How To Not Write Tests

Golden Masters

shown using ruby

https://github.com/kytrinyx/approvals

```ruby
require 'approvals/rspec'

describe 'Complicated Operation' do
  it 'does something in a certain case' do
    verify do
      a_complicated_operation
    end
  end
end
```

- approvals (ruby)
- no good ones? (python)

but very simple to
write you own

```python
import json

def something_complicated():
    return [1,2,37,42]

with open('something_complicated_approved.json', 'wb') as f:
    json.dump(something_complicated(), f)
```

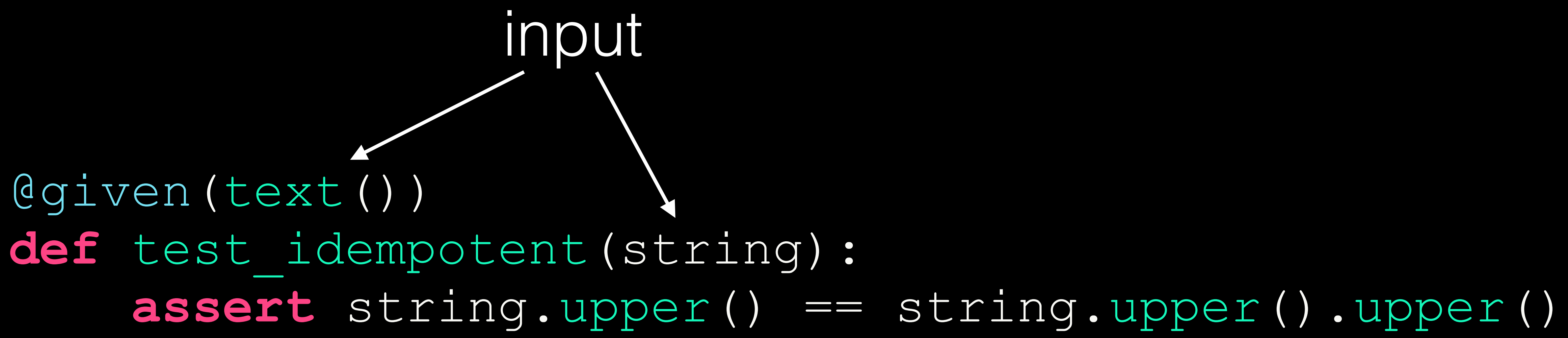
```
$ git add something_complicated_approved.json
$ git commit

$ git status # if it ever changes, you broke something
$ git diff   # see what changed
```

Property Based Testing

shown using python

http://hypothesis.readthedocs.org/

```python
def test_manually():
    assert "foo".upper() == "FOO"
    assert "123".upper() == "123"
    assert "CamelCase!".upper() == "CAMELCASE!"
```



input

```
@given(text())
def test_idempotent(string):
    assert string.upper() == string.upper().upper()
```

- quickcheck (haskell, erlang, c)
- hypothesis (python)
  http://hypothesis.readthedocs.org/
- no good ones (ruby)