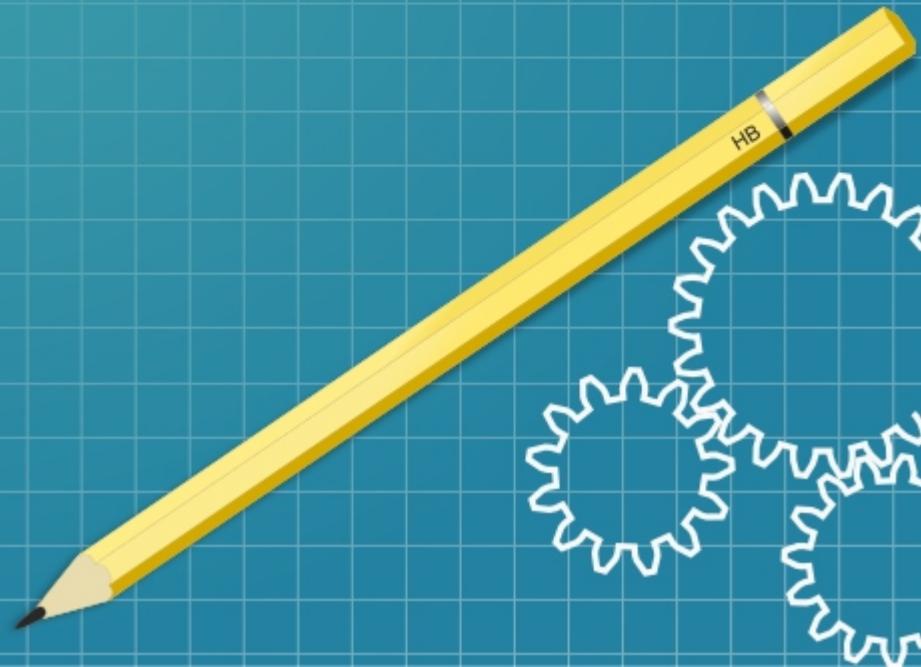


Code Refactoring in Javascript



Agenda



- Introduction to Refactoring
- Initial setup exercise
 - break -
- Refactoring examples
- Your own refactoring journey

Where are my black socks?



Where are my black socks?



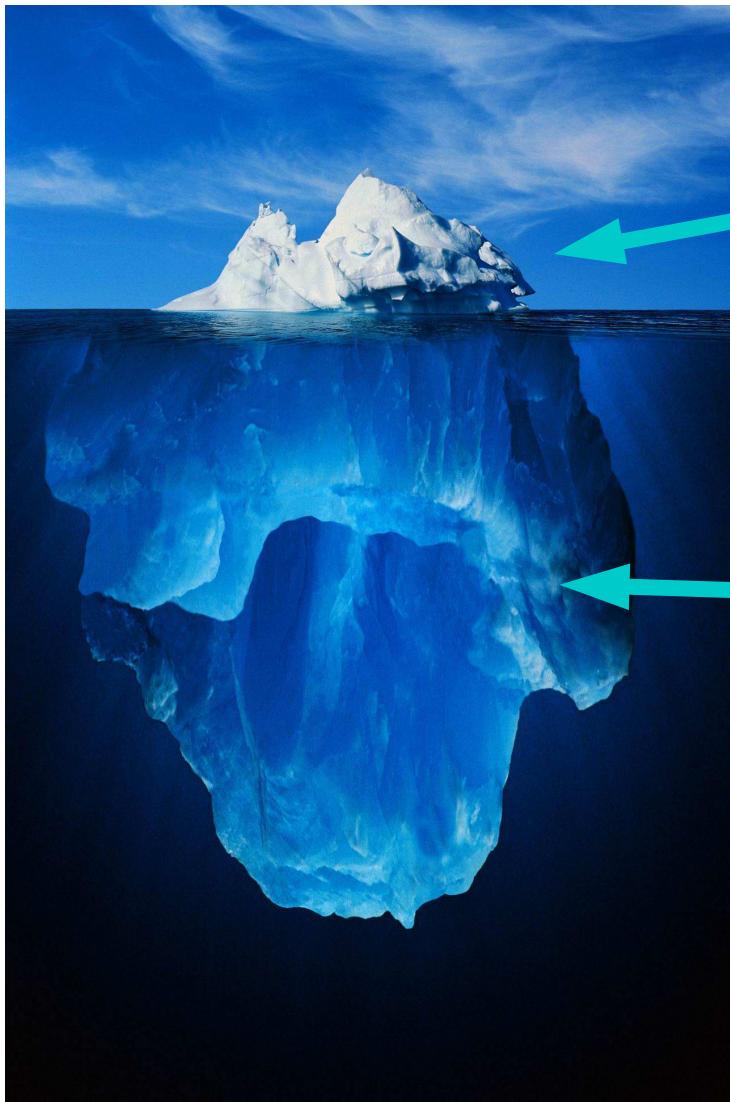
I need a 12mm spanner...



