

How to use utility of GitHub

20155714 주은미 디지털이미징

1. GITHUB GLOSSARY

GitHub에 대해서 설명하기 전, 우선 GitHub의 용어들을 살펴보자면 다음과 같다.

Blame: 파일의 마지막 변경을 보여준다. 예를 들어, 무엇이 언제 추가되었는지를 확인하거나, 어느 commit이 버그를 만들었는지 blame을 따라 추적할 수 있다.

Branch: repository의 다른 branch들에 영향을 주지 않고 개발하고 싶을 때 사용한다. 각 repository마다 하나의 디폴트 branch가 있고, 다른 branch들도 추가할 수 있다. 또한 branch들을 병합할 수 있다. 일반적으로는 메인 프로젝트에서 branch를 만들어 변경 작업을 수행하고 작업이 완료되면, 'master'라는 프로젝트의 메인 디렉토리에 병합하는 식으로 이루어진다.

Clone: repository를 컴퓨터에 복사하는 것이다. Clone을 가지고 선호하는 편집기를 사용하여 파일을 편집할 수 있다. 또한 remote 버전으로 연결한다면, repository의 변경을 온라인이 아닌 상태에서도 여전히 추적할 수 있게 한다.

Collaborator: repository의 주인이 기여를 위해 해당 repository로 초대하는 사람으로 collaborator라고 하며, 해당 repository에 접근 가능한 사람을 말한다.

Commit: 파일(또는 파일들)의 개별적인 변화를 말한다. Commit을 하면 그 순간의 repository의 체크포인트가 만들어지고 프로젝트 이전의 어느 상태로든 복원이 가능하다.

Contributor: 프로젝트에 기여한 사람을 말한다. 하지만 collaborator 접근권한은 갖지 못한다.

Fork: 다른 사용자의 repository를 개인적으로 복사하여 자신의 개정에 가져오는 것이다.

Issue: repository에 관련되는 개선이 필요한 부분을 말한다. Issue는 누구나 만들 수 있으며, 생성된 후에는 collaborator들이 적당하게 수정할 것이다.

Open source: Open source 소프트웨어는 누구든지 자유롭게 사용하고, 변경하고, 공유할 수 있다.

Repository: GitHub의 가장 기본적인 요소이다. 프로젝트 폴더와 비슷하다.

2. UTILITY OF GITHUB

GitHub를 사용할 때, 가장 먼저 해야 할 일은 repository를 생성하는 것이다. Repository는 프로젝트 폴더라고 생각할 수 있고, 프로젝트 관리를 하는 곳이다.

서버에 repository를 만들었지만 보통 작업은 컴퓨터에서 이루어질 것이다. 따라서 자신의 로컬 컴퓨터에 원격 repository를 복제(Clone)하는 과정이 필요하다. Git은 이렇게 remote(온라인) repository와 local repository가 존재하며 local repository에 remote repository를 복제하여 컴퓨터로 작업을 한 후 서버(remote repository)에 반영하고 싶은 변화만 선택적으로 업데이트하면 된다. 이러한 기능은 팀으로써 소프트웨어 개발을 할 때 개개인들이 독립적으로 개발할 수 있게 도와준다.

Commit은 해당 repository, 파일에 수정을 했을 때 만들어지며, 그 순간의 체크포인트가 함께 생성된다. 체크포인트는 개발 도중 이전의 어떤 상태로든 돌아가는 것을 가능하게 하기 때문에 commit의 단위를 결정하는 것도 중요하다. 문서작업을 예로 들면, 한 문장을 수정하고 commit할 수 있고, 한 장을 수정하고 commit을 할 수 있다. 이때 한 문장을 한 commit의 단위로 할지, 한 장을 한 commit을 단위로 할지 결정하는 것이다.

Branch는 원래의 코드와 독립적으로 개발할 수 있도록 하는 기능이다. Branch를 사용하면 개별적으로 개발한 후, 원래의 코드에 병합하는 것이 가능하다. Branch를 만들어 작업했을 때 해당 branch의 작업이 성공적으로 완료된다면, 이전 branch와 병합할 수 있고, 실패한다면 이전 branch의 체크포인트로 돌아간 후 실패한 branch를 삭제하면 된다.

1:2 •

위에서 **GitHub**의 기능을 간단히 설명했다. **GitHub**는 개인, 그룹의 소프트웨어 프로그램 개발, 관리를 돕는 도구이다. 위의 기능들을 적절히 활용하면, 프로그램 개발을 매우 효율적으로 할 수 있을 것이다.

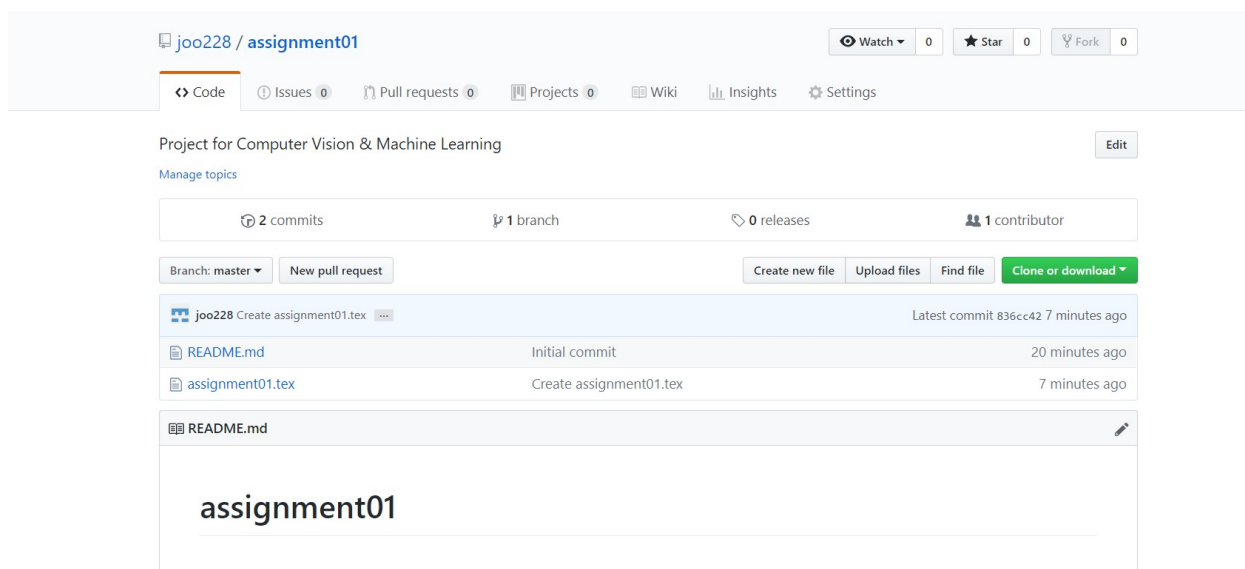


Fig. 1. <https://github.com/joo228/assignment01.git>