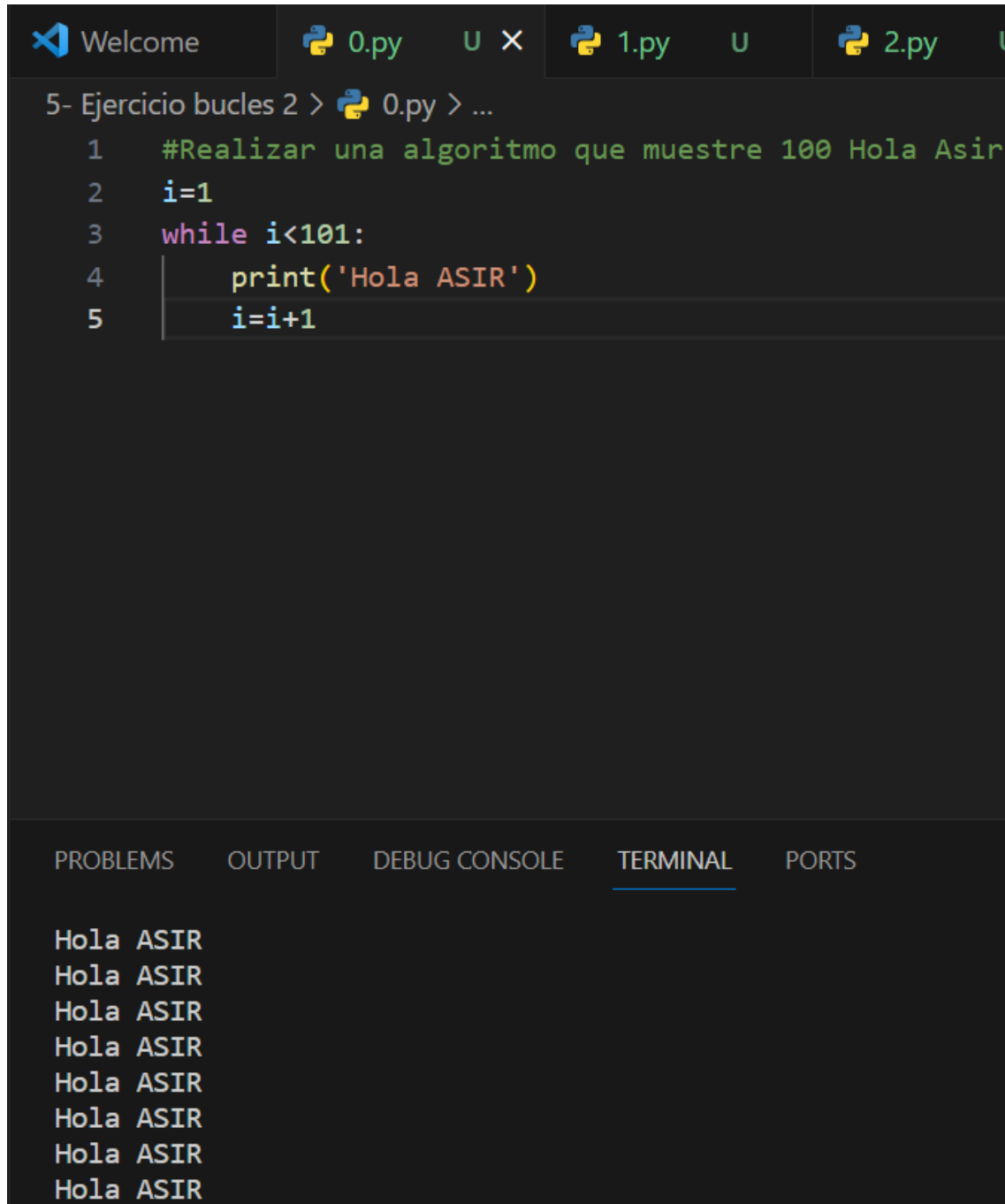


De cada uno de los ejercicios debe subir una captura de la ejecución

Ejercicio 0

Realizar un algoritmo que muestre 100 Hola Asir



The screenshot shows a Python IDE with three tabs: 'Welcome', '0.py', and '1.py'. The '0.py' tab is active, displaying the following code:

