

# Github 개념 정리

## | Git이란 무엇일까? (홍성인)

: Git은 파일의 변경 사항을 시간에 따라 추적하는 버전 관리 시스템.

: 여러 개발자가 동시에 동일한 프로젝트에서 작업할 수 있도록 함.

### 1. Git의 5가지 중요 개념

#### a. 저장소 (Repository/ 레포 or Repo)

- 저장소는 파일과 그들의 수정 기록의 모음.
- 로컬 저장소는 여러분의 기계에, 원격 저장소는 서버에 호스팅.

#### b. 커밋 (Commit)

- 커밋은 프로젝트의 특정 시점의 스냅샷.
- 저장하려는 변경 사항의 집합을 나타냄.

#### c. 브랜치 (Branch)

- 다양한 부분에서 개발자가 동시에 작업할 수 있게함.
- 주 브랜치는 보통 master 또는 main이라 불리며, 각 개발자들은 작업을 위한 특정(개별) 브랜치를 만든다.

#### d. 머지 (Merge)

- 머지는 다른 브랜치에서의 변경 사항을 통합하는 것.
- 기능이나 버그 수정을 완료한 후에는 해당 브랜치를 주 브랜치로 머지(병합) 한다.

#### e. 풀 리퀘스트 (Pull Request)

- Git 기반의 업무방식에서 '풀 리퀘스트'는 주 브랜치에 변경 사항을 제안하는 방법.
- 다른 사람들이 변경 사항을 검토한 후에 머지(병합)하기 전에 논의하며 코드를 리뷰할 수 있다.

## | Github는 'Git의 소셜 네트워크' (김현규)

: 깃 허브(github)는 깃(git) 저장소를 호스팅하는 웹 기반 플랫폼

: 깃(git) 위에 그래픽 사용자 인터페이스(GUI)를 추가, 협업을 위한 기능 제공

예시) github는 마우스가 있는 컴퓨터, git은 마우스가 없던 시절 코드 기반의 저장소.

### 1. Github의 5가지 중요한 개념

#### a. 원격 저장소 (Remote Repository)

- 코드를 중앙 서버에 저장하고 여러 개발자가 작업하기에 용이

#### b. 협업 (Collaboration)

- Github는 여러 명이 동시에 작업하는 협업 프로젝트를 지원
- 각 참여자는 Github의 자신의 작업을 업로드하고 다른 참여자의 작업을 가져와서 병합(marge) 할 수 있음.

#### c. 커뮤니티

- Github는 활발한 커뮤니티를 가지고 있음.
- 다른 개발자와 질문과 답변을 나누고, 피드백을 받을 수 있음.

#### d. 이슈 트래킹

- 프로젝트 관리를 위해 버그, 기능 개선, 작업 등을 이슈로 등록하고 관리

#### e. 코드 리뷰

- 변경된 코드에 대한 의견을 주고받고 품질을 향상시키기 위한 리뷰 프로세스를 제공

## Git과 Github의 장점 (양해솔)

### 1. 효율적인 버전 관리

: git은 소스 코드의 모든 변경 사항을 추적하고 기록하여, 이전 버전으로 되돌릴 수 있고, 협업 프로젝트에서 효율적으로 사용할 수 있다.

### 2. 분산 저장

: git은 분산 버전 관리 시스템이기 때문에, 소스 코드가 중앙 서버에 저장되는 것이 아니라, 모든 참여자가 자신의 컴퓨터에 저장합니다. 이렇게 하면, 중앙 서버가 다운되더라도 소스 코드를 복구 할 수 있다.

### 3. 온라인 공유

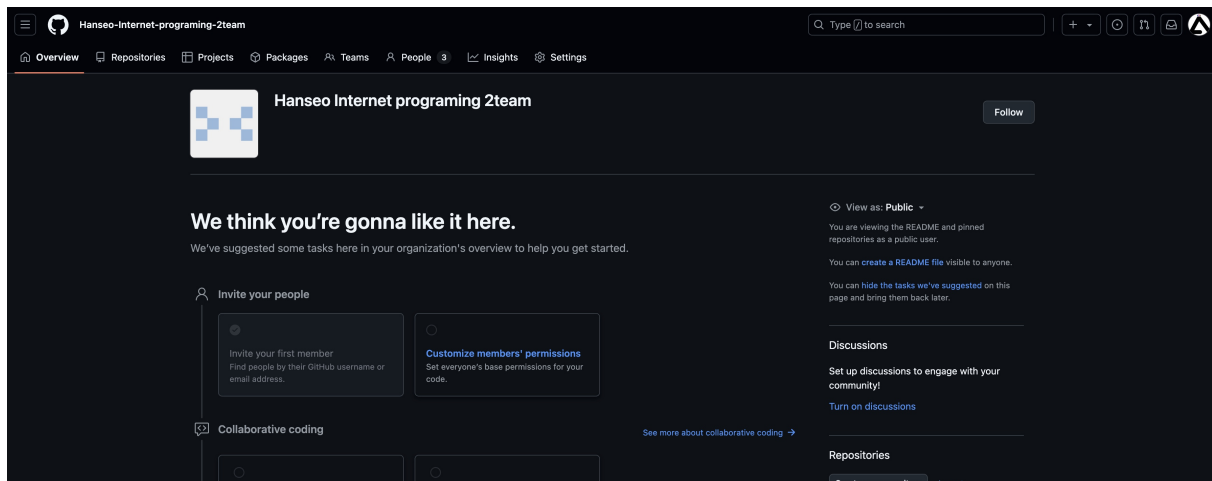
: Github는 Git으로 관리되는 프로젝트를 온라인에 쉽게 공유 가능, 이를 통해 다른 개발자와 협업하기 용이하고 오픈소스 프로젝트에 기여할 수 있다.

## 요약 (양해솔)

git은 버전 관리를 위한 도구

github는 git 저장소를 호스팅하고 공동 작업하기 위한 플랫폼(Github? 'Git의 소셜네트워크')

## 2조의 Github Repository 협업을 위한 팀 레포 생성 (신동우)



## 자료 출처

- <https://tlsdnjs12.tistory.com/7>
- <https://yanacoding.tistory.com/4>
- <https://jdcyber.tistory.com/45>
- <https://junvelee.tistory.com/21>