



Programación 3



Docente: ING: Darwin Asdrúbal Chamba Flores

Autor: Ariana González Pinto

Carrera: Tecnología Superior en Desarrollo de Software

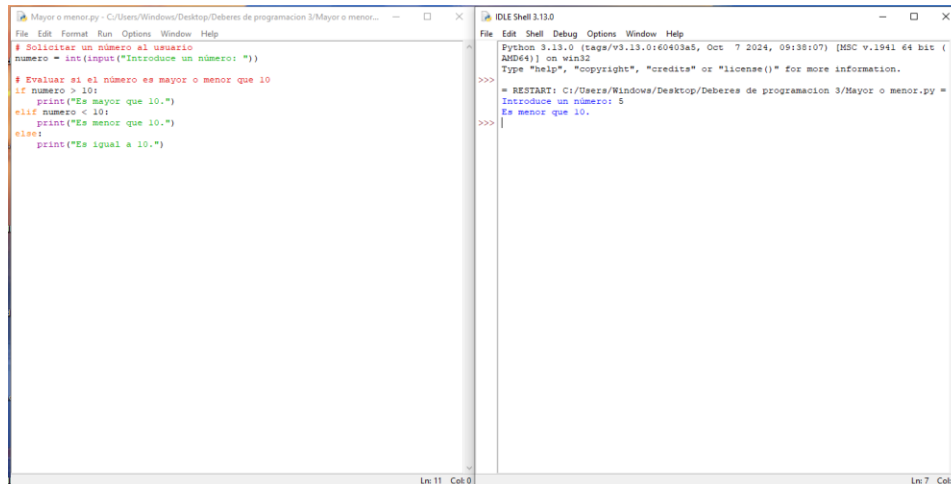




Resolver

1. Mayor o menor:

- Escribe un programa que solicite un número y determine si es mayor o menor que 10.
- Ejemplo: Entrada: 5 → Salida: *"Es menor que 10"*.



```
File Edit Format Run Options Window Help
# Solicitar un número al usuario
numero = int(input("Introduce un número: "))

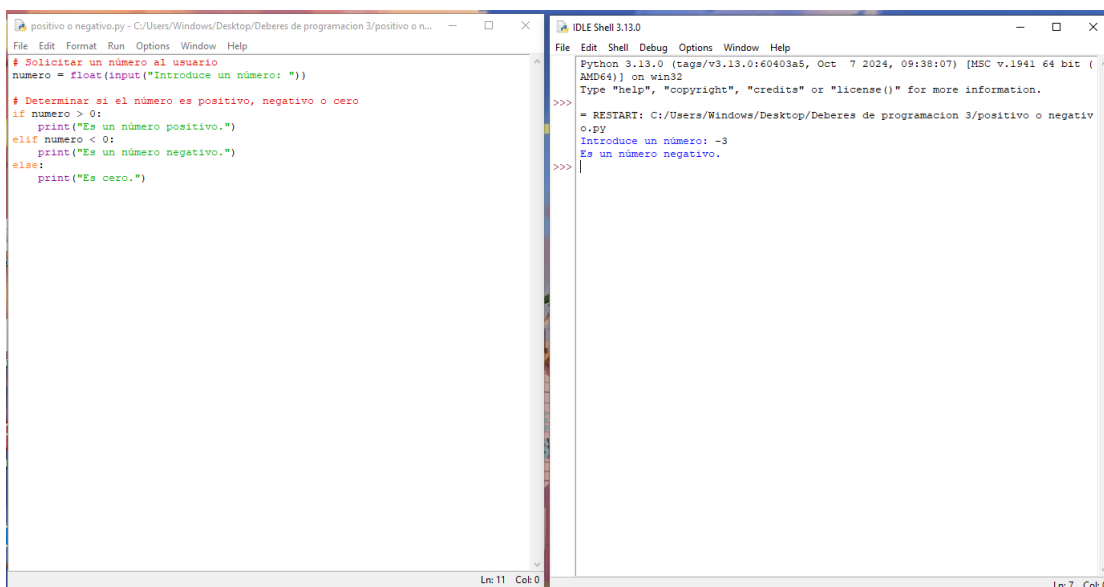
# Evaluar si el número es mayor o menor que 10
if numero > 10:
    print("Es mayor que 10.")
elif numero < 10:
    print("Es menor que 10.")
else:
    print("Es igual a 10.")

Ln: 11 Col: 0
```

```
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Mayor o menor.py =
Introduce un número: 5
Es menor que 10.
>>>
```

