

Ejecucion de test con pytest

algunas formas comunes de ejecutar pruebas con Pytest son

Basica

para ejecutar todos los test que tenemos en nuestro proyecto, podemos ejecutarlos simplemente con:

```
pytest
```

este ejecutara todos los archivos que comiencen con `test_`

Ejecutar un archivo

Hay ocasiones que necesitamos ejecutar una test suite y para esto podemos hacerlo de la siguiente manera

```
pytest archivo_de_pruebas.py
```

o tambien

```
python -m pytest archivo_de_pruebas.py
```

Ejecutar un test en especifico

Tambien podemos encontrarnos frente a la necesidad de probar solo un test de manera individual, por lo que pytest nos provee

```
pytest -k nombre_de_funcion_de_prueba
```

Algunas opciones adicionales

hay ocasiones en la que no queremos que se ejecute un solo test, y tenemos varias opciones para hacer esto

1. desde la linea de comandos podemos excluir colocando `not` y luego el nombre del test

```
pytest -k "not test_buscar_en_ingles"
```

2. pytest nos ofrece un marcador llamado `@pytest.mark.skip(reason="en desarrollo")` y nos saldra en consola el siguiente mensaje en la ejecucion `test/test_case_stack.py::Tests::test_ir_usuarios SKIPPED (en desarrollo)` este se coloca como decorador arriba de nuestra funcion test.

Opciones para mas detalles en la ejecucion

Aqui hay algunas opciones comunes junto con **-k**:

1. Modo verbose **-v**: proporciona información detallada sobre las pruebas que se están ejecutando

```
test -k "nombre_de_la_prueba" -v
```

2. Mostrar información de introducción **-l**: muestra información de introducción sobre las pruebas seleccionadas

```
pytest -k "nombre_de_la_prueba" -l
```