

# XTREM TDD



## Game rules

This game is played in teams.

Sit around a table and position the cards face down without revealing them or laying them out.

We highly recommend you to use **mob programming** during the whole game.

## Game principle

As soon as you see a number (or a letter) on a card, you have the right to look for the matching card in the deck and flip it to read it.

Attention point: some numbers may be hidden!!!!

## Discarding cards

At the top of some cards you may find numbers and/or letters that are **crossed out**, you have to immediately **remove** these cards from the game by **putting them aside**.

## Material



Playing cards



A computer

### Knowledge Base



Git repository



## Types of cards

### Grey and Green cards

You will find places, information, and references to draw other cards.



### Blue and Red cards

During the game you can combine blue cards with red cards to draw a new card.



You can do it by adding their numbers together.

$$\begin{matrix} 13 \\ + \\ 27 \end{matrix} =$$



### Yellow cards

Those are key moments in the adventure.

Stop playing with the cards, spend time understanding the given practice and use it in the source code.

Use the QR code at the back of the card to find details. Once implemented you can continue with the cards...



### Success cards

Allow you to remember practices, techniques and approaches you have discovered and debrief them.

## Let's start the game

You know enough to start the game.

Communication, sharing ideas, experimentation will be the keys to your success.

Take the adventure card and follow the instructions.

# XTREM TDD



Welcome in this T.D.D adventure !!!

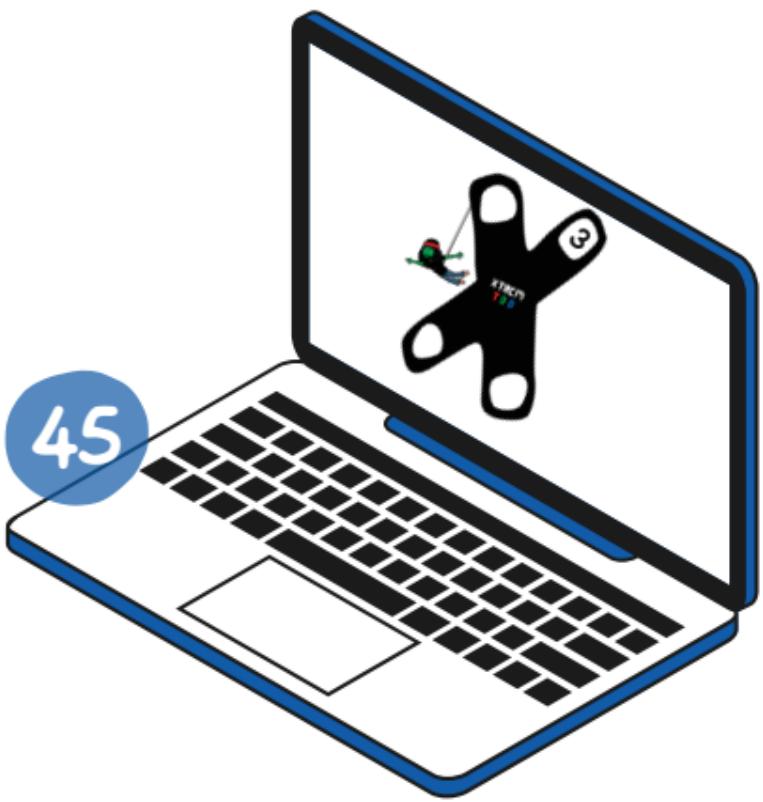
You have seen and experienced many stories of unmaintainable, untested code that no one knows about.

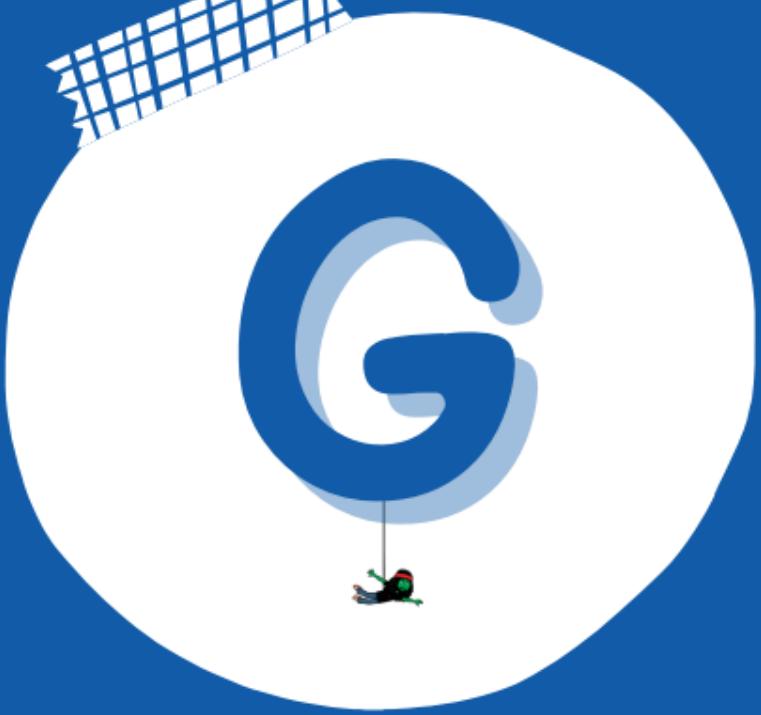
More than ever determined to demonstrate that **software craftsmanship** allows us to avoid this kind of situation, you decide to participate to this adventure.

**Your mission:** work in teams on an existing money project that you will have to make it evolve and maintain by applying given practices, techniques and approaches.



Your adventure begins as soon as you  
flip this card.





XTREM TDD

G



# Make the failing test pass

by writing production code

0

"Write only enough production code to **pass** a failing test."





XTREM TDD





## No for loops

What do you **take**  
from this step?



It can help you reduce **code complexity**  
and avoid breaking  
**command / query separation.**



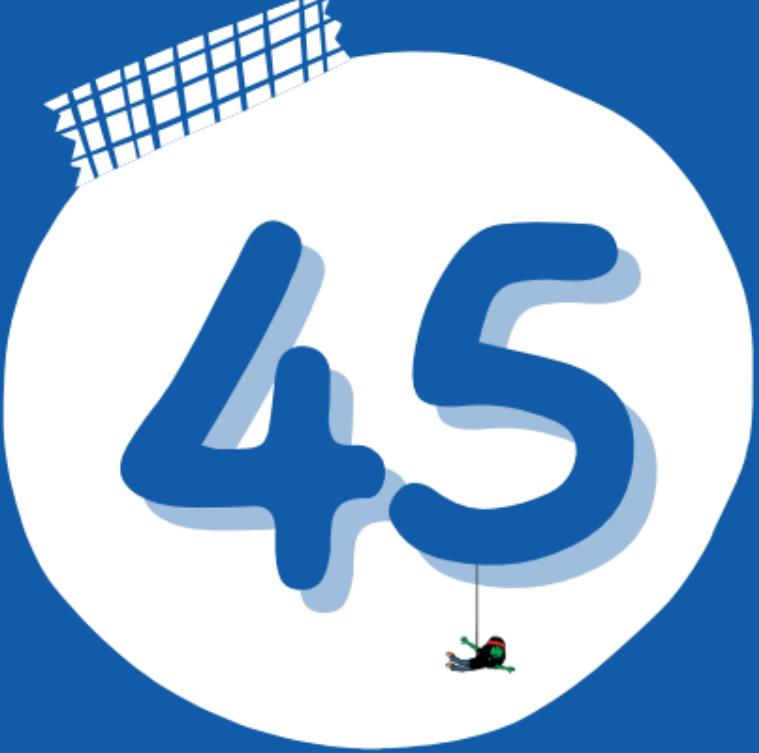
XTREM TDD

5



**FAKE NEWS**

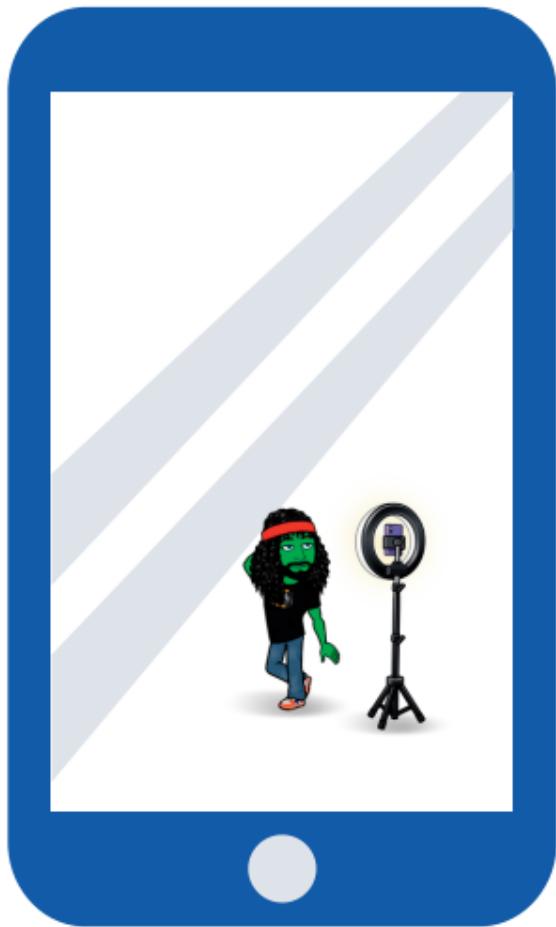




XTREM TDD

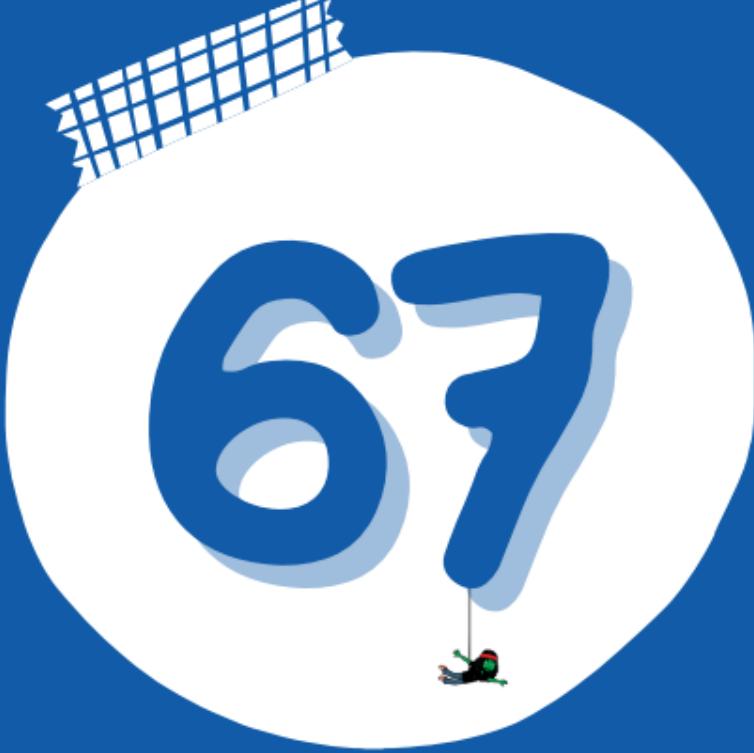


45



A useful object:  
**your smartphone**



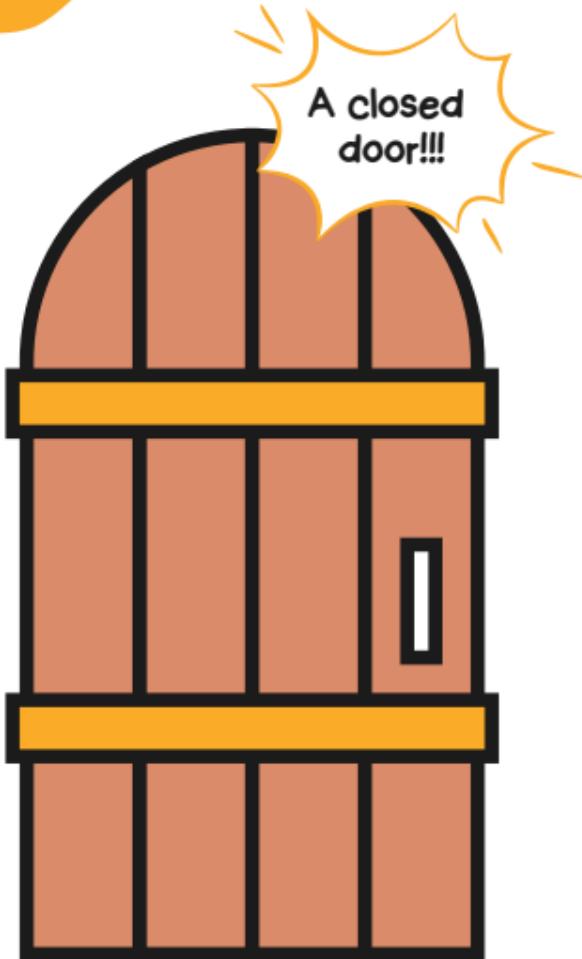


XTREM TDD



67

48



One test provides no value at all.