I need a 12mm spanner...



Coding activity



Writing code

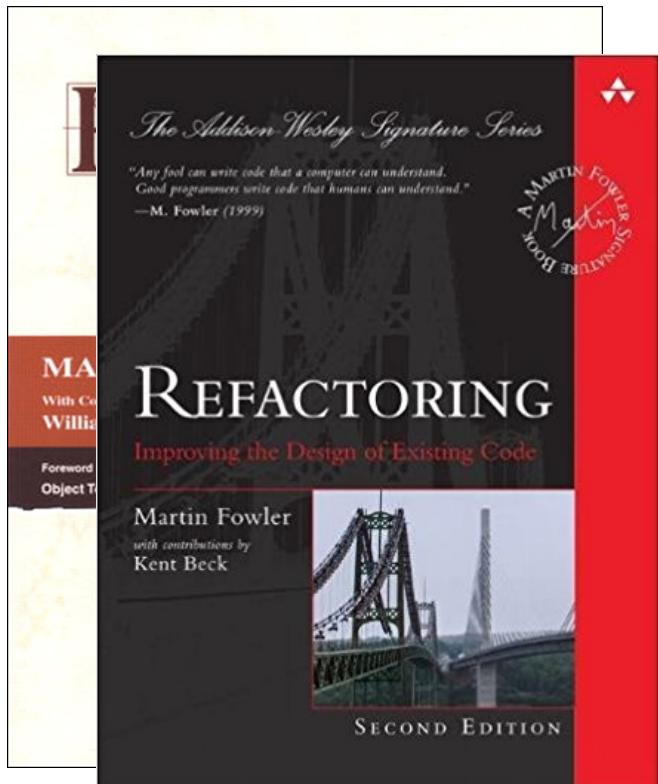
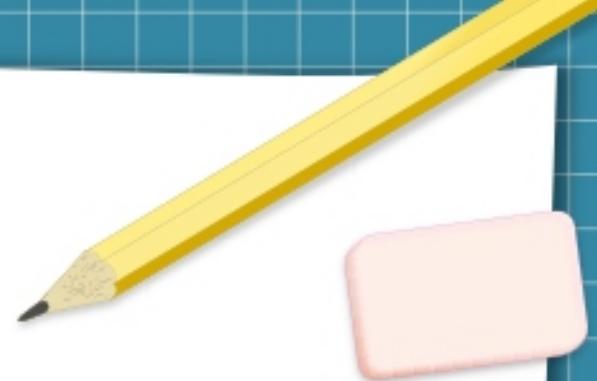
Reading code



*“Any fool can write code that a computer
can understand. Good programmers write
code that humans can understand.”*

- Martin Fowler

Refactoring



Refactoring



“Refactoring is the process of changing a software system in such a way that it does not alter the external behavior of the code yet improves its internal structure.”

- Martin Fowler

Hello demo exercise



- Set up in pairs
....and exchange keyboard often!
- Get the codebase

```
$ git clone  
github.com/iliabartolini/code_refactoring  
$ cd code_refactoring  
$ npm install  
$ npm test
```
- Check the /1_hello/ example



Catalog of Refactorings



Martin Fowler

10 December 2013

This catalog of refactorings includes those refactorings described in my original book on Refactoring, together with the Ruby Edition.