2. Número positivo o negativo:

- Solicita al usuario un número y determina si es positivo, negativo o cero.
- Ejemplo: Entrada: -3 → Salida: *"Es un número negativo"*.



```
File Edit Format Run Options Window Help
# Solicitar un número al usuario
numero = float(input("Introduce un número: "))

# Determinar si el número es positivo, negativo o cero
if numero > 0:
    print("Es un número positivo.")
elif numero < 0:
    print("Es un número negativo.")
else:
    print("Es cero.")

Ln: 11 Col: 0
```

```
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/positivo o negativ
o.py
Introduce un número: -3
Es un número negativo.
>>>
```





3. Par o impar:

- Solicita un número al usuario y determina si es par o impar.
- Ejemplo: Entrada: 4 → Salida: "Es par".

```
File Edit Format Run Options Window Help
# Solicitar un número al usuario
numero = int(input("Introduce un número: "))

# Determinar si el número es par o impar
if numero % 2 == 0:
    print("Es par.")
else:
    print("Es impar.")

Ln: 9 Col: 0
```

4. Aprobado o reprobado:

- Solicita la calificación de un estudiante y determina si está aprobado (mayor o igual a 7) o reprobado.
- Ejemplo: Entrada: 6.5 → Salida: "Reprobado".

```
File Edit Format Run Options Window Help
# Solicitar la calificación al usuario
calificacion = float(input("Introduce la calificación del estudiante: "))

# Determinar si está aprobado o reprobado
if calificacion >= 7:
    print("Aprobado.")
else:
    print("Reprobado.")

Ln: 9 Col: 0
```



Ejercicios intermedios

5. Descuento en una tienda:

- Una tienda ofrece un 20% de descuento si el cliente gasta más de \$100. Escribe un programa que calcule el monto final.
- Ejemplo: Entrada: \$120 → Salida: *"Monto final: \$96"*.

```
descuentos tienda.py - C:/Users/Windows/Desktop/Deberes de programacion 3/descuentos t...
File Edit Format Run Options Window Help
# Solicitar el monto gastado por el cliente
monto = float(input("Introduce el monto gastado: $"))

# Calcular el monto final con o sin descuento
if monto > 100:
    monto_final = monto * 0.8 # Aplicar un descuento del 20%
    print(f"Monto final: ${monto_final:.2f}")
else:
    print(f"Monto final: ${monto:.2f}")

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/descuentos tienda.py
Introduce el monto gastado: $120
Monto final: $96.00
>>>
```

6. Edad para votar:

- Solicita la edad del usuario y determina si es elegible para votar (mayor o igual a 18 años).
- Ejemplo: Entrada: 17 → Salida: *"No puedes votar"*.



```
Edad para votar.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Edad para votar...
File Edit Format Run Options Window Help
# Solicitar la edad del usuario
edad = int(input("Introduce tu edad: "))

# Determinar si es elegible para votar
if edad >= 18:
    print("Puedes votar.")
else:
    print("No puedes votar.")

Ln: 9 Col: 0

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Edad para votar.py
Introduce tu edad: 28
Puedes votar.
>>>
```

