

Python Básico

Prof. Ariel Palazzesi

Comenzamos a grabar la clase

Módulo 2:

Control de flujo

Clase 05

Bucles while

1. Bucle 'while'.
2. Contadores.
3. Acumuladores.

Clase 06

Bucles while

1. Bucle for.
2. Recorriendo cadenas con for.
3. Uso de range()

Clase 07

Listas y Tuplas I

1. Creación y manipulación de listas y Tuplas.
2. Diferencias entre listas y tuplas.
3. Métodos básicos de listas: len(), append(), insert(), extend(), remove(), pop(), clear(), in
4. Slices de listas, tuplas y cadenas.

Bucle for

Bucle for

✓ El **bucle for** es una estructura de control que se utiliza para iterar sobre una secuencia de elementos, como una **lista**, **cadena**, **rango** o cualquier otro **objeto iterable**.

Su sintaxis es simple y compacta, y consta de tres partes: la palabra clave **for**, una **variable de iteración** que toma el valor de cada elemento en la secuencia en cada iteración, la palabra clave **in**, y la secuencia sobre la que se va a iterar.

El bucle for recorre cada elemento de la secuencia y ejecuta un bloque de código para cada uno de ellos.

Bucle For | Función range()



✓ **range()** se utiliza con **for** cuando sabemos la cantidad de repeticiones a realizar:

```
for i in range(inicio, fin, paso):  
    sentencia1  
    sentencia2  
primer sentencia fuera del for
```

- **i**: incrementa su valor en **paso** unidades en cada iteración.
- **inicio**: Es el valor inicial de **i**
- **fin**: El ciclo se repite mientras **i** sea menor que fin.
- **paso**: valor en que se incrementa **i** en cada iteración.

Bucle For | Función range() | Ejemplos

Código Python - Un parámetro

```
for num in range(10):  
    print(num, end=" ")
```

Terminal

```
0 1 2 3 4 5 6 7 8 9
```

Código Python - Dos parámetros

```
for num in range(3, 11):  
    print(num, end=" ")
```

Terminal

```
3 4 5 6 7 8 9 10
```

Código Python - Tres parámetros

```
for num in range(5, 16, 2):  
    print(num, end=" ")
```

Terminal

```
5 7 9 11 13 15
```

Bucle for | Contadores y acumuladores

✓ La variable de **for** asume la función del **contador**.

Este programa de ejemplo permite ingresar 5 valores por teclado y obtener su suma y su promedio:

Código Python

```
suma = 0
for cont in range(5):
    num= int(input("Ingrese un número: "))
    suma = suma + num

print("La suma es:",suma)
print("El promedio es:",suma/(cont+1))
```

Terminal

```
Ingrese un número: 2
Ingrese un número: 6
Ingrese un número: 9
Ingrese un número: 15
Ingrese un número: 3
La suma es: 35
El promedio es: 7.0
```


Bucle For | Iteración sobre cadenas



✓ La **iteración sobre cadenas** se refiere al proceso de recorrer cada carácter individual en una cadena de texto utilizando un bucle.

Código Python

```
cadena = "Python"

for letra in cadena:
    print(letra)
```

Terminal

```
P
y
t
h
o
n
```

Listas (¡spoiler!)

Listas (Adelanto)



✓ Una **lista** es una secuencia ordenada de elementos. Cada elemento es un dato. Pueden ser del mismo o distinto tipo, aunque esto último es poco frecuente.

Las listas en Python son un tipo **contenedor compuesto**, se usan para almacenar conjuntos de elementos relacionados, y las analizaremos en profundidad en el video siguiente.

Listas (Adelanto)

✓ Para crear una lista en Python, se usan corchetes `[]` y se separan los elementos con comas. Una lista puede contener cualquier tipo de dato, como números, cadenas o incluso otras listas.

Código Python

```
# Ejemplo de lista de números
numeros = [10, 20, 30, 40, 50]

# Ejemplo de lista de colores
colores = ["Rojo", "Verde", "Azul"]
```



Listas (Adelanto)

✓ Podés acceder a los elementos de una lista utilizando su índice. El primer elemento tiene el índice 0, el segundo el índice 1, y así sucesivamente.



