

Test Doubles: Mocks, Stubs and Fakes



Emily Bache
TECHNICAL AGILE COACH

@emilybache coding-is-like-cooking.info

Summary

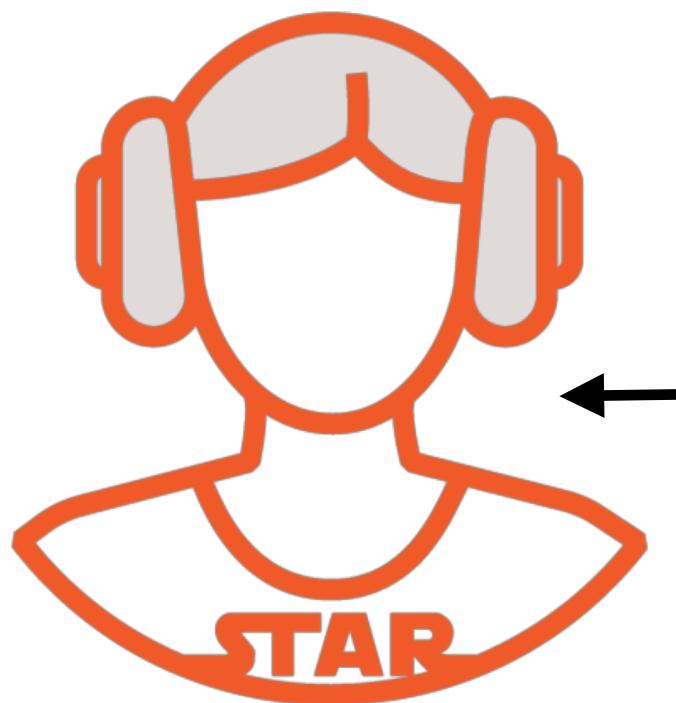
What is a Test Double?

Why use a Test Double?

Inject Test Doubles using Monkeypatching

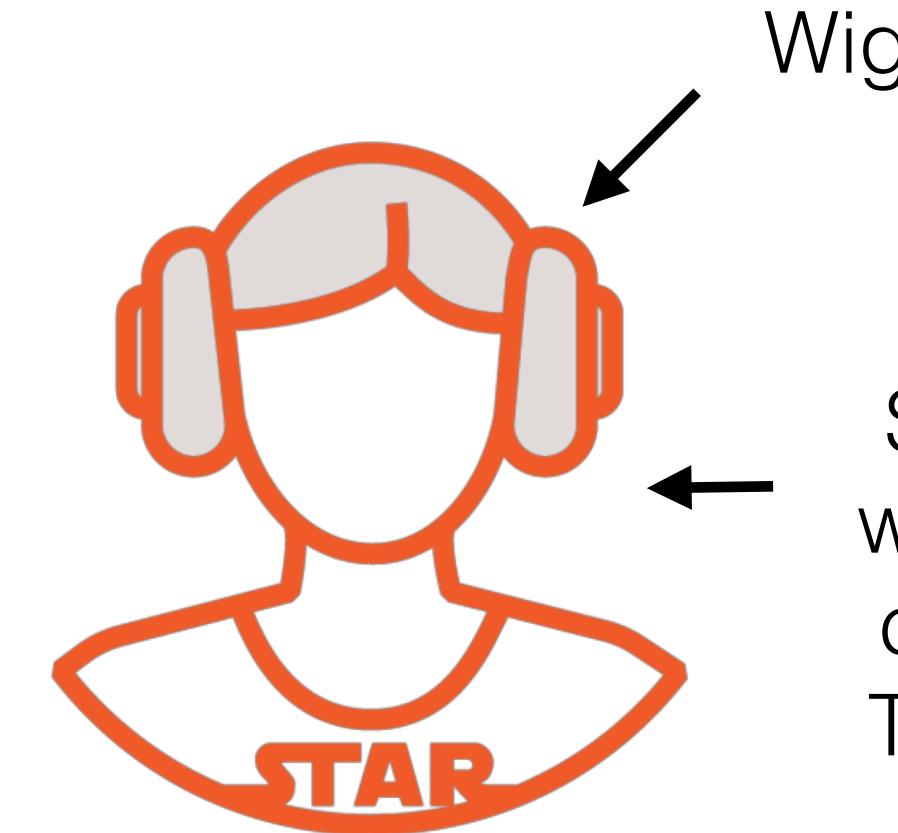
Test Double - Like a Stunt Double

The real actor



Famous
actress

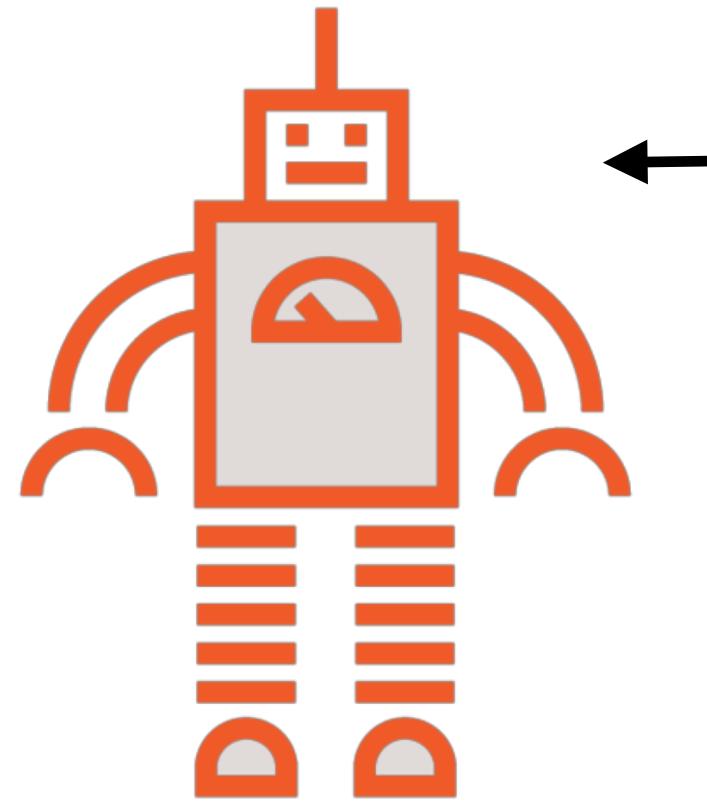
The stunt double



Wig
Some
woman
called
Tracey

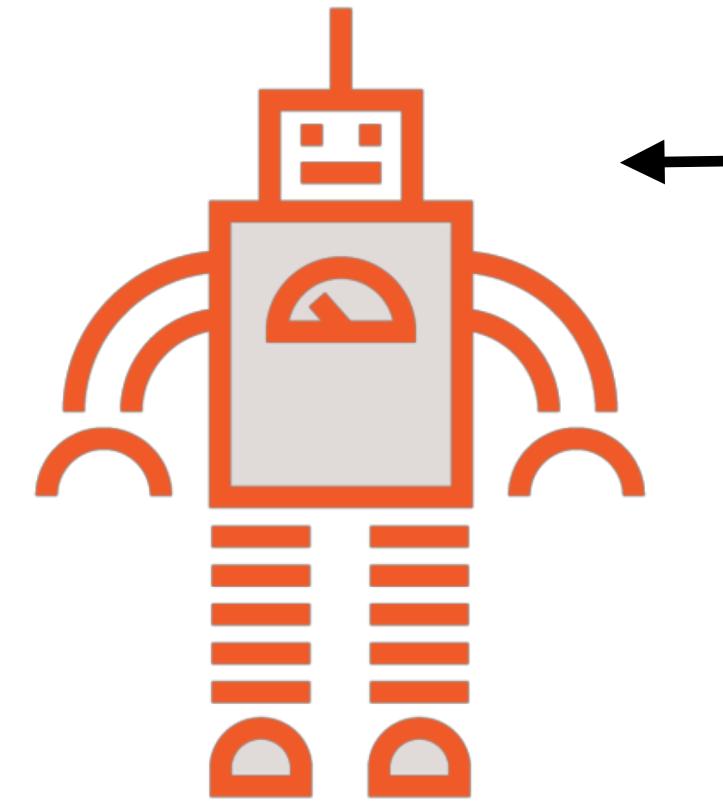
Test Double - Like a Stunt Double

The real object



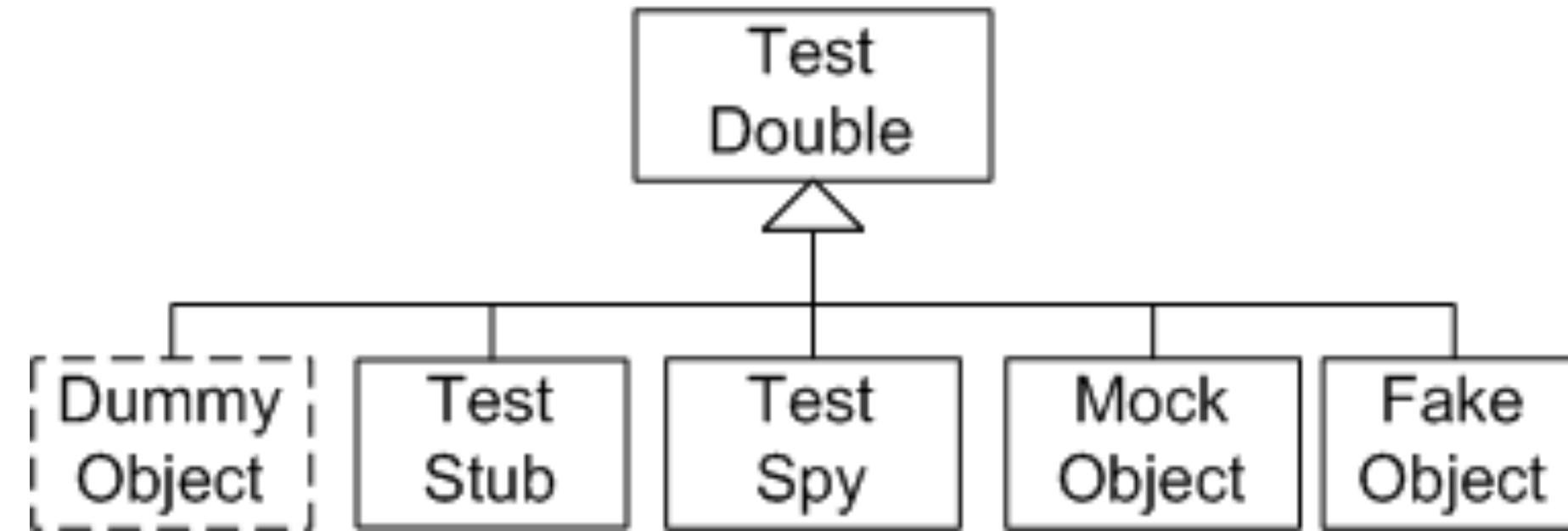
Complex,
production
logic

The test double



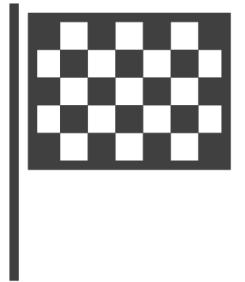
Everything it
does is
controlled
by the test