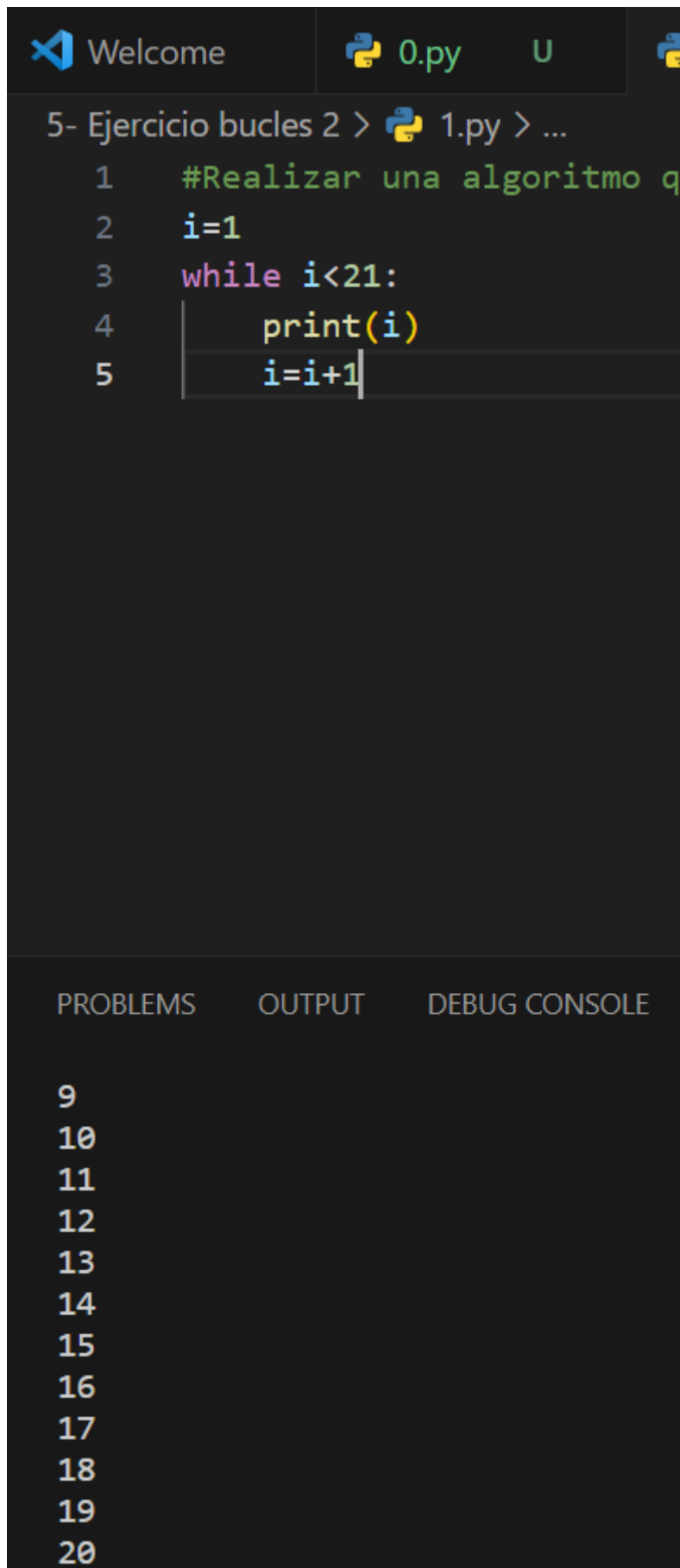
```
5- Ejercicio bucles 2 > 0.py > ...  
1 #Realizar un algoritmo que muestre 100 Hola Asir  
2 i=1  
3 while i<101:  
4     print('Hola ASIR')  
5     i=i+1
```

