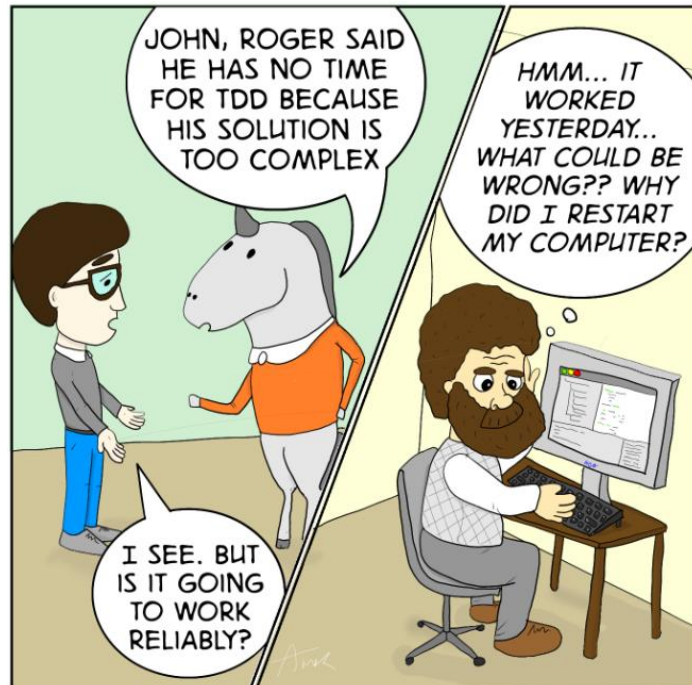


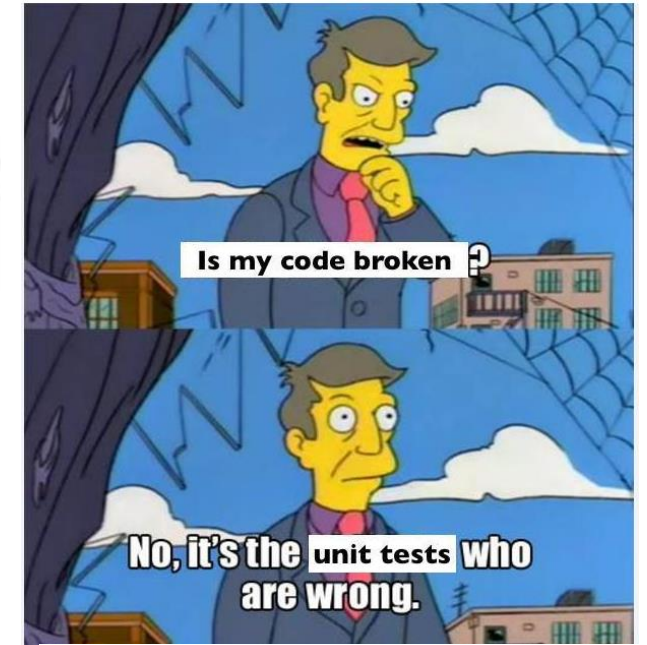
TDD – Test Driven Development

Presentation by : Indumathi Chinnaswamy

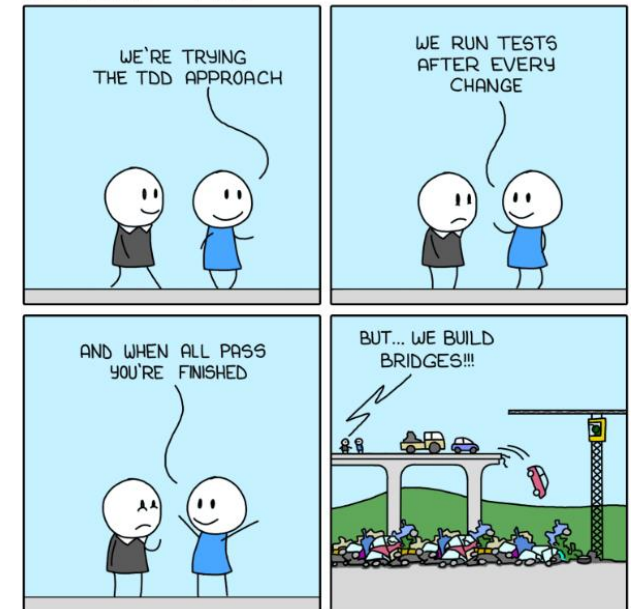
Date: 25.11.2022



DEVELOPMENT DRIVEN TESTS



APPLIED TDD



Why testing is important?

