

Gestor de Biblioteca amb Tests Unitaris

Exercici python 5

Entorns de desenvolupament

Jaume Albert Salas Pastor

06/01/2025

Índex

1. Introducció.....	3
2. Estructura del codi font.....	4
3. Descripció de les funcionalitats.....	5
4. Implementació dels tests.....	6
5. Captures de pantalla amb els resultats.....	7
6. Conclusions.....	7

1. Introducció

Objectiu

L'objectiu d'aquest exercici és implementar un programa en Python que gestioni una **biblioteca de llibres** i utilitzar la llibreria **unittest** per **realitzar tests** unitaris que validin les diferents funcionalitats desenvolupades. Això permet aprendre i aplicar **bones pràctiques de desenvolupament**, com ara provar el codi per garantir-ne la **qualitat i el correcte funcionament**.

Context

En el desenvolupament de programari, els tests unitaris són una eina fonamental per assegurar que cada component del codi funciona com s'espera. Aquest exercici és clau per entendre el procés de creació i execució de tests, i la seva importància en el cicle de vida del programari.

2.Estructura del codi font

El projecte està organitzat en dos fitxers principals:

1. **biblioteca.py**: Conté la classe Biblioteca, que implementa les funcionalitats per gestionar els llibres.
2. **test_biblioteca.py**: Conté els tests unitaris per verificar el correcte funcionament de les funcionalitats de la classe Biblioteca.

Estructura dels arxius

- **biblioteca.py**
 - Classe Biblioteca
 - Mètodes:
 - afegirllibre(titol, autor, any_publicacio)
 - cercarllibre(titol)
 - eliminarllibre(titol)
 - llistarllibres()
- **test_biblioteca.py**
 - Classe TestBiblioteca
 - Mètodes de test:
 - test_afegirllibre
 - test_cercarllibre_existent
 - test_eliminarllibre_existent
 - test_llistarllibres
 - test_eliminar_totsllibres

3.Descripció de les funcionalitats

La classe Biblioteca implementa les següent funcions:

1. Afegir llibre:

- Permet afegir un llibre a la biblioteca indicant el títol, l'autor i l'any de publicació.
- El llibre s'afegeix com **un diccionari a una llista** interna.

2. Cercar llibre pel títol:

- Busca un llibre a la biblioteca pel títol.
- Retorna el llibre si existeix o None si no es troba.

3. Eliminar llibre pel títol:

- Elimina un llibre de la biblioteca pel seu títol.

4. Llistar tots els llibres:

- Retorna una llista amb tots els llibres disponibles a la biblioteca.

4. Implementació dels tests

Els tests unitaris han estat implementats al fitxer `test_biblioteca.py`, utilitzant la llibreria `unittest`.

Mètodes de test:

1. `test_afegir_llibre`:

- Verifica que es poden afegir llibres correctament.
- Comprova que el llibre s'ha afegit amb els camps indicats.

2. `test_cercar_llibre_existent`:

- Prova la cerca d'un llibre existent.
- Comprova que es retorna la informació correcta.

3. `test_eliminar_llibre_existent`:

- Prova l'eliminació d'un llibre existent.
- Comprova que el llibre s'ha eliminat correctament.

4. `test_llistar_llibres`:

- Verifica que es poden llistar tots els llibres de la biblioteca.
- Comprova que el nombre de llibres retornats coincideix.

5. `test_eliminar_tots_llibres`:

- Prova l'eliminació de tots els llibres de la biblioteca.
- Comprova que la llista de llibres queda buida al final.

5. Captures de pantalla amb els resultats



```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS  COMENTARIOS

PS E:\FP Informatica\1º DAM\py> & C:/Users/jaume/AppData/Local/Programs/Python/Python312/
python.exe "e:/FP Informatica/1º DAM/py/biblioteca+test/test_biblioteca.py"
.....
-----
Ran 5 tests in 0.001s

OK
PS E:\FP Informatica\1º DAM\py> |
```

6. Conclusions

En aquest exercici, he après:

- Com estructurar i implementar una classe en Python per gestionar dades (en aquest cas, una biblioteca de llibres).
- La importància de realitzar tests unitaris per assegurar que el codi funciona com s'espera.
- Com utilitzar la llibreria unittest per crear i executar tests unitaris.

Valoració personal

Realitzar aquest exercici m'ha ajudat a comprendre millor les bones pràctiques de desenvolupament de programari. A més, treballar amb exemples de tests m'ha fet veure com identificar possibles errors i comprovar que el codi sigui robust i fiable.