

Refactoring

Reto Lehre



03.09.2020



Content

- Introduction
- Refactor Cycle
- Parallel Change
- Refactor Example
- Review

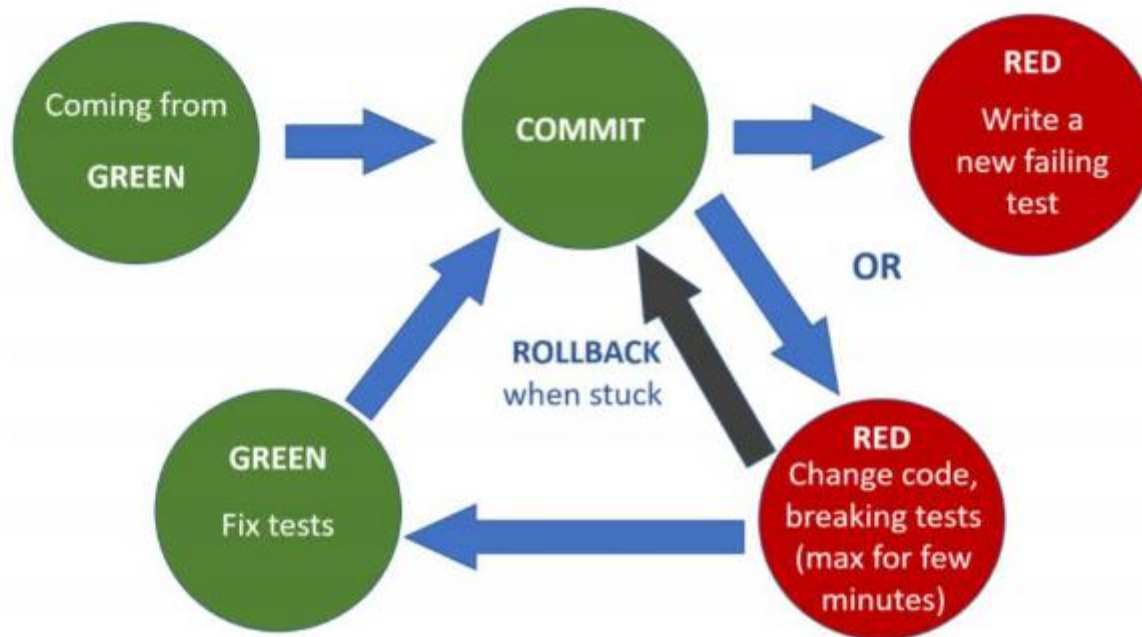
Content

- Introduction
- Refactor Cycle
- Parallel Change
- Refactor Example
- Review

Introduction

- Refactoring is the process of restructuring existing code without changing its external behavior.
- Refactoring is about making the code more readable, simpler, cleaner and turn it into a better architecture.
- 80% of the value in refactoring comes from improving readability. 20% of the remaining value comes from design change.
- Begin in green and stay in green while refactoring.

Refactor Cycle




Extended TDD cycle: the Refactor cycle

Parallel Change

- Refactor technique to implement breaking changes savely.
expand -> migrate -> contract
- Expand
Introduce new functionality by adding new code. Add a new test.
- Migrate
Clients have to migrate to the new code.
- Contract
Once all clients use the new code the old one can be deleted.

Refactor Example

```
public void processOrder(int productId, String productId, String productId)
```



-> Input variable controls the code. Split the method in two separate methods.

```
public void processOrder(int productId, String productId) {  
    public void processOrder(int productId, String productId) {
```

```

var (IPAddresses ip* : address) {

    Address* result;

    //Google for Berlin's best places for vacation in Germany
    result = api.getAddressList(ip*, 1, 10);

    //Google for the best places for vacation in Germany for each ip
    if (result == null) {

        logger.info("Google API for the address list for Berlin: " + ip*, ip*);
    }

    result = api.getAddressList(ip*);
}

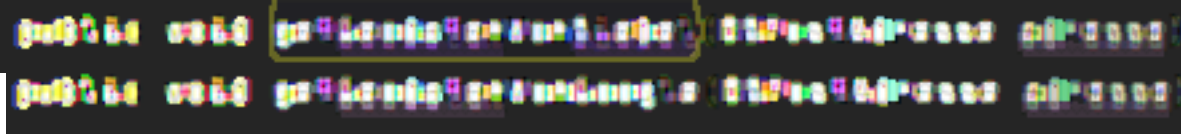
```

-> Too many commands. Method length about 100 Lines. Code is really hard to read.

How can you start refactoring, after understanding the code?

-> extract methods

The code does just do two main things.



Go on with the 'Object Calistenics' Rules and change the architecture of the code.

Review

- Most of the people are scared of refactoring.
- Even small refactorings turn into a big win.
- Master your IDE -> Shortcuts.
- Object calisthenics is a good guide for refactorings.
- How to handle with really bad code.

List of references

- https://en.wikipedia.org/wiki/Code_refactoring
- Alcor Academy

Refactoring



?

