

DIPLOMATURA SUPERIOR EN INTELIGENCIA ARTIFICIAL

Módulo 1: Programación orientada a IA

Ejercicios para practicar en Python:

Clase 3: Programación Modular – Programación Orientada a Objetos

- 1) Defina una función que reciba una palabra y retorne la cantidad de vocales que contiene.
- 2) Describa con sus palabras lo que realiza la siguiente función:

```
def ne(num1, num2):  
    if (num1 > num2):  
        aux = num1  
        num1 = num2  
        num2 = aux  
  
    for i in range(num1+1, num2):  
        print(i)
```

- 3) Implemente una función que reciba como parámetro un número entero y retorne la cantidad de dígitos que posee y la suma de los mismos.
- 4) Escriba un programa que genere aleatoriamente 100 números reales, los almacene en una lista, y luego los muestre ordenados de menor a mayor. Obligatorio: utilizar funciones.
- 5) (a) Cree una clase llamada Estudiante que reciba nombre, apellido y nota final al crear el objeto. La clase debe tener métodos para:
 - Mostrar el nombre completo concatenando nombre y apellido.
 - Indicar si el estudiante aprobó o desaprobó (nota < 6).(b) Desarrolle un programa que cree varios objetos Estudiante y muestre por pantalla su nombre completo y estado (aprobado/desaprobado).
- 6) (a) Cree una clase llamada CuentaBancaria que reciba titular y saldo inicial al crear el objeto. La clase debe tener métodos para:
 - Depositar un monto.
 - Retirar un monto.
 - Mostrar el saldo actual.

(b) Desarrolle un programa que cree una cuenta, realice operaciones de depósito y retiro, y muestre los resultados después de cada operación.

7) (a) Cree una clase llamada EquipoFutbol que tenga los siguientes atributos: nombre del equipo, ciudad a la que pertenece el equipo y cantidad de puntos obtenidos en el campeonato. La clase debe tener los siguientes métodos:

- `Mostrar_Info` → que muestre la información del equipo. La descripción debe incluir el nombre del equipo, su ciudad y los puntos acumulados.
- `Sumar_Puntos` → que reciba el resultado de un partido ("gana", "empata", "pierde") y actualice los puntos según las reglas:
 - **Gana:** suma 3 puntos.
 - **Empata:** suma 1 punto.
 - **Pierde:** suma 0 puntos.

(b) Desarrolle un programa que cree al menos dos objetos de la clase y simule el resultado de un partido, mostrando la información de los equipos **antes y después** de actualizar los puntos.

8) (a) Cree una clase llamada Producto que reciba tres atributos: nombre, precio, cantidadDisponible. La clase debe tener los siguientes métodos:

- `MostrarDescripción` → que muestre la descripción del producto. La descripción debe incluir el nombre, el precio y la cantidad disponible.
- `Vender` → que disminuye la cantidad disponible. Si la cantidad a vender excede el stock disponible, en el Main debe mostrar un mensaje de error.
- `Reponer` → que aumenta la cantidad disponible.
- `CalcularValorTotal` → que calcule y retorne el valor total del stock disponible (precio x cantidadDisponible).

(b) Desarrolle un programa que cree al menos dos productos diferentes y venda y reponga productos asegurándose de que el stock se actualice correctamente.