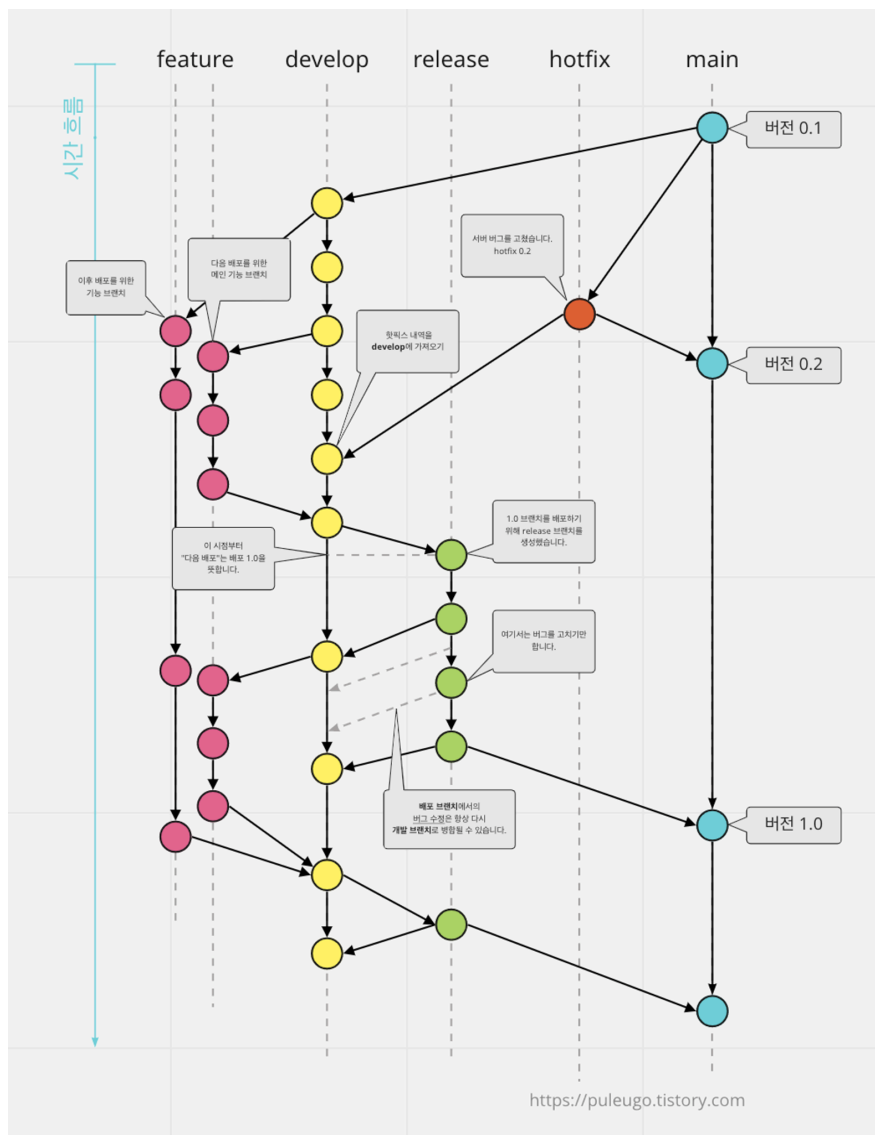


# Github Convention

2024년 2월 20일

SCV Team (Naver Boostcamp AI Tech 6기 CV-09)

## Git Flow



- branch는 main , develop, +@를 가진다.
- main branch는 수정하지 않는다.
- develop branch에서 본인의 +@ branch를 생성 후 작업한다. develop branch에서 작업하지 않는다.
- 본인의 branch에서 작업이 완료되면 develop branch로 merge를 진행하며 이때 리뷰어들의 승인을 받은 후 merge를 진행한다.
- 1 issue , 1 branch, 1 pr 을 지향한다.
- 예외의 상황들은 팀원들과의 논의 후 진행한다.

## Commit Convention

- 더는 간결하게 작성하고 명사로 끝낸다. body는 선택, footer는 필수
- header 키워드는 첫 글자만 대문자. 이외 전부 소문자로 통일한다.
- header에 마침표를 넣지 않는다.
- footer에 이슈번호 태그한다.
- 어떻게 보다는 무엇과 왜 위주로 작성한다.

Feat: mse 구현

<BLANK LINE>

<body>

<BLANK LINE>

#12

## Issue

- 제목은 [키워드] + 명사 로 작성한다.
- 제목의 영단어는 모두 소문자로 작성한다. 하지만 줄임말 등의 대문자 사용 단어들은 제외한다. ex) API , EDA, CI/CD etc..
- 목적에 맞는 템플릿에 맞춰서 작성한다. 빈 내용은 지양하며 템플릿내에서 빈 내용의 소제목은 소제목도 함께 삭제한다.
- ## to do 안의 내용이 없다면 to do 자체를 삭제한다.

## Issue: Feature request

새로운 기능을 추가할 때 사용하는 템플릿 If this doesn't look right, [choose a different type](#).



Add a title

[FEAT] ~~~~~ 구현

Add a description

Write

Preview

H B I

## Background

- ~~~~~ 가 필요해 ~~~~~ 를 구현한다.

## to do

- [ ] ~~~~~ 구현

Markdown is supported
 Paste, drop, or click to add files

## Pull Request

- 제목은 Commit Convention과 동일하게 가져간다.
- 또한 내용도 이슈와 마찬가지로 컨벤션을 가져간다.
- 빈 내용은 삭제하며, closed와 fixed 중 하나를 선택한다. ex) closed: #12

### ## Issue Tags

- closed: | fixed: #
- see also: #

### Issue Tags

- closed: [\[FEAT\] make dif #4](#)

- see also → 함께 보면 좋은 issue 태그
- \*\*\* 팀원들의 동의 하에 merge를 진행한다. \*\*\*
- reviewer 와 assigness 를 적극 활용한다.
- merge branch 설정 주의
  - 일반적인경우 base: develop , compare: 개인의 branch

- Pull Request 생성시 본문 맨위의 이슈번호는 지운다.

## Branch Name

- 소문자 사용
- 기능/이슈번호/간단한 설명 ex) feat/12-make-add