Increase your capacity
to deliver more value

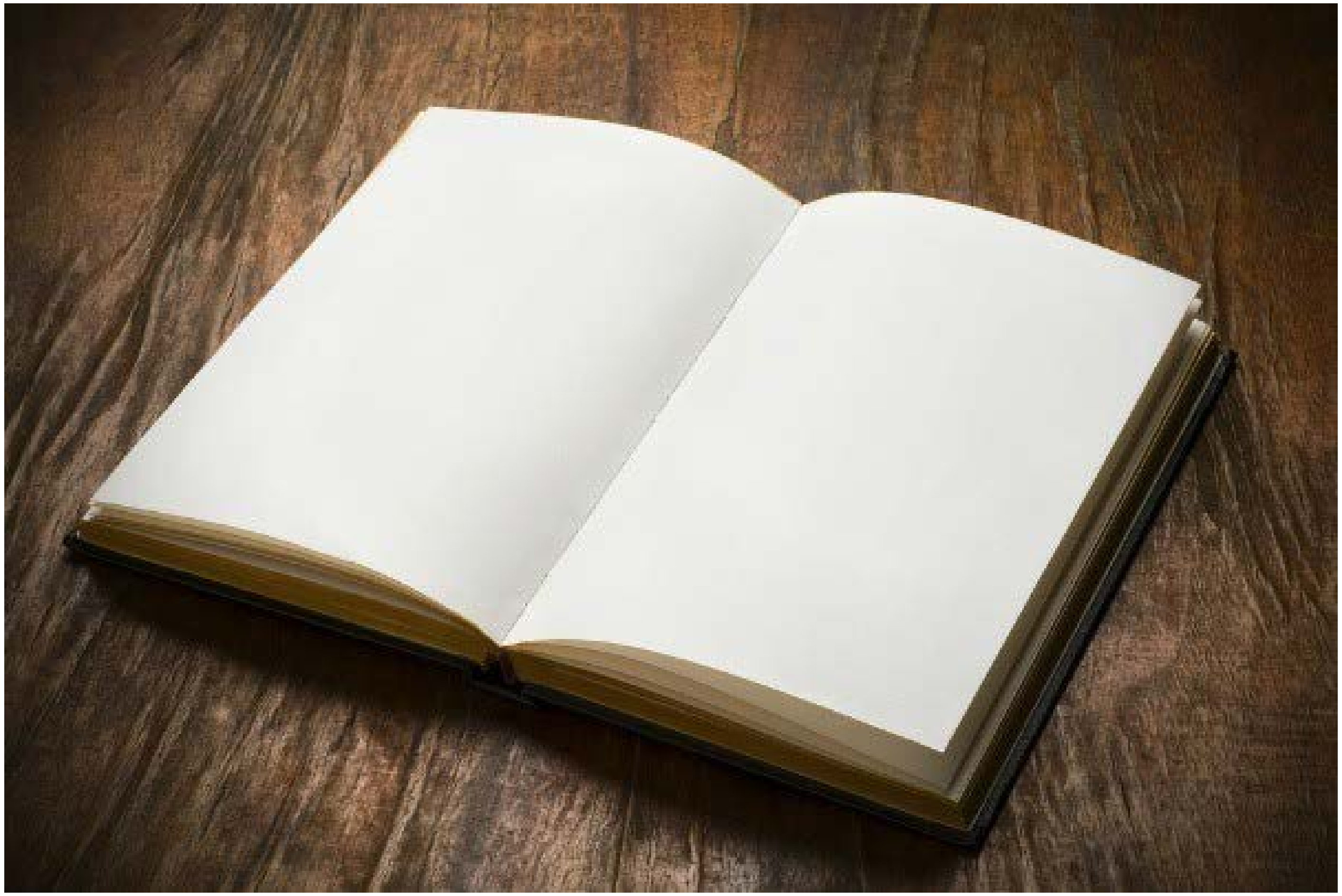
Code Review

Better to find an error twice
than never to find it at all

Aims at creating a system that is easy
to maintain and extend over time

42skillz

WHY REVIEW?



WHY CODE REVIEW?

