

## TALLERES - Curso Python Essentials 1

Realizar los siguientes programas en lenguaje Python.

El Instructor indicará procedimientos de entrega, ya sea individual o grupal.

### Ejercicio No.1

Escribe un programa en una línea de código, que imprima la siguiente frase en dos líneas:

"Python es un lenguaje de programación poderoso."

"Puedes crear aplicaciones increíbles con él."

### Ejercicio No.2

Escribe un programa que imprima tu nombre en dos líneas. En la primera línea, muestra tu nombre y, en la segunda línea, muestra tu apellido.

### Ejercicio No.3

Escribe un programa que imprima un triángulo formado por asteriscos (\*) en tres líneas.

### Ejercicio No.4

Escribe un programa que imprima las palabras "Hola", "Mundo" y "Python" en la misma línea, separadas por guiones.

### Ejercicio No.5

Escribe un programa que imprima los números del 1 al 5 en una línea, separados por espacios, y luego imprima un punto al final.

**Ejercicio No.6**

Escribe un programa que imprima tu nombre y apellido en una sola línea, separados por un guion y finalizados con un signo de exclamación.

**Ejercicio No.7**

Escribe un programa que imprima las letras de tu nombre en una sola línea, separadas por espacios y finalizadas con un símbolo de interrogación.

**Ejercicio No.8**

Escribe un programa que imprima el valor entero 42.

Escribe un programa que imprima el valor octal 0o75 en decimal.

Escribe un programa que imprima el valor hexadecimal 0x1F en decimal.

Escribe un programa que imprima el valor flotante 3.14.

Escribe un programa que imprima el valor booleano True.

Escribe un programa que imprima la cadena "Hola, mundo!".

Escribe un programa que imprima la cadena vacía "".

**Ejercicio No.9**

Imprime en pantalla el resultado de multiplicar el número 3 por el número 2 y sumarle 5.

**Ejercicio No.10**

Imprime en pantalla el resultado de restar el número 1 al número 4 y multiplicarlo por el número 3.

### Ejercicio No.11

Imprime en pantalla el resultado de elevar el número 2 a la potencia del número 3 y restarle 1.

### Ejercicio No.12

Imprime en pantalla el resultado de dividir el número 10 entre la suma del número 2 y el número 3.

### Ejercicio #13

Escribir un programa que solicite al usuario que ingrese su nombre. El nombre se debe almacenar en una variable llamada nombre. A continuación, se debe mostrar en pantalla el texto “Ahora estás en la matrix, [usuario]”, donde “[usuario]” se reemplazará por el nombre que el usuario haya ingresado.

*Ejemplo de ejecución:*

```
Tu nombre: Patricia
Ahora estás en la matrix, Patricia
```

### Ejercicio #14

Escribir un programa que solicite al usuario ingresar un número con decimales y almacenarlo en una variable. A continuación, el programa debe solicitar al usuario que ingrese un número entero y guardarlo en otra variable. En una tercera variable se deberá guardar el resultado de la suma de los dos números ingresados por el usuario. Por último, se debe mostrar en pantalla el texto “El resultado de la suma es [suma]”, donde “[suma]” se reemplazará por el resultado de la operación.

*Ejemplo de ejecución:*

```
Primer número: 14.2
Segundo número: 19
El resultado de la suma es 33.2
```

## Ejercicio #15

Escribir un programa que solicite al usuario dos números y los almacene en dos variables. En otra variable, almacena el resultado de la suma de esos dos números y luego muestra ese resultado en pantalla.

A continuación, el programa debe solicitar al usuario que ingrese un tercer número, el cual se debe almacenar en una nueva variable. Por último, muestra en pantalla el resultado de la multiplicación de este nuevo número por el resultado de la suma anterior.

*Ejemplo de ejecución:*

```
Ingresá un número: 1
Ingresá otro número: 2
Suma: 3
Ingresá un nuevo número: 3
Multiplicación de la suma por el último número: 9
```

## Ejercicio #16

Escribir un programa que solicite al usuario ingresar la cantidad de kilómetros recorridos por una motocicleta y la cantidad de litros de combustible que consumió durante ese recorrido. Mostrar el consumo de combustible por kilómetro.

*Ejemplo de ejecución:*

```
Kilómetros recorridos: 260
Litros de combustible gastados: 12.5
El consumo por kilómetro es de 20.8
```

### Ejercicio #17

Escribir un programa que solicite al usuario el ingreso de una temperatura en escala Fahrenheit (debe permitir decimales) y le muestre el equivalente en grados Celsius. La fórmula de conversión que se usa para este cálculo es:  $_{Celsius} = (5/9) * (Fahrenheit - 32)$

*Ejemplo de ejecución:*

```
Ingresá una temperatura expresada en Farenheit: 75
23.888888888888889
```

### Ejercicio #18

Escribir un programa para solicitar al usuario el ingreso de un número entero y que luego imprima un valor de verdad dependiendo de si el número es par o no. Hay que recordar que un número es par si el resto, al dividirlo por 2, es 0.

*Ejemplo de ejecución:*

```
Número entero: 7254
True
```

### Ejercicio #19

Realizar un programa que solicite un número al usuario, este evalúe si el número es par o impar, y despliegue un mensaje si es par así: El número es par. Y si este número llega a ser impar despliegue un mensaje así: El número es impar.

## Ejercicio #20

Escribir un programa que solicite al usuario el ingreso de dos números diferentes y muestre en pantalla al mayor de los dos.

*Ejemplo de ejecución:*

```
Un número: 592
Otro número distinto: 1726
1726 es mayor
```

## Ejercicio #21

Escribir un programa para solicitar al usuario tres números y mostrar en pantalla al menor de los tres.