Different Kinds of Test Double

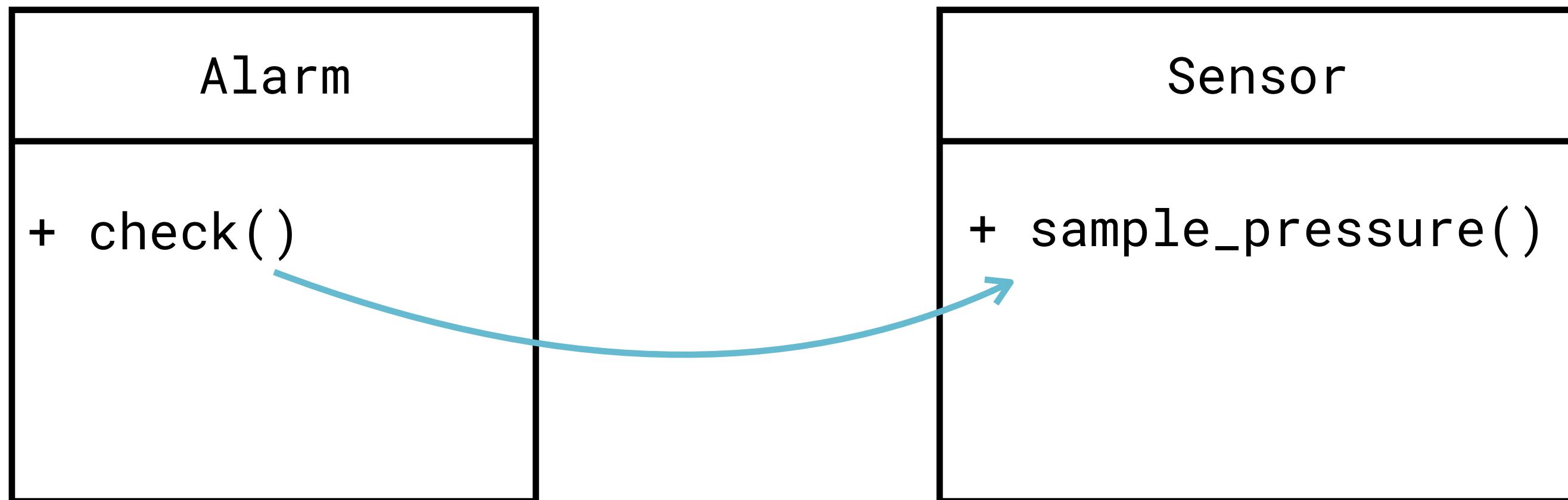
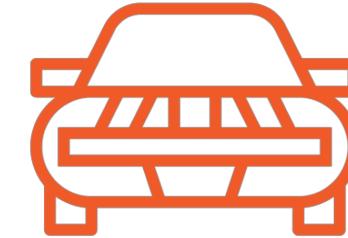


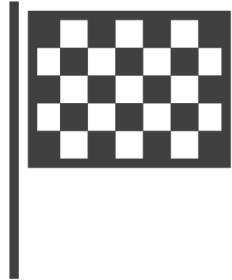
from <http://xunitpatterns.com>

see also: https://en.wikipedia.org/wiki/Test_Double

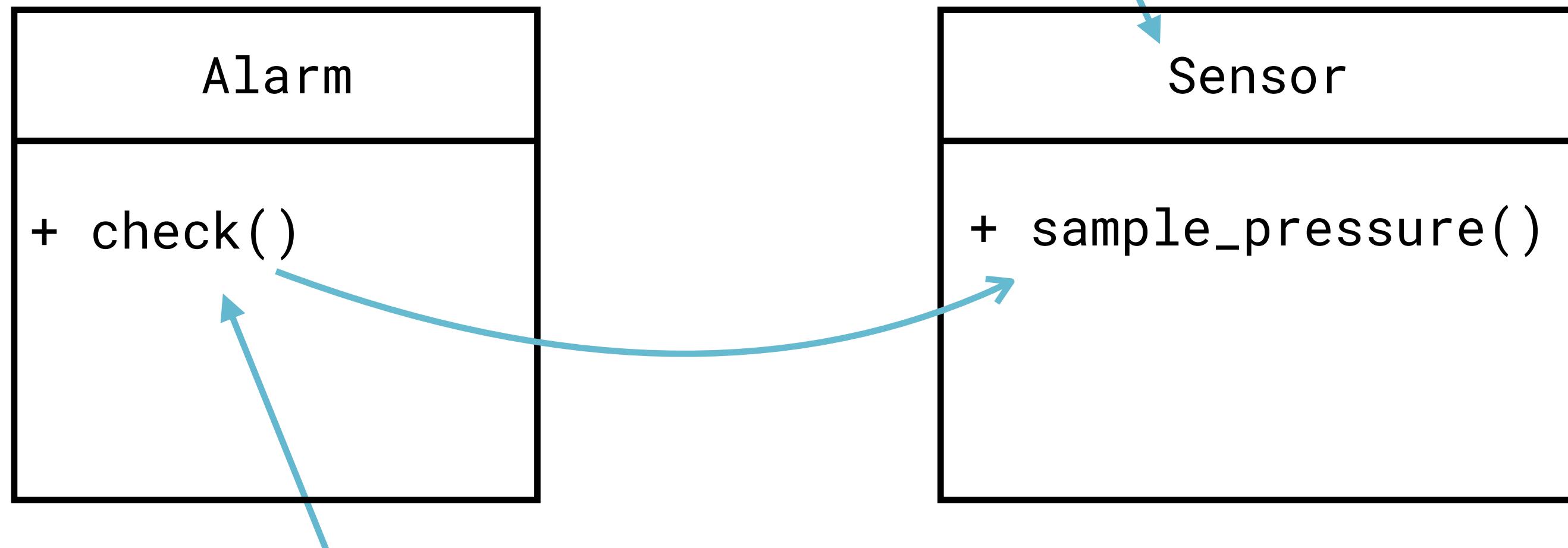
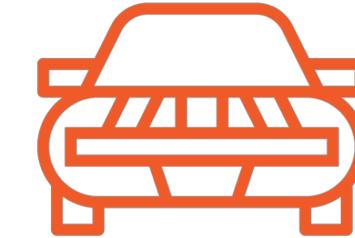


Racing Car Example



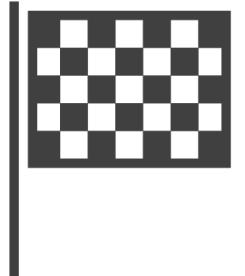


Racing Car Example

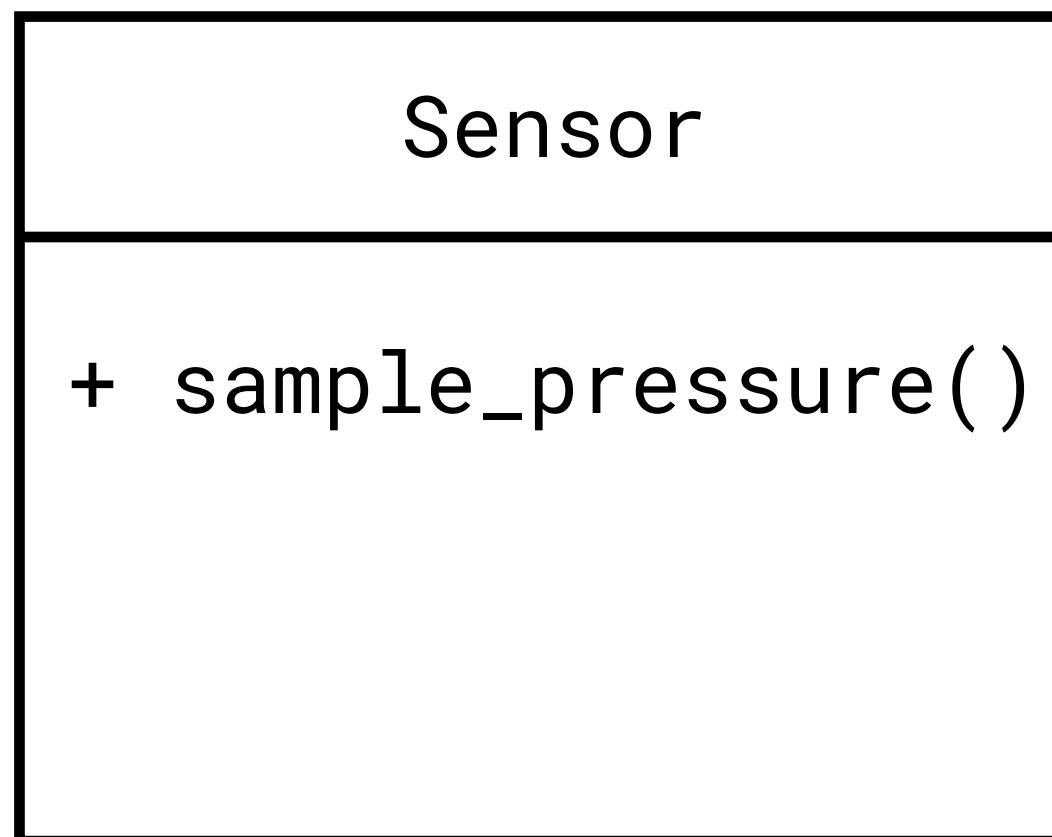
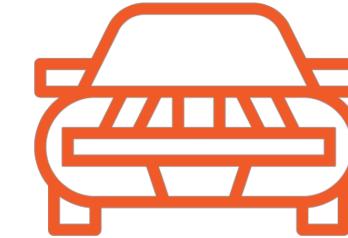


Replace this collaborator with a test double

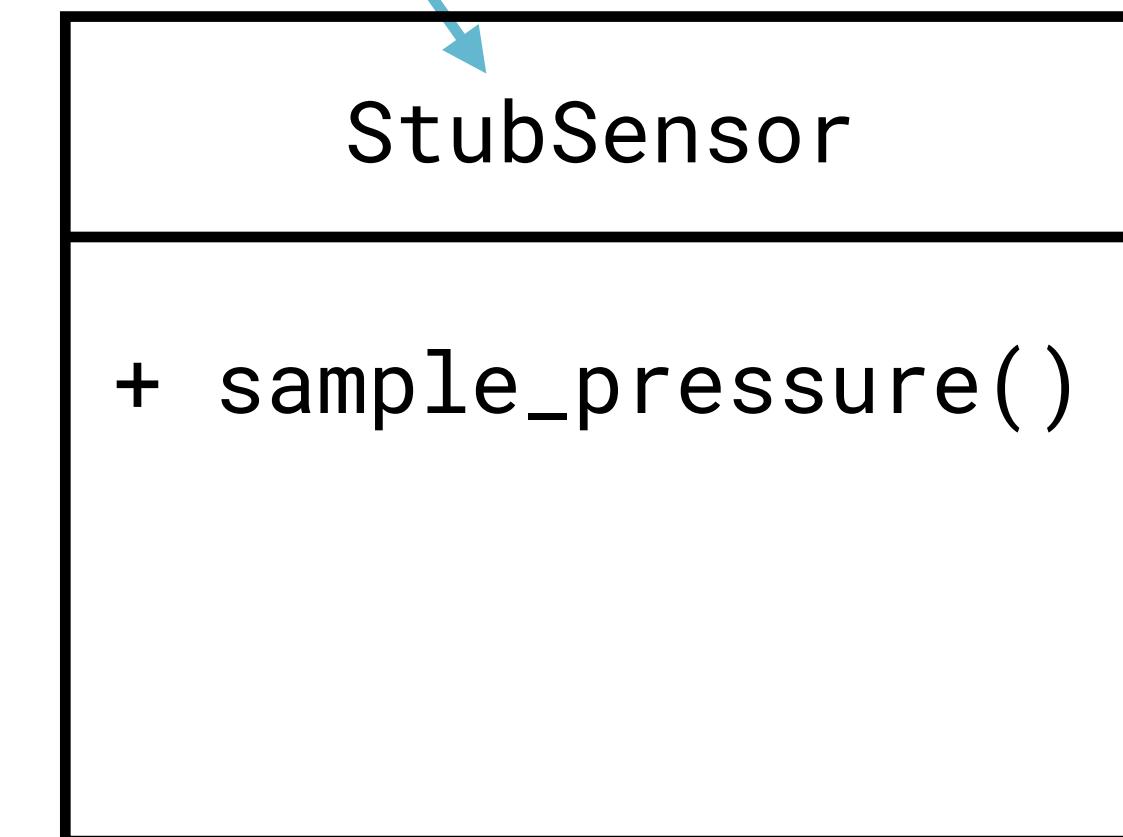
Want to test this method



Racing Car Example



Same methods as a Sensor



A Stub has the same methods as the class it replaces,
but the implementation is very simple

```
class StubSensor:  
    def sample_pressure(self):  
        return 15
```

Stub

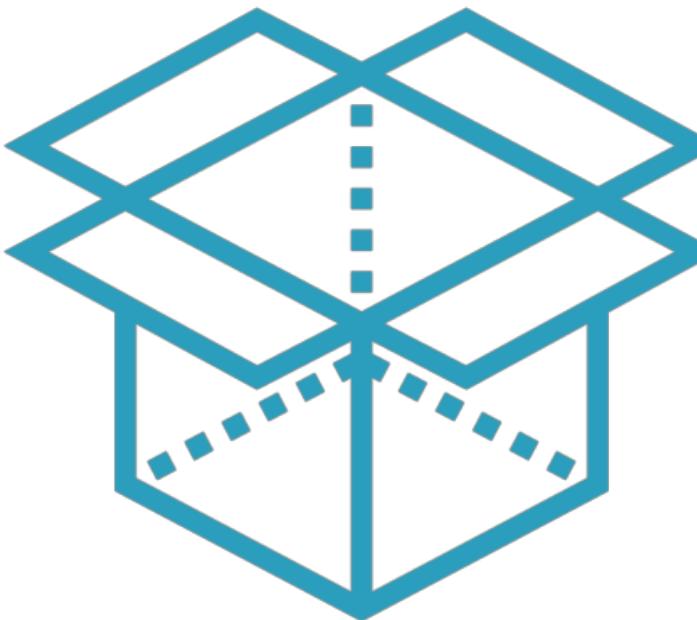
```
def test_normal_pressure_alarm_stays_off():
    stub_sensor = Mock(Sensor)
    stub_sensor.sample_pressure.return_value= 18
    alarm = Alarm(sensor=stub_sensor)
    alarm.check()
    assert not alarm.is_alarm_on
```