Its

expected  
result

will open this door.





XTREM TDD



9

31

R

G

B



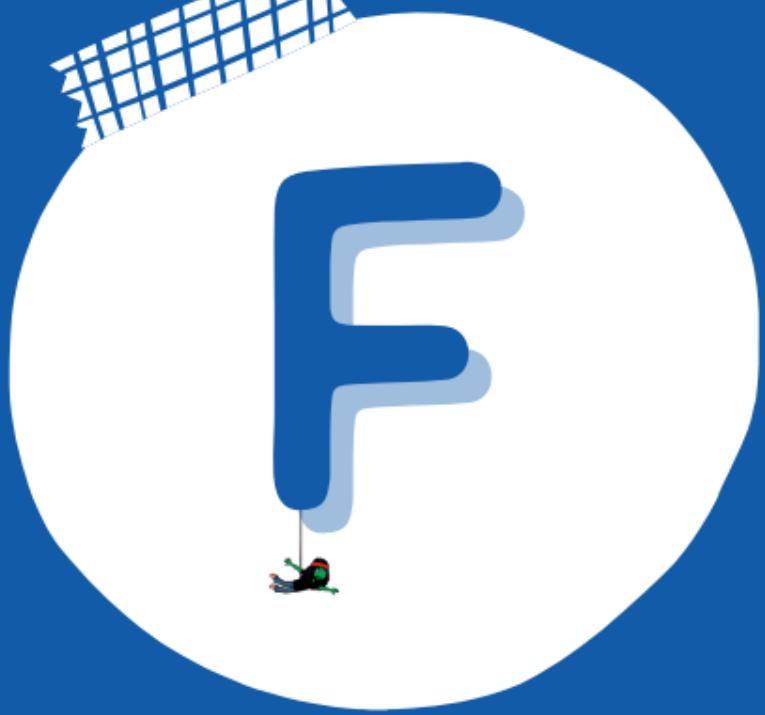
Another closed  
door !!!



Implement portfolio features  
using **TDD** to open it.

Flip this card, scan the QR code,  
follow the instructions then continue.



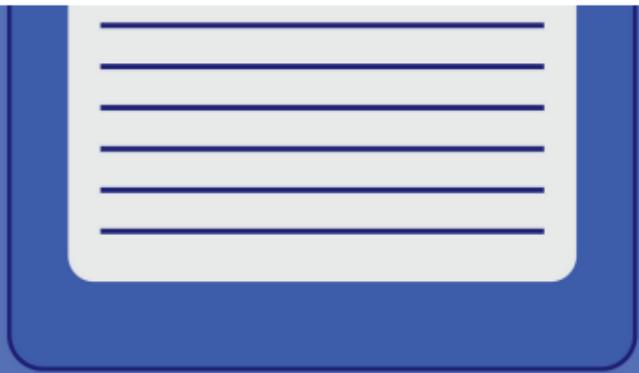


XTREM TDD

F



A very old floppy disk...





XTREM TDD

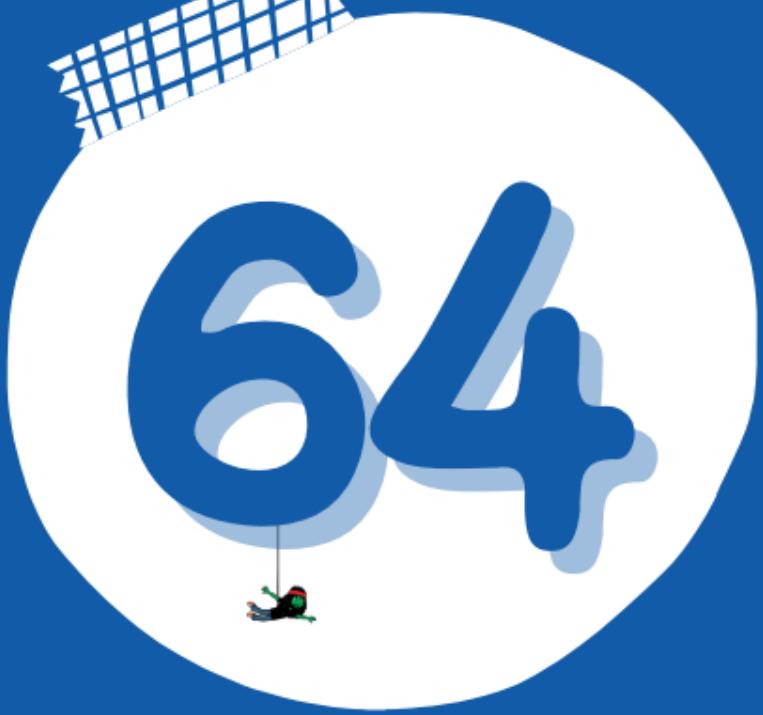


84



What are  
those signs?





XTREM TDD



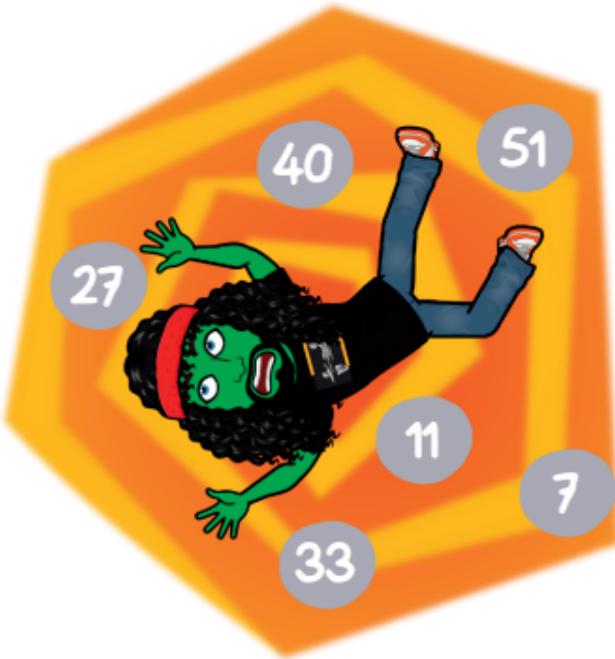
64

86



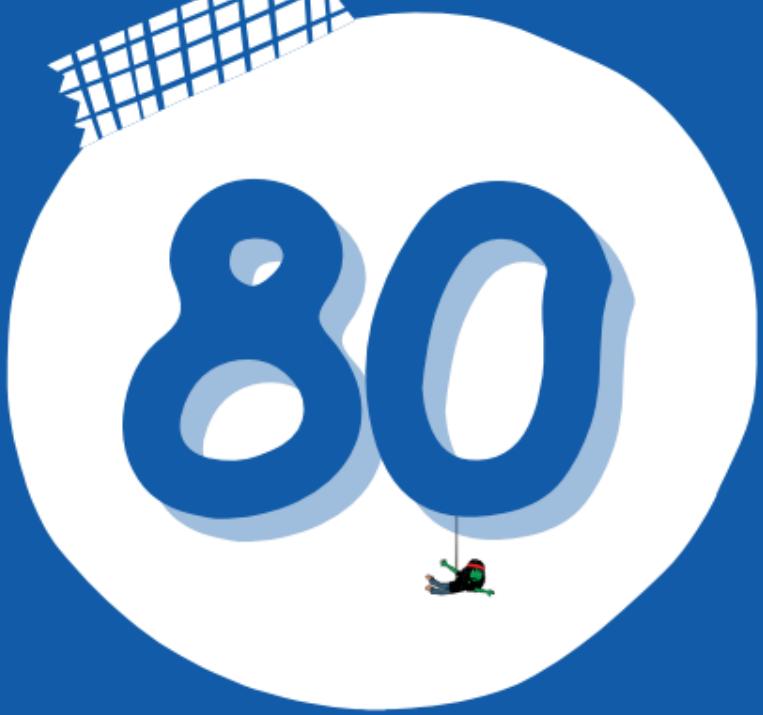
A movie poster announcing  
a new adventure.

# Alice in Monadslang !!!



A Guillaume Faas / Yoan Thirion film  
Starring vavr, language-ext





XTREM TDD



11/62

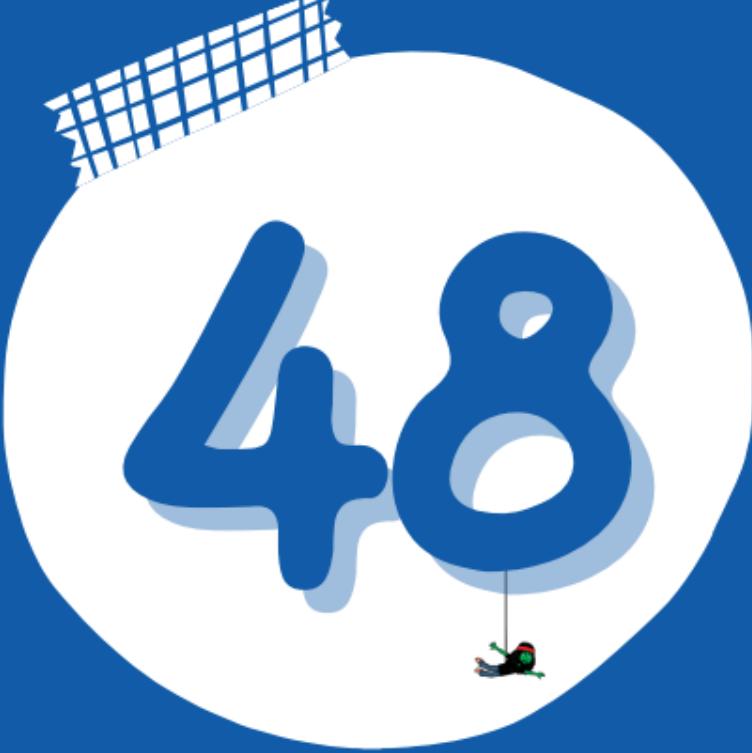


## Property-Based Testing to challenge our Domain Model

What did you **discover** about the Domain Model thanks to it?



How do you think you could **apply** this approach?



XTREM TDD



48

45

3



Scan this QR code and clone the  
repository behind it.



Take some time for deep dive into it and discover  
the business, the source code.

Then draw

67





XTREM TDD

95

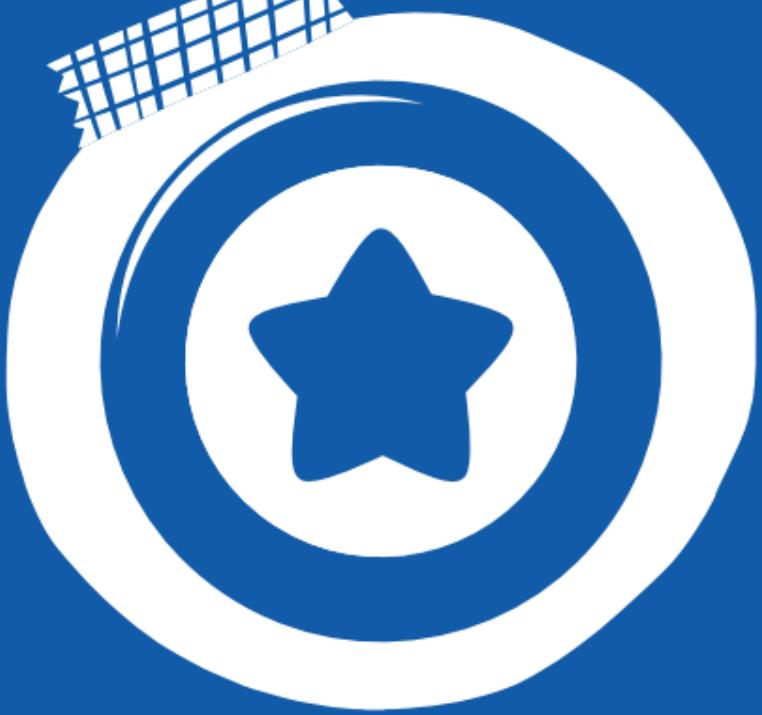
25



Glad you have removed  
"for loop(s)" !!!

84





XTREM TDD



20



A coin...





XTREM TDD

7



... is a monadic container type which represents a computation that may either result in an exception, or return a successfully computed value.



