

Estructura de Datos

Manual

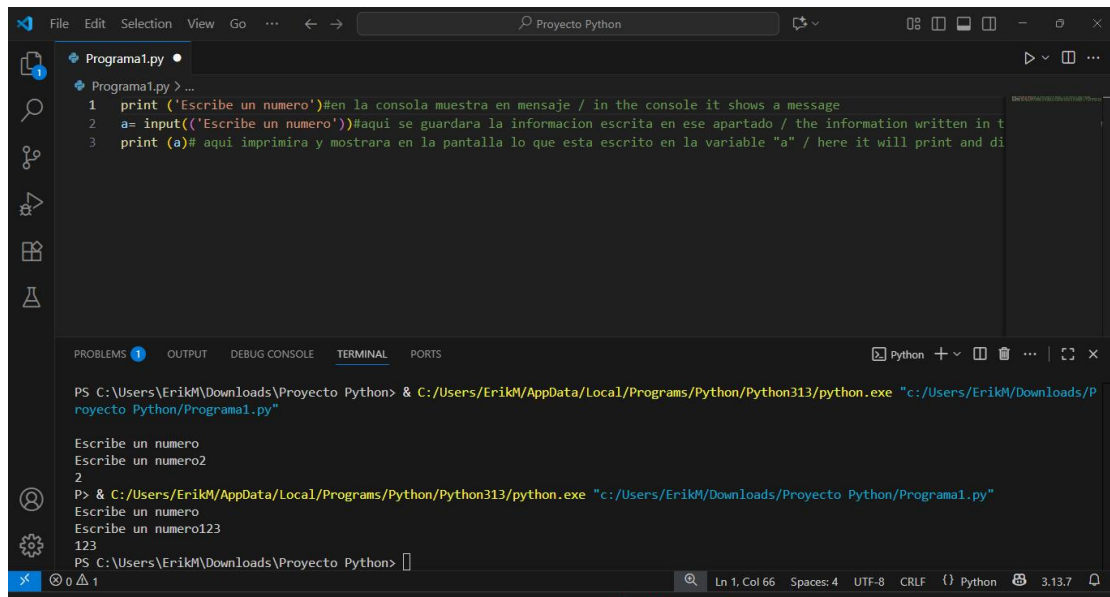


Erik Israel Mendez Gomez

Osvaldo Rene Roa

Informática 3

Practica 1



The image shows a Visual Studio Code editor window with a file named 'Programa1.py' open. The code in the editor is as follows:

```
1 print('Escribe un numero')#en la consola muestra en mensaje / in the console it shows a message
2 a= input(('Escribe un numero'))#aqui se guardara la informacion escrita en ese apartado / the information written in t
3 print(a)# aqui imprimira y mostrara en la pantalla lo que esta escrito en la variable "a" / here it will print and di
```

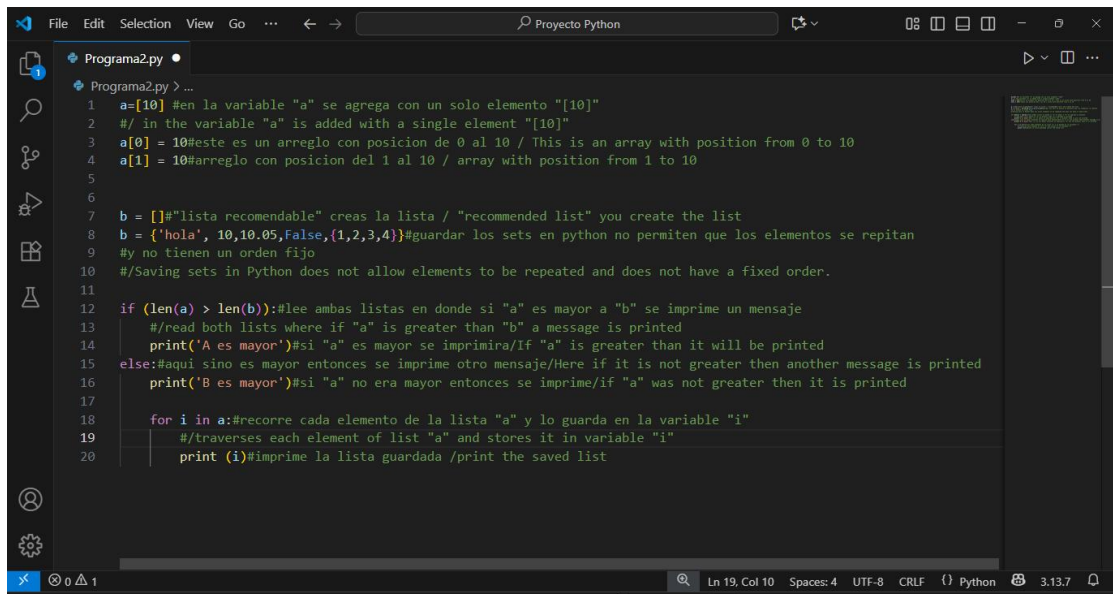
Below the editor, the 'TERMINAL' tab is active, showing the command prompt output:

```
PS C:\Users\ErikM\Downloads\Proyecto Python> & C:/Users/ErikM/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/ErikM/Downloads/P
royecto Python/Programa1.py"

Escribe un numero
Escribe un numero2
2
P> & C:/Users/ErikM/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/ErikM/Downloads/Proyecto Python/Programa1.py"
Escribe un numero
Escribe un numero123
123
PS C:\Users\ErikM\Downloads\Proyecto Python>
```

The status bar at the bottom indicates the current line and column (Ln 1, Col 66), encoding (UTF-8), line endings (CRLF), and the Python version (3.13.7).