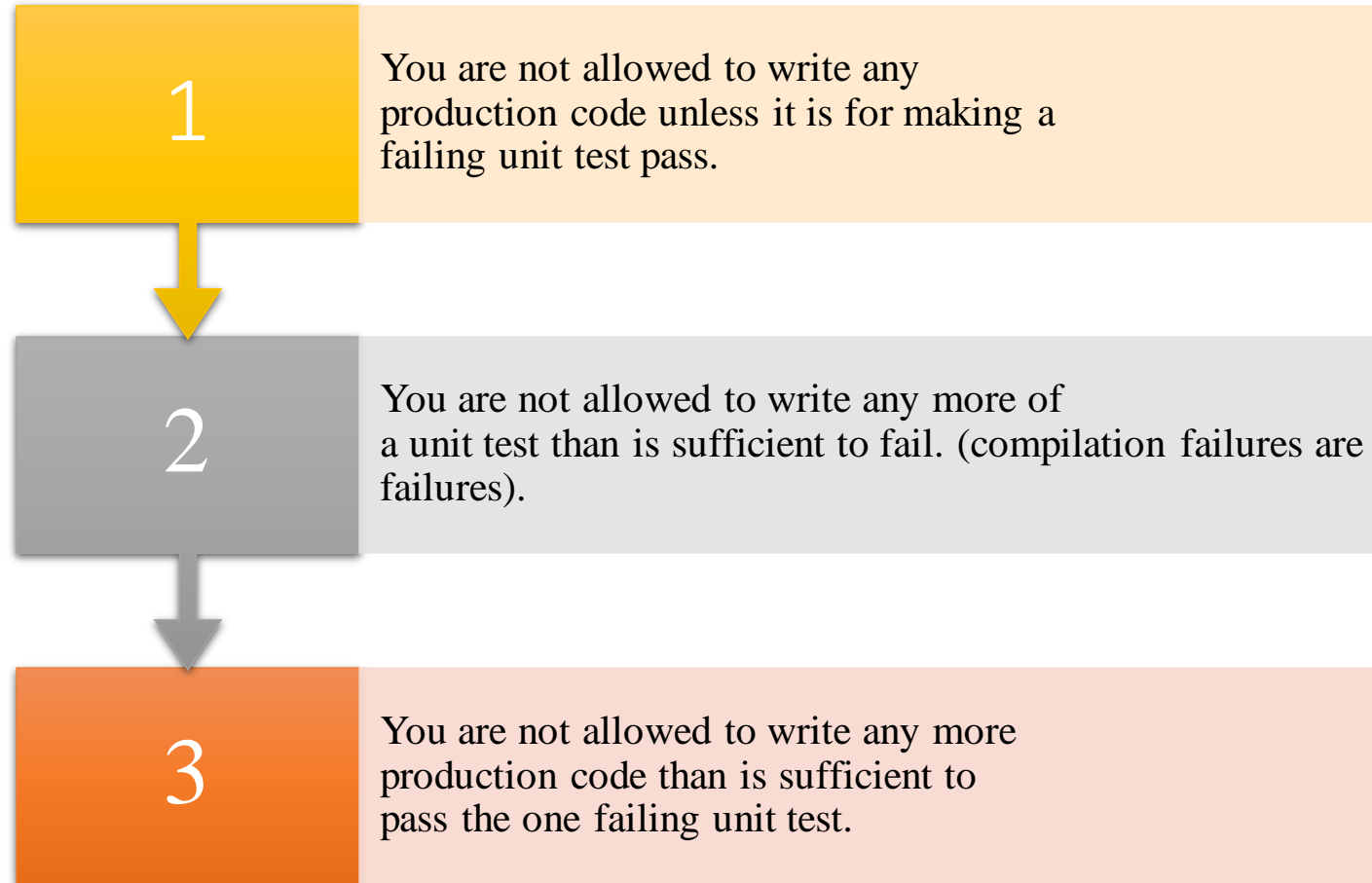
"If you don't have tests, how do you know your code is **doing the thing right** and **doing the right thing**?"