7. Mayor de tres números:

- Solicita tres números y determina cuál es el mayor.
- Ejemplo: Entrada: 4, 9, 2 → Salida: *"El número mayor es 9"*.

```
Mayor de tres numeros.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Mayor ...
File Edit Format Run Options Window Help
# Solicitar tres números al usuario
numero1 = float(input("Introduce el primer número: "))
numero2 = float(input("Introduce el segundo número: "))
numero3 = float(input("Introduce el tercer número: "))

# Determinar cuál es el número mayor
mayor = max(numero1, numero2, numero3)

# Mostrar el resultado
print(f"El número mayor es {mayor}.")

Ln: 11 Col: 0

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Mayor de tres numeros.py
Introduce el primer número: 4
Introduce el segundo número: 9
Introduce el tercer número: 2
El número mayor es 9.0.
>>>
```



8. Clasificación de edades:

- Solicita una edad y clasifica al usuario como niño (0-12), adolescente (13-17) o adulto (18+).
- Ejemplo: Entrada: 15 → Salida: *"Eres adolescente"*.

The image shows a screenshot of a Python script and its execution. On the left, a text editor window titled 'clasificacion de edades.py' contains the following code:

```
File Edit Format Run Options Window Help
# Solicitar la edad del usuario
edad = int(input("Introduce tu edad: "))

# Clasificar según la edad
if 0 <= edad <= 12:
    print("Eres niño.")
elif 13 <= edad <= 17:
    print("Eres adolescente.")
elif edad >= 18:
    print("Eres adulto.")
else:
    print("Edad no válida.")
```

On the right, the IDLE Shell window shows the execution of the script. It displays the prompt 'Introduce tu edad: 28' and the output 'Eres adulto.'.

9. Calculadora básica:

- Solicita dos números y una operación (+, -, *, /) y realiza el cálculo correspondiente.
- Ejemplo: Entrada: 3, 2, '+' → Salida: *"Resultado: 5"*.



```
calculadora basica.py - C:/Users/Windows/Desktop/Deberes de programacion 3/calculadora ...
File Edit Format Run Options Window Help
# Solicitar dos números y la operación
numero1 = float(input("Introduce el primer número: "))
numero2 = float(input("Introduce el segundo número: "))
operacion = input("Introduce la operación (+, -, *, /): ")

# Realizar el cálculo según la operación
if operacion == "+":
    resultado = numero1 + numero2
elif operacion == "-":
    resultado = numero1 - numero2
elif operacion == "*":
    resultado = numero1 * numero2
elif operacion == "/":
    if numero2 != 0:
        resultado = numero1 / numero2
    else:
        resultado = "Error: División por cero."
else:
    resultado = "Operación no válida."

# Mostrar el resultado
print(f"Resultado: {resultado}")

Ln: 23 Col: 0
```

```
IDLE Shell 3.13.0
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/calculadora basica .py
Introduce el primer número: 4
Introduce el segundo número: 7
Introduce la operación (+, -, *, /): +
Resultado: 11.0
>>>
```

10. Determinar un año bisiesto:

- Solicita un año y determina si es bisiesto (divisible entre 4 pero no entre 100, excepto si es divisible entre 400).
- Ejemplo: Entrada: 2024 → Salida: *"Es bisiesto"*.

```
Año viciesto.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Año viciesto.py (3...
File Edit Format Run Options Window Help
# Solicitar el año al usuario
anio = int(input("Introduce un año: "))

# Determinar si el año es bisiesto
if (anio % 4 == 0 and anio % 100 != 0) or (anio % 400 == 0):
    print("Es bisiesto.")
else:
    print("No es bisiesto.")

Ln: 9 Col: 0
```

```
IDLE Shell 3.13.0
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Año viciesto.py
Introduce un año: 2024
Es bisiesto.
>>>
```



11. Validar contraseñas:

- Escribe un programa que solicite una contraseña y valide si es correcta (ejemplo: contraseña fija es 12345).
- Ejemplo: Entrada: 12345 → Salida: *"Acceso concedido"*.