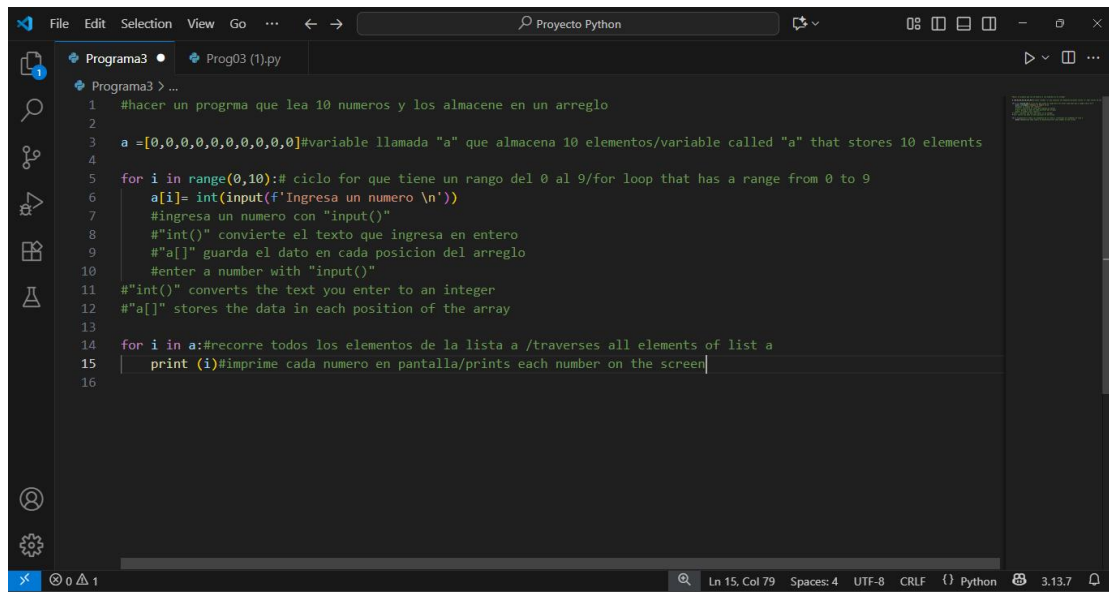
Practica 2



```
File Edit Selection View Go ... ← → Proyecto Python
Programa2.py
Programa2.py > ...
1 a=[10] #en la variable "a" se agrega con un solo elemento "[10]"
2 #/ in the variable "a" is added with a single element "[10]"
3 a[0] = 10#este es un arreglo con posicion de 0 al 10 / This is an array with position from 0 to 10
4 a[1] = 10#arreglo con posicion del 1 al 10 / array with position from 1 to 10
5
6
7 b = []#"lista recomendable" creas la lista / "recommended list" you create the list
8 b = {'hola', 10,10.05,False,(1,2,3,4)}#guardar los sets en python no permiten que los elementos se repitan
9 #y no tienen un orden fijo
10 #/Saving sets in Python does not allow elements to be repeated and does not have a fixed order.
11
12 if (len(a) > len(b)):#lee ambas listas en donde si "a" es mayor a "b" se imprime un mensaje
13     #/read both lists where if "a" is greater than "b" a message is printed
14     print('A es mayor')#si "a" es mayor se imprimira/If "a" is greater than it will be printed
15 else:#aqui sino es mayor entonces se imprime otro mensaje/Here if it is not greater then another message is printed
16     print('B es mayor')#si "a" no era mayor entonces se imprime/if "a" was not greater then it is printed
17
18     for i in a:#recorre cada elemento de la lista "a" y lo guarda en la variable "i"
19         #/traverses each element of list "a" and stores it in variable "i"
20         print (i)#imprime la lista guardada /print the saved list
```

Ln 19, Col 10 Spaces: 4 UTF-8 CRLF Python 3.13.7

Practica 3



The image shows a screenshot of a Visual Studio Code editor window. The title bar at the top reads "Proyecto Python". The editor has two tabs open: "Programa3" and "Prog03 (1).py". The active tab is "Programa3", which contains a Python script. The script is a program that reads 10 numbers from the user and stores them in an array. It uses a for loop to iterate from 0 to 9, prompting the user to enter a number. The input is converted to an integer and stored in the array. Finally, it uses another for loop to print each number on the screen. The code is written in Spanish and English, with comments in both languages. The status bar at the bottom indicates the current line and column (Ln 15, Col 79), the number of spaces (4), the encoding (UTF-8), the line ending (CRLF), the language (Python), and the version (3.13.7).

```
1 #hacer un progrma que lea 10 numeros y los almacene en un arreglo
2
3 a =[0,0,0,0,0,0,0,0,0,0]#variable llamada "a" que almacena 10 elementos/variable called "a" that stores 10 elements
4
5 for i in range(0,10):# ciclo for que tiene un rango del 0 al 9/for loop that has a range from 0 to 9
6     a[i]= int(input(f'Ingresa un numero \n'))
7     #ingresa un numero con "input()"
8     #"int()" convierte el texto que ingresa en entero
9     #"a[]" guarda el dato en cada posicion del arreglo
10    #enter a number with "input()"
11    #"int()" converts the text you enter to an integer
12    #"a[]" stores the data in each position of the array
13
14 for i in a:#recorre todos los elementos de la lista a /traverses all elements of list a
15     print (i)#imprime cada numero en pantalla/prints each number on the screen
16
```

