

## M2: FUNDAMENTOS DE PROGRAMACIÓN PYTHON PARA INGENIEROS DE DATOS

### Glosario:

1. **Sintaxis:** Conjunto de reglas que define la estructura del código en Python, asegurando que sea legible y ejecutable.
2. **Variable:** Espacio en memoria que almacena datos y puede cambiar de valor durante la ejecución de un programa.
3. **Tipo de Dato:** Categoría de valores que pueden ser manipulados en Python, como `int`, `float`, `str`, y `bool`.
4. **Operador:** Símbolo o palabra que indica una operación, como suma (+), resta (-), comparación (==), entre otros.
5. **Condicional (`if`):** Estructura de control que permite ejecutar un bloque de código si se cumple una condición específica.
6. **Bucle (`for`, `while`):** Sentencia que permite repetir un bloque de código mientras se cumple una condición o se recorren elementos en una colección.
7. **Lista:** Estructura de datos en Python que almacena una colección ordenada de elementos que pueden ser de diferentes tipos.
8. **Diccionario:** Estructura de datos que almacena pares de clave-valor, permitiendo un acceso rápido a los valores mediante una clave única.
9. **Conjunto (Set):** Colección desordenada de elementos únicos, útil para operaciones de pertenencia y deduplicación.
10. **Función:** Bloque de código reutilizable que realiza una tarea específica y puede retornar un valor.

11. **Módulo:** Archivo en Python que contiene funciones y variables relacionadas, permitiendo la organización modular del código.
12. **Paquete:** Conjunto de módulos organizados en una estructura de directorios, facilitando la gestión de proyectos grandes.
13. **Excepción:** Error o situación inesperada durante la ejecución del programa, que puede ser gestionada con el manejo de excepciones (`try-except`).
14. **Polimorfismo:** Característica de la programación orientada a objetos que permite utilizar una misma interfaz para distintos tipos de datos.
15. **Encapsulamiento:** Principio de POO que limita el acceso directo a los datos de una clase, protegiéndolos mediante métodos.
16. **Herencia:** Mecanismo que permite que una clase derive de otra y herede sus atributos y métodos.
17. **Lambda:** Función anónima en Python que se define en una línea y es útil para operaciones rápidas.
18. **Argumento:** Valor pasado a una función cuando se llama, que influye en la ejecución de la función.
19. **Return:** Palabra clave que indica el valor que una función envía de vuelta al llamador.
20. **Biblioteca Estándar:** Conjunto de módulos preinstalados en Python, que cubre una amplia gama de tareas comunes.