Test Stub Using ‘unittest.mock’

Stubs

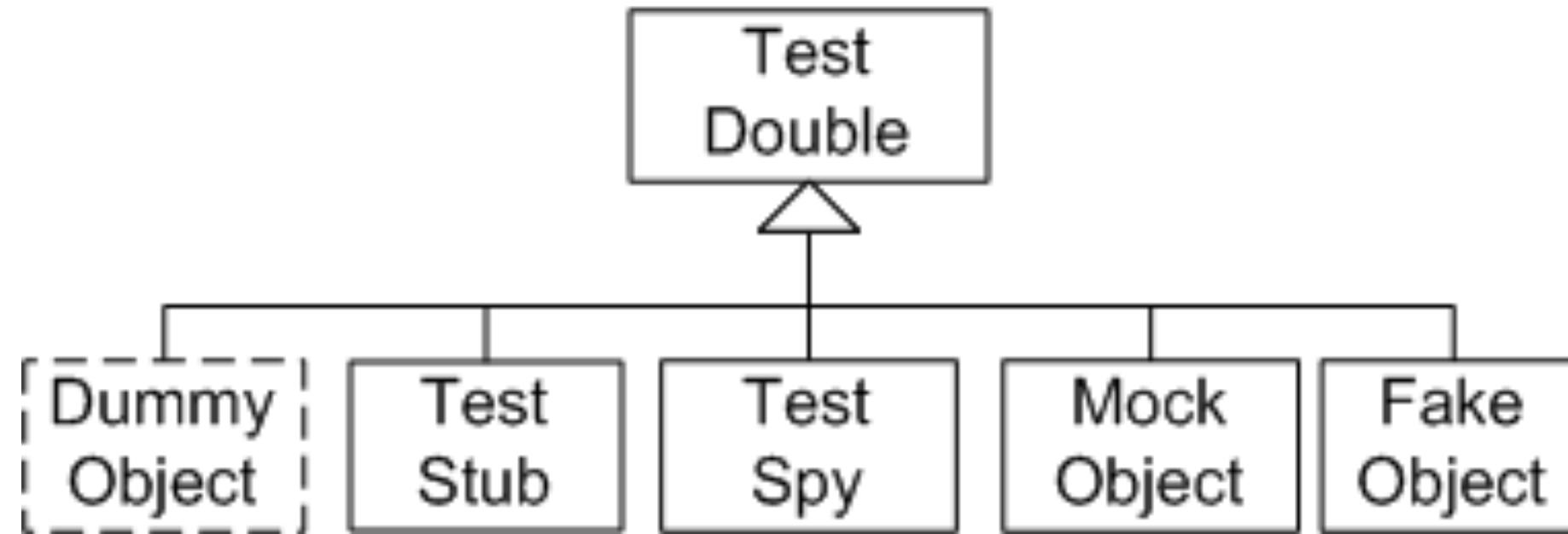


Looks good from the outside



**Contains nothing except what
you put there**

Different kinds of test double

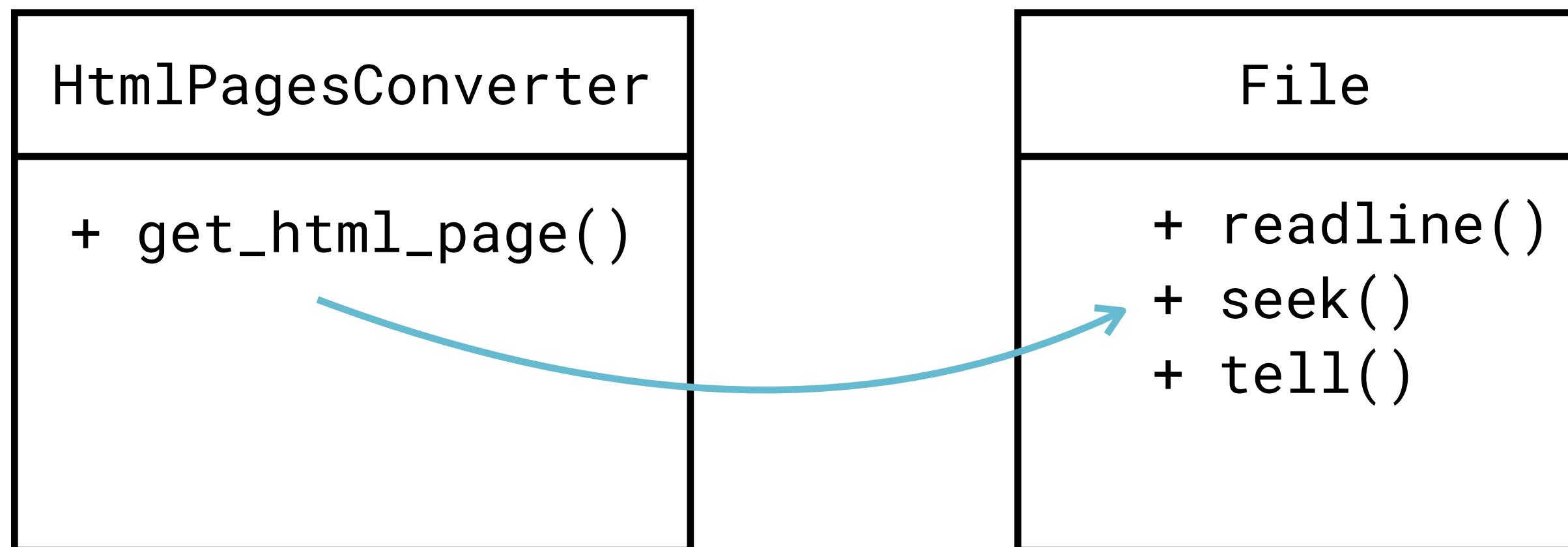


from <http://xunitpatterns.com>

see also: https://en.wikipedia.org/wiki/Test_Double



Html Converter Example

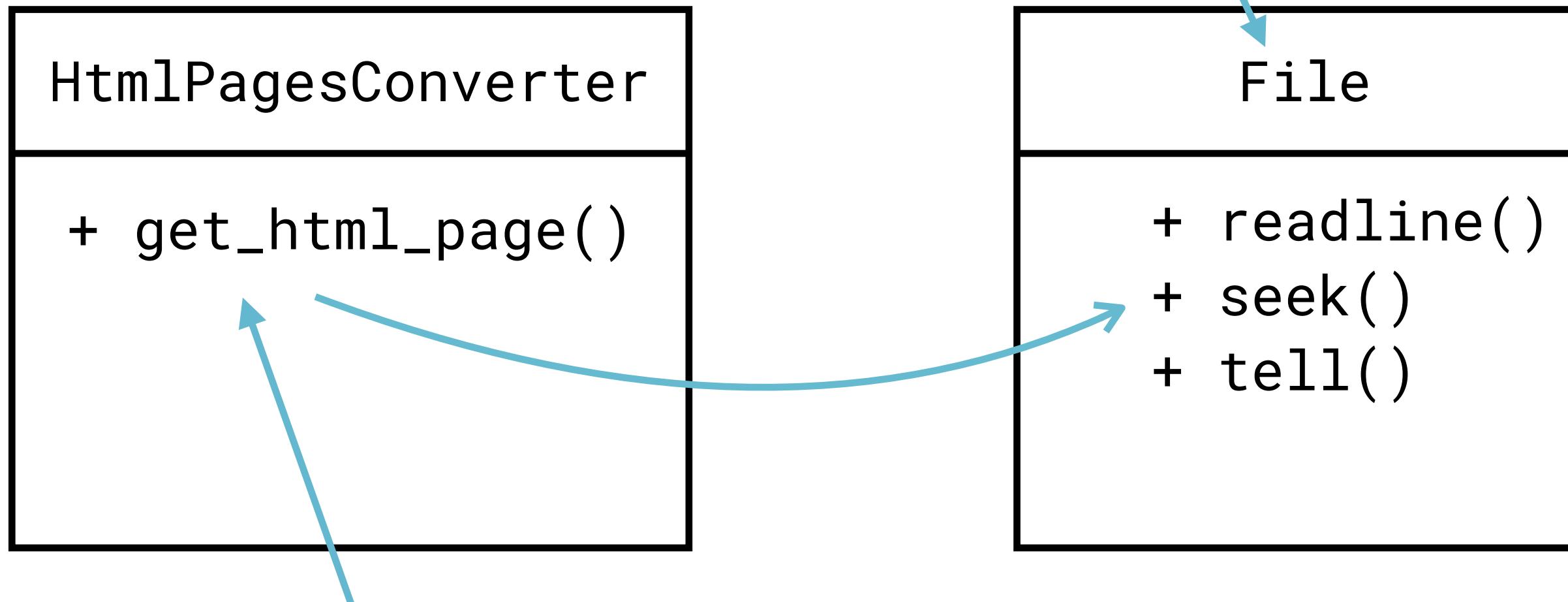




Html Converter Example



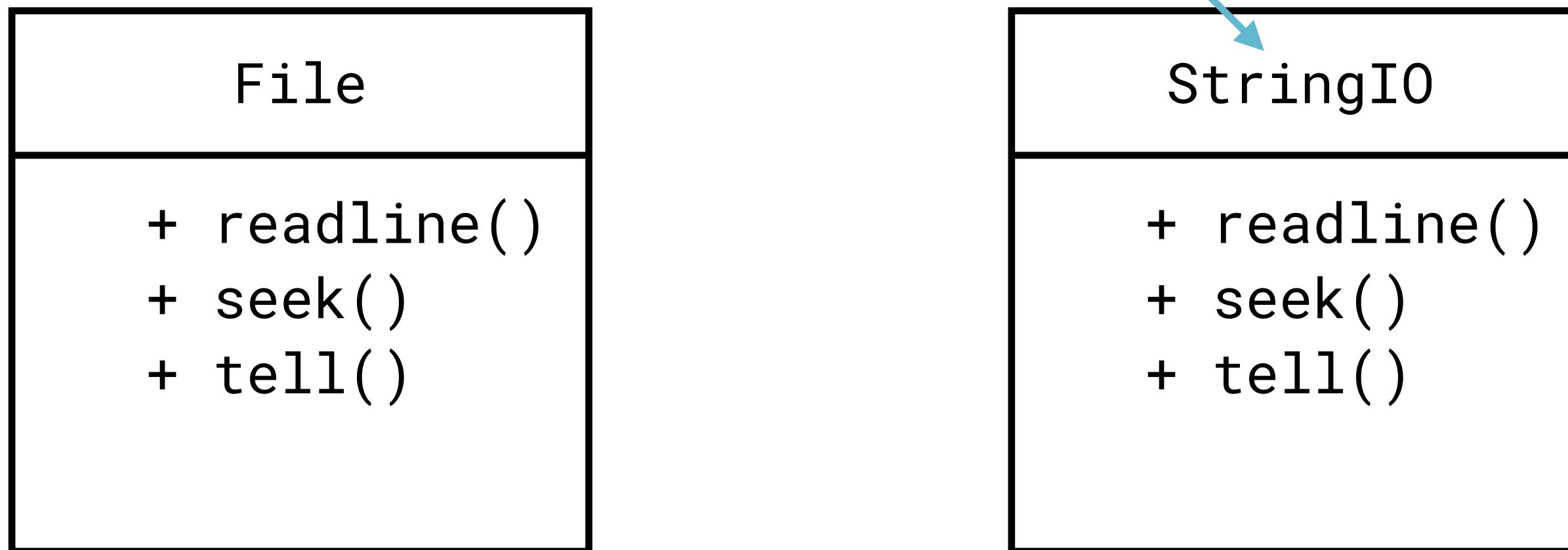
Replace this collaborator with a test double



