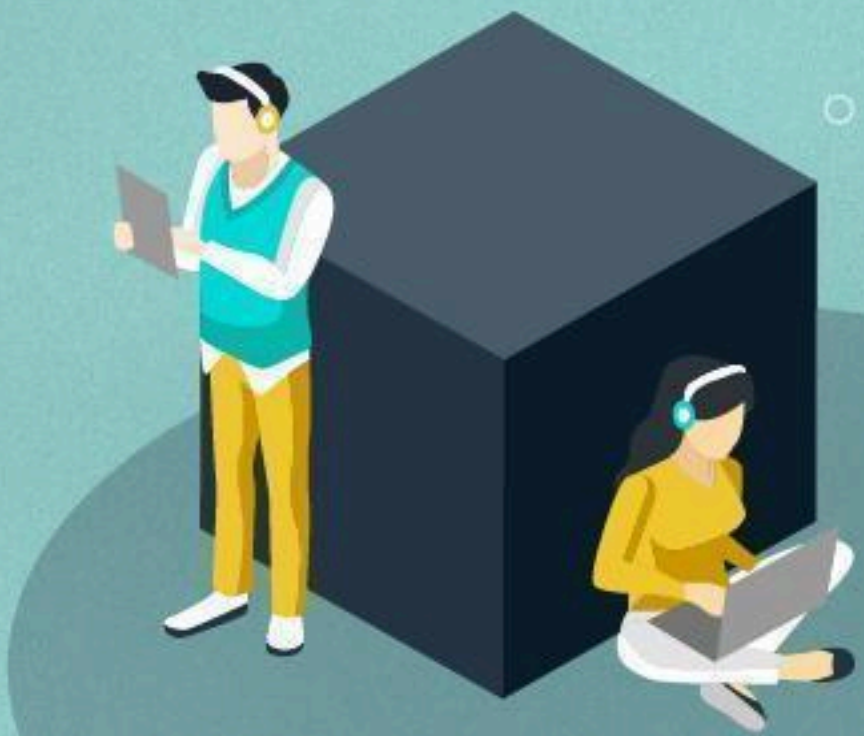
«Talento Tech»

