

# DOCUMENTACIÓN TEST

Verónica Toro Gómez

## VERÓNICA TORO GÓMEZ

La finalidad de los test es probar la base de datos de tu aplicación, en este caso, yo he usado mi aplicación matrícula realizada en django, y he usado el propio archivo test.py que viene por defecto en mi aplicación.

```
from django.test import TestCase
from matricula.models import Curso, Alumno
# Create your tests here.

class CursoTestCase(TestCase):
    def setUp(self):
        Curso.objects.create(abrev="BACH", denom="Bachillerato", imagen="fotos/thor.jpg")
        Curso.objects.create(abrev="ESO", denom="Secundaria" )

    def test_curso_crear_curso(self):
        BACH = Curso.objects.get(abrev="BACH")
        ESO = Curso.objects.get(abrev="ESO")
        self.assertEqual(BACH.__str__(), "Bachillerato")
        self.assertEqual(ESO.imagen, "")

class AlumnoTestCase(TestCase):
    def setUp(self):
        Alumno.objects.create(dni="11111111A", nombre="Verónica", apellidos="Toro Gómez", imagen="fotos/caty.jpg")

    def test_alumno_crear_alumno(self):
        DNI = Alumno.objects.get(dni="11111111A")
        self.assertEqual(len(DNI.dni), 9)
```

Hemos importado las clases Curso y alumno de los modelos de matricula, para poder utilizarlos en el test. También hemos importado TestCase que viene por defecto en django.test para poder realizar las pruebas

Hemos creado la clase CursoTestCase y le introducimos por parámetro TestCase, esta opción es por defecto y se implementará en todas las clases del test, y luego la función def setUp(self), esto es innato de las funciones de test.

El funcionamiento es sencillo, en el test alumno crea un alumno y luego comprueba que se ha creado con la longitud de dni 9. Y en el test curso comprueba que se ha creado correctamente un curso sin imagen (para eso debe permitirse en la base de datos null)

Dando como resultado en los dos ok, significando que la aplicación funciona correctamente.