

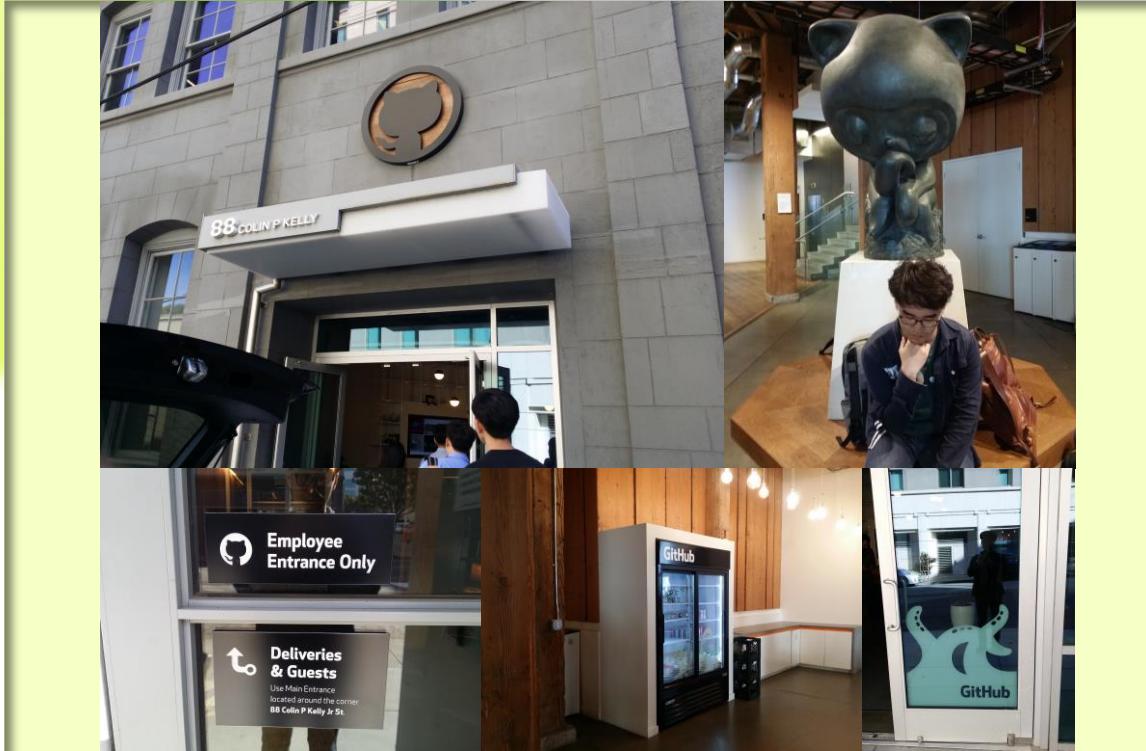
GitHub

박현민

Table of Contents

- Me
- Git
 - 설치 및 환경 설정
 - 커맨드 및 실습
- GitHub
 - 가입
 - 기능 및 실습
 - GitHub Desktop
- Reference

Me



박현민

hyeonmin.park@kennysoft.kr

CV: <http://cv.kennysoft.kr>



Chapter 1

GIT

Git이란???



- 깃(Git /gɪt/[3])은 프로그램 등의 소스 코드 관리를 위한 분산 버전 관리 시스템이다. 기하학적 불변 이론을 바탕으로 설계됐고, 빠른 수행 속도에 중점을 두고 있는 것이 특징이다. 최초에는 리눅스 토르발스가 리눅스 커널 개발에 이용하려고 개발하였으며, 현재는 다른 곳에도 널리 사용되고 있다.
- 깃의 작업 폴더는 모두, 전체 기록과 각 기록을 추적할 수 있는 정보를 포함하고 있으며, 완전한 형태의 저장소이다. 네트워크에 접근하거나 중앙 서버에 의존하지 않는다.
- 현재 주니오 하마노(Junio Hamano)가 소프트웨어 관리를 감독하고 있다. 깃은 GNU 일반 공중 사용 허가서 v2 하에 배포되는 자유 소프트웨어이다.
- 역사[편집]
- 깃의 개발은 리눅스 커널의 수많은 개발자들이 한때 자신들의 프로젝트의 유지보수를 위해 사용하였던 자유 소스 관리 시스템(SCM)인 비트키퍼에 대한 접근을 포기한 이후인 2005년 4월에 시작되었다.[4]

Git이란?

DVCS: 분산 버전 관리 시스템
(Distributed Version Control System)

Git이란? (*Cont'd*)

- **분산**

모든 곳에 동일한 내용이 분산되어 보관

- **버전 관리**

각 버전의 히스토리를 가지고 있으며 과거 상태로 복구 가능

Why Git?

■ When I was young...

이름	수정한 날짜	유형	크기
110528.zip	2015-04-20 노선...	압축(ZIP) 파일	4,590KB
110625_경원대_java.zip	2015-04-20 오전...	압축(ZIP) 파일	4,361KB
ANDROID 확장 - KENNYCAMERA Import.hwp	2015-04-20 오전...	한컴오피스 2018 ...	1,342KB
AVR FPGA Line Tracer (Source).show	2012-12-08 오전...	한컴오피스 2018 ...	681KB
AVR과 FPGA 연동하기.show	2013-01-12 오전...	한컴오피스 2018 ...	669KB
AVR과 FPGA를 이용한 라인트레이서 (가전대 Script v2).hwp	2013-02-02 오전...	한컴오피스 2018 ...	16KB
AVR과 FPGA를 이용한 라인트레이서 (가전대 논문 BETA).hwp	2013-01-26 오후...	한컴오피스 2018 ...	4,946KB
AVR과 FPGA를 이용한 라인트레이서 (종남대 Script).hwp	2013-02-01 오후...	한컴오피스 2018 ...	24KB
AVR과 FPGA를 이용한 라인트레이서 (종남대 논문 제출본).hwp	2013-01-14 오후...	한컴오피스 2018 ...	2,501KB
AVR과 FPGA를 이용한 라인트레이서 (포스터 v1).pptx	2013-01-26 오후...	Microsoft PowerP...	2,267KB
AVR과 FPGA를 이용한 라인트레이서 (포스터 v2).pptx	2013-01-28 오전...	Microsoft PowerP...	2,717KB
AVR과 FPGA를 이용한 라인트레이서 (포스터 v3).pptx	2013-01-28 오후...	Microsoft PowerP...	2,642KB
AVR과 FPGA를 이용한 라인트레이서 v1.pptx	2013-01-16 오전...	Microsoft PowerP...	1,909KB
AVR과 FPGA를 이용한 라인트레이서 v2.show	2013-01-17 오후...	한컴오피스 2018 ...	2,258KB
AVR과 FPGA를 이용한 라인트레이서 v3.show	2013-01-18 오전...	한컴오피스 2018 ...	2,188KB
AVR과 FPGA를 이용한 라인트레이서 v4.pptx	2013-01-26 오후...	Microsoft PowerP...	2,279KB
AVR과 FPGA를 이용한 라인트레이서 v5.pptx	2013-02-01 오후...	Microsoft PowerP...	15,171KB
AVR과 FPGA를 이용한 라인트레이서 v6.pptx	2013-02-01 오후...	Microsoft PowerP...	15,170KB
AVR과 FPGA를 이용한 라인트레이서 v7.pptx	2013-02-01 오후...	Microsoft PowerP...	15,473KB
AVR과 FPGA를 이용한 라인트레이서 v8.pptx	2013-02-02 오전...	Microsoft PowerP...	15,425KB
AVR과 FPGA를 이용한 라인트레이서 v9.pptx	2013-02-02 오전...	Microsoft PowerP...	14,527KB
AVR과 FPGA를 이용한 라인트레이서 v10 (FINAL).pptx	2013-02-27 오후...	Microsoft PowerP...	14,527KB
AVR과 FPGA를 이용한 라인트레이서.pptx	2015-04-20 오전...	Microsoft PowerP...	1,912KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (김민재 BETA).pptx	2013-01-12 오전...	Microsoft PowerP...	111KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (김민재).hwp	2013-01-13 오후...	한컴오피스 2018 ...	1,089KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (김민재).pptx	2013-01-15 오후...	Microsoft PowerP...	883KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (박현민 BETA).hwp	2013-01-12 오후...	한컴오피스 2018 ...	767KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (박현민).hwp	2013-01-14 오전...	한컴오피스 2018 ...	1,273KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (박현민).pptx	2013-01-15 오전...	Microsoft PowerP...	331KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (이성록).pptx	2013-01-15 오후...	Microsoft PowerP...	505KB
AVR과 FPGA를 이용한 지능형 로봇(라인트레이서) 설계 (조유현).pptx	2013-01-15 오후...	Microsoft PowerP...	958KB
CD.zip	2012-11-03 오후...	압축(ZIP) 파일	125,376KB
Line_Tracer.c	2007-11-09 오후...	C Source file	5KB
경원대영재교육원고.jpg	2014-04-20 오후...	JPG 파일	124KB
과학영재교육원 15회 뉴스레터.PDF	2014-04-16 오후...	Adobe Acrobat 문서	4,714KB

47개 항목 1개 항목 선택할 14.1MB

Why Git? (*Cont'd*)

버전이 올라가도 동일한 파일을 계속 복사하는 것은 낭비

수정 이유도 함께 저장 가능

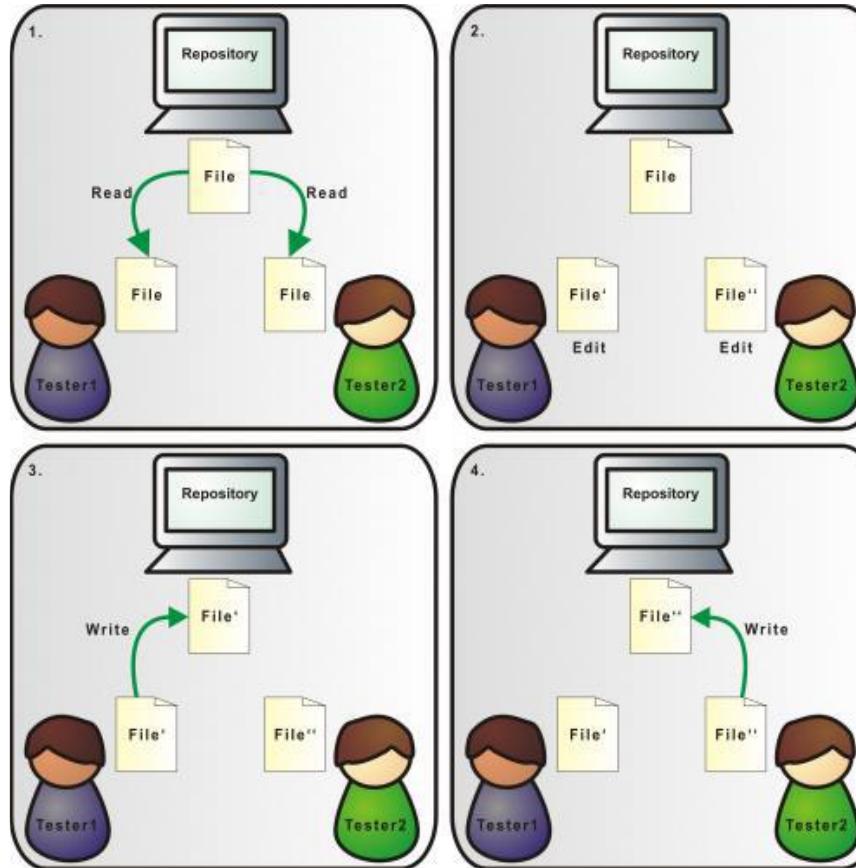
Why Git? (2)

- 백업



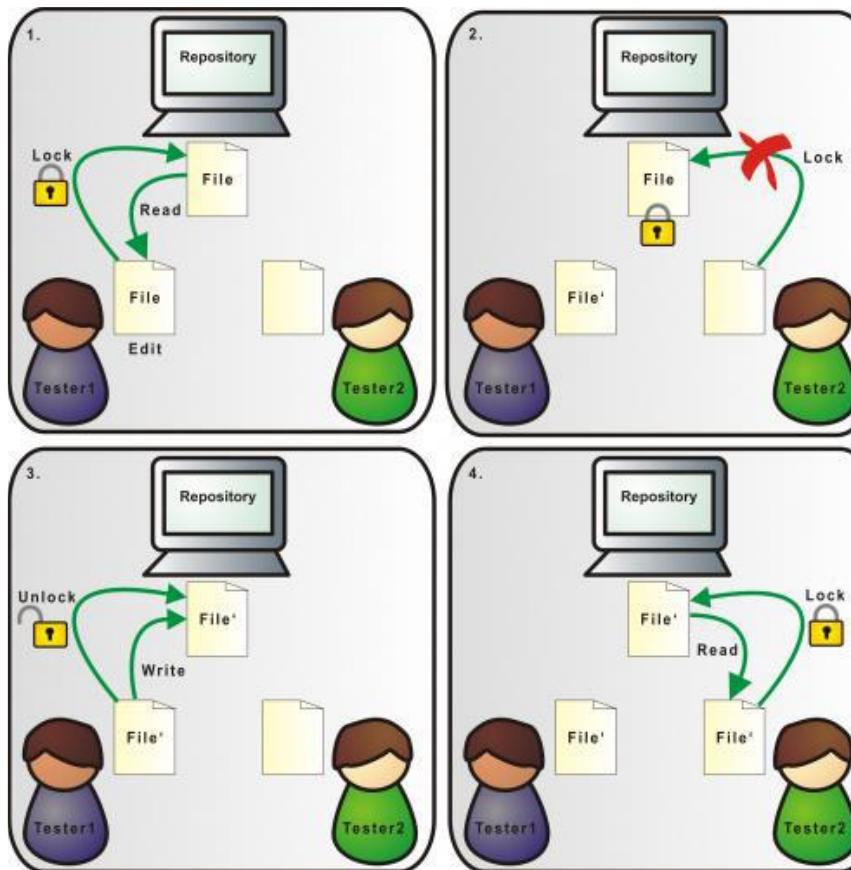
Why Git? (3)

- 협업: Overwritten problem



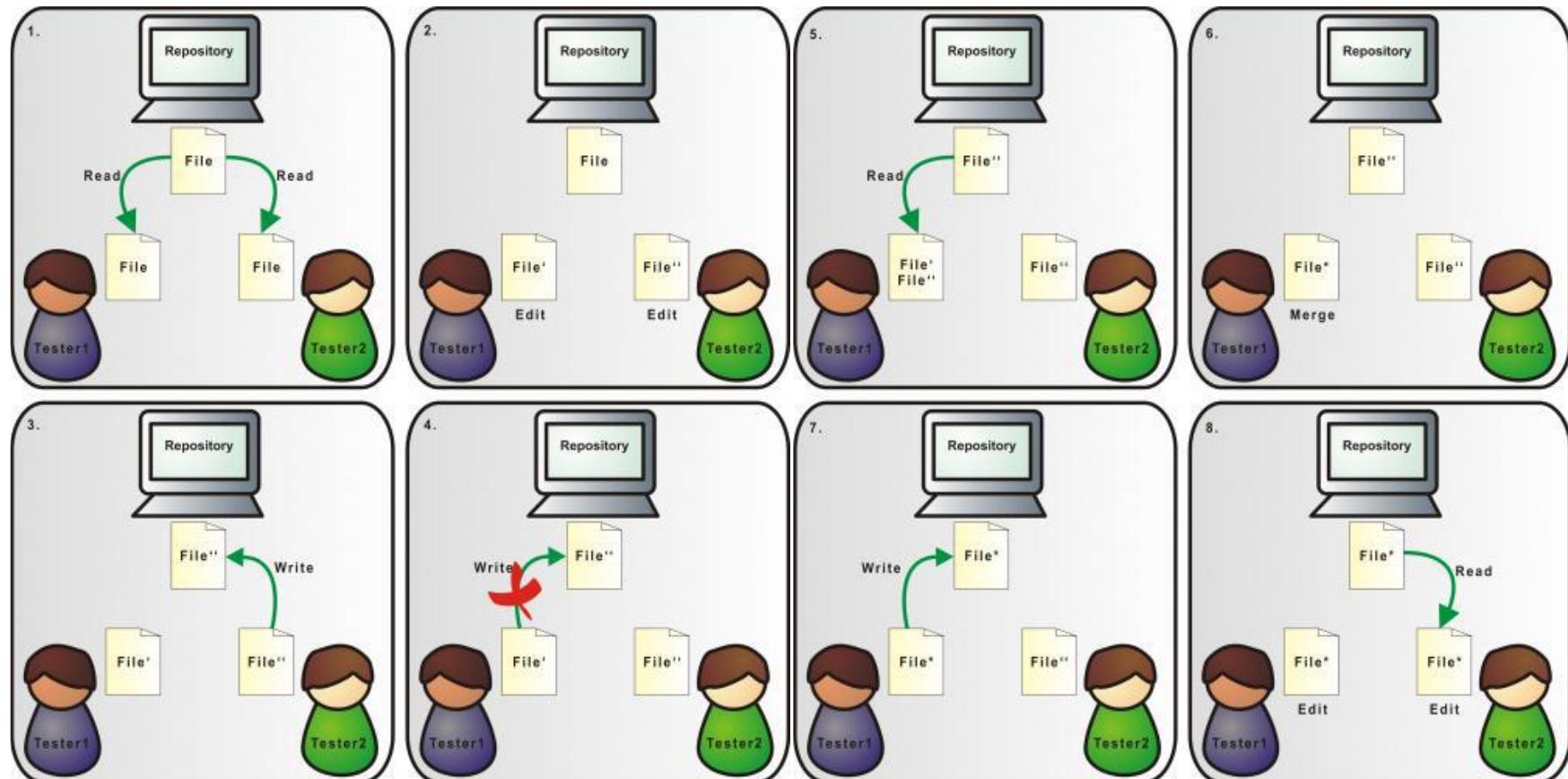
Why Git? (3) (Cont'd)

- 협업: Lock-Modify-Unlock approach

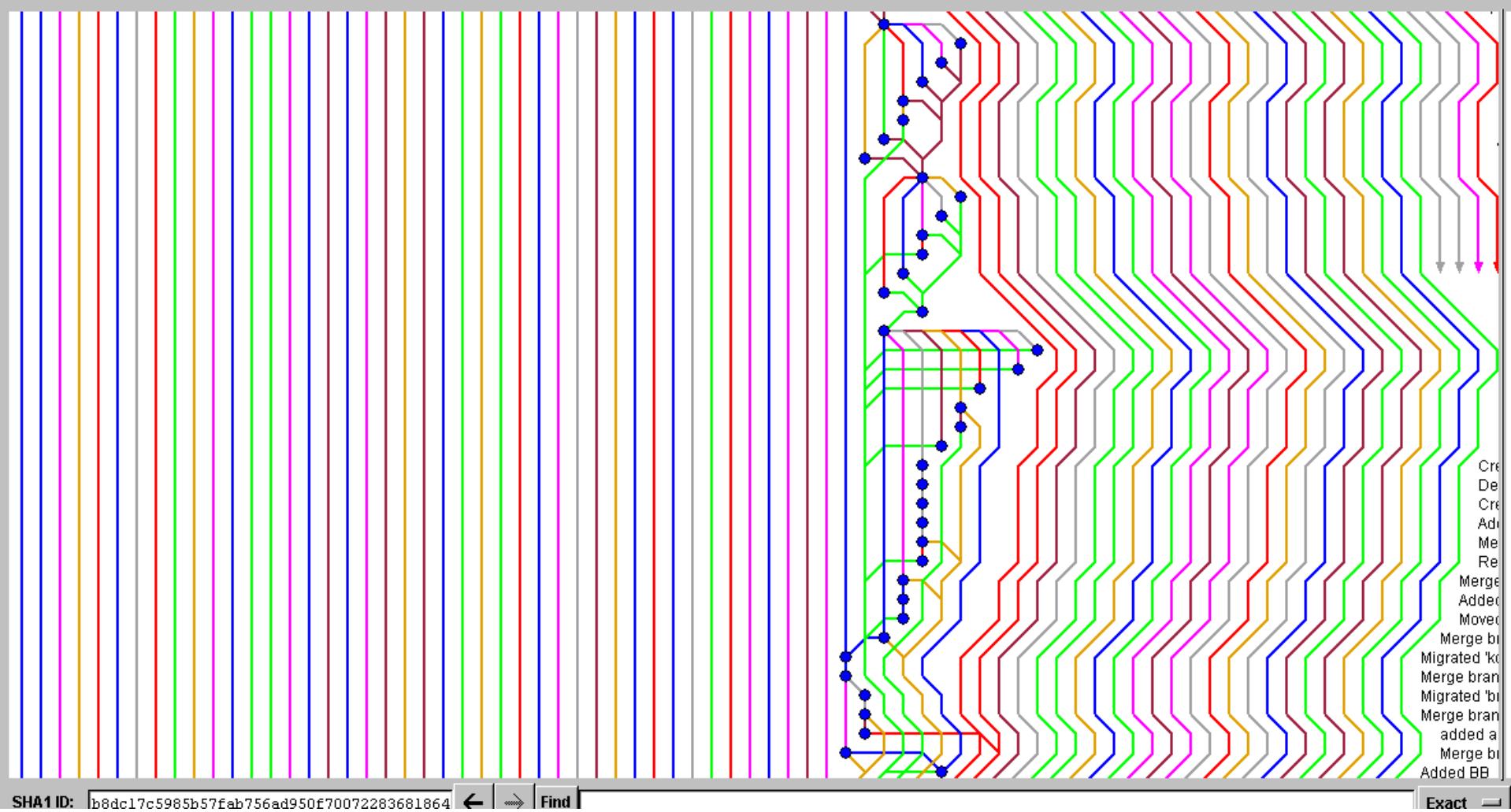


Why Git? (3) (Cont'd)

