

정규 교육 세미나

ToBig's 9기 신현경

Git & Github

Contents

Unit 01 | Git & Github 소개

Unit 02 | Git 사용하기

Unit 03 | Github 사용하기

Unit 04 | 과제

Unit 01 | Git & Github 소개



git

= Version Control System 중 하나

Unit 01 | Git & Github 소개

버전 관리(Version Control)는 왜 필요할까?

Unit 01 | Git & Github 소개

버전 관리는 왜 필요할까?



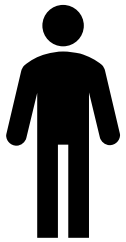
A, B, C



학교 수업 프로젝트 < BS >게임 완성!

Unit 01 | Git & Github 소개

버전 관리는 왜 필요할까?



A



< BS > Version 1

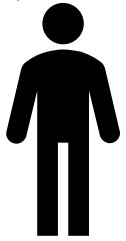


< BS > Version 2

Unit 01 | Git & Github 소개

버전 관리는 왜 필요할까?

망했다



A



< BS > Version 1



< BS > Version 2

Unit 01 | Git & Github 소개

버전 관리는 왜 필요할까?

망했다

팀원 모두 학점 F받음.....ㅠㅠ

< BS > Version 1

< BS > Version 1.1

Unit 01 | Git & Github 소개

버전 관리는 왜 필요할까?

망했다

이전 버전으로 돌아갈 수 있도록
버전 관리가 필요하겠어!

