



testdriving

R

Fail

G

Pass

B

Refactor

THE *failing* TEST



Describe

what you want the code to do



Given
initial state

When
something happens

Then
I expect a specified result



Start
with
acceptance
test

And go **top** **down**



do
NOT
overthink it

can't do CODE?



-

ScoreCalculation.feature

Feature: Score Calculation

In order to know my performance

As a player

I want the system to calculate my total score

Scenario: Gutter Game

Given a new bowling game

When all of my balls are landing in the gutter

Then my total score should be 0

Scenario: All Strikes

Given a new bowling game

When all of my rolls are strikes

Then my total score should be 300

use
natural
language

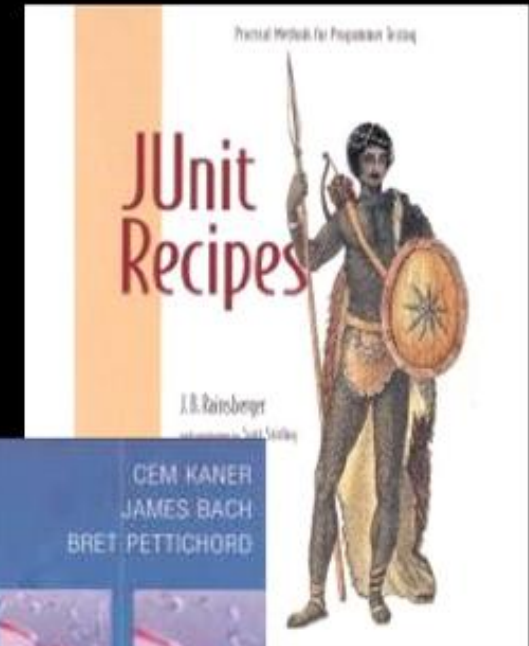
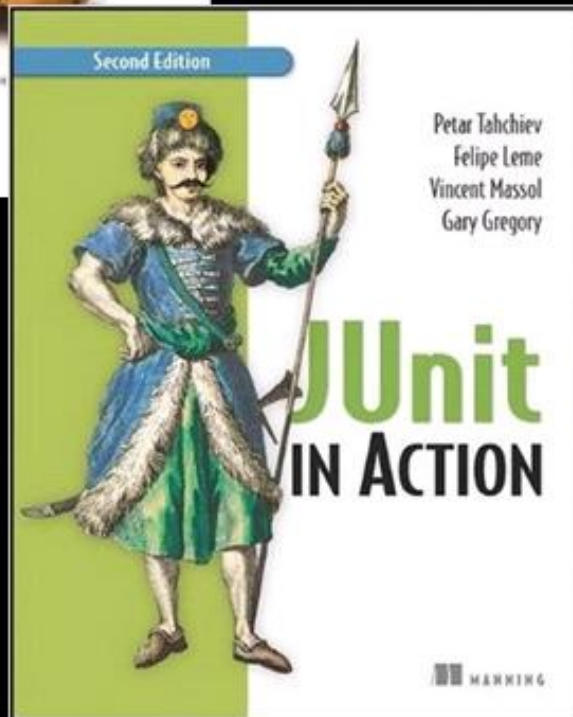
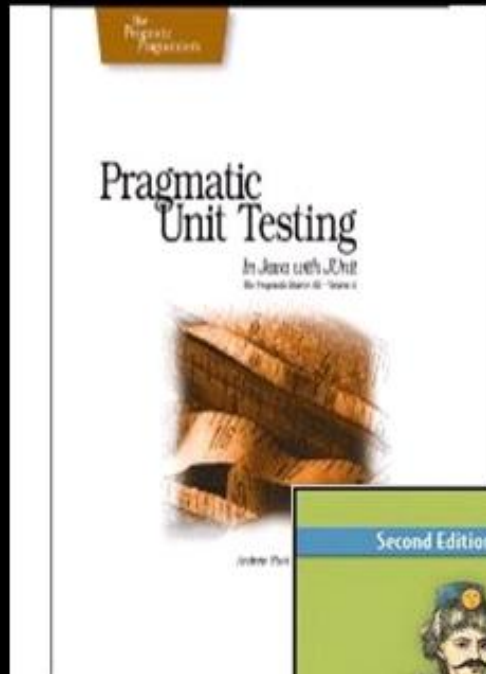
make

