



BITS Pilani

Pilani | Dubai | Goa | Hyderabad

Object Oriented Analysis & Design Module-1 (RL 1.3.5)

Sanjay Joshi



Test Driven Development

Test Driven Development



- TDD is a technique whereby you write your test cases **before** you write any implementation code
- Tests drive or dictate the code that is developed
- An indication of “intent”

Tests provide a specification of “what” a piece of code actually does

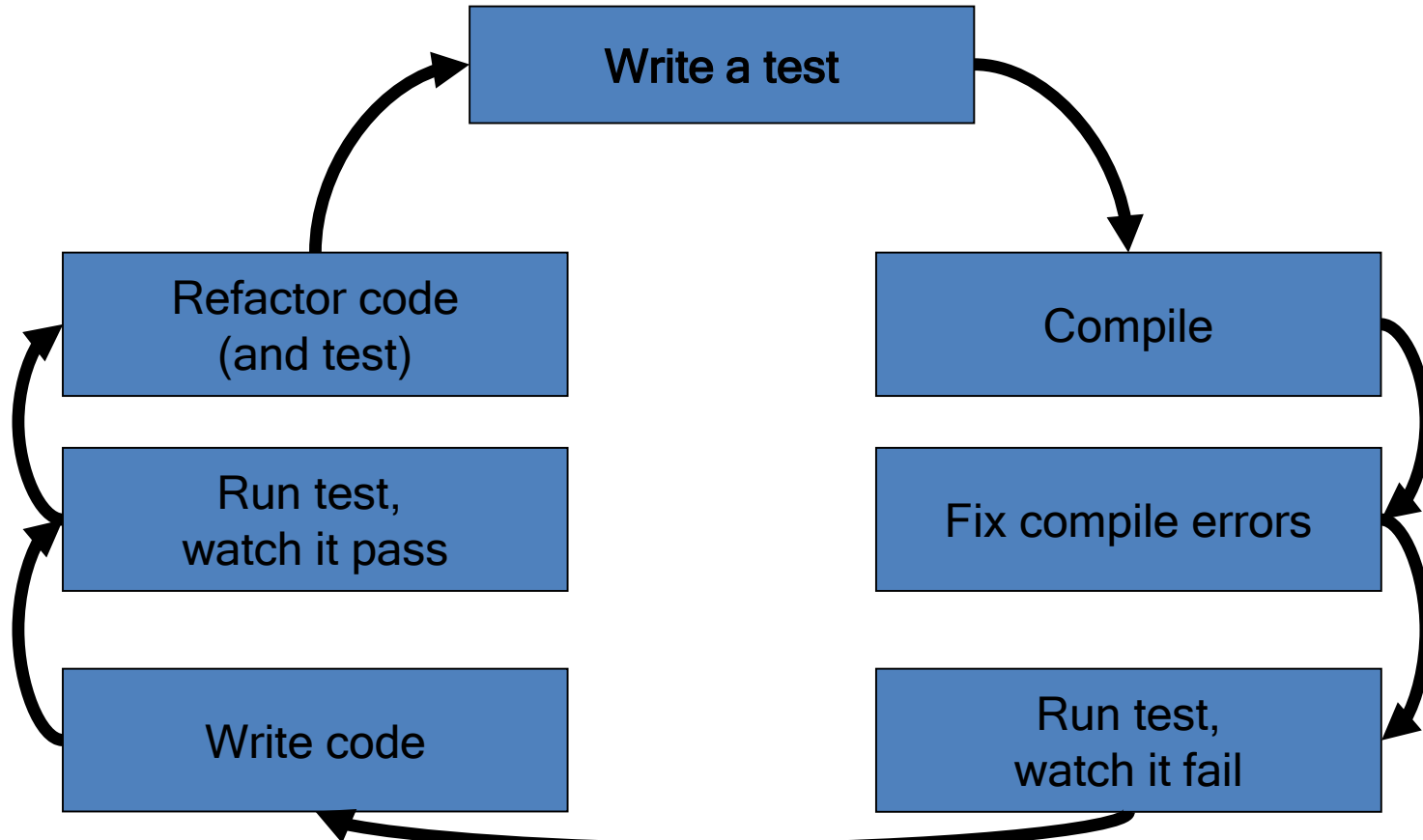
- Some might argue that “tests are part of the documentation”

Why is TDD



- TDD can lead to more modularized, flexible, and extensible code
- Clean code
- Leads to better design
- Better code documentation
- More productive
- Good design

Introduction to TDD



Introduction to TDD



- In Extreme Programming Explained (The Green Book), Bill Wake describes the test / code cycle:
 1. Write a single test
 2. Compile it. It shouldn't compile because you've not written the implementation code
 3. Implement **just enough** code to get the test to compile
 4. Run the test and see it **fail**
 5. Implement **just enough** code to get the test to pass
 6. Run the test and see it **pass**
 7. **Refactor** for clarity
 8. Repeat

Introduction to TDD

- Example
- Given a string swap the last two characters
- Sample input & output set is

Blank String \Rightarrow Blank String

A \Rightarrow A

AB \Rightarrow BA

ABCD \Rightarrow ABDC