< BS > Version 1

< BS > Version 1.1

Unit 01 | Git & Github 소개

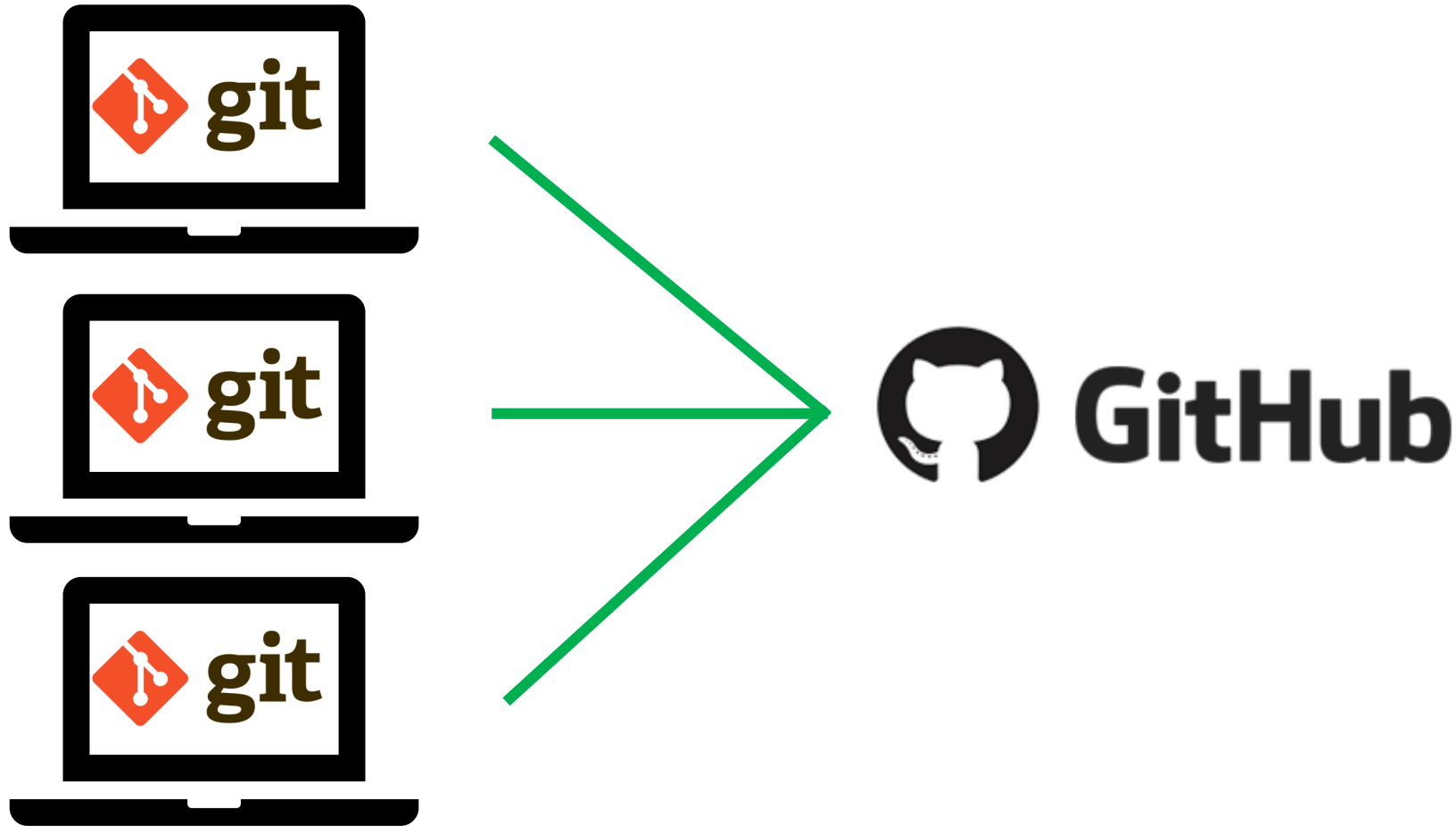


- Managing version
- Backup
- Recovery
- Collaboration

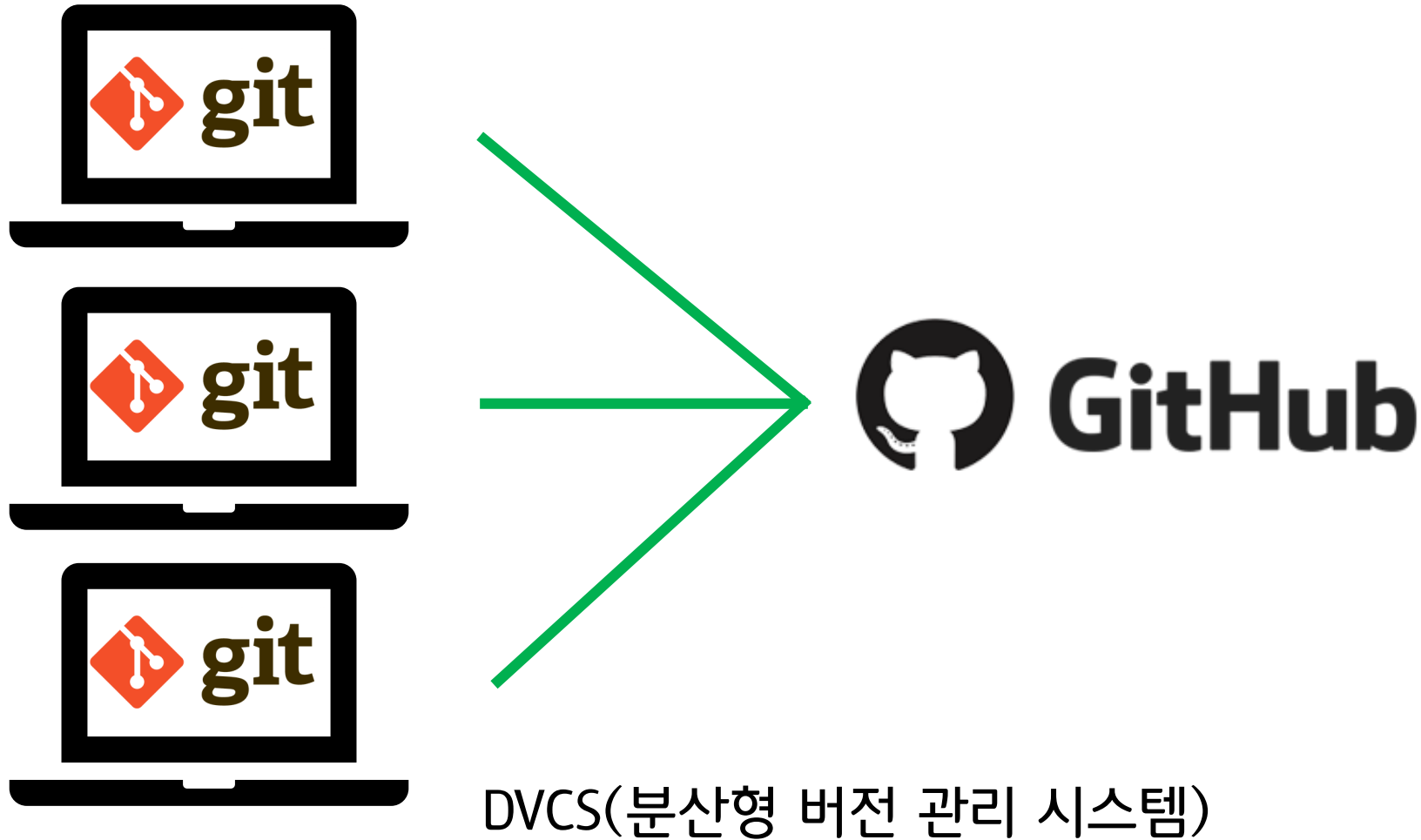
Unit 01 | Git & Github 소개



Unit 01 | Git & Github 소개



Unit 01 | Git & Github 소개



Summary

- Git is version management tool using local repository
- Github is remote repository for collaboration by multiple git users