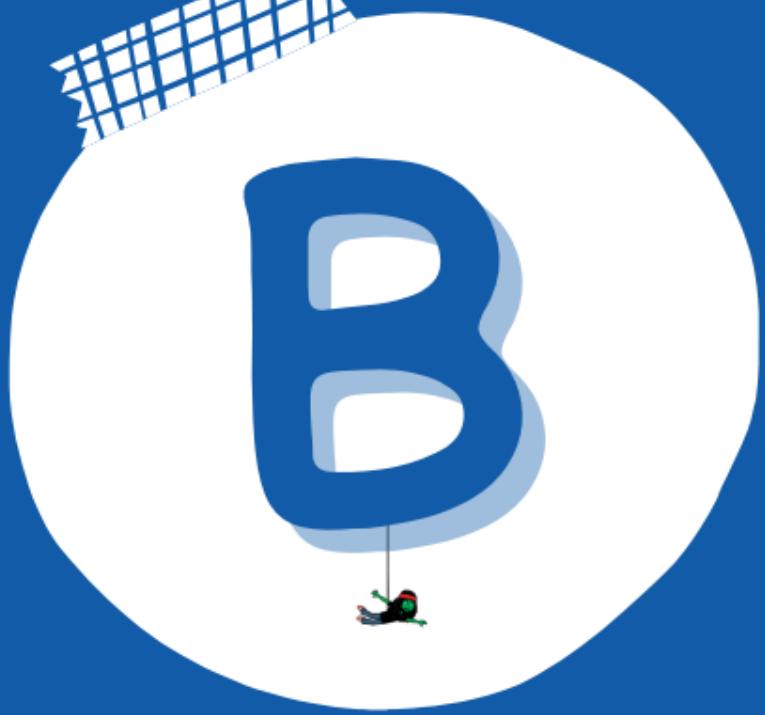
XTREM TDD

3



What could we do with that?





XTREM TDD

B

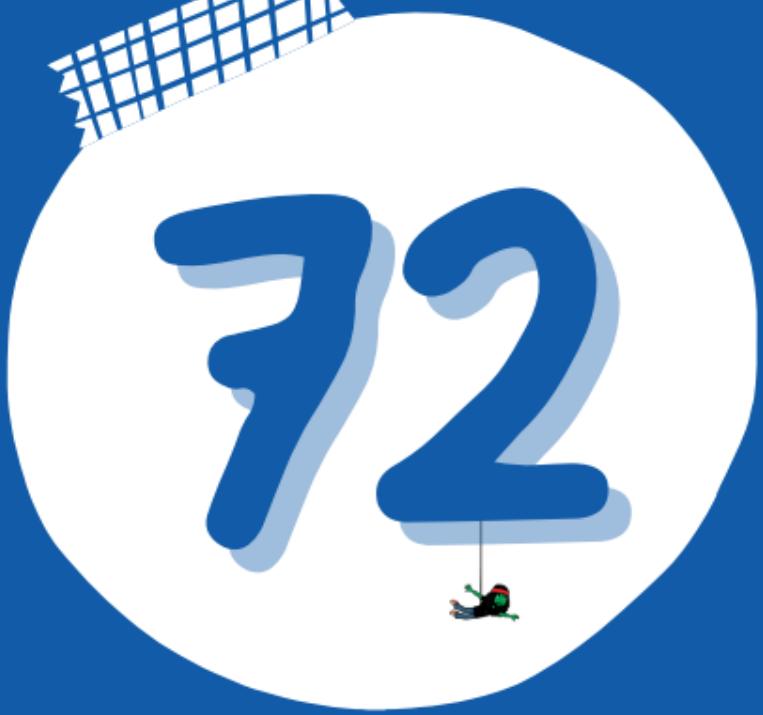


# Refactor

Follow clean code practices

Proper naming / extract logic / simplify code

01



XTREM TDD



Referential transparency

Thread-safety

No invalid state

It looks like notes from a former  
trainee on immutability.



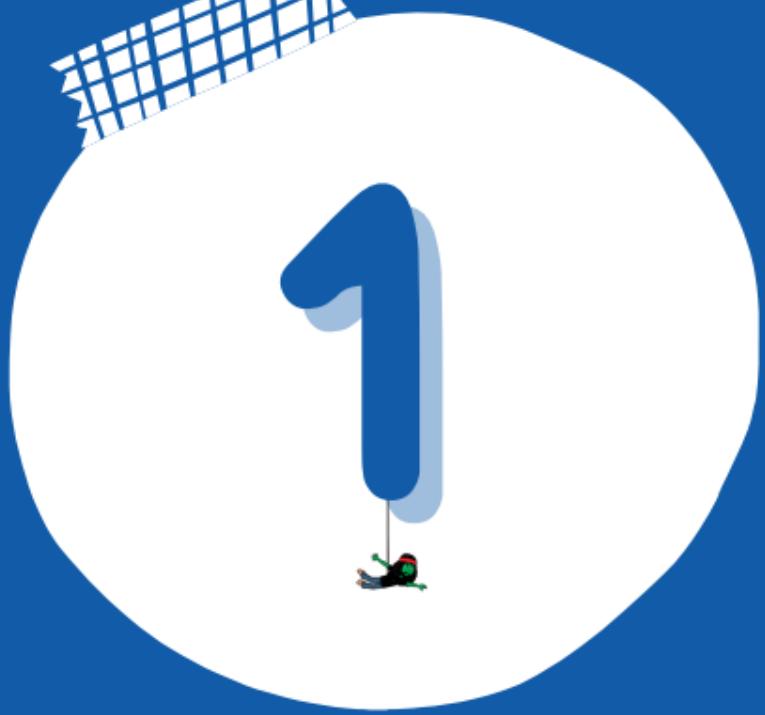
XTREM TDD

27



# Option

1



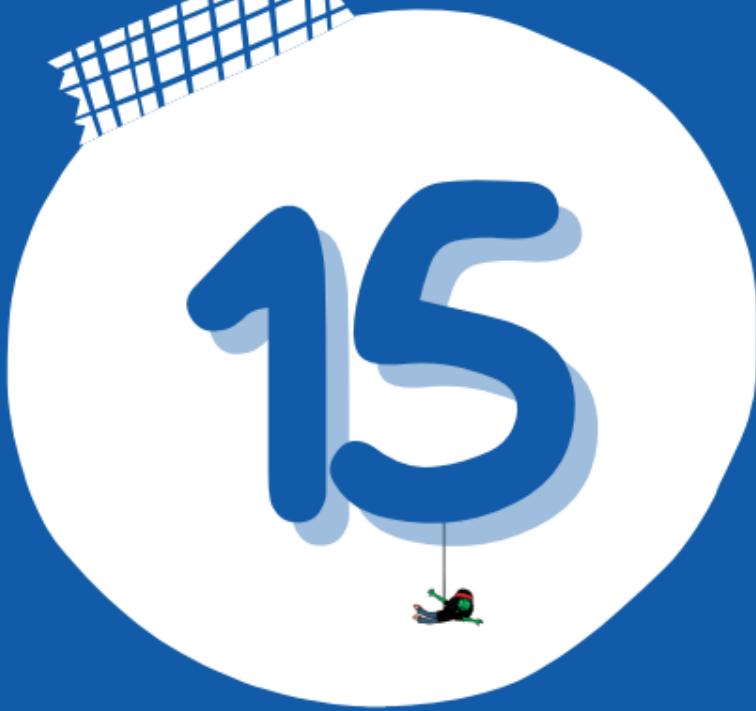
XTREM TDD

1



Use all available information  
to find the next card.





XTREM TDD



15

67



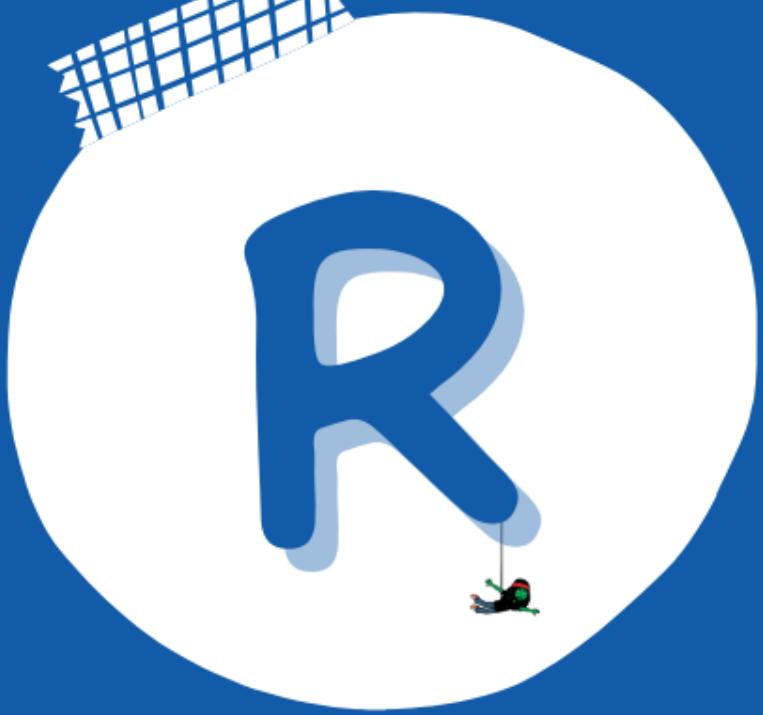
Congratulations !!!

You just opened your first door.



Flip this card, scan the QR code,  
follow the instructions then draw





XTREM TDD

R



# Write a failing test

based on a specific requirement

1

"Write only enough of a test to  
demonstrate a **failure**."





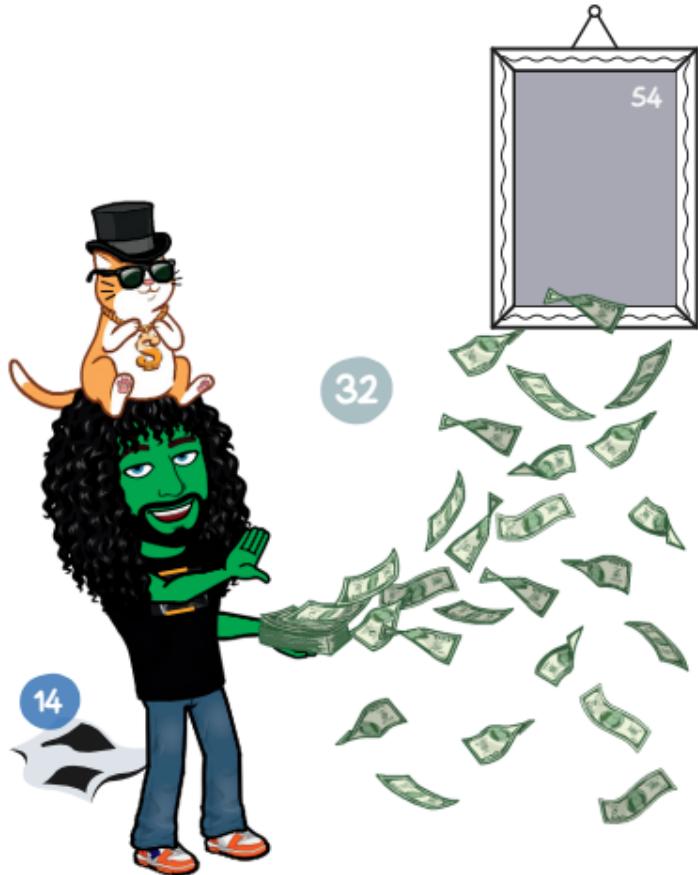
XTREM TDD

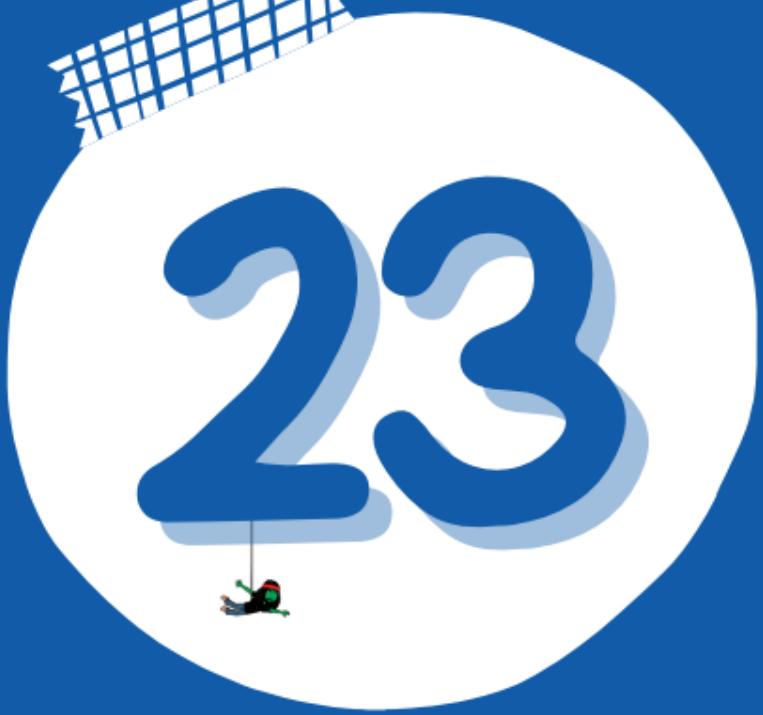
22

9



Congratulations !!!



