



# Iniciación con Python

Clase 04 - “Ruta de avance”

# ¡Les damos la bienvenida!



Vamos a comenzar a grabar la  
clase

## Clase **03.**

### ► Datos en Python

1. Conversión entre tipos de datos.
2. Operadores algebraicos.
3. Expresiones.
4. Uso de input()
5. Programas con entrada, procesamiento y salida de datos.

## Clase **04.**

### ► Ruta de avance

1. Definimos los requisitos del Proyecto Integrador.
2. Menú de opciones.
3. Pedir, procesar y mostrar datos.

## Clase **05.**

### ► Condicionales

1. Cadena de caracteres.
2. Operadores lógicos.
3. Control de flujo: estructuras condicionales (if, else, elif)

# Pero antes...

---



¡Resolvamos los “**Ejercicios Prácticos**”  
de la clase anterior!



# Ruta de avance



# PFI

El **Proyecto Final Integrador (PFI)** consiste en el desarrollo de una aplicación Python, que utilizando la terminal o consola permita al usuario gestionar el inventario de una pequeña tienda o comercio.

La aplicación debe ser capaz de registrar, actualizar, eliminar y mostrar productos en el inventario. Además, debe incluir funcionalidades para realizar búsquedas y generar reportes de productos con bajo stock.



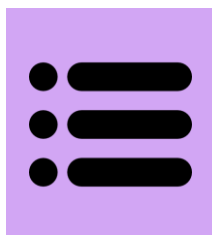


# Proyecto Final Integrador

---

El proyecto incluirá los medios necesarios para interactuar con una base de datos, implementará un menú con las opciones disponibles, y mecanismos (funciones) para el mantenimiento de los datos de los productos, incluyendo:

- **Registro:** Alta de productos nuevos.
- **Visualización:** Consulta de datos de productos.
- **Actualización:** Modificar la cantidad en stock de un producto.
- **Eliminación:** Dar de baja productos.
- **Listado:** Listado completo de los productos en la base de datos.
- **Reporte de Bajo Stock:** Lista de productos con cantidad bajo mínimo.



# Menú de opciones





# Menú de opciones

Uno de los sistemas más populares utilizados para permitir al usuario seleccionar qué función del programa necesita utilizar en un momento dado es el llamado “Menú de opciones”.

Cómo se vé en la imagen, consiste en una lista de las opciones disponibles y un mecanismo para pedir al usuario que selecciona una.

Comencemos a trabajar en el PFI creando una versión preliminar de su menú de opciones.

```
Menu de Opciones
```

```
1. Opcion 1
```

```
2. Opcion 2
```

```
3. Opcion 3
```

```
4. Opcion 4
```

```
0. SALIR
```

```
Ingrese una opcion:
```



# Menú de opciones

```
# Menú de opciones
print("Menú de Gestión de Productos\n")
print("1. Alta de productos nuevos")
print("2. Consulta de datos de productos")
print("3. Modificar la cantidad en stock de un producto")
print("4. Dar de baja productos")
print("5. Listado completo de los productos")
print("6. Lista de productos con cantidad bajo mínimo")
print("7. Salir")

# Solicitar al usuario que seleccione una opción
opcion = int(input("Por favor, selecciona una opción (1-7): "))

# Mostramos el nro de la opción seleccionada
print("Has seleccionado:", opcion)
```

A la izquierda, una posible versión del menú de opciones para nuestro proyecto.

Verás que hay tres líneas que comienzan con "#". Esas líneas son ignoradas por Python, y constituyen comentarios que el desarrollador puede incluir en su código para facilitar su lectura.



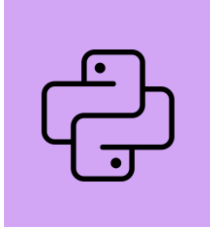
# Menú de opciones

El código de la diapositiva anterior utilizar varias líneas `print()` para generar el texto con las opciones disponibles, e `input()` para leer la opción elegida por el usuario. Recordemos que `input()` regresa un texto, así que usamos `int()` para convertir lo ingresado en un número entero.

```
Menú de Gestión de Productos
```

```
1. Registro: Alta de productos nuevos
2. Visualización: Consulta de datos de productos
3. Actualización: Modificar la cantidad en stock de un producto
4. Eliminación: Dar de baja productos
5. Listado: Listado completo de los productos en la base de datos
6. Reporte de Bajo Stock: Lista de productos con cantidad bajo mínimo
7. Salir
Por favor, selecciona una opción (1-7): 2
Has seleccionado: 2
```

La próxima clase aprenderemos a utilizar estructuras de control para poder ejecutar un bloque de código u otro en función de la opción elegida.