The screenshot shows a Python script named 'validar contraseña.py' and its execution in the IDLE Shell. The script defines a correct password '12345', prompts the user for a password, and checks if it matches. The shell shows the user inputting '12345' and receiving the output 'Acceso concedido.'

```
validar contraseña.py - C:/Users/Windows/Desktop/Deberes de programacion 3/validar contr...
File Edit Format Run Options Window Help
# Definir la contraseña correcta
contraseña_correcta = "12345"

# Solicitar la contraseña al usuario
contraseña = input("Introduce la contraseña: ")

# Validar la contraseña
if contraseña == contraseña_correcta:
    print("Acceso concedido.")
else:
    print("Acceso denegado.")
|

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/validar contraseña.PY
Introduce la contraseña: 12345
Acceso concedido.
>>>
```

12. Juego de números:

- Genera un número aleatorio entre 1 y 10 y solicita al usuario que adivine el número. Usa `if` para verificar si acertó o no.
- Ejemplo: Entrada: 5 → Salida: *"¡Felicidades, acertaste!"* o *"Intenta de nuevo."*.

The screenshot shows a Python script named 'Juego de numeros.py' and its execution in the IDLE Shell. The script generates a random number between 1 and 10, prompts the user to guess it, and checks if the guess is correct. The shell shows the user inputting '1' and receiving the output '¡Felicidades, acertaste!'.

```
Juego de numeros.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Juego de nu...
File Edit Format Run Options Window Help
import random

# Generar un número aleatorio entre 1 y 10
numero_aleatorio = random.randint(1, 10)

# Solicitar al usuario que adivine el número
adivinanza = int(input("Adivina el número entre 1 y 10: "))

# Verificar si el usuario acertó
if adivinanza == numero_aleatorio:
    print("¡Felicidades, acertaste!")
else:
    print("Intenta de nuevo.")

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Juego de numeros.py
Adivina el número entre 1 y 10: 1
¡Felicidades, acertaste!
>>>
```




13. Calcular el signo zodiacal:

- Solicita el día y mes de nacimiento y determina el signo zodiacal del usuario.
- Ejemplo: Entrada: 22, marzo → Salida: *"Tu signo es Aries"*.

The image shows a screenshot of a Python script and its execution. The script, named 'calculo de signo zodiacal.py', is located at 'C:/Users/Windows/Desktop/Deberes de programacion 3/calculo...'. It prompts the user to enter their birth day and month, then determines the zodiac sign based on a series of conditional statements. The execution in the IDLE Shell shows the user entering '1998' for the day and '04' for the month, resulting in the output 'Tu signo es Piscis'.

```
File Edit Format Run Options Window Help
# Solicitar el día y mes de nacimiento
dia = int(input("Introduce el día de tu nacimiento: "))
mes = input("Introduce el mes de tu nacimiento: ").lower()

# Determinar el signo zodiacal
if (mes == "marzo" and dia >= 21) or (mes == "abril" and dia <= 19):
    signo = "Aries"
elif (mes == "abril" and dia >= 20) or (mes == "mayo" and dia <= 20):
    signo = "Tauro"
elif (mes == "mayo" and dia >= 21) or (mes == "junio" and dia <= 20):
    signo = "Géminis"
elif (mes == "junio" and dia >= 21) or (mes == "julio" and dia <= 22):
    signo = "Cáncer"
elif (mes == "julio" and dia >= 23) or (mes == "agosto" and dia <= 22):
    signo = "Leo"
elif (mes == "agosto" and dia >= 23) or (mes == "septiembre" and dia <= 22):
    signo = "Virgo"
elif (mes == "septiembre" and dia >= 23) or (mes == "octubre" and dia <= 22):
    signo = "Libra"
elif (mes == "octubre" and dia >= 23) or (mes == "noviembre" and dia <= 21):
    signo = "Escorpio"
elif (mes == "noviembre" and dia >= 22) or (mes == "diciembre" and dia <= 21):
    signo = "Sagitario"
elif (mes == "diciembre" and dia >= 22) or (mes == "enero" and dia <= 19):
    signo = "Capricornio"
elif (mes == "enero" and dia >= 20) or (mes == "febrero" and dia <= 18):
    signo = "Acuario"
else:
    signo = "Piscis"

# Mostrar el signo zodiacal
print(f"Tu signo es {signo}.")

Ln: 33 Col: 0
```

```
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/calculo de signo zodiacal.py
Introduce el día de tu nacimiento: 1998
Introduce el mes de tu nacimiento: 04
Tu signo es Piscis.
>>>

Ln: 8 Col: 0
```