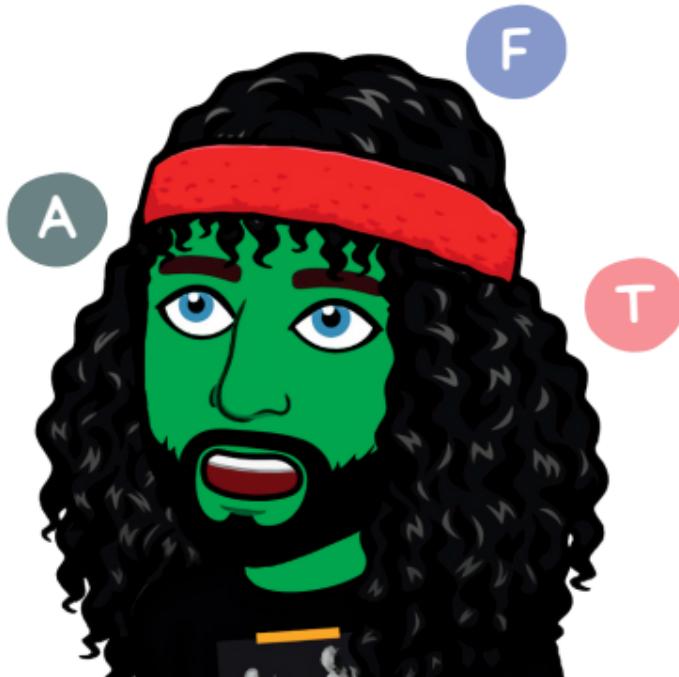
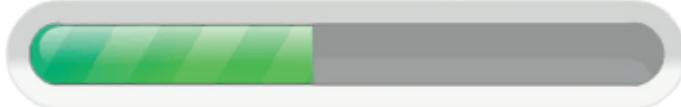


XTREM TDD

23

6

68



Congratulations you have  
beaten primitive obsession !!!





XTREM TDD



25/62

4

36

84

10

95



To  $\infty$  and beyond



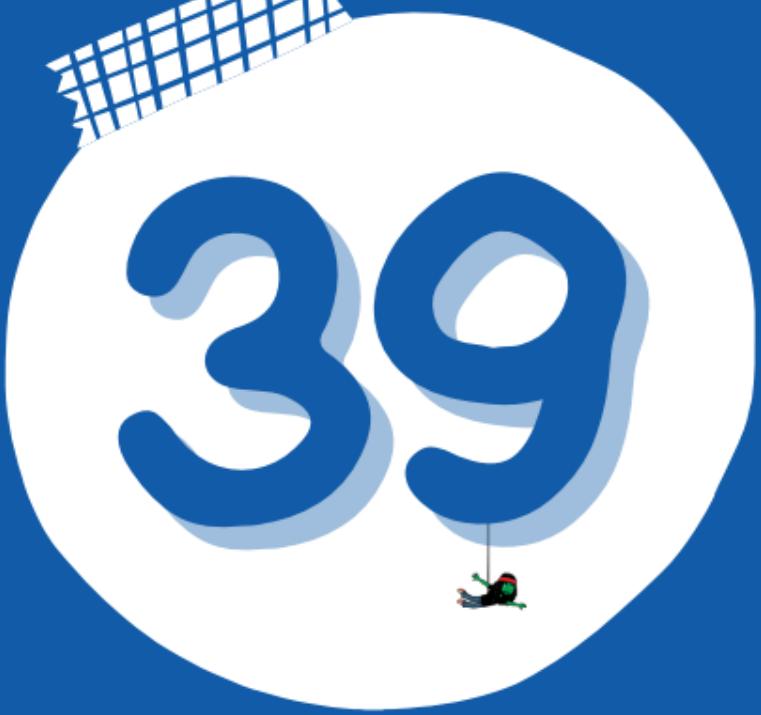
with immutability.



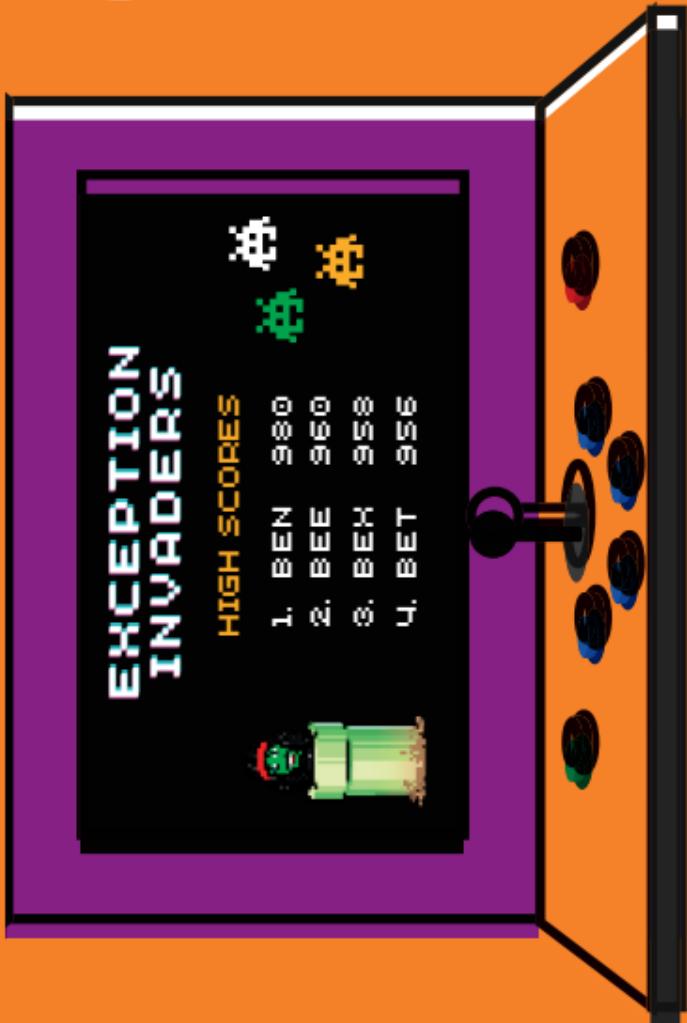
XTREM TDD



... is a monadic container type which  
represents an optional value : Some or None.



XTREM TDD



IT STARTS, THE BEST SCORES APPEAR ON THE SCREEN



XTREM TDD

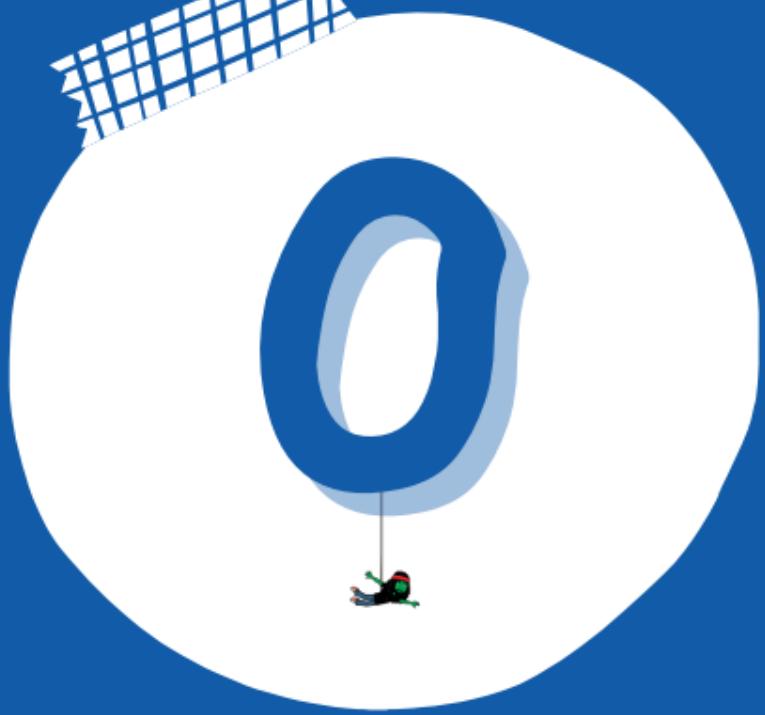
24

16



What an adventure...  
It is almost time to stop...





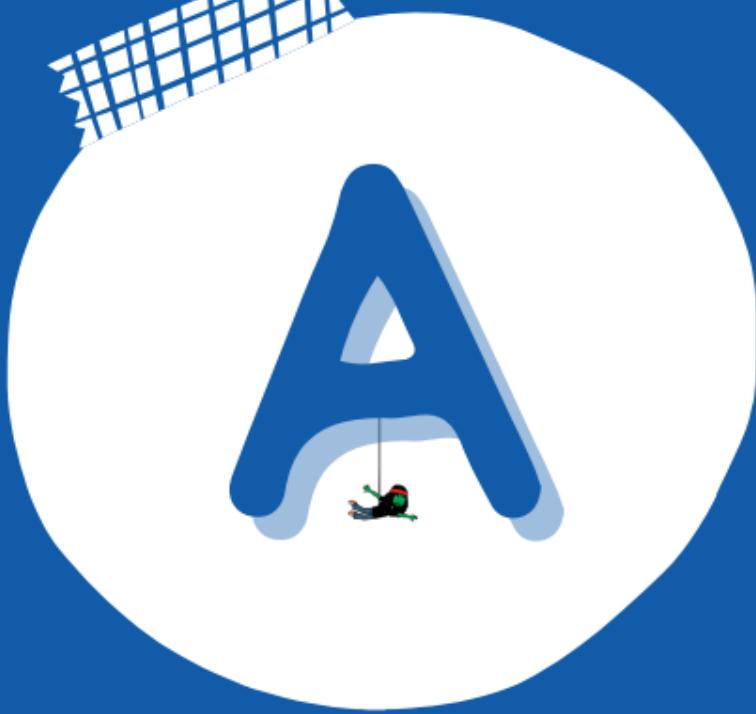
XTREM TDD

0



Use all available information  
to find the next card.





XTREM TDD

A



A very old floppy disk...





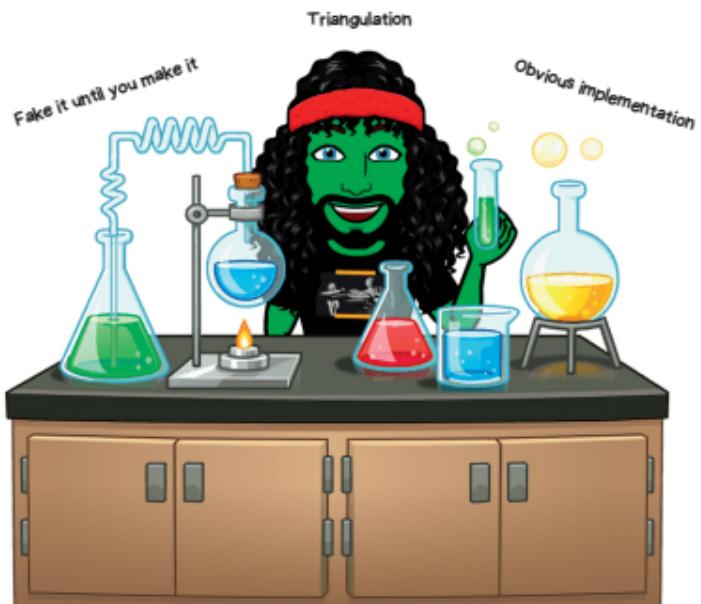
XTREM TDD





## Implement Portfolio with TDD

What do you **think** about TDD?  
What did you **discover** about it?