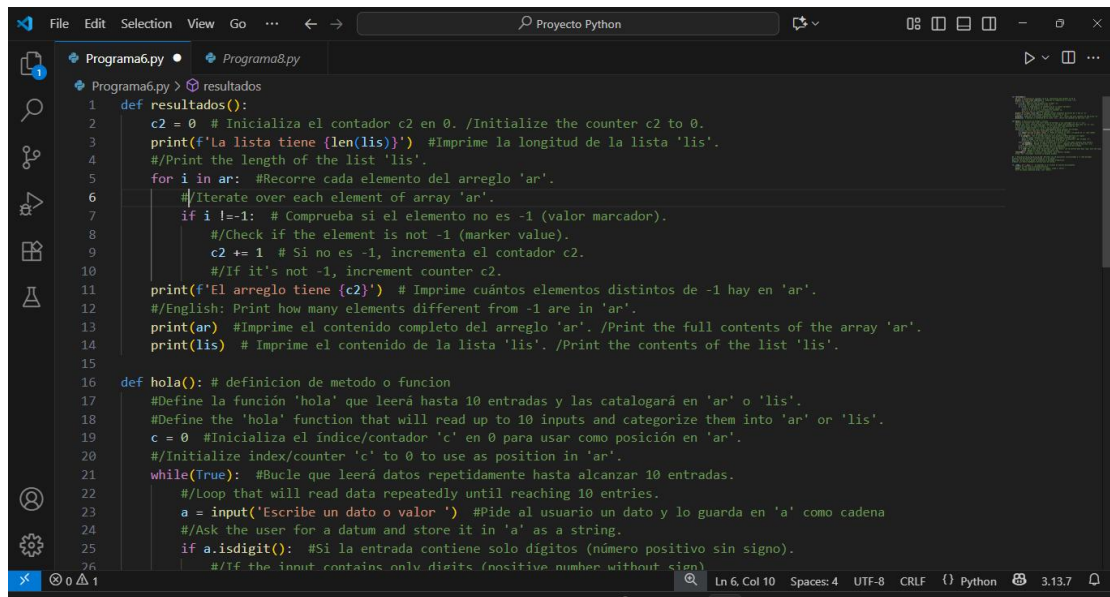
Practica 4

Practica 5

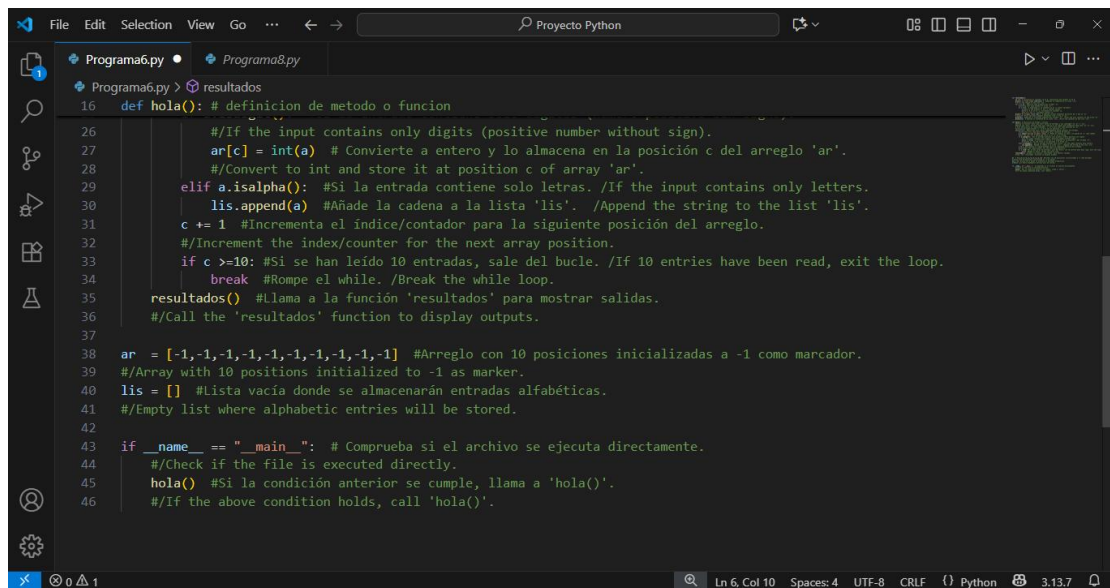
```
1 #Hacer un programa que lea 10 datos, si el dato es un numero se almacenara en un arreglo
2 # si es un caracter o caracteres se meta en una lista, cuando finalice el programa
3 # nos mostrara cuantos elementos numericos y cuantos caracteres hay en cada estructuras
4
5 car = 0 #Variable declared but unused
6 #/Variable declarada pero no utilizada
7 Arreglo = 0 # Counter for total entered elements
8 #/Contador para los elementos totales ingresados
9 n = 0 #Counter for total entered elements
10 #/ Contador para los elementos ingresados
11 a = [0,0,0,0,0,0,0,0,0,0] # Initialize list 'a' with one numeric element (10)
12 #/ Inicializa la lista 'a' con un elemento numérico (10)
13 l = [] # Initialize empty list 'l' to store alphabetic characters
14 #/ Inicializa la lista vacía 'l' para guardar caracteres alfabéticos
15
16 while (n < 10): # Loop until 10 valid elements are entered
17     #/ Bucle hasta que se ingresen 10 elementos válidos
18     b = input('Ingresa un dato: ') #Request input from user / Solicita un dato al usuario
19
20     if b.isdigit(): # Check if input is numeric /Verifica si el dato ingresado es numérico
21         a[Arreglo]=b #Add numeric input to list 'a' / Agrega el dato numérico a la lista 'a'
22         Arreglo +=1 #Increase numeric counter / Incrementa el contador de numéricos
23         n += 1 # Increment counter / Incrementa el contador
24     elif b.isalpha(): # Check if input is alphabetic
25         #/ Verifica si el dato ingresado es alfabético
26         l.append(b) # Add alphabetic input to list 'l'
```

```
22     Arreglo +=1 #Increase numeric counter / Incrementa el contador de numericos
23     n += 1 # Increment counter / Incrementa el contador
24     elif b.isalpha(): # Check if input is alphabetic
25         #/ Verifica si el dato ingresado es alfabético
26         l.append(b) # Add alphabetic input to list 'l'
27         #/ Agrega el dato alfabético a la lista 'l'
28         n += 1 # Increment counter / Incrementa el contador
29     else:
30         print('No valido') # Show error message for invalid input
31         #/ Muestra mensaje de error para datos no válidos
32
33 print(f'El total de elementos numericos guardados es {(Arreglo)}')
34 #Show total numeric elements stored / Muestra el total de elementos numéricos guardados
35 print(f'El total de caracteres guardados es {len(l)}')
36 # Show total alphabetic elements stored / Muestra el total de caracteres guardados
37 """print(f'Elementos numericos guardados {a}')
38 # Display list of numeric elements / Muestra la lista de elementos numéricos
39 print(f'Elementos de caracteres guardados {l}')
40 # Display list of alphabetic elements / Muestra la lista de elementos alfabéticos"""
```

