



REFACTORING

with Webstorm 2016.1

Paweł Kaczmarek

To refactor... or not to refactor...

- Code smells
 - Design Patterns
 - SOLID
 - * Tests
-

SOLID

- S - Single responsibility principle
 - O - Open / Close principle
 - L - Liskov substitution principle
 - I - Interface segregation principle
 - D - Dependency inversion (injection) principle
-

Code Smells

"A **code smell** is a surface indication that usually corresponds to a deeper problem in the system".

Martin Fowler

Refactoring - techniki

<https://sourcemaking.com/refactoring>

CS - Bloaters

- Za długa funkcja
- Za dużo parametrów w funkcji
- Za duża klasa
- Mnożenie tych samych zmiennych w różnych częściach aplikacji
- Mnożenie zmiennych trzymających stałe wartości

CS - Object Oriented Abusers

- Skomplikowane instrukcje switch / wiele następujących po sobie instrukcji if
- Tymczasowe zmienne, które tylko czasami trzymają wartość
- Więcej niż jedna klasa realizująca wskazaną funkcjonalność

CS - Change Preventers

- Kod odpowiedzialny za jedną funkcjonalność znajduje się w wielu miejscach
- Bardzo ściśle powiązanie ze sobą metod w obrębie jednej klasy (obiektu)

CS - Dispensables

- Niepotrzebne komentarze
- Duplikaty kodu
- Zbędne klasy
- "Martwy" kod
- Kod "na później"

CS - Couplers

- Metoda z obiektu częściej odwołuje się do innych obiektów niż do swojego rodzica
- Używanie prywatnych pól obiektów poza tymi obiektami.
- Klasa istnieje tylko po to, żeby uruchamiać metody z innej klasy.

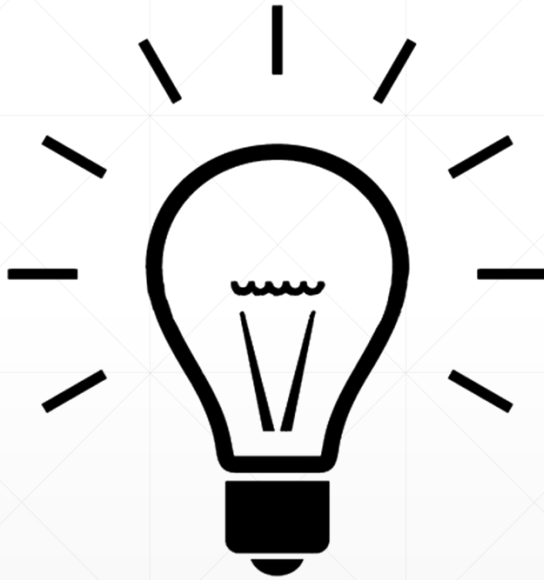
CS - copy/paste



"If I am copying and pasting code, I am doing something wrong".

Every Good Programmer

Example - live refactoring



source: <http://elijahmanor.com/talks/js-smells>

Webstorm - Refactor...

- ctrl + alt + shift + T
 - Menu -> Refactor
 - Right click -> Refactor
-

Webstorm - Find usages (alt + F7, ctrl + alt + F7)

- Wyszukiwanie



Webstorm - Working with tool windows

- alt + 1 - project files
 - alt + 3 - search results
 - alt + 6 - ToDos
 - alt + 9 - vcs (git) toolbox

 - press alt and press and hold it again - help shows up
-

Webstorm - Rename (shift + F6)

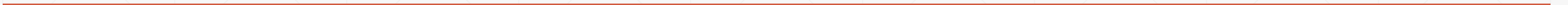
- variable
 - function name
 - method name
 - file name
 - html tags, ids, style name etc.
-

Webstorm - Safe delete (alt + del)

- variable
 - function
 - method
 - class
-

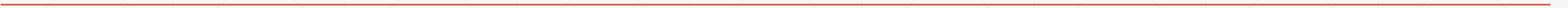
Webstorm - Extracting variable

- ctrl + alt + V
- Create a new variable and use the selected expression as its value



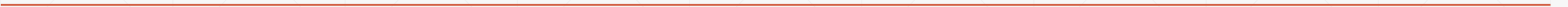
Webstorm - Extracting parameter

- ctrl + alt + P
- Turn the selected expression into a new method parameter



Webstorm - Extracting method

- ctrl + alt + M
- Turn the selected code fragment into a method / function



<https://www.jetbrains.com/help/webstorm/2016.1/refactoring.html>

Linting

" **Linting** is the process of running a program that will analyse code for potential errors. "

~StackOverflow

Linting tools

- JSLint
 - JSHint
 - * JSCS (merged with ESLint)
 - **ESLint**
 - JSCPD
-

ESLint

- + konfigurowalny
 - + czytelne komunikaty
 - + wsparcie dla ES6
 - + możliwość podpięcia customowych narzędzi raportujących

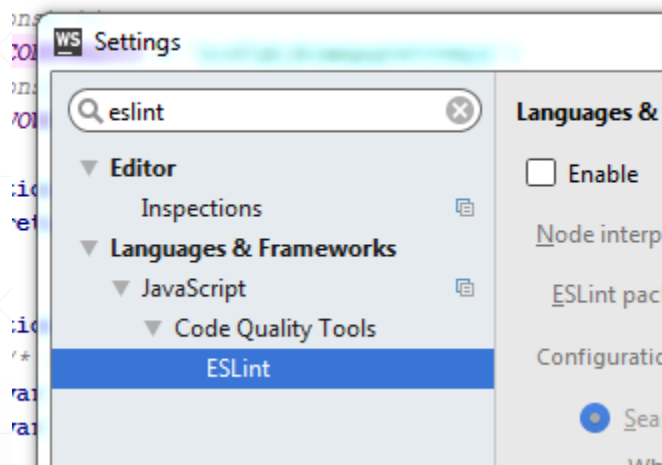
 - - wymaga początkowej konfiguracji
-

ESLint

<http://eslint.org/>

```
npm install -g eslint
```

ESLint + Webstorm



Resources

- <http://elijahmanor.com/talks/js-smells>
 - <http://mikamantyla.eu/BadCodeSmellsTaxonomy.html>
 - <https://sourcemaking.com/refactoring>
-