the message

meaningful



Further resources



Do not

even

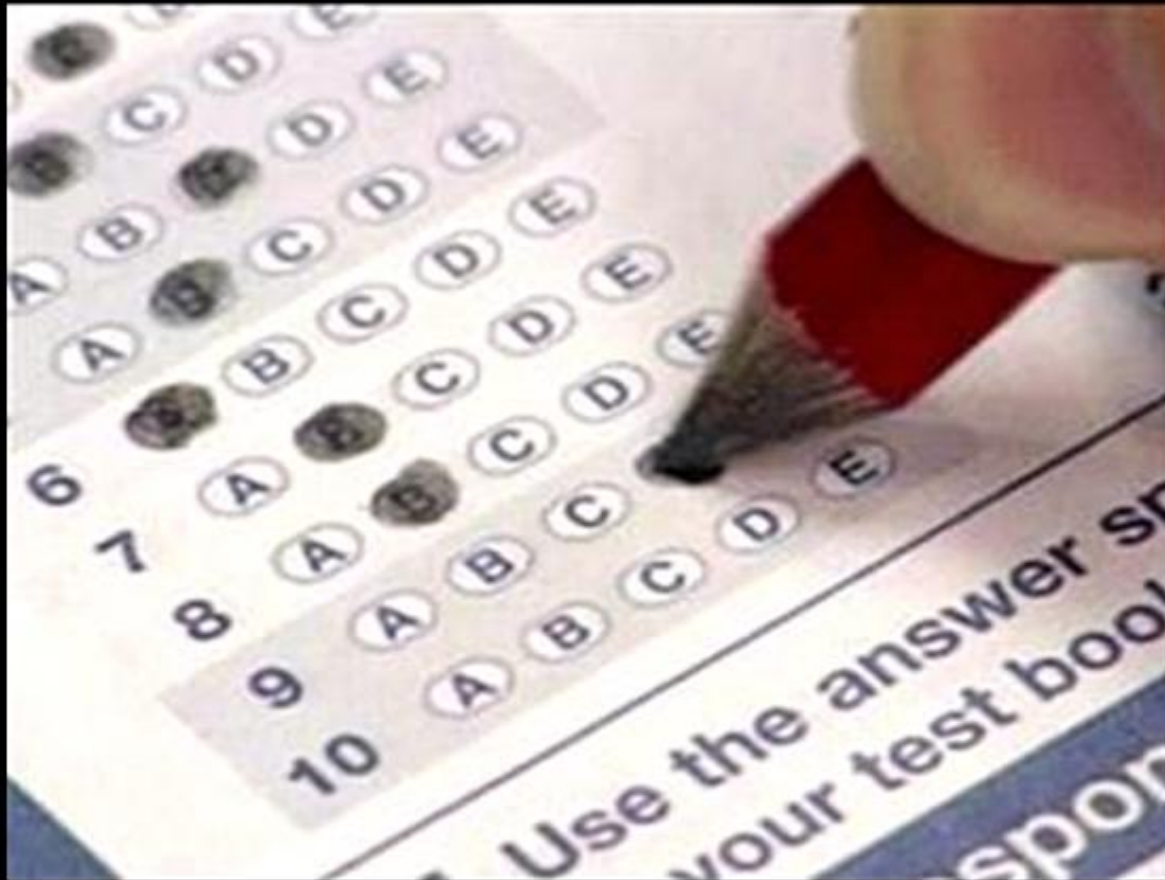
think

about

refactoring...