Practica 6



```
File Edit Selection View Go ... Proyecto Python
Programa6.py Programa8.py
Programa6.py > resultados
1 def resultados():
2     c2 = 0 # Inicializa el contador c2 en 0. /Initialize the counter c2 to 0.
3     print(f'La lista tiene {len(lis)}') #Imprime la longitud de la lista 'lis'.
4     #/Print the length of the list 'lis'.
5     for i in ar: #Recorre cada elemento del arreglo 'ar'.
6         #/Iterate over each element of array 'ar'.
7         if i != -1: # Comprueba si el elemento no es -1 (valor marcador).
8             #/Check if the element is not -1 (marker value).
9             c2 += 1 # Si no es -1, incrementa el contador c2.
10            #/If it's not -1, increment counter c2.
11        print(f'El arreglo tiene {c2}') # Imprime cuántos elementos distintos de -1 hay en 'ar'.
12        #/English: Print how many elements different from -1 are in 'ar'.
13        print(ar) #Imprime el contenido completo del arreglo 'ar'. /Print the full contents of the array 'ar'.
14        print(lis) # Imprime el contenido de la lista 'lis'. /Print the contents of the list 'lis'.
15
16 def hola(): # definicion de metodo o funcion
17     #Define la función 'hola' que leerá hasta 10 entradas y las catalogará en 'ar' o 'lis'.
18     #Define the 'hola' function that will read up to 10 inputs and categorize them into 'ar' or 'lis'.
19     c = 0 #Inicializa el índice/contador 'c' en 0 para usar como posición en 'ar'.
20     #/Initialize index/counter 'c' to 0 to use as position in 'ar'.
21     while(True): #Bucle que leerá datos repetidamente hasta alcanzar 10 entradas.
22         #/Loop that will read data repeatedly until reaching 10 entries.
23         a = input('Escribe un dato o valor ') #Pide al usuario un dato y lo guarda en 'a' como cadena
24         #Ask the user for a datum and store it in 'a' as a string.
25         if a.isdigit(): #Si la entrada contiene solo dígitos (número positivo sin signo).
26             #/If the input contains only digits (positive number without sign).
```



```
File Edit Selection View Go ... Proyecto Python
Programa6.py Programa8.py
Programa6.py > resultados
16 def hola(): # definicion de metodo o funcion
26     #/If the input contains only digits (positive number without sign).
27     ar[c] = int(a) # Convierte a entero y lo almacena en la posición c del arreglo 'ar'.
28     #/Convert to int and store it at position c of array 'ar'.
29     elif a.isalpha(): #Si la entrada contiene solo letras. /If the input contains only letters.
30         lis.append(a) #Añade la cadena a la lista 'lis'. /Append the string to the list 'lis'.
31         c += 1 #Incrementa el índice/contador para la siguiente posición del arreglo.
32         #/Increment the index/counter for the next array position.
33         if c >= 10: #Si se han leído 10 entradas, sale del bucle. /If 10 entries have been read, exit the loop.
34             break #Rompe el while. /Break the while loop.
35     resultados() #Llama a la función 'resultados' para mostrar salidas.
36     #/Call the 'resultados' function to display outputs.
37
38     ar = [-1,-1,-1,-1,-1,-1,-1,-1,-1,-1] #Arreglo con 10 posiciones inicializadas a -1 como marcador.
39     #/Array with 10 positions initialized to -1 as marker.
40     lis = [] #Lista vacía donde se almacenarán entradas alfabéticas.
41     #/Empty list where alphabetic entries will be stored.
42
43     if __name__ == "__main__": # Comprueba si el archivo se ejecuta directamente.
44         #/Check if the file is executed directly.
45         hola() #Si la condición anterior se cumple, llama a 'hola()'.
46         #/If the above condition holds, call 'hola()'.

```


Practica 7

```
1 # Hacer un programa que lea nombre, edad, y sexo de 5 personas, estos elementos tienen que estar
2
3 def inicio(): #Define la función 'inicio' que controla el flujo principal.
4     #Define the 'inicio' function that controls the main flow.
5     c = 0 #Inicializa el contador 'c' en 0. /Initialize the counter 'c' to 0.
6     while (True): #Inicia un bucle infinito que repetirá la lectura hasta alcanzar 5 registros.
7         #Start an infinite loop that will repeat the reading until 5 records are reached.
8         nombre() #Llama a la función 'nombre' para leer y validar un nombre.
9         #Call the 'nombre' function to read and validate a name.
10        edad() #Llama a la función 'edad' para leer y validar una edad.
11        #Call the 'edad' function to read and validate an age.
12        sexo() #Llama a la función 'sexo' para leer y validar el sexo (M/F).
13        #Call the 'sexo' function to read and validate sex (M/F).
14        c+=1 #Incrementa el contador 'c' después de leer un registro completo.
15        #Increment the counter 'c' after reading a complete record.
16        if c >= 5: # Comprueba si se han leído 5 o más registros.
17            # Check if 5 or more records have been read.
18            break # Si el contador llega a 5, rompe el bucle y termina la función.
19            # If the counter reaches 5, break the loop and end the function.
20
21
22
23 def nombre(): #Define la función 'nombre' para pedir y validar un nombre.
24     #Define the 'nombre' function to request and validate a name.
25     while(True): #Bucle que se repetirá hasta recibir un nombre válido.
26         #Loop that repeats until a valid name is received
```

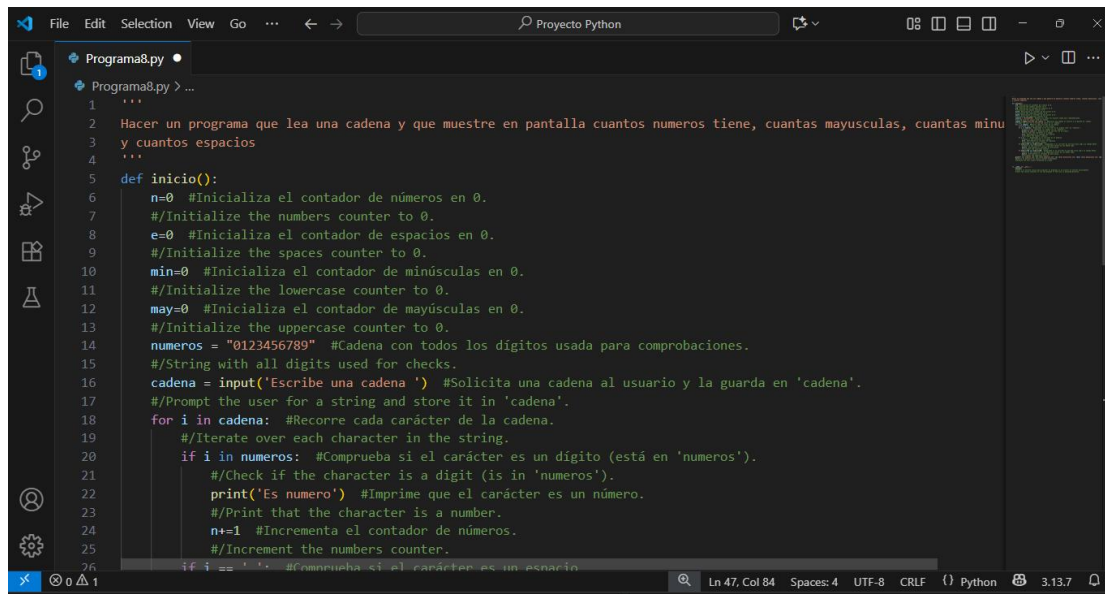
```
26     #Loop that repeats until a valid name is received.
27     n=input('Escribe un nombre ') #Pide al usuario que escriba un nombre y lo guarda en 'n'.
28     #Ask the user to write a name and store it in 'n'.
29     if n.isalpha(): #Verifica que 'n' solo contenga letras (sin espacios ni dígitos).
30         #Verify that 'n' contains only letters (no spaces or digits).
31         lis.append(n) #Si es válido, añade el nombre a la lista 'lis'.
32         #If valid, append the name to the list 'lis'.
33         break #Sale del bucle una vez añadido el nombre.
34     #Exit the loop once the name has been appended.
35     else:
36         print('Nombre no valido') #Mensaje si el nombre no es válido.
37         #Message shown if the name is not valid.
38
39
40 def edad(): #Define la función 'edad' para pedir y validar la edad.
41     #Define the 'edad' function to request and validate age.
42     while(True): #Bucle que se repetirá hasta recibir una edad válida.
43         #Loop that repeats until a valid age is received.
44         e=input('Escribe una edad ') #Pide al usuario que escriba una edad y la guarda en 'e'.
45         #Ask the user to write an age and store it in 'e'.
46         if e.isdigit(): #Verifica que 'e' esté compuesta solo por dígitos.
47             #Verify that 'e' is composed only of digits.
48             lis.append(e) # Si es válido, añade la edad a la lista 'lis'.
49             #If valid, append the age to the list 'lis'.
```