Want to test this method



Html Converter Example

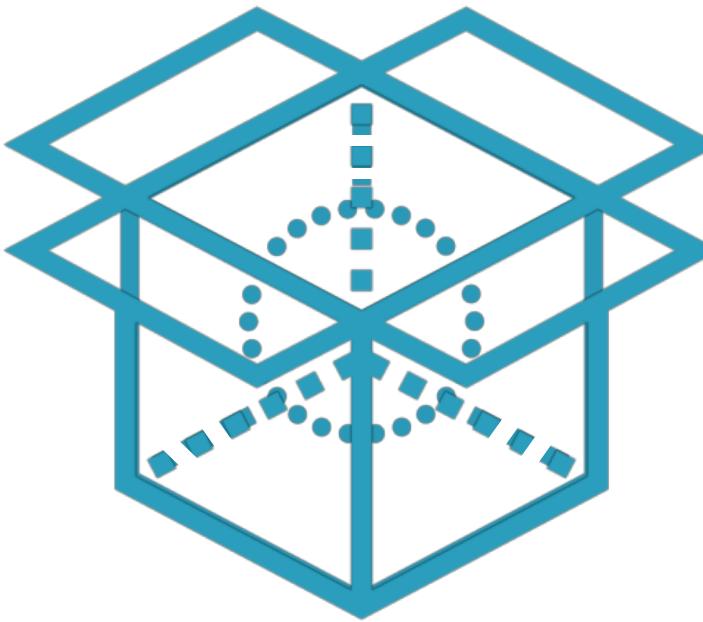


A Fake has an implementation with logic and behaviour,
but is not suitable for production

Fakes

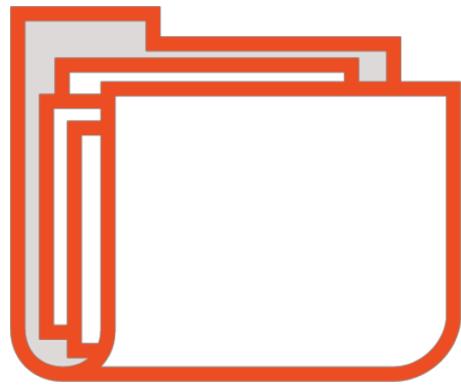


Looks good from the outside



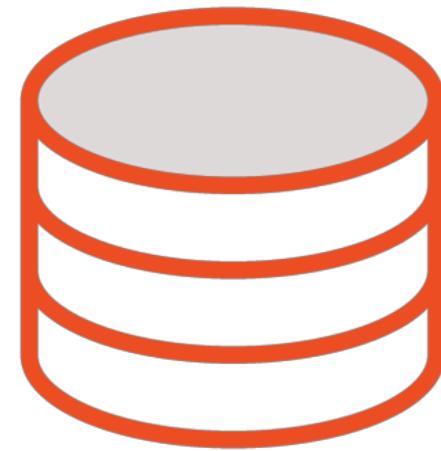
**Has an implementation with
logic and behaviour**