yet

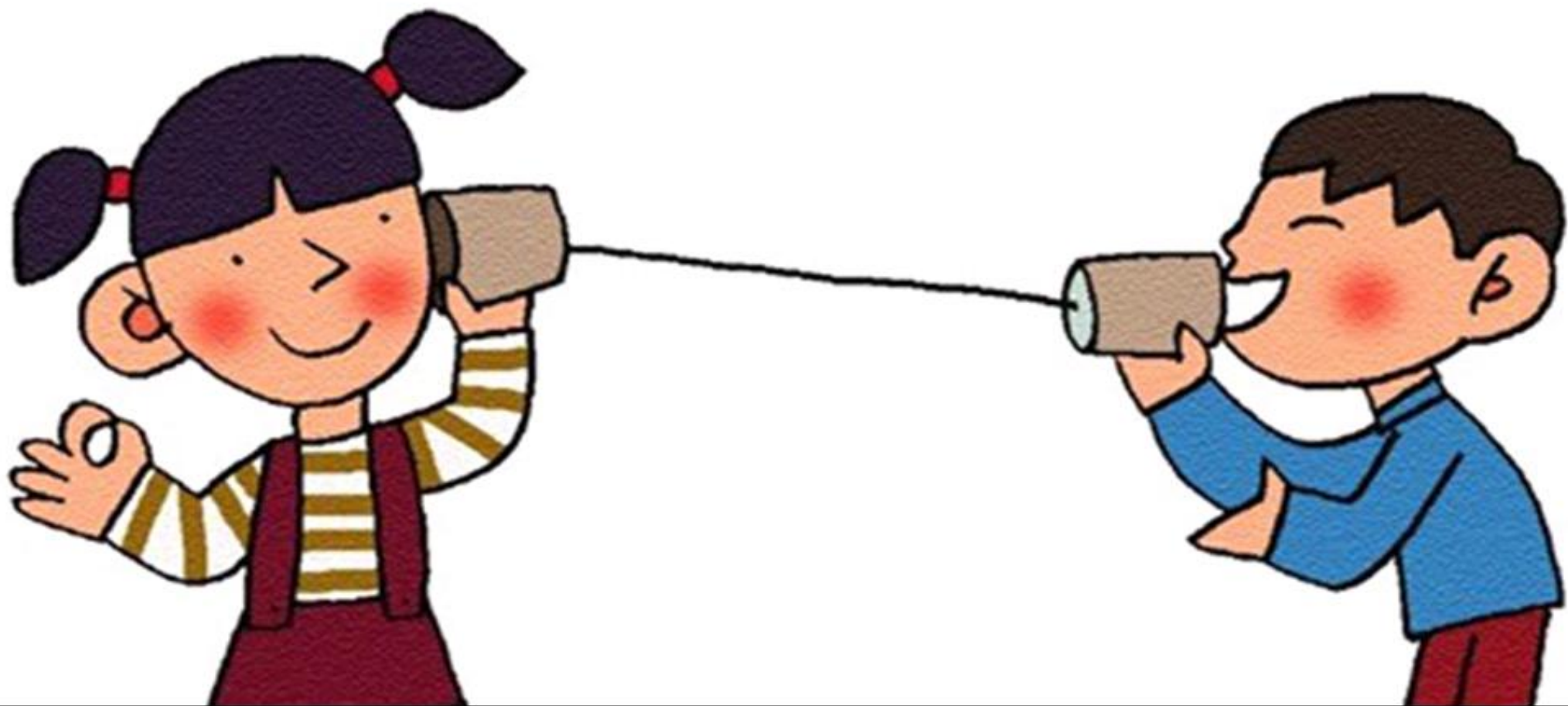




Passing test



Try
the
simplest
possible
solution



But don't try
to cheat yourself

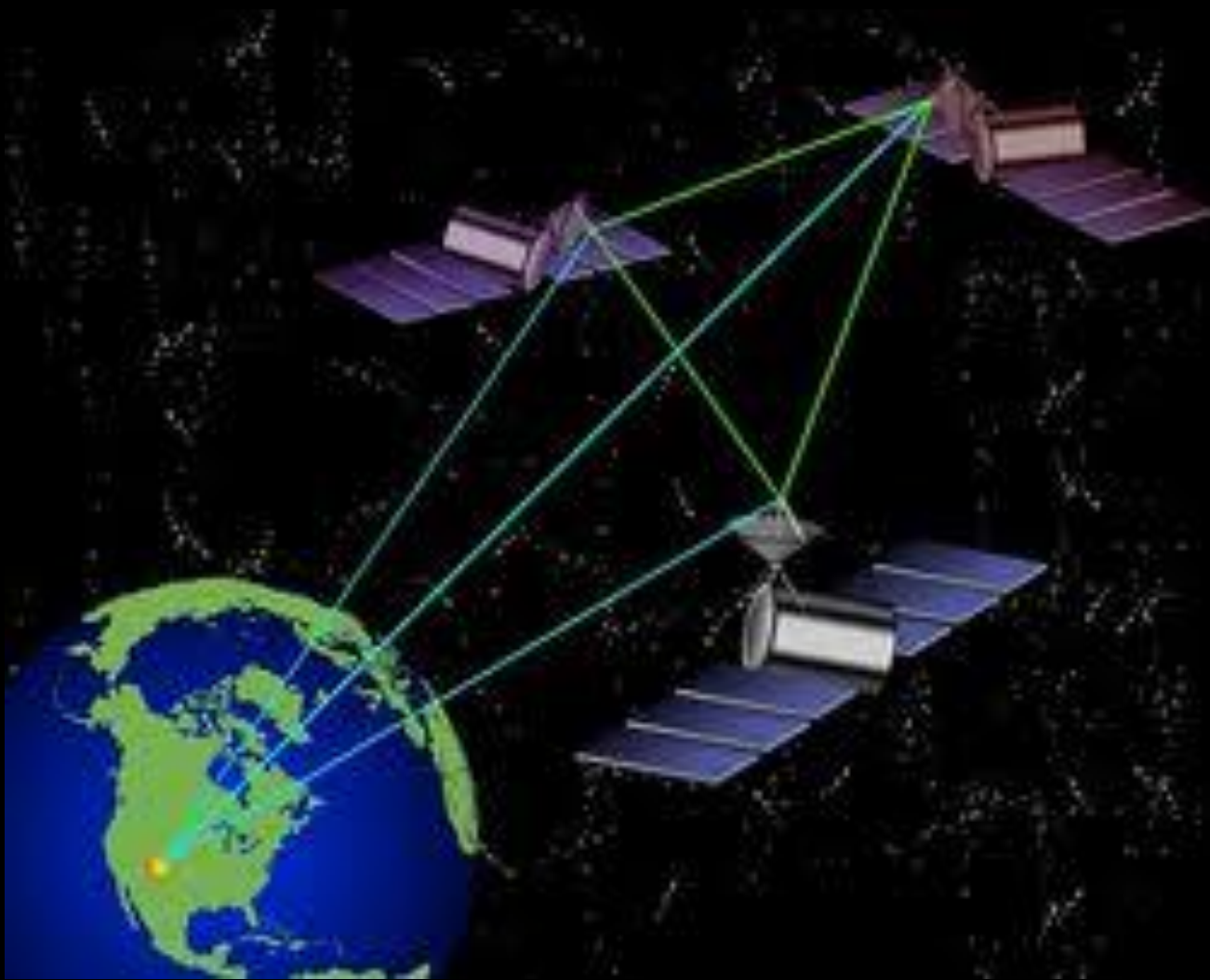
```
public int inc(int n) {  
    return n + 1;  
}
```

Most
of the time
use
**obvious
implementation**


```
public boolean hasReactorLeaked() {  
    return false;  
}
```

Fake it if you really
can't do **obvious**
implementation.

Triangulate



Learn patterns for simplification



After some time
it'll come **naturally**

And
again...

Do not

even

think

about

refactoring...