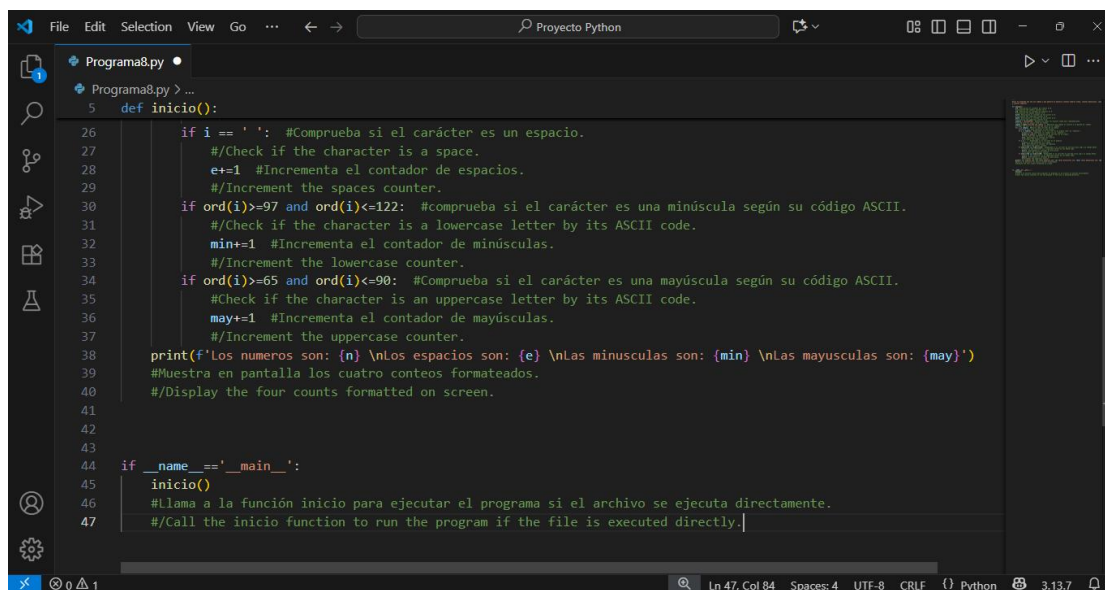
```
40 def edad(): #Define la función 'edad' para pedir y validar la edad.
41     break # Sale del bucle una vez añadida la edad.
42     #/Exit the loop once the age has been appended.
43 else:
44     print('Edad no valida') # Mensaje si la edad no es válida.
45     #/Message shown if the age is not valid.
46
47 def sexo(): # Define la función 'sexo' para pedir y validar el sexo (M/F).
48     #/Define the 'sexo' function to request and validate sex (M/F).
49     while(True): # Bucle que se repetirá hasta recibir un sexo válido.
50         #/Loop that repeats until a valid sex is received.
51         s=input('Escribe un sexo (M/F) ') #Pide al usuario que indique el sexo y lo guarda en 's'
52         #/Ask the user to indicate sex and store it in 's'.
53         if s=='M' or s=='F': #Verifica que la entrada sea exactamente 'M' o 'F'.
54             #/Verify that the input is exactly 'M' or 'F'.
55             lis.append(s) # Si es válido, añade el sexo a la lista 'lis'.
56             #/If valid, append the sex to the list 'lis'.
57             break #Sale del bucle una vez añadido el sexo.
58             #/Exit the loop once the sex has been appended.
59         else:
60             print('Sexo no valido') # Mensaje si la entrada no es 'M' ni 'F'.
61             #/ Message shown if the input is not 'M' or 'F'.
62
63 def resultados(): #Define la función 'resultados' (en el código actual imprime desde la lista).
```

```
72
73 def resultados(): #Define la función 'resultados' (en el código actual imprime desde la lista).
74     #/Define the 'resultados' function (currently prints from the list).
75     i = 0 #Inicializa el índice 'i' en 0.
76     #/Initialize the index 'i' to 0.
77     while(True): #Bucle infinito que imprimirá elementos de 'lis' (tal como está, no incrementa
78         #/ Infinite loop that will print elements of 'lis' (as written, it does not increment 'i'
79         print(lis[i]) #Imprime el elemento de la lista en la posición 'i'.
80         #/Print the element of the list at position 'i'.
81
82
83
84
85
86 lis = [] #Inicializa la lista vacía 'lis' donde se almacenarán nombres, edades y sexos.
87 #/Initialize the empty list 'lis' where names, ages and sexes will be stored.
88 inicio() #Llama a la función 'inicio' para comenzar el proceso de ingreso.
89 #/Call the 'inicio' function to start the input process.
```

Practica 8



```
1 '''
2 Hacer un programa que lea una cadena y que muestre en pantalla cuantos numeros tiene, cuantas mayusculas, cuantas minu
3 y cuantos espacios
4 '''
5 def inicio():
6     n=0 #Inicializa el contador de números en 0.
7     #/Initialize the numbers counter to 0.
8     e=0 #Inicializa el contador de espacios en 0.
9     #/Initialize the spaces counter to 0.
10    min=0 #Inicializa el contador de minúsculas en 0.
11    #/Initialize the lowercase counter to 0.
12    may=0 #Inicializa el contador de mayúsculas en 0.
13    #/Initialize the uppercase counter to 0.
14    numeros = "0123456789" #Cadena con todos los dígitos usada para comprobaciones.
15    #/String with all digits used for checks.
16    cadena = input('Escribe una cadena ') #Solicita una cadena al usuario y la guarda en 'cadena'.
17    #/Prompt the user for a string and store it in 'cadena'.
18    for i in cadena: #Recorre cada carácter de la cadena.
19        #/Iterate over each character in the string.
20        if i in numeros: #Comprueba si el carácter es un dígito (está en 'numeros').
21            #/Check if the character is a digit (is in 'numeros').
22            print('Es numero') #Imprime que el carácter es un número.
23            #/Print that the character is a number.
24            n+=1 #Incrementa el contador de números.
25            #/Increment the numbers counter.
26    if i == ' ': #Comprueba si el carácter es un espacio
```

