

This project wants to teach people the importance of clean code, by letting them experiment with refactoring the code to be more obfuscated.

## 1.0 Rules

---

- Run the tests all the time
- Tests have to be green all the time
- Do not modify the test
- The tests are complete ==> if the tests are green, the code is still correct
- Java: run tests in your IDE- or command-line `./mvnw clean test`
- Javascript: run tests in your IDE - or command-line: `npm install npm run test`

### 1.1 Possible Obfuscating steps

- Add impossible if
- Add impossible else
- Invert if
- Eliminate return in if
- Add unused variable
- Add unused variable so that that it is used
- Make scope for variable larger
- Promote variable to field
- Promote variable to static
- Promote function to static
- Make scope for function larger
- Add bi-directional changes in var
- Use several vars for the same result
- Make var mutable
- Make condition longer
- Use field ISO parameter
- Use parameter ISO existing field
- Add confusing comments
- Add comments in a different speaking language
- Change function names in a different language
- Inline Function
- Extract Function
- Remove Final
- Add return statements all over the place
- Self-implement language functions
- Split declaration and assignment
- Add dead code
- Add dead code in different software language
- Rename
- Rename singular/plural
- Rename to meaningless name
- Rename to other meaning
- Bad indentation (should be the last change)

### 1.2 A succesful sequence of obfuscating steps:

A good sequence that is almost always succesful to **create unreadable code**.

- Extract variable
- Rename variable to meaningless name
- Remove final
- Split into declaration and assignment
- Move declaration up a few blocks (Make scope for variable larger)
- Move assignment up a few blocks

### Links

- [self-register](#) in the github organization
- [this github organization](#)

- [Join the mural](#)
- [teams meeting](#) (for screen sharing)

## 1.4 if you want to read more

- <https://refactoring.guru/refactoring/smells>
- <https://refactoring.guru/refactoring>
- [https://en.wikipedia.org/wiki/Code\\_smell](https://en.wikipedia.org/wiki/Code_smell)
- <https://www.sihui.io/code-smell-cheat-sheet>
- <https://blog.codinghorror.com/code-smells/>
- <https://refactoring.guru/refactoring/techniques>
- <https://refactoring.guru/refactoring/catalog>
- <https://www.kentbeck.com/>
- <https://refactoring.com/>
- Workflow of refactoring <https://www.youtube.com/watch?v=vqEg37e4Mkw>