- **To err is human**
- Lots of **errors escape the originator** more easily than anyone else
- Review are **educational**

TO ENABLE …

- Visibility into **state of project**
- **Opportunities** for personnel **to discuss topics related to project**
- **Intermediate milestones** and sense of progress
- **Assessment** of technical adequacy of project

USE THE POWER OF A TEAM TO …

- Point out **needed improvements** of a product
- Confirm the parts in **which improvement is not needed or desired**
- Make technical work more uniform and **predictable in quality**, which makes it more manageable

42skillz

REVIEW PATH

- **Unit test**
  - ✓**First properties** are applied
  - ✓Check **Code Smells**
  - ✓**Coding Guidelines**

- **Business code**
  - ✓**Clean Code** rules are applied
  - ✓Apply **SOLID Principles** are applied
  - ✓Check **Code Smells**
  - ✓**Coding Guidelines**

COLLECTIVE ORGANIZATION

- Participants
  - Author, Reviewer, Facilitator, Developers
- That occurs
  - **Periodically**
  - In response to **a particular condition**
- Should be scheduled before a release
- **Must be prepared** by the reviewer
- Should be preferred **in large organization**

PAIR ORGANIZATION

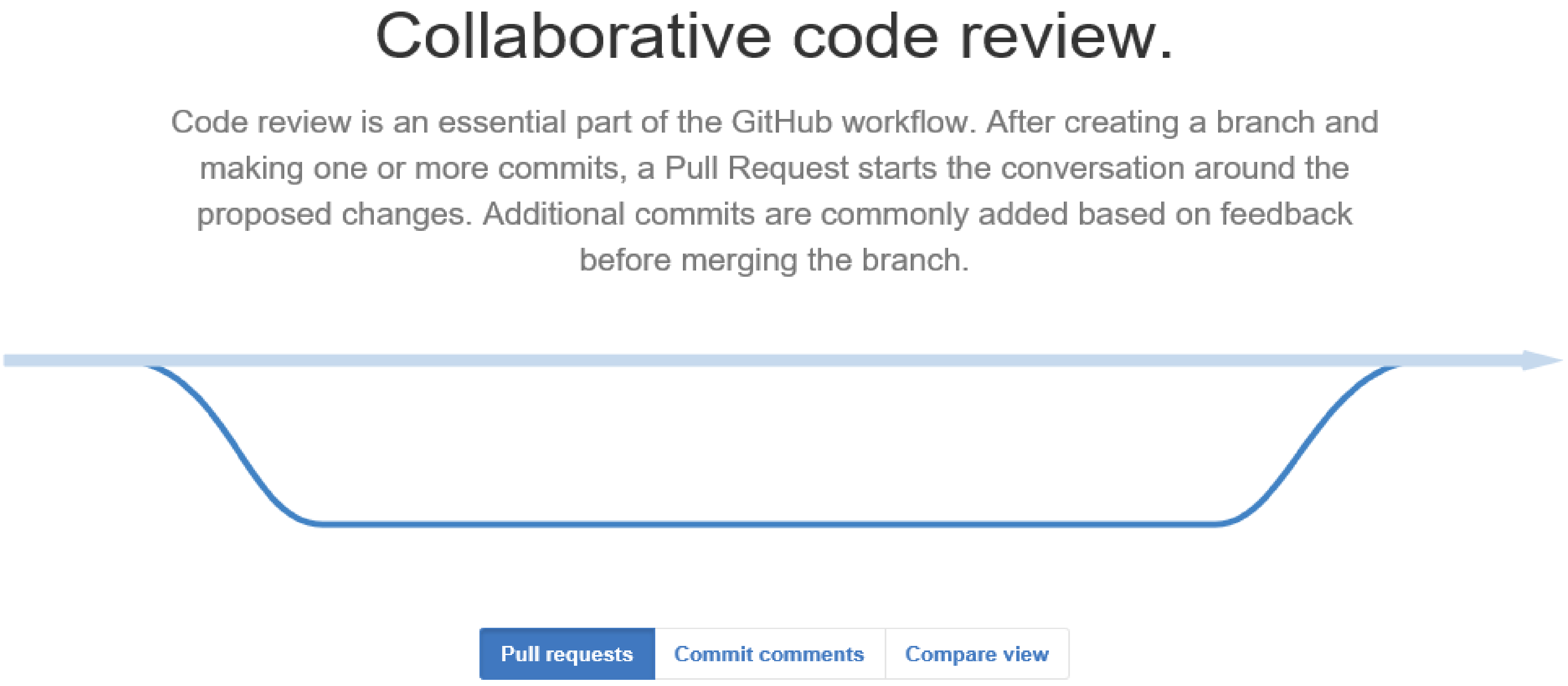
- Participants
  - Author, Reviewer
- That occurs
  - On a particular date
    - **After a tricky development**
  - In response to a particular condition
    - **At the end of user story**
- **Must be prepared** by the reviewer
- Should be preferred with a **small team**

