

COMP
110

Unit Tests

Unit Testing

- Writing functions to test other functions
- Using pytest (other frameworks exist)

Syntax: Writing a unit test

Test file names: end with `_test.py`

Test function names: begin with `test_`

def `test_name()` -> None:

```
    assert <boolean expression>
```

Syntax: Writing a unit test

Test file names: end with `_test.py`

Test function names: begin with `test_`

def `test_name()` -> None:

 # Other code can go here!

`assert <boolean expression>`

Testing For Desired Output

- Checking that your function returns the correct output given a specific input

Example in VSCode...

Testing For Desired Behavior

- Checking that your function does what you want it to do rather than just checking what it returns.
- This can be useful for functions that *mutate* their input.

Example in VSCode...

Types of Tests

Use case: testing properties for how we expect our program to be used

Edge case: testing instances outside “typical” usage (e.g. empty inputs, incorrect inputs, etc.)

Both can test desired output or behavior!

Syntax: Running a unit test in the terminal

Command line: `python -m pytest path/to/testfile.py`