

CMPUT 301 GH Actions and TDD

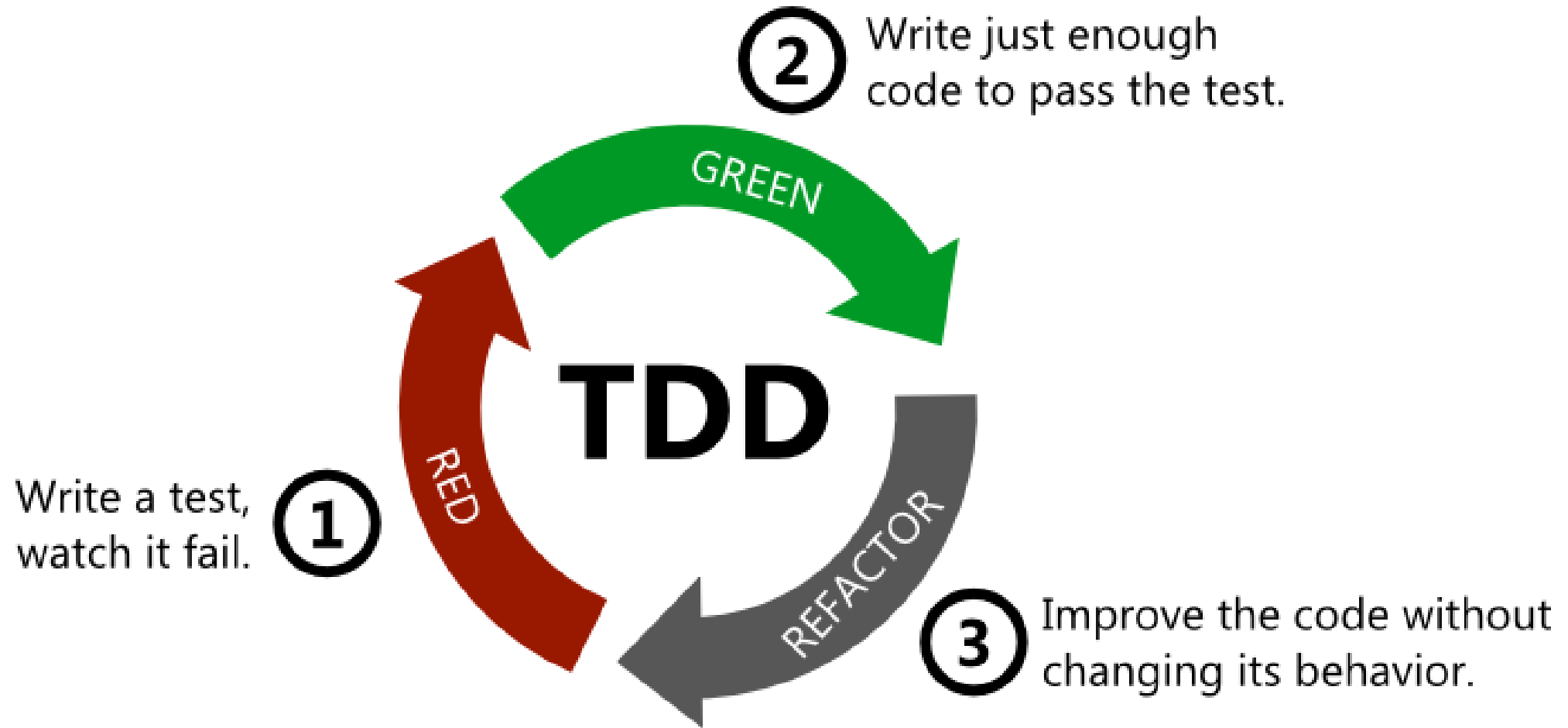
TDD Overview

Software development technique where you write tests **before** the functionality

I.e.,

1. Write a test (which will obviously fail, since there is no functionality)
2. Write the functionality (test should pass)
3. Refactor

TDD Cycle



Why bother with TDD?

TDD forces you to think first about what you are trying to do, what the input and output should be, before trying to program anything.

It also forces you to start small, and build up the functionality, rather than code a large block of code, which is likely to be difficult to test.

But I don't know what I am going to program

This means the requirements are not fully fleshed out. Think about what you need, and maybe even talk with the client for clarification.

GH Actions

The screenshot displays a GitHub Actions workflow run in progress. The background shows a terminal window with the following output:

```
✓ should respond with 404 json
✓ should 404 with unknown user

when requesting an invalid route
✓ should respond with 404 json

1123 passing (4s)

=====
Writing coverage object [/home/runner/build/express-coverage-1123-coverage.json]
Writing coverage reports at [/home/runner/build/express-coverage-1123-coverage.json]
=====

===== Coverage summary =====
Statements : 98.81% ( 1916/1939 ), 38 ignored
Branches   : 94.58% ( 751/794 ), 22 ignored
Functions  : 100% ( 267/267 )
Lines      : 100% ( 1872/1872 )
=====

The command "npm run test-ci" exited with 0.

$ npm run lint

> express@4.17.1 lint /home/runner/build/express
> eslint .

The command "npm run lint" exited with 0.

store build cache

$ # Upload coverage to coveralls

Done. Your build exited with 0.
```

Overlaid on the terminal is a summary box with the following content:

- All checks have passed** (4 successful checks) [Hide all checks](#)
- ✓ **build** Successfully in 59s — build
- ✓ **test** Successfully in 59s — build
- ✓ **publish** Successfully in 59s — build
- ✓ **This branch has no conflicts with the base branch**
Merging can be performed automatically.

At the bottom of the summary box, there is a green button labeled **Merge pull request** and a link: [You can also open this in GitHub Desktop or view command line instructions.](#)

<https://docs.github.com/en/actions/learn-github-actions/understanding-github-actions>