Código Python

```
# Lista de ejemplo
frutas = ["manzana", "banana", "cereza"]

# Accedemos a los elementos por índice
print(frutas[0]) # Salida: manzana
print(frutas[1]) # Salida: banana
print(frutas[2]) # Salida: cereza
```


Listas y bucles | for



✓ Esta es la sintáxis básica del bucle for para recorrer una lista.

Código Python

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

for numero in numeros:
    print(numero)

print("Fin.")
```

Terminal

```
10
20
30
40
50
Fin.
```

Listas y bucles | while

Código Python

```
numeros = [10, 20, 30, 40, 50]
indice = 0

while indice < len(numeros):
    print("Valor en", indice, end="")
    print(":", numeros[indice])
    indice += 1

print("Fin del bucle.")
```

Terminal

```
Valor en 0: 10
Valor en 1: 20
Valor en 2: 30
Valor en 3: 40
Valor en 4: 50
Fin del bucle.
```

Listas y bucles | for

Código Python

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

for indice in range(len(numeros)):
    print("Valor en", indice, end="")
    print(":", numeros[indice])

print("Fin del bucle.")
```

Terminal

```
Valor en 0: 10
Valor en 1: 20
Valor en 2: 30
Valor en 3: 40
Valor en 4: 50
Fin del bucle.
```

Desafíos

Desafíos de la clase:

✓ Desafío 1: Contador de vocales en una cadena

Escribe un programa en Python que solicite al usuario una cadena de texto y cuente cuántas vocales contiene (tanto mayúsculas como minúsculas).

Utiliza un bucle **for** para recorrer la cadena y determinar cuántas vocales hay.



Desafíos de la clase (Resolución):



Desafío 1: Contador de vocales en una cadena

Código Python

```
cadena = input("Ingrese una cadena de texto: ")
contador_vocales = 0 # Inicializar un contador de vocales

# Recorrer cada carácter de la cadena
for letra in cadena:
    if letra.lower() in "aeiou":
        contador_vocales += 1

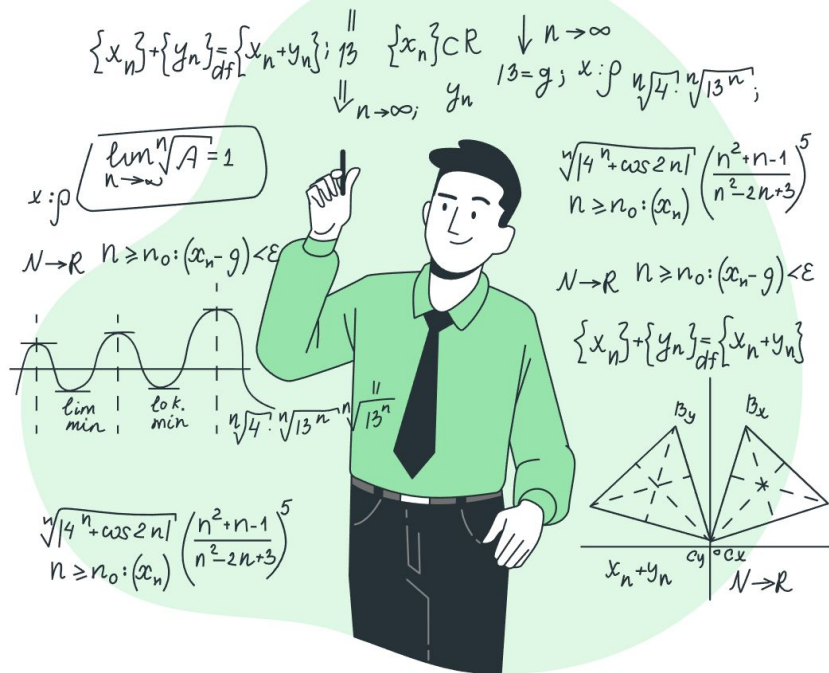
# Mostrar el resultado
print("La cantidad de vocales es:", contador_vocales)
```

Desafíos de la clase:

✓ Desafío 2: Suma de los primeros N números

Crea un programa que pida al usuario un número entero N y calcule la suma de todos los números enteros desde 1 hasta N.

Utiliza un bucle **for** y la función **range()** para realizar la suma.