14. Sistema de calificaciones:

- Solicita una calificación numérica y devuelve la letra correspondiente:
 - 90-100: A.
 - 80-89: B.
 - 70-79: C.
 - 60-69: D.
 - Menor a 60: F.
- Ejemplo: Entrada: 85 → Salida: *"Tu calificación es B"*.



The screenshot shows two windows from the IDLE Python environment. The left window, titled 'Sistema de calificaciones.py', contains a Python script that prompts the user for a numerical grade and then outputs a corresponding letter grade (A, B, C, D, or F) based on a set of if-elif-else conditions. The right window, titled 'IDLE Shell 3.13.0', shows the execution of the script. It displays the prompt 'Introduce tu calificación: 90' and the output 'Tu calificación es A.'

```
# Sistema de calificaciones.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Siste...
File Edit Format Run Options Window Help
# Solicitar la calificación numérica
calificacion = float(input("Introduce tu calificación: "))

# Determinar la letra correspondiente
if 90 <= calificacion <= 100:
    letra = "A"
elif 80 <= calificacion <= 89:
    letra = "B"
elif 70 <= calificacion <= 69:
    letra = "C"
elif 60 <= calificacion <= 59:
    letra = "D"
else:
    letra = "F"

# Mostrar el resultado
print(f"Tu calificación es {letra}.")

Ln: 18 Col: 0
```

```
IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> = RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Sistema de calificaciones.py
Introduce tu calificación: 90
Tu calificación es A.

>>>
```

Ln: 7 Col: 0

15. Control de acceso:

- Solicita un nombre de usuario y contraseña, y valida si ambos son correctos. Permite tres intentos antes de bloquear el acceso.
- Ejemplo: Entrada: Usuario: admin, Contraseña: 1234 → Salida: *"Bienvenido, admin."*

The screenshot shows two windows from the IDLE Python environment. The left window, titled 'control de acceso.py', contains a Python script that defines correct username and password, sets a maximum of 3 attempts, and uses a while loop to prompt the user for credentials. If the credentials are correct, it prints a welcome message and breaks the loop. If incorrect, it decrements the attempt count and prints a denial message. If attempts are exhausted, it prints a blocked access message. The right window, titled 'IDLE Shell 3.13.0', shows the execution of the script. It displays the prompts 'Introduce tu nombre de usuario: admin' and 'Introduce tu contraseña: 1234', followed by the output 'Bienvenido, admin.'

```
control de acceso.py - C:/Users/Windows/Desktop/Deberes de programacion 3/control de ac...
File Edit Format Run Options Window Help
# Definir el usuario y contraseña correctos
usuario_correcto = "admin"
contraseña_correcta = "1234"

# Intentos restantes
intentos = 3

# Bucle para permitir 3 intentos
while intentos > 0:
    # Solicitar nombre de usuario y contraseña
    usuario = input("Introduce tu nombre de usuario: ")
    contraseña = input("Introduce tu contraseña: ")

    # Validar si ambos son correctos
    if usuario == usuario_correcto and contraseña == contraseña_correcta:
        print(f"Bienvenido, {usuario}.")
        break # Salir del bucle si el acceso es correcto
    else:
        intentos -= 1
        if intentos > 0:
            print(f"Acceso denegado. Te quedan {intentos} intentos.")
        else:
            print("Acceso bloqueado. Has agotado todos los intentos.")

Ln: 24 Col: 0
```

```
IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> = RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/control de acceso.py
Introduce tu nombre de usuario: admin
Introduce tu contraseña: 1234
Bienvenido, admin.

>>>
```