A **scientific approach** to Software Development

### Write an hypothesis

- Prediction of falsifiable outcome
- Perform an experiment
- Measure the result

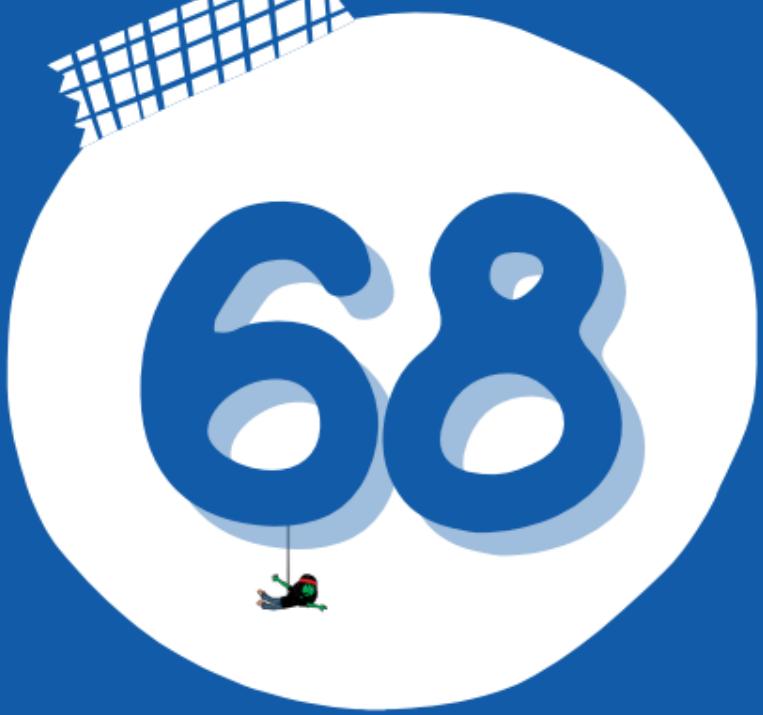
### Observe the outcome

Can you improve the code ?

### Run the experiment

Compare the actual to the predicated one





XTREM TDD



68

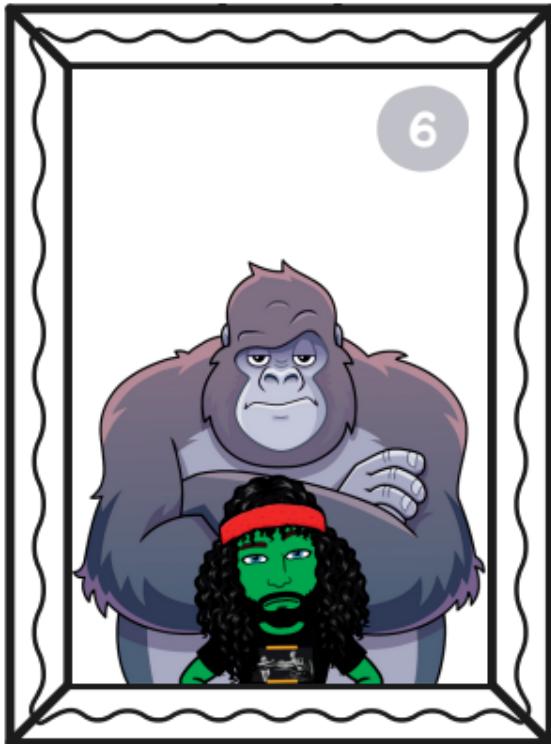
14

54

22



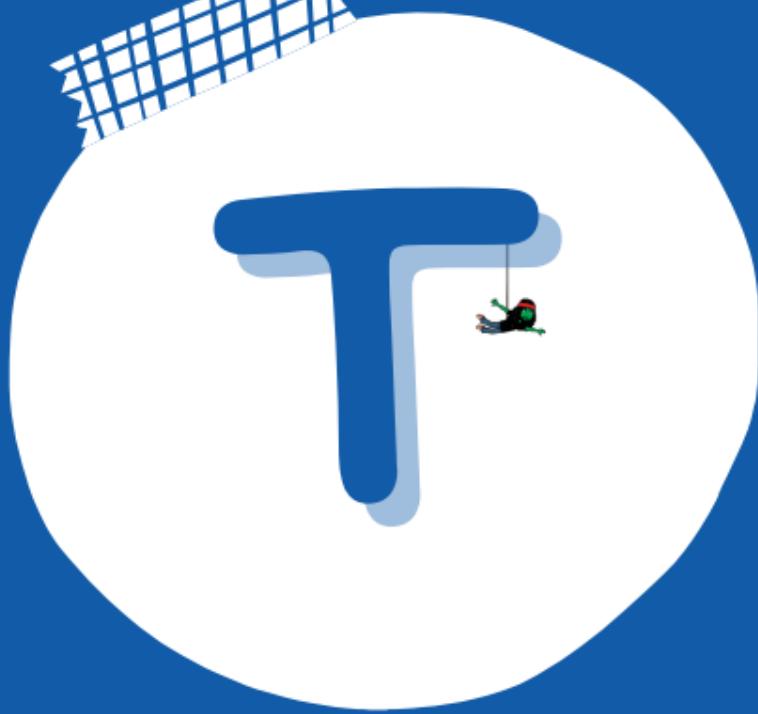
Let's fight  
primitive obsession !!!



The rag is ideal for dusting the painting,  
you need to open the door behind it now.

Flip this card, scan the QR code  
and follow the instructions.



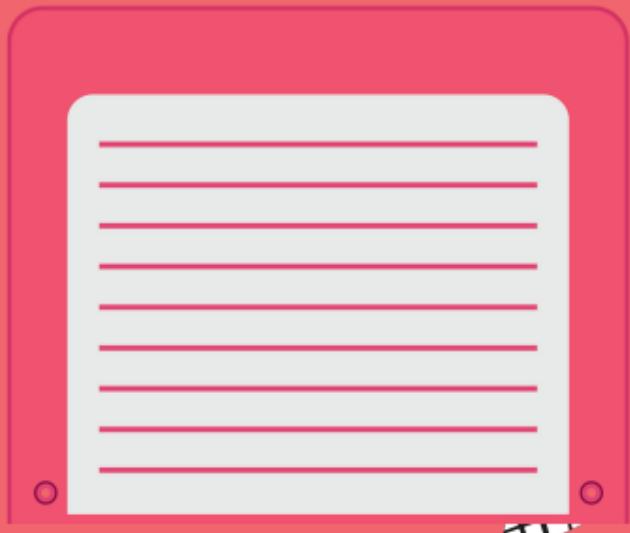
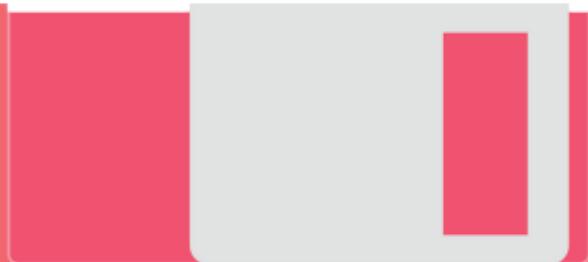


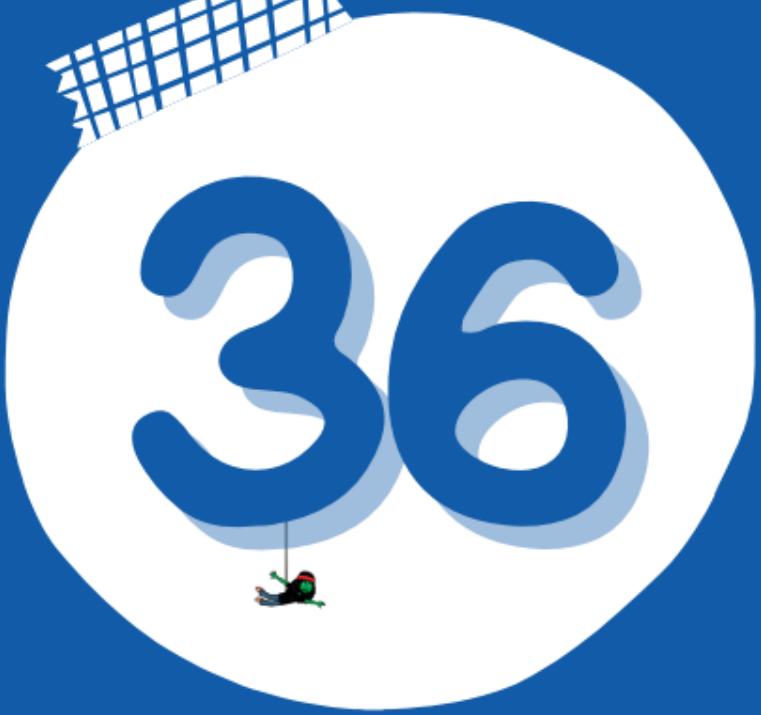
XTREM TDD

T



A very old **floppy disk...**





XTREM TDD

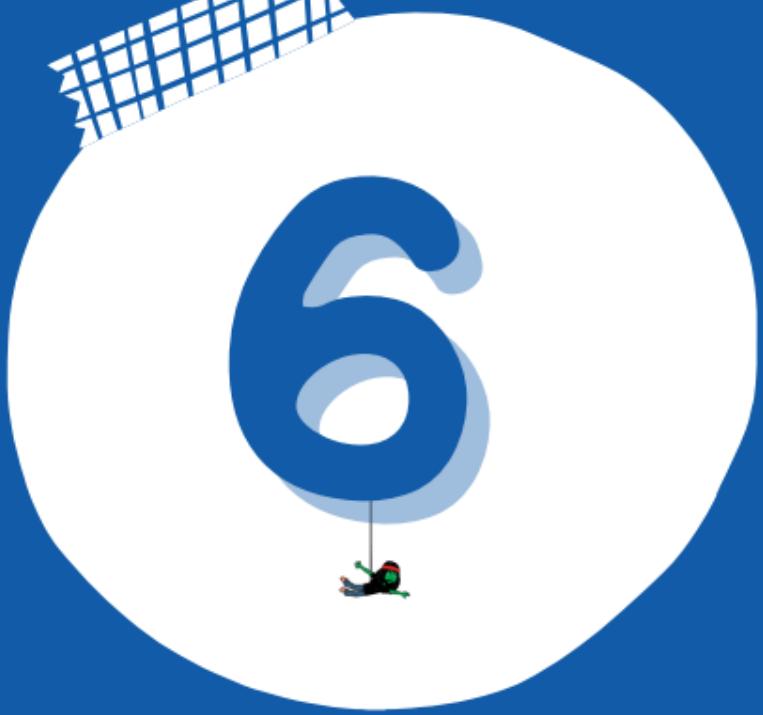


34/62



36





XTREM TDD



35/62

6



42

A tree could be very helpful  
to fight primitive obsession.





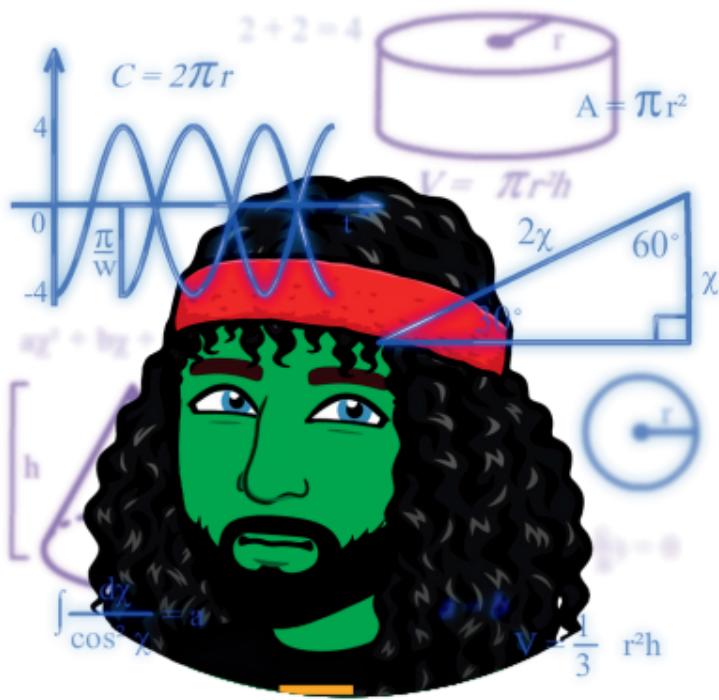
XTREM TDD



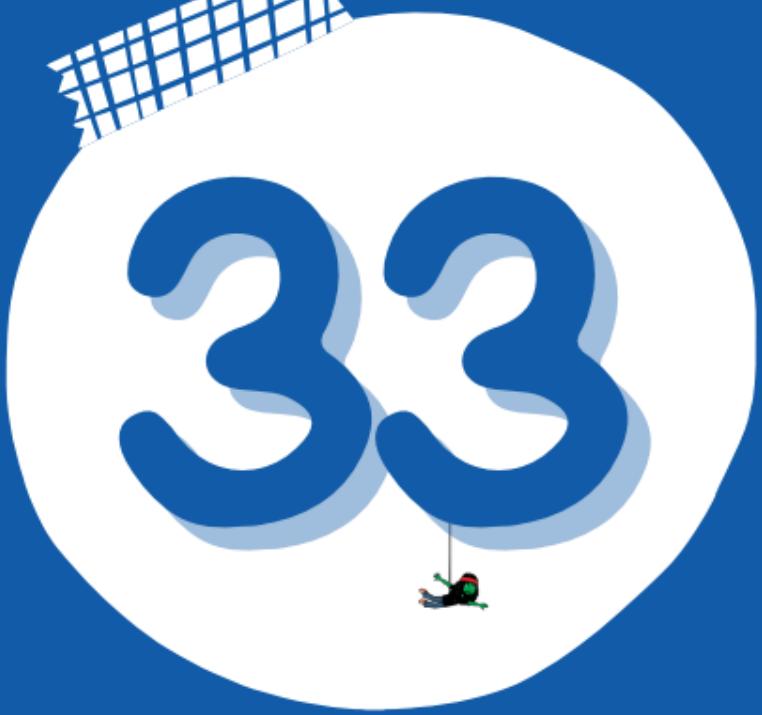


## Avoid exceptions

What are **errors** and **exceptions** in the end?