CODE OF CONDUCT

- **Speak about code** not about people
  - The goal is to improve code
  - See egoless programming https://blog.codinghorror.com/the-ten-commandments-of-egoless-programming/
- Use **sustainable time frame**
  - Don’t spent to much time!
  - Preferred 300 LOC / Per hour

42skillz

TOOLING


Collaborative code review.

Code review is an essential part of the GitHub workflow. After creating a branch and making one or more commits, a Pull Request starts the conversation around the proposed changes. Additional commits are commonly added based on feedback before merging the branch.

42skillz

BENEFITS

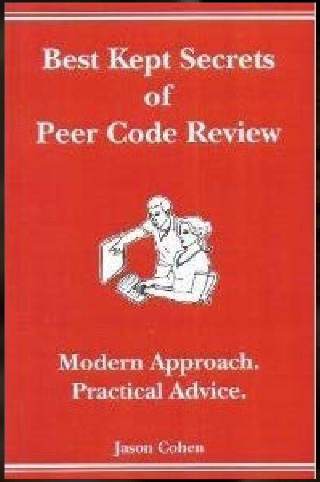
- **10X reduction in errors in products**
  - Collective organization is the more efficient and expensive
  - Pairing organization is pragmatic and cheaper
- **50-80% cost reduction!**
  - Reduce the waste to find the bugs
  - People more informed make fewer errors

42skillz

TO SUM-UP

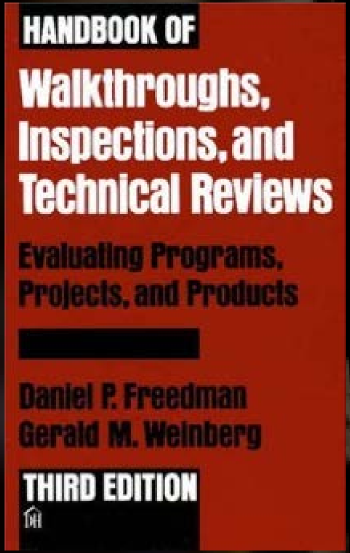
- Code Review is an effective practice to **reduce errors and cost reduction**
- But it is also a perfect way **to share information** and get **a better project's understanding**
- Pair programming is a friend of **code review**

