

Git pull, stash

김형준

Git pull

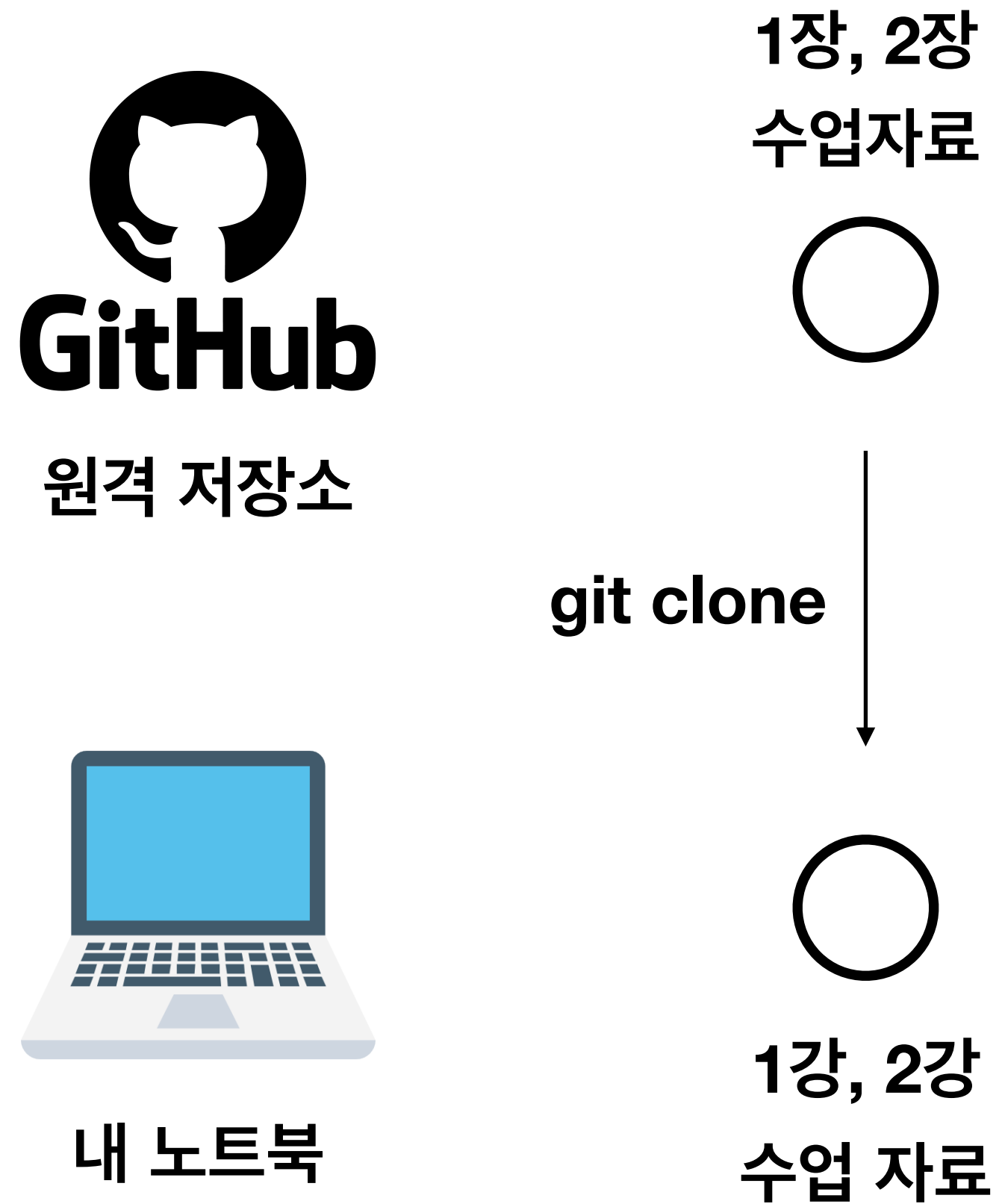
수업 자료 공유 원격 저장소에 새 강의자료가 업로드 됐습니다.
Git pull 명령어로 이를 내려받을 수 있습니다.

Bash ▾

```
git pull origin main
```

Git pull

지난 시간에 깃허브에 업로드 된 강의자료를
내 노트북으로 `git clone` 명령어로 다운 받았습니다.