Avoiding exceptions help you make your code more explicit - **no more lies...**

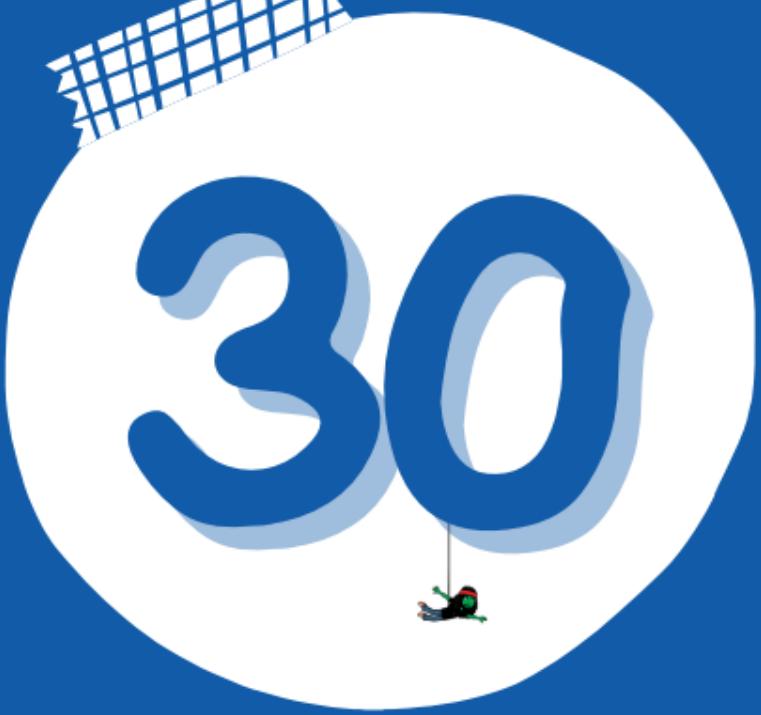


XTREM TDD

33



... is a monadic container type which  
represents a value of two possible types.



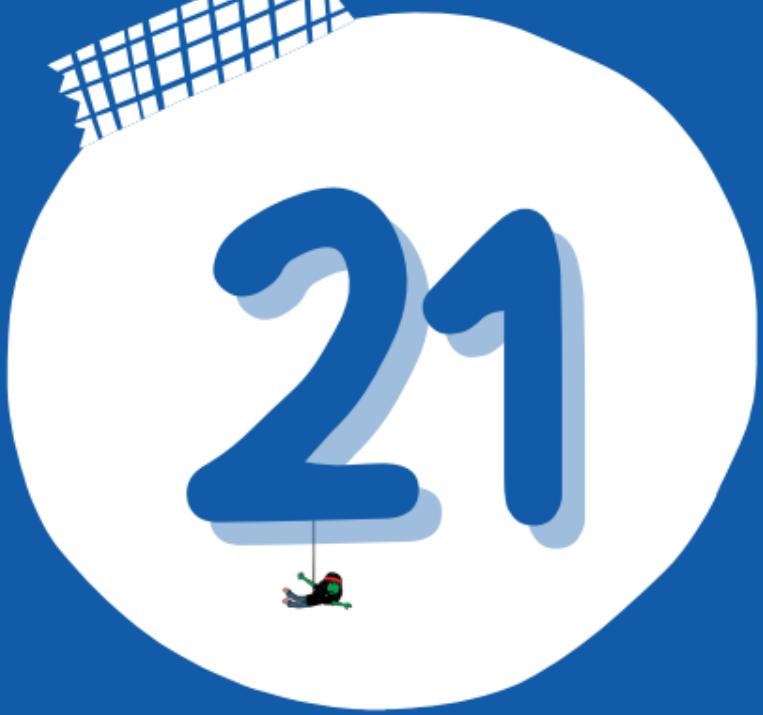
XTREM TDD

30



A lot of discussions,  
learnings and fun.





XTREM TDD



39/62



Let's mutate some code

What did you **learn**?  
How/when can it be **useful** ?



Tests are code too, proceed with them with the **same care** that for your production code.



XTREM TDD



14



An old rag...





XTREM TDD



25



You are not allowed to use  
"for loop" anymore...



To continue your journey,  
this is happening right behind me...





XTREM TDD



42/62



37



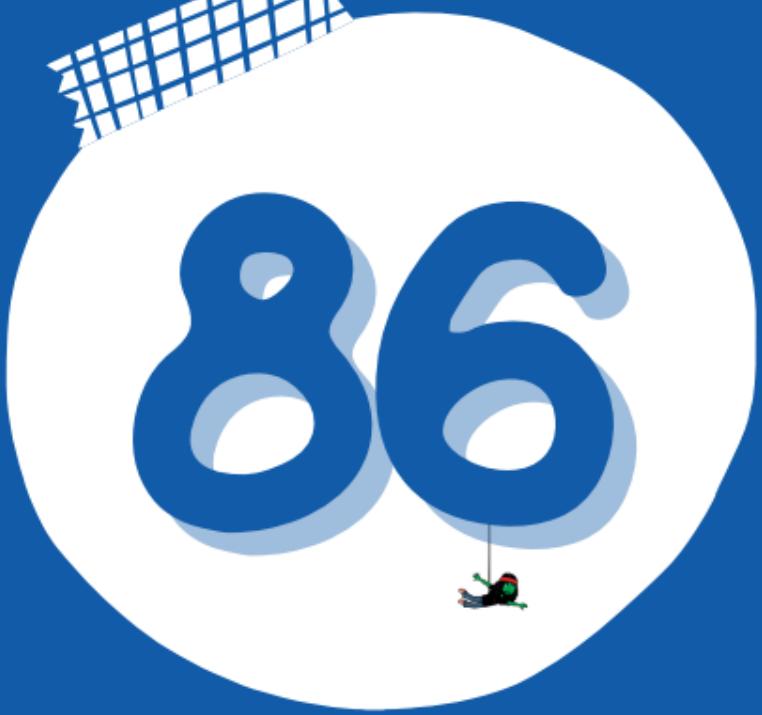
You have successfully made your code immutable and learned its benefits...



It is now time to **play another game**.

91





XTREM TDD

85



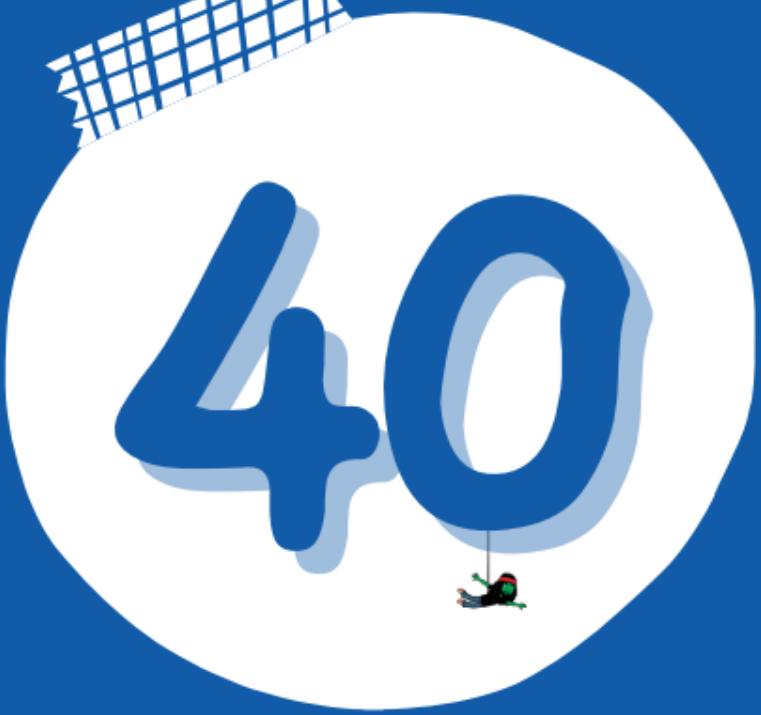
Congratulations, you have  
successfully removed exceptions.



There should be a way to  
improve your code...

64





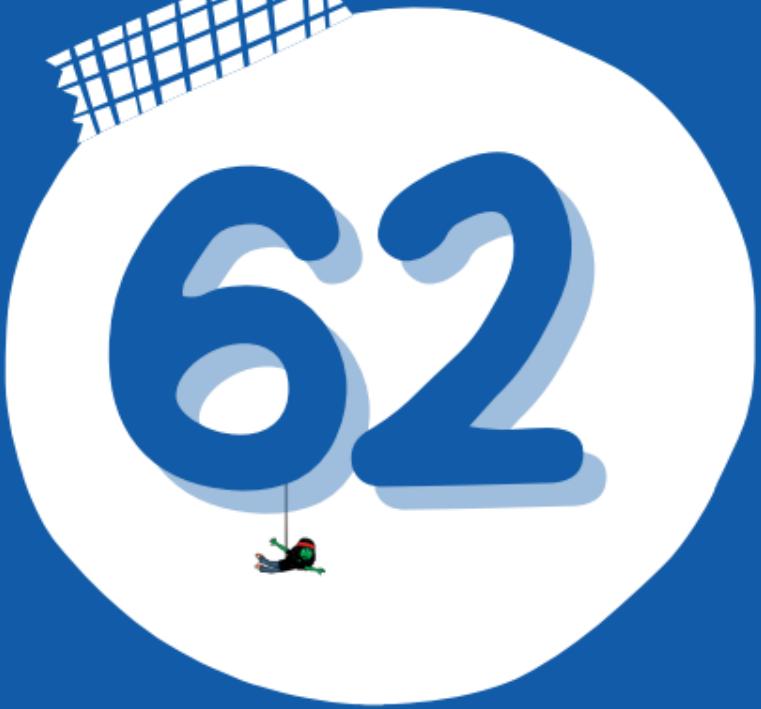
XTREM TDD

40



Either

3



XTREM TDD



45/62



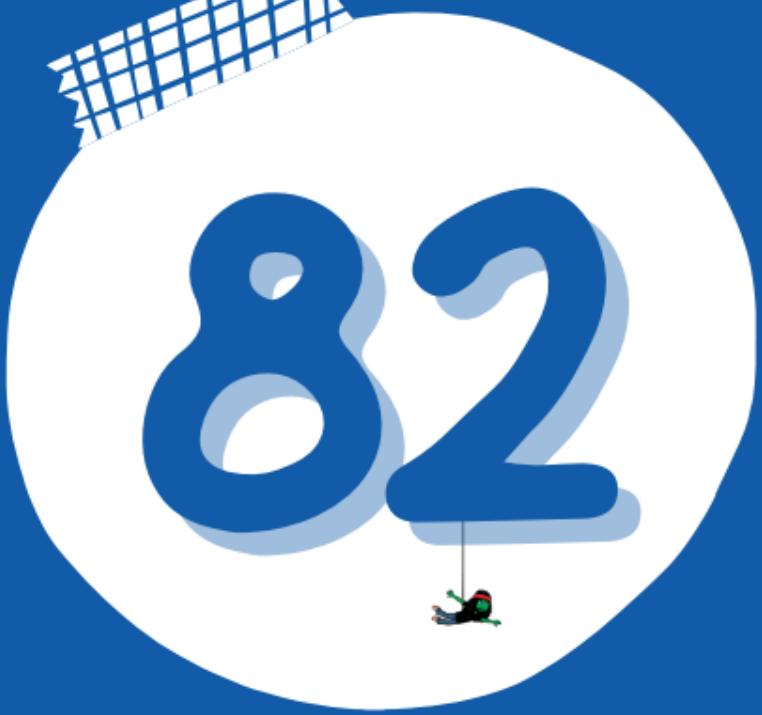


## Only Immutable Types

What has been the **impact**  
on the code?



