

# Transformation Priority Premise

Daum Cafe  
Mikael



# 발표목차

- Transformation의 정의
- Refactoring의 정의
- Transformation in RGB
- Transformation의 방향성
- 예제



# Transformation

- Refactoring의 Counterpart
- Refactoring
  - 행위의 변경 없이 구조를 변경
- Transformation
  - 구조의 변경 없이 행위를 변경
- 좋은 알고리즘을 얻기 위한 우선 순위 존재



# Refactoring

- A change made to the internal structure of software to make it easier to understand and cheaper to modify without changing its observable behavior.
- ex.
  - rename
  - extract method
  - 주목할 만한 변경 없이



# Transformation

- 구조 변경 없이 행위 변경 가능 ???
- ex
  - addition → subtraction
  - 단일 문장 → while loop
- 주목할 만큼의 구조 변경 없이 행위 변경



# Transformation in RGB

- Transformation
  - 실패하는 테스트: 구조 유지, 행위 성공시키기
- Refactoring
  - 성공하는 테스트: 행위 유지, 구조 개선시키기
- 실패하는 테스트를 성공시키는 규칙
  - As the tests get more specific, the code gets more generic.



# Transformation의 방향성

- 테스트를 성공시키는데 사용됨
- 코드를 general하게 함 — Generalizer
- 구조 변경없이 행위를 일반화



# 예제

- PrimeFactor
  - <https://github.com/msbaek/memo/blob/master/CC-E24-Transformation-Priority-Premise-Part1.md>
- Sort
  - <https://github.com/msbaek/memo/blob/master/CC-E24-Transformation-Priority-Premise-Part2.md>
- Transformation List
  - [https://github.com/msbaek/memo/blob/master/CC\\_E24\\_Transformation\\_List.md](https://github.com/msbaek/memo/blob/master/CC_E24_Transformation_List.md)
- 강의영상: <https://youtu.be/rRm0IuvX-Kc>