Using the Catalog ►

Tags

- associations
- encapsulation
- generic types
- interfaces
- class extraction
- GOF Patterns
- local variables
- vendor libraries
- errors
- type codes
- method calls
- organizing data
- inheritance
- conditionals

Add Parameter

- Change Bidirectional Association to Unidirectional**
- Change Reference to Value**
- Change Unidirectional Association to Bidirectional**
- Change Value to Reference**
- Collapse Hierarchy**
- Consolidate Conditional Expression**
- Consolidate Duplicate Conditional Fragments**
- Decompose Conditional**
- Duplicate Observed Data**

Pull Up Constructor Body

- Pull Up Field**
- Pull Up Method**
- Push Down Field**
- Push Down Method**
- Recompose Conditional**
- Remove Assignments to Parameters**
- Remove Control Flag**
- Remove Middle Man**
- Remove Named Parameter**
- Remove Parameter**
- Remove Setting Method**

See: <https://refactoring.com/catalog/>



REFACTORING · GURU ·

★ Premium Course

⌘ Refactoring

What is Refactoring

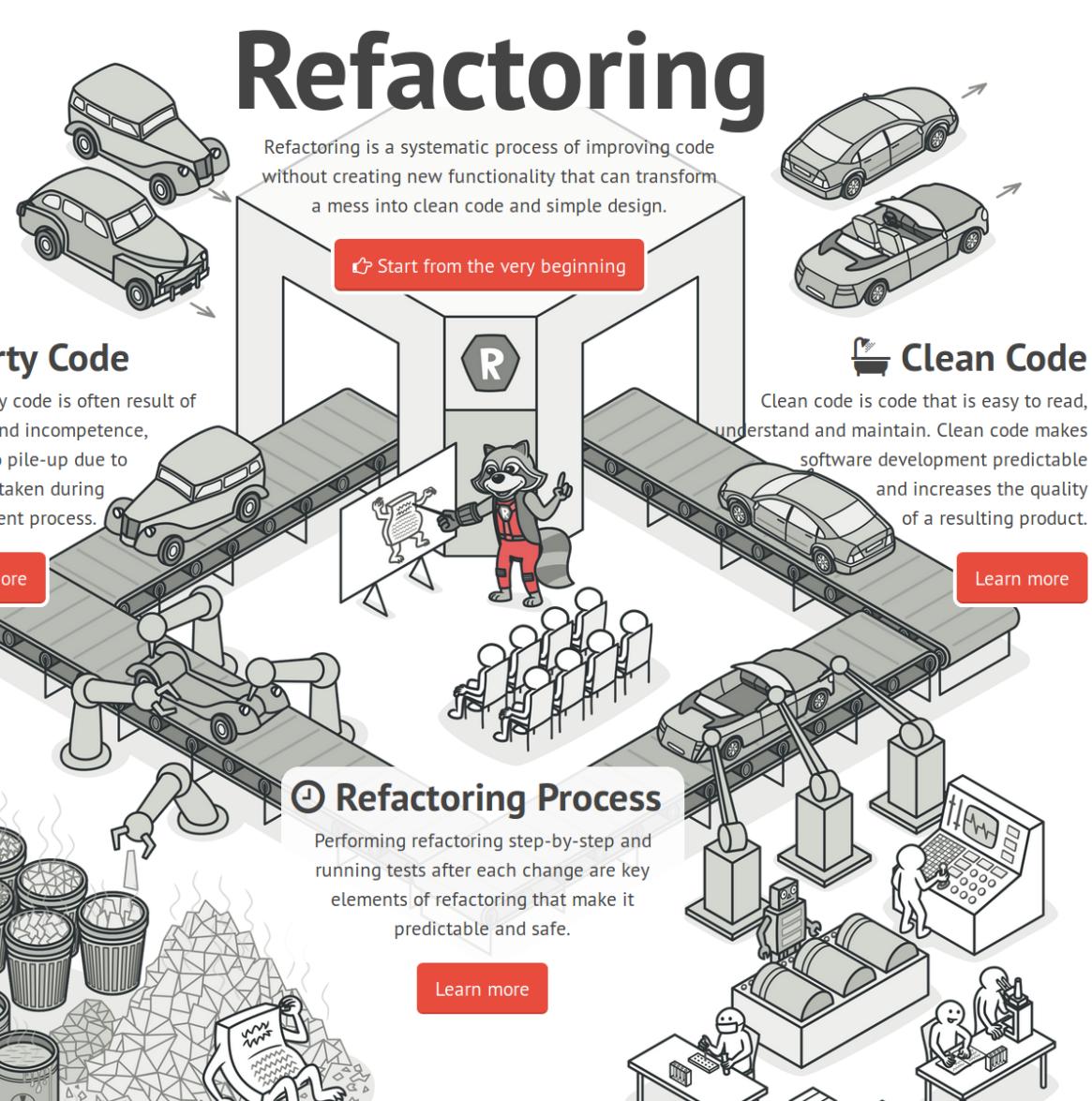
Catalog

Code Smells

Refactorings

─ Design Patterns

👤 Log in 📧 Feedback



See: <https://refactoring.guru>

Let's see...



Rename method, variable, class, ...

Replace magic numbers

Extract / Inline method or function

Extract / Inline variable

Move method, Extract class

Replace if/switch with polymorphism

Pull up / Push down method, Extract interface

Replace inheritance with delegation

Hello demo exercise



- Set up in pairs
....and exchange keyboard often!
- Get the codebase

```
$ git clone  
github.com/iliabartolini/code_refactoring  
$ cd code_refactoring  
$ npm install  
$ npm test
```
- Check the /1_hello/ example

Hello demo exercise



- Set up in pairs
....and exchange keyboard often!
- Get the codebase

```
$ git clone  
github.com/iliabartolini/code_refactoring  
$ cd code_refactoring  
$ npm install  
$ npm test
```
