







Component: Test Data

Continued

```
shirt = {'color': 'blue', 'size': 'xl'}
```



```
from my_test_framework.data import shirts
def test_add_shirt(self, shirt):
    # navigate to the product page for `shirt`
    # verify that you can find `shirt`
    # add `shirt` to cart
    # verify that `shirt` is in cart
```





# An important detour into parametrization

Parametrization is a test design superpower. If you have series of tests that vary only in the specific test data, for example what should happen with a blue XL t-shirt vs. a red XL t-shirt, then you should parametrize.

Parametrization works roughly like this:

1. Identify a scenario where you are dealing with essential the same test but with different data or action parameters.
2. Generalize the test steps.
3. Build a data model that can be iterated over.

With pytest, for a test method that is parametrized, during test collection a new test method for each parameter is dynamically created.



# Component: Test Data Continued

data/shirts.py

```
shirt = {'color': 'blue', 'size': 'xl'}
```

tests/test\_simple.py

```
from my_test_framework.data import shirts
def test_add_shirt(self, shirt):
    # navigate to the product page for `shirt`
    # verify that you can find `shirt`
    # add `shirt` to cart
    # verify that `shirt` is in cart
```