Common Things to Replace with a Fake



File

Replace with `StringIO`



Database

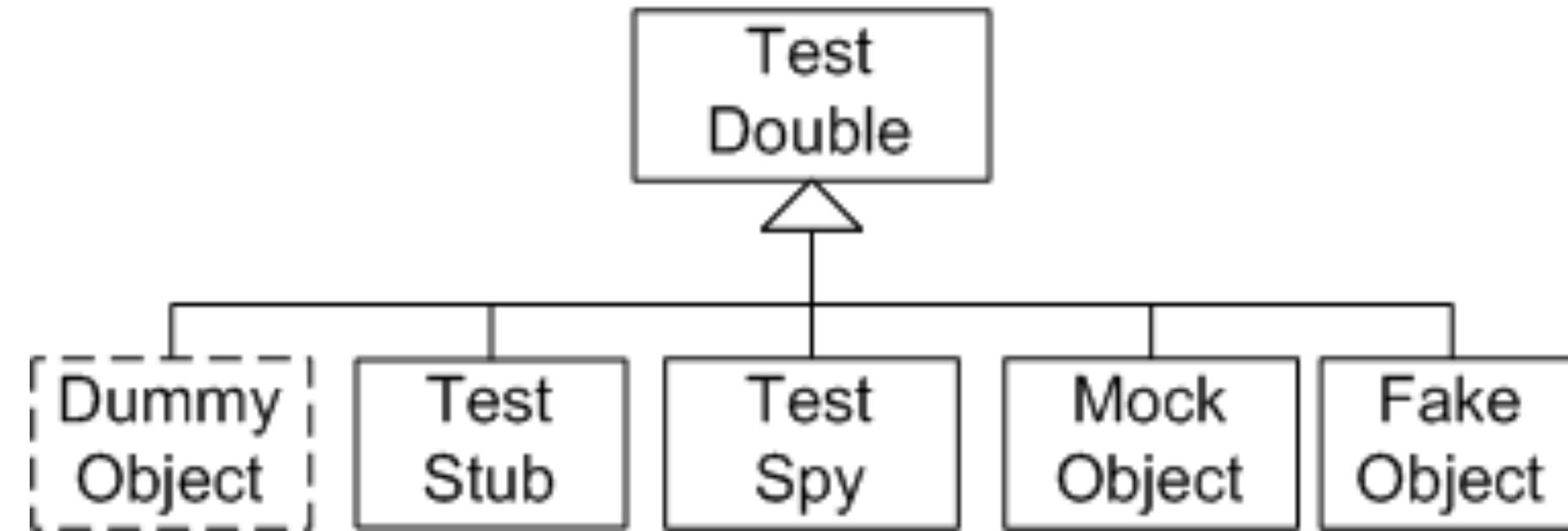
Replace with in-memory database



WebServer

Replace with lightweight Web Server

Different Kinds of Test Double



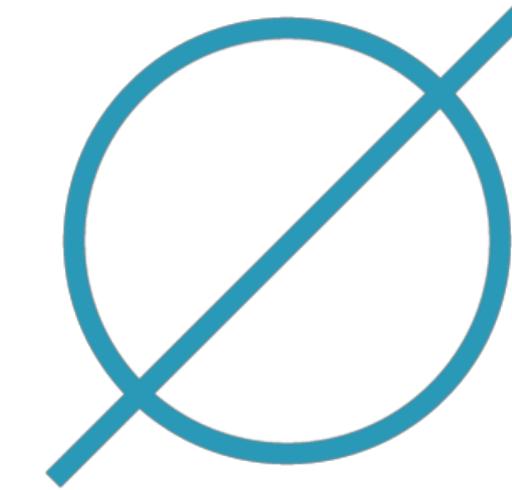
from <http://xunitpatterns.com>

see also: https://en.wikipedia.org/wiki/Test_Double

Dummies

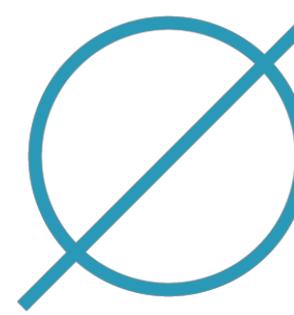


It doesn't matter what it looks like



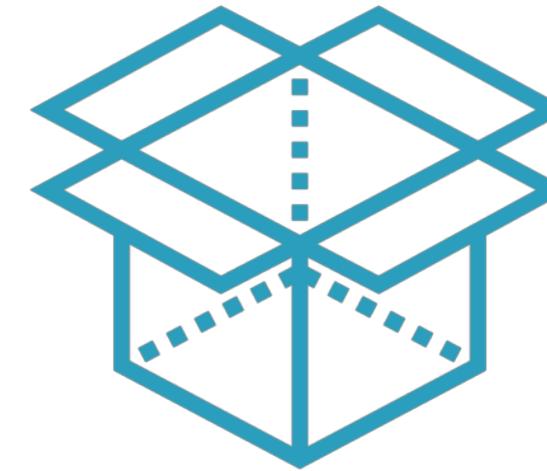
A dummy is not used

Test Doubles replace collaborators



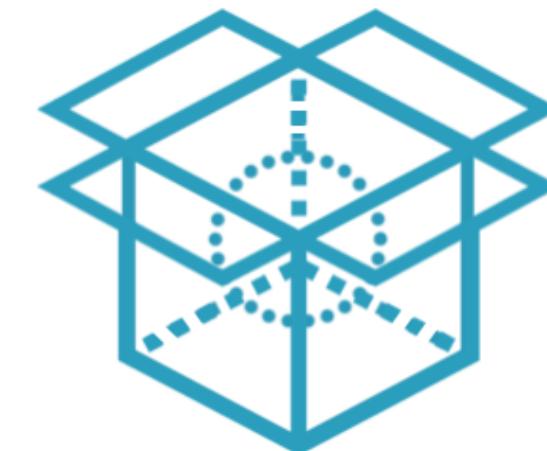
Dummy

Usually None



Stub

Responds with fixed,
pre-prepared data



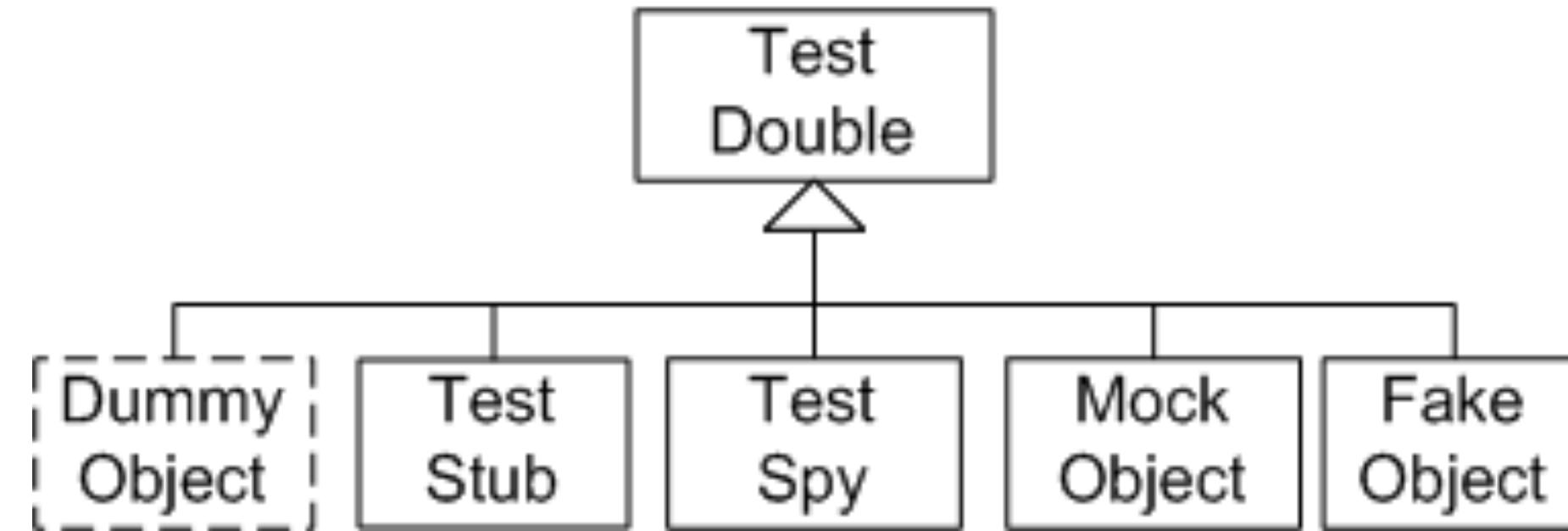
Fake

Has an implementation,
but unsuitable for
production



More similar to the
collaborator being replaced

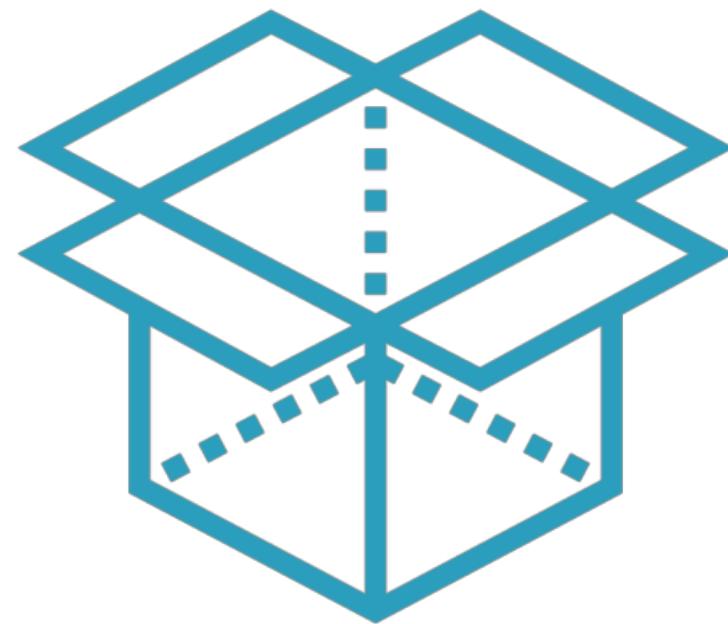
Different Kinds of Test Double



from <http://xunitpatterns.com>

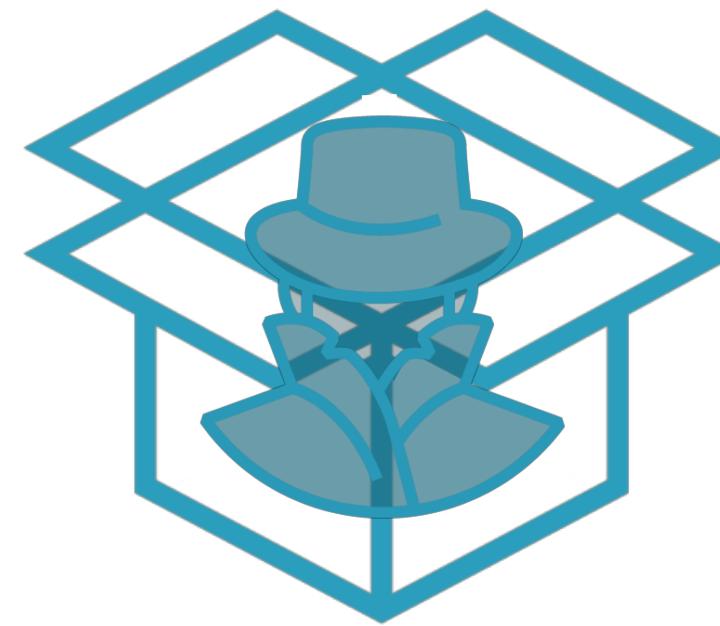