- Check the /1_hello/ example

Smells...



Source: iStock Michał Ludwiczak

Code smells and refactoring



Symptom

- Duplicated code
- Long function/class
- Long list of parameters
- Code “envy”
- Divergent changes
- Common if/switch lists
- Speculative generic code
- Code comments
// are like a deodorant

Treatment

- Extract duplicate part
- Extract & split
- Introduce class
- Move responsibility
- Split responsibilities
- Introduce polymorphism, ...
- Delete, delete, delete
- Rename / Extract method

Practice!



Check the examples in order

/2_date_parser/

/3_direction/

Tools & Language support



A screenshot of a code editor showing a context menu open over a piece of JavaScript code. The code is as follows:

```
1 export default function hello() {  
2     return "Hello, World";  
3 }  
4  
5 }
```

The context menu is open at the end of the final closing brace. The menu items are:

- Copy Reference (Ctrl+Alt+Shift+C)
- Paste (Ctrl+V)
- Paste from History... (Ctrl+Shift+V)
- Paste Simple (Ctrl+Alt+Shift+V)
- Column Selection Mode (Alt+Shift+Insert)
- Find Usages (Alt+Shift+7)
- Refactor (selected, highlighted in blue)
- Folding
- Go To
- Generate... (Alt+Insert)
- ▶ Run 'hello.js' (Ctrl+Shift+F10)
- (Debug 'hello.js'
- >Create 'hello.js'...
- Show in Files
- Open in Terminal
- Local History
- Git
- Compare with Clipboard
- File Encoding
- Diagrams
- Use JavaScript Library
- Fix ESLint Problems
- Create Gist...

Below the Refactor item, there is a submenu:

- Rename... (Shift+F6)
- Change Signature... (Ctrl+F6)
- Move... (F6)
- Copy... (F5)
- Safe Delete... (Alt+Delete)
- Extract (selected, highlighted in blue)
- Inline... (Ctrl+Alt+N)
- Pull Members Up...

At the bottom of the submenu, there are additional options:

- Variable... (Ctrl+Alt+V)
- Field... (Ctrl+Alt+F)
- Parameter... (Ctrl+Alt+P)
- Method... (Ctrl+Alt+M)
- Superclass...

Let's focus on baby steps!