Below the code editor, the 'TERMINAL' tab is selected, showing the output of the program:

```
Hola ASIR  
Hola ASIR  
Hola ASIR  
Hola ASIR  
Hola ASIR  
Hola ASIR  
Hola ASIR  
Hola ASIR  
Hola ASIR
```

Ejercicio 1

Realizar un algoritmo que se repita 20 veces y muestre la iteración en la que esta



The image shows a screenshot of a Python IDE. The top bar has tabs for 'Welcome', '0.py', and 'U'. The main editor area shows a Python script with the following code:

```
5- Ejercicio bucles 2 > 1.py > ...  
1  #Realizar una algoritmo q  
2  i=1  
3  while i<21:  
4      print(i)  
5      i=i+1
```

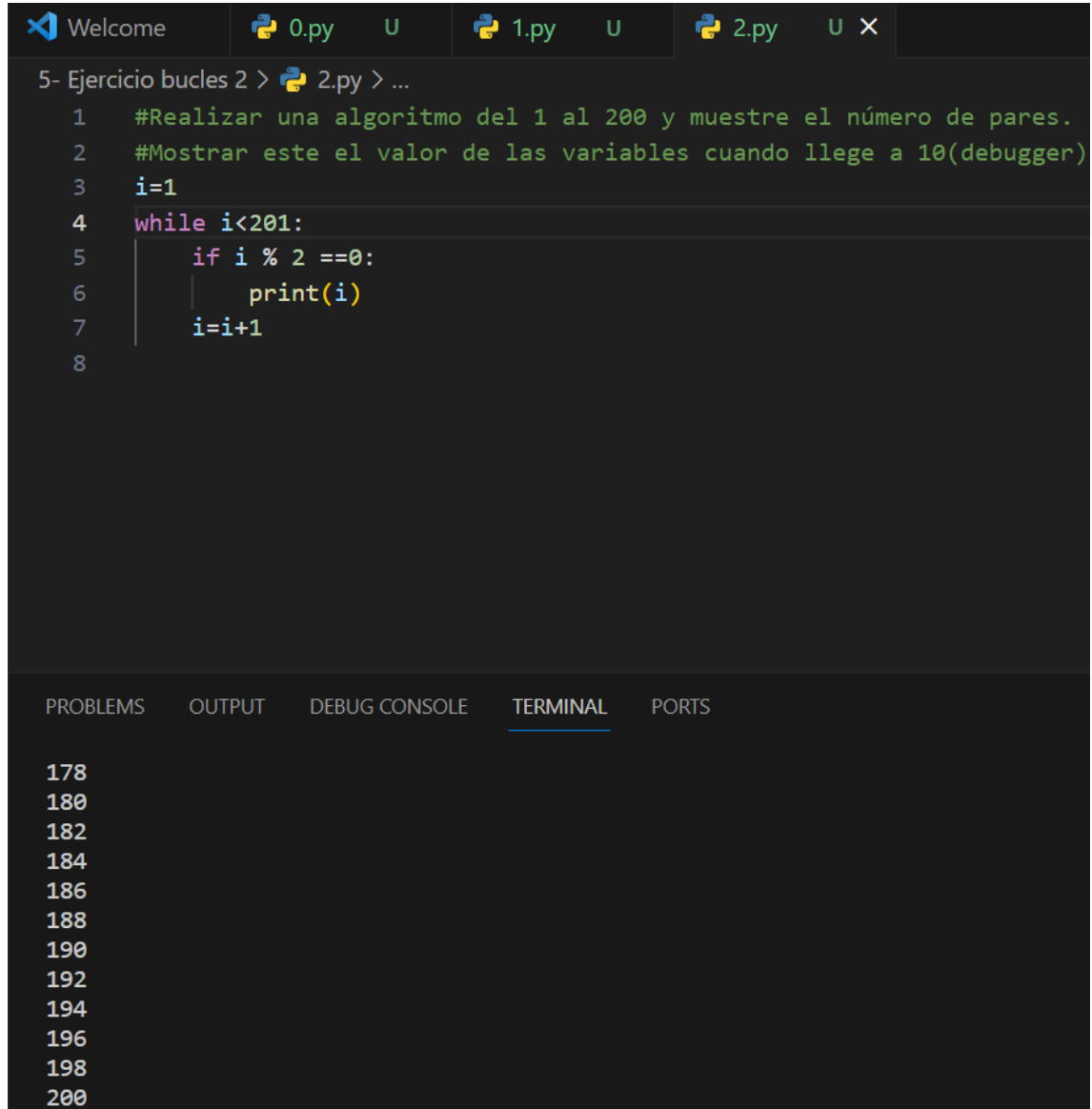
Below the editor, there are three tabs: 'PROBLEMS', 'OUTPUT', and 'DEBUG CONSOLE'. The 'OUTPUT' tab is active, displaying the output of the program:

```
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20
```

