

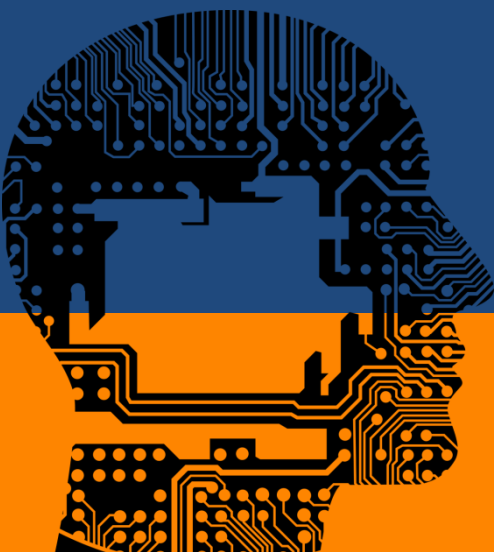


JAVA 웹 개발자 양성과정

Git과 GitHub

4강 – branch

By SoonGu Hong



JAVA 웹 개발자 양성과정

Git과 GitHub

1. branch

* branch??

- 프로그램을 개발할 때 여러 명이 작업할 경우 동일한 소스코드를 함께 공유하고 다루게 됩니다.
- 이 때 동일 버전에서 각자 맡은 일을 **독립적**으로 하고 싶을 때 브랜치를 만들어 사용합니다.

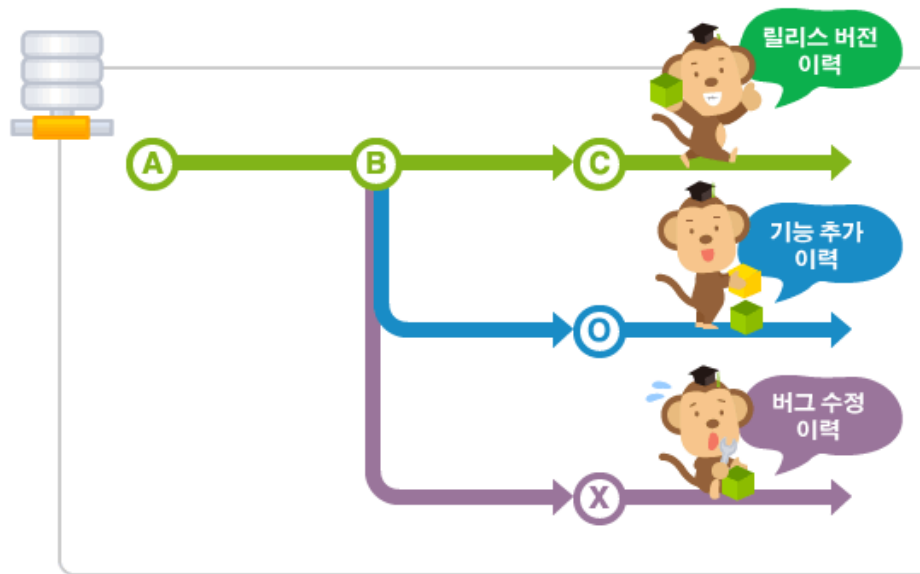


그림 출처: https://backlog.com/git-tutorial/kr/stepup/stepup1_1.html

* branch관련 주요 명령어

\$ **git branch** : 생성된 브랜치 목록 보기
\$ **git branch 브랜치명** : 새로운 브랜치 생성
\$ **git log --all --graph --oneline** :
커밋로그를 전부 보여주며(all옵션), 그래프형태로(graph
옵션), 간략히(oneline) 보여줌
\$ **git checkout 브랜치명** : 해당 브랜치로 이동

* branch 만들기 실습

1. \$ git branch로 만들어진 브랜치목록을 봅니다. master 브랜치가 있어요. 그러면 \$ git branch doolri 명령어를 입력해 봅시다.

```
$ git branch doolri
```

2. \$ git log --all --graph --oneline 을 입력해봅니다.

```
$ git log --all --graph --oneline
* 4c81855 (HEAD -> master, doolri) study폴더 hello.txt 수정함
* 213adce study폴더에 hello.txt 생성함
* f88c316 (origin/master) 메인 페이지를 추가!
* f3adab0 README.md를 추가!
```

3. 새로운 브랜치 dounat를 만들고 log명령어를 통해 확인합니다.

```
* 4c81855 (HEAD -> master, dounat, doolri) study폴더 hello.txt 수정함
* 213adce study폴더에 hello.txt 생성함
* f88c316 (origin/master) 메인 페이지를 추가!
* f3adab0 README.md를 추가!
```

* branch 관리 실습

1. 현재 상태에서 study폴더에 team.md파일 생성 후 애드, 커밋 실행

```
* 0e3e6d5 (HEAD -> master) 우리 팀 목표 생성
* 4c81855 (dounat, doolri) study폴더 hello.txt 수정 함
* 213adce study폴더에 hello.txt 생성 함
```

2. \$ git checkout doolri 명령으로 브랜치를 master -> doolri로 이동

```
$ git checkout doolri
Switched to branch 'doolri'
```

```
* 0e3e6d5 (master) 우리 팀 목표 생성
* 4c81855 (HEAD -> doolri, dounat) study폴더 hello.txt 수정 함
```

team.md가 사라졌나요???

