

Test Doubles

As described by Martin Fowler in his article "[Mocks aren't stubs](#)"



Martin Fowler

What is a test double?

Test Double is a generic term used to describe any case where you replace a production object for testing purposes. Test doubles come in many forms, for example...



Dummy

Empty, minimal implementations. Usually just used to fill parameter lists.



Fake

Have some made-up or simplified implementation used only for testing. (ex: in memory database)



Stub

Provide pre-canned returns to method calls. (ex: `dice.roll()` always produces a 4)



Spy

Often calls REAL behavior, but keeps track of how it was called (how many times, by whom, with what params, etc) to test interactions as well as state.



Mock

Stubbed behavior with pre-programmed expectations for how they'll be called. Interaction-based testing.

Example - Dice Cup

```
class DieCupTest {  
    DieCup testObj;  
  
    @Test  
    void initializeHasRightNumberOfDie() {  
        Die die1 = new Die(numSides: 6);  
        Die die2 = new Die(numSides: 12);  
        List<Die> diceToInject = new ArrayList<Die>();  
        diceToInject.add(die1);  
        diceToInject.add(die2);  
  
        testObj = new DieCup(diceToInject);  
  
        assertEquals(diceToInject.size(), testObj.dice.size());  
    }  
}
```

Initialization is easy enough...



```
public class DieCup {  
    List<Die> dice = new ArrayList<Die>();  
  
    public DieCup(List<Die> dice) {  
        this.dice = dice;  
    }  
}
```