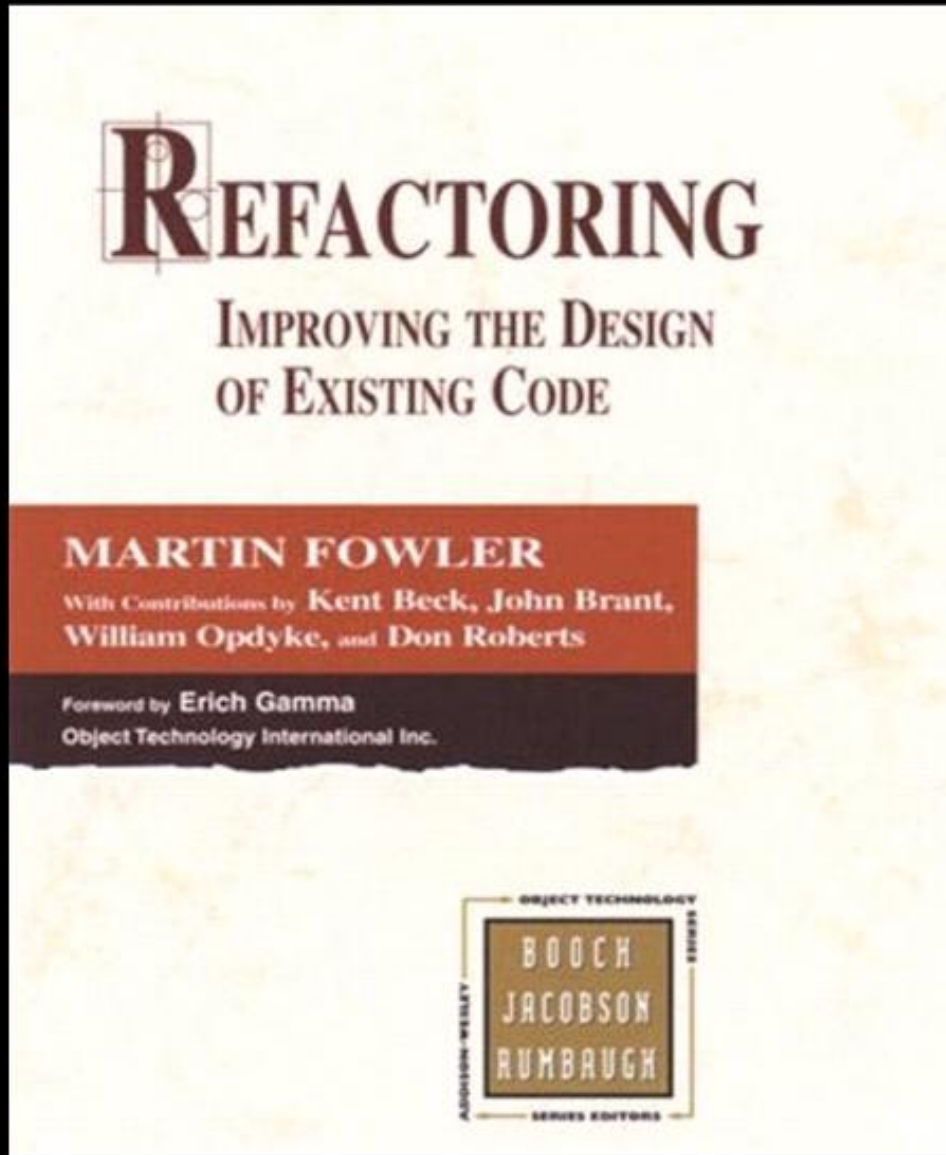
yet



A wide-angle photograph of a deep blue ocean with gentle ripples. The horizon is a straight line in the distance, with a pale, hazy sky above it. The word "Refactor" is centered in the middle of the image in a white, sans-serif font.

