

tt()

Refactoring

(Shaping your code)

Refactoring

In **computer programming** and **software design**, **code refactoring** is the process of restructuring existing **computer code** (...) without changing its external behavior. Refactoring is intended to improve the design, structure, and/or implementation of the **software** (...), while preserving its **functionality**. Potential advantages of refactoring may include improved code **readability** and reduced **complexity**; these can improve the **source code's maintainability**.

Bron: https://en.wikipedia.org/wiki/Code_refactoring

Refactoring

why?

- code vereenvoudigen
- leesbaarheid verbeteren
- onderhoudbaarheid verbeteren

Bron: <https://nl.wikipedia.org/wiki/Refactoren>



Refactoring

- code vereenvoudigen

DRY, KISS,

Vermijd spaghetti code

lange functies vermijden / uit elkaar halen

Componetization (break code down into reusable semantic units)

Refactoring

- leesbaarheid verbeteren

Naamgeving variabelen (wat bevat de variabele?)

Naamgeving functies (wat doet de functie?)

Overweeg het gebruik van hulpfuncties

Refactoring

- onderhoudbaarheid verbeteren

Conventies (als zij er zijn)

Kan je je eigen code volgend jaar nog lezen?

Comments, in-line omschrijving verantwoordelijkheid functies

Refactoring

- Het proces documenteren

In de wiki

Wat heb je gedaan om de kwaliteit van de code te verbeteren?

Nog meer pasta
(Een voorbeeld)

Beoordeling

Beoordeling

- Vrijdag a.s., 1 december
- Mondeling (30')
- Inschrijving
- Criteria



**Uncaught SyntaxError
Unexpected end of input**