Unit 02 | Git 사용하기

Local repository

Remote repository

Unit 02 | Git 사용하기

Local repository

내 PC에 파일이 저장되는
개인 전용 저장소

Remote repository

파일이 원격 저장소 전용
서버에서 관리되며
여러 사람이 함께 공유
하기 위한 저장소

Unit 02 | Git 사용하기

Local repository

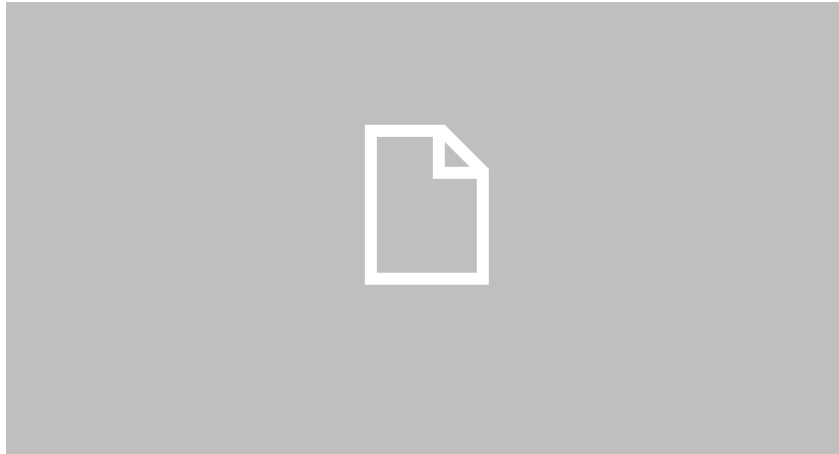


Remote repository



Unit 02 | Git 사용하기

Local repository



올리기

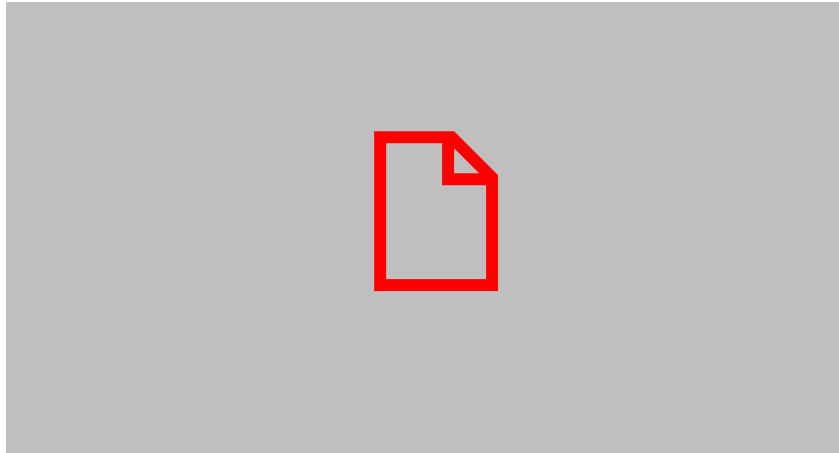


Remote repository

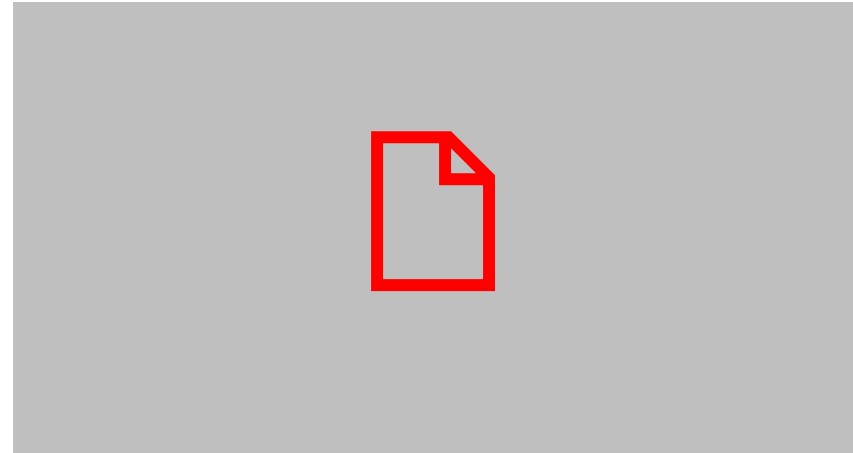


Unit 02 | Git 사용하기

Local repository



Remote repository



←
받아 오기

Unit 02 | Git 사용하기

Directory



=



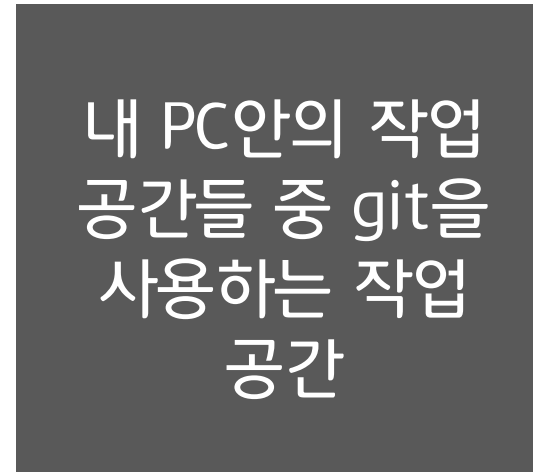
Unit 02 | Git 사용하기

Directory



→
init

Working Directory



Unit 02 | Git 사용하기

Working Directory

.git/

Index

Local Repository

Unit 02 | Git 사용하기

Working Directory

내 PC안의 작업
공간들 중 git을
사용하는 작업
공간

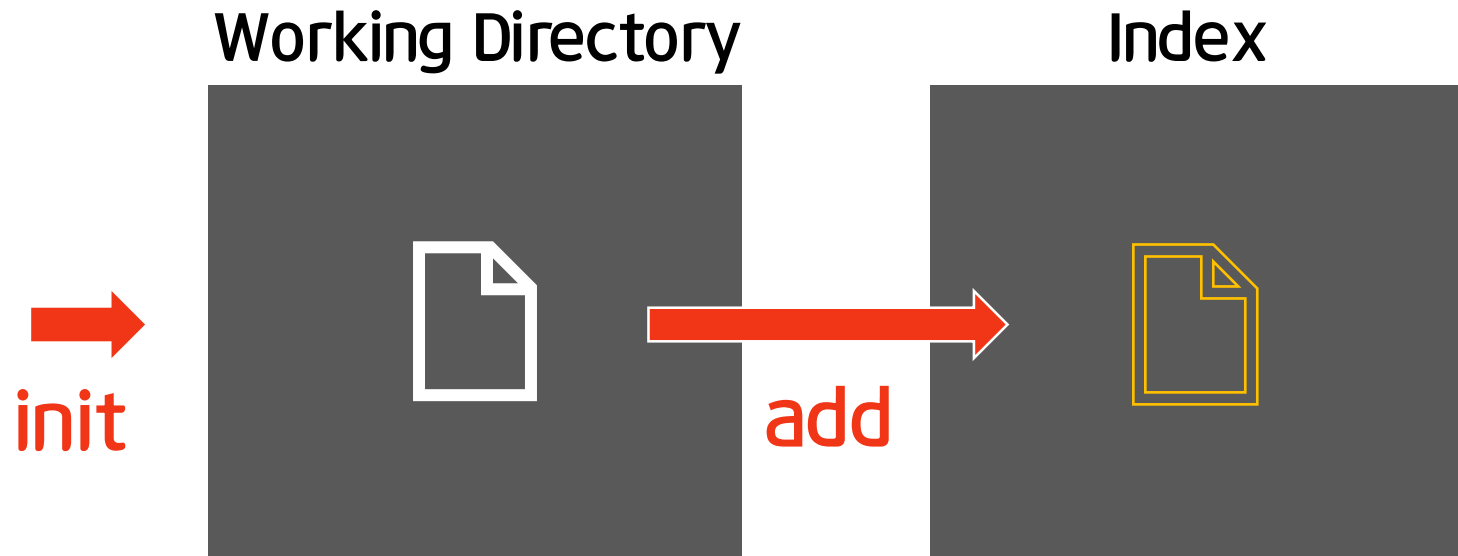
Index

임시 버전들이
올라가는 공간

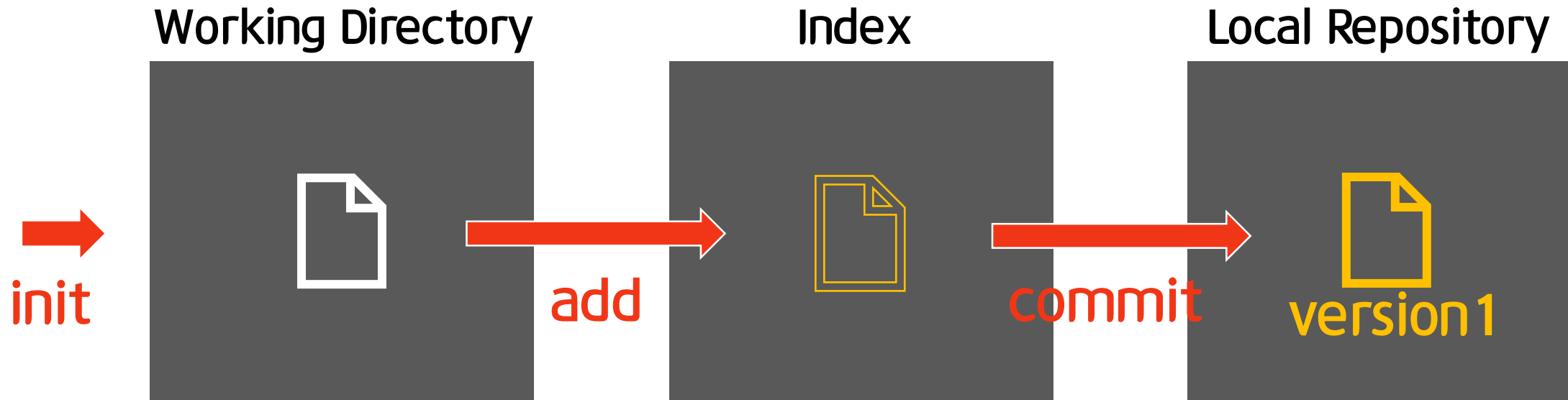
Local Repository

최종 확정본이
올라가는 공간

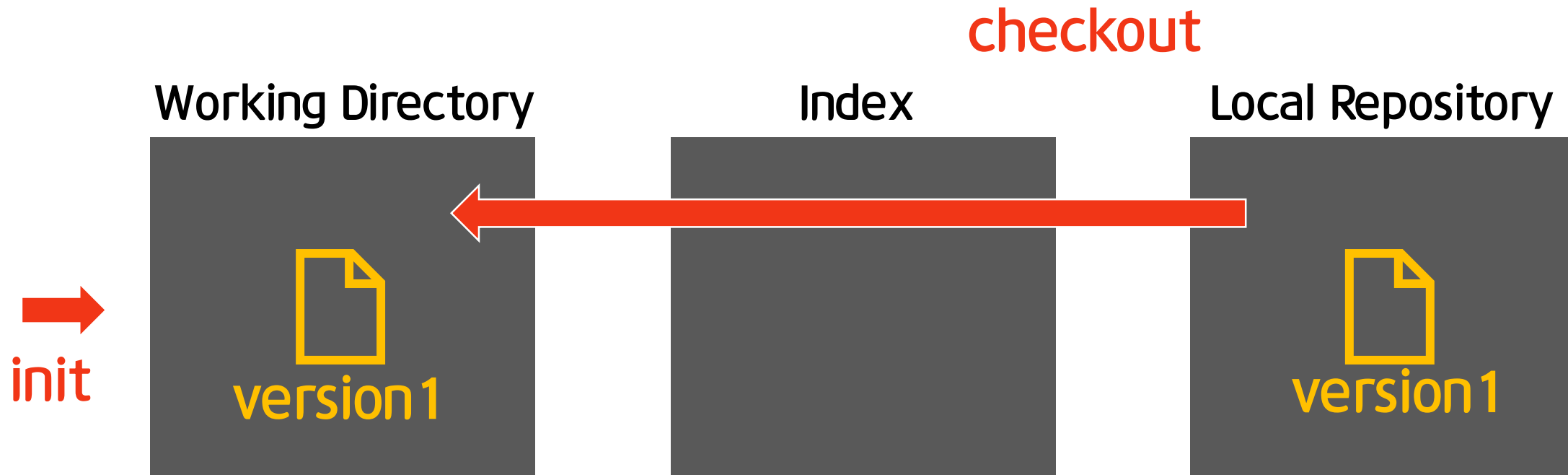
Unit 02 | Git 사용하기



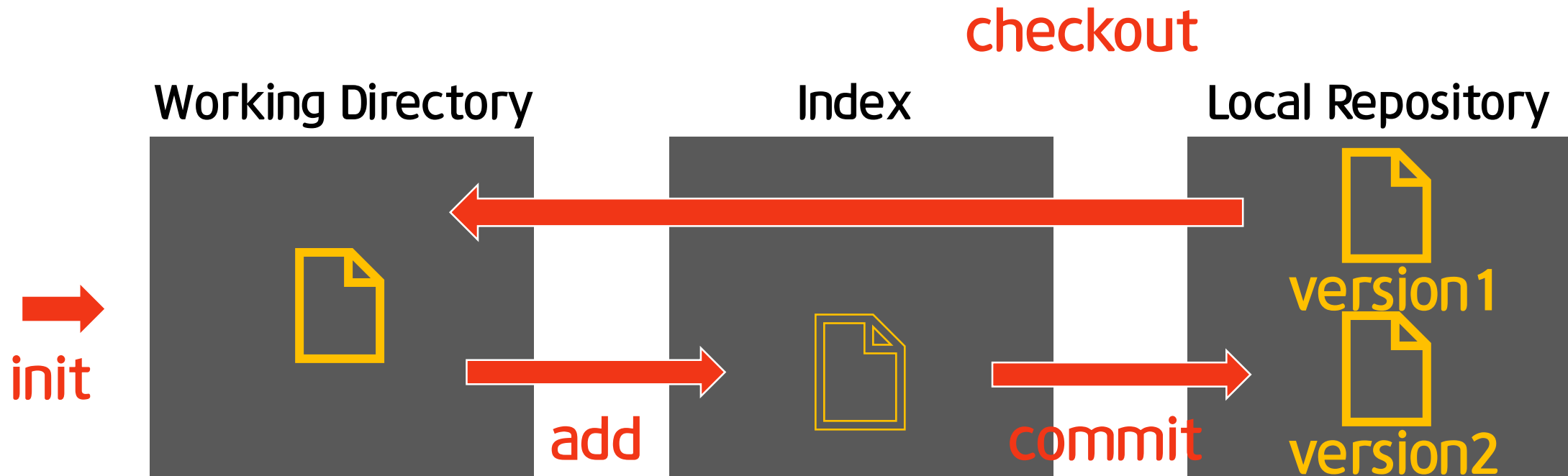
Unit 02 | Git 사용하기



Unit 02 | Git 사용하기



Unit 02 | Git 사용하기



Unit 02 | Git 사용하기

참고) 간단한 bash 명령어 정리

mkdir : 디렉토리를 만들

ls : 현재 디렉토리의 list

ls : 파일명만 보여줌

ls -a : 디렉토리 내 모든 것 (ex. 폴더, ..)

ls -al : 디렉토리 내 모든 것 + 접근권한 + 생성날짜 등

pwd : 현재 위치

cd : 디렉토리 변경

cd : 최상위 폴더로 이동

cd dir_name : dir 로 이동

vim file_name : 파일 생성, 편집

file_name 이 현재 dir에 없는 경우 생성

file_name 이 현재 dir에 있는 경우 편집

(편집 방법들

i - 삽입, :w - 저장,

:q - 나가기, :wq - 저장 후 나가기)

cat file_name : 파일의 내용 출력

Unit 02 | Git 사용하기

0 환경 설정

전역 사용자명/이메일 구성

```
git config --global user.name "yourname"
```

```
git config --global user.email "youremailaddress"
```

저장소별 사용자명/이메일 구성(해당 저장소 디렉토리로 이동 후)

```
git config user.name "yourname"
```

```
git config user.email "youremailaddress"
```

Unit 02 | Git 사용하기

0 환경 설정

전역 설정 정보 조회

```
git config --global --list
```

저장소별 설정 정보 조회

```
git config --list
```

Unit 02 | Git 사용하기

1 로컬 저장소 만들기

방법 1: 아예 저장소를 새로 만들기

방법 2: 이미 만들어져 있는 원격 저장소를 로컬 저장소로 복사해오기

