

4. Estructuras de Control en Python - Actividades.

Los siguientes ejercicios están tomados del capítulo 4 del libro digital “Introducción a la Programación con Python 3.”

Ejercicio 57 (página 100)

Diseña un programa que lea un número flotante por teclado y muestre por pantalla el mensaje «El número es negativo» sólo si el número es menor que cero.

Ejercicio 58(página 100)

Diseña un programa que lea un número flotante por teclado y muestre por pantalla el mensaje «El número es positivo» sólo si el número es mayor o igual que cero.

Ejercicio 59(página 100)

Diseña un programa que lea la edad de dos personas y diga quién es más joven, la primera o la segunda. Ten en cuenta que ambas pueden tener la misma edad. En tal caso, hazlo saber con un mensaje adecuado.

Ejercicio 60 (página 100)

Diseña un programa que lea un carácter de teclado y muestre por pantalla el mensaje «Es paréntesis» sólo si el carácter leído es un paréntesis abierto o cerrado.

Ejercicio 63(página 101)

Diseña un programa que, dado un número entero, muestre por pantalla el mensaje «El número es par» cuando el número sea par y el mensaje «El número es impar» cuando sea impar.

Ejercicio 64 (página 101)

Diseña un programa que, dado un número entero, determine si este es el doble de un número impar. (Ejemplo: 14 es el doble de 7, que es impar).

Ejercicio 65 (página 101)

Diseña un programa que, dados dos números enteros, muestre por pantalla uno de estos mensajes: «El segundo es el cuadrado del primero», «El segundo es menor que el cuadrado del primero» o bien «El segundo es mayor que el cuadrado del primero», dependiendo de la verificación de la condición correspondiente al significado de cada mensaje.

Ejercicio 66 (página 101)

Un capital de C euros a un interés del x por cien anual durante n años se convierte en: $C \cdot (1 + x/100)^n$ euros Diseña un programa Python que solicite la cantidad C, el interés x y el número de años n y calcule el capital final **solo si x es una cantidad positiva**.

Ejercicio 67 página 101)

Realiza un programa que calcule el desglose mínimo en billetes y monedas de una cantidad exacta de euros. Hay billetes de 500, 200, 100, 50, 20, 10 y 5 € y monedas de 2 y 1 €. Por ejemplo, si deseamos conocer el desglose de 434 €, el programa mostrará por pantalla el siguiente resultado:

- ♠ 2 billetes de 200 €
- ♠ 1 billete de 20 €
- ♠ 1 billete de 10 €
- ♠ 2 monedas de 2 €

Ejercicio 70 (página 105)

Diseña un programa Python que lea un carácter cualquiera desde el teclado, y muestre el mensaje «Es una mayúscula» cuando el carácter sea una letra mayúscula y el mensaje «Es una minúscula» cuando sea una minúscula. En cualquier otro caso, no mostrará mensaje alguno.

Ejercicio 71 (página 105)

Amplía la solución al ejercicio anterior para que cuando el carácter introducido no sea una letra muestre el mensaje «No es una letra».

Ejercicio 72 (página 105)

Amplía el programa del ejercicio anterior para que pueda identificar las letras eñe minúscula y mayúscula

Ejercicio 73 (página 105)

Modifica el programa que propusiste como solución al ejercicio 65 sustituyendo todas las condiciones que sea posible por cláusulas else de condiciones anteriores.

Ejercicio 78 (página 112)

Diseña un programa que calcule el máximo de 5 números enteros.

Ejercicio 79 (página 112)

Diseña un programa que calcule la menor de cinco palabras dadas; es decir, la primera palabra de las cinco en orden alfabético. Aceptaremos que las mayúsculas son «alfabéticamente» menores que las minúsculas, de acuerdo con la tabla ASCII.

Ejercicio 81 (página 113)

Diseña un programa que, dados cinco números enteros, determine cuál de los cuatro últimos números es más cercano al primero. (Por ejemplo, si el usuario introduce los números 2, 6, 4, 1 y 10, el programa responderá que el número más cercano al 2 es el 1).

Ejercicio 85 (página 114)

Diseña un programa que, dado un número real que debe representar la calificación numérica de un examen, proporcione la calificación cualitativa correspondiente al número dado.

Ejercicio 86 (página 114)

Diseña un programa que, dado un carácter cualquiera, lo identifique como vocal minúscula, vocal mayúscula, consonante minúscula, consonante mayúscula u otro tipo de carácter. (Considera únicamente letras del alfabeto inglés).

Ejercicio 90 (página 117)

Modifica el programa circulo.py para que también acepte letras mayúsculas.

Ejercicio 100 (página 123)

Implementa un programa que muestre todos los múltiplos de 6 entre 6 y 150, ambos inclusive.

Ejercicio 101 (página 123)

Implementa un programa que muestre todos los múltiplos de n entre n y $m \cdot n$, ambos inclusive, donde n y m son números introducidos por el usuario.

Ejercicio 102 (página 123)

Implementa un programa que muestre todos los números potencia de 2 entre 2^0 y 2^{30} , ambos inclusive

Ejercicio 104 (página 125) y 105.

Diseña un programa que calcule $\sum_{i=n}^m i$ donde n y m son números enteros que deberá introducir el usuario por teclado.

Ejercicio 106 (página 125)

Queremos hacer un programa que calcule el factorial de un número entero positivo. El factorial de n se denota con n!, pero no existe ningún operador Python que permita efectuar este cálculo directamente. Sabiendo que $n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot (n - 1) \cdot n$

y que $0! = 1$, haz un programa que pida el valor de n y muestre por pantalla el resultado de calcular n!.

Ejercicio 109 (página 127)

Diseña un programa que solicite la lectura de un número entre 0 y 10 (ambos inclusive). Si el usuario teclea un número fuera del rango válido, el programa solicitará nuevamente la introducción del valor cuantas veces sea necesario.

Ejercicio 111 (página 128)

Haz un programa que vaya leyendo números y mostrándolos por pantalla hasta que el usuario introduzca un número negativo. En ese momento, el programa mostrará un mensaje de despedida y finalizará su ejecución.

Ejercicio 112 (página 128)

Haz un programa que vaya leyendo números hasta que el usuario introduzca un número negativo. En ese momento, el programa mostrará por pantalla el número mayor de cuantos ha visto.

Ejercicio 113 (página 129)

¿Es correcto este otro programa? ¿En qué se diferencia del anterior? ¿Cuál te parece mejor (si es que alguno de ellos te parece mejor)?

```
circulo.py
1 from math import pi
2
3 radio = float(input('Dame el radio de un círculo: '))
4
5 opción = ''
6 while opción < 'a' or opción > 'c':
7     print('Escoge una opción: ')
8     print('a) Calcular el diámetro.')
9     print('b) Calcular el perímetro.')
10    print('c) Calcular el área.')
11    opción = input('Teclea a, b o c y pulsa el retorno de carro: ')
12    if opción < 'a' or opción > 'c':
13        print('Solo hay tres opciones: a, b o c. Tú has tecleado', opción)
14
15 if opción == 'a':
16     diámetro = 2 * radio
```

```
17     print('El diámetro es', diámetro)
18 elif opción == 'b':
19     perímetro = 2 * pi * radio
20     print('El perímetro es', perímetro)
21 elif opción == 'c':
22     área = pi * radio ** 2
23     print('El área es', área)
```

Ejercicio 114 (página 130)

Modifica el programa para que pida el valor del radio cada vez que se solicita efectuar un nuevo cálculo.