3. 이제 둘리 브랜치에서 study폴더에 doolri.md를 생성 후 애드 커밋.

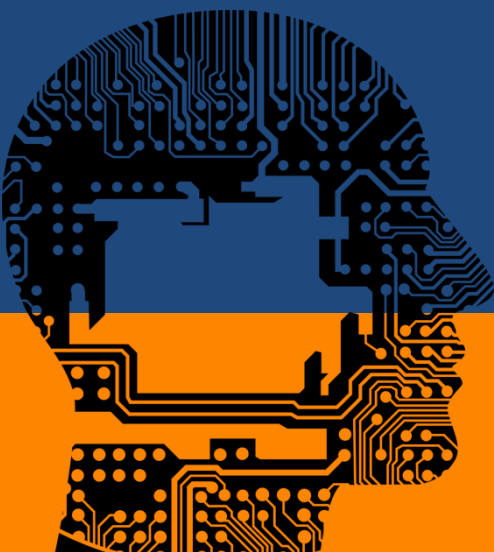
```
* 22f7086 (HEAD -> doolri) 둘리의 목표 추가
| * 0e3e6d5 (master) 우리 팀 목표 생성
| /
* 4c81855 (dounat) study폴더 hello.txt 수정 함
* 213adce study폴더에 hello.txt 생성 함
```

4. 도우너 브랜치에서도 비슷하게 해보세요!

JAVA 웹 개발자 양성과정

Git과 GitHub

2. merge



* merge??

- 브랜치를 나누어 작업한 결과는 최종적으로 통합 브랜치(master)에 합쳐야겠죠?
- 이 때 merge 명령을 통해 여러 개의 브랜치를 통합할 수 있습니다.

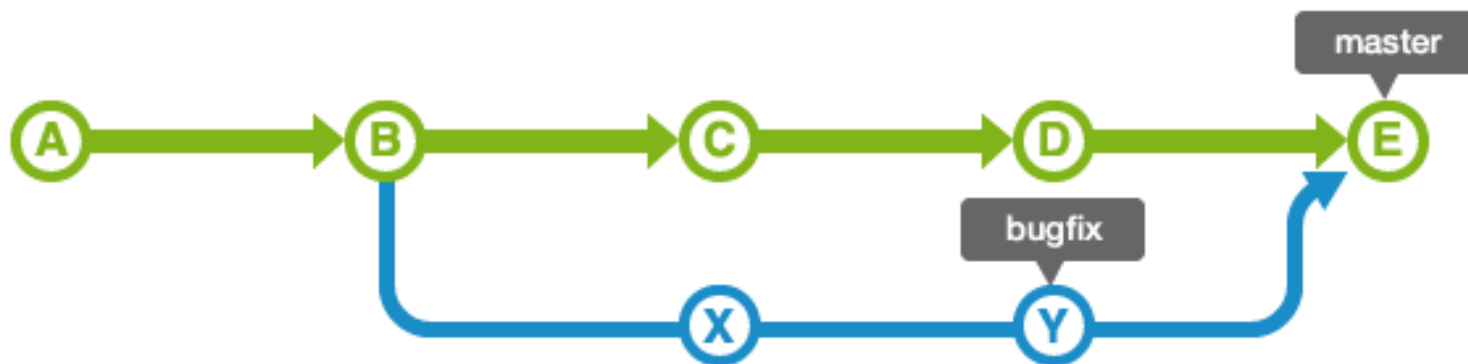


그림 출처: https://backlog.com/git-tutorial/kr/stepup/stepup1_4.html

* merge 실습

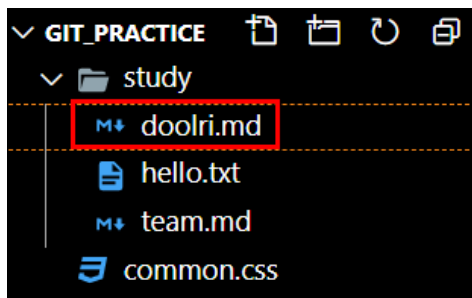
1. \$ git checkout master 명령으로 master로 HEAD를 이동시킵니다.
그리고 \$ git merge doolri 명령으로 master에 doolri를 병합시킵니다.