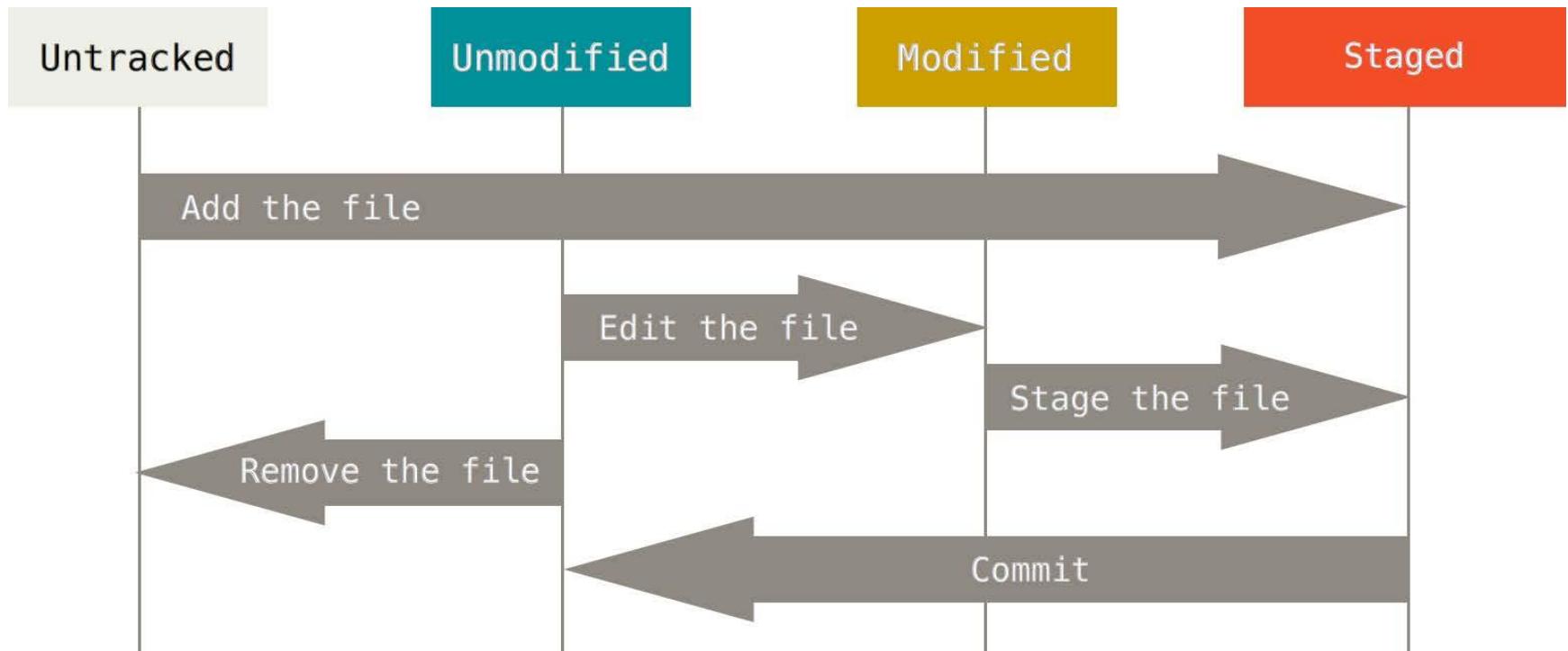
- 협업: Copy-Modify-Merge approach



Don't be afraid



기본 개념





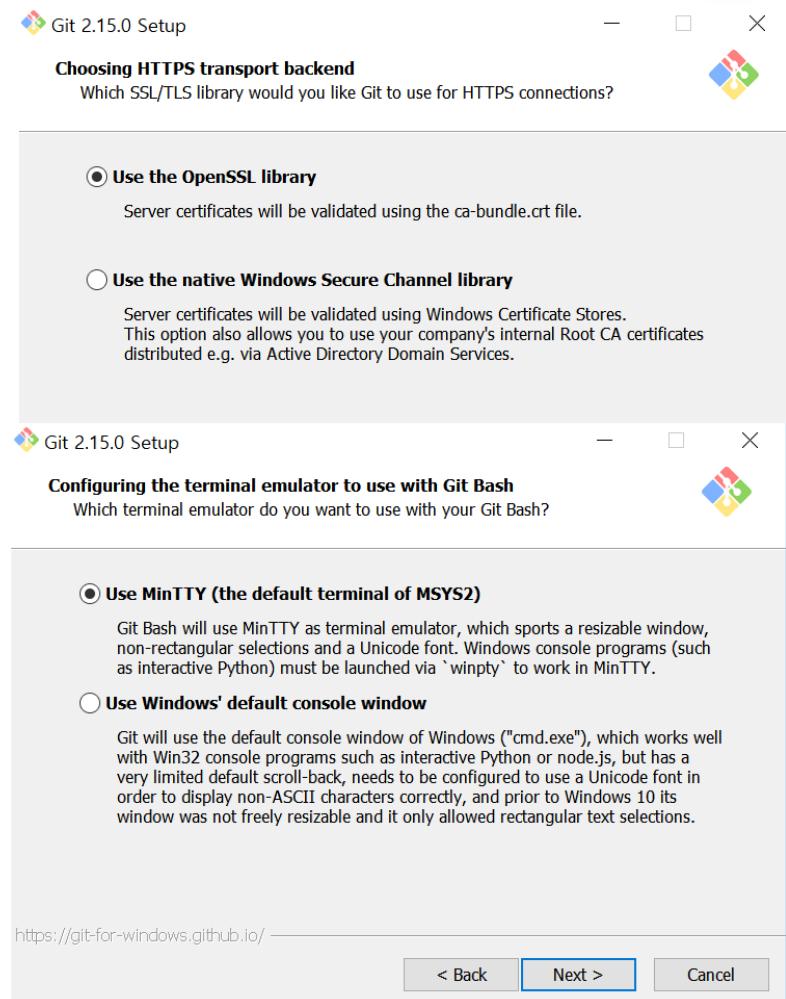
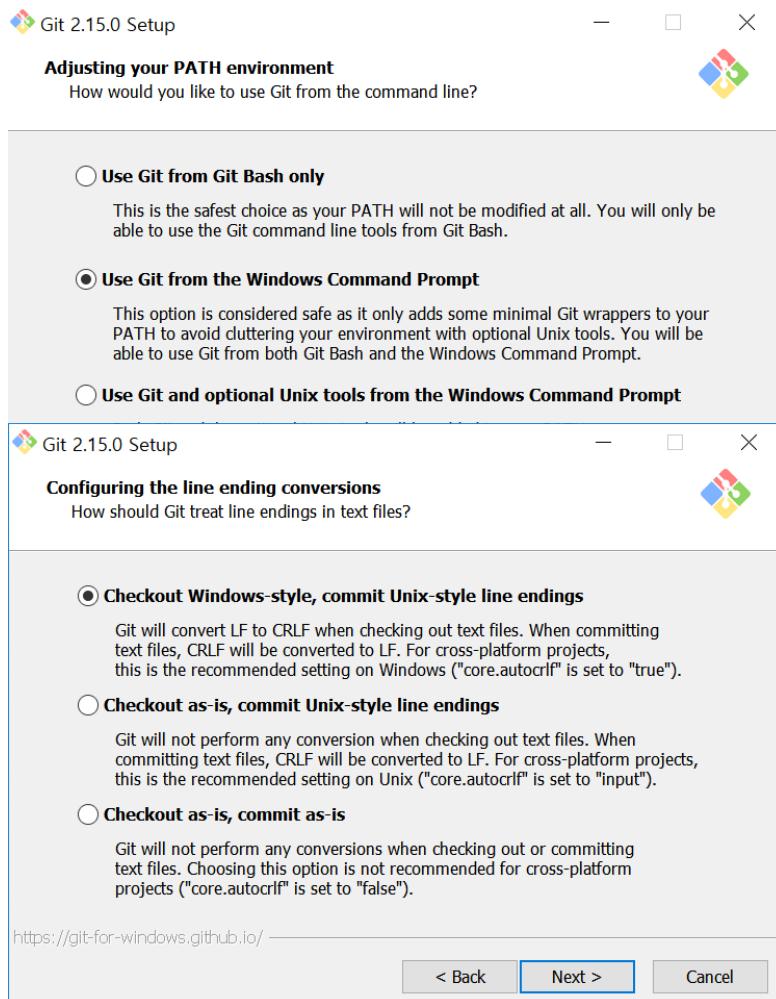
Chapter 1-1

GIT: 설치 및 환경 설정

git-for-windows 설치

<https://git-scm.com/download/win>

git-for-windows 설치 (Cont'd)



Git Version

2.30부터 초기 branch명의 master 하드코딩 제거

환경 설정

git config 커맨드를 통해 진행

이제부터 Windows Terminal을 사용한다.

(n): git 명령어 설명

[n]: 실습 (따라해보기)

(0) config



```
C:\git\github-hol>git config --global --list
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
user.name=Hyeonmin Park
user.email=hyeonmin.park@kennysoft.kr
diff.tool.sourcetree.cmd='''"$LOCAL" "$REMOTE"
mergetool.sourcetree.cmd='''mergetool.sourcetree.trustexitcode=true
commit.gpgsign=false
init.defaultbranch=main
rerere.enabled=true

C:\git\github-hol>git config --list
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
```

설정이 저장되는 곳은 --global 또는 저장소

--list로 저장된 값 확인

속성 설정: (속성 이름) (속성 값)

[0] 계정 정보 설정

- git config --global user.name "Hyeonmin Park"
- git config --global user.email "hyeonmin.park@kennyssoft.kr"

이름과 이메일을 자기 것으로 수정!

도용 가능성 ⇒ GPG



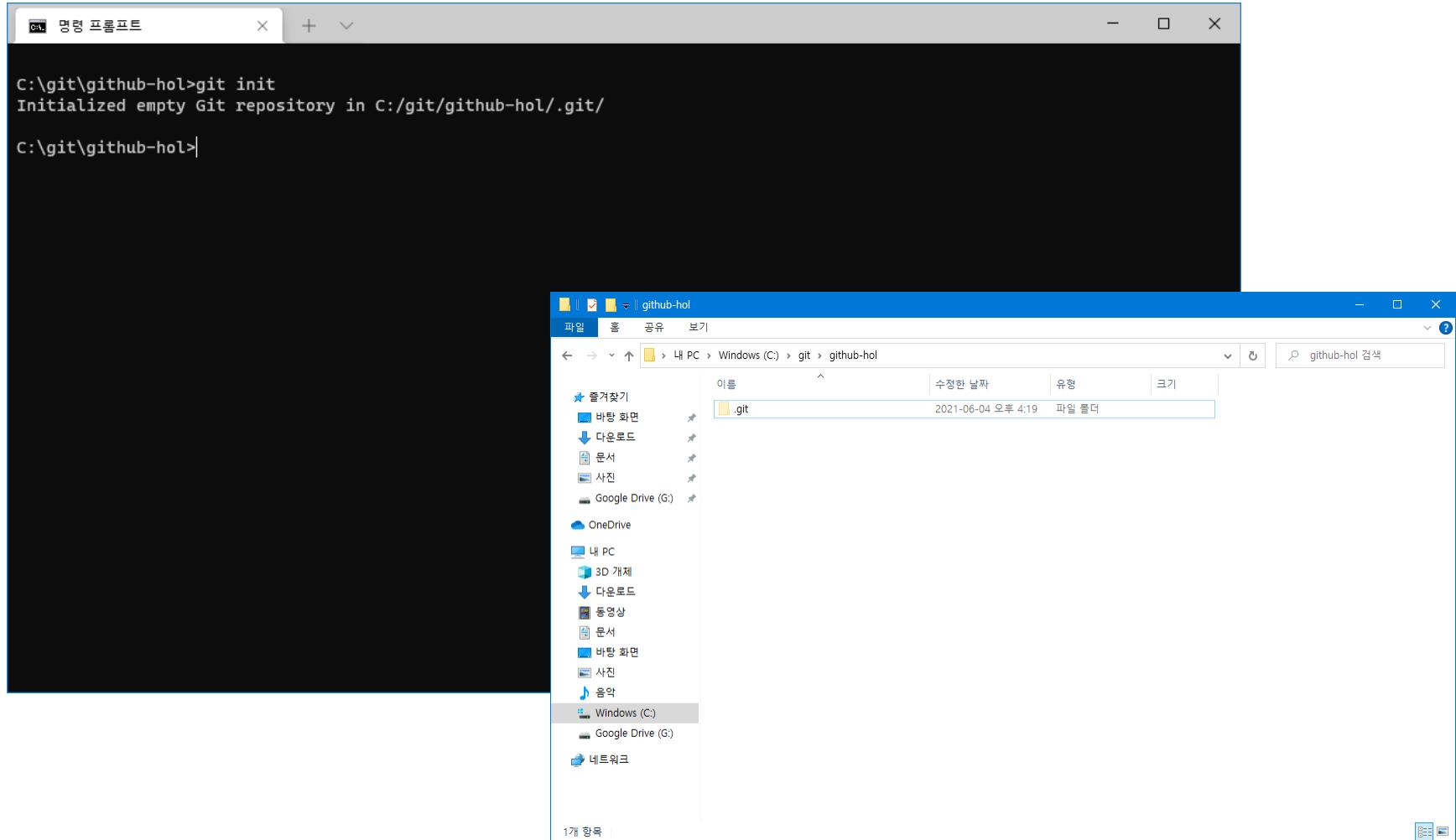
Chapter 1-2

GIT: 커맨드 및 실습

Providing

<https://gist.github.com/KENNYSOFT/4cf28c5d09382553e18c8eae11f134e7>

(1) init



(2) status

```
C:\git\github-hol>git init
Initialized empty Git repository in C:/git/github-hol/.git/
C:\git\github-hol>git status
On branch main

No commits yet

nothing to commit (create/copy files and use "git add" to track)

C:\git\github-hol>|
```

(2) status (*Cont'd*)

```
명령 프롬프트
No commits yet
nothing to commit (create/copy files and use "git add" to track)
C:\git\github-hol>echo "# GitHub-HoL" > README.md

C:\git\github-hol>dir
 Volume in drive C is Windows
 Volume Serial Number is CAE6-1484

 Directory of C:\git\github-hol

2021-06-04 오후 04:21    <DIR>      .
2021-06-04 오후 04:21    <DIR>      ..
2021-06-04 오후 04:21                17 README.md
              1 File(s)       17 bytes
              2 Dir(s)  31,118,942,208 bytes free

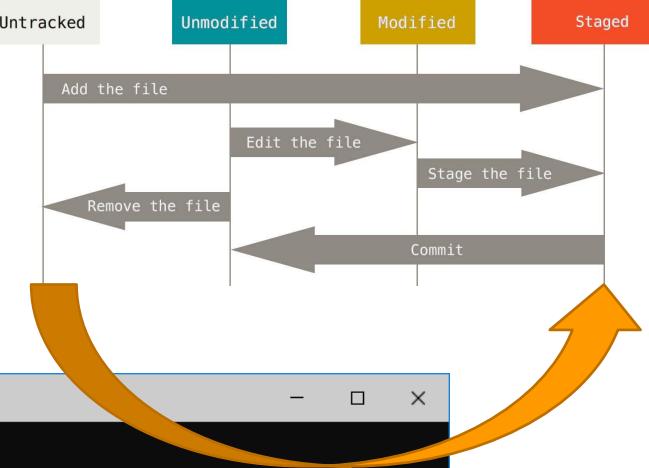
C:\git\github-hol>git status
On branch main

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)
C:\git\github-hol>|
```

(3) add



```
명령 프롬프트
2021-06-04 오후 04:21 <DIR> .
2021-06-04 오후 04:21 <DIR> ..
2021-06-04 오후 04:21 17 README.md
    1 File(s) 17 bytes
    2 Dir(s) 31,118,942,208 bytes free

C:\git\github-hol>git status
On branch main

No commits yet

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

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

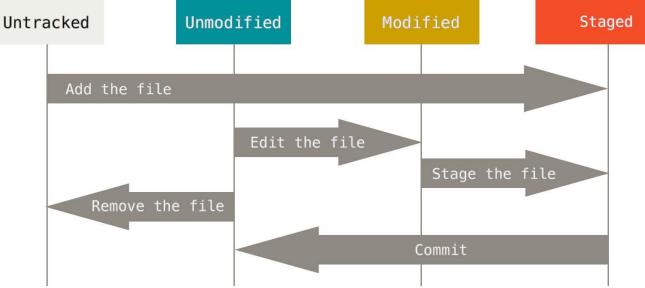
C:\git\github-hol>git add README.md

C:\git\github-hol>git status
On branch main

No commits yet

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

C:\git\github-hol>|
```



(3) add (Cont'd)

```

명령 프롬프트
c:\git\github-hol>echo "# GitHub: Hands-On Lab" > README.md 내용 조금 수정
c:\git\github-hol>git status
On branch main

No commits yet

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

Changes not staged for commit:
  (use "git add <file> ..." to update what will be committed)
  (use "git restore <file> ..." to discard changes in working directory)
    modified:  README.md

c:\git\github-hol>git add .
편하게 모든 파일을 추가할 수 있다.

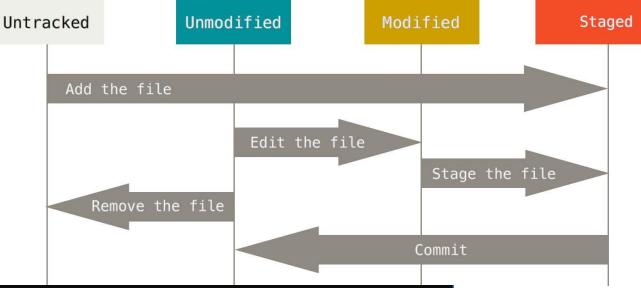
c:\git\github-hol>git status
On branch main

No commits yet

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

c:\git\github-hol>

```



(2) status -s

```
C:\git\github-hol>git status
On branch main

No commits yet

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

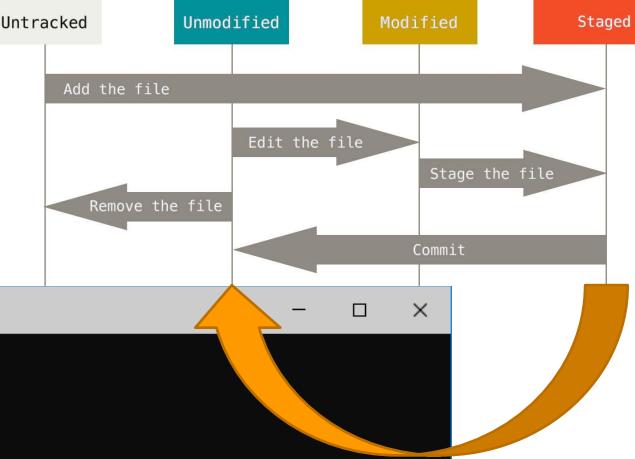
C:\git\github-hol>git status -s
A README.md

C:\git\github-hol>echo "# GitHub: Hands-On Lab (2021-06-04)" > README.md

C:\git\github-hol>git status -s
AM README.md
```

- [Staging area의 상태][Working tree의 상태] 파일 이름
 - ??: untracked files
 - A: new files that have been added to the staged area
 - M: modified and staged
 - U: modified but not yet staged
 - MM: modified, staged, and then modified again

(4) commit



```
C:\ 명령 프롬프트 x + v
(use "git restore <file> ..." to discard changes in working directory)
  modified: README.md

C:\git\github-hol>git add .
C:\git\github-hol>git status
On branch main

No commits yet

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

C:\git\github-hol>git status -s
A README.md

C:\git\github-hol>echo "# GitHub: Hands-On Lab (2021-06-04)" > README.md
C:\git\github-hol>git status -s
AM README.md

C:\git\github-hol>git commit -m "Initial commit"
[master (root-commit) daef4c7] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

C:\git\github-hol>|
```

-m 옵션은 필수는 아니지만 매우 강력히 추천한다.