💡 Immutability makes your code  
**more predictable.**



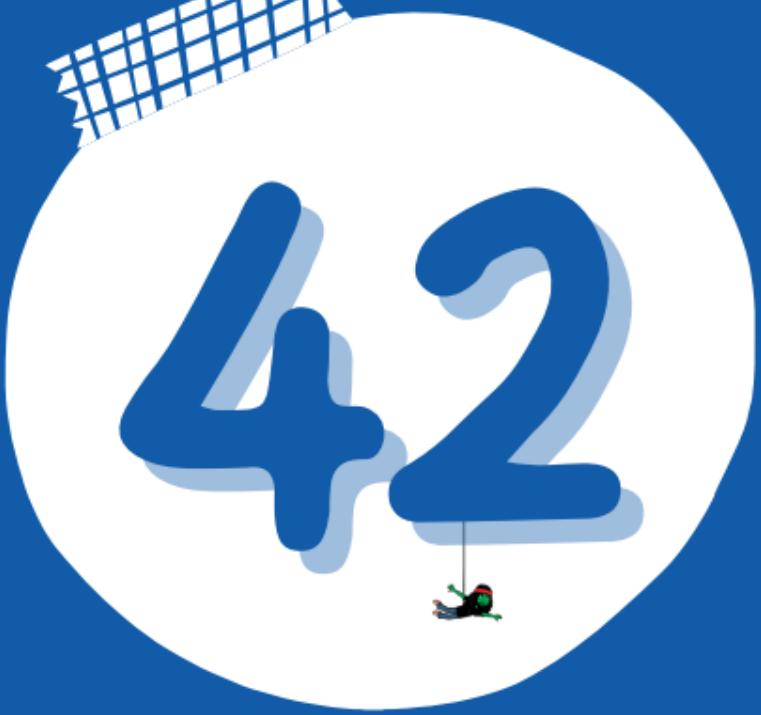
XTREM TDD

82



B + R - G





XTREM TDD



47/62

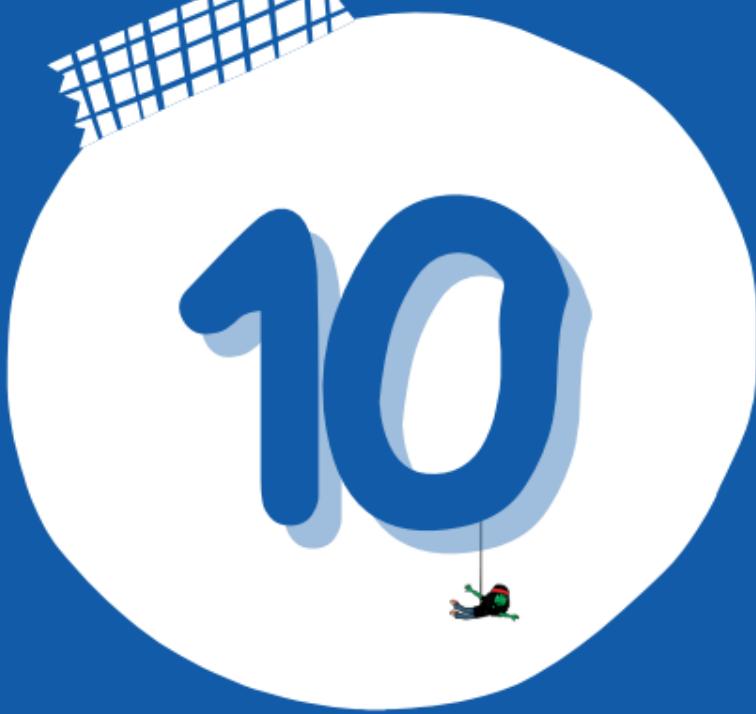


Fight primitive obsession

What do you think about the  
Strangler Pattern?  
When could you use it?



Who said TDD can not be  
used on existing code...



XTREM TDD

10



Red

30

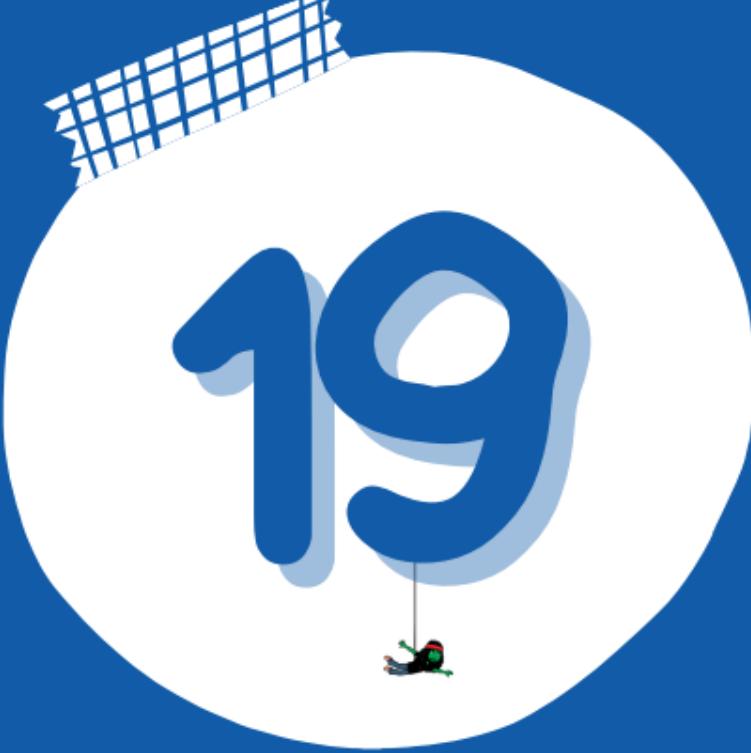
Blue

22

Green

15





XTREM TDD



19





11

XTREM TDD

11



Try

2



XTREM TDD

50

64

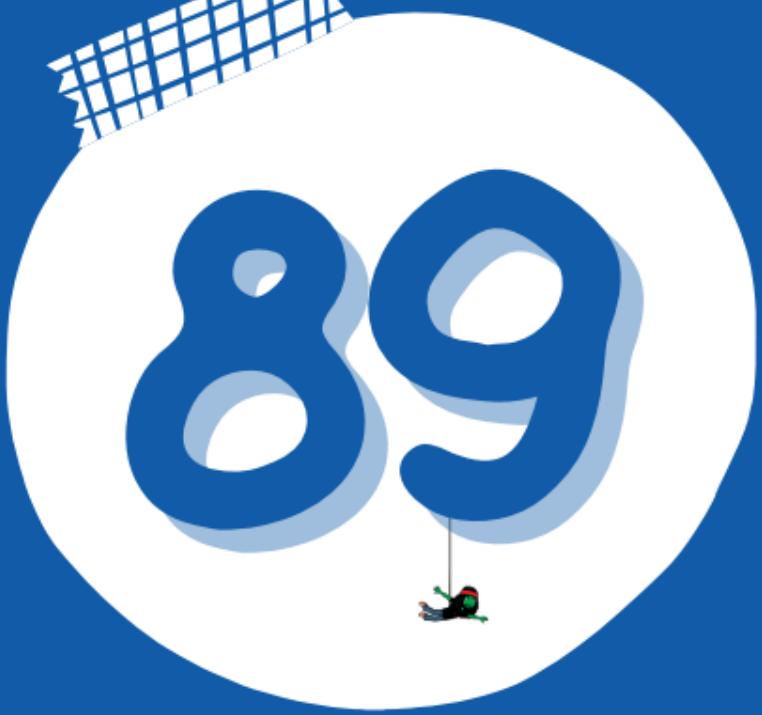


Now that monadic containers  
have no secrets for you,



let's move on...





XTREM TDD

89

30

59

## Congratulations!!!

Let's take a few minutes to make  
a learning review...

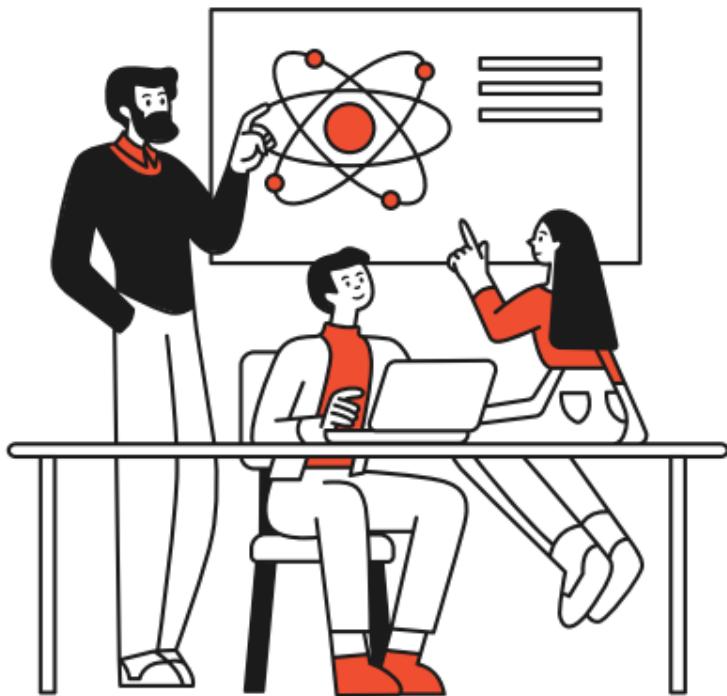


We hope you learned and enjoyed it...  
**#sharingiscaring**

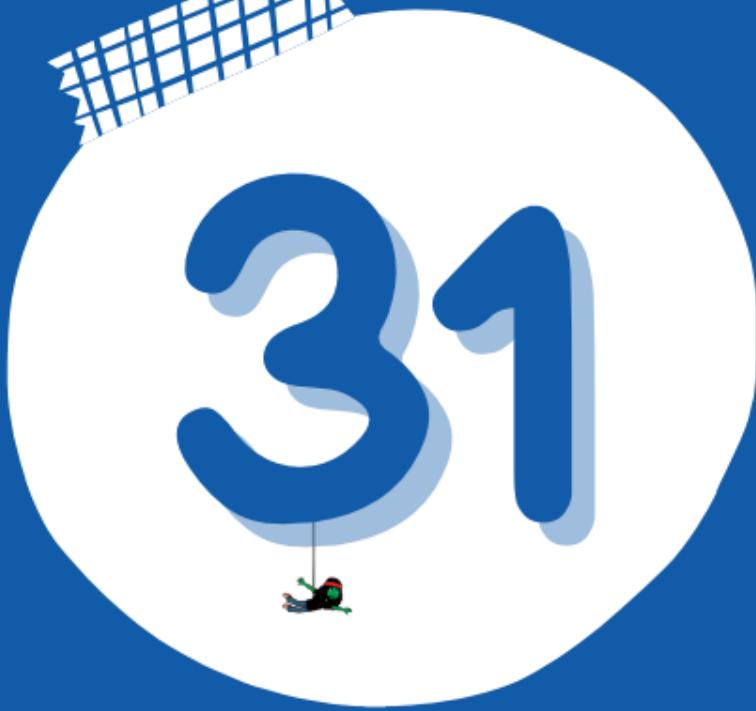


XTREM TDD

59



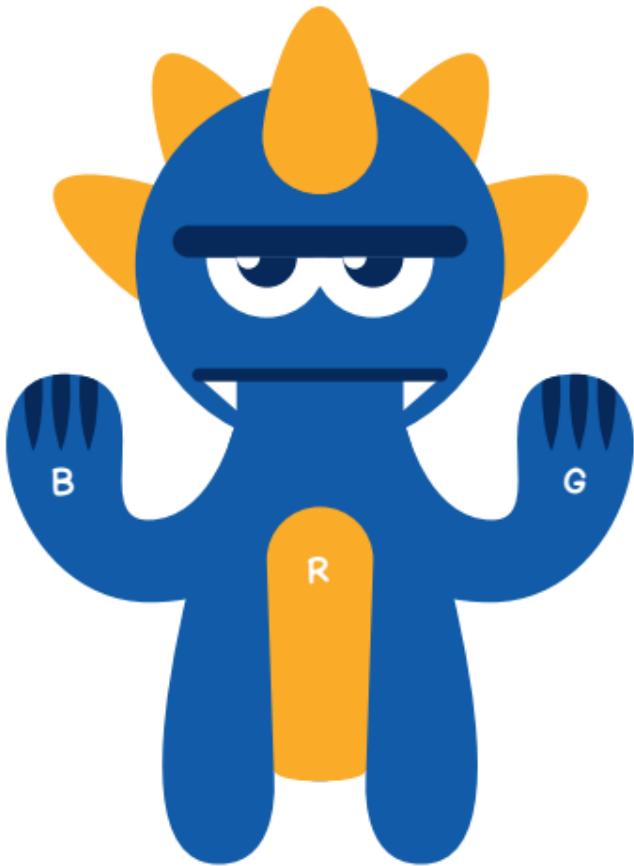
Awesome attendees!!!



XTREM TDD

31

15



Here is a  
killed mutant...





XTREM TDD

54



A very dirty painting...





XTREM TDD



41



? + ? - ?



