



# I. Entorno de Desarrollo

PROGRAMACIÓN ORIENTADA A OBJETOS

# 1.1 Estructura de un proyecto

Un **proyecto** en POO con Python es una *colección organizada de archivos y carpetas*

Contiene **código fuente**, **recursos** y **documentación** para desarrollar una aplicación

# Carpetas y archivos principales

## Estructura básica

- **Carpeta raíz:** Contiene todos los archivos del proyecto.
- **Archivo Principal:** main.py (inicia la aplicación)
- **Módulos:** Archivos .py con clases y funciones.
- **Carpeta `__pycache__`:** Almacena archivos compilados (.pyc)

# Estructura Recomendada

## ❖ **Organización del proyecto**

- **src o app:** Código fuente
- **Tests:** Pruebas unitarias
- **Docs:** Documentación (README.md)

# Ejemplo:

mi\_proyecto/

├── src/

| ├── \_\_init\_\_.py

| ├── main.py

| ├── modulo1.py

| └── modulo2.py

├── tests/

| ├── test\_modulo1.py

| └── test\_modulo2.py

├── docs/

| └── README.md

└── \_\_pycache\_\_/

# 1.2 Proceso de desarrollo de aplicaciones

•Ciclo de vida de desarrollo de software (SDLC):

1. Descripción del problema,
2. Definición de la solución
3. Diseño,
4. Desarrollo e implementación,
5. Depuración pruebas y
6. Documentación y mantenimiento.

# Metodologías Ágiles para el desarrollo de proyectos

**Metodologías ágiles:** Scrum, XP o Kanban para iteraciones rápidas

**Editar, Compilar, Probar, Depurar:** Se repite continuamente.

# 1.2.1 Compilación

Durante el proceso de compilación Python convierte archivos .py a .pyc

Los archivos .pyc generados automáticamente se guardan en `__pycache__`

Algunos IDEs permiten una configuración adicional al ejecutar o compilar, así como una visualización simple de errores comunes de sintaxis y semántica

# 1.2.2 Archivos que crea el Entorno de desarrollo

Dependiendo del IDE seleccionado es posible que se creen ciertos archivos, por ejemplo Visual Studio Code genera archivos y carpetas automáticamente para organizar el proyecto.

- Archivos **.py** que representan el código fuente
- Archivos **.pyc** Bytecode
- Archivos **.vs** Configuración del IDE
- **\_\_pycache\_\_** carpetas auxiliares

# 1.2.3 Depuración

Se le llama Depuración al proceso de **identificación** y **corrección de errores** y se considera esencial para asegurar el correcto funcionamiento del programa.

## Conceptos básicos en el proceso de depuración

- **Puntos de interrupción (breakpoints)**: Son marcas en las líneas de código para **Pausar** la ejecución.
- **Inspección de variables**: Visualizar y dar seguimiento el estado de las variables.
- **"Watch"** y **"Call Stack"**: Herramientas para rastrear el flujo del programa.