



# 코드리뷰

## 목차

### [코드리뷰](#)

#### [코드리뷰란?](#)

#### [코드리뷰의 중요성](#)

#### [HOW TO?](#)

#### [코드리뷰 CHECK POINT](#)

#### [코드리뷰 유형 \[Google\]](#)

#### [코드리뷰 주의사항](#)

#### [정리](#)

#### [참고](#)

## 코드리뷰

---

### 코드리뷰란?

코드리뷰란 한 개발자가 코드를 작성하면 다른 개발자가 정해진 방법으로 피드백을 주고받는 과정

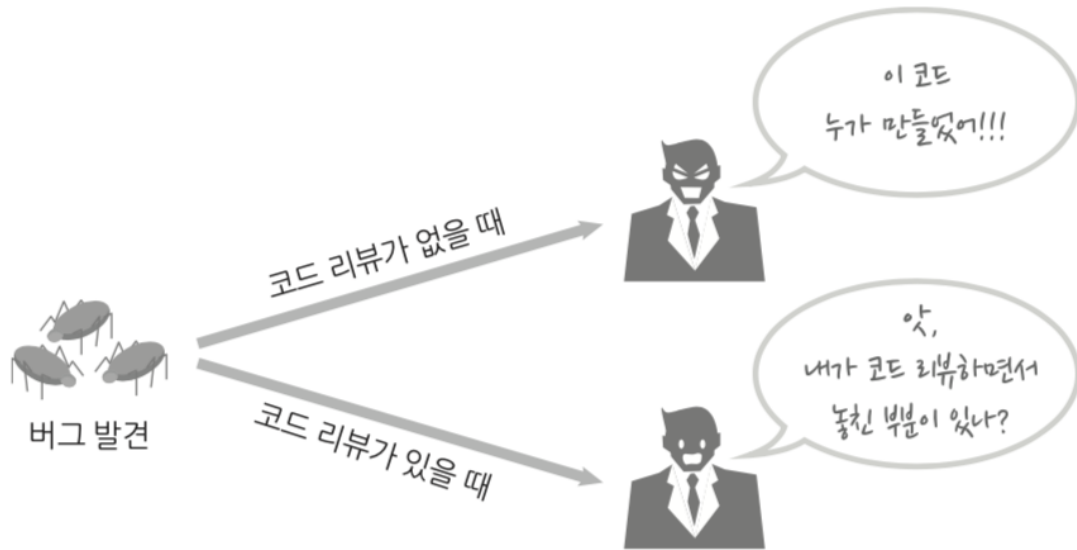
- 작성자(혹은 리뷰어)  
본인이 작성한 코드를 다른 사람에게 리뷰받는 사람
- 리뷰어  
다른 사람이 작성한 코드를 리뷰하는 사람

### 코드리뷰의 중요성

- ▼ 버그를 조기에 발견할 수 있다. → 정확성

“소프트웨어를 유지보수하는 조직에서 코드 한 줄을 변경한다고 했을 때, 코드리뷰가 도입되기 전에는 그러한 변경의 **55%** 정도가 문제를 일으켰다. 그러나 리뷰 과정이 도입된 이후에는 그러한 변경의 **2%** 정도에서만 문제가 발생했다.” - *by Steve McConnell (Code Complete)*

- ▼ 개발 표준 준수 → 이해 용이성, 일관성
- ▼ 지식 공유 → 성장
- ▼ 심리적 문화적 → 조직 공동의 협업물, 주인의식



## HOW TO?

1. 공식적 코드리뷰 [formal code review]
2. 가벼운 코드리뷰 [lightweight code review]

## 코드리뷰 CHECK POINT

1. 코드의 정확성 \_ 원점회기전략
  - 원점회기전략?
    - ▼ 문제를 가능한 일찍 찾아 해결함으로써 결함으로 인한 비용을 최소로 낮추는 전략
    - ▼
  - 기능의 정상 동작 여부 및 효율성
  - 설계의 적절성
  - 버그 발견 여부
2. 개발 표준의 준수 여부
  - 코드의 이해 용이성
    - 코드가 의도한 일을 정확히 수행하면서, 이해하기도 쉽다.
  - 코드의 일관성 [가독성 → 유지보수성]
    - 개발 표준 준수는 일관성과 단순함을통해 일관된 관리가 가능하다.

