

### Conceptos:

- Prueba Unitaria : Forma de comprobar si una porción de código funciona bien.
  - automatizables
  - Completas
  - Repetibles
  - Independientes

PyTest: Librería para crear pruebas de código:

- Ejecuta test en archivos **test\_\*.py** o **\_test.py**
- Ejecuta funciones test
- Devuelve cómo resultado "." o "F" dependiendo si pasan o no el test
- Se ejecuta en una carpeta y analiza todos los archivos "test"
- Se puede ejecutar directamente con el nombre del archivo
- Fixtures: `@pytest.fixture`
- Parametrización :Permite comprobar si los outputs están saliendo con los valores que se espera.

Desarrollar un test, no indica que no existan errores, pero aumenta la confianza en el código

### Test-Driven Development (TDD):

- Pasos:
  1. Escribir una prueba que falla de una funcionalidad del software
  2. Escribir el código mínimo de la función para que la prueba pase
  3. Refactorizar el código
- Red-Green-Refactor:
  - a. Rojo: Se escribe prueba automatizada del código → como debería funcionar
  - b. Verde:Se escribe el código hasta que la prueba pase
  - c. Refactor : Se refactoriza el código
- Beneficios:
  - Robustez
  - Evolutividad:
  - Mantenibilidad
  - Documentación:
- Cobertura de código
  - Métrica
  - Mide cantidad de código fuente cubierto por pruebas.
  - `pytest --cov=.`: Entrega:
    - Stmts:
    - Miss:
    - Cover :