(5) log

```
명령 프롬프트 + X
On branch main
No commits yet

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

C:\git\github-hol>git status -s
A README.md

C:\git\github-hol>echo "# GitHub: Hands-On Lab (2021-06-04)" > README.md

C:\git\github-hol>git status -s
AM README.md

C:\git\github-hol>git commit -m "Initial commit"
[main (root-commit) daef4c7] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

C:\git\github-hol>git log
commit daef4c7422a1f3043b15f2cc8022160d29e5bb7b (HEAD -> main)
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 16:24:54 2021 +0900

  Initial commit

C:\git\github-hol>
```

[1] 저장소 초기화 및 첫 커밋 만들기

- md github-hol
- cd github-hol
- git init
- echo "# GitHub-HoL" > README.md
- git status
- git add README.md
- echo "# GitHub: Hands-On Lab (2021-06-04)" > README.md
- git status
- git status -s
- git add .
- git status
- git commit -m "Initial commit"
- git log

(6) diff

```
명령 프롬프트 + ▾ C:\git\github-hol>git log
create mode 100644 README.md

C:\git\github-hol>git log
commit daef4c7422a1f3043b15f2cc8022160d29e5bb7b (HEAD → main)
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 16:24:54 2021 +0900

    Initial commit

C:\git\github-hol>git status
On branch main
Changes not staged for commit:
  (use "git add <file> ..." to update what will be committed)
  (use "git restore <file> ..." to discard changes in working directory)
    modified:   README.md

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

C:\git\github-hol>git diff
diff --git a/README.md b/README.md
index 11362ee..6279e40 100644
--- a/README.md
+++ b/README.md
@@ -1 +1,3 @@
- "# GitHub: Hands-On Lab"
+ "# GitHub: Hands-On Lab (2021-06-04)
+
+Thanks to HSpace

C:\git\github-hol>
```

echo를 잘못 사용했다.
설명 추가

(6) diff --cached

```
명령 프롬프트 + X
+++ b/README.md
@@ -1 +1,3 @@
-"># GitHub: Hands-On Lab"
+## GitHub: Hands-On Lab (2021-06-04)
+
+Thanks to HSpace

C:\git\github-hol>git add .

C:\git\github-hol>git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   README.md

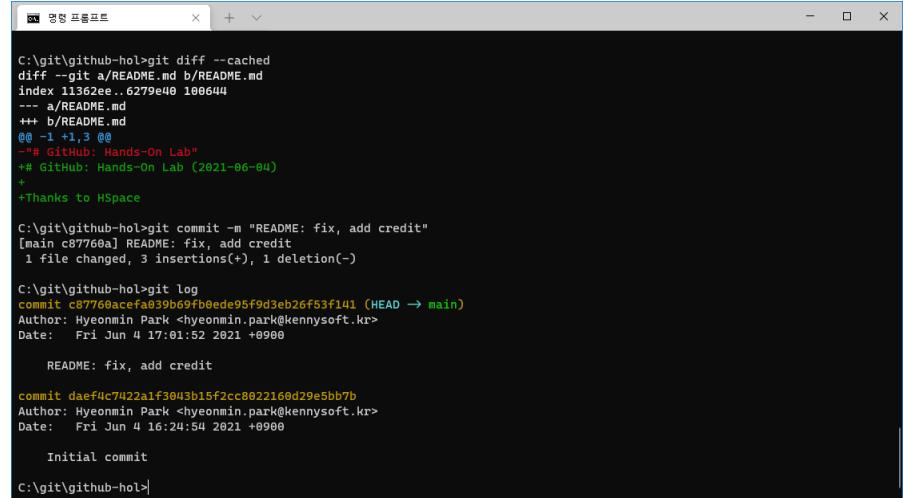
C:\git\github-hol>git diff

C:\git\github-hol>git diff --cached
diff --git a/README.md b/README.md
index 11362ee..6279e40 100644
--- a/README.md
+++ b/README.md
@@ -1 +1,3 @@
-"># GitHub: Hands-On Lab"
+## GitHub: Hands-On Lab (2021-06-04)
+
+Thanks to HSpace

C:\git\github-hol>|
```

[2] 새로운 커밋 만들기

- Level2 폴더 내의 README.md를 저장소로 복사한다. (덮어쓰기)
- git status
- git diff
- git add .
- git status
- git diff
- git diff --cached
- git commit -m "README: fix, add credit"
- git log



```
C:\git\github-hol>git diff --cached
diff --git a/README.md b/README.md
index 11362ee..6279e08 100644
--- a/README.md
+++ b/README.md
@@ -1 +1,3 @@
-# GitHub: Hands-On Lab
+## GitHub: Hands-On Lab (2021-06-04)
+
+Thanks to HSpace

C:\git\github-hol>git commit -m "README: fix, add credit"
[main c87760a] README: fix, add credit
 1 file changed, 3 insertions(+), 1 deletion(-)

C:\git\github-hol>git log
commit c87760acefa030b69fb0ede95f9d3eb26f53f141 (HEAD -> main)
Author: Hyeonmin Park <yeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 17:01:52 2021 +0900

    README: fix, add credit

commit daef4fc7422a1f3083b15f2cc8022160d29e5bb7b
Author: Hyeonmin Park <yeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 16:24:54 2021 +0900

    Initial commit

C:\git\github-hol>
```

(5) log --oneline, --graph

```
명령 프롬프트
README: fix, add credit

commit daef4c7422a1f3043b15f2cc8022160d29e5bb7b
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 16:24:54 2021 +0900

    Initial commit

C:\git\github-hol>git log --oneline
c87760a (HEAD -> main) README: fix, add credit
daef4c7 Initial commit

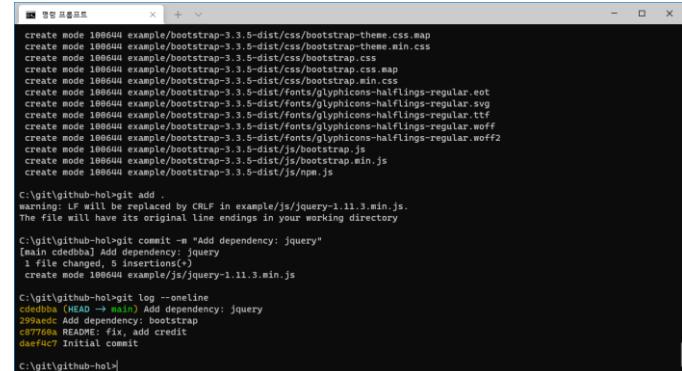
C:\git\github-hol>git log --graph
* commit c87760acea039b69fb0ede95f9d3eb26f53f141 (HEAD -> main)
| Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
| Date:   Fri Jun 4 17:01:52 2021 +0900
|
| README: fix, add credit
|
* commit daef4c7422a1f3043b15f2cc8022160d29e5bb7b
| Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
| Date:   Fri Jun 4 16:24:54 2021 +0900
|
|     Initial commit

C:\git\github-hol>git log --oneline --graph
* c87760a (HEAD -> main) README: fix, add credit
* daef4c7 Initial commit

C:\git\github-hol>
```

[3] 또 커밋 만들기

- Level3 폴더 내의 example\bootstrap-3.3.5-dist 폴더를 저장소로 복사한다.
- git add .
- git commit -m "Add dependency: bootstrap"
- Level3 폴더 내의 example\js 폴더를 저장소로 복사한다.
- git add .
- git commit -m "Add dependency: jquery"
- git log --oneline



```
C:\git\github-hol>git add .
warning: LF will be replaced by CRLF in example/js/jquery-1.11.3.min.js.
The file will have its original line endings in your working directory
C:\git\github-hol>git commit -m "Add dependency: jquery"
[main ceddbba] Add dependency: jquery
 1 file changed, 6 insertions(+)
create mode 100644 example/js/jquery-1.11.3.min.js

C:\git\github-hol>git log --oneline
ceddbba (HEAD -> main) Add dependency: jquery
299adec Add dependency: bootstrap
C:\git\github-hol>git commit -m "Initial commit"
daefact Initial commit

C:\git\github-hol>
```

[4] 계속 커밋 만들기

- Level4 폴더 내의 example 폴더를 저장소로 복사한다.
- git add .
- git commit -m "Add implementation"
- example\ajax\db_connection.php 파일을 열고 6번째 줄의 \$password 값을 "password1234" 등 임의로 변경한다.

현재 시나리오



```
C:\git\github-hol>git status
On branch main
Changes not staged for commit:
  (use "git add <file> ..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   example/ajax/db_connection.php

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

C:\git\github-hol>git diff
diff --git a/example/ajax/db_connection.php b/example/ajax/db_connection.php
index e23ecd8..e978786 100644
--- a/example/ajax/db_connection.php
+++ b/example/ajax/db_connection.php
@@ -3,7 +3,7 @@
 // Connection variables
 $host = ""; // MySQL host name eg. localhost
 $user = ""; // MySQL user. eg. root ( if your on localserver )
-$password = ""; // MySQL user password (if password is not set for your root user then keep it empty )
+$password = "password1234"; // MySQL user password (if password is not set for your root user then keep it empty )
 $database = "test"; // MySQL Database name

 // Connect to MySQL Database
C:\git\github-hol>
```

내 비밀번호를 입력했으니 저 파일은 커밋하면 안 되겠군!

깃에서 파일을 지워버리자.

(7) rm



```
C:\git\github-hol>git status
On branch main
Changes not staged for commit:
  (use "git add <file> ..." to update what will be committed)
  (use "git restore <file> ..." to discard changes in working directory)
    modified:   example/ajax/db_connection.php

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

C:\git\github-hol>git diff
diff --git a/example/ajax/db_connection.php b/example/ajax/db_connection.php
index e23ecd8..e978786 100644
--- a/example/ajax/db_connection.php
+++ b/example/ajax/db_connection.php
@@ -3,7 +3,7 @@
 // Connection variables
 $host = ""; // MySQL host name eg. localhost
 $user = ""; // MySQL user. eg. root ( if your on localserver )
-$password = ""; // MySQL user password (if password is not set for your root user then keep it empty )
+$password = "password1234"; // MySQL user password (if password is not set for your root user then keep it empty )
 $database = "test"; // MySQL Database name

 // Connect to MySQL Database

C:\git\github-hol>git rm example/ajax/db_connection.php
error: the following file has local modifications:
  example/ajax/db_connection.php
(use --cached to keep the file, or -f to force removal)

C:\git\github-hol>
```

로컬 변경점이 있어서 파일을 삭제할 수 없다고 한다.

일단 -f 옵션을 줘서 지워보자.

(7) rm (*Cont'd*)



```
cmd 명령 프롬프트 x + v
(use "git add <file> ..." to update what will be committed)
(use "git restore <file> ..." to discard changes in working directory)
  modified:   example/ajax/db_connection.php

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

C:\git\github-hol>git diff
diff --git a/example/ajax/db_connection.php b/example/ajax/db_connection.php
index e23ecd8..e978786 100644
--- a/example/ajax/db_connection.php
+++ b/example/ajax/db_connection.php
@@ -3,7 +3,7 @@
 // Connection variables
 $host = ""; // MySQL host name eg. localhost
 $user = ""; // MySQL user. eg. root ( if your on localserver )
-$password = ""; // MySQL user password (if password is not set for your root user then keep it empty )
+$password = "password1234"; // MySQL user password (if password is not set for your root user then keep it empty )
 $database = "test"; // MySQL Database name

 // Connect to MySQL Database

C:\git\github-hol>git rm example/ajax/db_connection.php
error: the following file has local modifications:
  example/ajax/db_connection.php
(use --cached to keep the file, or -f to force removal)

C:\git\github-hol>git rm -f example/ajax/db_connection.php
rm 'example/ajax/db_connection.php'

C:\git\github-hol>
```

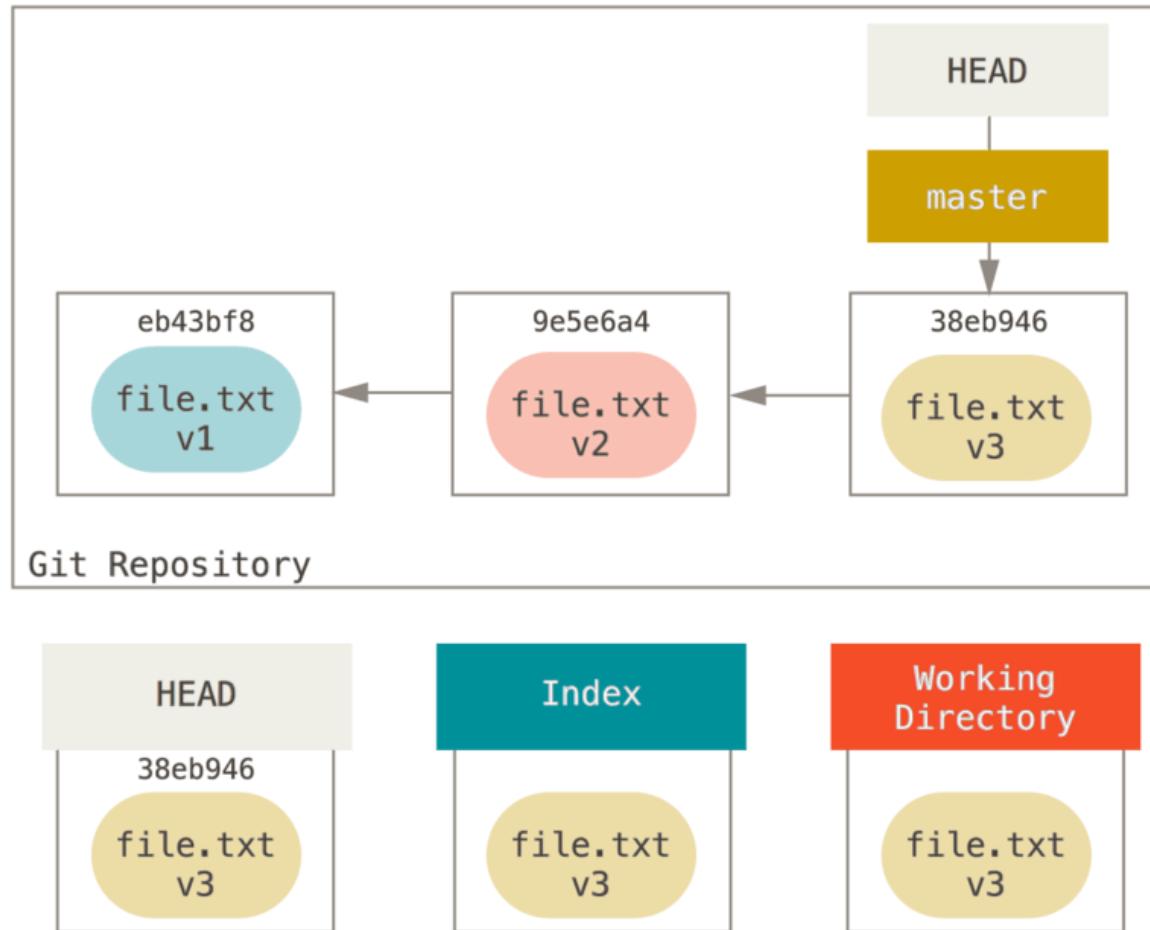
진짜 파일도 같이 지워졌다!

파일 복구

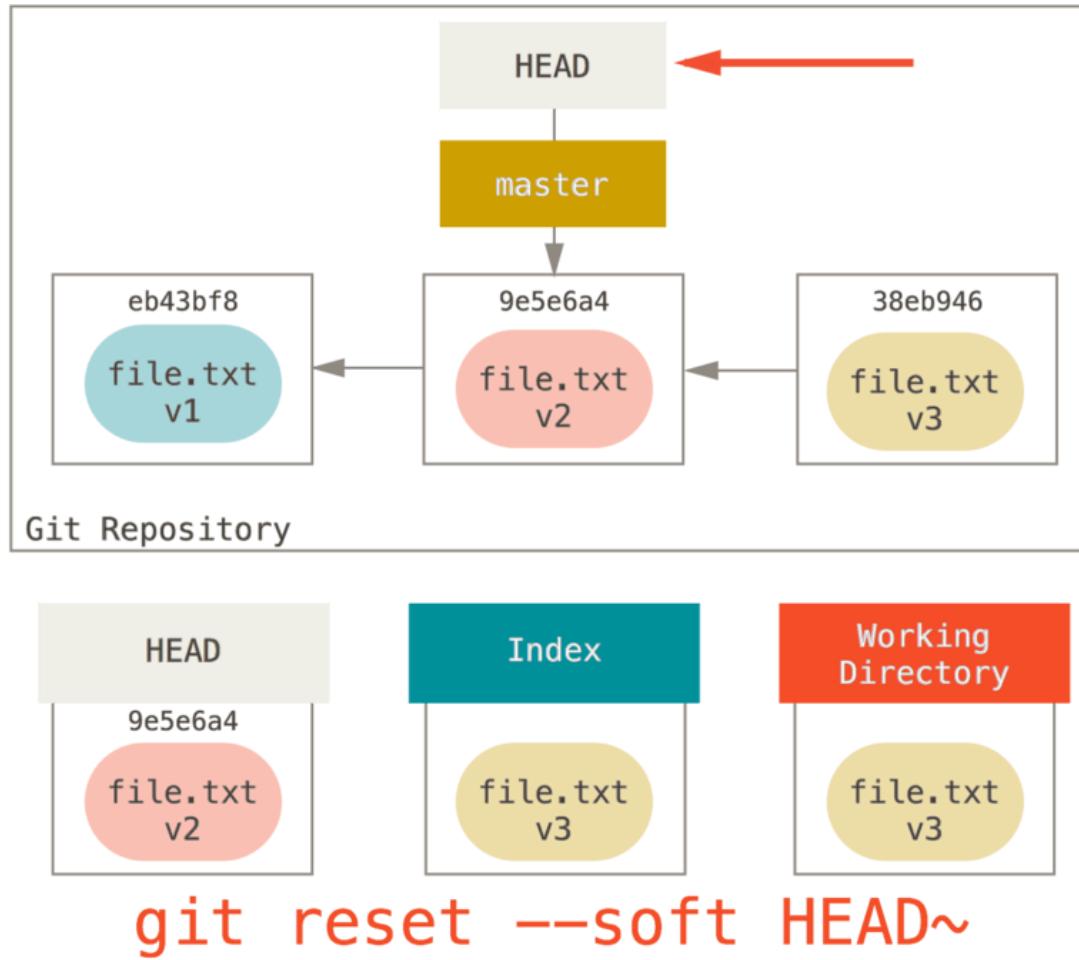
최근 커밋의 상태로 돌아가보자.

이후 슬라이드에서 file.txt는 db_connection.php,
v3은 삭제된 상태라고 생각하면 된다.