Refactor

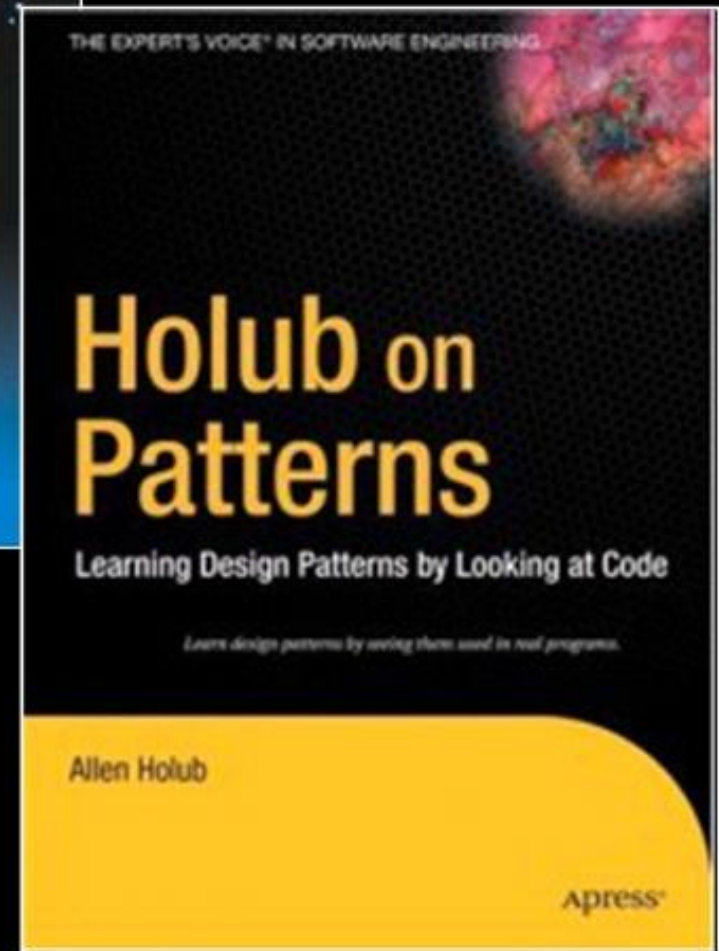
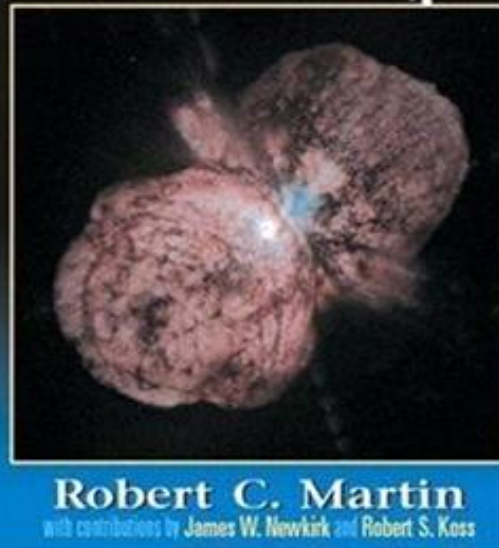
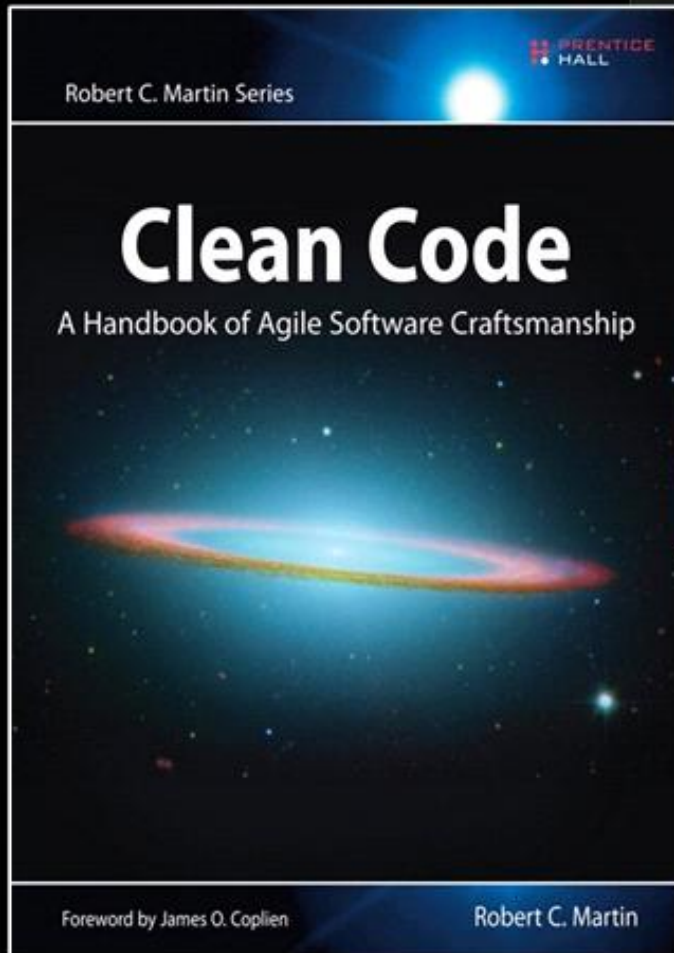
Made it work?