What is TDD?

- Before you write code, think about what it will do. Write a test that will use the method you haven't even written yet.
- A test is not something you "do", it is something you "write" and run once, twice, thrice and more.
- TDD is a technique whereby you write your test cases before you write any implementation code.

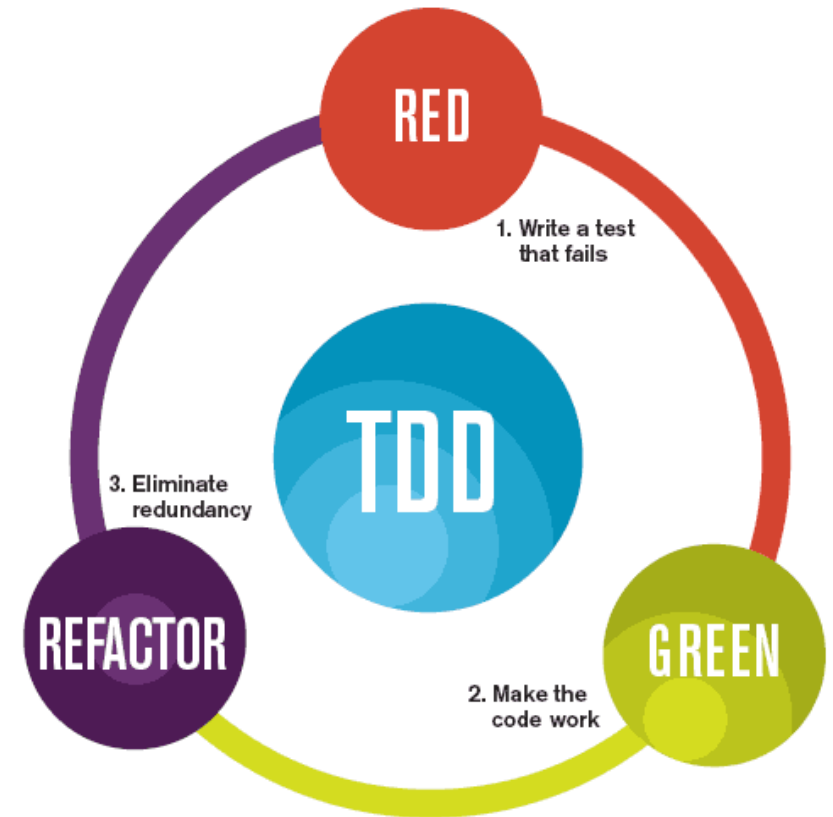
The Three Rules of TDD

Robert C. Martin (“Uncle Bob”) provides a concise set of rules for practicing TDD.