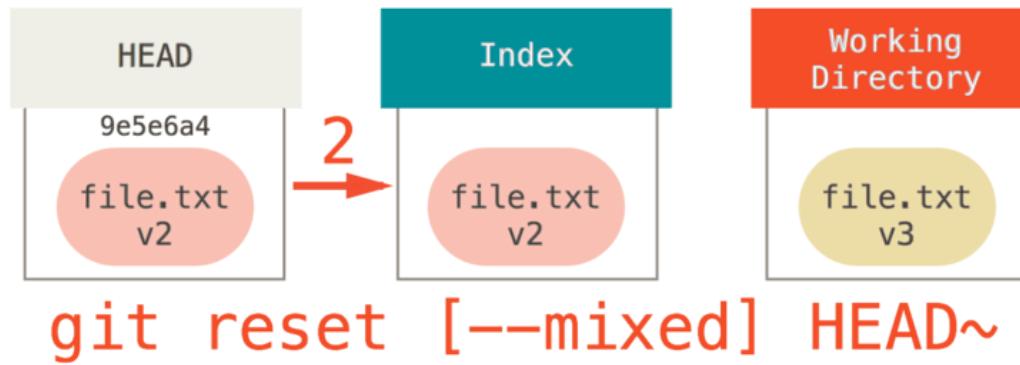
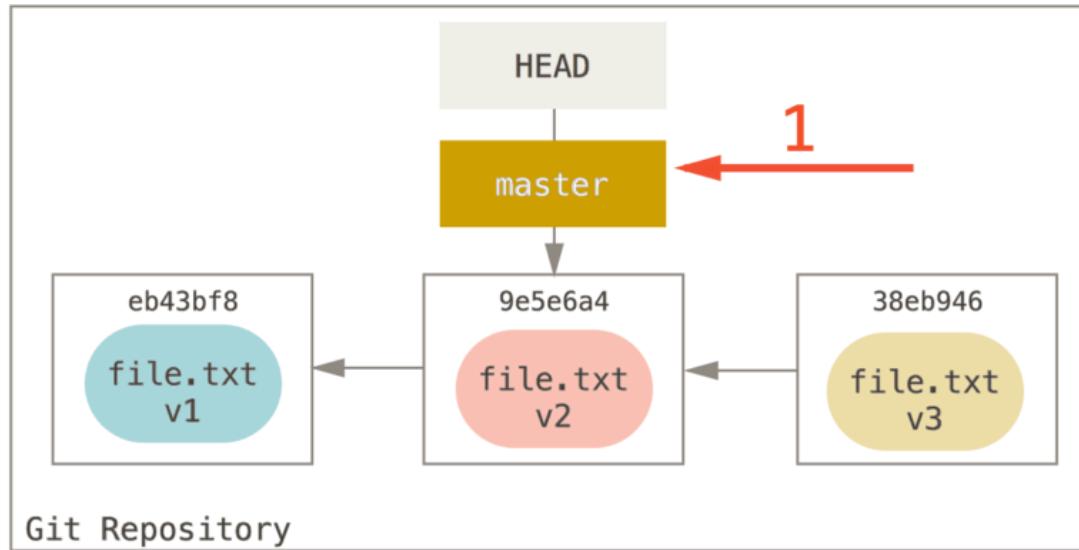
파일 상태 예시



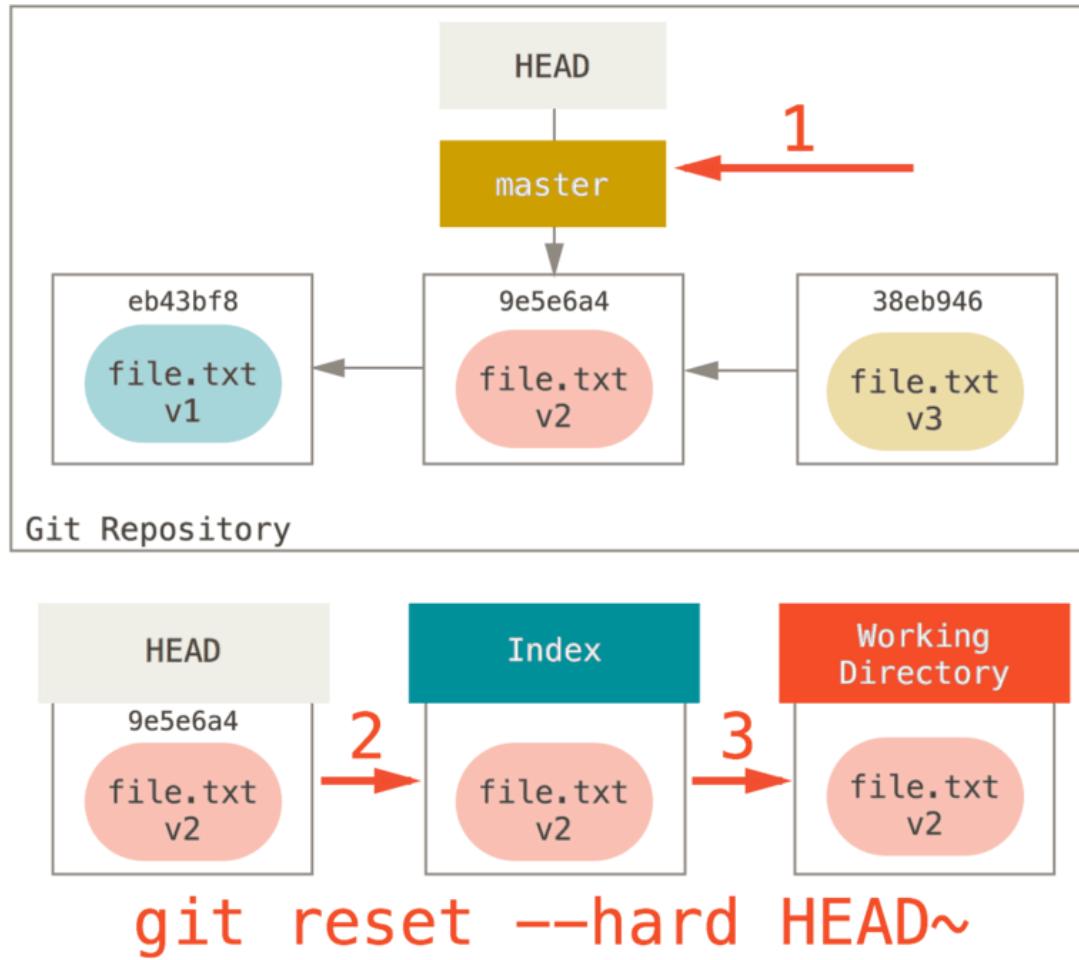
(8) reset --soft



(8) reset [--mixed]



(8) reset --hard



.gitignore

특정 경로의 파일, 폴더 및 하위 내용, 확장자 등을
Index로 올리지 않고 싶을 때 사용

저장소 루트 또는 폴더에 .gitignore라는 파일을 만들어 사용
(윈도 탐색기에서는 이름이 없는 파일을 만들 수 없으니
다른 방법이 필요하다.)

\가 아니라 /를 사용해야 함에 유의하여라.

.gitignore (Cont'd)

```
// Connect to MySQL Database

C:\git\github-hol>git rm example\ajax\db_connection.php
error: the following file has local modifications:
  example/ajax/db_connection.php
(use --cached to keep the file, or -f to force removal)

C:\git\github-hol>git rm -f example\ajax\db_connection.php
rm 'example/ajax/db_connection.php'

C:\git\github-hol>git status -s
D example/ajax/db_connection.php

C:\git\github-hol>git reset --hard HEAD
HEAD is now at 0ca7ec1 Add implementation

C:\git\github-hol>echo example/ajax/db_connection.php > .gitignore

C:\git\github-hol>git add .

C:\git\github-hol>git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .gitignore
    modified:   example/ajax/db_connection.php

C:\git\github-hol>
```

아직은 modified라고 인식된다.

이미 Staged라서 계속 추적되므로

rm --cached -r . 이후 add . 과정이 필요하다.

.gitignore (Cont'd)

```
PS 명령 프롬프트 x + v
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/css/bootstrap-theme.min.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/css/bootstrap.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/css/bootstrap.min.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/fonts/glyphicons-halflings-regular.svg.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/js/bootstrap.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/js/bootstrap.min.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/bootstrap-3.3.5-dist/js/npm.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/index.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/js/jquery-1.11.3.min.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in example/js/script.js.
The file will have its original line endings in your working directory

C:\git\github-hol>git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .gitignore
    deleted:   example/ajax/db_connection.php

C:\git\github-hol>
```

완료 후 파일이 있음에도 deleted라고 인식된다.

[5] 파일 삭제, 복구, .gitignore

- git rm -f example\ajax\db_connection.php
- 파일이 실제로 삭제된 것 확인
- git reset --hard HEAD
- example\ajax\db_connection.php 파일을 열고 6번째 줄의 \$password 값을 "password1234" 등 임의로 변경한다.
- echo example/ajax/db_connection.php > .gitignore
- git add .
- git status
- git rm --cached -r .
 - 실패한다면 git rm /ajax/db_connection.php
- git add .
- git status

(9) stash

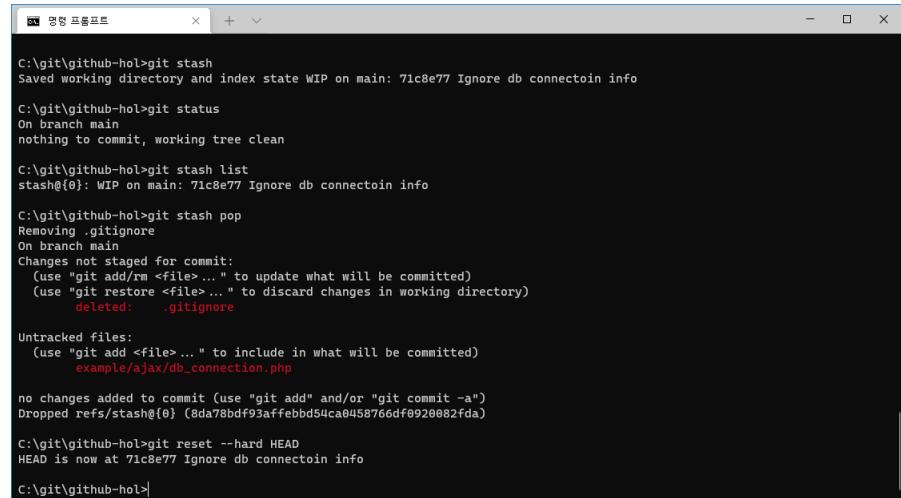
현재 Index를 임시로 저장하고
가장 최근 커밋의 상태로 되돌림 (reset --hard과 같은 결과)

stash list: 저장된 상태들 목록 보기

stash pop: 저장된 상태로 되돌리고 그 상태 삭제
(stash apply를 할 경우 임시 저장된 상태가 계속 남아있어
나중에 꼬일 수 있음)

[6] stash 써 보기

- git commit -m "Ignore db connectoin info"
- git rm .gitignore
- git status
- git stash
- git status
- git stash list
- git stash pop
- git reset --hard HEAD



```
C:\git\github-hol>git stash
Saved working directory and index state WIP on main: 71c8e77 Ignore db connectoin info
C:\git\github-hol>git status
On branch main
nothing to commit, working tree clean
C:\git\github-hol>git stash list
stash@{0}: WIP on main: 71c8e77 Ignore db connectoin info
C:\git\github-hol>git stash pop
Removing .gitignore
On branch main
Changes not staged for commit:
  (use "git add <file> ..." to update what will be committed)
  (use "git restore <file> ..." to discard changes in working directory)
    deleted:   .gitignore

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

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (8da78bdf93affebbd54ca0458766df0920082fda)

C:\git\github-hol>git reset --hard HEAD
HEAD is now at 71c8e77 Ignore db connectoin info
C:\git\github-hol>
```

(4) commit --amend

```
명령 프롬프트 - git log
commit 71c8e77edff990e0c5ef767b5f96cf279b8266ac (HEAD → main)
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 17:19:35 2021 +0900

    Ignore db connectoin info      connectoin? typo!

commit 0ca7ec138486247a8394fbcd041cdada71a14288
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 17:08:46 2021 +0900

    Add implementation

commit cdedbbae753eba859264579f05dc5095d2b6c91
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 17:07:57 2021 +0900

    Add dependency: jquery

commit 299aedc6f4201dale92374c51dec94dfc035f93
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 17:07:28 2021 +0900

    Add dependency: bootstrap

commit c87760acefa039b69fb0ede95f9d3eb26f53f141
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date:   Fri Jun 4 17:01:52 2021 +0900

    README: fix, add credit
:|
```

오타 하나 고치자고 reset, commit을 다시 하기에는 귀찮다

(4) commit --amend (*Cont'd*)



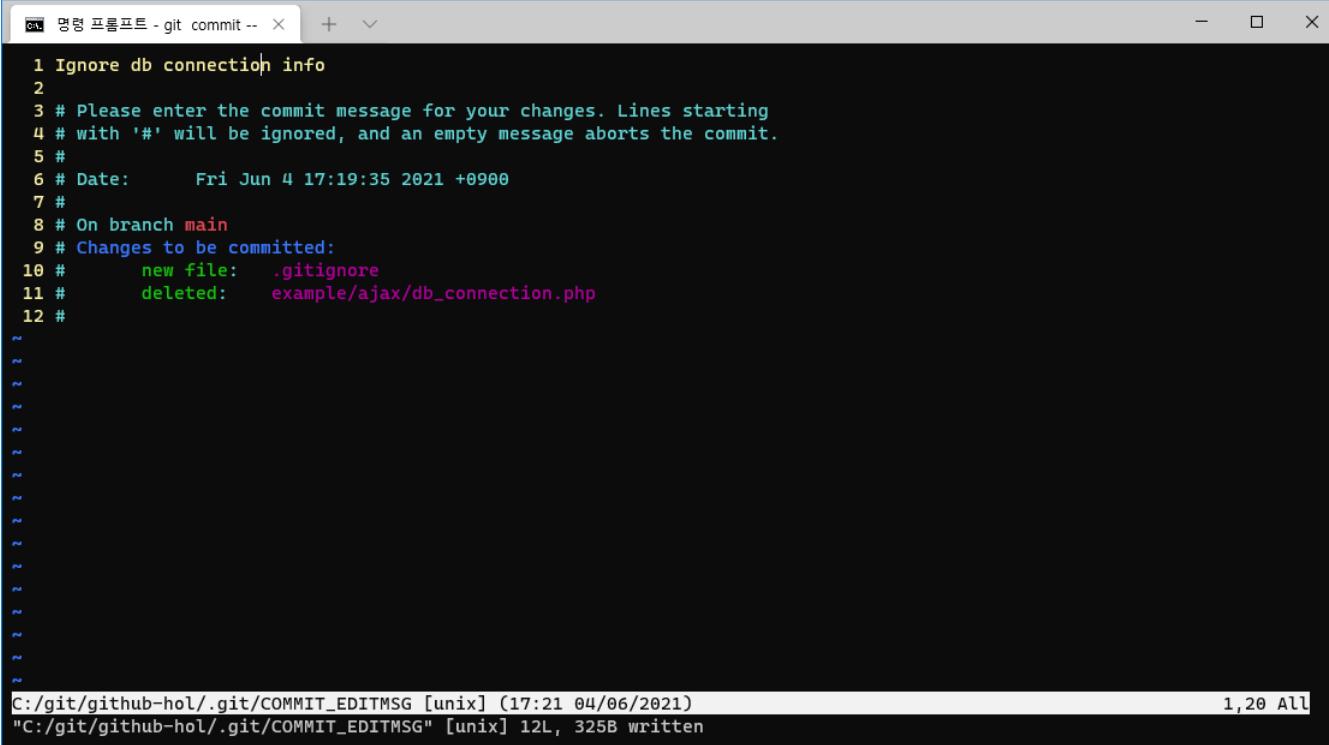
```
ca 명령 프롬프트 - git commit -- + ▾
1 Ignore db connectoin info
2
3 # Please enter the commit message for your changes. Lines starting
4 # with '#' will be ignored, and an empty message aborts the commit.
5 #
6 # Date:      Fri Jun 4 17:19:35 2021 +0900
7 #
8 # On branch main
9 # Changes to be committed:
10 #       new file:   .gitignore
11 #       deleted:    example/ajax/db_connection.php
12 #

C:/git/github-hol/.git/COMMIT_EDITMSG [unix] (17:21 04/06/2021) 1,20 All
-- INSERT --
```

vi로 잘못된 부분 수정 후 저장하고 나가면 완료

수정 모드: i, 삭제: Esc X, 저장하고 나가기: Esc :wq

(4) commit --amend (*Cont'd*)



```
1 Ignore db connection info
2
3 # Please enter the commit message for your changes. Lines starting
4 # with '#' will be ignored, and an empty message aborts the commit.
5 #
6 # Date:      Fri Jun 4 17:19:35 2021 +0900
7 #
8 # On branch main
9 # Changes to be committed:
10 #       new file:   .gitignore
11 #       deleted:    example/ajax/db_connection.php
12 #

C:/git/github-hol/.git/COMMIT_EDITMSG [unix] (17:21 04/06/2021)          1,20 All
"C:/git/github-hol/.git/COMMIT_EDITMSG" [unix] 12L, 325B written
```

vi로 잘못된 부분 수정 후 저장하고 나가면 완료

수정 모드: i, 삭제: Esc X, 저장하고 나가기: Esc :wq

[7] 커밋 메시지 바꾸기

- git commit --amend
- vi로 내용을 수정한다.
- git log --oneline

```
[main 29aeace] Ignore db connection info
Date: Fri Jun 4 17:19:35 2021 +0900
2 files changed, 1 insertion(+), 15 deletions(-)
create mode 100644 .gitignore
delete mode 100644 example/ajax/db_connection.php

C:\git\github-hol>git log --oneline
29aeace (HEAD → main) Ignore db connection info
0ca7ec1 Add implementation
cdedbba Add dependency: jquery
299aecd Add dependency: bootstrap
c87760a README: fix, add credit
daef4c7 Initial commit

C:\git\github-hol>
```

(10) rebase --interactive

현 상황의 base, 즉 이전 커밋들을 조정

본래 rebase는 브랜치 간 작업에서 빛을 발하지만
일단 예전 커밋 메시지를 바꾸기 위해
interactive 모드를 사용해 보자.