# Proceso de datos





# Precio total con IVA

Este programa calcula el precio total de un producto incluyendo el IVA (Impuesto al Valor Agregado).

```
# Entrada: Solicitar el precio del producto y el porcentaje de IVA
precio = float(input("Precio del producto: "))
iva = float(input("Porcentaje de IVA (por ejemplo, 21 para 21%): "))

# Proceso: Calcular el monto del IVA y el precio total
monto_iva = (precio * iva) / 100
precio_total = precio + monto_iva

# Salida: Mostrar el precio total incluyendo IVA
print("El precio total con IVA es:", precio_total)
```



# Explicación del código

---

Este programa toma el precio de un producto y un porcentaje de IVA como entrada, calcula el monto del IVA y lo suma al precio original del producto, y luego imprime el precio total que el usuario deberá pagar, incluyendo el IVA.

## Entrada de datos:

- El programa primero solicita al usuario que ingrese el precio del producto utilizando la función `input()`. Este valor se convierte a un número decimal (tipo `float`) y se guarda en la variable `precio`.
- Luego, solicita el porcentaje de IVA que se desea aplicar al producto, también utilizando `input()`. Este valor se convierte a un número decimal (tipo `float`) y se guarda en la variable `iva`.



# Explicación del código

## Cálculo del IVA y del precio total

- El programa calcula el monto del IVA multiplicando el precio del producto (precio) por el porcentaje de IVA (iva), y luego divide ese resultado por 100 para obtener el valor del IVA en unidades monetarias. Este valor se almacena en la variable monto\_iva.
- A continuación, el programa suma el precio del producto (precio) y el monto del IVA (monto\_iva) para obtener el precio total del producto, incluyendo el IVA. Este valor se almacena en la variable precio\_total.





# Explicación del código

## Salida de datos

- Finalmente, el programa muestra el precio total del producto con IVA incluido utilizando la función `print()`.

```
Precio del producto: 100  
Porcentaje de IVA (por ejemplo, 21 para 21%): 21  
El precio total con IVA es: 121.0
```

Ejemplo de ejecución del programa.



# Cálculo del descuento

Este programa calcula el precio final de un producto después de aplicar un descuento:

```
# Entrada: Solicitar el precio del producto y el porcentaje de descuento
precio = float(input("Precio del producto: "))
descuento = float(input("Porcentaje de descuento (por ejemplo, 10 para 10%): "))

# Proceso: Calcular el monto del descuento y el precio final
monto_descuento = (precio * descuento) / 100
precio_final = precio - monto_descuento

# Salida: Mostrar el precio final después del descuento
print("El precio final después del descuento es:", precio_final)
```



# Explicación del código

---

Este programa pide el precio de un producto y un porcentaje de descuento (entrada), calcula el monto del descuento, lo resta del precio original del producto y luego imprime el precio final que el usuario deberá pagar después de aplicar el descuento.

## Entrada de datos:

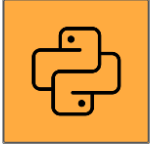
- El programa solicita al usuario que ingrese el precio del producto utilizando la función `input()`. Este valor se convierte a un número decimal (tipo `float`) y se guarda en la variable `precio`.
- Luego, solicita el porcentaje de descuento que se desea aplicar al producto, también utilizando `input()`. Este valor se convierte a un número decimal (tipo `float`) y se guarda en la variable `descuento`.



# Explicación del código

## Cálculo del descuento y del precio final:

- El programa calcula el monto del descuento multiplicando el precio del producto (precio) por el porcentaje de descuento (descuento), y luego divide ese resultado por 100 para obtener el valor del descuento en unidades monetarias. Este valor se almacena en la variable monto\_descuento.
- A continuación, el programa resta el monto del descuento (monto\_descuento) del precio original del producto (precio) para obtener el precio final del producto después de aplicar el descuento. Este valor se almacena en la variable precio\_final.