## 코드리뷰 유형 [Google]

1. 그린필드 코드 리뷰
  - 완전히 새로운 코드 대상
  - 지속 가능성을 보장을 위한 설계, 테스트 확인을 통해 실제 문제 해결에 포커스
2. 동작 변경, 개선, 최적화

- 기존 코드를 수정하는 것.
  - API 수정, 기존 구현 개성, 성능 최적화
3. 버그 수정과 롤백
- 롤백 : 변경이 반영되기 전 상태로 코드를 되돌리도록 구성한 것
4. 리팩터링과 대규모 변경
- 리팩터링(refactoring) : 결과 변경 없이 코드의 구조를 재조정하는 것으로 가독성을 높이고 유지보수를 편하게 한다.

## 코드리뷰 주의사항

1. 리뷰어, 작성자의 마인드
  - 배움의 기회
  - 코드는 협업의 결과물
2. 코드 맥락을 이해할 수 있는 설명 추가!
3. 하나의 이슈 당 하나의 코드리뷰
4. 의견 표현 방식
  - 왜 개선이 필요한지 설명할 것

안 좋은 리뷰

“data 변수 말고 다른 변수명으로 하세요.”  
 “data 변수 말고 grade로 하세요.”

변경에 대한 이유를 설명하지 않고 제안도 애매하게 하는 리뷰입니다. 리뷰이 입장에서는 왜 다른 변수명으로 변경을 해야 하는지 알 방법이 없겠죠.

좋은 리뷰

“data라는 이름은 현재의 자료구조가 무엇인지 그 의도를 알기가 어렵네요.  
 학점 정보를 담고 있는 자료구조 같은데 이와 관련된 변수명을 짓는다면  
 현재 정의한 자료구조가 무엇인지 그 의도를 쉽게 파악할 수 있을 것 같아요.”

- 스스로 고민하고 개선 방법을 선택할 수 있도록 할 것

안 좋은 리뷰

`"arr.filter(item => item % 2 === 0);`으로 사용하세요."

filter의 사용 코드를 알려주는 피드백입니다. 리뷰이 입장에서는 filter의 동작 방식을 제대로 이해하지 못한 채 수정하게 됩니다. 또한 비슷한 코드가 여러 군데 있음에도 불구하고 이곳만 수정할 수도 있습니다.

좋은 리뷰

"자바스크립트에는 배열에는 다양한 내장 메서드들이 존재합니다.  
그중 filter 메서드를 활용해 보세요.  
이 메서드를 활용하면 코드량을 줄일 수 있습니다."

- 일관성

안 좋은 리뷰

"함수 표현식을 사용하셨군요. `reduce`의 사용도 잘 활용했네요."

이 피드백은 코드의 전체를 본 것이 아니라 독립적으로 본 경우입니다. 일관된 코드인지 확인하기 위해서 코드를 전체를 봐야 합니다.

좋은 리뷰

"함수를 선언하는 두 가지 방식(함수 표현식, 함수 선언식)을 사용했는데 특별한 이유가 아니라면 함수를 선언하는 방식을 스스로 정하고 그 컨벤션 규칙을 따르도록 해보세요. 일관되게 동작하는 코드는 훨씬 보기 좋고 쉽게 수정할 수 있습니다. 그리고 `reduce`를 통해서 새로운 자료구조를 만든 건 잘했습니다. 하지만 같은 반복처리를 하는 과정에서 `calculateEarningRate` 에서 `for`문을 사용한 건 아쉽네요. `reduce`와 비슷한 방식의 다른 고차 함수를 찾아서 써보면 더 일관된 코드를 유지할 수 있을 거 같습니다."

- 상대방의 의도를 파악하여 이야기 할 것

안 좋은 리뷰

"클래스 상속은 필요 없습니다."

리뷰이의 의도는 파악하지 않은 채 클래스 상속은 필요 없다고 단정을 짓고 있습니다. 먼저 왜 이렇게 짰는지 의도를 파악하는 것이 좋습니다.