Unit 02 | Git 사용하기

1 로컬 저장소 만들기

방법 1: 아예 저장소를 새로 만들기

폴더를 하나 만들고, 그 안에서 `git init`을 실행하세요.
새로운 git 저장소가 만들어집니다.

Unit 02 | Git 사용하기

1 로컬 저장소 만들기

방법 1: 아예 저장소를 새로 만들기

```
LG@DESKTOP-RBH8J5B MINGW64 ~  
$ mkdir tobig  
  
LG@DESKTOP-RBH8J5B MINGW64 ~  
$ cd tobig  
  
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig  
$ git init  
Initialized empty Git repository in C:/Users/LG/tobig/.git/
```

Unit 02 | Git 사용하기

1 로컬 저장소 만들기

방법 2: 이미 만들어져 있는 원격 저장소를 로컬 저장소로 복사해오기

로컬 저장소를 복제 하려면

```
git clone /로컬/저장소/경로 실행
```

원격 서버의 저장소를 복제하려면

```
git clone 사용자명@호스트:/원격/저장소/경로 실행
```

Unit 02 | Git 사용하기

1 로컬 저장소 만들기

방법 2: 이미 만들어져 있는 원격 저장소를 로컬 저장소로 복사해오기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git clone https://github.com/rbgirshick/py-faster-rcnn.git
Cloning into 'py-faster-rcnn'...
remote: Counting objects: 1544, done.
remote: Total 1544 (delta 0), reused 0 (delta 0), pack-reused 1544
Receiving objects: 100% (1544/1544), 1.36 MiB | 1.43 MiB/s, done.
Resolving deltas: 100% (903/903), done.
```

강의테스트
Add topics

2 commits

1 branch

0 releases

1 contributor

Edit

Create new file Upload files Find file Clone or download

Clone with HTTPS ⓘ Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/HKtobigs9/TobigsTest. 📄

Open in Desktop Download ZIP



Unit 02 | Git 사용하기

2 staging하기

변경된 파일은 인덱스에 추가할 수 있음.

`git add <파일이름>` // 파일 생성/수정한 후 알리기

`git add .` // 이전 commit이후의 모든 파일의 생성/수정 알리기

Unit 02 | Git 사용하기

2 staging하기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ vi abc.txt

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        abc.txt

nothing added to commit but untracked files present (use "git add" to track)

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git add abc.txt
warning: LF will be replaced by CRLF in abc.txt.
The file will have its original line endings in your working directory.

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   abc.txt
```

Unit 02 | Git 사용하기

3 로컬 저장소에 올리기

실제로 변경 내용을 확정하려면 commit으로 local repository에 올려야 함.

