```
describe('ItemCtrl with SpyReturn', function() {
    beforeEach(module('notesApp'));

    var ctrl, itemService;

    beforeEach(inject(function($controller, ItemService) {
        spyOn(ItemService, 'list')
            .and.returnValue([{id: 1, label: 'Mock'}]);
        itemService = ItemService;
        ctrl = $controller('ItemCtrl');
    }));

    it('should load mocked out items', function() {
        expect(itemService.list).toHaveBeenCalled();
        expect(itemService.list.calls.count()).toEqual(1);
        expect(ctrl.items).toEqual([{id: 1, label: 'Mock'}]);
    });
});
```

SPIES

➤ In Jasmine, mocks are referred to as spies.

Spies allow us to hook into certain

functions, and check whether they:

1. were called

2. how many times they were called

3. what arguments they were called with

4. define value that has to be retuned

UNIT TESTING SERVER CALLS

init() function in controller

```javascript
this.init = function() {
    var self = this;
    OnlineMortgage.get(function(obj) {
        $scope.onlineMortgage = obj;
        self.initMortgageAmount();
    });
};
this.init();
```