XTREM TDD



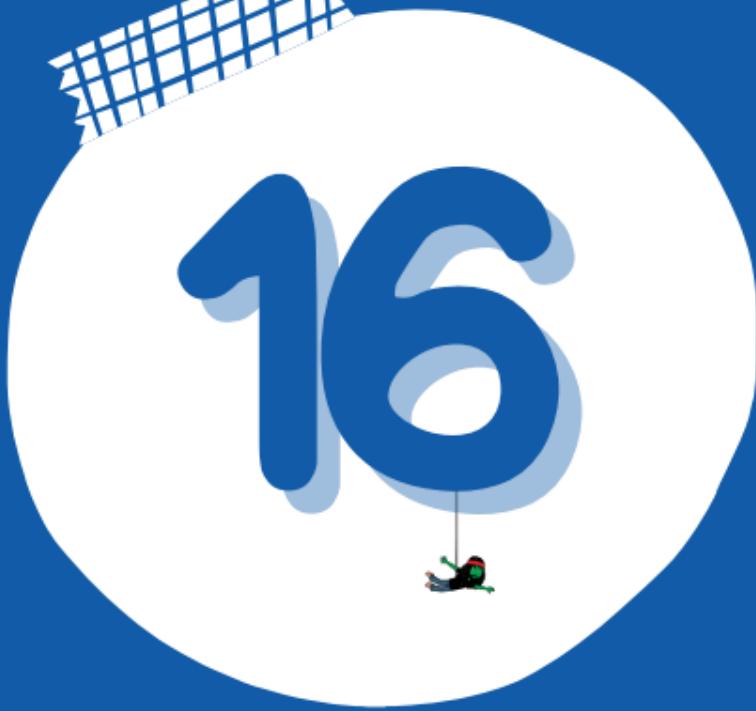
31



LET'S PLAY A  
DIFFERENT GAME!!!

19





XTREM TDD



58/62

16

50



One upon a time, some bugs  
appeared because of a  
**bad domain understanding...**



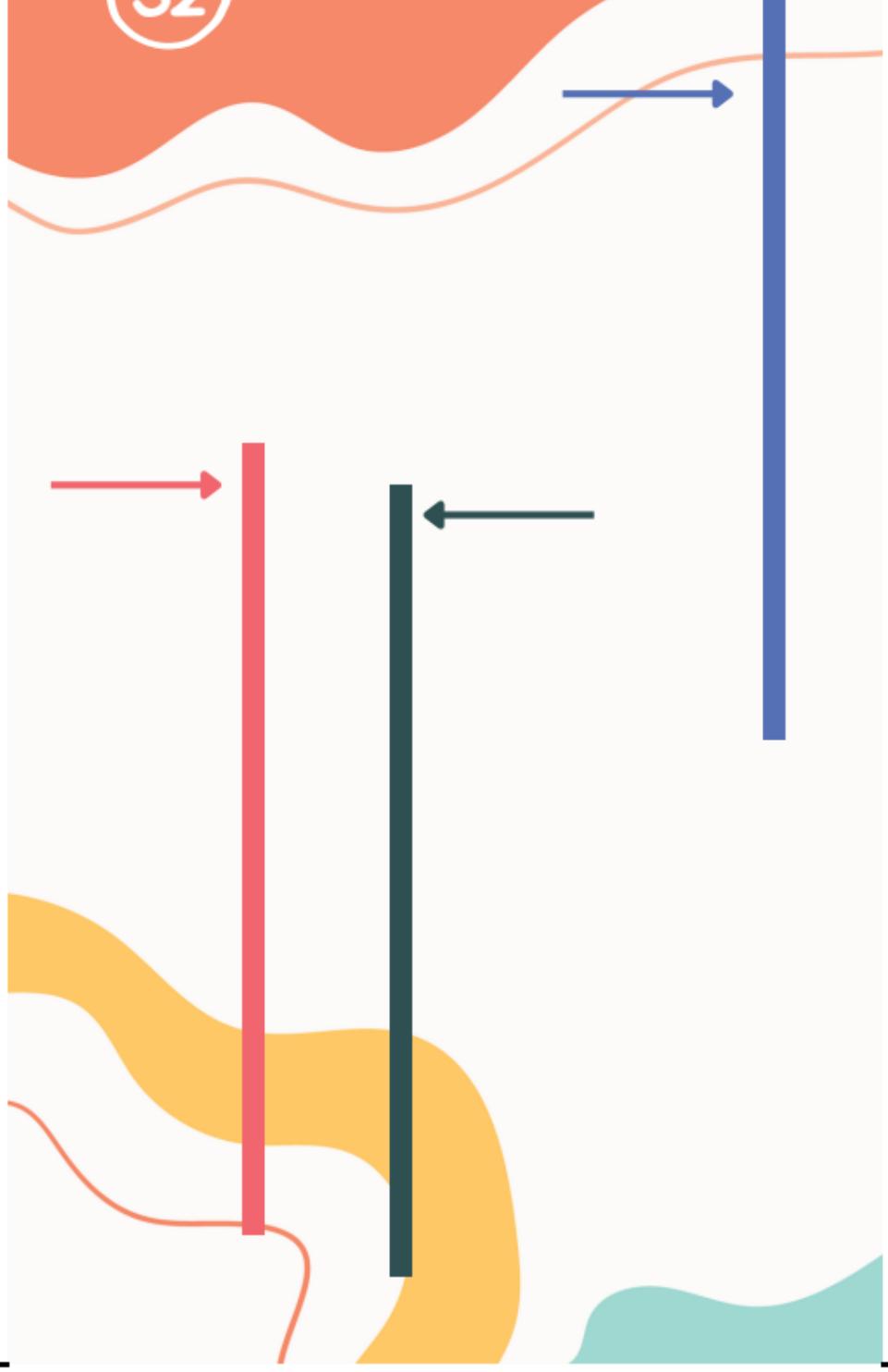
Let's challenge our implementation  
with **Property-Based Testing**.

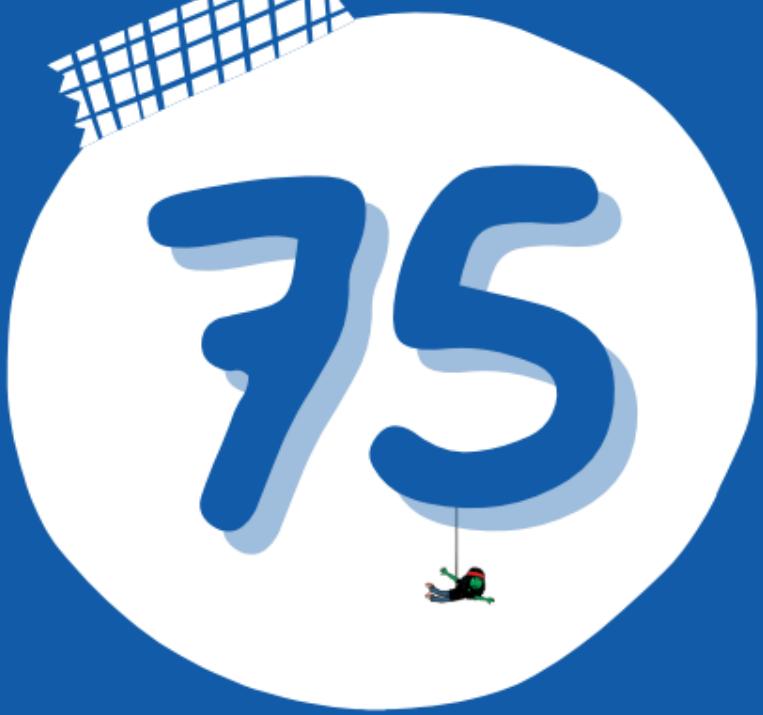




XTREM TDD

32





XTREM TDD



60/62

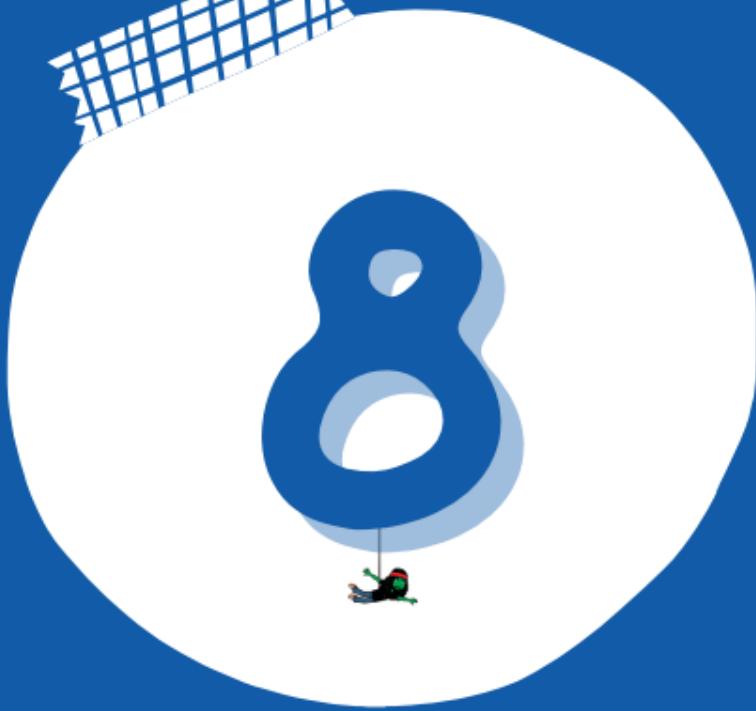


## Use existing monad

What do you **think** about it?

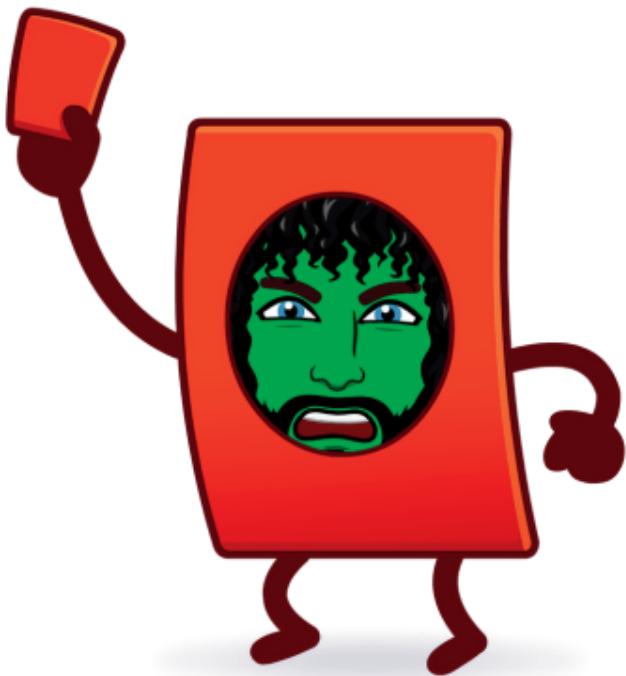
What could you **do** with it?





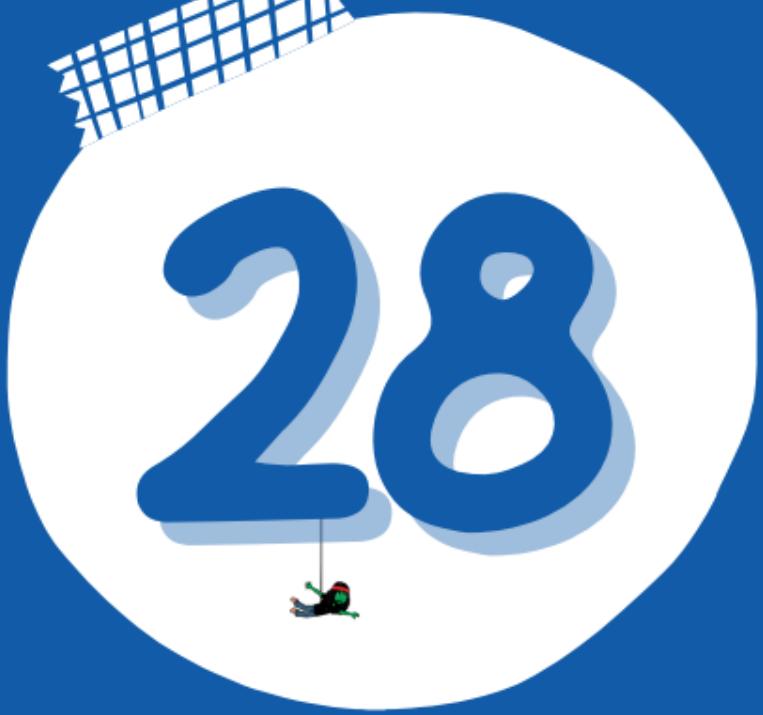
XTREM TDD

8



Sometimes it is not that easy...





XTREM TDD

28



Nothing interesting here...