`git commit -m "설명"` // 설명과 함께 commit하기

`git commit <파일이름>` // add한 파일들 중 일부만 commit하기

`git commit -am "설명"` // add했던 파일을 수정한 경우 add와 commit을 동시에

Unit 02 | Git 사용하기

3 로컬 저장소에 올리기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git commit -m "abc.txt version1"
[master (root-commit) bd22401] abc.txt version1
1 file changed, 1 insertion(+)
create mode 100644 abc.txt

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git status
On branch master
nothing to commit, working tree clean

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git commit
On branch master
nothing to commit, working tree clean
```

Unit 02 | Git 사용하기

4 확인 하기

중간중간 상황 확인하는 명령어

`git status` // 현재 파일들의 상태를 볼 수 있음.

`git log` // commit된 내역을 볼 수 있는 명령어

Unit 02 | Git 사용하기

4 확인 하기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git log
commit bd22401155ca030cf76c3fb74838ada42fde34a7 (HEAD -> master)
Author: HKtobigs9 <HKtobigs9@gmail.com>
Date: Sat Aug 25 19:26:59 2018 +0900

    abc.txt version1

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Unit 02 | Git 사용하기

5 지운 파일을 로컬 저장소에서 복원하기

`git checkout --<파일 이름>` //로컬 저장소에 있는 파일을 복원시키기

Unit 02 | Git 사용하기

5 지운 파일을 로컬 저장소에서 복원하기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig_0829 (master)
$ rm abc.txt

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig_0829 (master)
$ ls
TobigsTest/

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig_0829 (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        deleted:    abc.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        TobigsTest/

no changes added to commit (use "git add" and/or "git commit -a")
```

Unit 02 | Git 사용하기

5 지운 파일을 로컬 저장소에서 복원하기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig_0829 (master)
$ git checkout -- abc.txt

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig_0829 (master)
$ ls
abc.txt  TobigsTest/
```

Unit 02 | Git 사용하기

실습

- 1 내가 직접 저장소를 만들어서 abc.txt파일을 생성해 commit해보기
- 2 이미 만들어져 있는 원격 저장소
<https://github.com/HKtobigs9/TobigsTest.git> 를 로컬 저장소로 복사해오기
- 3 abc.txt를 실수로 지우고 로컬 저장소에서 다시 복원시키기

Unit 03 | Github 사용하기

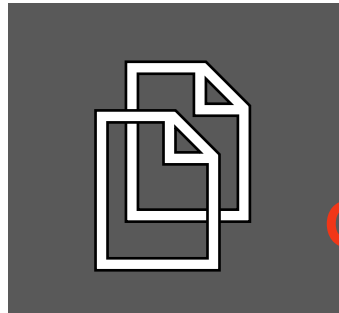
0 작업 흐름

Working Directory



→
add

Index



→
commit

Local Repository



Remote repository



Unit 03 | Github 사용하기

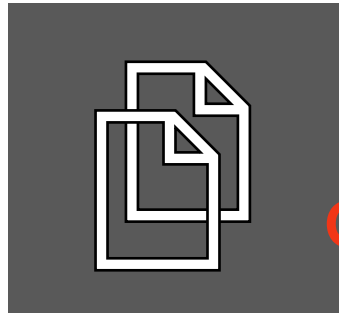
0 작업 흐름

Working Directory



→
add

Index



→
commit

Local Repository



→
remote

Remote repository



Unit 03 | Github 사용하기

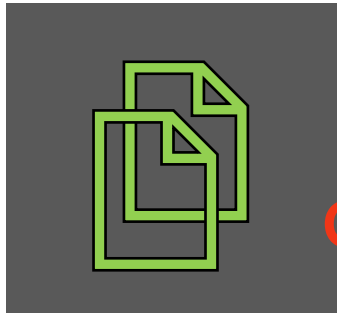
0 작업 흐름

Working Directory



→
add

Index



→
commit

Local Repository



→
push

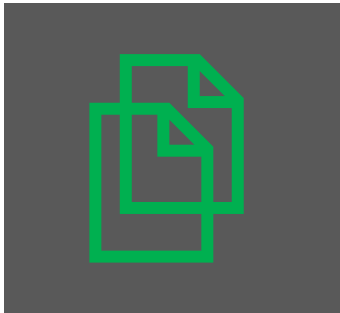
Remote repository



Unit 03 | Github 사용하기

0 작업 흐름

Working Directory



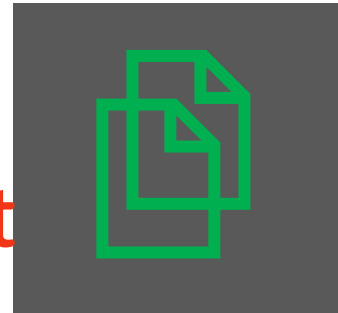
→
add

Index



→
commit

Local Repository



←
pull

Remote repository



Unit 03 | Github 사용하기

0 작업 흐름

Working Directory



→
add

Index



→
commit

Local Repository



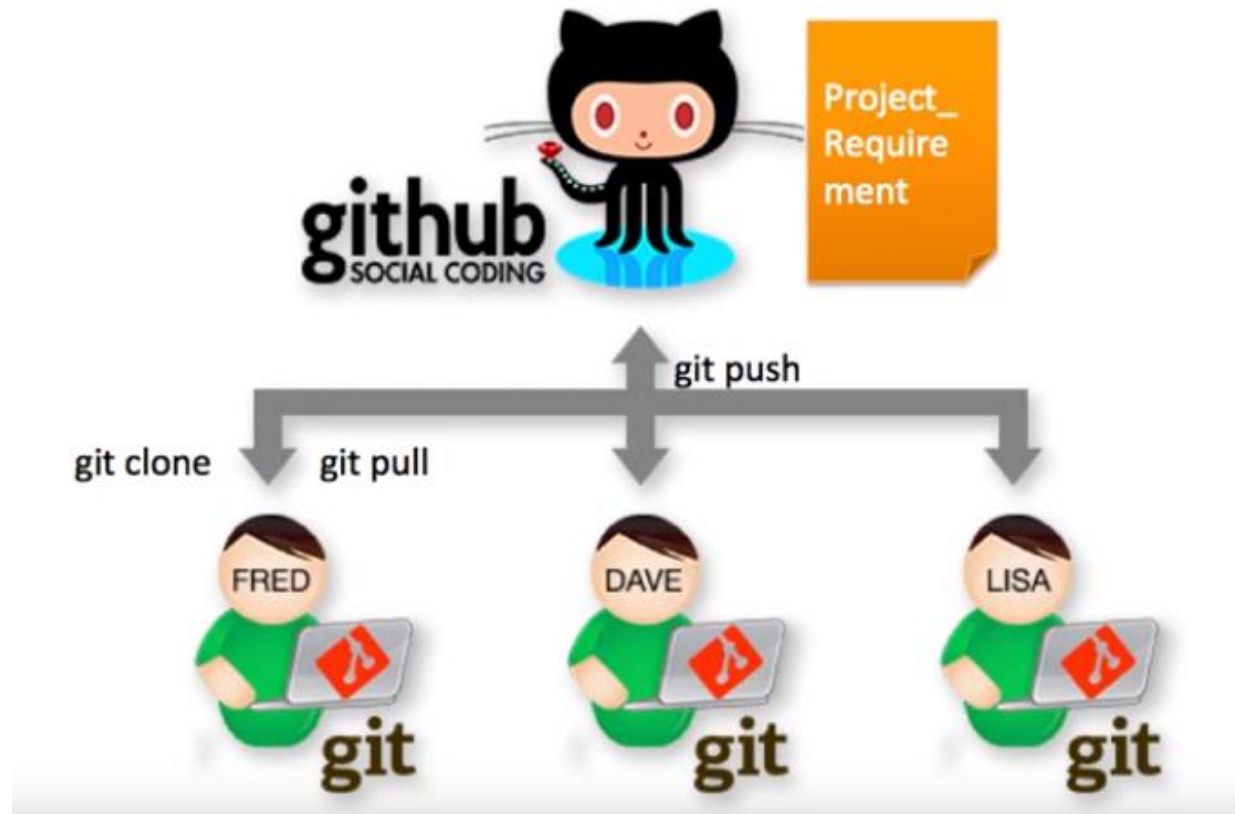
←
clone

Remote repository



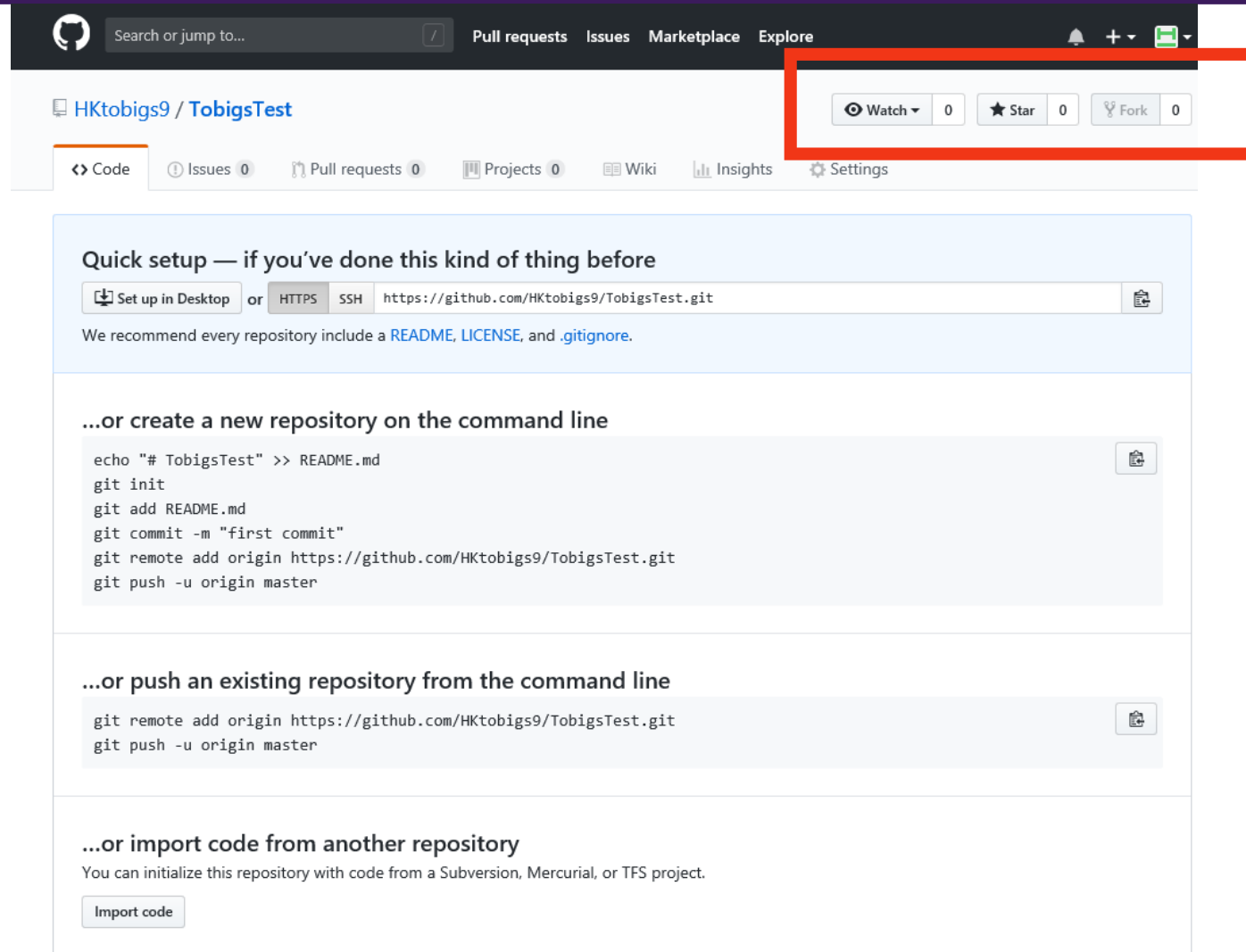
Unit 03 | Github 사용하기

Github Architecture



Unit 03 | Github 사용하기

New Repository 생성



Unit 03 | Github 사용하기

New Repository 생성

Search or jump to... Pull requests Issues Marketplace Explore

HKtobigs9 / TobigsTest Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/HKtobigs9/TobigsTest.git`

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# TobigsTest" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/HKtobigs9/TobigsTest.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/HKtobigs9/TobigsTest.git
git push -u origin master
```