좋은 리뷰

"Component 클래스를 상속 받는 건 좋네요. 그러나 자식 클래스에서 부모 클래스를 호출만 하기 때문에 모든 클래스에서 상속받는 건 오히려 중복 코드 같아 보이기도 하는데 이렇게 작성해 주신 이유가 있을까요?"

- 개인적인 성향이나, 주관적 의견은 피할 것

안 좋은 리뷰

“return isWeekend(date) ? weekend : weekday; 로 작성해 보세요.  
코드가 깔끔해집니다.”

## 5. 리뷰를 너무 미루지 말 것!

### 정리

코드리뷰는 엔지니어들을 이어주는 매개체이자, 테스트, 정적 분석, 지속적 통합 등 거의 모든 다른 프로세스가 엮여 있는 최우선 개발 워크플로이며 개발자들의 지식 공유와 성장의 가장 최적화된 방법이다.

### 참고

[How to 코드리뷰] LGTM이 뭐지? : 유형별로 나눠보는 구글의 코드 리뷰

BY 한빛미디어] 코드 리뷰는 작성자 이외의 사람이 코드를 검토하는 프로세스입니다. 주로 코드를 코드...

<https://post.naver.com/viewer/postView.naver?volumeNo=34276636&memberNo=25379965&vType=VERTICAL>



효과적인 코드리뷰를 위한 리뷰어의 자세

안녕하세요, 톡FE파트에서 특명함 서비스를 개발하고 있는 Kay입니다.저는 2022년 신입 공채 기술 온보딩 교육의 코드 리뷰어로 활동을 했는데요, 이를 통해 얻었던 경험과 효과적인 코드 리뷰를 위한 리뷰어의 자세에 대해 소개하고자 합니다.신입 공채 기술 온보딩 교육은 6주간 미션 중심 과

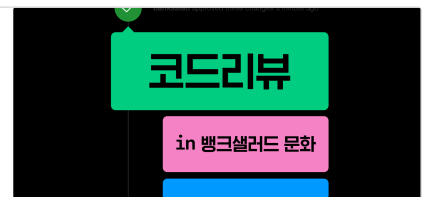
<https://tech.kakao.com/2022/03/17/2022-newkrew-onboarding-codereview/>



코드 리뷰 in 뱅크샐러드 개발 문화 | 뱅크샐러드

안녕하세요, 뱅크샐러드 BanksaladX iOS Engineer 정진규입니다. 코드 리뷰란 개발자가 작성한 코드를 다른 사람들이 검토하고 피드백을 전달하며, 다시 작성자가 반영하는 과정을 말합니다. 이 과정을 통해 서비스의 기술적인 구현 부분에 대해 공유·습득할 수 있으며, 일관된 아키텍처를 유지

<https://blog.banksalad.com/tech/banksalad-code-review-culture/>



### YouTube

우리 팀의 코드리뷰 문화, 이렇게 조금씩 발전했어요! #우아콘2022 #Day3\_함께하고성장하는우리  
코드리뷰를 두려워하고 그 중요성을 몰랐던 사람들이 이젠 누구보다 더 코드리뷰를 좋아하게 되었습니다. 어쩌다가 우리는 코드리뷰를 좋아하게 되었을까요? 배민스토어웹프론트개발팀의 코드리뷰 프로세스, 문화의 발전 과정, 각 팀원들의 경험과 소감까지! 코드리뷰에 진심인 저희 팀의 이야기를 들려드립니다. [발표자 소개] 강석진

<https://youtu.be/PBFUwGPp8DY>



### 도서

### 구글 엔지니어는 이렇게 일한다 | 타이터스 윈터스 - 교보문고

구글 엔지니어는 이렇게 일한다 | 구글은 어떻게 개발하고 코드를 관리하는가 지난 50년의 세월과 이 책이 입증한 사실이 한 가지 있습니다. 바로 '소프트웨어 엔지니어링의 발전은 결코 정체되지 않는다'라는 것입니다. 빠른 기술 변화 속에서 소프트웨어 엔지니어링의 중요성이 더욱 강조되면서

 <https://product.kyobobook.co.kr/detail/S000061352347>

TOILET  
Software Engineering at Google



구글리가 전하는 문화,  
프로세스, 도구의 모든 것