Ejercicio 2

Realizar una algoritmo del 1 al 200 y muestre el número de pares.

Mostrar este el valor de las variables cuando llegue a 10(debugger)



The screenshot shows a code editor with three tabs: '0.py', '1.py', and '2.py'. The active tab is '2.py', which contains the following Python code:

```
5- Ejercicio bucles 2 > 2.py > ...  
1 #Realizar una algoritmo del 1 al 200 y muestre el número de pares.  
2 #Mostrar este el valor de las variables cuando llegue a 10(debugger)  
3 i=1  
4 while i<201:  
5     if i % 2 ==0:  
6         print(i)  
7         i=i+1  
8
```

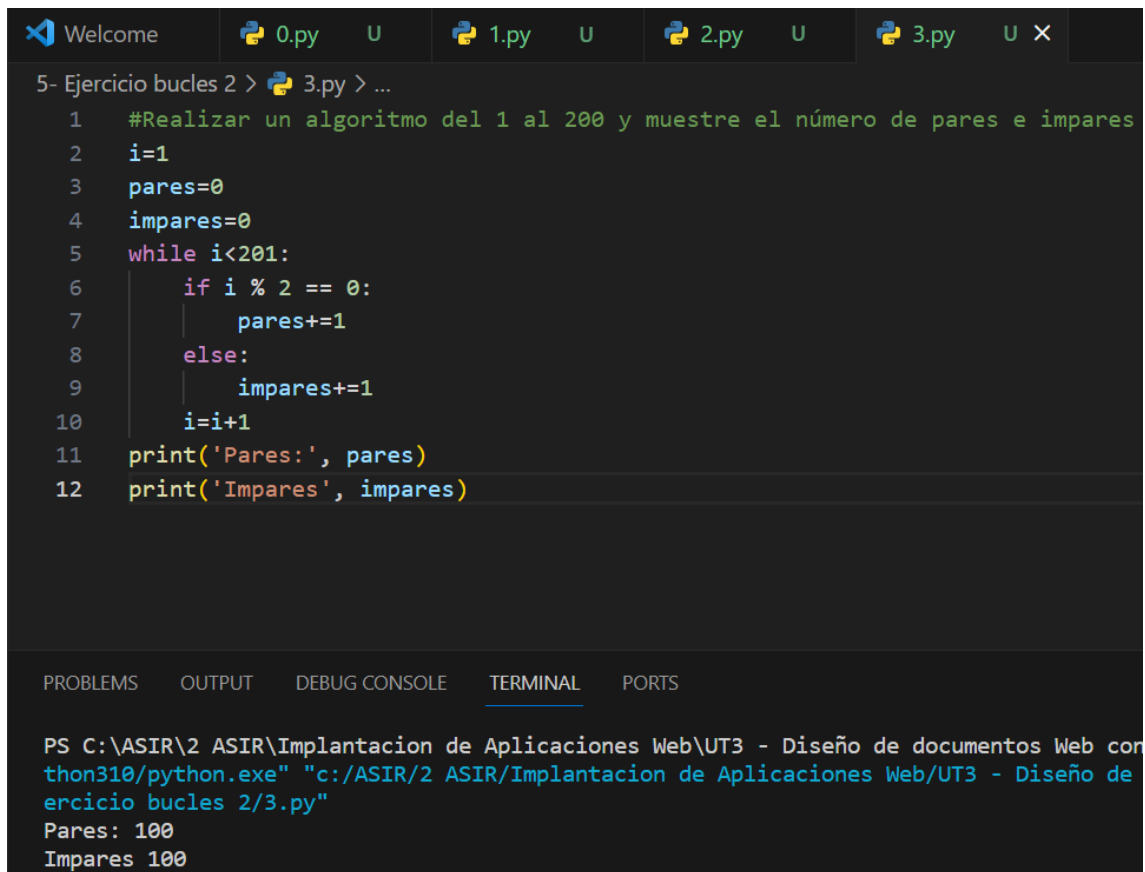
Below the code editor, the 'TERMINAL' tab is active, displaying the output of the program:

```
178  
180  
182  
184  
186  
188  
190  
192  
194  
196  
198  
200
```