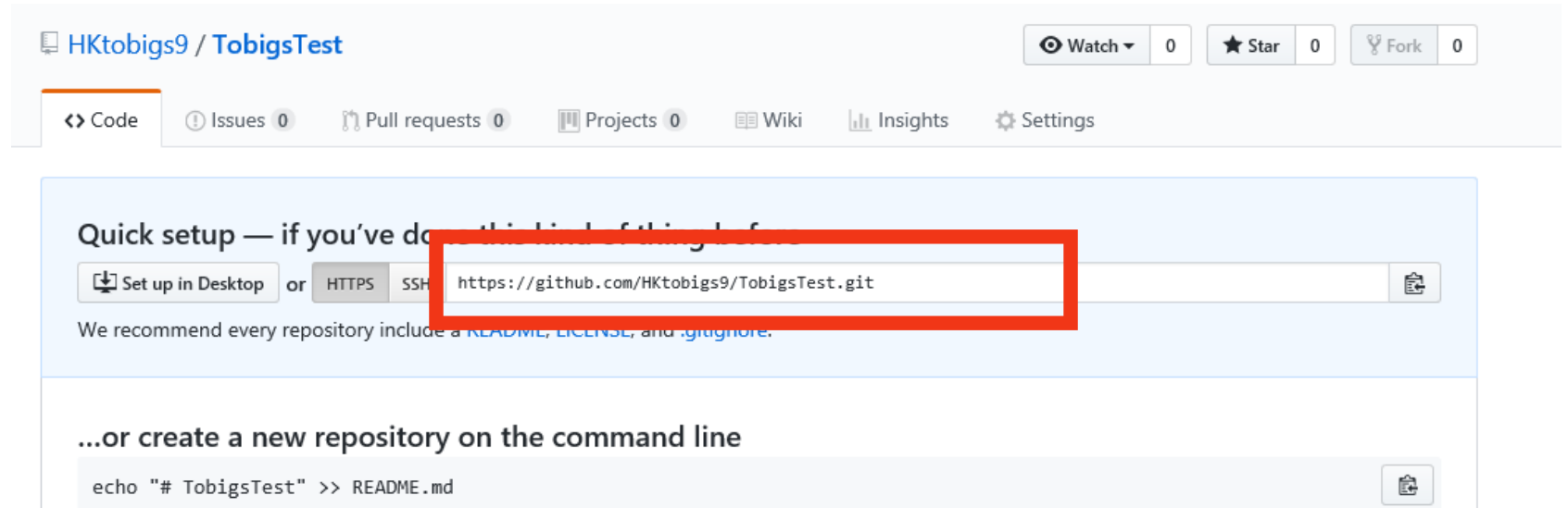
...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Unit 03 | Github 사용하기

1 원격 저장소와 연결



원격저장소를 관리할 수 있는 명령어 git remote를 이용하여 연결

`git remote add origin 사용자명@호스트: /원격/저장소/경로` 실행

- 꼭 origin이라고 하시지 않아도 됩니다. 별칭일 뿐이니깐요.

이제 orgin이라는 이름으로 원격 저장소 주소 등록 완료됨!

Unit 03 | Github 사용하기

2 원격 저장소에 작업한 것을 올리는 push

원격저장소에 commit을 저장하기 위해

`git push -u origin master` 실행

- origin은 원격저장소 이름, master은 현재 사용하는 컴퓨터의 브랜치 이름
- -u 옵션은 향후에 수정된 내용을 push 혹은 pull하기 쉽게 할 수 있도록 하는 옵션이다.

