♨[Group Study] – Github by jinuman

**Local : init -> work -> diff -> add -> commit**

**Remote : pull -> work -> commit -> pull -> push**

* 로컬 저장소 : 자신의 컴퓨터 / 원격 저장소 : Github
* Github
  + Git으로 관리되고 있는 버전을 보관해주는 온라인 서비스중에 가장 대표적인 서비스 == 원격저장소(Remote Repository)
  + 협업, 소통, 업무관리를 담당하는 오픈소스 사이트
  + public 즉 공개되는 저장소는 공짜이고 private 저장소는 돈을 내야 한다

비슷한 서비스로는 Gitlab.com

♧왜 잔디나 구글드라이브 같은 사용하기 쉬운 툴이 있는데 개발자들은 코드를 보관하기 위해 Github을 쓰는걸까??

1. Git으로 관리한 commit들(History) 관리 및 저장
2. 코드리뷰! 오픈소스의 장이다..

Github의 또다른 좋은 점은 연동이 되는 툴들이 많다. 그러므로 문서 및 일정관리는 다른 툴을 이용하고 Github과 연동해서 그 툴에서 알림을 받을 수 있도록 하는 것도 좋은 방법!

X 
GitHub, 
Shopping 
Inc. [LISI 
Junk 
x github repo x 
https://github.com,/jinuman/market-service-study 
Study for Develop 
Beauty 
DevelopBIog 
This repository Search 
jinuman / market-service-study 
git - How c X 
14. Discard; x \ 
KHU 
pull requests 
Issues 
Marketplace 
Gist 
O Network V 
@Watch 
0 
O jinuman/me 
Unstar 
Fork 
00b 
Pull requests O Projects O 
Wiki 
Settings 
Insights 
'Ale 
0 
Edit 
Add topics 
9 commits 
Branch: master • 
New pull request 
V 1 branch 
Initial commit 
Update README.md 
Community 
pulse 
IdL Graphs 
O O releases 
Create new file 
Upload files 
1 contributor 
Find file 
Clone or download • 
—jinuman committed on GitHub Update README.md 
.gitignore 
README.md 
README.md 
Latest commit 39f2d79 23 minutes ago 
23 days ago 
23 minutes ago 
market-service-study 
This service is basically for users who want to lend or borrow some longing products. 
Aiming to develop as a group study. 
71=1 
617 
2017-07-22 

* Fork : 프로젝트를 복제하는 기능
  + 자신의 저장소로 가져온 다음에 마치 자기 것인 것처럼 마음대로 수정 가능
  + 옆에 숫자는 얼마나 많은 사람들이 이 프로젝트를 Forking해갔는지를 나타낸다.
* Code : Code나 파일들을 보여주는 곳
  + 9 commits : 이 프로젝트에 총 9개의 commit이 존재 했다는 정보
  + branch : 보통 작업을 필요할 때마다 분기하는 기능으로 쓰인다. 클릭하면 branch들의 목록을 볼 수 있고 기본(default) branch로는 master로 항상 설정되어있다.
* Fork한 사람이 pull-request를 하면 원작자가 추가된 내용을 보고 판단해서 반영해도 되겠다 싶으면 반영할 수 있는 기능
* Issue : 버그 발견이나 질문등을 쓰는 게시판 같은 공간(업무상)이라고 생각하면 된다.
  + Label, Milestone등 기능이 있다. 나중에 업무를 할 때 체계적으로 버전을 나누어서 배포할 목적일 경우 협업하는 사람들끼리 커뮤니케이션 용도로 사용할 듯 싶다.
  + 어떤 특정 사람에게 이슈를 넘겨서 이메일로 통보를 받아서 그 업무를 해결하게 할 수도 있다.
* Pull-Request
  + fork를 한 사람이 pull-request를 하면 원작자가 추가된 내용을 보고 판단해서 반영해도 되겠다 싶으면 반영할 수 있는 기능
  + 코드 리뷰에 쓰이는 방법이다!
* Graphs : 오픈소스의 입장에서 이 프로젝트가 얼마나 많은 사람들과 얼마나 활발히 참여되고 있는지 볼 수 있는 곳이다.
* Clone or download : 원격 저장소를 자신의 로컬로 복제할 때 쓰이는 clone버튼이다.
  + 저걸 누르면 자동으로 원격 저장소의 URL?이 복사된다. bash에서 git clone {복사한 url} 해서 자신의 local로 가져갈 수 있다.
* 그 외
  + Watch : 어떤 프로젝트를 watch하면 그 프로젝트의 변화를 알림으로 받을 수 있고 watch를 한 프로젝트들만 쉽게 찾을 수 있다.
    - Watch옆에 숫자는 얼마나 이 프로젝트가 흥행이된 오픈소스인지를 보여주는 척도이다..
* **협업모델**
  + Fork & Pull : collaborator를 안하고 한명이 관리하는 방식
  + 공유 저장소 모델 : collaborator를 해서 중요한 것만 pull-request 하는 방식