



# JAVA 웹 개발자 양성과정

## Git과 GitHub

### 3강 – 버전 컨트롤

By SoonGu Hong



# JAVA 웹 개발자 양성과정

## Git과 GitHub

### 1. 버전 정보 읽기

## \* 버전 정보 읽기 주요 명령어

\$ **git status** : working tree의 정보를 보여줌  
\$ **git diff** : working tree의 변경사항을 보여줌  
\$ **git log** : commit된 버전정보를 보여줌  
\$ **git log --stat** : 버전정보를 자세히 보여줌  
\$ **git log -p** : 버전정보를 더 자세히 보여줌

## \* 버전 정보 읽기 실습

1. 실습 폴더에 study라는 폴더를 만들고 그 안에 hello.txt를 만들자! 내용은 아무렇게나 적으세요.  
그리고 \$ git status 입력!

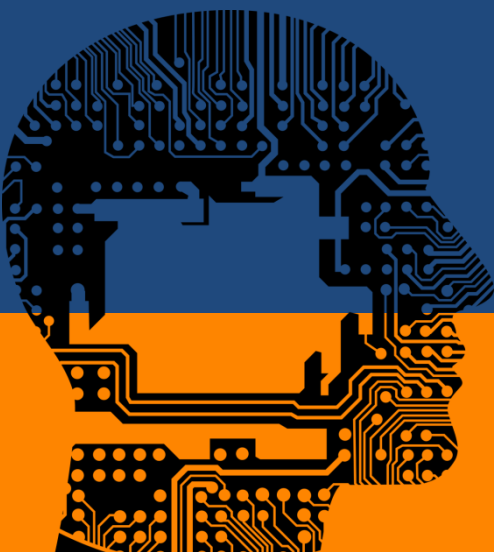
```
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   study/hello.txt
```

2. 커밋으로 버전을 만들어 보시다.

```
KITT@kitt-23 MINGW64 /d/git_practice (master)
$ git commit -m "study폴더에 hello.txt 생성함"
```

3. hello.txt의 내용을 수정한 뒤 \$ git diff 로 변경사항 확인!

```
KITT@kitt-23 MINGW64 /d/git_practice (master)
$ git diff
diff --git a/study/hello.txt b/study/hello.txt
index c25ed0e..a538016 100644
--- a/study/hello.txt
+++ b/study/hello.txt
@@ -1,2 +1,3 @@
 hello!
-
+bye~~~
\ No newline at end of file
+엄준식파이팅!
\ No newline at end of file
```



# JAVA 웹 개발자 양성과정

## Git과 GitHub

### 2. 버전간 이동 checkout

## \* checkout



## \* checkout 실습

1. 전 단계에서 수정한 hello.txt를 커밋합니다.

```
$ git commit -m "study폴더 hello.txt 수정"
```

2. git log로 커밋목록을 확인 후 이동하고 싶은 커밋의 아이디를 복사합니다.

```
commit 213adceb98d787948ecde5cb4afeaf86b7472836
Author:
Date: Tue Dec 8 14:52:37 2020 +0900

study폴더에 hello.txt 생성함
```

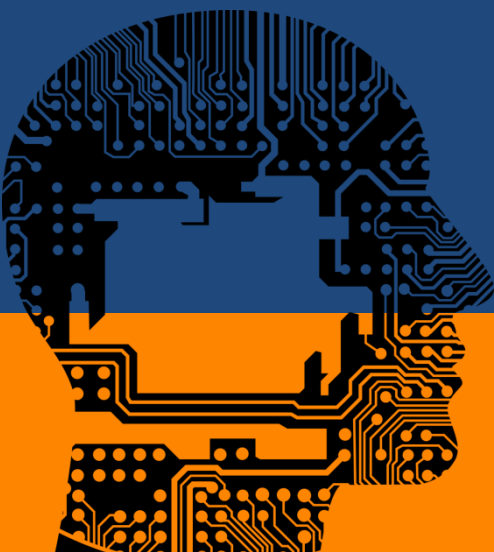
여기!!  
복사

3. \$ git checkout 복사한 커밋 아이디

```
KITT@kitt-23 MINGW64 /d/git_practice (master)
$ git checkout 213adceb98d787948ecde5cb4afeaf86b7472836
```

4. 실제로 폴더에 가보면 수정사항 반영 전으로 돌아가 있습니다.  
그리고 언제든지 커밋의 아이디를 통해 이동할 수 있어요~~
5. 다시 원상태로 돌아가고 싶다면??

```
$ git checkout master
```



# JAVA 웹 개발자 양성과정

## Git과 GitHub

### 3. 버전삭제 reset



## \* reset 실습

1. \$ git log --oneline 명령으로 커밋 아이디를 체크해요  
**여기서 주의할점!** 저처럼 커밋 아이디를 스샷찍어놓으세요.  
이따 복구할 때 필요합니다!

```
KITT@kitt-23 MINGW64 /d/git_practice (master)
$ git log --oneline
4c81855 (HEAD -> master) study폴더 hello.txt 수정함
213adce study폴더에 hello.txt 생성함
f88c316 (origin/master) 메인 페이지를 추가!
f3adab0 README.md를 추가!
```

2. \$ git reset --soft f88c316 으로 2번 커밋으로 돌아가 보아요.

```
KITT@kitt-23 MINGW64 /d/git_practice (master)
$ git reset --soft f88c316

KITT@kitt-23 MINGW64 /d/git_practice (master)
$ git log --oneline
f88c316 (HEAD -> master, origin/master) 메인 페이지를 추가!
f3adab0 README.md를 추가!
```

3. 참고로 **--soft** 옵션은 reset시에 해당 커밋으로 돌아가며 다른 커밋을 삭제하지만 파일은 unstaged 상태로 돌려놔요. 즉 커밋은 사라지지만 수정 파일은 지우지 않는다는 점~~
4. **--hard** 옵션을 주면 이후 커밋시에 만든 파일들도 짤없이 날아가버린답니다. 물론 사라진 커밋아이디를 기억하고 있다면 reset으로 복구도 가능해요!! 마지막으로 \$ git reset --hard 4c81855로 복구하고 마칠게요!

**감사합니다**  
**THANK YOU**