# Explicación del código

## Salida de datos:

- Para finalizar, el programa muestra el precio final del producto con el descuento aplicado utilizando la función print().

```
Precio del producto: 1200  
Porcentaje de descuento (por ejemplo, 10 para 10%): 5  
El precio final después del descuento es: 1140.0
```

Ejemplo de ejecución del programa.



# Compra con varios artículos

Este programa calcula el costo total de varios artículos comprados, basándose en el precio unitario y la cantidad comprada:

```
# Entrada: Solicitar el precio unitario y la cantidad de artículos
precio_unitario = float(input("Ingresa el precio unitario del artículo: "))
cantidad = int(input("Ingresa la cantidad de artículos comprados: "))

# Proceso: Calcular el costo total
costo_total = precio_unitario * cantidad

# Salida: Mostrar el costo total de la compra
print("El costo total de la compra es:", costo_total)
```



# Explicación del código

---

Se toma como entrada el precio unitario de un artículo y la cantidad de artículos que se van a comprar. Luego, se calcula el costo total multiplicando el precio unitario por la cantidad de artículos y se muestra el resultado al usuario, que es el monto total a pagar por la compra.

## Entrada de datos:

- El programa solicita al usuario que ingrese el precio unitario del artículo utilizando la función `input()`. Este valor se convierte a un número decimal (tipo `float`) y se guarda en la variable `precio_unitario`.
- Luego, solicita la cantidad de artículos que se van a comprar, utilizando nuevamente `input()`. Este valor se convierte a un número entero (tipo `int`) y se guarda en la variable `cantidad`.



# Explicación del código

## Cálculo del costo total:

- El programa calcula el costo total de la compra multiplicando el precio unitario del artículo (`precio_unitario`) por la cantidad de artículos comprados (`cantidad`). Este resultado se almacena en la variable `costo_total`.

## Salida de datos:

- Finalmente, el programa muestra el precio total del producto con IVA incluido utilizando la función `print()`.

```
Ingresa el precio unitario del artículo: 125
Ingresa la cantidad de artículos comprados: 5
El costo total de la compra es: 625.0
```

Ejemplo de ejecución del programa.



# ¡Vamos a la práctica!



# Ejercicios Prácticos



Optativos | No entregables

## Ticket de la compra

1. Escribir un programa que solicite el nombre, la cantidad y el valor unitario de tres productos.
2. Luego, debe calcular el importe de IVA (21%) de cada producto.
3. Por último, debe mostrar en la terminal el ticket de la operación con todos los datos de la compra.

## Consumo de combustible

Realizar una aplicación en Python que;  
A partir de la cantidad de litros de combustible que consume un coche por cada 100 km de recorrido, el costo de cada litro de combustible y la longitud del viaje realizado (en kilómetros), muestra un detalle de los litros consumidos y el dinero gastado.



# Ruta de aprendizaje

Clase 04 - “Ruta de avance”



# Ruta de Avance



Optativos | No entregables

## Guía estructurada

La **Ruta de Aprendizaje** te proporcionará directrices claras para avanzar en tu proyecto integrador final, ayudándote a mantenerte enfocado y alineado con los objetivos del curso.

## Mantené el Ritmo:

La **Ruta de Aprendizaje** te permitirá monitorear tu progreso de manera constante, asegurándote de que mantengas el ritmo adecuado y cumplas con los hitos establecidos para finalizar tu proyecto con éxito.



**Frecuencia:** Cada 8 clases (4, 12)

## Establecimiento de Objetivos a Corto Plazo

Cada 8 clases, recibirás sugerencias sobre los avances que deberías hacer en tu proyecto, en línea con los temas y habilidades que hemos desarrollado en las clases anteriores. Esto te permitirá integrar de manera efectiva los conocimientos adquiridos e ir avanzando poco a poco en la construcción de la “**Pre-Entrega**” y la “**Entrega Final del Proyecto**”



# ¡NUEVO CUESTIONARIO EN CAMPUS!

La resolución del cuestionario es de carácter obligatorio para desbloquear los contenidos de las próximas 2 clases