(10) rebase --interactive (*Cont'd*)

```
명령 프롬프트 - git rebase --i... + ▾
1 pick 0ca7ec1 Add implementation
2 pick 29aeace Ignore db connection info
3
4 # Rebase cdedbba..29aeace onto cdedbba (2 commands)
5 #
6 # Commands:
7 # p, pick <commit> = use commit
8 # r, reword <commit> = use commit, but edit the commit message
9 # e, edit <commit> = use commit, but stop for amending
10 # s, squash <commit> = use commit, but meld into previous commit
11 # f, fixup <commit> = like "squash", but discard this commit's log message
12 # x, exec <command> = run command (the rest of the line) using shell
13 # b, break = stop here (continue rebase later with 'git rebase --continue')
14 # d, drop <commit> = remove commit
15 # l, label <label> = label current HEAD with a name
16 # t, reset <label> = reset HEAD to a label
17 # m, merge [-C <commit> | -c <commit>] <label> [<oneline>]
18 # .      create a merge commit using the original merge commit's
19 # .      message (or the oneline, if no original merge commit was
20 # .      specified). Use -c <commit> to reword the commit message.
21 #
22 # These lines can be re-ordered; they are executed from top to bottom.
23 #
24 # If you remove a line here THAT COMMIT WILL BE LOST.
25 #
26 # However, if you remove everything, the rebase will be aborted.
27 #
~ C:/git/github-hol/.git/rebase-merge/git-rebase-todo[+][unix] (17:48 04/06/2021) 1,5 All
-- INSERT --
```

(10) rebase --interactive (*Cont'd*)

```
명령 프롬프트 - git rebase --i... + ▾ - □ ×
1 edit 0ca7ec1 Add implementation
2 pick 29aeace Ignore db connection info
3
4 # Rebase cdedbba..29aeace onto cdedbba (2 commands)
5 #
6 # Commands:
7 # p, pick <commit> = use commit
8 # r, reword <commit> = use commit, but edit the commit message
9 # e, edit <commit> = use commit, but stop for amending
10 # s, squash <commit> = use commit, but meld into previous commit
11 # f, fixup <commit> = like "squash", but discard this commit's log message
12 # x, exec <command> = run command (the rest of the line) using shell
13 # b, break = stop here (continue rebase later with 'git rebase --continue')
14 # d, drop <commit> = remove commit
15 # l, label <label> = label current HEAD with a name
16 # t, reset <label> = reset HEAD to a label
17 # m, merge [-C <commit> | -c <commit>] <label> [<oneline>]
18 # .      create a merge commit using the original merge commit's
19 # .      message (or the oneline, if no original merge commit was
20 # .      specified). Use -c <commit> to reword the commit message.
21 #
22 # These lines can be re-ordered; they are executed from top to bottom.
23 #
24 # If you remove a line here THAT COMMIT WILL BE LOST.
25 #
26 # However, if you remove everything, the rebase will be aborted.
27 #
~ C:/git/github-hol/.git/rebase-merge/git-rebase-todo[+] [unix] (17:48 04/06/2021) 1,4 All
:wq
```

(10) rebase --interactive (Cont'd)

```
명령 프롬프트 error: The following untracked working tree files would be overwritten by merge:  
example/ajax/db_connection.php  
Please move or remove them before you merge.  
Aborting  
hint: Could not execute the todo command  
hint:  
hint:     edit 0ca7ec138486247a8394fbed041cdaaa71a14288 Add implementation  
hint:  
hint: It has been rescheduled; To edit the command before continuing, please  
hint: edit the todo list first:  
hint:  
hint:     git rebase --edit-todo  
hint:     git rebase --continue  
Could not apply 0ca7ec1... Add implementation  
  
C:\git\github-hol>git status  
interactive rebase in progress; onto cdedbba  
Last command done (1 command done):  
  edit 0ca7ec1 Add implementation  
Next commands to do (2 remaining commands):  
  edit 0ca7ec1 Add implementation  
  pick 29aeace Ignore db connection info  
(use "git rebase --edit-todo" to view and edit)  
You are currently editing a commit while rebasing branch 'main' on 'cdedbba'.  
(use "git commit --amend" to amend the current commit)  
(use "git rebase --continue" once you are satisfied with your changes)  
  
Untracked files:  
  (use "git add <file> ..." to include in what will be committed)  
    example/ajax/ 오류 발생!  
.gitignore 때문에 그렇다.
```

(10) rebase --interactive (*Cont'd*)

```
C:\git\github-hol>del /q example/ajax
Parameter format not correct - "jax".

C:\git\github-hol>del /q example\ajax

C:\git\github-hol>git rebase --continue
Stopped at 0ca7ec1 ... Add implementation
You can amend the commit now, with

  git commit --amend

Once you are satisfied with your changes, run

  git rebase --continue

C:\git\github-hol>git commit --amend -m "Implement service"
[detached HEAD 3b0933a] Implement service
  Date: Fri Jun 4 17:08:46 2021 +0900
  8 files changed, 381 insertions(+)
   create mode 100644 example/ajax/addRecord.php
   create mode 100644 example/ajax/db_connection.php
   create mode 100644 example/ajax/deleteUser.php
   create mode 100644 example/ajax/readRecords.php
   create mode 100644 example/ajax/readUserDetails.php
   create mode 100644 example/ajax/updateUserDetails.php
   create mode 100644 example/index.php
   create mode 100644 example/js/script.js

C:\git\github-hol>
```

[8] 예전 커밋 메시지 바꾸기

- git rebase --interactive HEAD~2
- 첫째 줄의 pick을 edit으로 바꾸고 저장한다.
- 에러가 발생하는 것 확인 (발생하지 않았으면 다음 del 스킵)
- del /q example\ajax
 - PowerShell, MSYS 등에서는 다를 수 있는데 아무튼 폴더를 지우면 된다.
- git rebase --continue
- "Stopped at *b5b1ee3...* Add implementation" 문구를 확인한다.
- git commit --amend -m "Implement service"
- git rebase --continue
- git log --oneline

브랜치

가지치기

코드를 통째로 복사하고 나서

원래 코드와는 상관없이 독립적으로 개발 진행

(11) branch



```
C:\ 명령 프롬프트
C:\git\github-hol>git status
On branch main
nothing to commit, working tree clean

C:\git\github-hol>git branch
* main

C:\git\github-hol>git branch query-add

C:\git\github-hol>git branch
* main
  query-add

C:\git\github-hol>|
```

현재 커밋을 가리키는 새로운 브랜치 만들기

(11) branch -d

```
C:\ 명령 프롬프트 + ▾
C:\git\github-hol>git status
On branch main
nothing to commit, working tree clean

C:\git\github-hol>git branch
* main

C:\git\github-hol>git branch query-add

C:\git\github-hol>git branch
* main
  query-add

C:\git\github-hol>git branch meaningless

C:\git\github-hol>git branch
* main
  meaningless
  query-add

C:\git\github-hol>git branch -d meaningless
Deleted branch meaningless (was c21a143).

C:\git\github-hol>git branch
* main
  query-add

C:\git\github-hol>|
```

필요 없는 브랜치 삭제

(12) checkout

```
명령 프롬프트
* main

C:\git\github-hol>git branch query-add

C:\git\github-hol>git branch
* main
  query-add

C:\git\github-hol>git branch meaningless

C:\git\github-hol>git branch
* main
  meaningless
  query-add

C:\git\github-hol>git branch -d meaningless
Deleted branch meaningless (was c21a143).

C:\git\github-hol>git branch
* main
  query-add

C:\git\github-hol>git checkout query-add
Switched to branch 'query-add'

C:\git\github-hol>git status
On branch query-add
nothing to commit, working tree clean

C:\git\github-hol>|
```

해당 브랜치로 이동

(12) checkout -b

```
명령 프롬프트
meaningless
query-add

C:\git\github-hol>git branch -d meaningless
Deleted branch meaningless (was c21a143).

C:\git\github-hol>git branch
* main
  query-add

C:\git\github-hol>git checkout query-add
Switched to branch 'query-add'

C:\git\github-hol>git status
On branch query-add
nothing to commit, working tree clean

C:\git\github-hol>git checkout -b query-delete
Switched to a new branch 'query-delete'

C:\git\github-hol>git status
On branch query-delete
nothing to commit, working tree clean

C:\git\github-hol>git branch
  main
  query-add
* query-delete

C:\git\github-hol>|
```

브랜치를 만들면서 이동

[9] 브랜치 옮겨다니며 작업하기

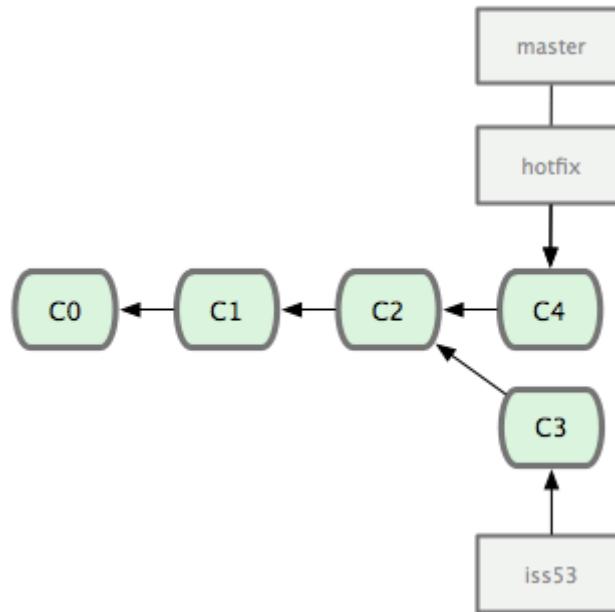
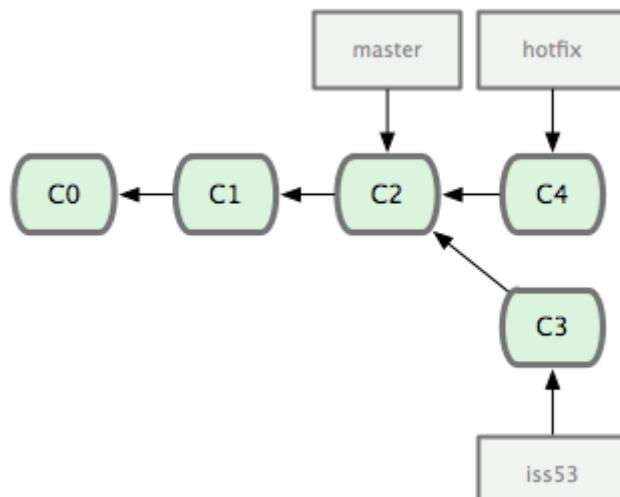
- git branch query-add
- git checkout query-add
- example\ajax\addRecord.php:L12 \$query 값을 "INSERT VALUES INTO DB" 등 임의로 변경한다.
- git commit -a -m "query: implement add"
- git checkout main
- git checkout -b query-delete
- example\ajax\deleteUser.php:L12 \$query 값을 "DELETE FROM DB" 등 임의로 변경 한다.
- git commit -a -m "query: implement delete"
- git checkout main
- git checkout -b query-delete-2
- example\ajax\deleteUser.php:L12 \$query 값을 "DROP DB"
- git commit -a -m "query: implement delete with drop"
- git checkout main
- git branch

2021-06-04

```
C:\git\github-hol>git log --oneline --graph --all
* 1368d1a (query-delete-2) query: implement delete with drop
| * 347bd44 (query-delete) query: implement delete
|/
| * 36764be (query-add) query:implement add
|/
* c21a143 (HEAD -> main) Ignore db connection info
* 3b0933a Implement service
* cdedbba Add dependency: jquery
* 299aecd Add dependency: bootstrap
* c87760a README: fix, add credit
* daef4c7 Initial commit
```

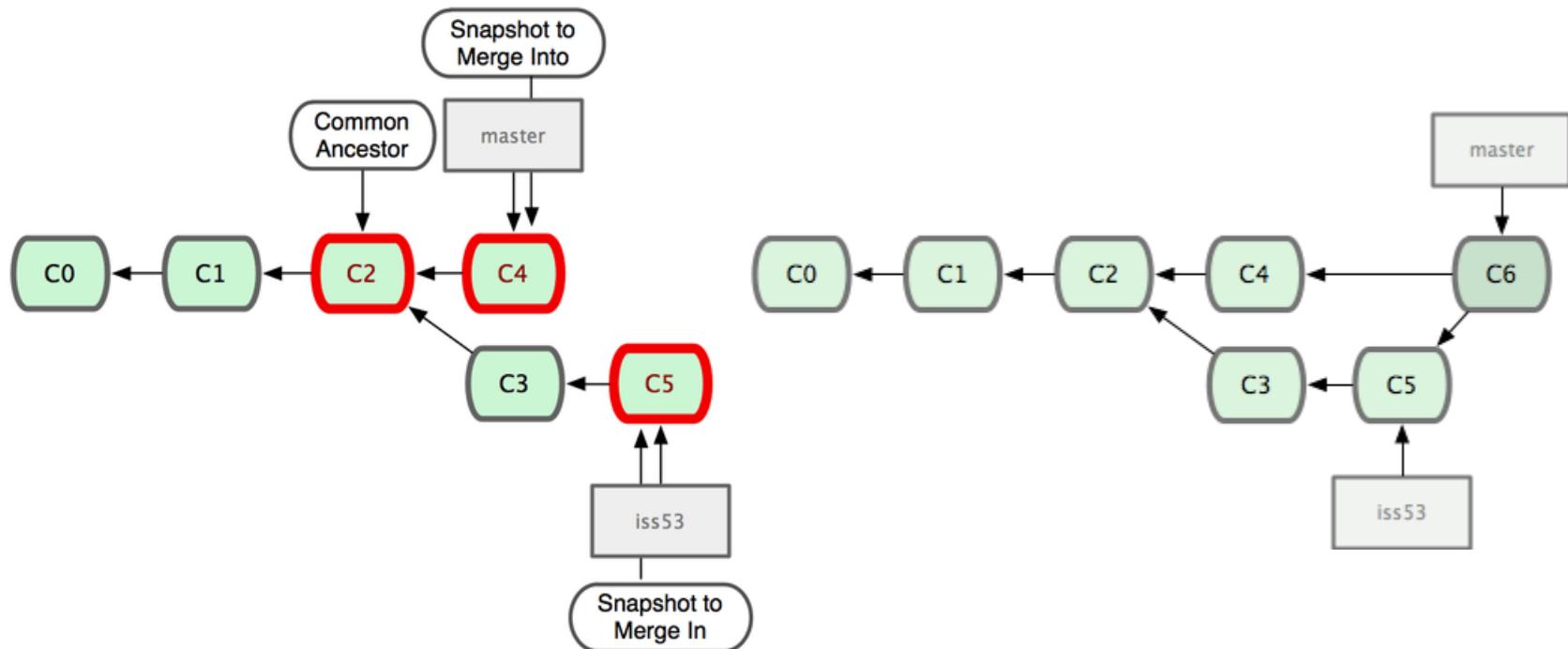
(13) merge

- Fast-forward



(13) merge

- Non-fast-forward: 3-way merge



Conflict

병합을없애시다(병합은나의원수)
우리가 2년 동안코드를 작성했는데
없深交 병합 작업이 충돌을발
생시켰습니다 병합을죽입니다코
드는 아주좋았습니다 코드
는아직않뻗었는데싸가지엄는
병합이 터뜨렸습니다 조심하십시오

개미를죽입니다(개미는나의원수)
우리가 2년 동안달팽이를키웠는데
없深交 봉개 미새끼가 달팽이를죽
였습니다 개미를죽이시다 달
팽이는 아주착했습니다 달팽
이는 아직안죽었는데씨가지었습니다
불개미가죽었습니다 조심하십시오

Conflict (*Cont'd*)

<<<<<< HEAD

(현재 HEAD에 있는 내용)

=====

(병합하고자 하는 브랜치에서 위와 충돌하는 내용)

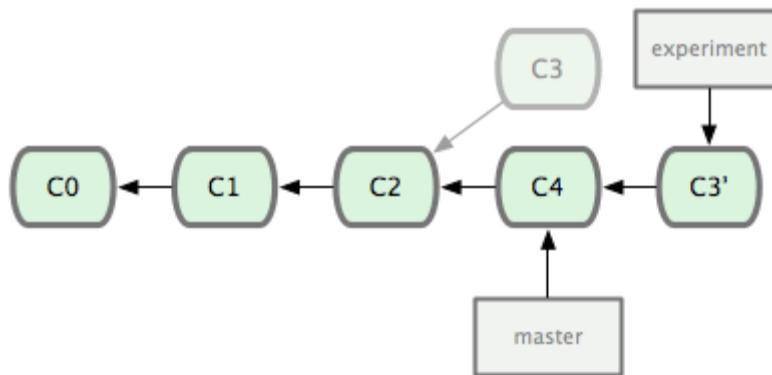
>>>>> query-delete-2

읽어보고 알아서 잘 해결 후 git add

변경하지 않으면 저 내용이 그대로 들어간다.

그 예시는 결과 저장소에서 볼 수 있다.

(10) rebase



C3의 base를 C2에서 C4로 변경
로그가 깔끔해 지는 장점이 있다.

[10] 병합, 충돌 해결하기

- git merge query-add
- Fast-forward로 병합됨을 확인한다.
- git checkout query-delete
- git rebase main
- git checkout main
- git merge query-delete
- Fast-forward로 병합됨을 확인한다.
- git merge query-delete-2
- 충돌이 발생함을 확인하고, example\ajax\deleteUser.php:L12-16을 1줄로 만들며 \$query의 값을 "DELETE FROM DB" or "DROP DB" 등 임의로 변경한다.
- git add .
- git commit
- 커밋 메시지를 확인하고 vi에서 나온다.
- git log --oneline --graph

```
C:\git\github-hol>git log --oneline --graph
* 5c48a4a (HEAD → main) Merge branch 'query-delete-2'
|\ 
| * 1368d1a (query-delete-2) query: implement delete with drop
| * | 1a99a2f (query-delete) query: implement delete
| * | 36764be (query-add) query:implement add
|/
* c21a143 Ignore db connection info
* 3b0933a Implement service
* cdedbba Add dependency: jquery
* 299aedc Add dependency: bootstrap
* c87760a README: fix, add credit
* daef4c7 Initial commit
```

(14) tag

```
명령 프롬프트

C:\git\github-hol>git log --oneline --graph
* 5c48a4a (HEAD → main) Merge branch 'query-delete-2'
|\ \
| * 1368dia (query-delete-2) query: implement delete with drop
| * 1a99a2f (query-delete) query: implement delete
| * 36764be (query-add) query:implement add
|/
* c21a143 Ignore db connection info
* 3b0933a Implement service
* cdedbba Add dependency: jquery
* 299aedc Add dependency: bootstrap
* c87760a README: fix, add credit
* daef4c7 Initial commit

C:\git\github-hol>git tag

C:\git\github-hol>git tag v0.11 c21a143

C:\git\github-hol>git tag
v0.11

C:\git\github-hol>git tag v0.12

C:\git\github-hol>git tag
v0.11
v0.12

C:\git\github-hol>|
```

커밋에 태그 붙이기

(15) reflog

```
명령 프롬프트 - git reflog
c21a143 (HEAD → main) HEAD@{0}: checkout: moving from query-add to main
36764be (query-add) HEAD@{1}: checkout: moving from main to query-add
c21a143 (HEAD → main) HEAD@{2}: checkout: moving from query-delete-2 to main
1368d1a (query-delete-2) HEAD@{3}: checkout: moving from main to query-delete-2
c21a143 (HEAD → main) HEAD@{4}: reset: moving to HEAD~1
1368d1a (query-delete-2) HEAD@{5}: checkout: moving from query-delete-2 to main
1368d1a (query-delete-2) HEAD@{6}: checkout: moving from main to query-delete-2
1368d1a (query-delete-2) HEAD@{7}: commit: query: implement delete with drop
c21a143 (HEAD → main) HEAD@{8}: checkout: moving from query-delete to main
347bd44 (query-delete) HEAD@{9}: commit: query: implement delete
c21a143 (HEAD → main) HEAD@{10}: checkout: moving from main to query-delete
c21a143 (HEAD → main) HEAD@{11}: checkout: moving from query-add to main
36764be (query-add) HEAD@{12}: commit: query:implement add
c21a143 (HEAD → main) HEAD@{13}: checkout: moving from main to query-add
c21a143 (HEAD → main) HEAD@{14}: checkout: moving from query-delete to main
c21a143 (HEAD → main) HEAD@{15}: checkout: moving from query-add to query-delete
c21a143 (HEAD → main) HEAD@{16}: checkout: moving from main to query-add
c21a143 (HEAD → main) HEAD@{17}: rebase (continue) (finish): returning to refs/heads/main
c21a143 (HEAD → main) HEAD@{18}: rebase (continue) (pick): Ignore db connection info
3b0933a HEAD@{19}: commit (amend): Implement service
0ca7ec1 HEAD@{20}: rebase: fast-forward
cdedbba HEAD@{21}: rebase (start): checkout HEAD~2
29aeace HEAD@{22}: commit (amend): Ignore db connection info
71c8e77 HEAD@{23}: reset: moving to HEAD
71c8e77 HEAD@{24}: reset: moving to HEAD
71c8e77 HEAD@{25}: commit: Ignore db connectoin info
0ca7ec1 HEAD@{26}: reset: moving to HEAD
0ca7ec1 HEAD@{27}: reset: moving to HEAD
0ca7ec1 HEAD@{28}: commit: Add implementation
:|
```

지금까지 거쳐온 모든 커밋들 보기



Chapter 2

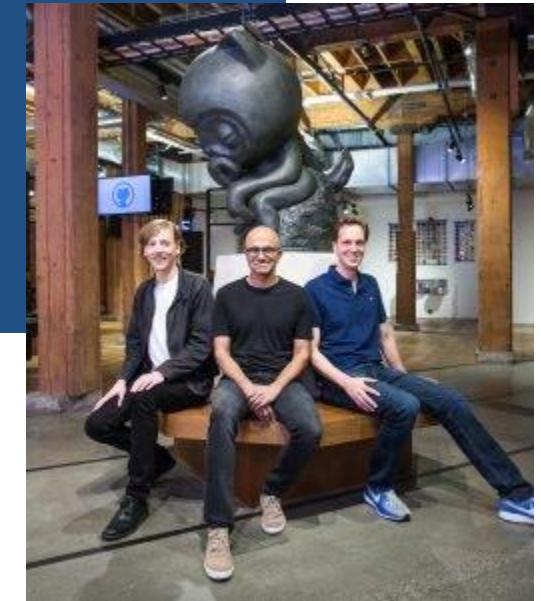
GITHUB

GitHub란?

