Test-Driven Development (TDD)

Web Development Boot Camp
Unit 10

What is test-driven development?

Test-Driven Development

"Test-Driven Development (TDD) is a software development process relying on software requirements being converted to test cases before software is fully developed, and tracking all software development by repeatedly testing the software against all test cases."

([From Wikipedia, Test-driven Development](#))

TDD

01   Write FAILING test(s) BEFORE writing code

02   Write code to PASS tests

03   Refactor code

04   Use tests throughout development to avoid regression

Types of Tests

Tests can be categorized into three types of tests:

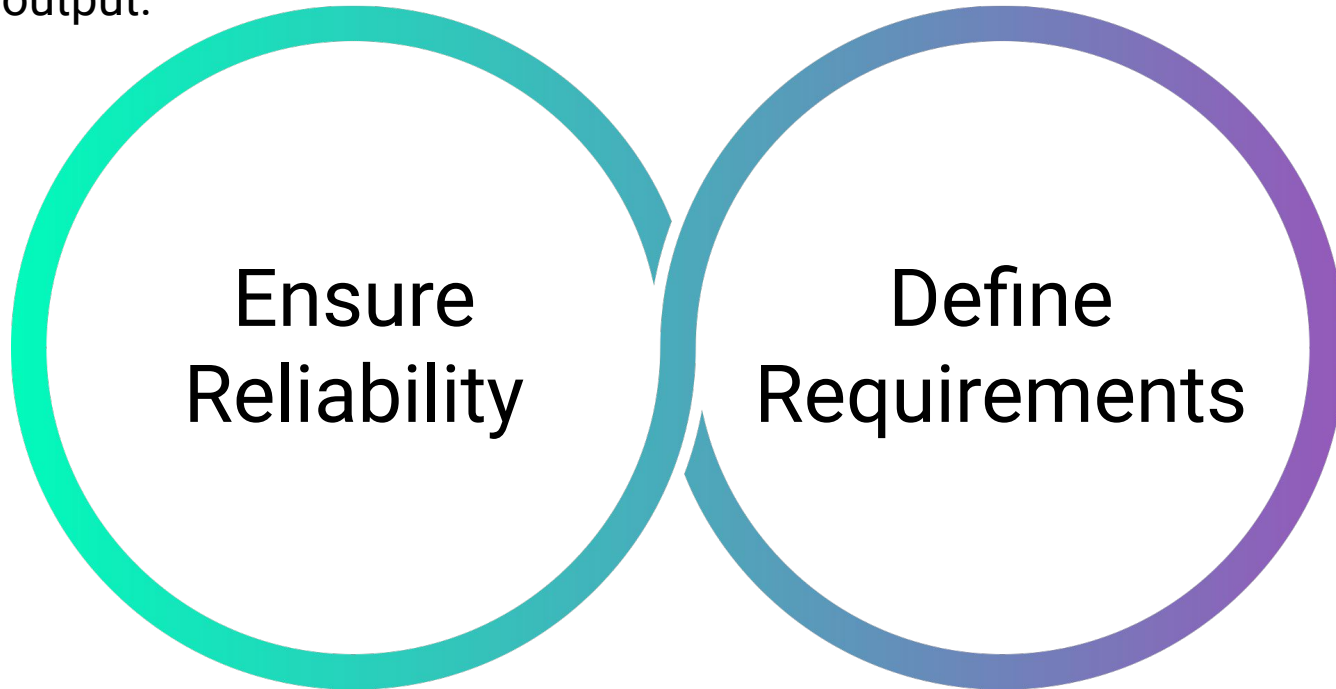**U**nit tests

Integration tests

End-to-end tests

What are unit tests?

Unit Tests

Unit tests ensure reliability and validate the expected behavior of things like functions or classes. Unit tests verify that the unit being tested returns the expected output.

Ensure Reliability

Define Requirements

What are integration tests?

Integration Tests

Integration tests ensure the cooperation of units in your application.

Integration tests might focus on an API, user input, or a user interface, which could involve subsequent actions like writing to a database or logging output.

What are end-to-end tests?

End-to-end Tests

End-to-end tests verify that the application will function as expected from the perspective of a user. These tests usually focus almost entirely on an application's user interface.

Tests Before Code

Writing tests before writing code is the core tenet of TDD.

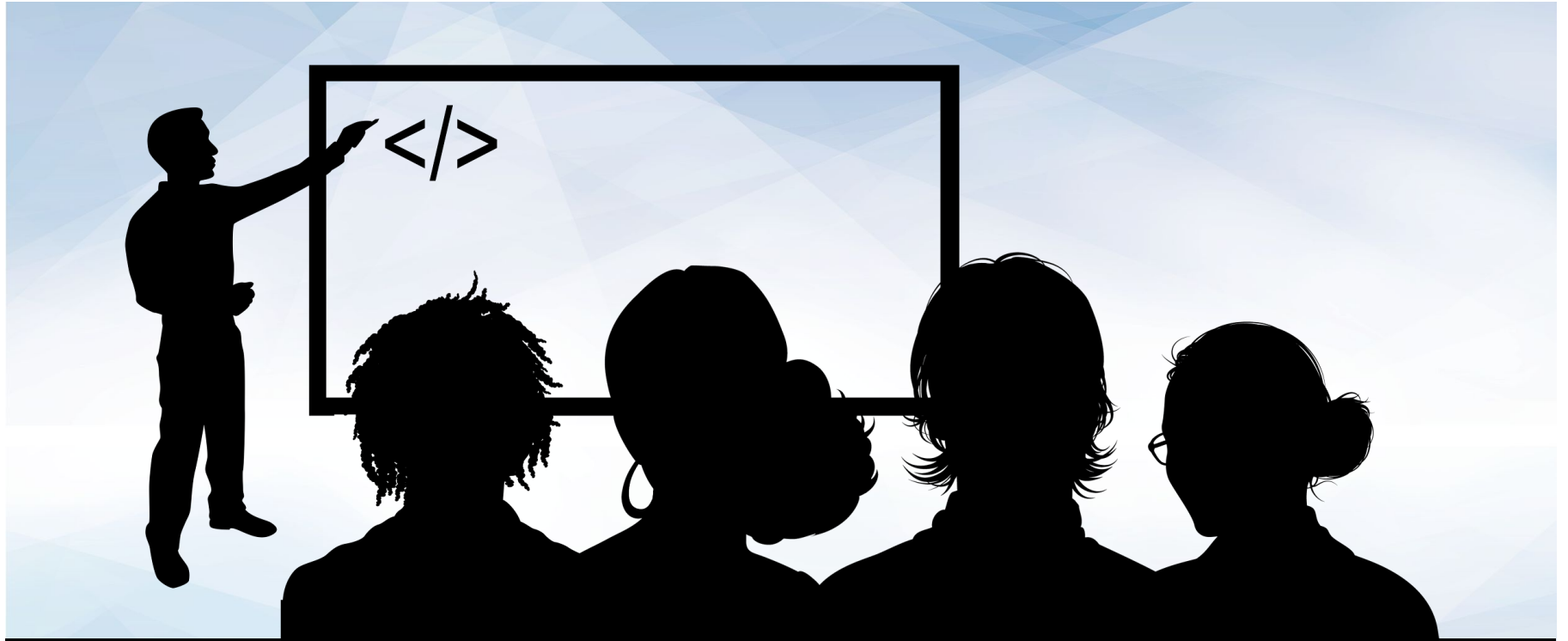The code we write is dictated by the need to make tests pass.

Avoid writing code without having a failing test to pass.

TDD Workflow

Write a failing test

Make the test pass

Refactor the code

Instructor Demonstration
TDD