

Ejercicios Angular2 Tests

[ejercicioTest0001:](#)

- Crear un nuevo proyecto `ng new nombre-del-proyecto`
- Bajo el directorio src crear "test-array.spec.ts"
 - Comprobar que la longitud de un array de números (number[]) no debe ser cero
 - Demostrar que dado un array de números cuyos elementos son pares e impares, la suma de los elementos pares del array también es un número par
 - Usaremos los elementos clave de Jasmine: "describe", "it", "expect", "beforeEach", "afterEach" y "toBe", esto es, sin intervención del api de testing específica de Angular
- Sugerencias:
 - examinar el [API](#) de Array
 - en el repo [angular\pruebas\ejercicios\ejercicioTest0001](#)

Ejercicios Angular2 Tests

[ejercicioTest0002:](#)

- Crear un nuevo proyecto `ng new nombre-del-proyecto`
- Crear un componente `ng g component test`
- Crear un servicio `ng g service test`
- *Enganchar* el servicio al componente. Arrancar la app `ng serve -o` y ver que funciona
- Arrancar los test generados por Angular y ver que funcionan: `ng test`
- Crear un método en el servicio que devuelva un array de números:
`obtenerNumeros():number[]`
- En el template del componente, usando un `*ngFor`, mostrar todos los números devueltos por el servicio
- Aprovechar los tests generados por Angular para probar la interacción del componente con el servicio y el propio servicio
- Para probar el servicio, comprobar que devuelve un array cuya longitud es mayor que cero
- Para comprobar la interacción del componente:
 - simular el servicio usando un stub

- Localizar el elemento li, y demostrar que su contenido es el esperado
- Sugerencias:
 - En el repo [angular\pruebas\ejemplos\ejemploTest0004](#)
 - en el repo [angular\pruebas\ejercicios\ejercicioTest0002](#)