see also: https://en.wikipedia.org/wiki/Test_Double

Mocks and Spies Do More Than Stubs



Stub

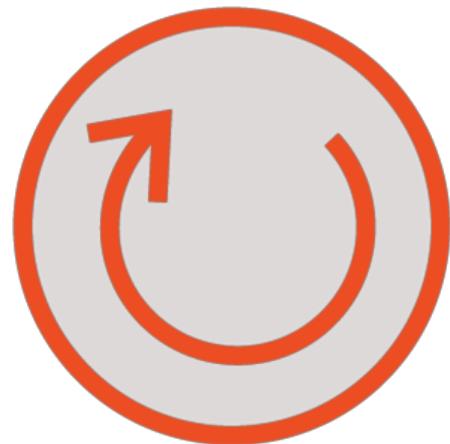
Will not fail the test



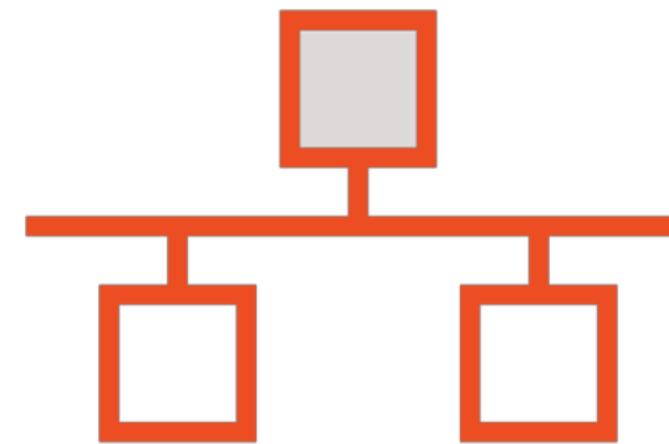
Mock or Spy

**Can fail the test
if it's not called correctly**

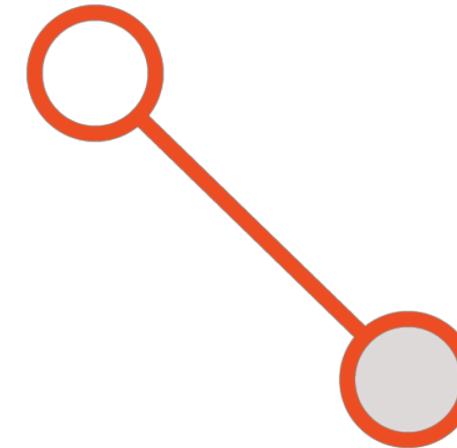
Three Kinds of Assert



Return value
or an exception



State change
Use an API to query
the new state



Method call
Did a specific method
get called on a
collaborator



Increasing complexity

```
def test_convert_quotes():
    fake_file = io.StringIO("quote: ' ")
    converter = HtmlPagesConverter(open_file=fake_file)
    converted_text = converter.get_html_page(0)
    assert converted_text == "quote: &#x27;<br />"
```

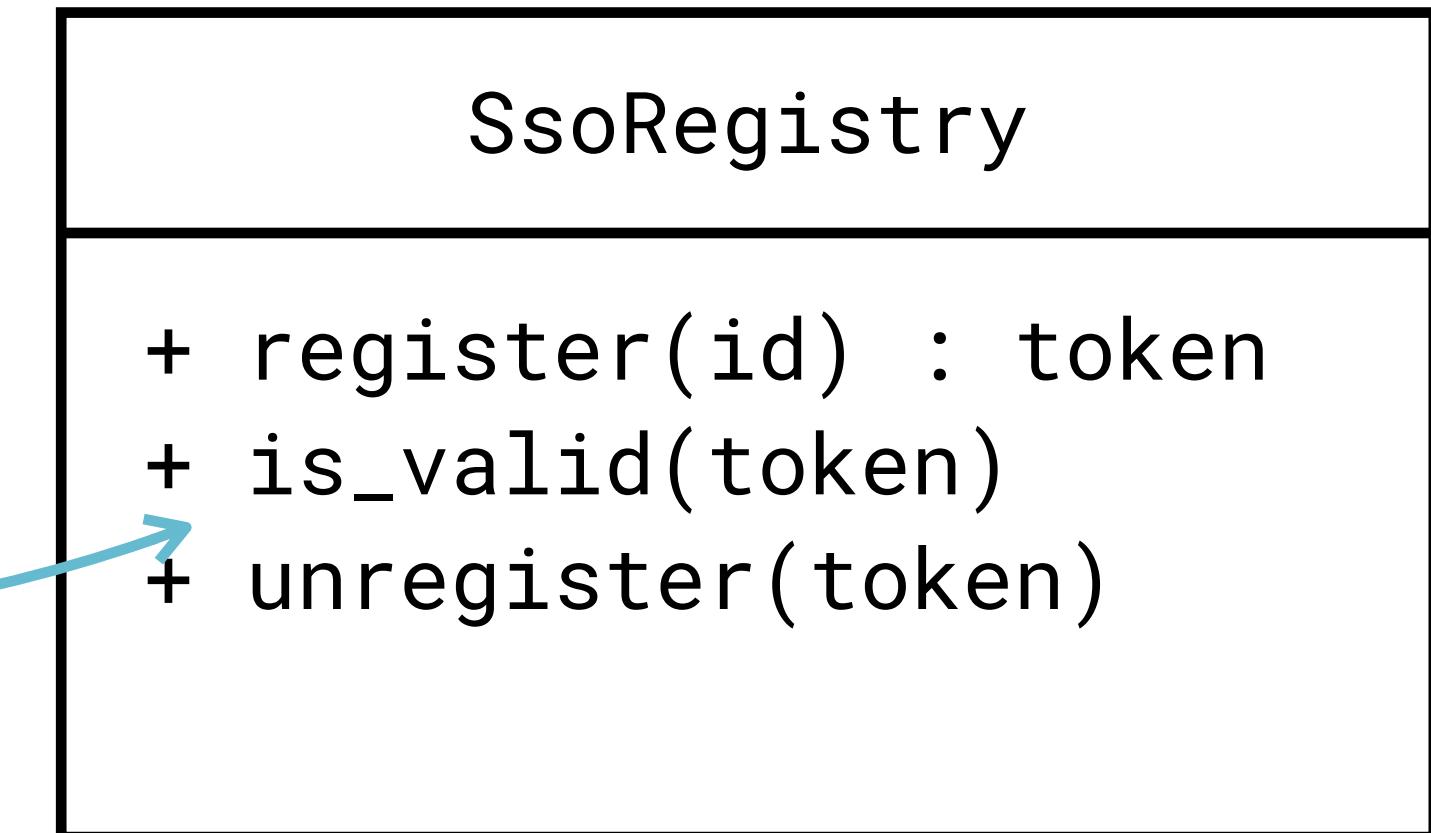
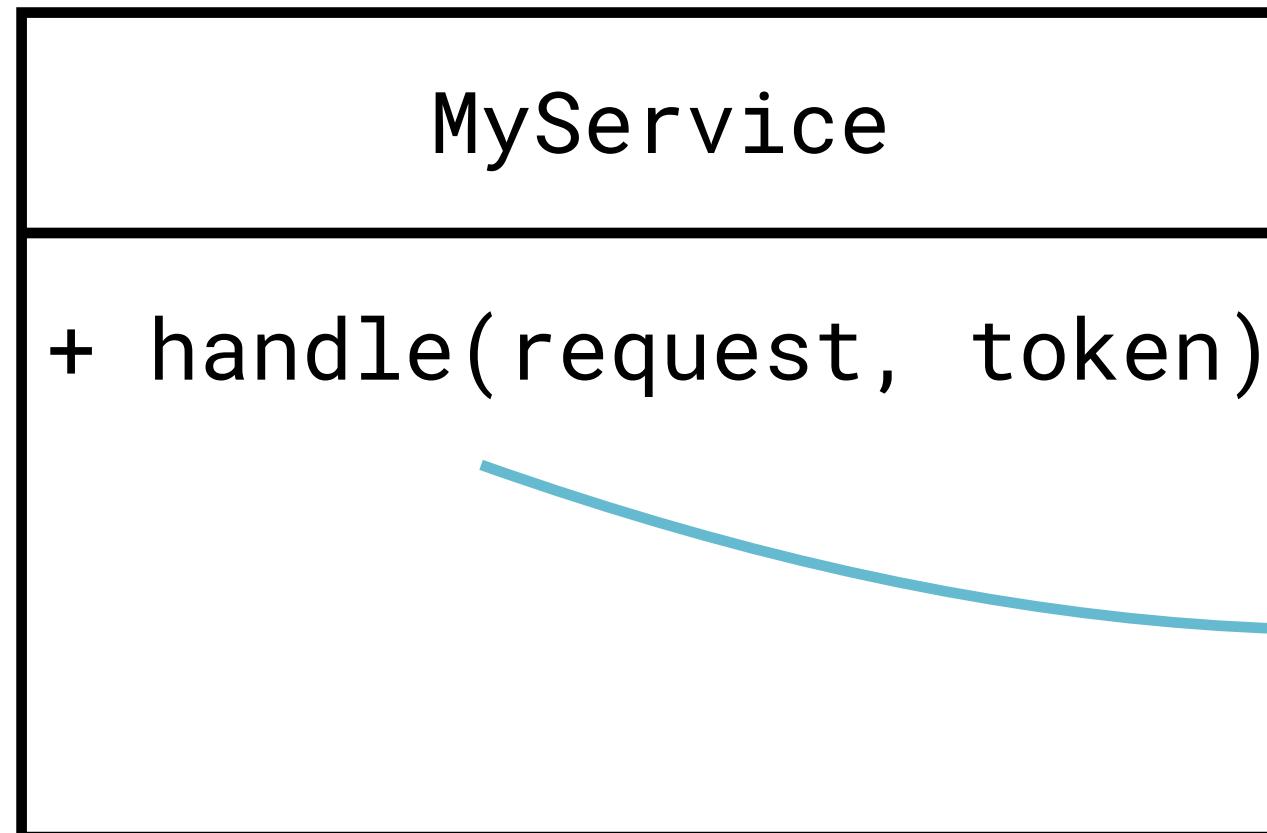
Assert on the method return value

```
def test_alarm_is_off_by_default(self):  
    alarm = Alarm()  
assert not alarm.is_alarm_on
```