Ejercicio 3 (contador)

Realizar un algoritmo del 1 al 200 y muestre el número de pares e impares

EJERCICIOS DE REPETITIVAS



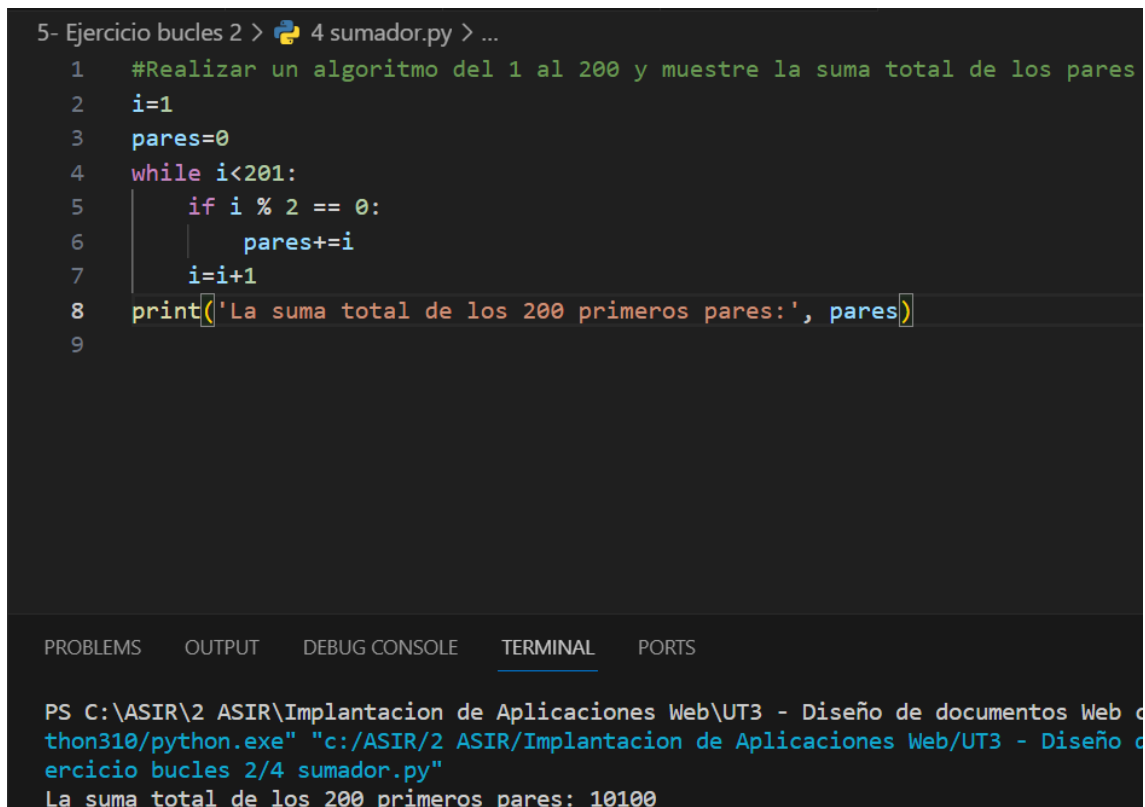
```
5- Ejercicio bucles 2 > 3.py > ...
1  #Realizar un algoritmo del 1 al 200 y muestre el número de pares e impares
2  i=1
3  pares=0
4  impares=0
5  while i<201:
6      if i % 2 == 0:
7          pares+=1
8      else:
9          impares+=1
10     i=i+1
11 print('Pares:', pares)
12 print('Impares', impares)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\ASIR\2 ASIR\Implantacion de Aplicaciones Web\UT3 - Diseño de documentos Web con
thon310/python.exe" "c:/ASIR/2 ASIR/Implantacion de Aplicaciones Web/UT3 - Diseño de
ercicio bucles 2/3.py"
Pares: 100
Impares 100

Ejercicio 4 (sumador)

Realizar un algoritmo del 1 al 200 y muestre la suma total de los pares