Git pull

그 뒤로 수업을 들어면서 코드를 따라 치면서
수업 자료가 수정되었을 겁니다.

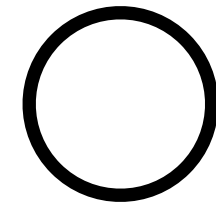


원격 저장소

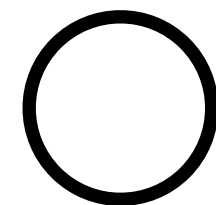


내 노트북

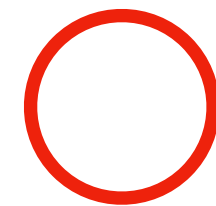
1장, 2장
수업자료



git clone



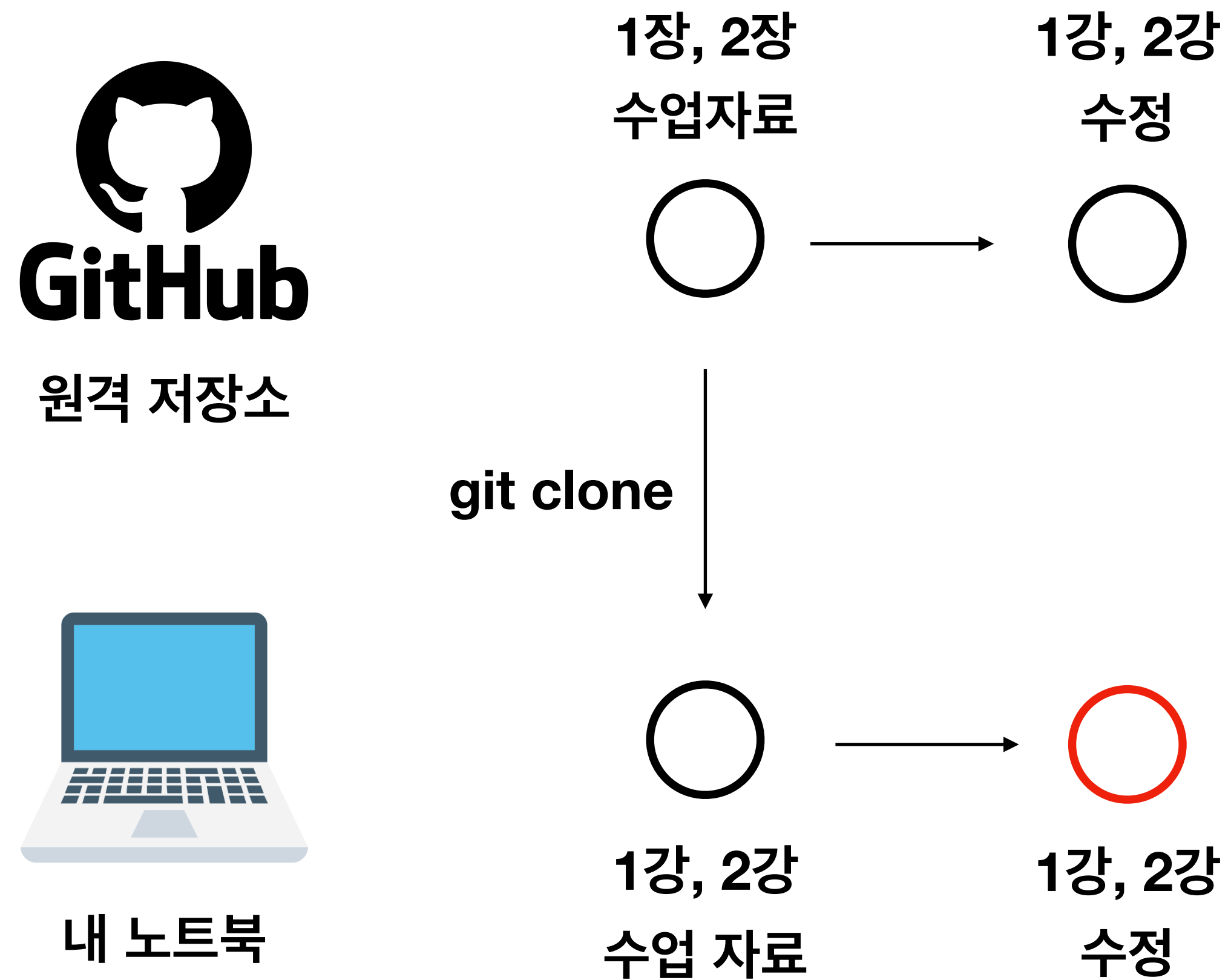
1강, 2강
수업 자료



수업들으면서
수정!