Ln: 8 Col: 0



Bucles

1. Imprimir números del 1 al 10:

- Usa un bucle para mostrar los números del 1 al 10 en la consola.
- **Salida esperada:** 1, 2, 3, ..., 10.

```
File Edit Format Run Options Window Help
# Usar un bucle para mostrar los números del 1 al 10
for i in range(1, 11):
    print(i, end=" ", " if i < 10 else "" )
|

Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Imprimir numeros d
el 1 al 10.py
1, 2, 3, 4, 5, 6, 7, 8, 9, 10
>>>
```

2. Suma de los primeros 10 números:

- Calcula y muestra la suma de los números del 1 al 10.
- **Salida esperada:** 55.



```
File Edit Format Run Options Window Help
# Inicializar la variable suma
suma = 0

# Usar un bucle para sumar los números del 1 al 10
for i in range(1, 11):
    suma += i

# Mostrar el resultado
print(suma)
```

```
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Suma de 10 numeros .PY
55
>>>
```

3. Tabla de multiplicar:

- Solicita un número al usuario y muestra su tabla de multiplicar del 1 al 10.
- **Ejemplo:** Entrada: 5 → Salida: 5 x 1 = 5, ..., 5 x 10 = 50.

```
File Edit Format Run Options Window Help
# Solicitar un número al usuario
numero = int(input("Introduce un número: "))

# Mostrar la tabla de multiplicar del 1 al 10
for i in range(1, 11):
    print(f"{numero} x {i} = {numero * i}")
```

```
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Tabla de multiplicar .PY
Introduce un número: 7
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
7 x 6 = 42
7 x 7 = 49
7 x 8 = 56
7 x 9 = 63
7 x 10 = 70
>>>
```



4. Contar números pares:

- Imprime todos los números pares entre 1 y 20.
- **Salida esperada:** 2, 4, 6, ..., 20.

The screenshot shows a Python IDE with two windows. The left window, titled 'contar numeros pares.py', contains the following code:

```
# Usar un bucle para imprimir los números pares entre 1 y 20
for i in range(2, 21, 2):
    print(i, end=" ", " if i < 20 else "")
```

The right window, titled 'IDLE Shell 3.13.0', shows the output of the script:

```
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/contar numeros pares.py
2, 4, 6, 8, 10, 12, 14, 16, 18, 20
>>>
```

5. Contador regresivo:

- Imprime los números del 10 al 1 en orden descendente.
- **Salida esperada:** 10, 9, 8, ..., 1.



```
File Edit Format Run Options Window Help
# Usar un bucle para imprimir los números del 10 al 1
for i in range(10, 0, -1):
    print(i, end=" ", " if i > 1 else "")

Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/contador regresivo
.PV
10, 9, 8, 7, 6, 5, 4, 3, 2, 1
>>>
```

Ejercicios intermedios

6. Factorial de un número:

- Calcula el factorial de un número ingresado por el usuario (n!).
- **Ejemplo:** Entrada: 5 → Salida: 120.

```
File Edit Format Run Options Window Help
# Solicitar un número al usuario
numero = int(input("Introduce un número: "))

# Inicializar el factorial como 1
factorial = 1

# Calcular el factorial usando un bucle
for i in range(1, numero + 1):
    factorial *= i

# Mostrar el resultado
print(f"El factorial de {numero} es {factorial}.")

Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/factorial de un numero.py
Introduce un número: 5
El factorial de 5 es 120.
>>>
```



7. Números primos:

- Encuentra e imprime todos los números primos entre 1 y 50.
- **Salida esperada:** 2, 3, 5, 7, ..., 47.