42skillz



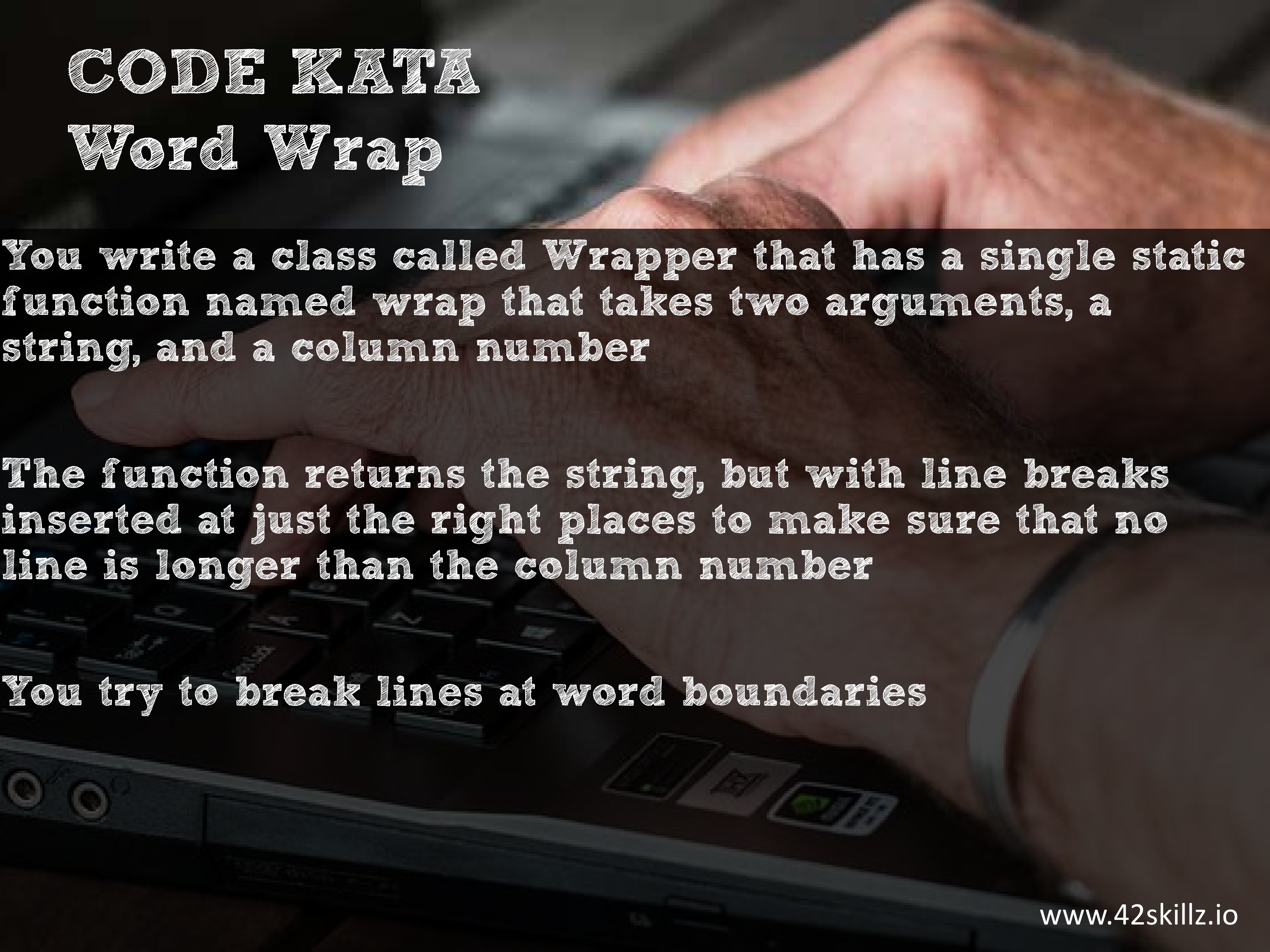
Best Kept Secrets of Peer Code Review

Jason Cohen



Handbook of Walkthroughs, Inspections, and Technical Reviews

Daniel P. Freedman
Gerald M. Weinberg

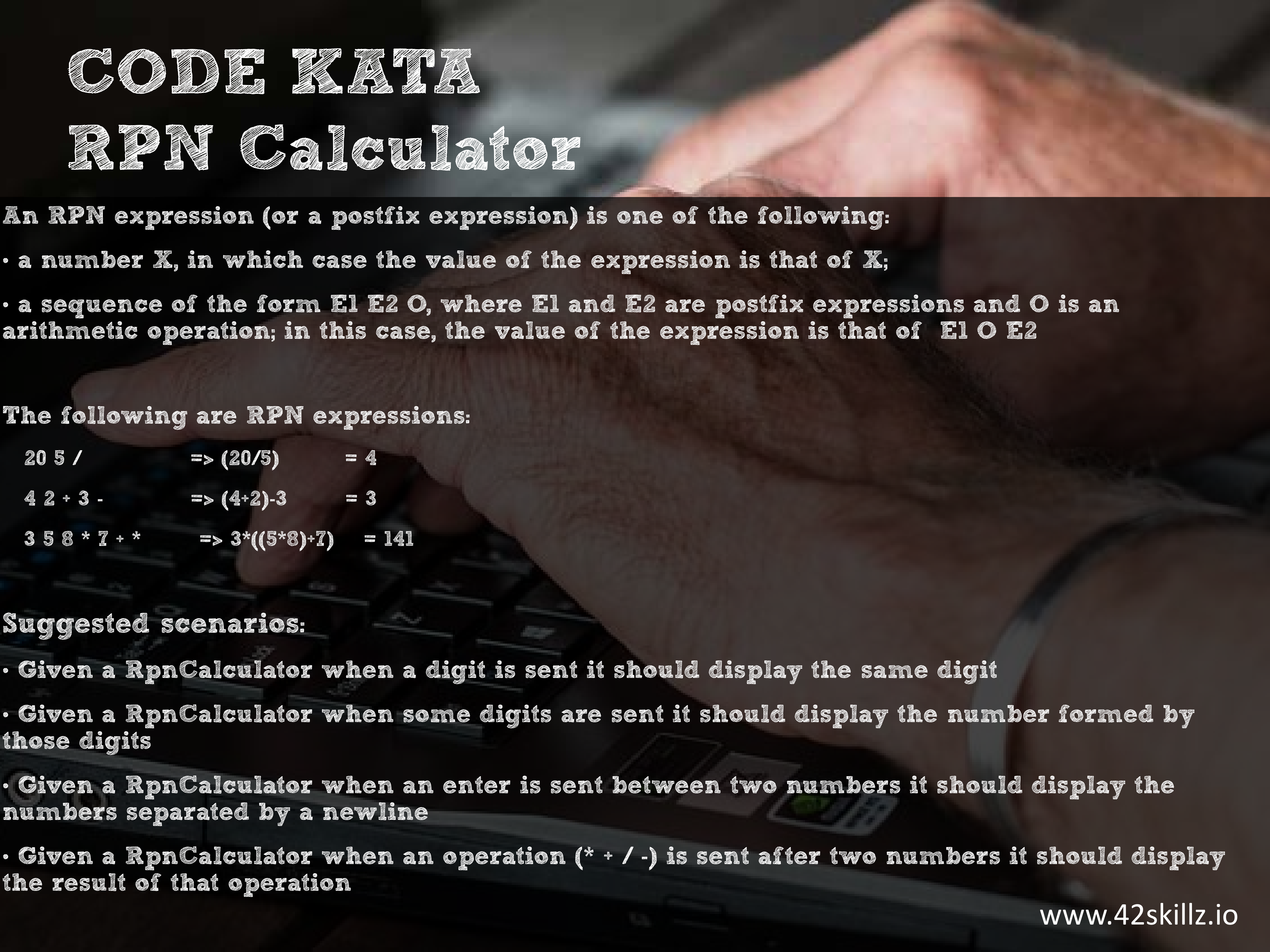


CODE KATA
Word Wrap

You write a class called Wrapper that has a single static function named wrap that takes two arguments, a string, and a column number

The function returns the string, but with line breaks inserted at just the right places to make sure that no line is longer than the column number

You try to break lines at word boundaries



CODE KATA RPN Calculator

An RPN expression (or a postfix expression) is one of the following:

· a number X, in which case the value of the expression is that of X;

· a sequence of the form E1 E2 O, where E1 and E2 are postfix expressions and O is an arithmetic operation; in this case, the value of the expression is that of E1 O E2

The following are RPN expressions:

```
20 5 /              => (20/5)        = 4
4 2 + 3 -           => (4+2)-3       = 3
3 5 8 * 7 + *       => 3*((5*8)+7)   = 141
```

Suggested scenarios:

· Given a RpnCalculator when a digit is sent it should display the same digit

· Given a RpnCalculator when some digits are sent it should display the number formed by those digits

· Given a RpnCalculator when an enter is sent between two numbers it should display the numbers separated by a newline

· Given a RpnCalculator when an operation (* + / -) is sent after two numbers it should display the result of that operation