```
5- Ejercicio bucles 2 > 4 sumador.py > ...
1  #Realizar un algoritmo del 1 al 200 y muestre la suma total de los pares
2  i=1
3  pares=0
4  while i<201:
5      if i % 2 == 0:
6          pares+=i
7      i=i+1
8  print('La suma total de los 200 primeros pares:', pares)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\ASIR\2 ASIR\Implantacion de Aplicaciones Web\UT3 - Diseño de documentos Web c
thon310/python.exe" "c:/ASIR/2 ASIR/Implantacion de Aplicaciones Web/UT3 - Diseño d
ercicio bucles 2/4 sumador.py"
La suma total de los 200 primeros pares: 10100

Ejercicio 5

Realizar una algoritmo que muestre la tabla de multiplicar del 3

```
5- Ejercicio bucles 2 > 5.py > ...
1  #Realizar un algoritmo del 1 al 200 y muestre la suma total de los pares
2  i=1
3  tablaMultiplicar = 3
4  while i<11:
5      print(tablaMultiplicar,'x',i,'=',tablaMultiplicar*i)
6      i+=1
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
thon310/python.exe" "c:/ASIR/2 ASIR/Implantacion de Aplicaciones Web/UT3 - Diseño de
ejercicio bucles 2/5.py"
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30
```

Ejercicio 6

Realizar una algoritmo que muestre la tabla de multiplicar de un número solicitado por teclado

EJERCICIOS DE REPETITIVAS

```
5- Ejercicio bucles 2 > 6.py > ...
1  #Realizar una algoritmo que muestre la tabla de multiplicar de un número solicitado por teclado
2  i=1
3  tablaMultiplicar = int(input('Introduzca un numero: '))
4  while i<11:
5      print(tablaMultiplicar,'x',i,'=',tablaMultiplicar*i)
6      i+=1
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python

```
ercicio bucles 2/6.py"
Introduzca un numero: 4
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36
4 x 10 = 40
```

Ejercicio 7

Crea una aplicación que permita adivinar un número. La aplicación genera un número **aleatorio** del 1 al 100. A continuación, va pidiendo números y va respondiendo si el número a adivinar es mayor o menor que el introducido, además de los intentos que te quedan (tienes **10 intentos** para acertarlo). El programa termina cuando se acierta el número (además te dice en cuantos intentos lo has acertado), si se llega al límite de intentos te muestra el número que había generado.