The image shows a screenshot of a Python IDE with two windows. The left window, titled 'numeros primos.py', contains the following Python code:

```
# Función para verificar si un número es primo
def es_primo(numero):
    if numero < 2:
        return False
    for i in range(2, int(numero ** 0.5) + 1):
        if numero % i == 0:
            return False
    return True

# Imprimir todos los números primos entre 1 y 50
for num in range(1, 51):
    if es_primo(num):
        print(num, end=" ", " if num < 50 else "")
```

The right window, titled 'IDLE Shell 3.13.0', shows the output of the script after execution:

```
Python 3.13.0 (tags/v3.13.0:60403a8, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/numeros primos.py
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47,
>>>
```

8. Sumar dígitos:

- Solicita un número entero y calcula la suma de sus dígitos.
- **Ejemplo:** Entrada: 123 → Salida: 6 (1 + 2 + 3).



```
sumar digitos.py - C:/Users/Windows/Desktop/Deberes de programacion 3/sumar digitos.py ...
File Edit Format Run Options Window Help
# Solicitar un número entero
numero = int(input("Introduce un número entero: "))

# Inicializar la suma de los dígitos
suma_digitos = 0

# Recorrer cada dígito del número y sumarlo
for digito in str(abs(numero)): # Usar abs para manejar números negativos
    suma_digitos += int(digito)

# Mostrar el resultado
print(f"La suma de los dígitos es {suma_digitos}.")

Ln: 13 Col: 0
```

```
IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/sumar digitos.py =
Introduce un número entero: 123
La suma de los dígitos es 6.
>>>
```

9. Invertir un número:

- Solicita un número entero y muestra su versión invertida.
- **Ejemplo:** Entrada: 1234 → Salida: 4321.

```
Invertir un numero.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Invertir un n...
File Edit Format Run Options Window Help
# Solicitar un número entero
numero = int(input("Introduce un número entero: "))

# Invertir el número convirtiéndolo en cadena y luego en entero
numero_invertido = int(str(abs(numero))[-1::-1]) # Usar abs para manejar números

# Si el número era negativo, agregar el signo negativo
if numero < 0:
    numero_invertido = -numero_invertido

# Mostrar el resultado
print(f"La versión invertida del número es {numero_invertido}.")

Ln: 13 Col: 0
```

```
IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Invertir un numero
.py
Introduce un número entero: 1234
La versión invertida del número es 4321.
>>>
```




10. Promedio de calificaciones:

- Solicita calificaciones al usuario (hasta que ingrese -1) y calcula el promedio.
- **Ejemplo:** Entradas: 5, 7, 8, -1 → Salida: Promedio: 6.67.

```
Promedio de calificaciones.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Pro...
File Edit Format Run Options Window Help

# Inicializar variables para la suma de calificaciones y la cantidad de entradas
suma_calificaciones = 0
cantidad_calificaciones = 0

# Solicitar calificaciones al usuario hasta que ingrese -1
while True:
    calificacion = float(input("Introduce una calificación (o -1 para terminar): "))

    # Salir del bucle si la calificación es -1
    if calificacion == -1:
        break

    # Sumar la calificación y aumentar el contador de entradas
    suma_calificaciones += calificacion
    cantidad_calificaciones += 1

# Calcular y mostrar el promedio
if cantidad_calificaciones > 0:
    promedio = suma_calificaciones / cantidad_calificaciones
    print(f"Promedio: {promedio:.2f}")
else:
    print("No se ingresaron calificaciones.")

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help

Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Promedio de calificaciones.py
Introduce una calificación (o -1 para terminar): 5
Introduce una calificación (o -1 para terminar): 7
Introduce una calificación (o -1 para terminar): 8
Introduce una calificación (o -1 para terminar): -1
Promedio: 6.67
>>>
```