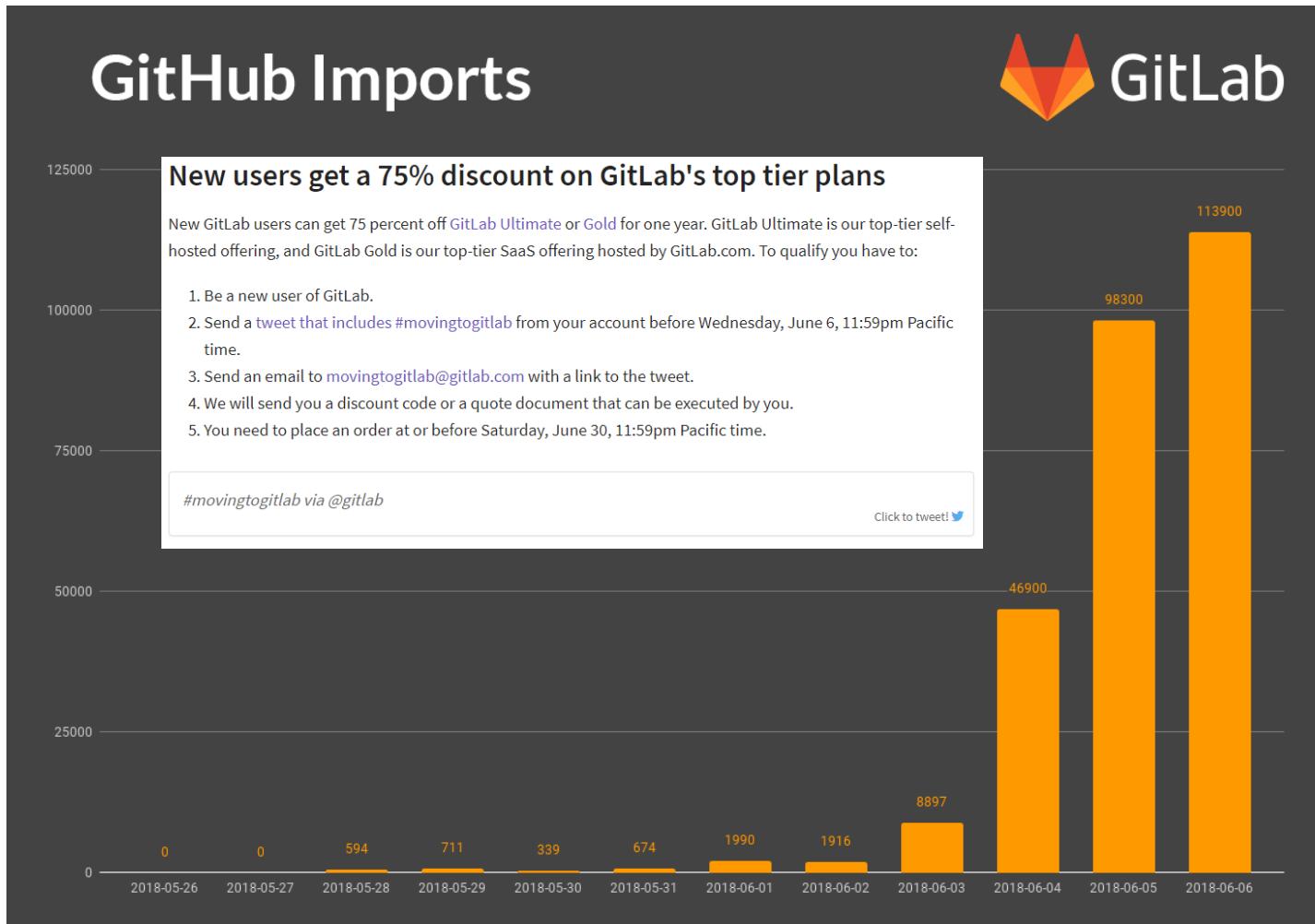
Remote Git 저장소를 제공해 주는 사이트

Fork, Pull Request 등의 자체 기능 제공

Microsoft + GitHub



KEEP CALM AND USE GITHUB



Why GitHub?

많이 쓰니까

포트폴리오 (회사에서 보니까)

Fork/PR, 이슈 트래커, Wiki, Pages 등

Chapter 2-1

GITHUB: 가입

Git 실습 Github 회원가입

The screenshot shows the GitHub sign-up process. At the top, there's a navigation bar with links for Personal, Open source, Business, Explore, Pricing, Blog, Support, a search bar, and Sign In/Sign up buttons. The main heading is "Join GitHub" with the subtext "The best way to design, build, and ship software." Below this, three steps are outlined: Step 1: Set up a personal account, Step 2: Choose your plan, and Step 3: Go to your dashboard. The "Create your personal account" section contains fields for Username, Email Address, and Password, each with descriptive placeholder text. To the right, a box highlights "You'll love GitHub" with features like Unlimited collaborators, Unlimited public repositories, Great communication, Friction-less development, and Open source community. At the bottom, terms of service and privacy policy links are mentioned, along with a prominent green "Create an account" button.

가입정보 기입

Step 1:
Set up a personal account

Step 2:
Choose your plan

Step 3:
Go to your dashboard

Create your personal account

Username

This will be your username — you can enter your organization's username next.

Email Address

You will occasionally receive account related emails. We promise not to share your email with anyone.

Password

Use at least one lowercase letter, one numeral, and seven characters.

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

Create an account

<https://github.com/join>

Git 실습 Github 메일인증하기

The screenshot shows an email from GitHub. At the top right is the GitHub logo. To its left, a gray box contains the Korean text "본인 메일확인해서 버튼 누르기". The main content area starts with "Hi @yukult400!" followed by a message: "Help us secure your GitHub account by verifying your email address (yukult400@naver.com). This lets you access all of GitHub's features." Below this is a blue button with white text that reads "Verify email address". A red arrow points from the text "본인 메일확인해서 버튼 누르기" to this button. At the bottom of the email, there is a link for those who cannot click the button: "https://github.com/users/yukult400/emails/21042026/confirm_verification/388b1d3024fc012d64848df72d2d8fc4d7dc3b5a". A final note at the bottom states: "You're receiving this email because you recently created a new GitHub account or added a new email address. If this wasn't you, please ignore this email." At the very bottom center is the GitHub logo.

본인 메일확인해서 버튼 누르기

Hi @yukult400!

Help us secure your GitHub account by verifying your email address (yukult400@naver.com). This lets you access all of GitHub's features.

Verify email address

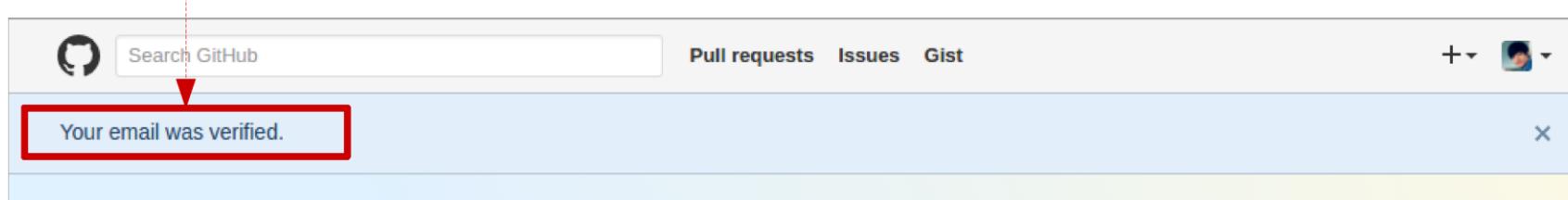
Button not working? Paste the following link into your browser:
https://github.com/users/yukult400/emails/21042026/confirm_verification/388b1d3024fc012d64848df72d2d8fc4d7dc3b5a

You're receiving this email because you recently created a new GitHub account or added a new email address. If this wasn't you, please ignore this email.

GitHub

Git 실습 Github 메일인증하기

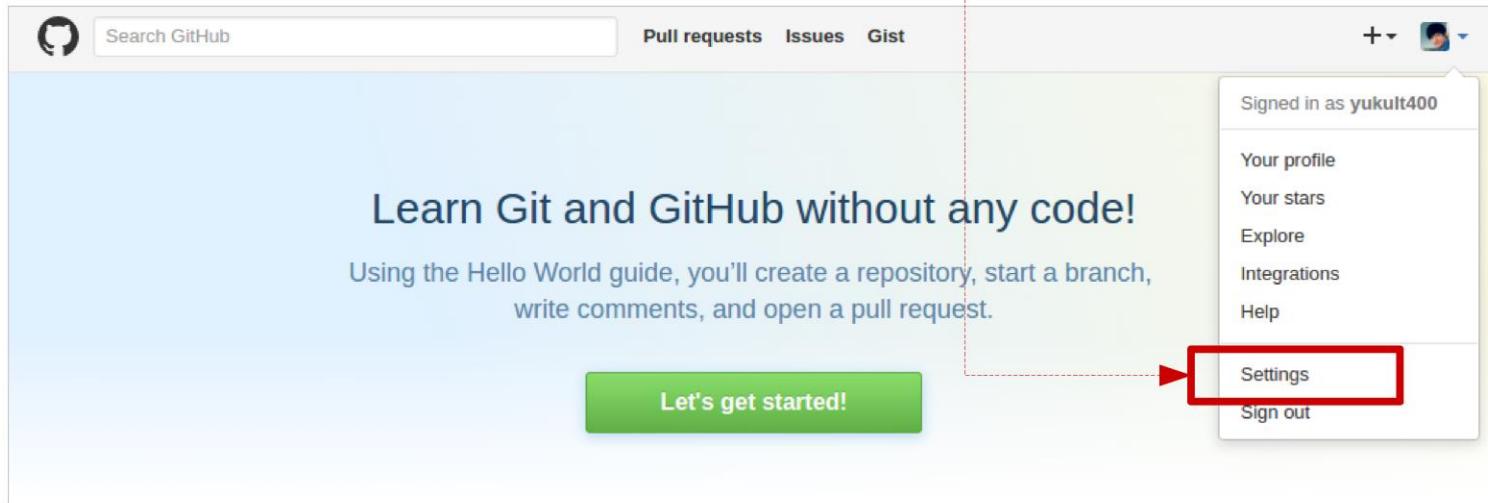
다시 Github 접속해서 메일인증 확인하기



<http://github.com>

Git 실습 Github 메일인증하기

기다려도 메일이 혹시 안왔으면설정버튼



<http://github.com>

Git 실습 Github 메일인증하기

메일이 혹시 안왔으면 인증메일 다시 보내기

The screenshot shows the GitHub 'Personal settings' page with the 'Emails' section highlighted by a red box. The 'yukult400@naver.com' email is listed as Primary, Public, and Unverified. A red box highlights the 'Verification email sent. Resend' link next to the email address. A red arrow points from the Korean text '메일이 혹시 안왔으면 인증메일 다시 보내기' to this link.

Search GitHub

Pull requests Issues Gist

+

Personal settings

Profile

Account

Emails ⚠

Notifications

Billing

SSH keys

Email

Your primary GitHub email address will be used for account-related notifications (e.g. account changes and billing receipts) as well as any web-based GitHub operations (e.g. edits and merges).

Please verify an email address to collaborate on GitHub, receive notifications, and help us protect your account.

yukult400@naver.com Primary Public Unverified

Verification email sent. Resend ⚠

Add email address Add

<https://github.com/settings/emails>



Chapter 2-2

GITHUB: 기능 및 실습

[0] 저장소 만들기

The screenshot shows the GitHub homepage. A red circle highlights the green 'New repository' button located in the 'Repositories' section of the sidebar. The sidebar also lists several repositories under 'KENNYSOFT'. The main area displays recent activity, including comments on pull requests and pushes to repositories like 'jin5866/KuiconBack'.

Apple Inc. is hiring: Siri - HomeKit QA Engineer in Cupertino, CA

Brought to you by GitHub Jobs

KENNYSOFT

New repository

Repositories

Find a repository...

- Kagamia/WzComparerR2
- KENNYSOFT/SerendipityEval...
- KENNYSOFT/MnetDownloader
- KENNYSOFT/symapse
- kajebiii/ps.kajebiii.ga
- KENNYSOFT/InvenAdClick
- KENNYSOFT/WzComparerR2

Show more

Your teams

Find a team...

- gshslatexitro/graduates-31st
- gshslatexitro/informatics
- koistudy2/owner

Pull requests Issues Marketplace Explore

Browse activity Discover repositories

Recent activity

- About face accessory simulation. (2 comments)
- please add export xml function (6 comments)
- Error when rendering Skeleton Image (8 comments)

All activity

- jin5866 pushed to jin5866/KuiconBack 7 hours ago
 - 2 commits to master
 - bf8bf61 스레드 만들어질때 멀티코어로 설정해줌
 - c3b2552 PcapParser.cs 코드 정리
- jin5866 pushed to jin5866/KuiconBack 10 hours ago
 - 1 commit to master
 - 96c9215 1. 멀티스레드를 위해 static 클래스를 dynamic 클래스로 변경
- wchoiac starred KENNYSOFT/AnalyzeKakaoTalk a day ago

KENNYSOFT/AnalyzeKakaoTalk ★ Star

<https://github.com/new>

[0] 저장소 만들기 (*Cont'd*)

The screenshot shows the GitHub interface for creating a new repository. At the top, there's a navigation bar with links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the navigation bar, the main title is 'Create a new repository'. A sub-instruction below it says, 'A repository contains all the files for your project, including the revision history.' The form fields include 'Owner' (set to 'KENNYSOFT'), 'Repository name' ('github-hol'), and a 'Description (optional)' field containing 'GitHub Hands-On Lab (7/12)'. There are two radio button options for visibility: 'Public' (selected) and 'Private'. Underneath, there's a checkbox for 'Initialize this repository with a README', which is checked. Below the checkbox are dropdown menus for '.gitignore' (set to 'None') and 'Add a license' (also set to 'None'). At the bottom of the form is a large green 'Create repository' button.

Create a New Repository X

GitHub, Inc. [US] | https://github.com/new

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

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: KENNYSOFT / Repository name: github-hol ✓

Great repository names are short and memorable. Need inspiration? How about friendly-barnacle.

Description (optional): GitHub Hands-On Lab (7/12)

Public Anyone can see this repository. You choose who can commit.

Private You choose who can see and commit to this repository.

Initialize this repository with a README This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None Add a license: None ⓘ

Create repository

© 2018 GitHub, Inc. Terms Privacy Security Status Help

Contact GitHub API Training Shop Blog About

(16) push

원격 저장소에 브랜치 올리기

-f: Fast-forward가 아니어도 강제로 올림

-u / --set-upstream:

이번에 지정한 remote와 브랜치를 기본 값으로 사용
(Tracking reference 생성)

인증은? → Personal Access Token 또는 ssh key

[1] 원격 저장소 주소 추가, push

- git remote add origin

<https://github.com/KENNYSOFT/github-hol.git>

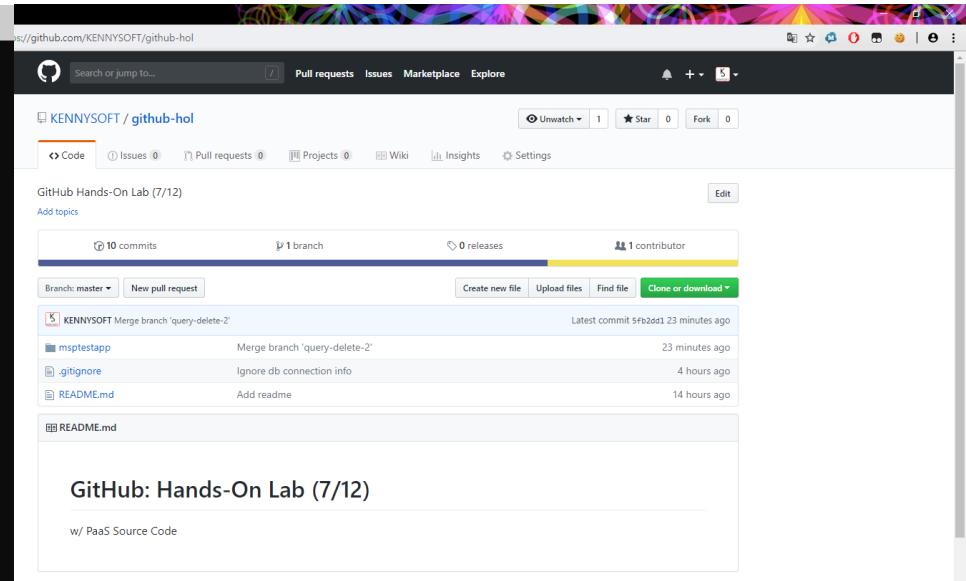
- 또는 ssh를 사용한다면 <git@github.com:KENNYSOFT/github-hol.git>

- git remote -v

- git push -u origin main

```
C:\git\github-hol>git remote -v
C:\git\github-hol>git remote add origin https://github.com/KENNYSOFT/github-hol.git
C:\git\github-hol>git remote -v
origin https://github.com/KENNYSOFT/github-hol.git (fetch)
origin https://github.com/KENNYSOFT/github-hol.git (push)

C:\git\github-hol>git push -u origin main
Enumerating objects: 69, done.
Counting objects: 100% (69/69), done.
Delta compression using up to 12 threads
Compressing objects: 100% (60/60), done.
Writing objects: 100% (69/69), 298.43 KiB | 1.52 MiB/s, done.
Total 69 (delta 16), reused 29 (delta 3), pack-reused 0
remote: Resolving deltas: 100% (16/16), done.
remote:
remote: Create a pull request for 'main' on GitHub by visiting:
remote:     https://github.com/KENNYSOFT/github-hol/pull/new/main
remote:
To github.com:KENNYSOFT/github-hol.git
 * [new branch]      main    -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
C:\git\github-hol>
```



[2] tag

- git tag v0.2
- git tag v0.1 *c21a143*
- git push --tags



```
Date: Fri Jun 4 20:33:07 2021 +0900
query: element delete
commit 130...996b58c6b243d78d628f2e746e899e8ff (query-delete-2)
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date: Fri Jun 4 20:33:45 2021 +0900
query: element delete with drop
commit 367...bd2db229a18fcab704376d263ee6304dd (query-add)
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date: Fri Jun 4 20:32:19 2021 +0900
query: element add
commit c21a143388cc14a1ca7d07e120ad6ee509cd56fb (tag: v0.11)
Author: Hyeonmin Park <hyeonmin.park@kennysoft.kr>
Date: Fri Jun 4 17:19:35 2021 +0900
Ignore db connection info

C:\git\github-hol>git push --tags
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:KENNYSOFT/github-hol.git
 * [new tag]      v0.11 → v0.11
 * [new tag]      v0.12 → v0.12

C:\git\github-hol>
```

[3] Release

The screenshot shows a GitHub repository page for 'KENNYSOFT/github-hol'. The top navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. Below the header, the repository name 'KENNYSOFT / github-hol' is displayed, along with statistics: 10 commits, 1 branch, 2 releases (circled in red), and 1 contributor. A red circle highlights the '2 releases' link. The main content area shows a list of recent commits from 'msptestapp' and '.gitignore'. At the bottom, there is a summary box with the text 'GitHub: Hands-On Lab (7/12)' and 'w/ PaaS Source Code'.