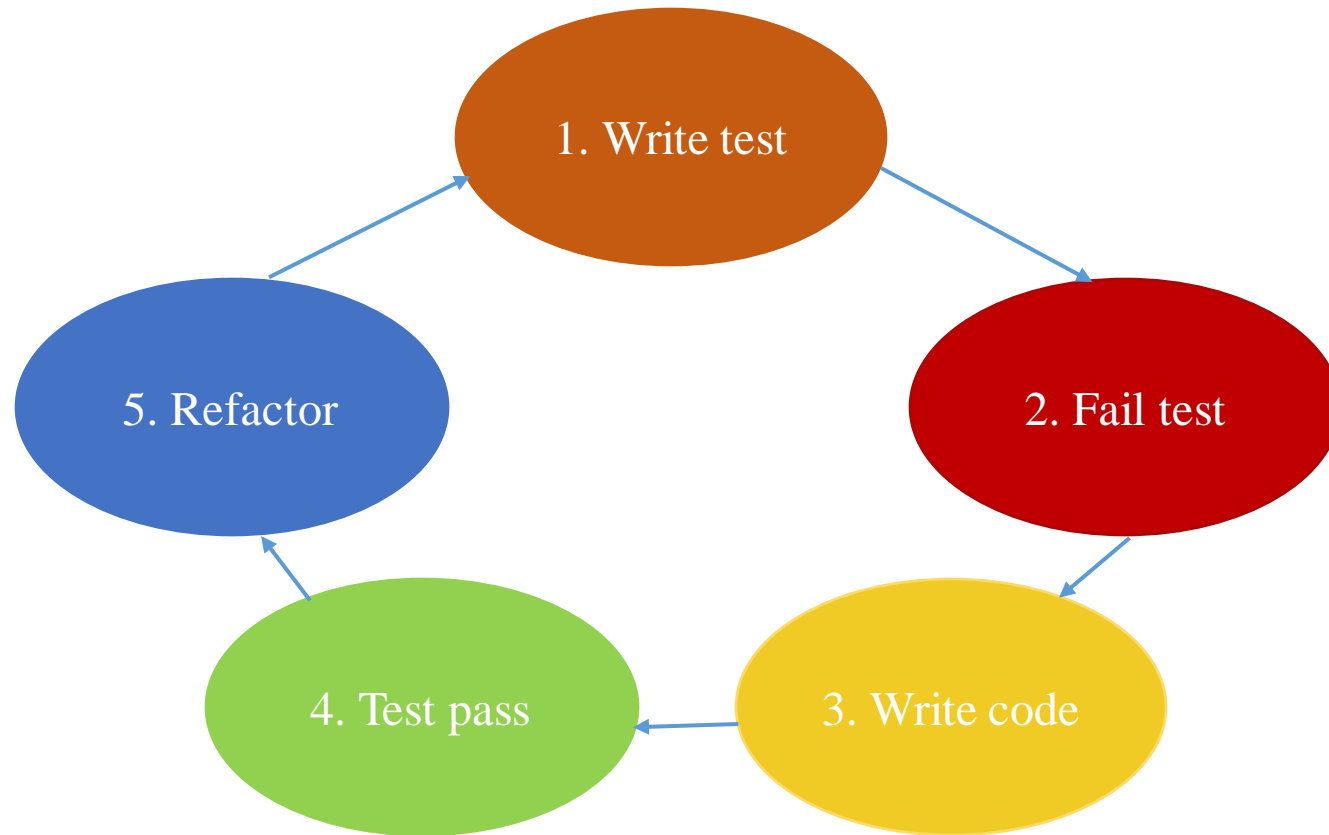
The TDD way

- Do the simple thing
 - Write a test that fails (red)
 - Make the test pass (green)
 - Refactor implementation code
- Use the compiler – let it tell you about the errors
- One assertion per test



The mantra of Test-Driven Development (TDD) is "red, green, refactor."

Stages of TDD



Benefits of TDD

- Refactor constantly
- Easy maintenance code
- Design is divided into phases
- Clear documentation of code
- Promotes continuous change

Why TDD?

- Early bug detection
- Better designed, cleaner and more extensible code
- Avoids duplication of code
- Good for teamwork
- Good for developers
- Confidence to refactor

Kent Beck Quotes

If you're having trouble succeeding, fail.

I'm not a great programmer; I'm just a good programmer with great habits.

Make it work, make it right, make it fast.

Issues with TDD

- Tests are additional code that must be maintained with the code.
- More test code than production code.
- User interface can be tested, but changes often.
- If the requirement is not clear or changes often.
- TDD is always underestimated and always considered as a matter of slowing down the deliverables.
- The test are only good as the developer who wrote them. (Developer should have the right knowledge about the test/ behavior)
- Legacy and complex code
- All the developers in the team must follow.

Thank you!