- 1) Identify a smell
 - 2) Make a refactoring
 - 3) Run tests!
 - 4) Repeat
-
- Write down the list of smells you see and refactoring you apply
 - Remember to swap keyboard ;)

Practice!



Check the examples in order

/4_order_receipt/

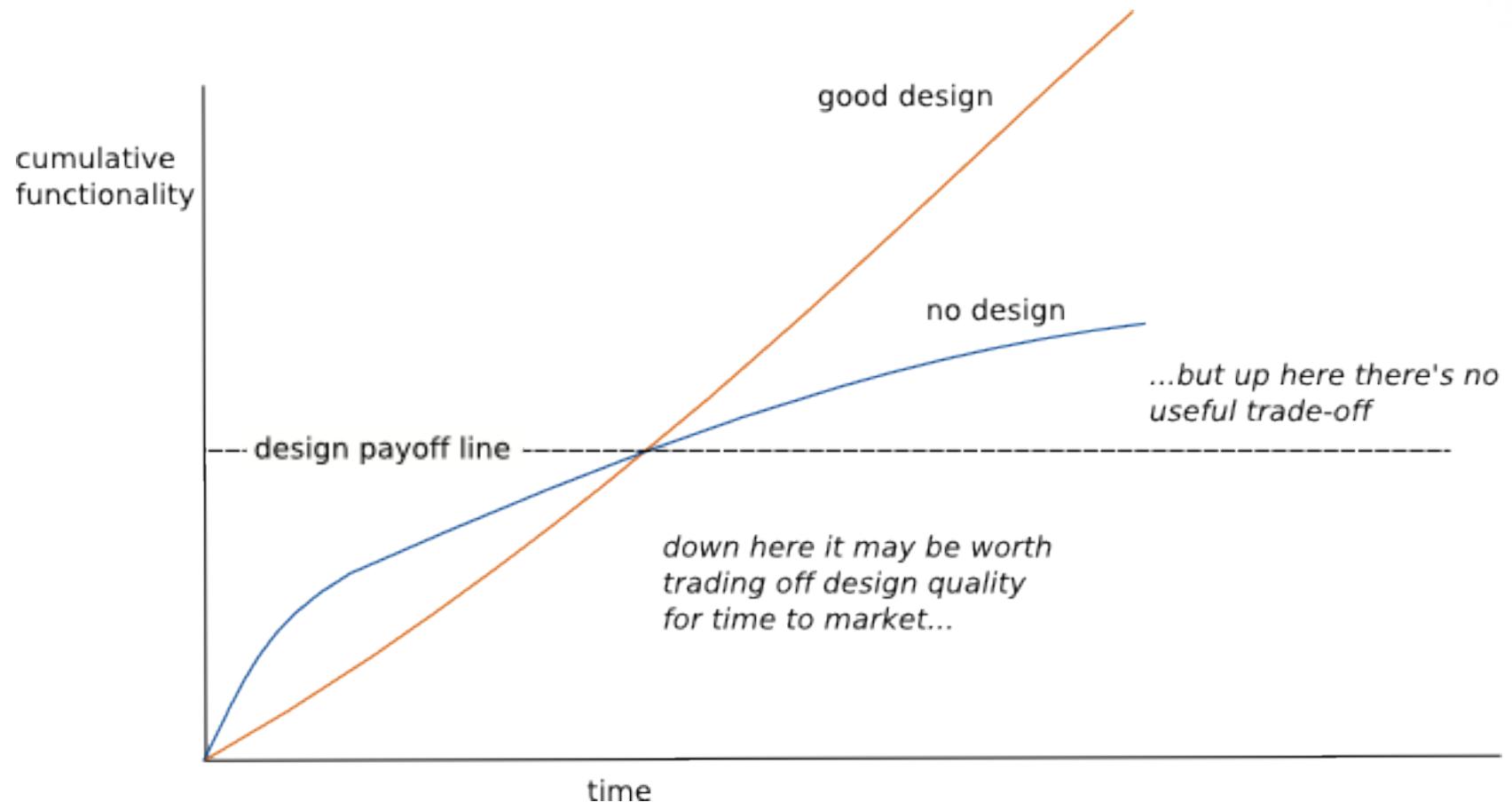
/5_rover/

Myths



“Refactoring takes time”