Git pull

수업 후에 저도 1강, 2강 수업때 수정한 코드를 커밋한 뒤, 업로드 했습니다.



Git pull

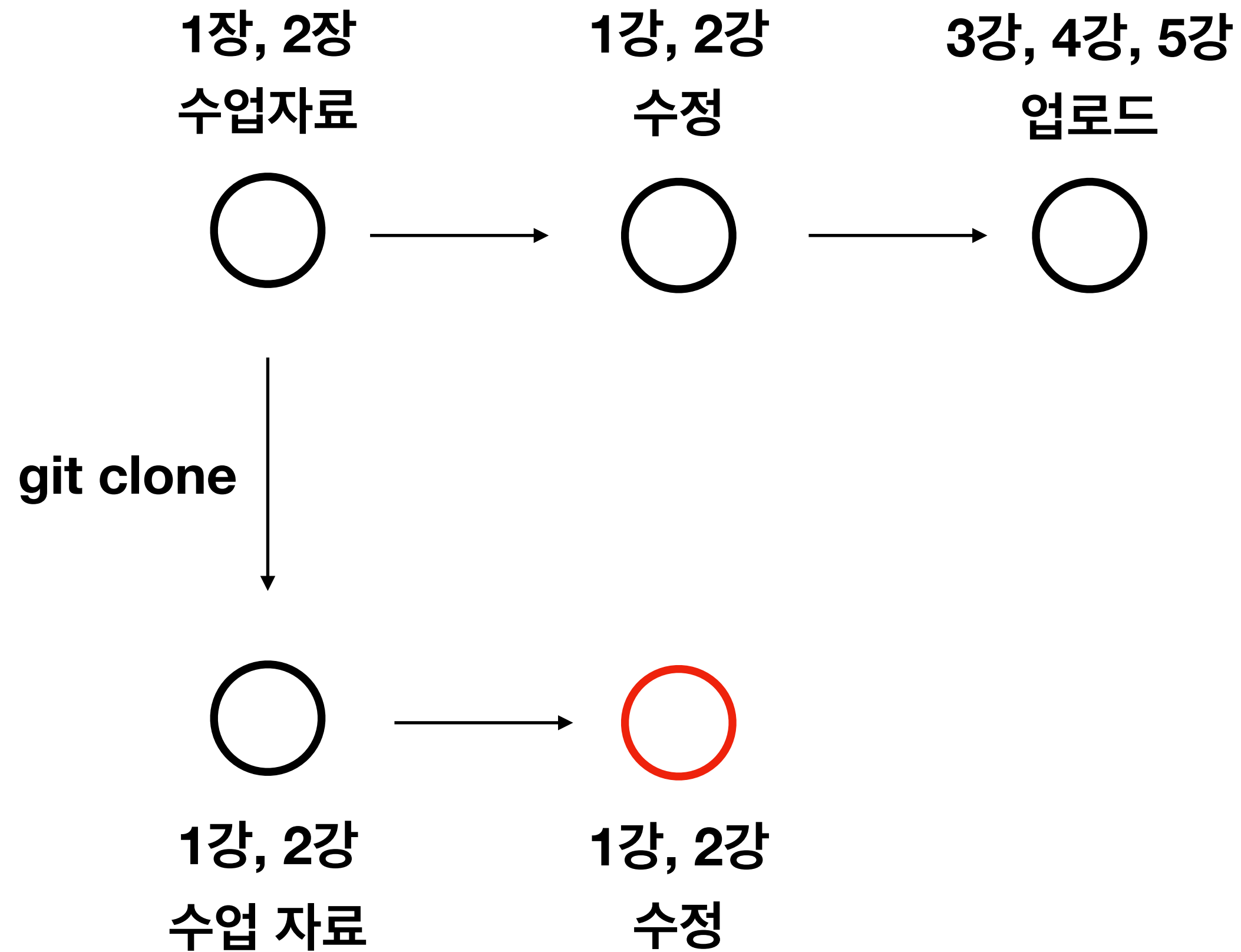
이 후, 3강, 4강, 5강이 업로드 되었습니다.