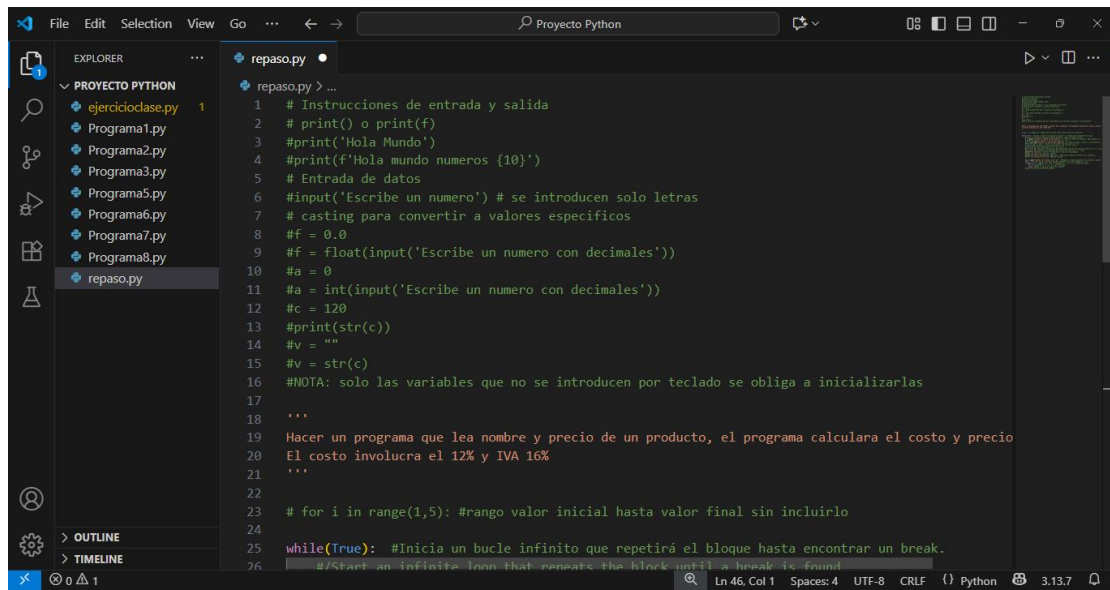


```
26    if i == ' ': #Comprueba si el carácter es un espacio.
27        #/Check if the character is a space.
28        e+=1 #Incrementa el contador de espacios.
29        #/Increment the spaces counter.
30    if ord(i)>97 and ord(i)<=122: #Comprueba si el carácter es una minúscula según su código ASCII.
31        #/Check if the character is a lowercase letter by its ASCII code.
32        min+=1 #Incrementa el contador de minúsculas.
33        #/Increment the lowercase counter.
34    if ord(i)>=65 and ord(i)<=90: #Comprueba si el carácter es una mayúscula según su código ASCII.
35        #/Check if the character is an uppercase letter by its ASCII code.
36        may+=1 #Incrementa el contador de mayúsculas.
37        #/Increment the uppercase counter.
38    print(f'Los numeros son: {n} \nLos espacios son: {e} \nLas minusculas son: {min} \nLas mayusculas son: {may}')
39    #Muestra en pantalla los cuatro conteos formateados.
40    #/Display the four counts formatted on screen.
41
42
43
44 if __name__=='__main__':
45     inicio()
46     #Llama a la función inicio para ejecutar el programa si el archivo se ejecuta directamente.
47     #/Call the inicio function to run the program if the file is executed directly.
```

Practica 9

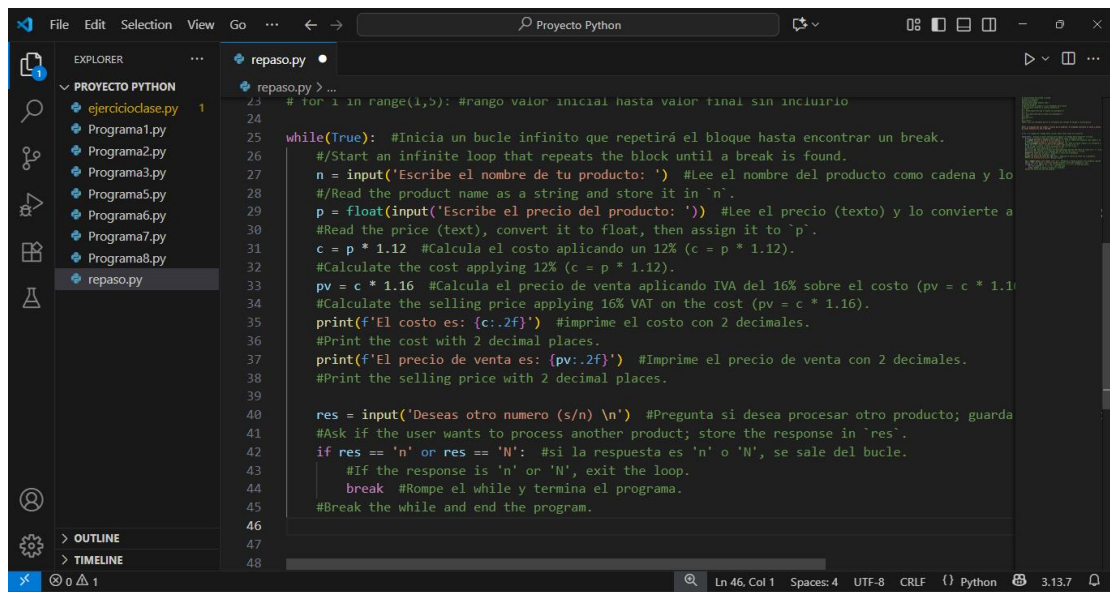
Practica 10

Repaso 1



The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying a project named 'PROYECTO PYTHON'. The file 'repaso.py' is selected. The main editor window shows the following code:

```
1 # Instrucciones de entrada y salida
2 # print() o print(f)
3 #print('Hola Mundo')
4 #print(f'Hola mundo numeros {10}')
5 # Entrada de datos
6 #input('Escribe un numero') # se introducen solo letras
7 # casting para convertir a valores especificos
8 #f = 0.0
9 #f = float(input('Escribe un numero con decimales'))
10 #a = 0
11 #a = int(input('Escribe un numero con decimales'))
12 #c = 120
13 #print(str(c))
14 #v = ""
15 #v = str(c)
16 #NOTA: solo las variables que no se introducen por teclado se obliga a inicializarlas
17
18 '''
19 Hacer un programa que lea nombre y precio de un producto, el programa calculara el costo y precio
20 El costo involucra el 12% y IVA 16%
21 '''
22
23 # for i in range(1,5): #rango valor inicial hasta valor final sin incluirlo
24
25 while(True): #Inicia un bucle infinito que repetirá el bloque hasta encontrar un break.
26     #Start an infinite loop that repeats the block until a break is found
```