```
5- Ejercicio bucles 2 > 7.py > ...
1  '''Crea una aplicación que permita adivinar un número.
2  La aplicación genera un número aleatorio del 1 al 100.
3  A continuación, va pidiendo números y va respondiendo si el
4  número a adivinar es mayor o menor que el introducido,
5  además de los intentos que te quedan (tienes 10 intentos para acertarlo).
6  El programa termina cuando se acierta el número
7  (además te dice en cuantos intentos lo has acertado),
8  si se llega al límite de intentos te muestra el número que había generado.'''
9
10 import random
11 aleatorio=random.randint(1,100)
12 intentos=10
13 i=0
14 while i<intentos:
15     numero=int(input('Introduce un numero: '))
16     if aleatorio == numero:
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Introduce un numero: 50
El numero es mas pequeño
Introduce un numero: 25
El numero es mas grande
Introduce un numero: 35
El numero es mas grande
Introduce un numero: 45
El numero es mas pequeño
Introduce un numero: 40
El numero es mas grande
Introduce un numero: 42
Lo has adivinado, el numero era 42
```

Ejercicio 8

Realizar un algoritmo que pida 10 números. El programa debe informar de cuantos números introducidos son mayores que 0, menores que 0 e iguales a 0.

EJERCICIOS DE REPETITIVAS

```
1  '''Realizar un algoritmo que pida 10 números El programa debe
2  informar de cuantos números introducidos son mayores que 0, menores que 0 e iguales a 0.'''
3  i=0
4  mayor=0
5  menor=0
6  igual=0
7  while i<10:
8      numero=int(input('Introduce un numero: '))
9      if numero == 0:
10         igual+=1
11     else:
12         if numero>0:
13             mayor+=1
14         else:
15             menor+=1
16         i+=1
17     print('Fin de la ejecución')
18     print('Hay', mayor, 'mayores que 0')
19     print('Hay', menor, 'menores que 0')
20     print('Hay', igual, 'numeros 0')
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

PS C:\ASIR\2 ASIR\Implantacion de Aplicaciones Web\UT3 - Diseño de documentos Web con HTML y CSS\Python310\python.exe "c:/ASIR/2 ASIR/Implantacion de Aplicaciones Web/UT3 - Diseño de documentos Web con HTML y CSS\ejercicio bucles 2/8.py"

Introduce un numero: 1
Introduce un numero: 2
Introduce un numero: 3
Introduce un numero: -2
Introduce un numero: 3
Introduce un numero: -3
Introduce un numero: 0
Introduce un numero: 0
Introduce un numero: 0
Hay 4 mayores que 0
Hay 3 menores que 0
Hay 3 numeros 0

Ejercicio 9

Algoritmo que pida caracteres 20 e imprima 'VOCAL' si son vocales y 'NO VOCAL' en caso contrario, el programa termina cuando se introduce un espacio.

```
9.py 10.py 11.py
5- Ejercicio bucles 2 > 9.py > ...
1  '''Algoritmo que pida caracteres 20 e imprima 'VOCAL' si son vocales y
2  'NO VOCAL' en caso contrario, el programa termina cuando se introduce un espacio.'''
3  contador=0
4  vocal="aeiouAEIOU"
5  while contador < 20:
6      caracter =input('Introduce un caracter: ')
7      if caracter in vocal:
8          print('Es una vocal')
9      else:
10         print('No es una vocal')
11     contador+=1
```

