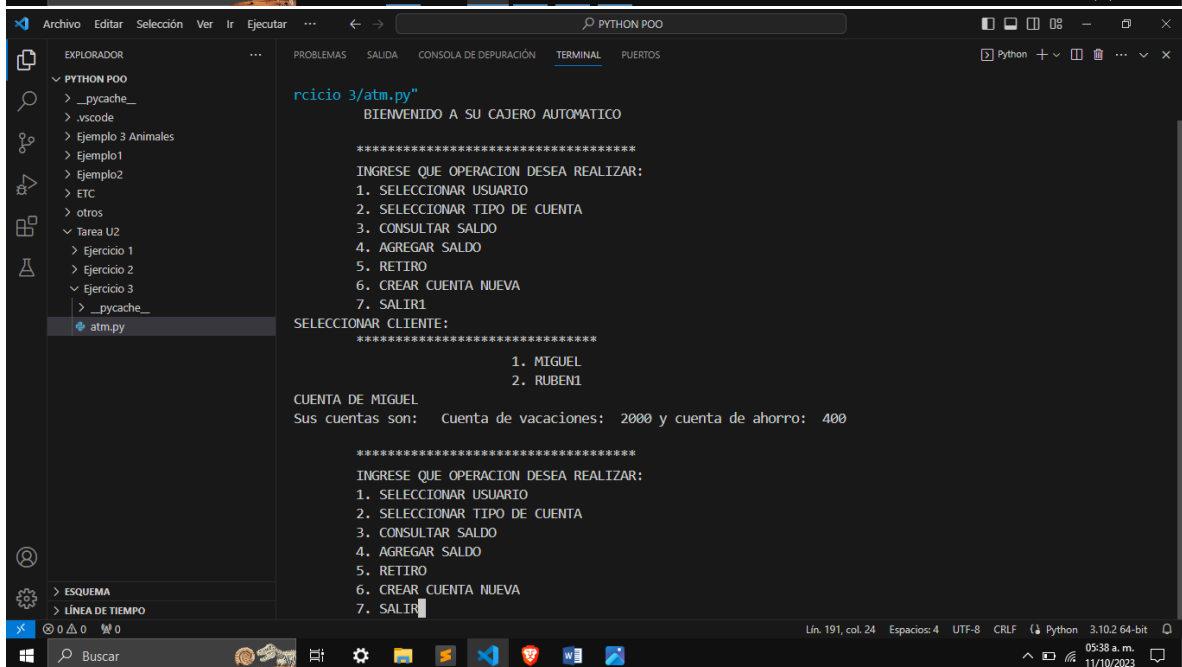


```
PS C:\Users\miguel angel montoya\OneDrive\Escritorio\PYTHON POO> & "C:/Users/miguel angel montoya/AppData/Local/
Programs/Python/Python310/python.exe" "c:/Users/miguel angel montoya/OneDrive/Escritorio/PYTHON POO/Tarea U2/Eje
rcio 3/atm.py"
BIENVENIDO A SU CAJERO AUTOMATICO
*****
INGRESE QUE OPERACION DESEA REALIZAR:
1. SELECCIONAR USUARIO
2. SELECCIONAR TIPO DE CUENTA
3. CONSULTAR SALDO
4. AGREGAR SALDO
5. RETIRO
6. CREAR CUENTA NUEVA
7. SALIR
```



```
PS C:\Users\miguel angel montoya\OneDrive\Escritorio\PYTHON POO> & "C:\Users\miguel angel montoya\AppData\Local\Programs\Python\Python310\python.exe" "c:/Users/miguel angel montoya/OneDrive/Escritorio/PYTHON POO/Tarea U2/Ejercicio 3/atm.py"
BIENVENIDO A SU CAJERO AUTOMATICO

*****
INGRESE QUE OPERACION DESEA REALIZAR:
1. SELECCIONAR USUARIO
2. SELECCIONAR TIPO DE CUENTA
3. CONSULTAR SALDO
4. AGREGAR SALDO
5. RETIRO
6. CREAR CUENTA NUEVA
7. SALIR1
SELECCIONAR CLIENTE:
*****
1. MIGUEL
2. RUBEN
```



```
rcicio 3/atm.py"
BIENVENIDO A SU CAJERO AUTOMATICO

*****
INGRESE QUE OPERACION DESEA REALIZAR:
1. SELECCIONAR USUARIO
2. SELECCIONAR TIPO DE CUENTA
3. CONSULTAR SALDO
4. AGREGAR SALDO
5. RETIRO
6. CREAR CUENTA NUEVA
7. SALIR1
SELECCIONAR CLIENTE:
*****
1. MIGUEL
2. RUBEN1

CUENTA DE MIGUEL
Sus cuentas son: Cuenta de vacaciones: 2000 y cuenta de ahorro: 400

*****
INGRESE QUE OPERACION DESEA REALIZAR:
1. SELECCIONAR USUARIO
2. SELECCIONAR TIPO DE CUENTA
3. CONSULTAR SALDO
4. AGREGAR SALDO
5. RETIRO
6. CREAR CUENTA NUEVA
7. SALIR
```