Assert on the alarm state

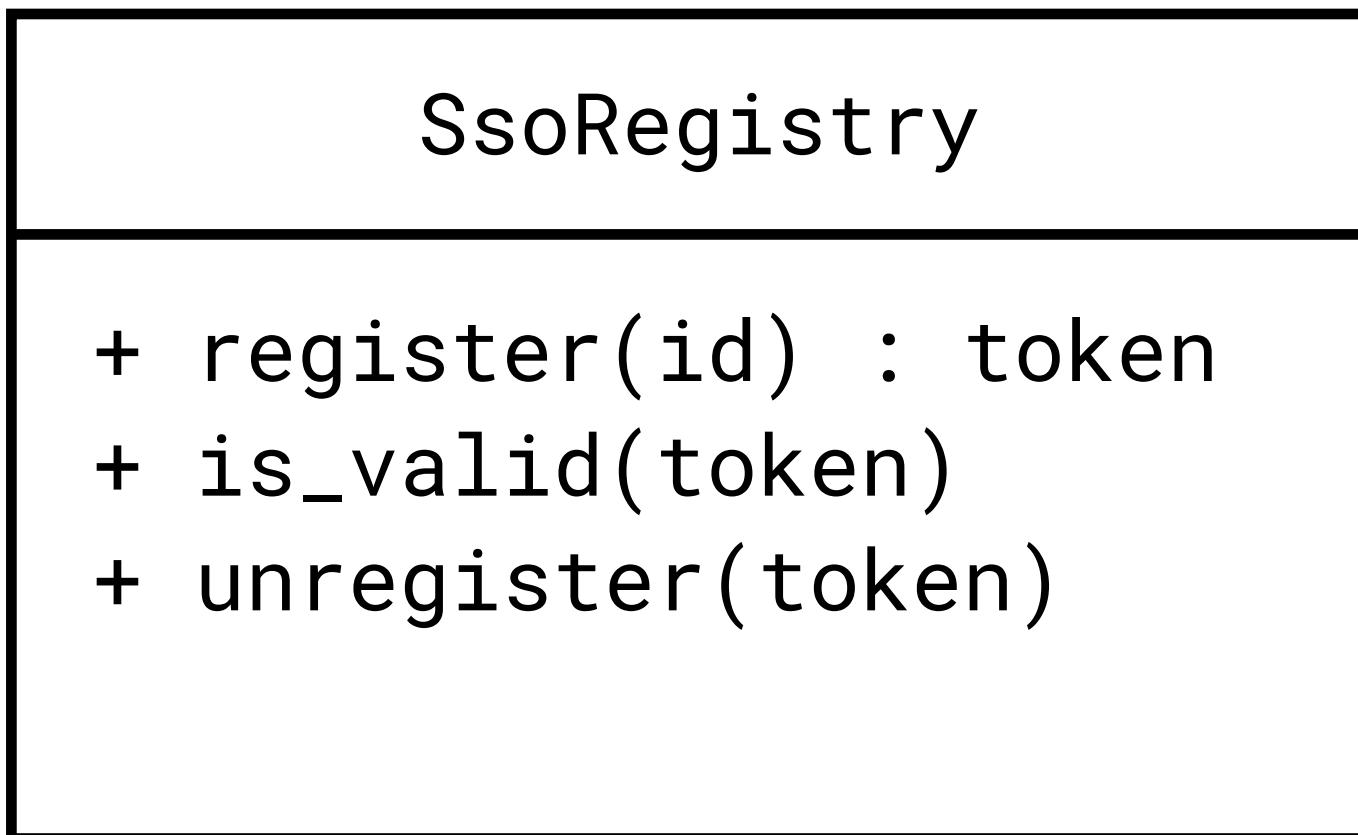


Single Sign On Example

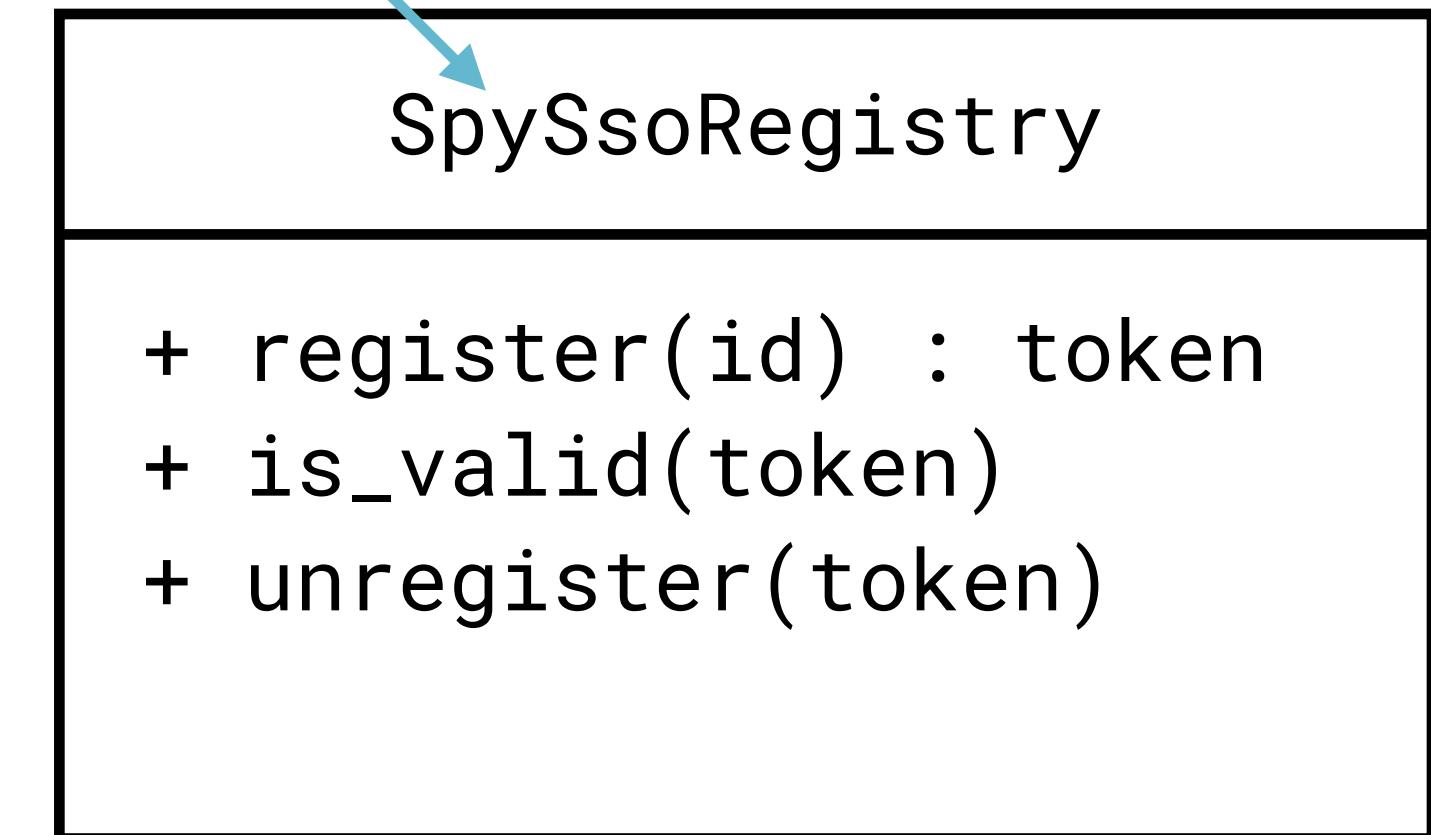




Single Sign On Example



Same methods as a SsoRegistry

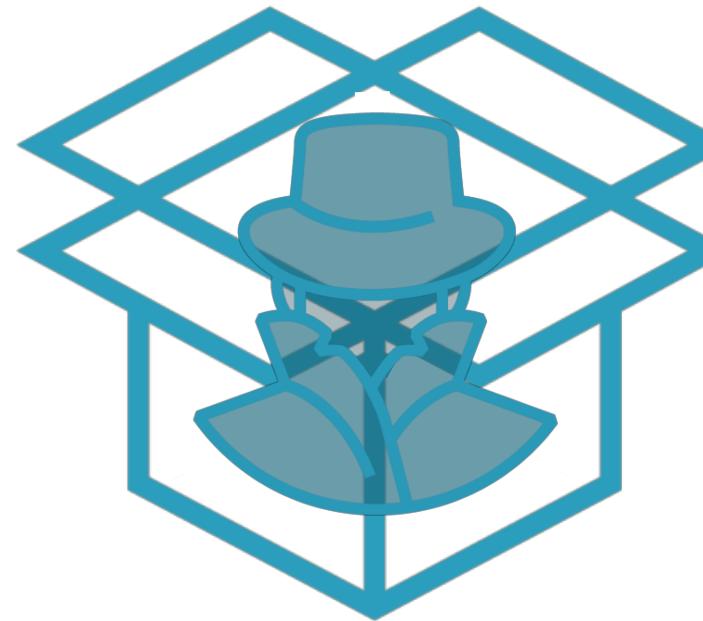


A Spy records the method calls it receives,
so you can assert they were correct

Spies

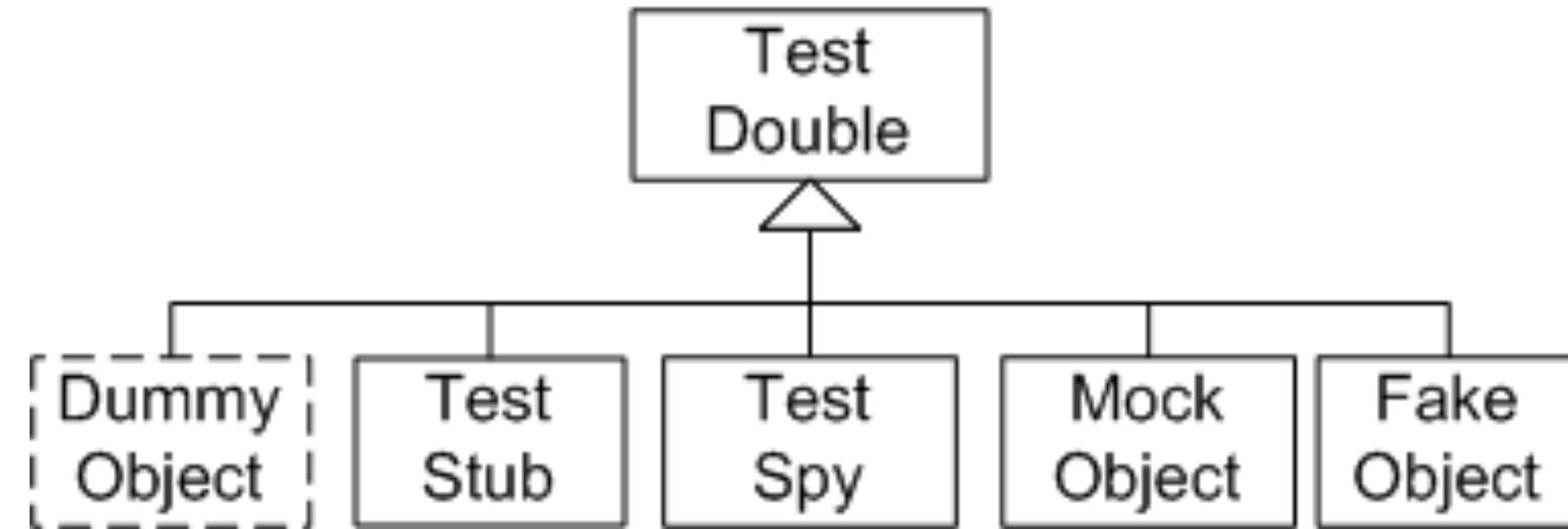


Looks good from the outside



Records the method calls it receives

Different Kinds of Test Double

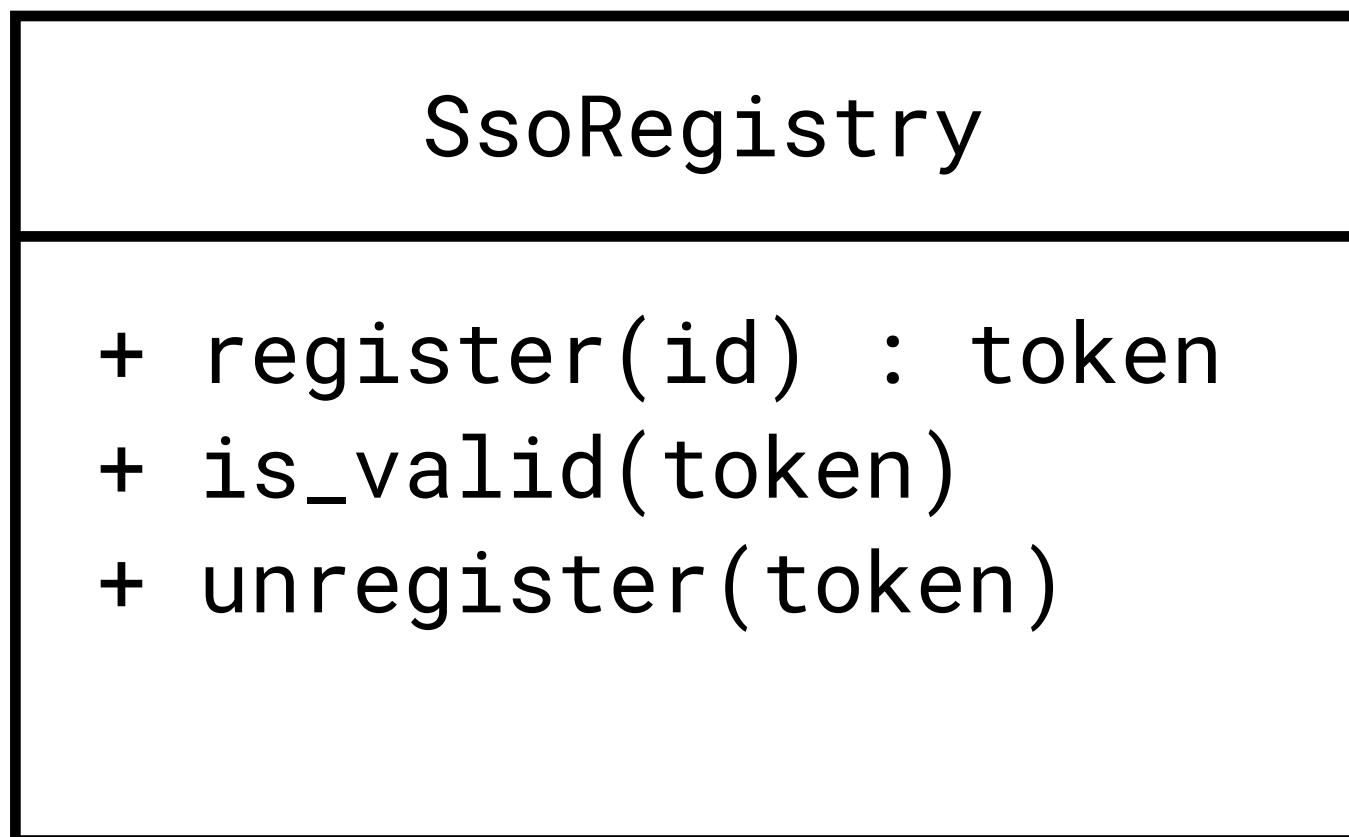


from <http://xunitpatterns.com>

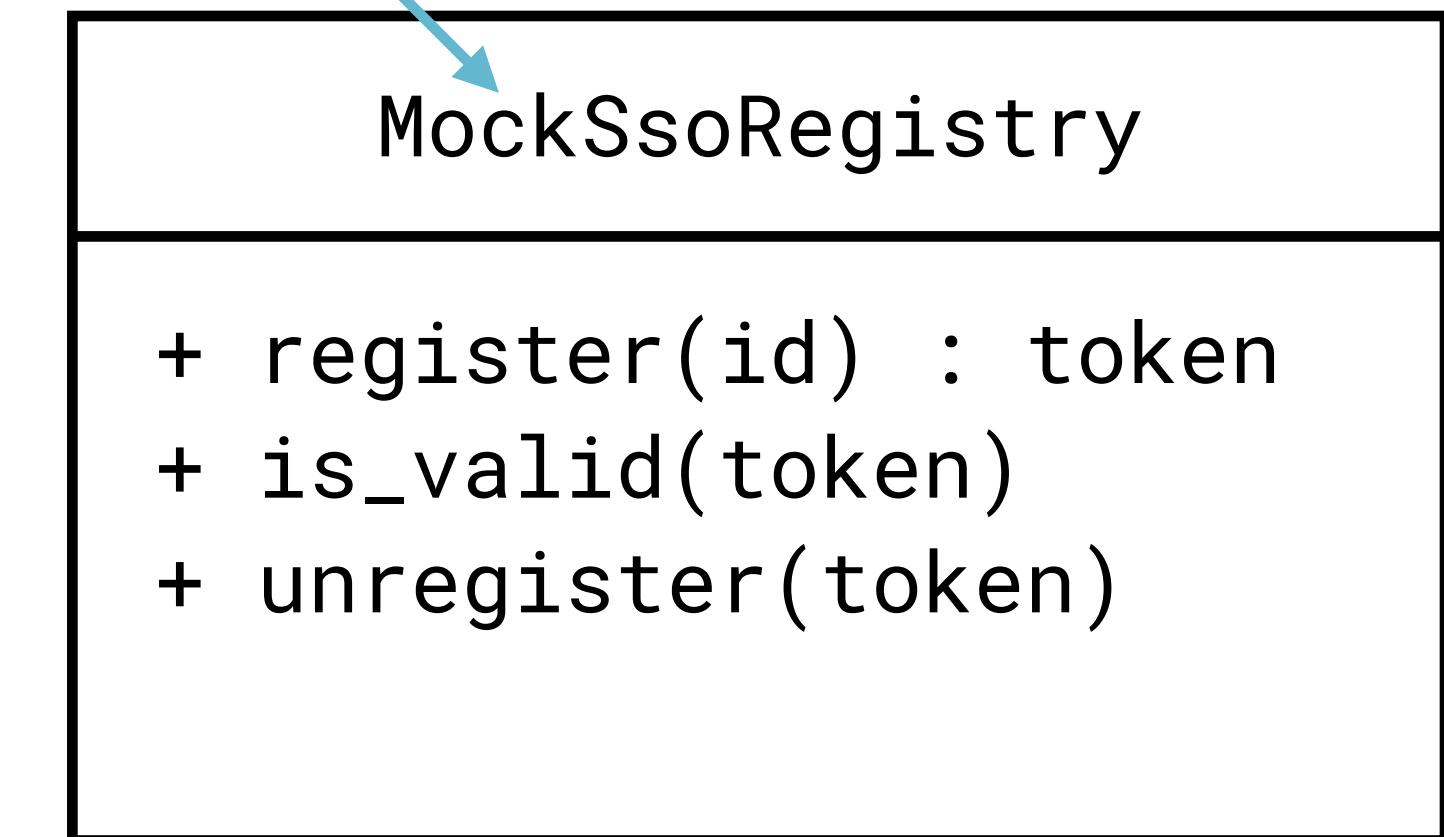
see also: https://en.wikipedia.org/wiki/Test_Double



Single Sign On Example



Same methods as a SsoRegistry

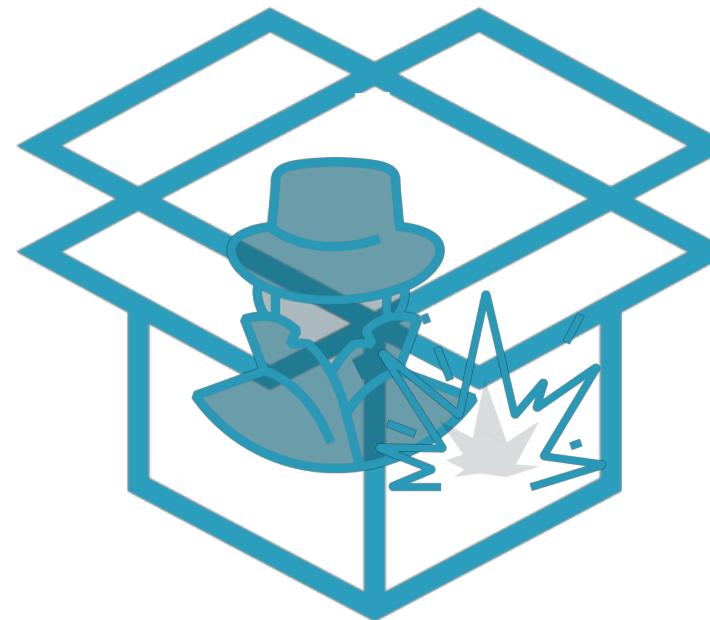


A Mock knows what method calls to expect,
and fails the test if they are incorrect.