Good design makes you faster



Myths

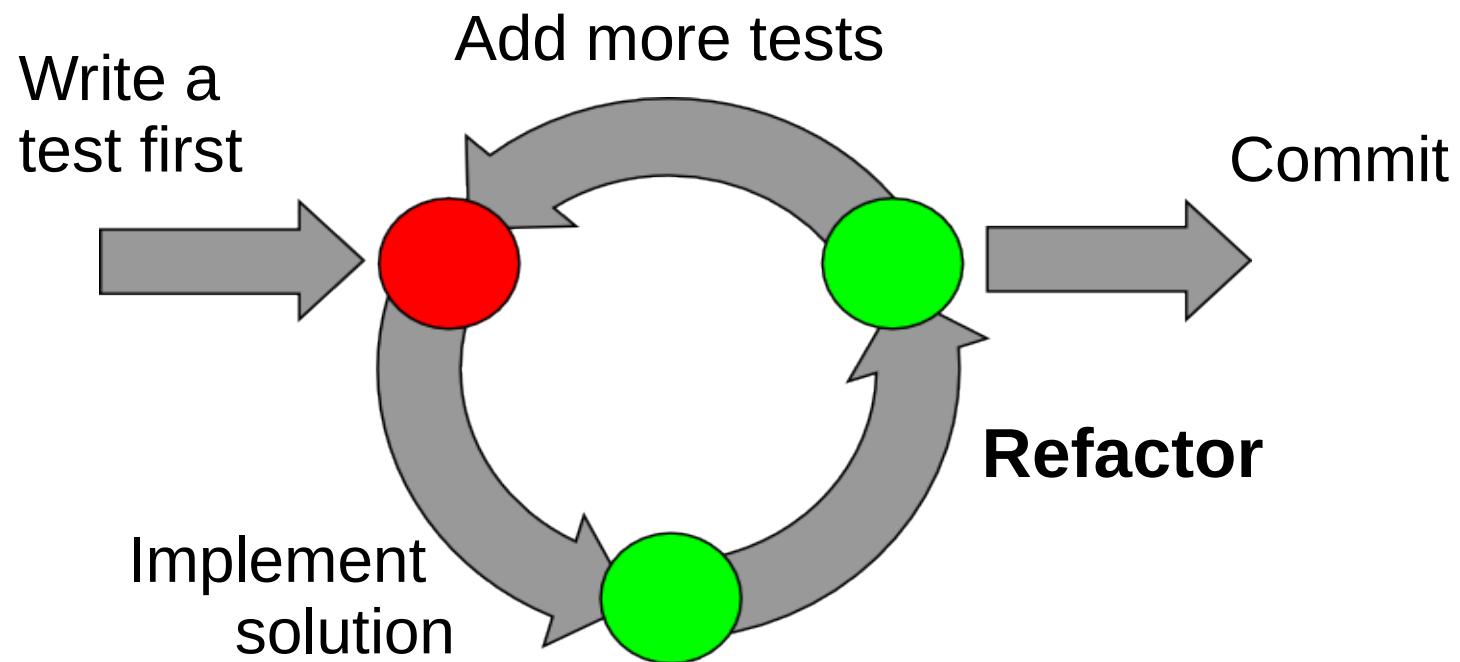


*“You don’t need to refactor
your test code”*

TDD

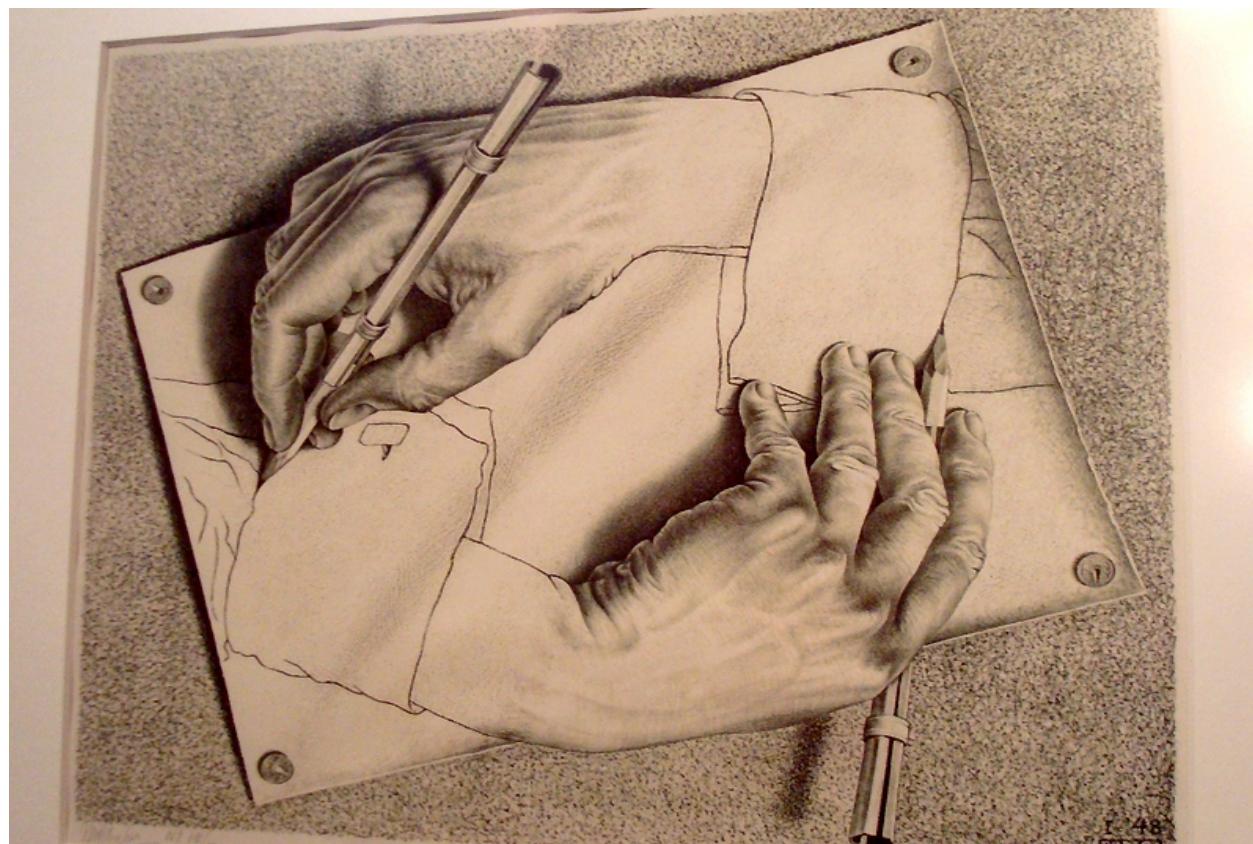


Test Driven Development



“Test code” is “code”

...and should be maintained
with the same quality criteria



Myths



*“We need to stop the project
for this refactoring”*

Boy scout rule

“Leave the campground cleaner than you found it.”



Summary



- Change code in “baby steps”
- Make use of a test suite
- Learn to identify “smells”
- Try to make your code “obvious”
- Continuously improve the “code design”
- Practice, practice, practice makes perfect

:)

Feedback



[bit.ly / f5-feedback](https://bit.ly/f5-feedback)

Thank you! :)

This work is licensed under a Creative Commons
Attribution-ShareAlike 3.0 Unported License.



It makes use of the works of...

Slides template:
Mateus Machado Luna

Images credits:

<https://www.flickr.com/photos/dietjay/3140774470/>

<https://www.flickr.com/photos/147489968@N06/>

<https://www.flickr.com/photos/silvertje/2197752384/>

https://www.flickr.com/photos/mtneer_man/7819976694/

<https://www.flickr.com/photos/lingoflora/2331681471/>

Hello demo exercise



- Set up in pairs
....and exchange keyboard often!
- Get the codebase

```
$ git clone  
github.com/iliabartolini/code_refactoring  
$ cd code_refactoring  
$ npm install  
$ npm test
```
- Check the /1_hello/ example