

Modules and unittest

Joachim Vandekerckhove

Winter 2025

Basic Test Structure in Python

- Tests are organized in classes inheriting from `unittest.TestCase`
- Each test method starts with `test_`
- Sometimes it is helpful to add a `setUp()` method

```
class TestCalculator(unittest.TestCase):  
    def setUp(self):  
        self.calc = Calculator()  
  
    def test_add(self):  
        self.assertEqual(self.calc.add(2, 3), 5)  
        self.assertEqual(self.calc.add(-1, 1), 0)
```

Assert Methods

Common assertions in unittest:

- `assertEqual(a, b)` - verify a equals b
- `assertAlmostEqual(a, b)` - for floating point
- `assertRaises(Error)` - verify exceptions
- `assertTrue(x)` - verify x is True
- `assertFalse(x)` - verify x is False

Testing Edge Cases

Example: Testing division by zero

```
def test_divide_by_zero(self):  
    with self.assertRaises(ValueError):  
        self.calc.divide(5, 0)
```

- Test both normal and edge cases
- Test invalid inputs
- Test boundary conditions

Testing Complex Behavior

Example: Testing operation history

```
def test_history(self):  
    self.calc.add(2, 3)  
    self.calc.multiply(4, 5)  
    history = self.calc.get_history()  
    self.assertEqual(len(history), 2)
```

- Test state changes
- Test sequences of operations
- Verify side effects

Best Practices

- Write tests before or while writing code (TDD)
- One test method per feature/scenario
- Clear, descriptive test names
- Independent tests (no dependencies between tests)
- Use `setUp()` for common initialization
- Keep tests simple and focused

Running Tests

- Run all tests in the project:

```
python -m unittest discover -s tests
```

- Run a specific test file:

```
python -m unittest tests.test_operations
```

- Run a specific test case:

```
python -m unittest tests.test_operations.TestClass.test_method
```

Project Structure for Tests

Key requirements:

- Tests should be in a `tests` directory
- Include empty `__init__.py` in `tests` directory
- Test files should follow naming pattern `test_*.py`
- Run tests from project root directory

Modules and unittest

Joachim Vandekerckhove

Winter 2025