Now
make it
right



Don't try to create
a work of art for a connoisseur



but do follow good practices



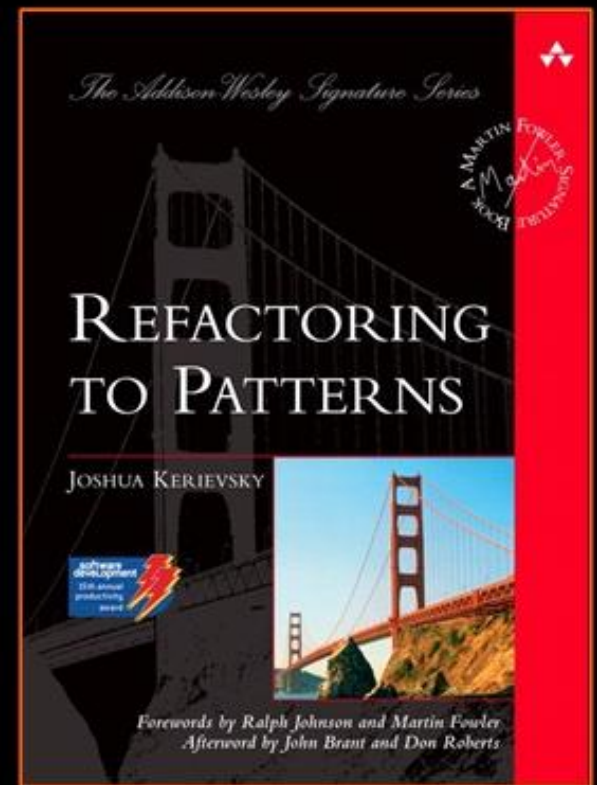
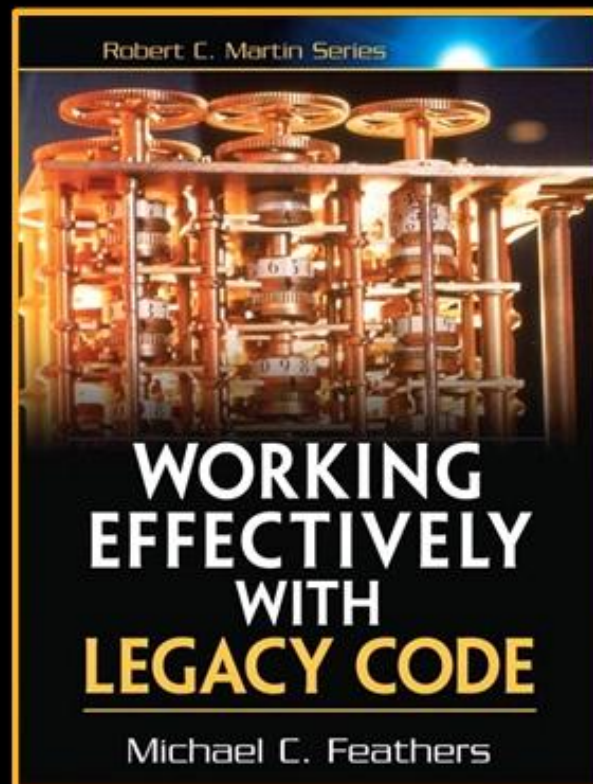
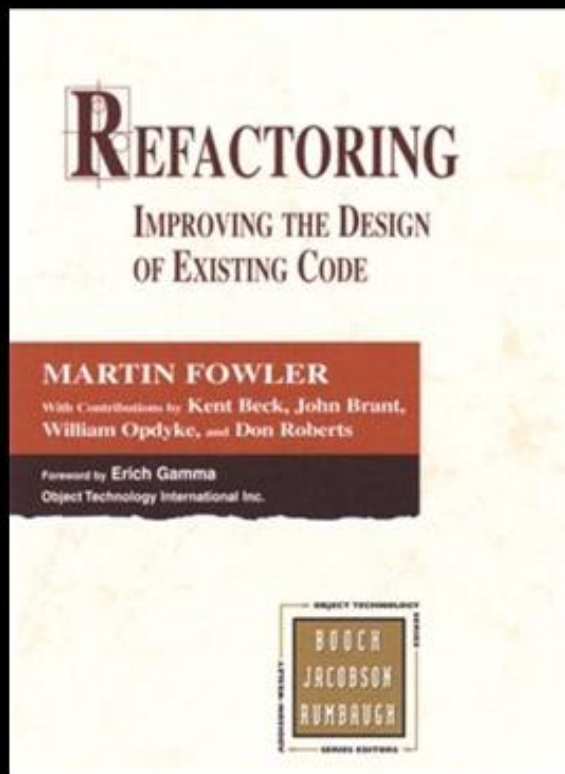
Refactoring code and tests
at the same time
means **TROUBLE**

Use automatic refactoring
as much as possible



Learn the **combos!**

Further resources



CONTROVERSIAL

WOMAN MARRIES A SHARK!

By JIM KINGS

Startling headline! But Andrea Lynn, 30, says she's not a shark. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years.

As the headline reads, the married lady who has been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years.

Everybody says she's a shark. But she's not. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years.

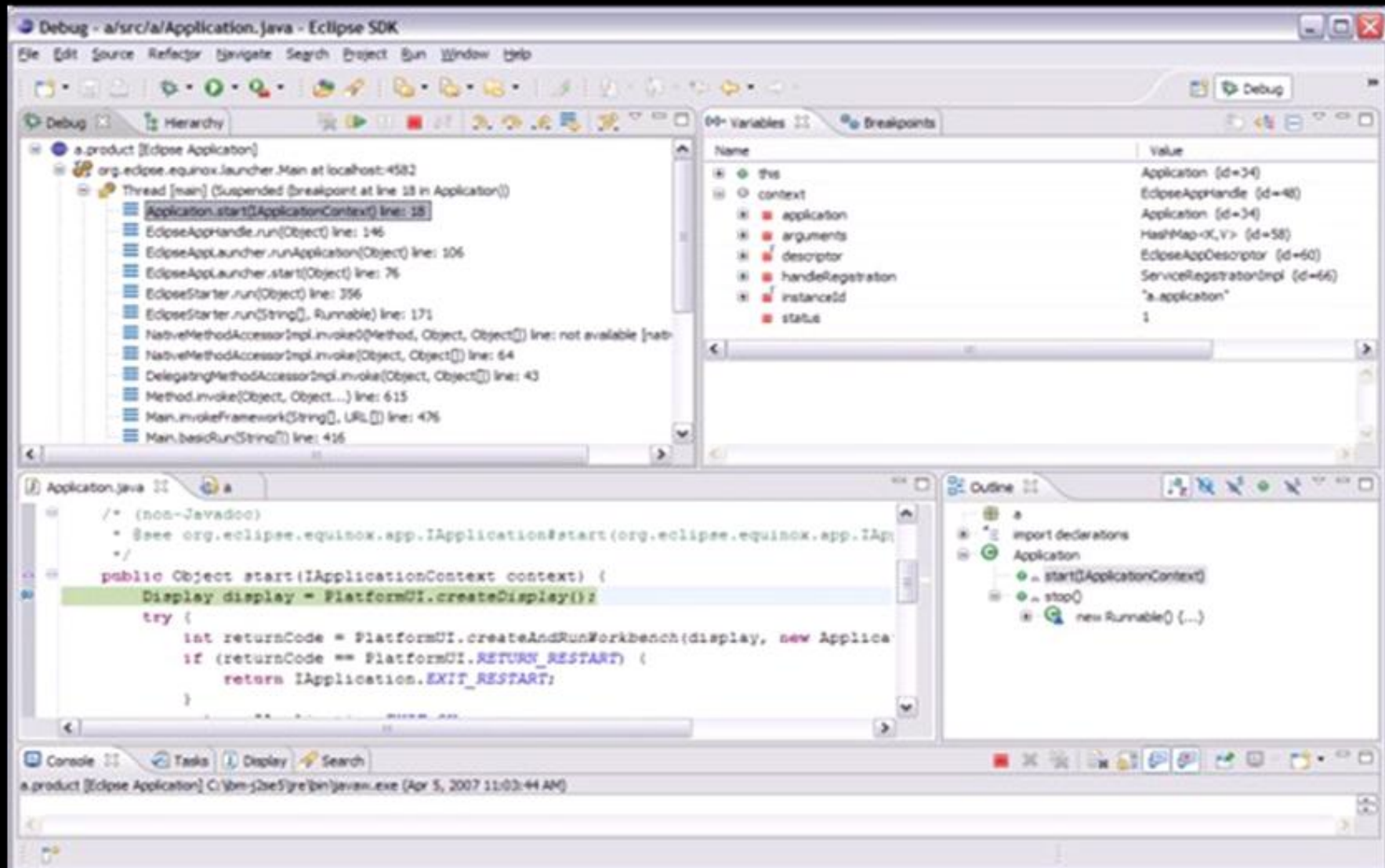
The girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years.