The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying a project named 'PROYECTO PYTHON'. The file 'repaso.py' is selected. The main editor window shows the following code:

```
23 # for i in range(1,5): #rango valor inicial hasta valor final sin incluirlo
24
25 while(True): #Inicia un bucle infinito que repetirá el bloque hasta encontrar un break.
26     #Start an infinite loop that repeats the block until a break is found.
27     n = input('Escribe el nombre de tu producto: ') #Lee el nombre del producto como cadena y lo
28     #Read the product name as a string and store it in 'n'.
29     p = float(input('Escribe el precio del producto: ')) #Lee el precio (texto) y lo convierte a
30     #Read the price (text), convert it to float, then assign it to 'p'.
31     c = p * 1.12 #Calcula el costo aplicando un 12% (c = p * 1.12).
32     #Calculate the cost applying 12% (c = p * 1.12).
33     pv = c * 1.16 #Calcula el precio de venta aplicando IVA del 16% sobre el costo (pv = c * 1.16).
34     #Calculate the selling price applying 16% VAT on the cost (pv = c * 1.16).
35     print(f'El costo es: {c:.2f}') #imprime el costo con 2 decimales.
36     #Print the cost with 2 decimal places.
37     print(f'El precio de venta es: {pv:.2f}') #Imprime el precio de venta con 2 decimales.
38     #Print the selling price with 2 decimal places.
39
40     res = input('Deseas otro numero (s/n) \n') #Pregunta si desea procesar otro producto; guarda
41     #Ask if the user wants to process another product; store the response in 'res'.
42     if res == 'n' or res == 'N': #si la respuesta es 'n' o 'N', se sale del bucle.
43         #If the response is 'n' or 'N', exit the loop.
44         break #Rompe el while y termina el programa.
45     #Break the while and end the program.
46
47
48
```

Repaso 2

```
1 a = 1#Asigna el valor entero 1 a la variable 'a' (coeficiente cuadrático).
2 #Assign the integer 1 to variable 'a' (quadratic coefficient).
3 b = 2# Asigna el valor entero 2 a la variable 'b' (coeficiente lineal).
4 #Assign the integer 2 to variable 'b' (linear coefficient).
5 c = -15#Asigna el valor entero -15 a la variable 'c' (término independiente).
6 #Assign the integer -15 to variable 'c' (constant term).
7 p = 0#Inicializa la variable 'p' con 0; aquí se usará para b**2.
8 #Initialize variable 'p' to 0; here it will be used for b**2.
9 m = 0#Inicializa la variable 'm' con 0; aquí se usará para 4*a*c.
10 #Initialize variable 'm' to 0; here it will be used for 4*a*c.
11 r = 0#Inicializa la variable 'r' con 0; representará el discriminante (b^2 - 4ac).
12 #Initialize variable 'r' to 0; it will represent the discriminant (b^2 - 4ac).
13 ra = 0.0#Inicializa 'ra' como float; se usará para la raíz cuadrada del discriminante.
14 #Initialize 'ra' as float; it will be used for the square root of the discriminant.
15 d = 0.0#Inicializa 'd' como float; se usará para el denominador 2*a.
16 #Initialize 'd' as float; it will be used for the denominator 2*a.
17 x1 = 0.0#Inicializa 'x1' como float; contendrá la primera raíz.
18 #Initialize 'x1' as float; it will hold the first root.
19 x2 = 0.0#Inicializa 'x2' como float; contendrá la segunda raíz.
20 #Initialize 'x2' as float; it will hold the second root.
21 p = b**2#Calcula b al cuadrado y lo asigna a 'p'.
22 #Compute b squared and assign it to 'p'.
23 m = 4*a*c#Calcula 4*a*c y lo asigna a 'm'.
24 #Compute 4*a*c and assign it to 'm'.
25 r = p - m#Calcula el discriminante r = b^2 - 4ac.
26 #Compute the discriminant r = b^2 - 4ac.
```

```
25 r = p - m#Calcula el discriminante r = b^2 - 4ac.
26 #Compute the discriminant r = b^2 - 4ac.
27 if r > 0:#Si el discriminante es mayor que 0 hay dos raíces reales distintas.
28 #If the discriminant is greater than 0 there are two distinct real roots.
29     print('Si se puede')#Imprime mensaje indicando que sí es posible calcular raíces reales.
30     #Print a message indicating it's possible to compute real roots.
31     ra = r ** (1/2)#Calcula la raíz cuadrada del discriminante y la guarda en 'ra'.
32     #Calculate the square root of the discriminant and store it in 'ra'.
33     d = 2*a#Calcula el denominador 2*a y lo guarda en 'd'.
34     #Compute the denominator 2*a and store it in 'd'.
35     x1 = (-b + ra) / d#Calcula la primera raíz usando la fórmula (-b + sqrt(r)) / (2a).
36     #Calculate the first root using the formula (-b + sqrt(r)) / (2a).
37     x2 = (-b - ra) / d#Calcula la segunda raíz usando la fórmula (-b - sqrt(r)) / (2a).
38     #Calculate the second root using the formula (-b - sqrt(r)) / (2a).
39     print(f'El valor de x1 es {x1:.2f} y de x2 es {x2:.2f}')
40     #Imprime los valores de x1 y x2 formateados con 2 decimales.
41     #Print the values of x1 and x2 formatted with 2 decimal places.
42 else:#Si r no es mayor que 0 (es 0 o negativo) entra aquí.
43 #If r is not greater than 0 (it is 0 or negative) enter here.
44     print('No se puede')#Imprime mensaje indicando que no es posible calcular dos raíces reales d
45     #Print a message indicating it's not possible to compute two distinct real roots.
```


