

Test-Driven Development: What, Why, and How...

Wednesday, 22 November, 2017

Andy Wharton

TDD: What...

Unit Testing:

- Test first, code later.

Red:

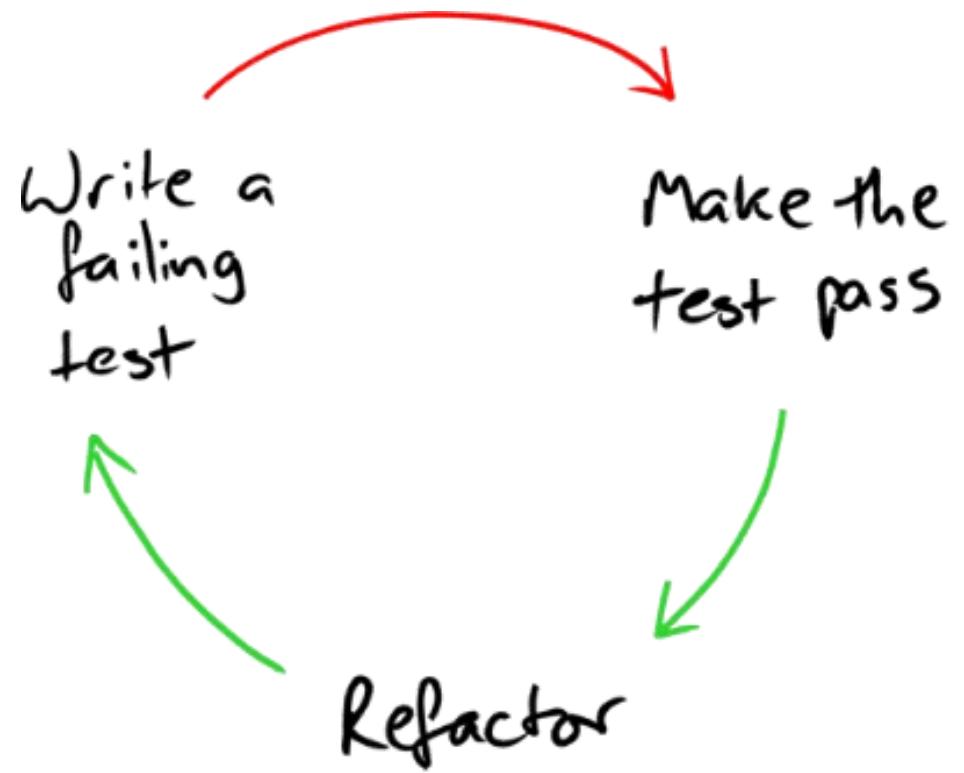
- Write a test.
- Test fails, 😠

Green:

- Write some code.
- Test passes, 😊

Refactor:

- “Clean code that works”, 😇



TDD: Why...

Better Code:

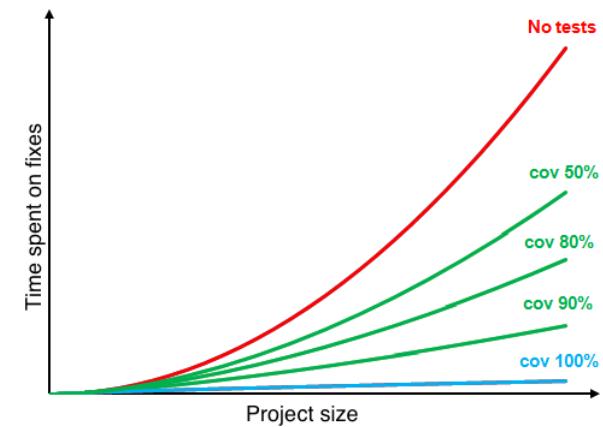
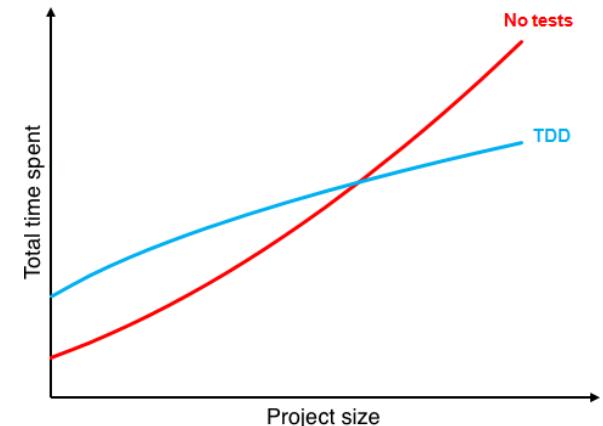
- Refactoring is part of the process.
- Interface design through tests.