He's a lot better than any man!



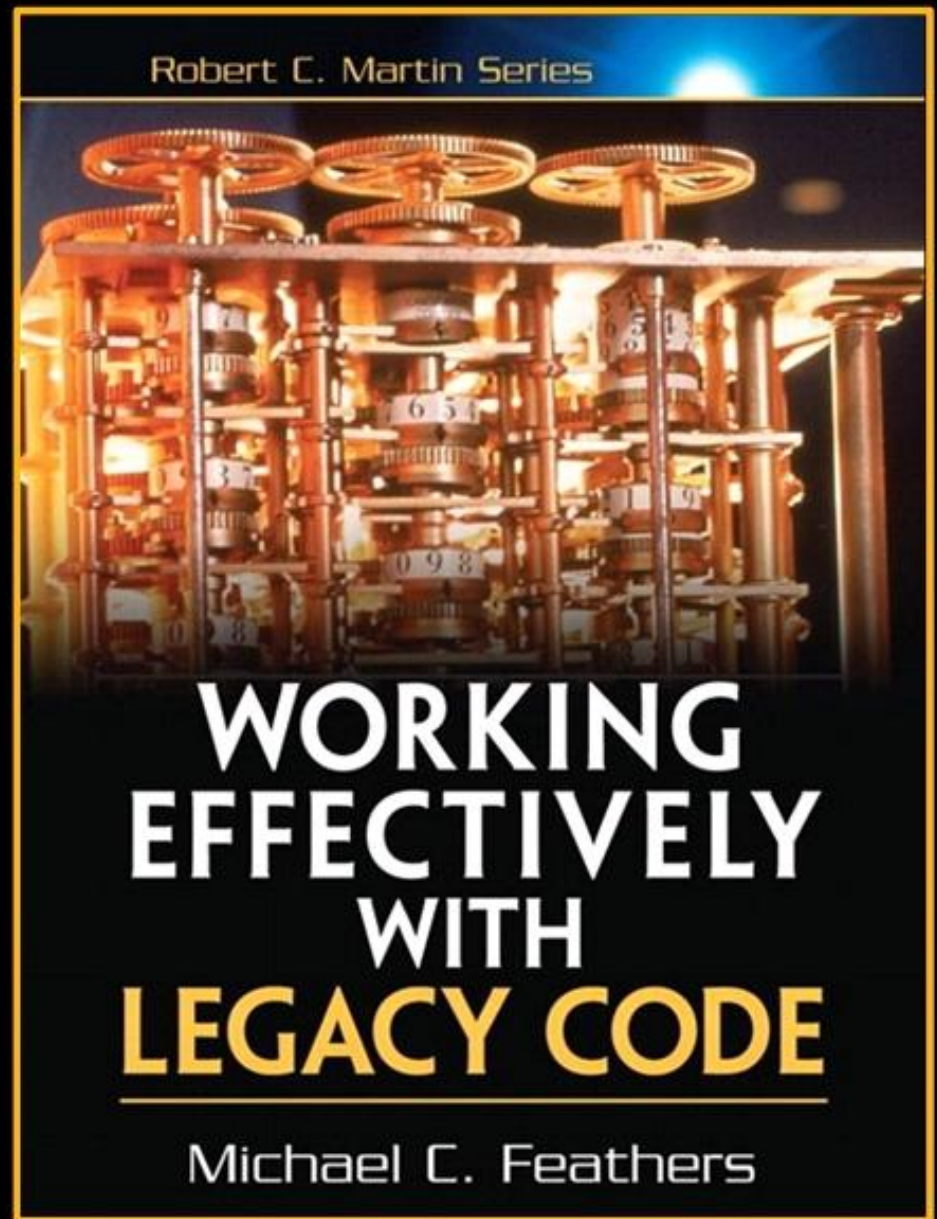
Happy bride Andrea Lynn, 30, says she's not a shark. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years. She's a girl who's been with her husband, a shark, for 10 years.