Data Analytics con Python

Clase 04



Buenos Aires Ciudad

Buenos Aires aprende
Agencia de Habilidades para el Futuro

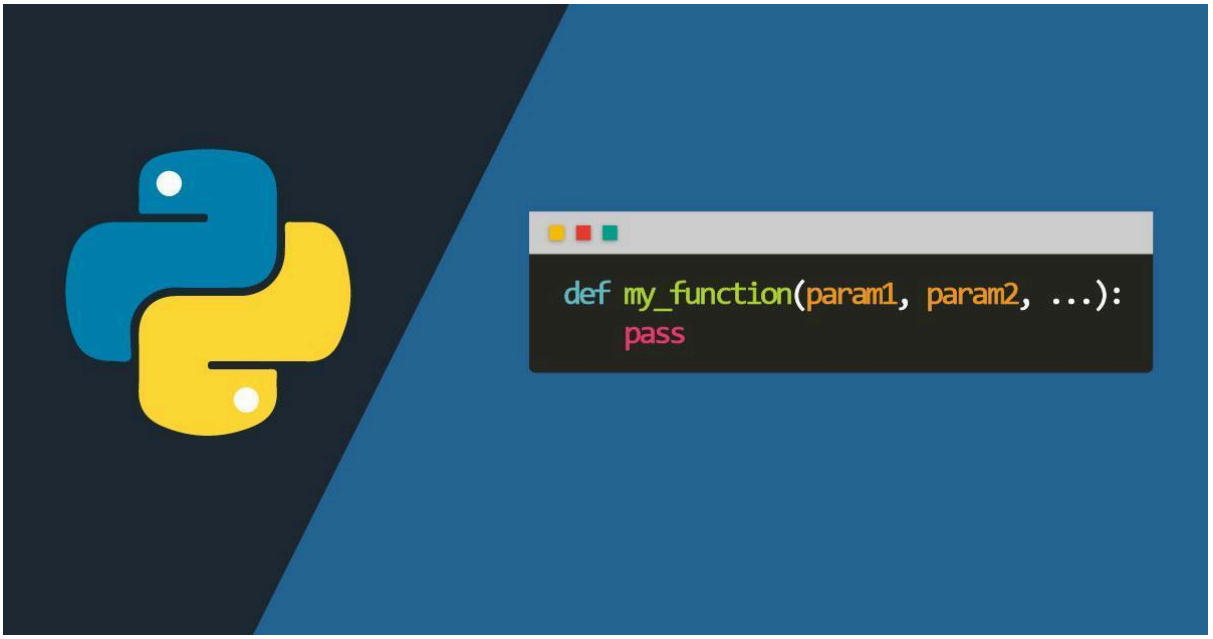
Clase N° 4 | Conceptos básicos

Temario:

- Introducción a las funciones en Python.
- Repaso

Buenos Aires aprende
Agencia de Habilidades para el Futuro

Introducción a las funciones en Python



¿Qué es una función?

Una función es un bloque de código reutilizable que realiza una tarea específica. Las funciones permiten estructurar el código de manera modular, haciendo que sea más fácil de leer, mantener y reutilizar.

¿Por qué usar funciones?

- **Reutilización del código:** Permite evitar la repetición de código.
- **Modularidad:** Facilita la lectura y el mantenimiento del código.
- **Organización:** Ayuda a organizar el código en bloques lógicos.

Sintaxis de una función

Para definir una función en Python, utilizamos la palabra clave `def`, seguida del nombre de la función, paréntesis y dos puntos. El bloque de código que pertenece a la función se indenta.

Buenos Aires aprende
Agencia de Habilidades para el Futuro

Código
##
```
def saludar():
    print("¡Hola, mundo!")
```
##

Para llamar a la función, simplemente escribimos su nombre seguido de paréntesis:

Código
##
```
saludar()
```

##

Función con parámetros

Las funciones pueden aceptar parámetros, que son variables que se pasan a la función para que las utilice.

Código
##
```
def saludar(nombre):
    print(f"¡Hola, {nombre}!")
```
##

Llamamos a la función pasando un argumento:

Código
##
```
saludar("Ana")
```

##

Función con retorno de valores

Las funciones pueden devolver valores utilizando la palabra clave `return`.

Buenos Aires aprende
Agencia de Habilidades para el Futuro

Código
##
```
def sumar(a, b):
    return a + b
```
##

Llamamos a la función y capturamos el valor devuelto:

Código
##
```
resultado = sumar(3, 5)
print(resultado)  # Imprime 8
```
##

Tipos de Funciones

- ### Funciones Integradas:

Python incluye una variedad de funciones integradas que podemos usar sin necesidad de definirlas. Ejemplos comunes incluyen **`print()`**, **`len()`**, **`input()`**, entre otras.

- ### Funciones de Usuario:

Son funciones definidas por el usuario para realizar tareas específicas dentro del programa.

Buenas prácticas al definir funciones

- **Nombres Descriptivos:** Usa nombres de funciones que describan claramente su propósito.
- **Un solo Propósito:** Cada función debe realizar una única tarea.
- **Comentarios y Docstrings:** Documenta tus funciones con comentarios y docstrings para describir lo que hacen y cómo se usan.