Higher Productivity:

- Faster feedback.
- Lower TCO.

Better Documentation:

- Detailed regression test suite.



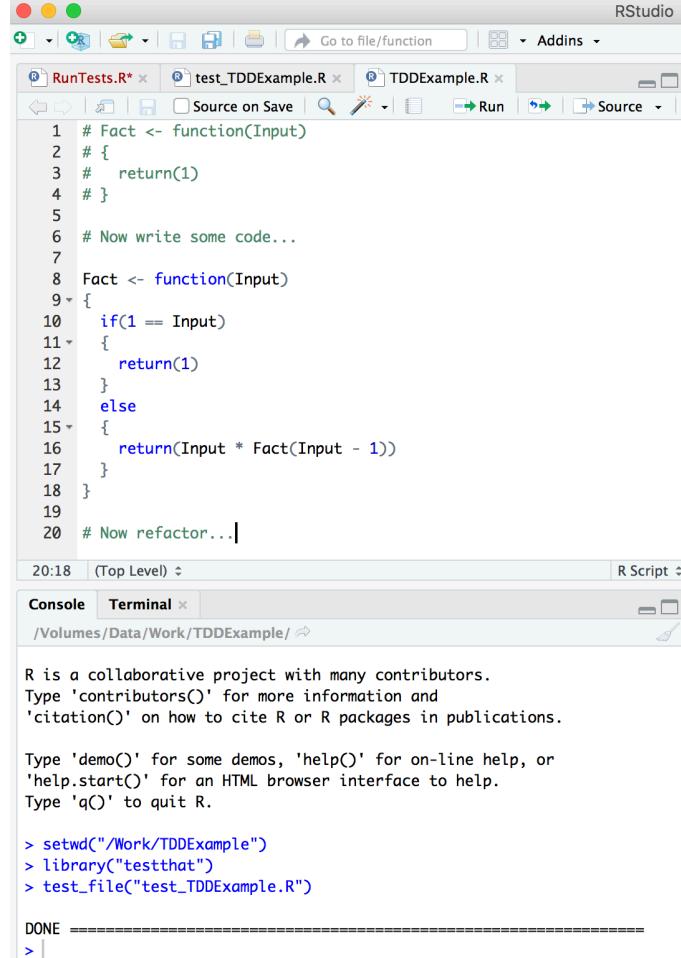
TDD: How (A Toy Example in R)...

Write The Test:

- In `test_TDDEExample.R`
 - `source("TDDEExample.R")`
 - `expect_that(Fact(1), equals(1))`
 - `expect_that(Fact(2), equals(2))`
 - `expect_that(Fact(3), equals(6))`
 - `expect_that(Fact(5), equals(120))`

Run The Test:

- In `RunTests.R`
 - `library("testthat")`
 - `test_file("test_TDDEExample.R")`



The screenshot shows the RStudio interface. The code editor (top panel) displays `RunTests.R*` with the following code:

```

1 # Fact <- function(Input)
2 # {
3 #   return(1)
4 # }
5
6 # Now write some code...
7
8 Fact <- function(Input)
9 {
10   if(1 == Input)
11   {
12     return(1)
13   }
14   else
15   {
16     return(Input * Fact(Input - 1))
17   }
18 }
19
20 # Now refactor...
  
```

The terminal (bottom panel) shows the R environment and the results of running the tests:

```

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> setwd("/Work/TDDEExample")
> library("testthat")
> test_file("test_TDDEExample.R")

DONE =====
> 
  
```

Questions?

Wednesday, 22 November, 2017

Andy Wharton