원격 저장소



내 노트북



Git pull

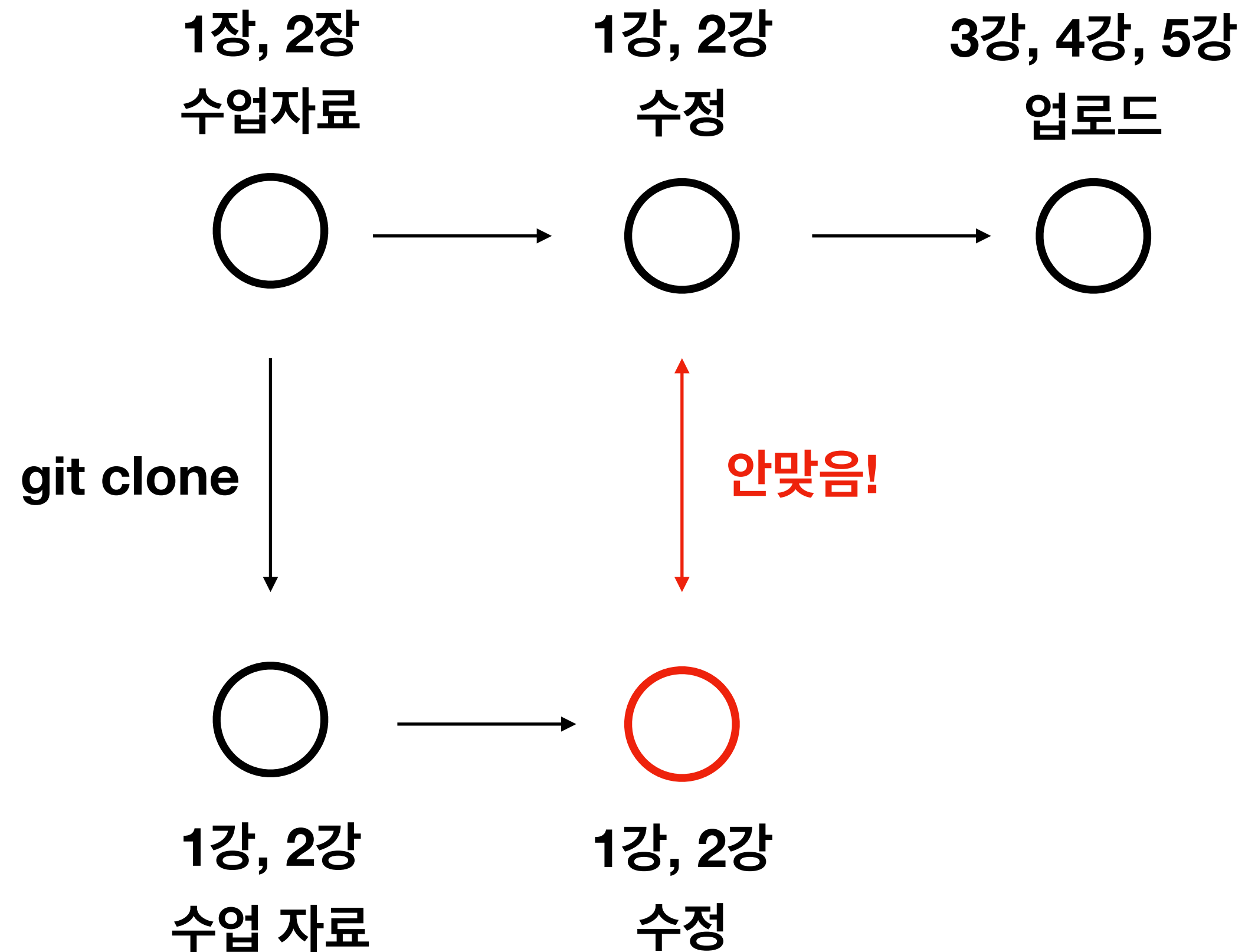
이를 그대로 git pull 명령어로 다운받으려고 하니,
1강과 2강 자료가 원격 저장소와 불일치합니다.