Ejemplo de Función Documentada:

Buenos Aires aprende
Agencia de Habilidades para el Futuro

Código
##
```
def calcular_area_rectangulo(ancho, alto):
    """
    Calcula el área de un rectángulo.

    Parámetros:
    ancho (float): El ancho del rectángulo.
    alto (float): El alto del rectángulo.

    Retorna:
    float: El área del rectángulo.
    """



    return ancho * alto
```
##

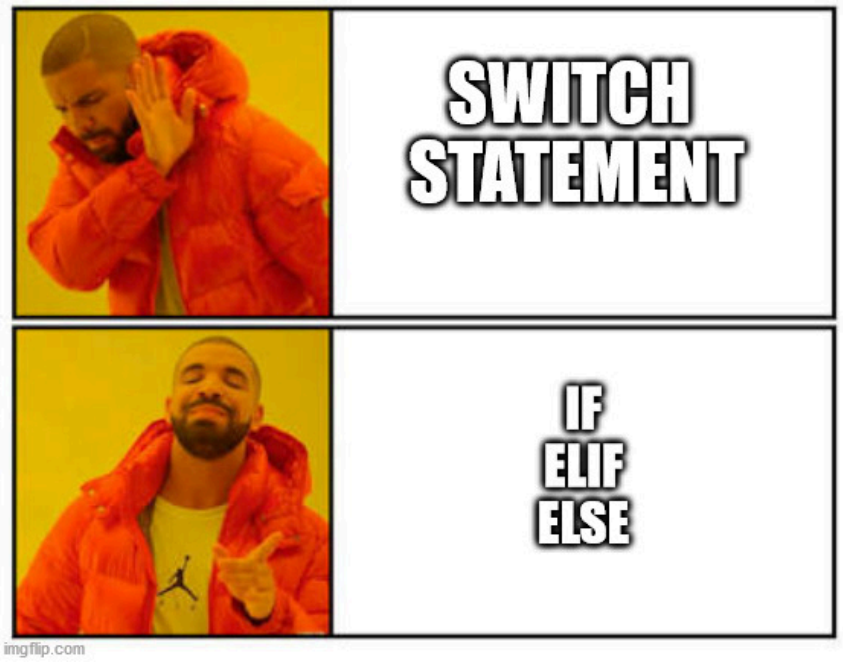
Repaso de clases anteriores

Clase 1: Fundamentos de Python

- **Tipos de Datos:** int, float, str, bool, list, tuple, dict.
- **Variables:** Espacios en memoria para almacenar datos.
- **Operadores:**
    - Aritméticos **(+, -, *, /)**
    - Comparación **(==, !=, >, <)**
    - Lógicos **(and, or, not)**
- **Entrada y Salida:**
    - **`input()`** para entrada de datos
    - **`print()`** para salida de datos.

Buenos Aires aprende
Agencia de Habilidades para el Futuro

Clase 2: Condicionales



- **Estructuras if, else, elif:** Permiten ejecutar bloques de código basados en condiciones.
- **Anidación de Condicionales:** Uso de múltiples niveles de condicionales para evaluar diferentes condiciones.

Clase 3: Bucles

- **Bucle for:** Iteración sobre una secuencia de elementos.
- **Bucle while:** Ejecución repetitiva de un bloque de código mientras una condición sea verdadera.
- **Funciones range():** Para generar una secuencia de números.

Ejemplos Prácticos

Ejemplo de Condicional y Bucle en una Función

Código
##

Buenos Aires aprende
Agencia de Habilidades para el Futuro

```python
def contar_pares(n):
    """
    Cuenta cuántos números pares hay entre 0 y n.

    Parámetros:
    n (int): El número hasta donde contar.

    Retorna:
    int: La cantidad de números pares.
    """
    contador = 0
    for i in range(n + 1):
        if i % 2 == 0:
            contador += 1
    return contador
```

```python
# Uso de la función
numero = 10
print(f"Hay {contar_pares(numero)} números pares entre 0 y {numero}.")
```

Desafío N° 4:

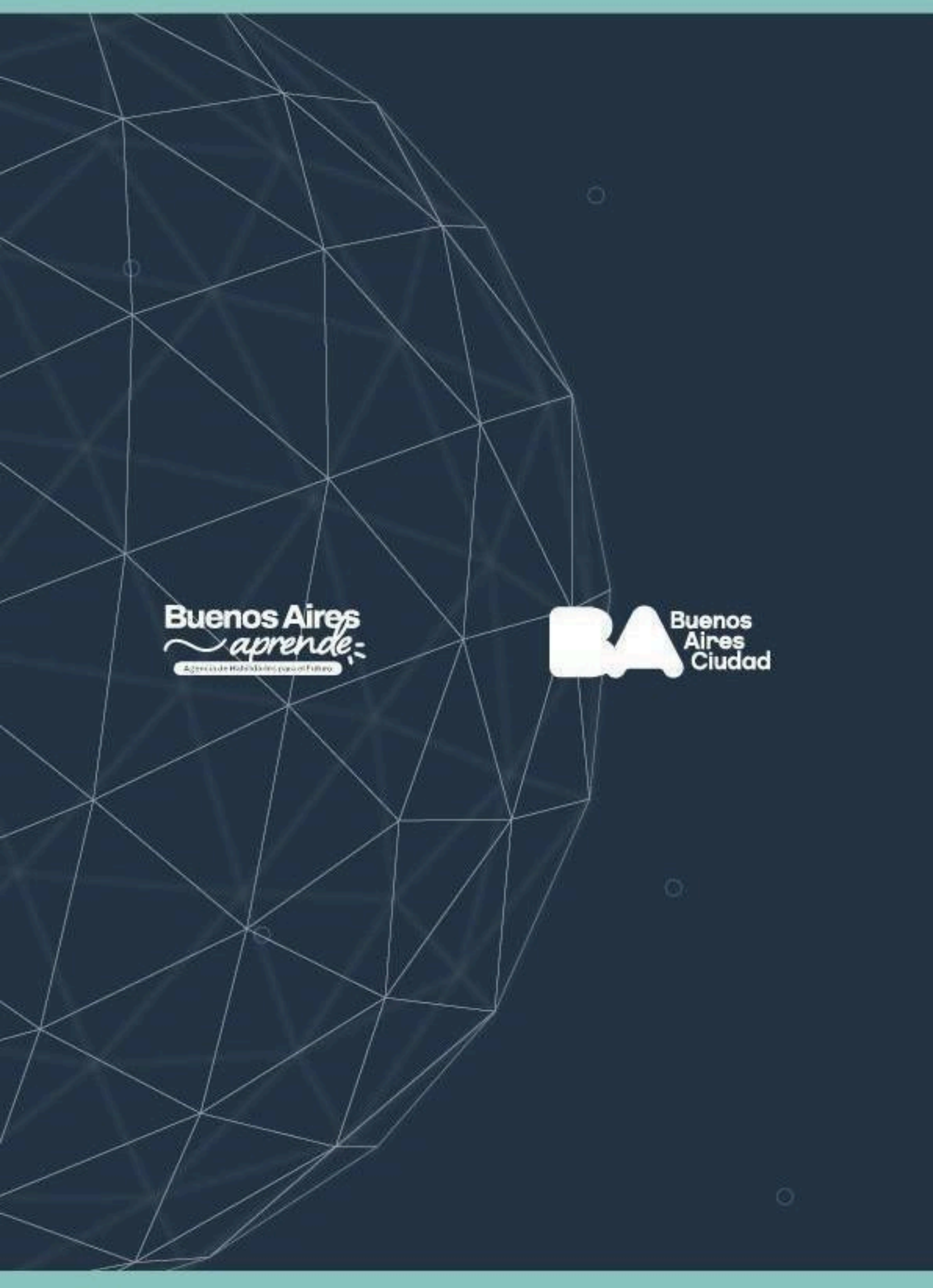
Buenos Aires aprende
Agencia de Habilidades para el Futuro

Definir una función de promedio:

a. Crea una función llamada **promedio** que reciba una lista de números llamada **numeros**.
b. La función debe retornar el promedio de los números en la lista.

```
numeros = [10, 20, 30, 40, 50]
```

Colab del desafío 4


Buenos Aires
aprende
Buenos
Aires
Ciudad