*Ejemplo de ejecución:*

```
Primer número: 20
Segundo número: 30
Tercer número: 10
Menor: 10
```

## Ejercicio #22

Escribir un programa que le solicite al usuario un número entero y muestre todos los números correlativos entre el 1 y el número ingresado por el usuario.

```
Ingresá un número: 3
1
2
3
```

## Ejercicio #23

Escribir un programa que, dada una frase por el usuario, muestre la cantidad total de vocales (tanto mayúsculas como minúsculas) que contiene.

*Ejemplo de ejecución:*

**Frase:** *Verde que te quiero verde*  
**Vocales:** 11

## Ejercicio #24

Escribir un programa que permita al usuario ingresar los montos de las compras de un cliente (se desconoce la cantidad de datos que cargará, la cual puede cambiar en cada ejecución), cortando el ingreso de datos cuando el usuario ingrese el monto 0. Si ingresa un monto negativo, no se debe procesar y se debe pedir que ingrese un nuevo monto. Al finalizar, informar el total a pagar teniendo que cuenta que, si las ventas superan el monto total de 1000, se le debe aplicar un 10% de descuento.

*Ejemplo de ejecución:*

**Monto de una venta:** \$ 100  
**Monto de una venta:** \$ 300  
**Monto de una venta:** \$ -1  
**Monto no válido.**  
**Monto de una venta:** \$ 2000  
**Monto de una venta:** \$ 0  
**Monto total a pagar:** \$ 2160.0

### Ejercicio #25

Escribir un programa que, dado un número entero por el usuario (guardado como int), muestre la suma de todos sus dígitos. Hay que recordar que vas a necesitar obtener cada uno de los dígitos por separado para poder sumarlos entre sí.

*Ejemplo de ejecución:*

```
Escribí un número: 7124
Suma de los dígitos: 14
```

### Ejercicio #26

Escribe un programa que solicite al usuario ingresar un año y luego determine e imprima si es un año bisiesto o no. (Un año es bisiesto si es divisible entre 4, pero no si es divisible entre 100, excepto si también es divisible entre 400).

### Ejercicio #27

Escribe un programa que solicite al usuario ingresar una frase y luego cuente e imprima cuántas vocales contiene la frase utilizando un bucle while.

### Ejercicio #28

el programa solicita al usuario información sobre el cliente y la fecha de la factura. Luego, se inicia un ciclo repetitivo para que el usuario pueda ingresar los productos de la factura junto con sus precios y cantidades. El ciclo se detiene cuando el usuario ingresa la palabra "fin". A medida que se ingresan los productos, se calcula el subtotal y se utilizan ciclos repetitivos adicionales para solicitar y mostrar los productos en la factura.. Finalmente, se calcula el impuesto y el total, y se imprime la factura con todos los detalles.



### Ejercicio #29

Escribe un programa que solicite al usuario ingresar números enteros positivos, y el programa seguirá solicitando números hasta que se ingrese un número negativo. Luego, mostrará la suma de todos los números ingresados.

### Ejercicio #30

Realiza un programa de gestión de tareas simple donde el usuario puede agregar, listar y completar tareas.

Se presenta un menú de opciones al usuario. La opción "1" permite agregar una tarea a la lista de tareas, la opción "2" muestra la lista de tareas, la opción "3" permite completar una tarea seleccionada y la opción "4" permite salir del programa. Se utiliza una lista para almacenar las tareas y se utilizan estructuras de control, como ciclos for y condicionales if-elif-else, para realizar diferentes acciones según la opción seleccionada por el usuario.

### Ejercicio #31

Escribe una función llamada `calcular_promedio` que tome una lista de números como argumento y devuelva el promedio de los números en la lista.

### Ejercicio #32

Escribe una función llamada `es_primo` que tome un número como argumento y devuelva `True` si es un número primo, y `False` en caso contrario.

### Ejercicio #33

Escribe una función llamada `es_anagrama` que tome dos palabras como argumentos y devuelva `True` si son anagramas (contienen las mismas letras en diferente orden), y `False` en caso contrario.

**Ejercicio #34**

Escribe una función llamada `interseccion_listas` que tome dos listas como argumento y devuelva una nueva lista que contenga los elementos que están presentes en ambas listas.

**Ejercicio #35**

Escribe una función llamada `eliminar_repetidos` que tome una lista como argumento y devuelva una nueva lista que contenga los elementos únicos de la lista original, en el mismo orden de aparición.

**Ejercicio #36**

Escribe una función llamada `encontrar_estudiante` que tome un diccionario donde las claves son los nombres de los estudiantes y los valores son sus edades. La función debe buscar un estudiante por su nombre y devolver su edad.

**Ejercicio #37**

Escribe una función llamada `convertir_entero` que tome una cadena como argumento y intente convertirla a un número entero. Maneja la excepción de valor no válido utilizando un bloque `try` y `except`.

**Ejercicio #38**

Escribe una función llamada `obtener_elemento` que tome una lista y un índice como argumentos y trate de obtener el elemento en la posición dada. Maneja la excepción de índice fuera de rango utilizando un bloque `try` y `except`.

### Ejercicio #39

Escribe una función llamada `sumar_elementos` que tome una lista de números como argumento y calcule la suma de todos los elementos. Maneja la excepción de tipos de datos no numéricos utilizando un bloque `try` y `except`.

### Ejercicio #40

Escribe una función llamada `acceder_diccionario` que tome un diccionario y una clave como argumentos, y trate de acceder al valor correspondiente. Maneja la excepción de clave no encontrada utilizando un bloque `try` y `except`.