Use debugger - it's a tool





Use
scratch
refactoring



..and remember
to revert
ALL your changes

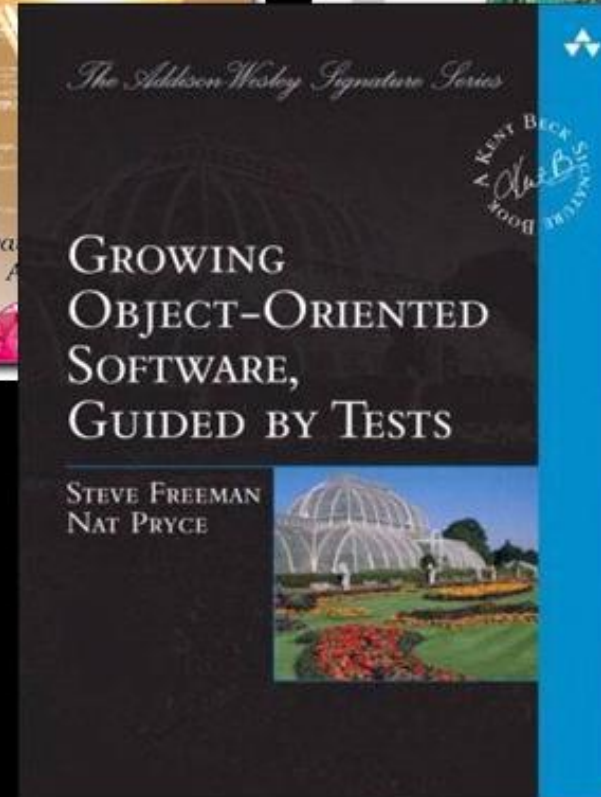
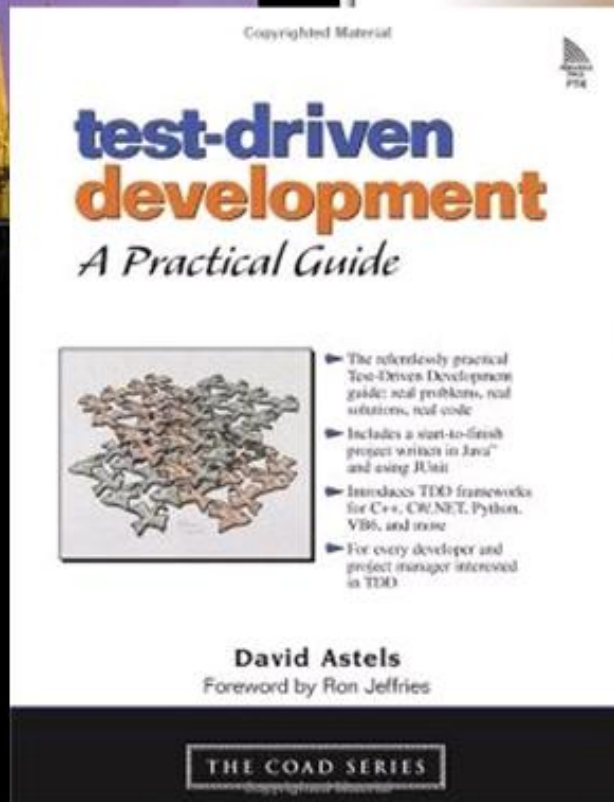
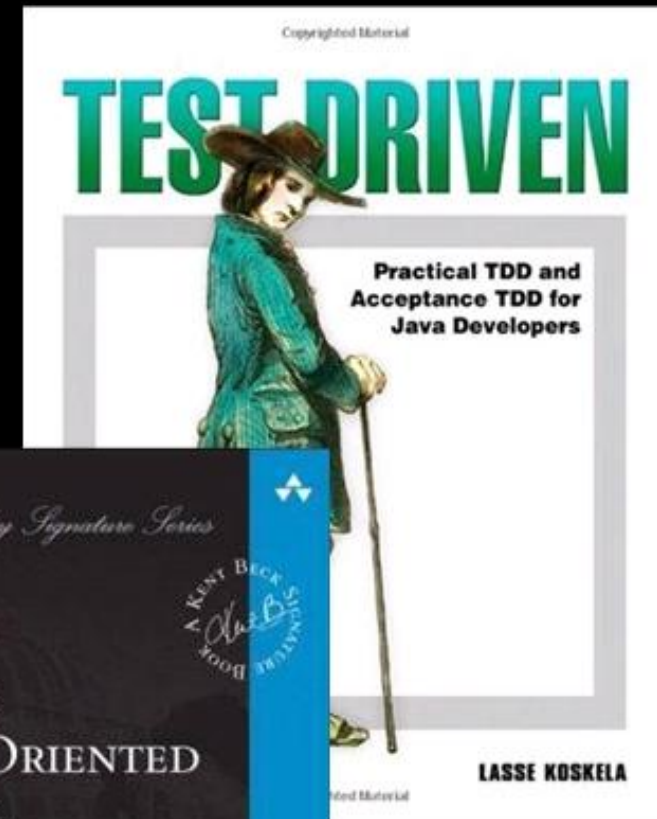
```
A:\>format c: /u
```

```
WARNING, ALL DATA ON NON-REMOVABLE DISK  
DRIVE C: WILL BE LOST!  
Proceed with Format (Y/N)?y
```




learning TDD
is one
of many steps
to become
**Master
Software
Craftsman**

Further resources



R

Fail

G

Pass

B

Refactor

it's really as simple as that ;)

Main take-aways

- Practice, practice and once more – **practice**
- **Divide** and **conquer**
- Programming is **much more** than **writing code**

Uncomfortable Questions

&

Hardly any Answers