Funciones

1. Saludo personalizado:

- Crea una función que reciba un nombre como parámetro y retorne un saludo.
- **Ejemplo:** Entrada: María → Salida: "Hola, María!".



```
Saludo personalizado.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Saludo pe...
File Edit Format Run Options Window Help
# Definir la función que saluda al usuario
def saludar(nombre):
    return f"Hola, {nombre}!"

# Solicitar el nombre al usuario
nombre_usuario = input("Introduce tu nombre: ")

# Llamar a la función y mostrar el saludo
saludo = saludar(nombre_usuario)
print(saludo)

Ln: 1 Col: 0

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Saludo personaliza
do.py
Introduce tu nombre: Maria
Hola, Maria!
>>>

Ln: 7 Col: 0
```

2. Suma de dos números:

- Escribe una función que reciba dos números como parámetros y retorne su suma.
- **Ejemplo:** Entrada: 3, 7 → Salida: 10.

```
Suma de dos numeros.py - C:/Users/Windows/Desktop/Deberes de programacion 3/Suma de...
File Edit Format Run Options Window Help
# Definir la función que calcula la suma de dos números
def sumar(num1, num2):
    return num1 + num2

# Solicitar dos números al usuario
numero1 = float(input("Introduce el primer número: "))
numero2 = float(input("Introduce el segundo número: "))

# Llamar a la función y mostrar el resultado
resultado = sumar(numero1, numero2)
print(f"La suma es: {resultado}")

Ln: 12 Col: 0

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Suma de dos numero
s.py
Introduce el primer número: 2
Introduce el segundo número: 8
La suma es: 10.0
>>>

Ln: 8 Col: 0
```



3. Número par o impar:

- Crea una función que reciba un número y retorne True si es par y False si es impar.
- **Ejemplo:** Entrada: 4 → Salida: True.

```
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> = RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Numero par o impar.py
Introduce un número: 8
True
>>> |

File Edit Format Run Options Window Help
# Definir la función que verifica si un número es par o impar
def es_par(numero):
    if numero % 2 == 0:
        return True
    else:
        return False

# Solicitar un número al usuario
numero_usuario = int(input("Introduce un número: "))

# Llamar a la función y mostrar el resultado
resultado = es_par(numero_usuario)
print(resultado)
```

4. Calcular el cuadrado:

- Escribe una función que reciba un número y retorne su cuadrado.
- **Ejemplo:** Entrada: 5 → Salida: 25.

```
File Edit Format Run Options Window Help
# Definir la función que calcula el cuadrado de un número
def cuadrado(numero):
    return numero ** 2

# Solicitar un número al usuario
numero_usuario = float(input("Introduce un número: "))

# Llamar a la función y mostrar el resultado
resultado = cuadrado(numero_usuario)
print(f"El cuadrado de {numero_usuario} es: {resultado}")

File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> = RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Calcular el cuadrado.py
Introduce un número: 2
El cuadrado de 2.0 es: 4.0
>>> |
```



5. Calcular el área de un círculo:

- Crea una función que reciba el radio de un círculo y retorne su área.
- Fórmula: $\text{Área} = \pi * \text{radio}^2$.
- **Ejemplo:** Entrada: 3 → Salida: 28.27 (aproximado).

```
Calculador el area de un circulo.py - C:/Users/Windows/Desktop/Deberes de programacion 3/C...
File Edit Format Run Options Window Help
import math

# Definir la función que calcula el área de un círculo
def area_circulo(radio):
    return math.pi * radio ** 2

# Solicitar el radio al usuario
radio_usuario = float(input("Introduce el radio del círculo: "))

# Llamar a la función y mostrar el resultado
resultado = area_circulo(radio_usuario)
print(f"El área del círculo con radio {radio_usuario} es: {resultado:.2f}")

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
= RESTART: C:/Users/Windows/Desktop/Deberes de programacion 3/Calculador el area d
e un circulo.py
Introduce el radio del círculo: 6
El área del círculo con radio 6.0 es: 113.10
>>>
```