Unit 03 | Github 사용하기

2 원격 저장소에 작업한 것을 올리는 push

HKtobigs9 / TobigsTest

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

강의테스트 Edit

Add topics

2 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

HKtobigs9 ab version2 Latest commit d95d0ff 20 minutes ago

ab.txt ab version2 20 minutes ago

Help people interested in this repository understand your project by adding a README. Add a README

Unit 03 | Github 사용하기

3 변경된 원격 저장소를 로컬 저장소에 동기화하려면 pull

원격저장소에서 로컬 저장소로 업데이트하려면(다른 사람의 수정한 것을 받기 위해)

`git pull origin master` 실행

- origin의 내용이 master로 복사됨.

Cf) 참고로 git pull을 할 때는 깃허브의 유저이름과 비밀번호를 쳐야 하는 경우가 많습니다. 매번 비밀번호를 치기 귀찮으면 `git config --global credential.helper 'store --file 경로'`하면 됩니다. 해당 경로에 비밀번호가 저장된 파일이 생성됩니다. 단, 파일로 저장되는 만큼 보안에 취약하기 때문에 주의해야 합니다.

Unit 03 | Github 사용하기

4 Branch 생성

Branch?

어학사전

영어사전

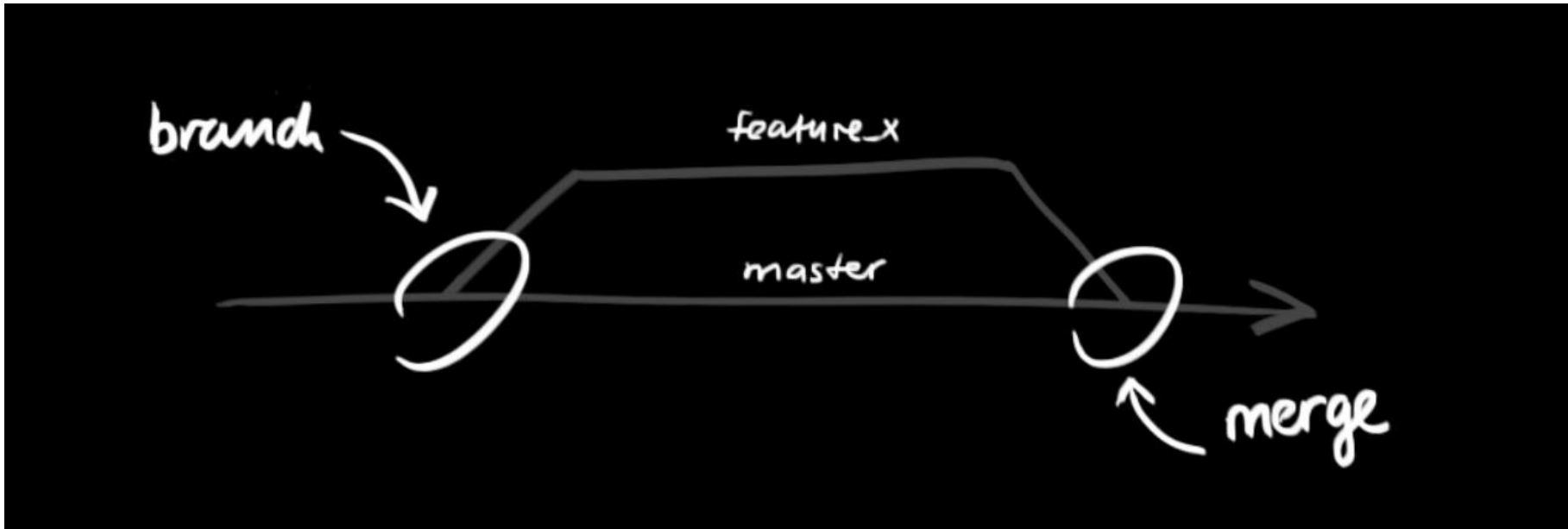
[branch](#) 미국식 [bræntʃ]  영국식 [bra:ntʃ]  ★★ [예문보기](#)

1. 나뉘어가지 2. 지사, 분점 3. (큰 조직의) 분과

Unit 03 | Github 사용하기

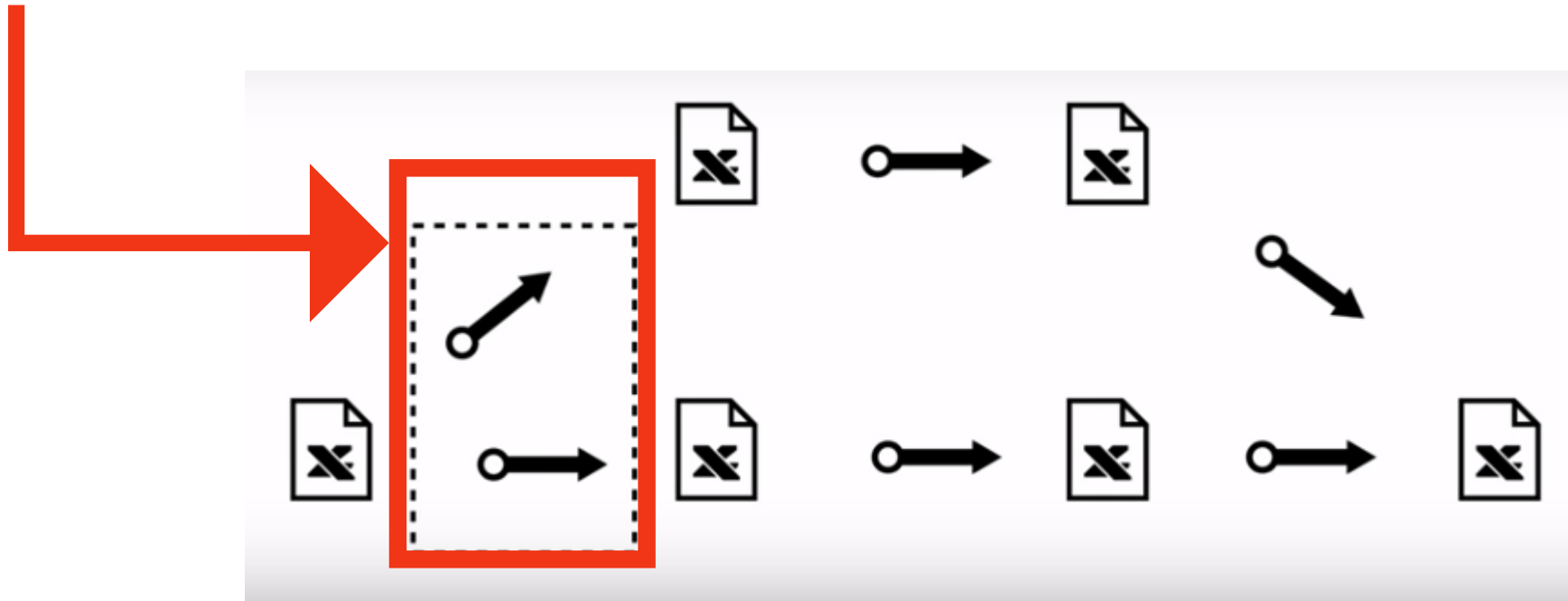
4 Branch 생성

Branch?



Unit 03 | Github 사용하기

4 Branch 생성



새로 만든 가지를 원격 저장소로 전송하기 전까지는 다른 사람들이 접근할 수 없음.