KENNYSOFT/github-hol

GitHub, Inc. [US] | https://github.com/KENNYSOFT/github-hol

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

Unwatch 1 Star 0 Fork 0

KENNYSOFT / github-hol

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

GitHub Hands-On Lab (7/12) Edit

Add topics

10 commits 1 branch 2 releases 1 contributor

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

KENNYSOFT Merge branch 'query-delete-2' Latest commit 5fb2dd1 44 minutes ago

msptestapp Merge branch 'query-delete-2' 44 minutes ago

.gitignore Ignore db connection info 4 hours ago

README.md Add readme 14 hours ago

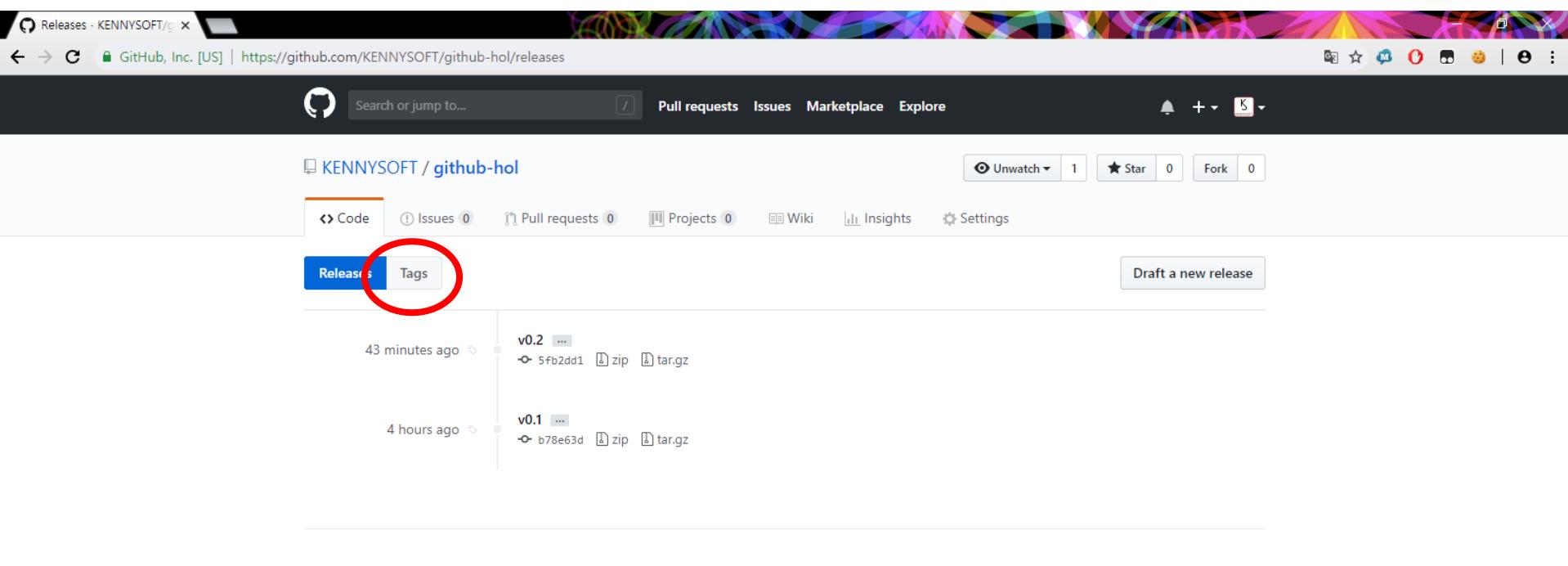
README.md

GitHub: Hands-On Lab (7/12)

w/ PaaS Source Code

<https://github.com/KENNYSOFT/github-hol/releases>

[3] Release (Cont'd)



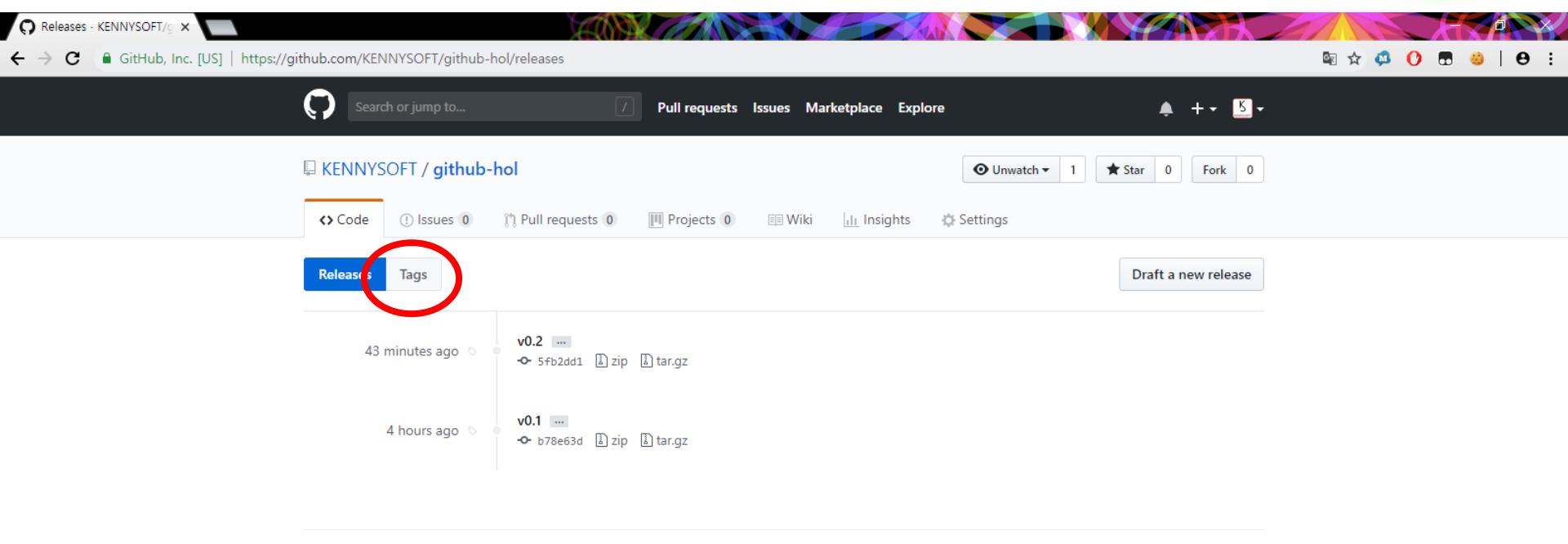
The screenshot shows a GitHub releases page for the repository KENNYSOFT/github-hol. The 'Releases' tab is active and highlighted with a red circle. Two releases are listed:

- v0.2 (43 minutes ago)
 - Commit: 5fb2dd1
 - Zip file
 - Tar.gz file
- v0.1 (4 hours ago)
 - Commit: b78e63d
 - Zip file
 - Tar.gz file

At the bottom of the page, there is a footer with links to GitHub's Terms, Privacy, Security, Status, Help, Contact GitHub, API, Training, Shop, Blog, and About pages.

https://github.com/KENNYSOFT/github-hol/tags

[3] Release (Cont'd)



The screenshot shows a GitHub releases page for the repository KENNYSOFT/github-hol. The 'Releases' tab is active and highlighted with a red circle. Two releases are listed:

- v0.2 (43 minutes ago)
 - Commit: 5fb2dd1
 - Zip file
 - Tar.gz file
- v0.1 (4 hours ago)
 - Commit: b78e63d
 - Zip file
 - Tar.gz file

At the bottom of the page, there is a footer with links to GitHub's Terms, Privacy, Security, Status, Help, Contact GitHub, API, Training, Shop, Blog, and About pages.

<https://github.com/KENNYSOFT/github-hol/releases>

[3] Release (*Cont'd*)

The screenshot shows the GitHub releases page for the repository `KENNYSOFT/github-hol`. The page displays two releases:

- v0.2** (44 minutes ago): Includes a commit `5fb2dd1` and download links for `zip` and `tar.gz`. To the right, there is a link to [Add release notes](#) with the note "(No release notes)".
- v0.1** (4 hours ago): Includes a commit `b78e63d` and download links for `zip` and `tar.gz`. To the right, there is a link to [Add release notes](#) with the note "(No release notes)".

A red circle highlights the [Add release notes](#) link for the v0.1 release.

At the bottom of the page, the footer includes links to GitHub's Terms, Privacy, Security, Status, Help, Contact GitHub, API, Training, Shop, Blog, and About.

The URL in the browser address bar is <https://github.com/KENNYSOFT/github-hol/releases/new?tag=v0.1>.

[3] Release (*Cont'd*)

The screenshot shows a GitHub release page for the repository KENNYSOFT/github-hol. The release is titled "v0.1 (7/12)". It includes two asset links: "Source code (zip)" and "Source code (tar.gz)". Below the assets, there is a link to "MWE: Minimum Working Example". The GitHub interface includes standard navigation bars like "Code", "Issues", "Pull requests", "Projects", "Wiki", "Insights", and "Settings". The release page also displays metrics such as 1 unwatched user, 0 stars, and 0 forks.

Release v0.1 (7/12) - KENNYSOFT / github-hol

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

KENNYSOFT / github-hol

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Releases Tags

Latest release v0.1 b78e63d

v0.1 (7/12)

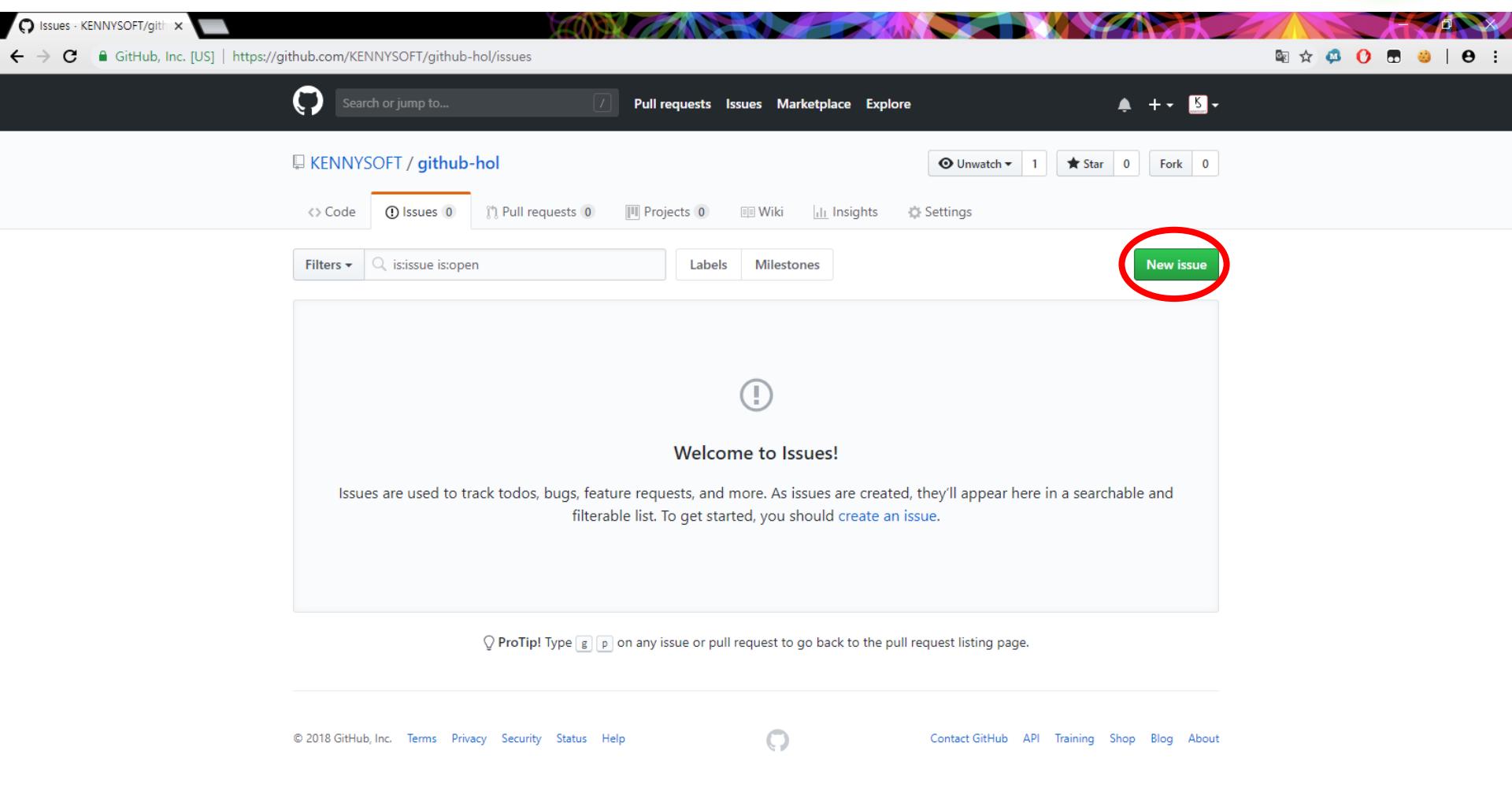
KENNYSOFT released this 38 seconds ago · 4 commits to master since this release

Assets

Source code (zip) Source code (tar.gz)

MWE: Minimum Working Example

[4] Issues



A screenshot of a GitHub Issues page. The URL in the address bar is <https://github.com/KENNYSOFT/github-hol/issues>. The page title is "KENNYSOFT / github-hol". The navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. On the right side of the header, there are buttons for Unwatch (1), Star (0), Fork (0), and a dropdown menu. Below the header, there are tabs for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. A search bar shows the query "is:issue is:open". To the right of the search bar are buttons for Labels and Milestones. A prominent green button labeled "New issue" is circled in red. The main content area features a large exclamation mark icon and the text "Welcome to Issues!". It explains that issues are used to track todos, bugs, feature requests, and more. A "ProTip!" message at the bottom suggests using "g p" to go back to the pull request listing page. At the bottom of the page, there are links for Contact GitHub, API, Training, Shop, Blog, and About. The footer also includes links for Terms, Privacy, Security, Status, and Help.

<https://github.com/KENNYSOFT/github-hol/issues/new>

[4] Issues (*Cont'd*)

The screenshot shows a GitHub issue page for a repository named 'KENNYSOFT / github-hol'. The issue is titled 'Need guest book #1' and is marked as 'Open'. It has 1 comment from the author, KENNYSOFT, who stated: 'Need guest book folder for collect feedback with pull requests.' The issue page includes a rich text editor for writing comments, a preview section, and various configuration options like assignees, labels, projects, milestones, and notifications.

GitHub, Inc. [US] | https://github.com/KENNYSOFT/github-hol/issues/1

KENNYSOFT / github-hol

Issues 1 | Pull requests 0 | Projects 0 | Wiki | Insights | Settings

Unwatch 1 | Star 0 | Fork 0

Need guest book #1

Open KENNYSOFT opened this issue just now · 0 comments

KENNYSOFT commented just now

Owner + 🧑

Need guest book folder for collect feedback with pull requests.

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Notifications

Unsubscribe

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

Comment

Close issue

1 participant

[4] Issues (*Cont'd*)

- Level4 폴더 내의 guestbook 폴더를 저장소로 복사한다.
- git add .
- git commit -m "Add guestbook; resolve #1"
- git push

The screenshot shows two browser tabs open on GitHub. The left tab displays the commit history for the 'master' branch, showing several commits related to implementing a guestbook feature. The right tab shows issue #1, titled 'Need guest book #1', which is now closed. The issue details page includes a comment from the user KENNYSOFT and a note indicating it was closed by a merge commit. The GitHub interface includes standard navigation bars like 'Pull requests', 'Issues', and 'Explore'.

(17) clone

원격 저장소를 통째로 복사하기

자동으로 그 주소가 origin이라는 remote로 추가된다.

[5] Fork, clone

The screenshot shows a GitHub repository page for 'KENNYSOFT/github-hol'. The top navigation bar includes tabs for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. On the right side of the header, there is a message in Korean: '부계정이다.' (It's a forked account). Below the header, the repository name 'KENNYSOFT / github-hol' is displayed, along with a 'Watch' button, a 'Star' button, and a 'Fork' button, which is circled in red. A tooltip for the 'Fork' button reads: 'Fork your own copy of KENNYSOFT/github-hol to your account'. The main content area shows 'GitHub Hands-On Lab (7/12)' with 11 commits, 1 branch, 2 releases, and 1 contributor. The commit list includes:

Author	Commit Message	Time Ago
KENNYSOFT	Add guestbook; resolve #1	Latest commit 1e7338e 3 minutes ago
guestbook	Add guestbook; resolve #1	3 minutes ago
msptestapp	Merge branch 'query-delete-2'	an hour ago
.gitignore	Ignore db connection info	5 hours ago
README.md	Add readme	14 hours ago

Below the commit list, there is a section titled 'GitHub: Hands-On Lab (7/12)' with the subtext 'w/ PaaS Source Code'.

[5] Fork, clone (*Cont'd*)

KENNYSOFT2/github-ho x Commits - KENNYSOFT/... x GitHub, Inc. [US] | https://github.com/KENNYSOFT2/github-ho

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

KENNYSOFT2 / github-ho forked from KENNYSOFT/github-ho

Unwatch 1 Star 0 Fork 1

Code Pull requests 0 Projects 0 Wiki Insights Settings

Forking KENNYSOFT/github-ho

It should only take a few seconds.

Refresh

© 2018 GitHub, Inc. Terms Privacy Security Status Help

Contact GitHub API Training Shop Blog About

github.com의 응답을 기다리는 중...

[5] Fork, clone (*Cont'd*)

The screenshot shows a GitHub repository page for 'KENNYSOFT2/github-hol'. The page includes a header with navigation links like 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the header, the repository name 'KENNYSOFT / github-hol' is displayed, along with a 'Fork' button. The main content area shows 'GitHub Hands-On Lab (7/12)' and a list of commits. On the right side, there's a 'Clone with HTTPS' section with a URL field containing 'https://github.com/KENNYSOFT2/github-hol'. This URL field is circled in red.

KENNYSOFT / **github-hol**
forked from KENNYSOFT/github-hol

Code Pull requests 0 Projects 0 Wiki Insights Settings

Unwatch 1 Star 0 Fork 1

GitHub Hands-On Lab (7/12) Edit

Add topics

11 commits 1 branch 2 releases 1 contributor

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

This branch is even with KENNYSOFT:master.

KENNYSOFT Add guestbook; resolve #1
guestbook Add guestbook; resolve #1
msptestapp Merge branch 'query-delete-2'
.gitignore Ignore db connection info
README.md Add readme

Clone with HTTPS Use SSH
Use Git -> Checkout with SVN using the Web URL
<https://github.com/KENNYSOFT2/github-hol>

Open in Desktop Download ZIP

README.md

GitHub: Hands-On Lab (7/12)

w/ PaaS Source Code

[5] Fork, clone (*Cont'd*)

- 작업하던 내용과 혼란이 생기지 않도록 다른 폴더로 이동한다.
- git clone <https://github.com/snlt7d/github-hol.git>
- cd github-hol
- git remote -v
- git remote add upstream
<https://github.com/KENNYSOFT/github-hol.git>

(18) fetch

원격 저장소의 브랜치 가져오기

(19) pull

fetch + merge

pull.rebase 설정을 통해
fetch + rebase가 되도록 바꿀 수 있다.

[6] upstream 따라가기

KENNYSOFT2/github-hol

GitHub, Inc. [US] | https://github.com/KENNYSOFT2/github-hol

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

Unwatch 1 Star 0 Fork 1

Code Pull requests 0 Projects 0 Wiki Insights Settings

GitHub Hands-On Lab (7/12) Edit

Add topics

11 commits 1 branch 2 releases 1 contributor

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

This branch is 1 commit behind KENNYSOFT:master.

Pull request Compare

Commit	Message	Time
KENNYSOFT	Add guestbook; resolve #1	Latest commit 1e7338e 14 minutes ago
guestbook	Add guestbook; resolve #1	14 minutes ago
msptestapp	Merge branch 'query-delete-2'	an hour ago
.gitignore	Ignore db connection info	5 hours ago
README.md	Add readme	15 hours ago

README.md

GitHub: Hands-On Lab (7/12)

w/ PaaS Source Code

[6] upstream 따라가기

- git fetch upstream main
- git merge upstream/main
- git push

성공시 1 behind가 다시 even이 됨

[7] 파일 편집

The screenshot shows a GitHub repository page for 'KENNYSOFT2 / github-hol'. The URL is https://github.com/KENNYSOFT2/github-hol/tree/master/guestbook. The page displays a commit history with one commit from 'KENNYSOFT' titled 'Add guestbook: resolve #1'. At the top right, there is a red circle highlighting the 'Create new file' button in the navigation bar.

Branch: master [github-hol / guestbook /](#)

This branch is 1 commit behind KENNYSOFT:master.

KENNYSOFT Add guestbook: resolve #1

..

TEMPLATE.md Add guestbook: resolve #1 31 minutes ago

Create new file Upload files Find file History

Unwatch 1 Star 0 Fork 1

© 2018 GitHub, Inc. Terms Privacy Security Status Help

Contact GitHub API Training Shop Blog About

[7] 파일 편집 (Cont'd)

The screenshot shows a web browser window with a GitHub repository page. The URL in the address bar is <https://github.com/KENNYSOFT2/github-hol/new/master/guestbook>. The page title is "KENNYSOFT2 / github-hol". The repository was forked from "KENNYSOFT/github-hol". There are 1 unwatched, 0 starred, and 1 forked repository. The main navigation tabs are "Code", "Pull requests 0", "Projects 0", "Wiki", "Insights", and "Settings". A search bar at the top says "Search or jump to...". Below the search bar, there are buttons for "Unwatch", "Star", and "Fork". The main content area shows a file named "KENNYSOFT2.md" with the following content:

```
1 - Name: KENNYSOFT
2 - Score: 10/10
3 - Feedback: [Wow!](https://www.youtube.com/watch?v=PZxPUM4bUzY)
```

Below the file content, there are buttons for "Edit new file" and "Preview". To the right of the preview button are dropdown menus for "Spaces", "2", and "No wrap".