Repaso 3

The screenshot shows the Visual Studio Code editor with the file 'repaso3.py' open. The Explorer sidebar on the left shows a project named 'PROYECTO PYTHON' containing several files: 'ejercicioclase.py', 'Programa1.py', 'Programa2.py', 'Programa3.py', 'Programa5.py', 'Programa6.py', 'Programa7.py', 'Programa8.py', 'repaso.py', 'repaso2.py', and 'repaso3.py'. The main editor area displays the code for the 'validar' function. The function is designed to validate a parameter 'a' by attempting to convert it to an integer and then a float. It includes comments in Spanish and English, and uses try-except blocks to handle 'ValueError' exceptions. The status bar at the bottom indicates the cursor is at line 90, column 59, with 4 spaces, UTF-8 encoding, CRLF line endings, and the Python language mode.

```
1 def validar(a):#Define la función 'validar' que recibe el parámetro 'a'.
2     #Define the function 'validar' that receives the parameter 'a'.
3     c = 0#Inicializa la variable entera 'c' con 0.
4     #Initialize the integer variable 'c' to 0.
5     d = 0.0#Inicializa la variable decimal 'd' con 0.0.
6     #Initialize the float variable 'd' to 0.0.
7     try:#Intenta ejecutar el bloque que sigue (intentar conversión a entero).
8         #Try to execute the following block (attempt conversion to int).
9         c = int(a)#Convierte 'a' a entero y lo asigna a 'c'.
10        #Convert 'a' to an integer and assign it to 'c'.
11        print('Es un valor numerico sin decimales')#Muestra mensaje indicando que es un número sin
12        #Print message indicating it is a number without decimals.
13    except ValueError:#Si ocurre un ValueError (no se puede convertir a int), entra aquí.
14        #If a ValueError occurs (cannot convert to int), enter here.
15        print('No es un valor numerico sin decimales')#Muestra mensaje indicando que no es un núm
16        #Print message indicating it is not an integer number.
17
18    try:#Intenta ejecutar el bloque que sigue (intentar conversión a float).
19        #Try to execute the following block (attempt conversion to float).
20        d = float(a)#Convierte 'a' a float y lo asigna a 'd'.
21        #Convert 'a' to a float and assign it to 'd'.
22        print('Es un valor numerico con decimales')#Muestra mensaje indicando que es un número con
23        #Print message indicating it is a number with decimals.
24    except ValueError:#Si ocurre un ValueError (no se puede convertir a float), entra aquí.
25        #If a ValueError occurs (cannot convert to float), enter here.
26        print('No es un valor numerico con decimales')#Muestra mensaje indicando que no es un núm
```

This screenshot shows the same Visual Studio Code editor with 'repaso3.py' open, but the view has scrolled down to the 'listas' function. The 'validar' function is partially visible at the top. The 'listas' function is designed to recognize the data type of a parameter 'd' and return it in its original type. It uses try-except blocks to handle 'ValueError' exceptions when converting 'd' to an integer. The status bar at the bottom shows the cursor at line 90, column 59, with 4 spaces, UTF-8 encoding, CRLF line endings, and the Python language mode.

```
24 except ValueError:#Si ocurre un ValueError (no se puede convertir a float), entra aquí.
25     #If a ValueError occurs (cannot convert to float), enter here.
26     print('No es un valor numerico con decimales')#Muestra mensaje indicando que no es un núm
27     #Print message indicating it is not a number with decimals.
28
29 def listas(d):#Define la función 'listas' que intenta reconocer el tipo de dato y devolverlo en s
30     #Define the function 'listas' that tries to recognize the data type and return it in its type
31     en = 0#Inicializa la variable entera 'en' con 0.
32     #Initialize the integer variable 'en' to 0.
33     dec = 0.0#Inicializa la variable decimal 'dec' con 0.0.
34     #Initialize the float variable 'dec' to 0.0.
35
36     try:#Intenta convertir 'd' a entero.
37         #Try to convert 'd' to an integer.
38         en = int(d)#Asigna el entero convertido a 'en'.
39         #Assign the converted integer to 'en'.
40         return en#Devuelve el entero si la conversión fue exitosa.
41         #Return the integer if conversion succeeded.
42    except ValueError:#Si falla la conversión a entero, se captura aquí.
43        #If integer conversion fails, it is caught here.
44        print('No es un entero')#Informa que no es un entero.
45        #Inform that it is not an integer.
46
47    try:#Intenta convertir 'd' a float.
```



```
29 def listas(d):#Define la función 'listas' que intenta reconocer el tipo de dato y devolverlo en s
46
47     try:#Intenta convertir 'd' a float.
48         #Try to convert 'd' to a float.
49         dec = float(d)#Asigna el float convertido a 'dec'.
50         #Assign the converted float to 'dec'.
51         return dec#Devuelve el float si la conversión fue exitosa.
52         #Return the float if conversion succeeded.
53     except ValueError:#Si falla la conversión a float, se captura aquí.
54         #If float conversion fails, it is caught here.
55         print('No es un numero con decimales')#Informa que no es un número con decimales.
56         #Inform that it is not a number with decimals.
57
58     return d#Si no es ni entero ni float, devuelve el valor original (como string).
59     #If it's neither int nor float, return the original value (as string).
60
61 def leer():#Define la función 'leer' que solicita un dato al usuario y lo procesa.
62     #Define the function 'leer' that requests data from the user and processes it.
63     d = input('Escribe un dato: ')#Lee una entrada del usuario y la guarda en 'd'.
64     #Read user input and store it in 'd'.
65     dato = listas(d)
66     #Llama a 'listas' para convertir 'd' a su tipo si es posible; guarda el resultado en 'dato'.
67     #Call 'listas' to convert 'd' to its type if possible; store the result in 'dato'.
68     lista.append(dato)
69     #Añade el dato (convertido o no) a la lista global 'lista'.
70     #Append the data (converted or not) to the global list 'lista'.
```

```
69     #Añade el dato (convertido o no) a la lista global 'lista'.
70     #Append the data (converted or not) to the global list 'lista'.
71
72     #Inicializa la lista vacía global donde se almacenarán los datos.
73     #Initialize the empty global list where data will be stored.
74     lista = []
75
76     if __name__ == '__main__':#Comprueba si el script está siendo ejecutado directamente (no importad
77         #Check if the script is being run directly (not imported).
78         while True:#Bucle infinito que permite leer repetidamente datos del usuario.
79             #Infinite loop that allows repeatedly reading user data.
80             leer()#Llama a la función 'leer' para solicitar y almacenar un dato.
81             #Call the 'leer' function to request and store one datum.
82
83
84             res = input('Deseas otro dato (S/N): ')#Pregunta al usuario si desea ingresar otro dato y
85             #Ask the user if they want to enter another datum and save the response in 'res'.
86             if res == 'n' or res == 'N':#Si la respuesta es 'n' o 'N', termina el bucle.
87                 #If the response is 'n' or 'N', break the loop.
88                 print(lista)#Imprime la lista con todos los datos almacenados.
89                 #Print the list with all stored data.
90                 break#Rompe el bucle 'while' terminando la ejecución principal.
                #Break the 'while' loop ending main execution.
```