Unit 03 | Github 사용하기

4 Branch 생성

새로운 branch 생성하려면

`git branch 브랜치명` 실행

- 잘 생성되었나 보려면 `git branch`라고 쳐보세요, 앞에 *표시가 있는게 현재 Repository가 있는 branch

Unit 03 | Github 사용하기

4 Branch 생성

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch
* master

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch shk

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch
* master
shk
```

Unit 03 | Github 사용하기

5 원하는 Branch에서 작업

새로 만든 branch에서 작업하려면

`git checkout` 새로운 브랜치명 실행

- branch변경이 잘 됐는지 보려면 `git branch`라고 쳐보세요, 앞에 *표시가 있는게 현재 Repository가 있는 branch
- master로 돌아가려면 `git checkout master`라고 치면 됩니다.
- `git checkout -b` 새로운 브랜치명을 하면 새로만들어지면서 checkout도 동시에 됩니다.

Unit 03 | Github 사용하기

5 원하는 Branch에서 작업

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git checkout shk
Switched to branch 'shk'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git branch
  master
*  shk

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git checkout master
Switched to branch 'master'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch
*  master
  shk
```


Unit 03 | Github 사용하기

5 원하는 Branch에서 작업

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git checkout shk
Switched to branch 'shk'

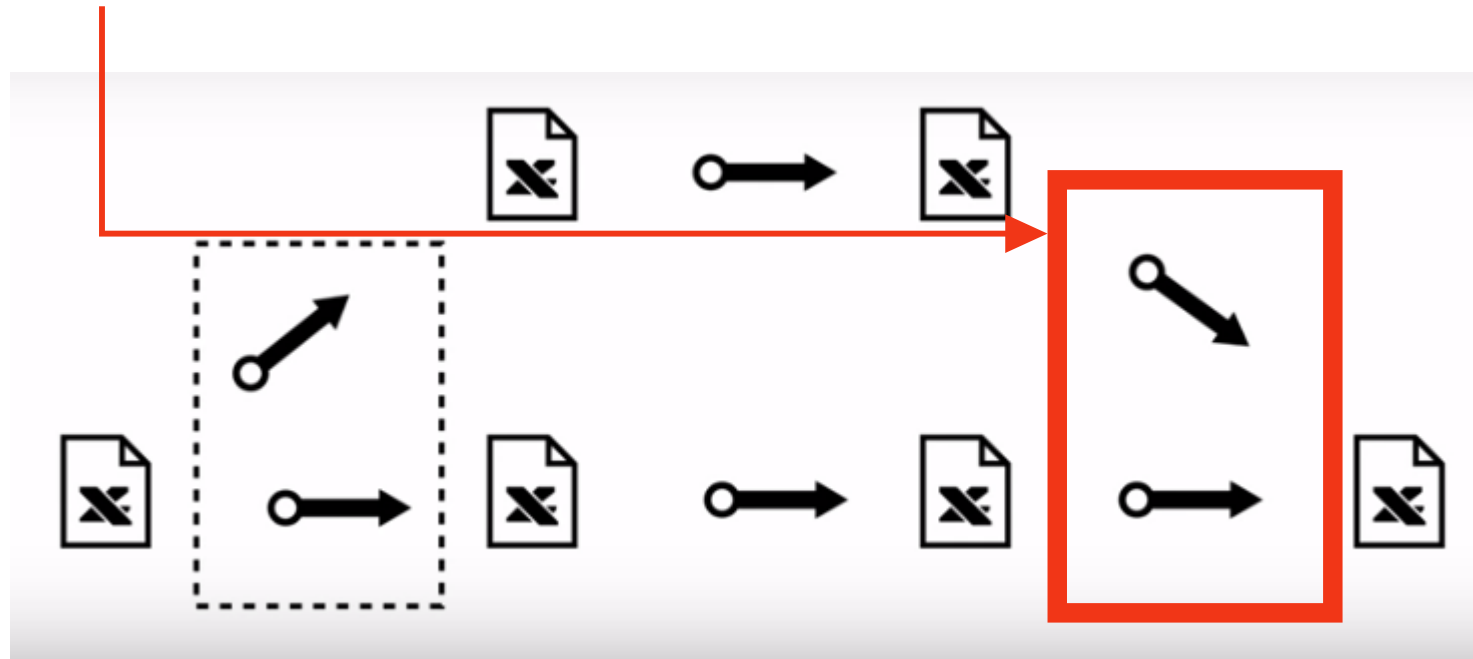
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git branch
* master
  shk

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git checkout master
Switched to branch 'master'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch
* master
  shk
```

Unit 03 | Github 사용하기

6 현재 가지와 merge(병합)하기



Unit 03 | Github 사용하기

6 현재 가지와 merge(병합)하기

만약 현재가지인 master와 합치려면

`git merge` 합치려는 브랜치명 실행

- 하지만 병합이 항상 성공하는게 아니라 가끔 충돌(conflicts)가 일어납니다.

이렇게 충돌이 발생하면 git이 알려주는 파일의 충돌 부분을 직접 수정해서 병합이 가능합니다. 충돌을 해결했다면 `git add 파일이름`으로 파일을 병합하라고 git에게 명령을 내려주세요.

Unit 03 | Github 사용하기

6 현재 가지와 merge(병합)하기

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git checkout master
Switched to branch 'master'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ cat abb.txt
cat: abb.txt: No such file or directory

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ ls
abc.txt

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git merge shk
Merge made by the 'recursive' strategy.
 abb.txt | 3 +++
 1 file changed, 3 insertions(+)
 create mode 100644 abb.txt

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ ls
abb.txt abc.txt
```

Unit 03 | Github 사용하기

7 master로 돌아온 후 만든 branch 제거

더 이상 만든 branch가 필요 없어서 제거하려면

`git branch -d 브랜치명` 실행

Unit 03 | Github 사용하기

7 master로 돌아온 후 만든 branch 제거

```
LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git checkout shk
Switched to branch 'shk'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git branch -d shk
error: Cannot delete branch 'shk' checked out at 'C:/Users/LG/tobig'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (shk)
$ git checkout master
Switched to branch 'master'

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch -d shk
Deleted branch shk (was bd22401).

LG@DESKTOP-RBH8J5B MINGW64 ~/tobig (master)
$ git branch
* master
```

Unit 04 | 과제

< 과제 >

팀장이 Github에 새로운 repository를 만들고, 모두 각자의 이름 branch를 생성해 **하나의 파일**을 팀원 모두 한번이상 수정 후 master에서 merge하기

<제출 방식>

제출은 datamarket에 팀원 중 한명이 각 조이름과 작업한 github주소 올리기!

ex) 1조, <https://github.com/aaa/TobigsTest> 입니다! 이런 식으로~

ex) 본인의 이름 branch,

현경 -> HyeonKyeong (0) , 철수 -> ChrisKim 이런 거 안되요.. (X)

Unit 04 | 과제

< 과제 >

팀 배정(팀장, 팀원, 팀원)

- 1조 – 강인구, 김동환, 박경빈
- 2조 – 박규리, 박성진, 신훈철
- 3조 – 이민주, 이준걸, 임진혁
- 4조 – 장유영, 장청아, 정윤호
- 5조 – 정하민, 최세영, 황이은
- 6조 – 백광제, 서석현
- 7조 – 신용재, 전민규, 정인호

Unit 04 | 과제

<도움될 만한 자료>

동영상 강의: <https://www.youtube.com/watch?v=hFJZw0fme6w>

더 다양한 Git 명령어: <https://medium.com/@joongwon/git-git-%EB%AA%85%EB%A0%B9%EC%96%B4-%EC%A0%95%EB%A6%AC-c25b421ecdbd>

Q & A

들어주셔서 감사합니다.