Desafíos de la clase (Resolución):



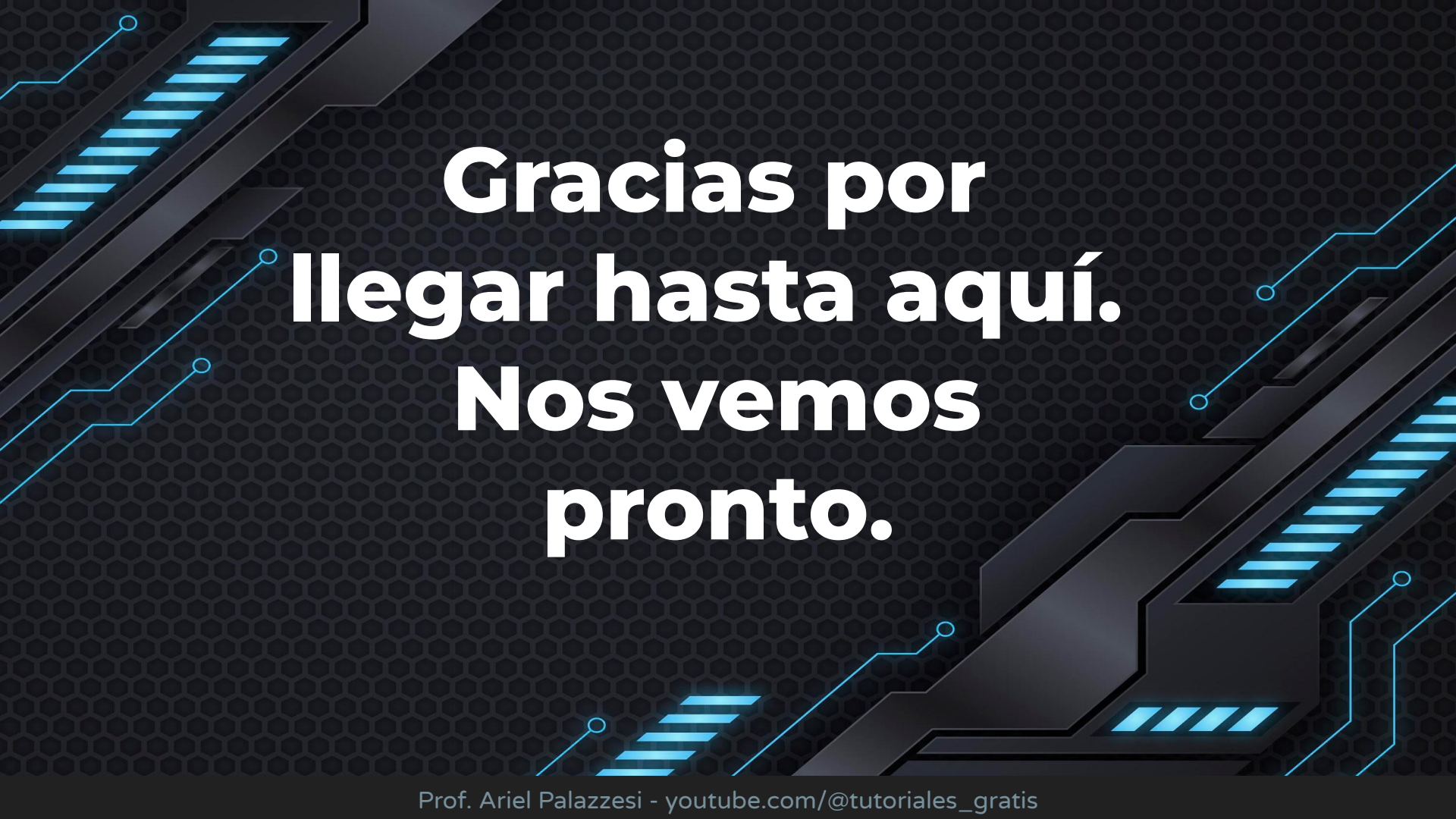
Desafío 2: Suma de los primeros N números

Código Python

```
N = int(input("Ingrese un número entero: "))
suma = 0 # Inicializar la variable para la suma

# Sumar todos los números desde 1 hasta N
for num in range(1, N + 1):
    suma += num

# Mostrar el resultado
print("La suma de los números del 1 al", N, "es:", suma)
```



**Gracias por
llegar hasta aquí.
Nos vemos
pronto.**