Mocks

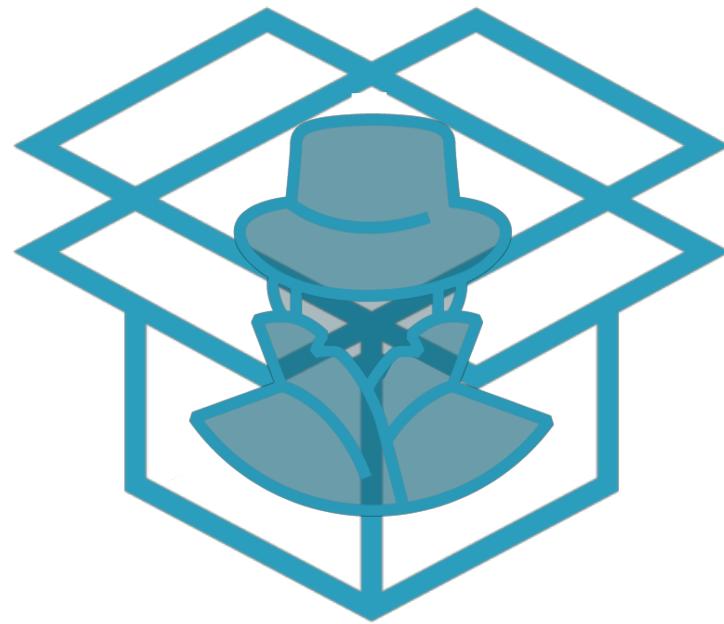


Looks good from the outside

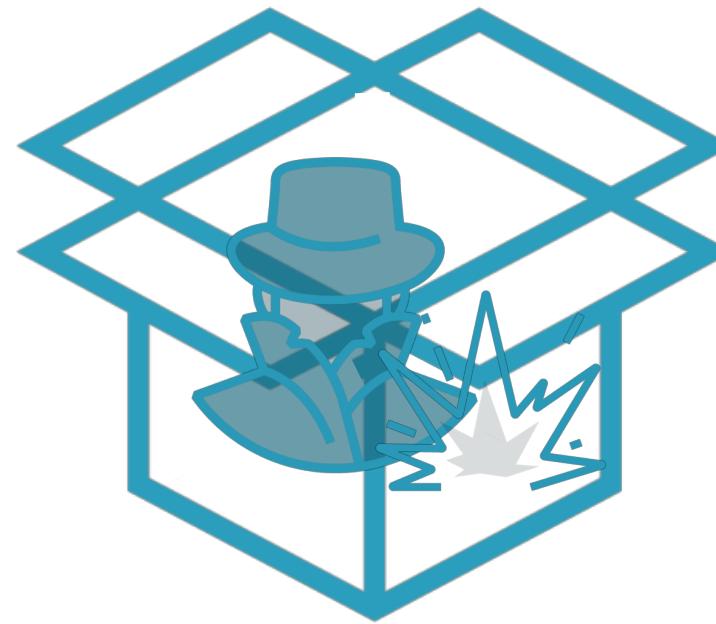


**Expect certain method calls,
otherwise raise an error**

Spies and Mocks



Spy



Mock

Test Doubles Form Two Groups

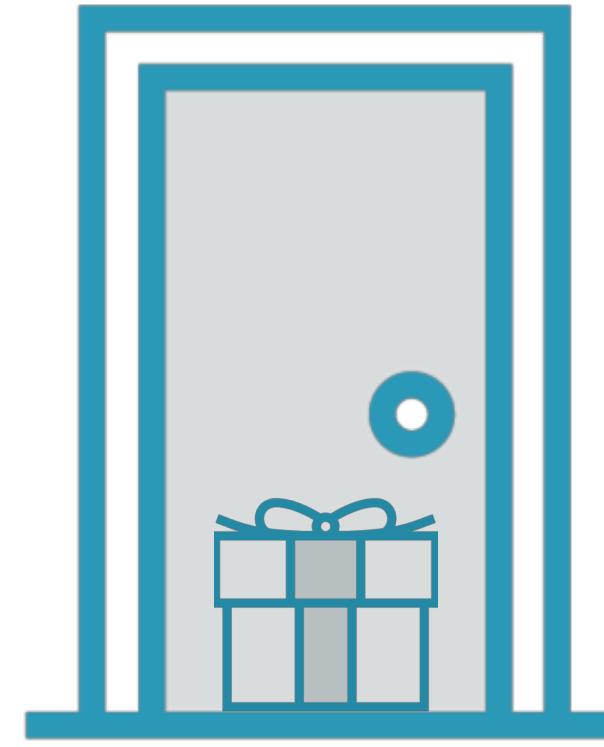
Replace inconvenient collaborators

Dummy
Stub
Fake

Ensure interactions are correct

Spy
Mock

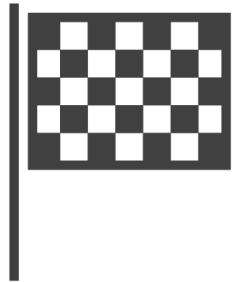
Monkey Patching



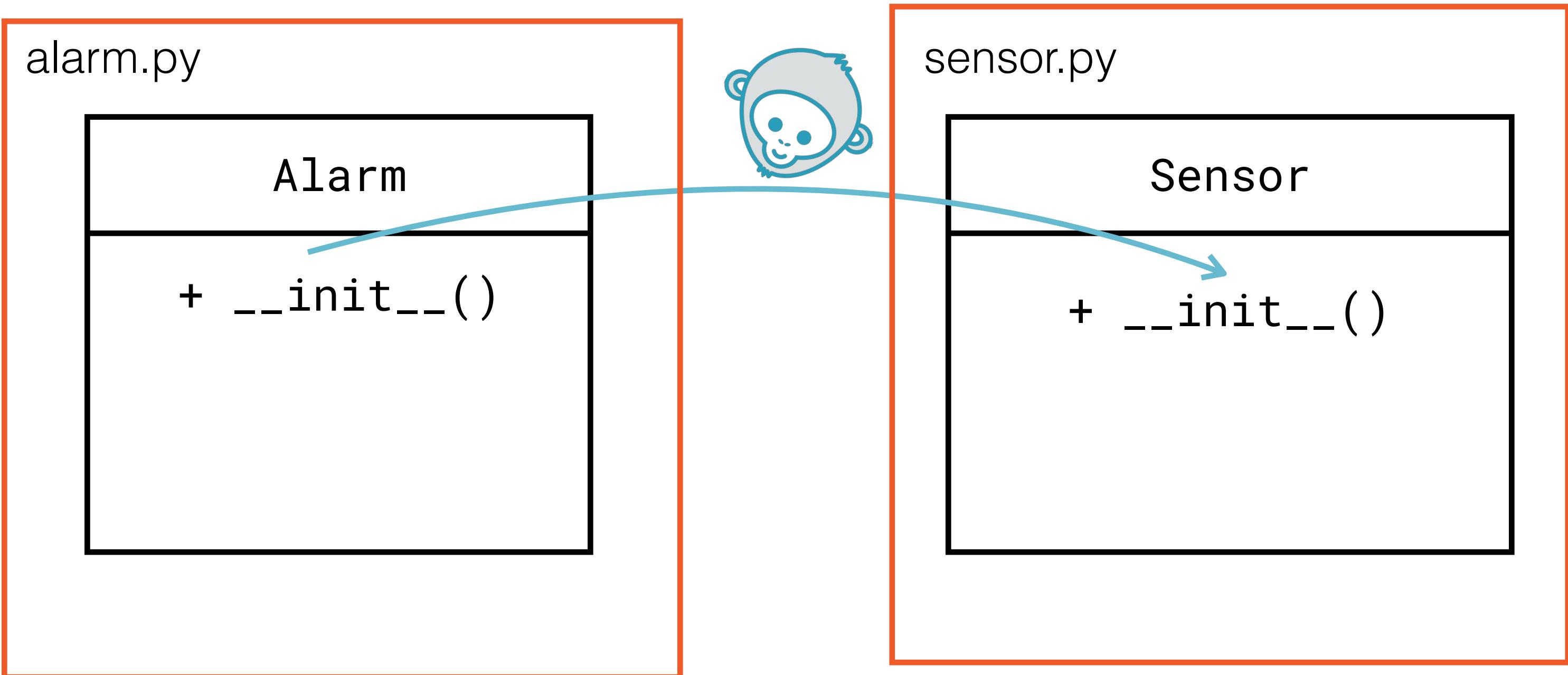
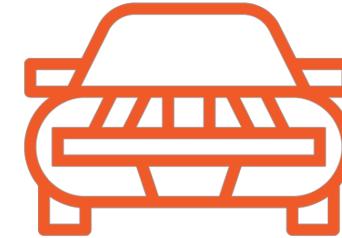
**Collaborator is difficult to
replace**

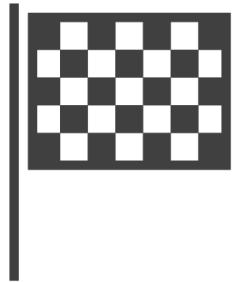


Use a monkey patch

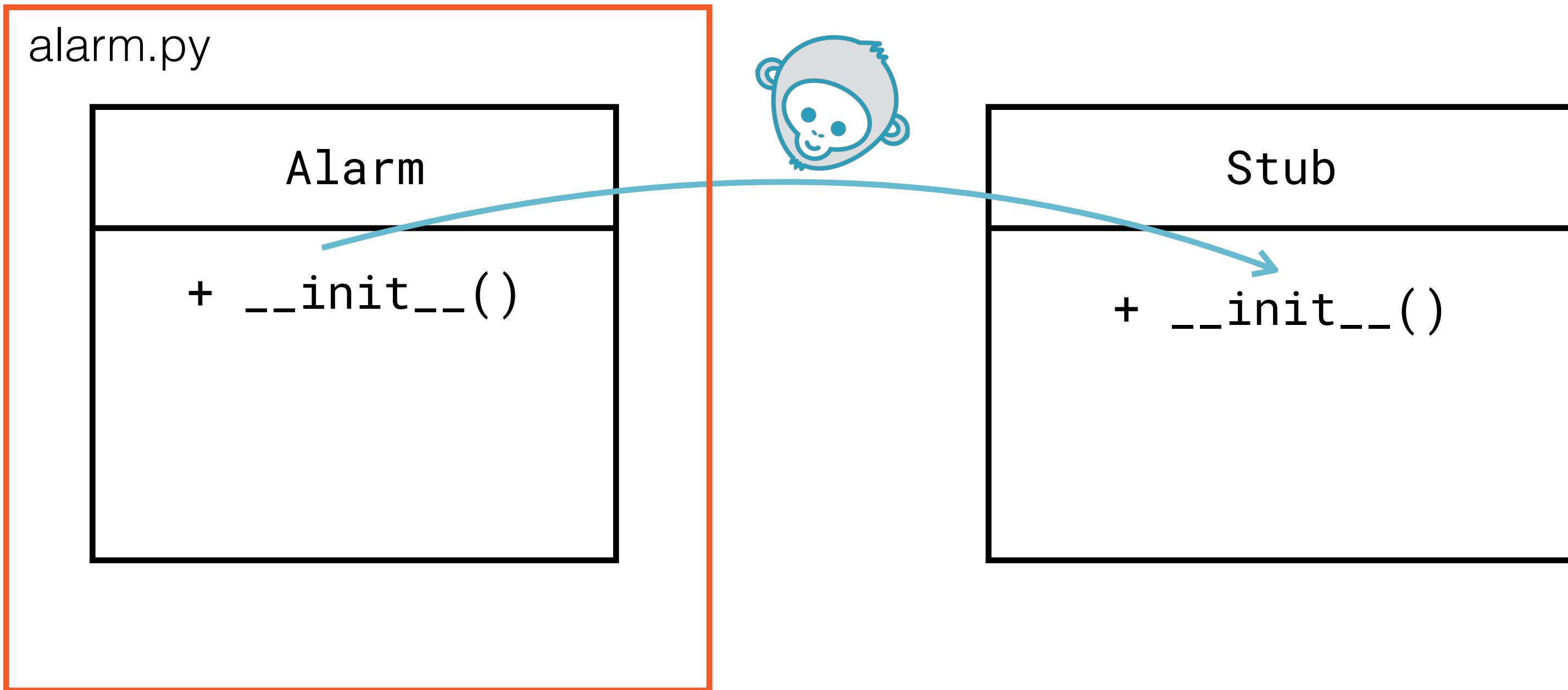
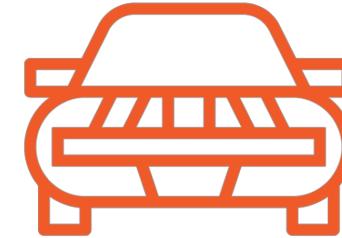


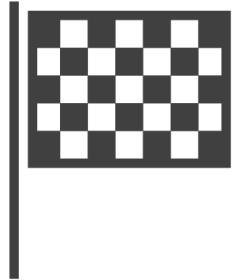
Monkeypatch Example



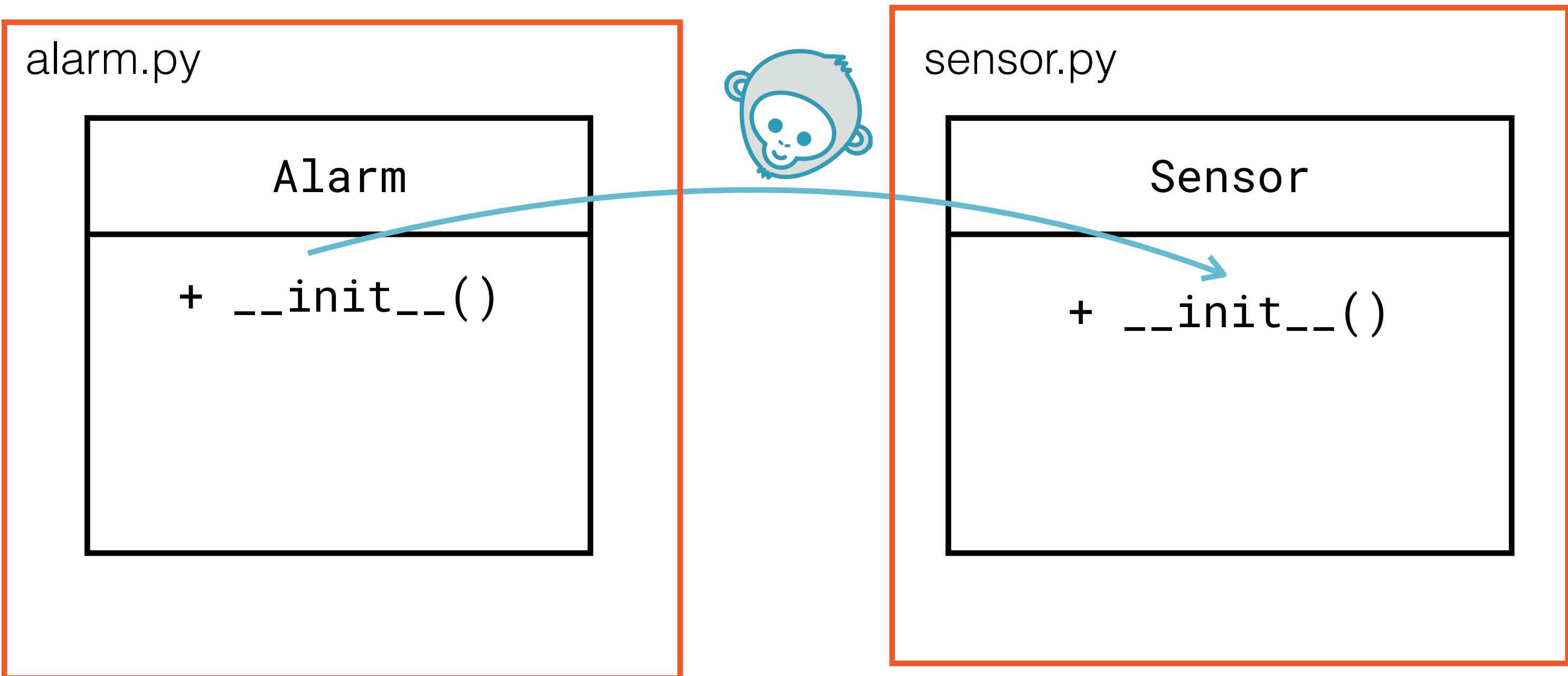
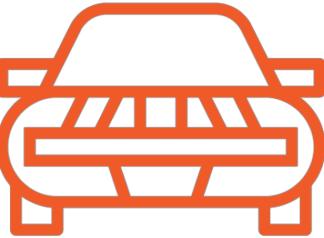


Monkeypatch Example





Monkeypatch As a Last Resort



Summary

What is a Test Double?

Why use a Test Double?

**Inject Test Doubles using
Monkeypatching**