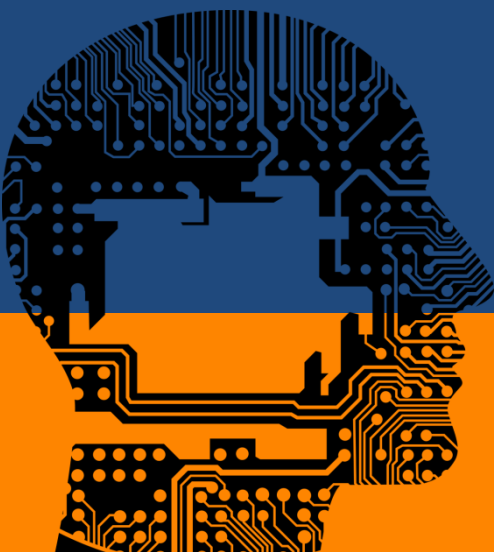
```
$ git merge doolri
```

2. 그러면 텍스트 에디터가 뜨면서 merge commit 메시지를 입력하라고 합니다. 커밋 메시지를 입력하고 로그를 봅니다.

```
* 55c3837 (HEAD -> master) Merge branch 'doolri'
|
| * 22f7086 (doolri) 둘 리 의 목 표 추 가
| * | 0e3e6d5 우 리 팀 목 표 생 성
| /
| * 6fde8b1 (dounat) 도 우 너 의 목 표 추 가
| /
| * 4c81855 study폴 더 hello.txt 수 정 함
```

3. 실제 master 브랜치의 마지막 커밋 워킹 디렉토리를 보면 둘리 브랜치에서 작업한 doolri.md파일이 들어온 것을 확인할 수 있습니다.





JAVA 웹 개발자 양성과정

Git과 GitHub

3. conflict

* conflict??

- 만약 둘리가 team.md파일에 “짜장면이 좋아” 라고 기록했는데 도우너가 같은 이름의 파일 team.md에 “짬뽕이 좋아”라고 기록한 뒤 두개를 merge하면 어떻게 될까요??
- 당연히 같은 이름의 파일의 같은 곳이 다르게 써져 있으니 병합하기 어렵겠죠?
- 이렇게 자동병합에 실패하는 경우를 conflict라고 합니다. 이 충돌은 수동으로 해결해줘야 합니다.

* conflict 실습

1. \$ git checkout dounat 명령으로 도우너 브랜치로 가서 team.md 파일을 생성한 뒤 애드 커밋합니다.
2. 마스터 브랜치로 체크아웃 한 뒤 \$ git merge dounat로 버전을 합쳐봅니다. 기존 마스터의 team.md파일과 도우너의 team.md가 충돌났다는 메시지와 함께 텍스트에디터가 아래와 같이 변경됩니다.

```
# Team GilDong
현재 변경 사항 수락 | 수신 변경 사항 수락 | 두
<<<<<<< HEAD (현재 변경 사항)
- 우리 팀은 짜장면으로 통일한다.
=====
- 우리 팀은 짬뽕으로 통일한다.
>>>>>>> dounat (수신 변경 사항)
```

3. 수동으로 수정 후 커밋하면 병합 충돌 해결 완료!!

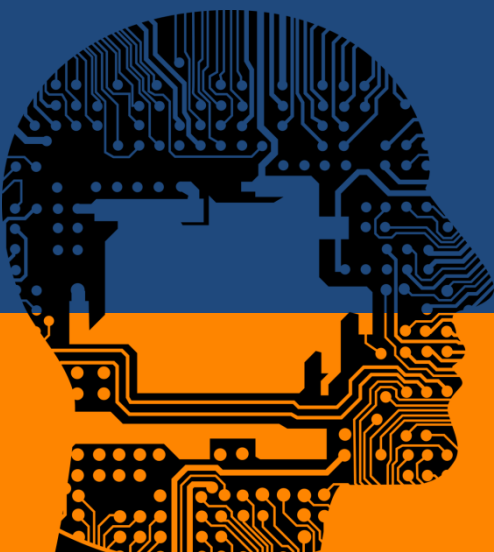
```
# Team GilDong
- 우리 팀은 짜장면으로 통일한다. 이게 법이다. 도우너 즐~
```

```
$ git commit -am "도우너 브랜치 Merge 완료"
```

JAVA 웹 개발자 양성과정

Git과 GitHub

4. branch 삭제



* branch 삭제 실습

1. 자 이제 master에서 출발한 2개의 브랜치 doolri와 dounat를 성공적으로 통합했습니다. 현재 로그를 봅시다.

```
* bc3a932 (HEAD -> master) 도우너 브랜치 Merge 완료
|
| * 8316c46 (dounat) team.md를 수정함
| * 6fde8b1 도우너의 목표 추가
| * | 55c3837 Merge branch 'doolri'
| |
| | * | 22f7086 (doolri) 둘리의 목표 추가
| |
| | * | 0e3e6d5 우리 팀 목표 생성
| |
| * 4c81855 study폴더 hello.txt 수정함
| * 213adce study폴더에 hello.txt 생성함
```

2. 이제 2개의 브랜치가 필요없어졌네요. 지워봅시다.
\$ git branch -d doolri
\$ git branch -d dounat

간단하죠??

감사합니다
THANK YOU