[7] 파일 편집 (Cont'd)

The screenshot shows a GitHub repository page for 'KENNYSOFT2 / github-hol'. The repository is a fork from 'KENNYSOFT/github-hol'. The 'Code' tab is selected. The 'Branch: master' dropdown shows 'github-hol / guestbook /'. The commit history lists:

- KENNYSOFT2 Create KENNYSOFT2.md (Latest commit 0904165 just now)
- ..
- KENNYSOFT2.md (Create KENNYSOFT2.md just now)
- TEMPLATE.md (Add guestbook: resolve #1 32 minutes ago)

At the bottom, there are links for GitHub: © 2018 GitHub, Inc. Terms Privacy Security Status Help, Contact GitHub API Training Shop Blog About.

[8] Pull Request

Pull Requests - KENNYSOFT2 / github-hol

KENNYSOFT2 / [github-hol](#)
forked from [KENNYSOFT/github-hol](#)

Unwatch 1 Star 0 Fork 1

Code Pull requests 0 Projects 0 Wiki Insights Settings

Filters is:pr is:open Labels Milestones New pull request

Welcome to Pull Requests!

Pull requests help you collaborate on code with other people. As pull requests are created, they'll appear here in a searchable and filterable list. To get started, you should [create a pull request](#).

ProTip! Click a checkbox on the left to edit multiple issues at once.

© 2018 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)

Contact GitHub API Training Shop Blog About

<https://github.com/KENNYSOFT2/github-hol/compare>

[8] Pull Request (*Cont'd*)

The screenshot shows a GitHub comparison page for the repository `KENNYSOFT / github-hol`. The URL is `https://github.com/KENNYSOFT/github-hol/compare/master...KENNYSOFT2:master`. The comparison is between the `base fork: KENNYSOFT/github-hol` and the `head fork: KENNYSOFT2/github-hol`, with the `base: master` and `compare: master` branches selected.

A red circle highlights the green **Create pull request** button. Below it, the page displays the following information:

- Commits on Jul 12, 2018**:
 - KENNYSOFT** created `KENNYSOFT2.md`
- Showing 1 changed file with 3 additions and 0 deletions.**
- Unified** and **Split** view options.
- Code diff:**

```
3  [REDACTED] guestbook/KENNYSOFT2.md
...
1  +- Name: KENNYSOFT
2  +- Score: 10/10
3  +- Feedback: [Wow!](https://www.youtube.com/watch?v=PZxPUM4bUzY)
```

[8] Pull Request (*Cont'd*)

The screenshot shows a GitHub pull request page for the repository 'KENNYSOFT/github-hol'. The pull request is titled 'Create KENNYSOFT2.md #2'. It has 1 commit from 'KENNYSOFT' merged into the 'master' branch. The pull request is currently open. The commit message is 'Create KENNYSOFT2.md' and it is verified. There are no conflicts with the base branch. The pull request has 1 review, 0 stars, and 1 fork. The repository has 1 pull request, 0 issues, 0 projects, and 0 wiki pages. The pull request details show a comment from 'KENNYSOFT' stating 'No description provided.' and a note that the branch has no conflicts with the base branch. The pull request can be edited.

Create KENNYSOFT2.md #2

Open KENNYSOFT wants to merge 1 commit into `KENNYSOFT:master` from `KENNYSOFT2:master`

Conversation 0 Commits 1 Checks 0 Files changed 1 +3 -0

KENNYSOFT commented just now
No description provided.

Create KENNYSOFT2.md Verified 0904165

Add more commits by pushing to the `master` branch on KENNYSOFT/github-hol.

This branch has no conflicts with the base branch
Only those with [write access](#) to this repository can merge pull requests.

Write Preview

Leave a comment

Reviewers
No reviews

Assignees
No one assigned

Labels
None yet

Projects
None yet

Milestone
No milestone

Notifications
Unsubscribe

You're receiving notifications because

[8] Pull Request (*Cont'd*)

The screenshot shows a GitHub pull request page for the repository 'KENNYSOFT / github-hol'. The pull request is titled 'Create KENNYSOFT2.md #2'. A green button labeled 'Open' indicates that KENNYSOFT2 wants to merge 1 commit into 'KENNYSOFT:master' from 'KENNYSOFT2:master'. The pull request has 1 commit, 0 checks, and 1 file changed. The commit message is 'Create KENNYSOFT2.md' and it is verified. The pull request is a first-time contributor.

Merge pull request dropdown menu:

- Create a merge commit** (selected): All commits from this branch will be added to the base branch via a merge commit.
- Squash and merge**: The 1 commit from this branch will be added to the base branch.
- Rebase and merge**: The 1 commit from this branch will be rebased and added to the base branch.

Continuous integration status: Continuous integration has not been set up. Several apps are available to automatically catch bugs and enforce style.

Branch conflicts: This branch has no conflicts with the base branch. Merging can be performed automatically.

Merge pull request button (circled in red with an orange arrow pointing to it).

PR을 받는 본계정 (Receiving PR) badge is visible in the top right corner.

[8] Pull Request (*Cont'd*)

The screenshot shows a GitHub pull request page for the repository 'KENNYSOFT/github-hol'. The pull request is titled 'Create KENNYSOFT2.md #2'. It has been merged by 'KENNYSOFT' from 'KENNYSOFT:master' into 'KENNYSOFT2:master' just now. The commit message is 'Create KENNYSOFT2.md'. The pull request has 1 commit, 0 files changed, and 0 checks. The status bar indicates '+3 -0' changes. The right sidebar shows options for Reviewers, Assignees, Labels, Projects, Milestone, and Notifications.

Create KENNYSOFT2.md #2

Merged KENNYSOFT merged 1 commit into KENNYSOFT:master from KENNYSOFT2:master just now

Conversation 0 Commits 1 Checks 0 Files changed 1

KENNYSOFT commented 3 minutes ago

No description provided.

Create KENNYSOFT2.md Verified 0904165

KENNYSOFT merged commit 9768778 into KENNYSOFT:master just now Revert

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

Comment

Reviewers

Assignees

Labels

Projects

Milestone

Notifications

Unsubscribe

[8] Pull Request (*Cont'd*)

KENNYSOFT / github-hol

Branch: master

Commits on Jul 12, 2018

- Merge pull request #2 from KENNYSOFT2/master ...
KENNYSOFT committed a minute ago
- Create KENNYSOFT2.md
KENNYSOFT2 committed 6 minutes ago
- Create _temp_for_upstream
KENNYSOFT committed 24 minutes ago
- Add guestbook; resolve #1
KENNYSOFT committed 38 minutes ago
- Merge branch 'query-delete-2'
KENNYSOFT committed 2 hours ago
- query: implement delete
KENNYSOFT committed 2 hours ago
- query: implement delete with drop
KENNYSOFT committed 2 hours ago
- query: implement add
KENNYSOFT committed 2 hours ago

Commits on Jul 11, 2018

[9] GitHub Pages

- git checkout --orphan gh-pages
- git rm --cached -r .
- git commit --allow-empty -m "Initial commit"
- git push origin gh-pages
- Orphaned 브랜치 관리는 상당히 귀찮기에 나머지 파일은 GitHub에서 직접 업로드한다.

[9] GitHub Pages (*Cont'd*)

The screenshot shows a GitHub repository page for 'KENNYSOFT/github-hol'. The repository has 1 commit, 2 branches, 2 releases, and 1 contributor. A red circle highlights the 'Upload files' button in the top right corner of the main content area.

KENNYSOFT / github-hol

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

GitHub Hands-On Lab (7/12) Edit

Add topics

1 commit 2 branches 2 releases 1 contributor

Branch: gh-pages New pull request Create new file Upload files Find file Clone or download

KENNYSOFT Initial commit Latest commit fddc732 8 minutes ago

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

© 2018 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub API Training Shop Blog About

blame

The screenshot shows a GitHub blame interface for the file `MainForm.cs`. The interface displays the code with line numbers and the commit history for each line. The commits are listed on the left, and the corresponding code changes are shown on the right.

Commit	Date	Line Number	Code
[KMS] Tooltip.CashPackage	2 years ago	1844	private Wz_File findItemWz()
		1845	{
		1846	foreach (Wz_Structure wz in openedWz)
		1847	{
		1848	foreach (Wz_File file in wz.wz_files)
		1849	{
		1850	if (file.Type == Wz_Type.Item)
		1851	{
		1852	return file;
		1853	}
		1854	}
		1855	}
		1856	return null;
		1857	}
		1858	
[KMS] UI.Search (2)	2 years ago	1859	private IEnumerable<KeyValuePair<int, StringResult>> searchStringLinker(IEnumerable<Dictionary<int, StringResult>> dicts, string key, bool exact
init	3 years ago	1860	{
		1861	string[] match = key.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);
		1862	foreach (Dictionary<int, StringResult> dict in dicts)
		1863	{
		1864	foreach (KeyValuePair<int, StringResult> kv in dict)
		1865	{
		1866	if (exact)
		1867	{
[KMS] UI.Search (2)	2 years ago	1868	if (string.Compare(kv.Key.ToString(), key, StringComparison.OrdinalIgnoreCase) == 0 string.Compare(kv.Value.Name, key, StringComparison.OrdinalIgnoreCase) == 0 string.Com
init	3 years ago	1869	yield return kv;
		1870	}
		1871	else
		1872	{

Wiki

Home - KENNYSOFT/github-hol x

GitHub, Inc. [US] | https://github.com/KENNYSOFT/github-hol/wiki

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

Your Wiki was created. X

 [KENNYSOFT / github-hol](#) Unwatch 1 Star 0 Fork 1

[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#) [Settings](#)

Home

Hyeonmin Park edited this page just now · 1 revision

Welcome to the github-hol wiki!

▼ Pages 1
[Home](#)

+ Add a custom sidebar

Clone this wiki locally
<https://github.com/KENNYSOFT/github-hol> Copy
[Clone in Desktop](#)

+ Add a custom footer

© 2018 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)  [Contact GitHub](#) [API](#) [Training](#) [Shop](#) [Blog](#) [About](#)

GitHub Gist

The screenshot shows a web browser window displaying the GitHub Gist interface. The URL in the address bar is <https://gist.github.com>. The page title is "GitHubGist". The main content area shows a gist titled "github-hol.md" with the following content:

```
1 # GitHub: Hands-On Lab (7/12)
2
3 w/ PaaS Source Code
4
5 - Slide [[PDF]](https://kennysoft.github.io/github-hol/[18.07.12]%20GitHub.pdf) [[PowerPoint]](https://kennysoft.github.io/github-hol/[18.07.12]%20GitHub.pptx)
6 - [Providing](https://kennysoft.github.io/github-hol/providing.zip)
7 - [Result repository](https://github.com/KENNYSOFT/github-hol)
```

Below the code editor, there are buttons for "Add file", "Create secret gist", and "Create public gist".

GitHub Gist (*Cont'd*)

The screenshot shows a web browser window with a GitHub Gist page. The URL in the address bar is <https://gist.github.com/KENNYSOFT/4cf28c5d09382553e18c0eaee11f134e7>. The page title is "github-hol.md". The main content area displays the file "github-hol.md" which contains the following text:

```
GitHub: Hands-On Lab (7/12)

w/ PaaS Source Code

• Slide [PDF] [PowerPoint]
• Providing
• Result repository
```

Below the code editor, there is a comment section with a "Leave a comment" input field and a "Comment" button.



Chapter 2-3

GITHUB DESKTOP

GitHub Desktop 설치

<https://desktop.github.com>

GitHub Desktop

The screenshot shows the GitHub Desktop application interface. The top navigation bar includes File, Edit, View, Repository, Branch, and Help. The current repository is "WzComparerR2" and the branch is "master". A "Fetch origin" status indicates it was fetched just now.

The main area displays a list of 49 changed files. One file, "AvatarForm.cs", is selected and shown in a detailed diff view. The diff highlights the following changes:

```
diff --git a/WzComparerR2.Avatar\UI\AvatarForm.cs b/WzComparerR2.Avatar\UI\AvatarForm.cs
--- a/WzComparerR2.Avatar\UI\AvatarForm.cs
+++ b/WzComparerR2.Avatar\UI\AvatarForm.cs
@@ -210,6 +210,6 @@ private void OnNewPartAdded(AvatarPart part)
@@ -226,22 +226,32 @@ for (int i = 4; i < 16; i++)
{
    if (part == avatar.Parts[i])
    {
        string action = i == 12 ? this.cmbEmotion.Text : this.cmbActionBody.Text;
        var cmb = i == 12 ? this.cmbEmotionFrame : this.cmbBodyFrame;
        var effectNode = PluginBase.PluginManager.FindWz($"Effect\\ItemEff.img\\{part.ID}\\effect\\{action}") ?? PluginBase.PluginManager.FindWz($"Effect\\ItemEff.img\\{part.ID}\\effect\\default");
        if (effectNode != null)
        {
            effectNode.effectDelay = effectDelays[i];
        }
    }
}
```

The commit message is "Summary (required)".

GitHub Desktop (*Cont'd*)

The screenshot shows the GitHub Desktop application interface. The top navigation bar includes File, Edit, View, Repository, Branch, and Help. The current repository is "WzComparerR2" and the branch is "master". A "Fetch origin" status message indicates it was last fetched just now.

The main area displays a list of 49 changed files. One file, "AvatarForm.cs", is selected and shown in a detailed diff view. The diff highlights changes made between lines 67 and 70. The changes include the addition of a field "ComboFrame[] effectDelays;" and the removal of fields "BackgroundWorker workerExport;" and "ProgressForm dialogProgress;". It also removes a string variable "lastPath;". Lines 69 and 70 introduce new fields "BackgroundWorker worker;" and "ProgressDialog progressDialog;". The commit message for this change is partially visible, mentioning "private void OnNewPartAdded(AvatarPart part)" and "avatar.Longcoat.Visible = false;".

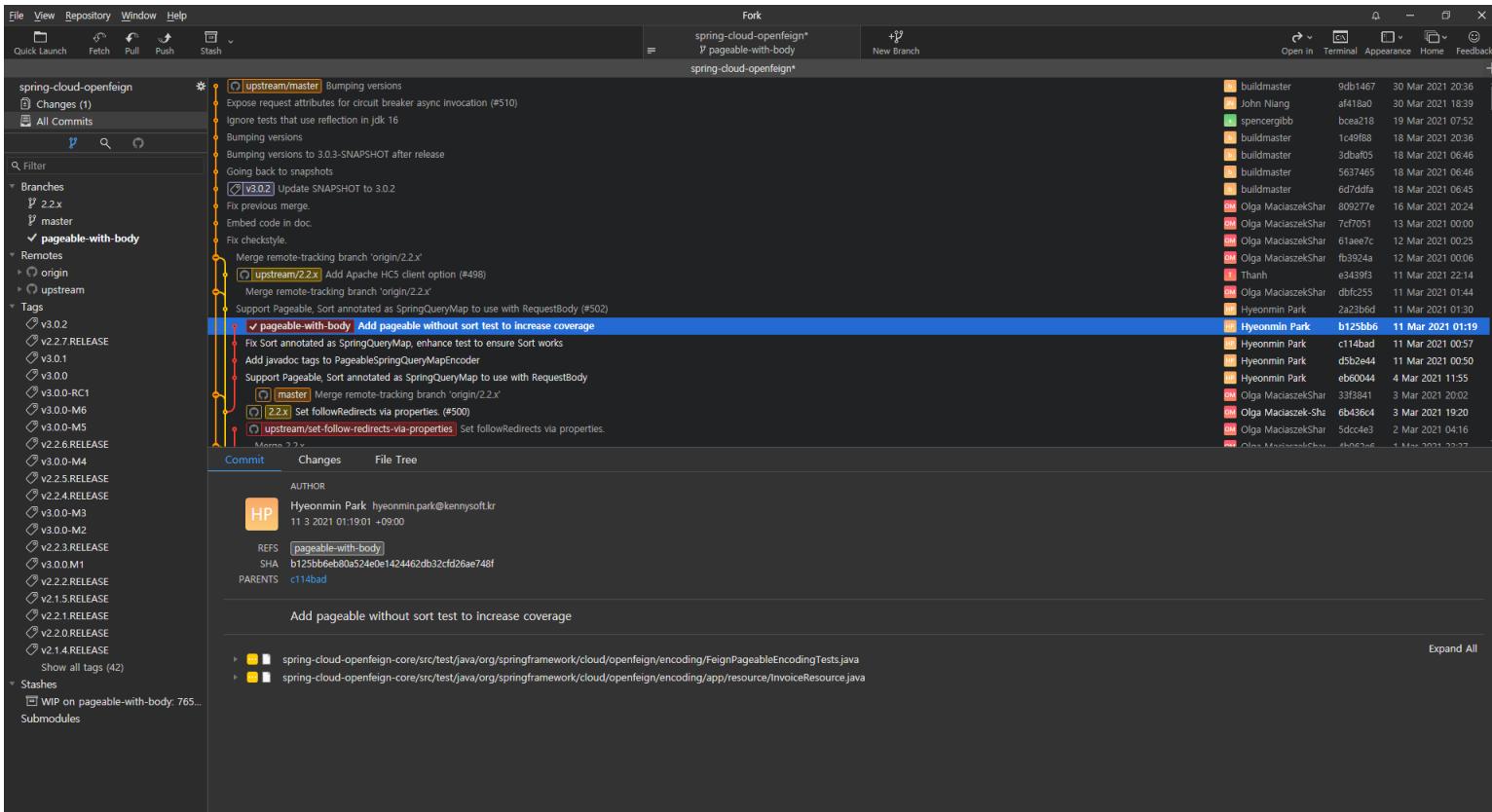
A yellow annotation "Hunk 단위로 선택 가능" is overlaid on the code editor, indicating that changes can be selected at the hunk level. The bottom left of the interface shows a summary button ("Summary (required)", "Description", and a plus sign), and a "Commit to master" button.

```
51 64     bool suspendUpdate;
52 65     bool needUpdate;
53 66     Animator animator;
54 67 +     ComboFrame[] effectDelays;
55 68 -     BackgroundWorker workerExport;
56 69 -     ProgressForm dialogProgress;
57 70 -     string lastPath;
58 71
59 72     /// <summary>
60 73     /// wz1节点选中事件。
@@ -210,6 +223,44 @@ private void OnNewPartAdded(AvatarPart part)
210 223             avatar.Longcoat.Visible = false;
211 224         }
212 225     }
226 +     for (int i = 4; i < 16; i++)
227 +     {
228 +         if (part == avatar.Parts[i])
229 +         {
230 +             string action = i == 12 ? this.cmbEmotion.Text : this.cmbActionBody.Text;
231 +             var cmb = i == 12 ? this.cmbEmotionFrame : this.cmbBodyFrame;
232 +             var effectNode = PluginBase.PluginManager.FindWz($"Effect\\ItemEff.img\\{part.ID}\\effect\\{action}") ?? PluginBase.PluginManager.FindWz($"Effect\\ItemEff.img\\{part.ID}\\effect\\default");
233 +             if (effectNode != null)
234 +             {
235 +                 effectNode.wzFile.Close();
```

Fork

<https://fork.dev>

Fork (*Cont'd*)



내돈내산 리뷰



The background of the slide features a subtle, out-of-focus graphic of overlapping circles. There are four main circles: a large light blue circle on the right, a medium green circle to its left, a smaller yellow-green circle below the green one, and a very small yellow-green circle at the top left. The overall effect is a clean, modern, and slightly abstract design.

Appendix

REFERENCE

Reference

- <https://github.com/Taeung/git-training>
- [https://ko.wikipedia.org/wiki/%EA%B9%83_\(%EC%86%8C%ED%94%84%ED%8A%B8%EC%9B%A8%EC%96%B4\)](https://ko.wikipedia.org/wiki/%EA%B9%83_(%EC%86%8C%ED%94%84%ED%8A%B8%EC%9B%A8%EC%96%B4))
- “Git” 강의자료 by 이기용 교수님 (숙명여자대학교)

Q&A

감사합니다