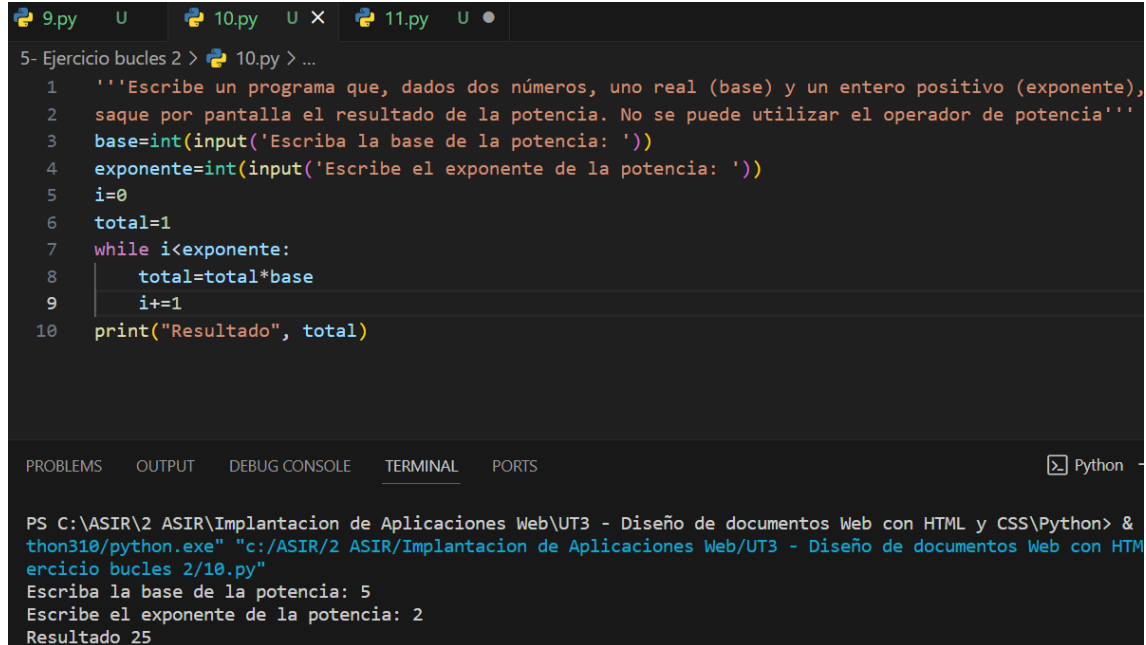
PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

PS C:\ASIR\2 ASIR\Implantacion de Aplicaciones Web\UT3 - Diseño de documentos Web con HTML y CSS\Python310\python.exe "c:/ASIR/2 ASIR/Implantacion de Aplicaciones Web/UT3 - Diseño de documentos Web con HTML y CSS\ejercicio bucles 2/9.py"

Introduce un caracter: a
Es una vocal
Introduce un caracter: A
Es una vocal
Introduce un caracter: d
No es una vocal
Introduce un caracter: D
No es una vocal

Ejercicio 10

Escribe un programa que, dados dos números, uno real (base) y un entero positivo (exponente), saque por pantalla el resultado de la potencia. No se puede utilizar el operador de potencia.



```
9.py U 10.py U X 11.py U •
5- Ejercicio bucles 2 > 10.py > ...
1  '''Escribe un programa que, dados dos números, uno real (base) y un entero positivo (exponente),
2  saque por pantalla el resultado de la potencia. No se puede utilizar el operador de potencia'''
3  base=int(input('Escriba la base de la potencia: '))
4  exponente=int(input('Escribe el exponente de la potencia: '))
5  i=0
6  total=1
7  while i<exponente:
8      total=total*base
9      i+=1
10 print("Resultado", total)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python
PS C:\ASIR\2 ASIR\Implantacion de Aplicaciones Web\UT3 - Diseño de documentos Web con HTML y CSS\Python> &
thon310/python.exe" "c:/ASIR/2 ASIR/Implantacion de Aplicaciones Web/UT3 - Diseño de documentos Web con HTML
ercicio bucles 2/10.py"
Escriba la base de la potencia: 5
Escribe el exponente de la potencia: 2
Resultado 25
```

Ejercicio 11

Algoritmo que muestre la tabla de multiplicar de los números 1,2,3,4 y 5.

EJERCICIOS DE REPETITIVAS

```
11.py U X
5- Ejercicio bucles 2 > 11.py > ...
1  '''Algoritmo que muestre la tabla de multiplicar de los números 1,2,3,4 y 5.'''
2  i = 1
3  while i <= 5:
4      print("Tabla del", i, ":")
5      numero = 0
6
7      while numero <=10:
8          resultado = i * numero
9          print(f"{i} x {numero} = {resultado}")
10         numero = numero + 1
11
12     print("-----")
13     i = i + 1

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

4 x 10 = 40
-----
Tabla del 5 :
5 x 0 = 0
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```