원격 저장소

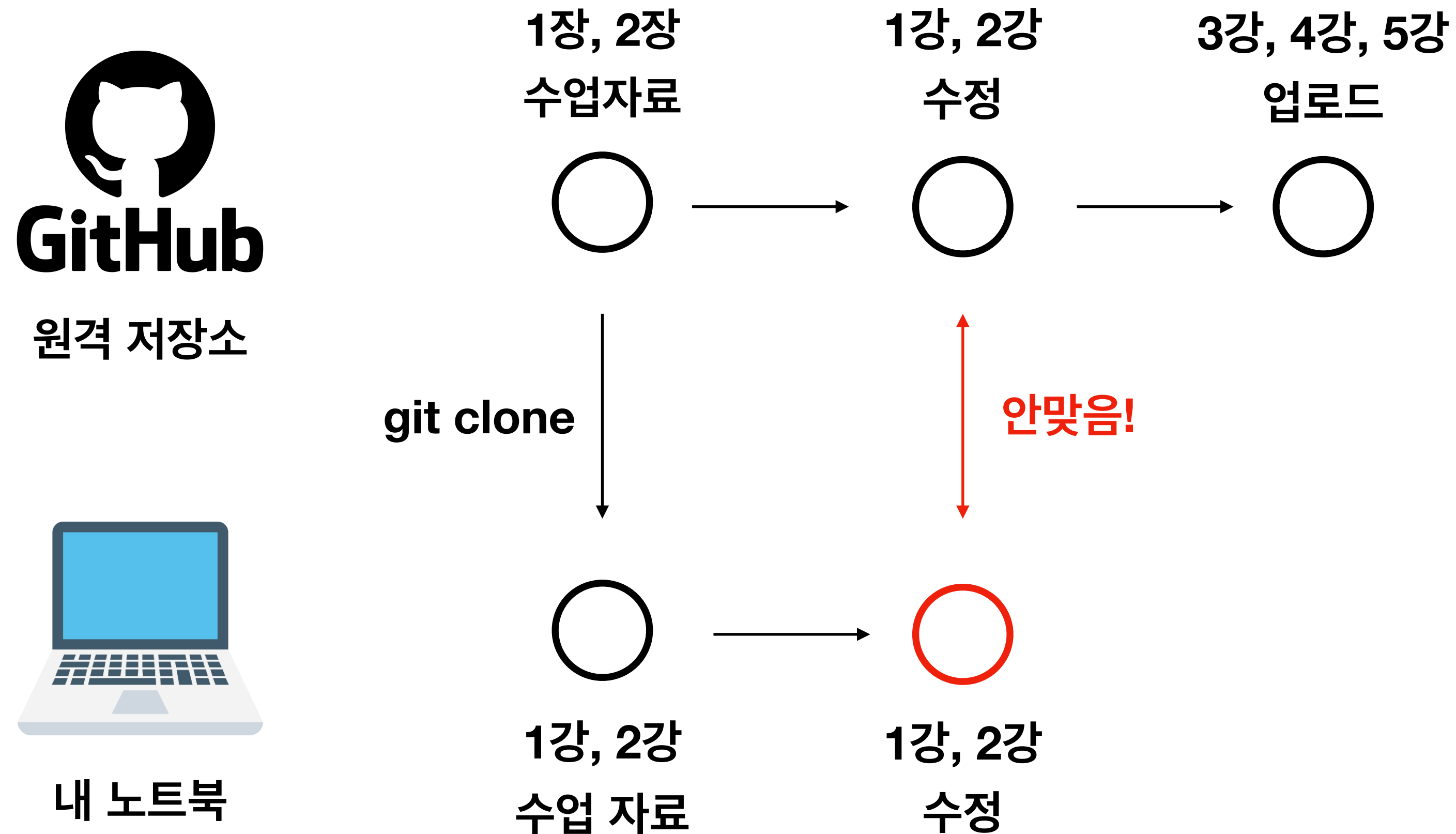


내 노트북



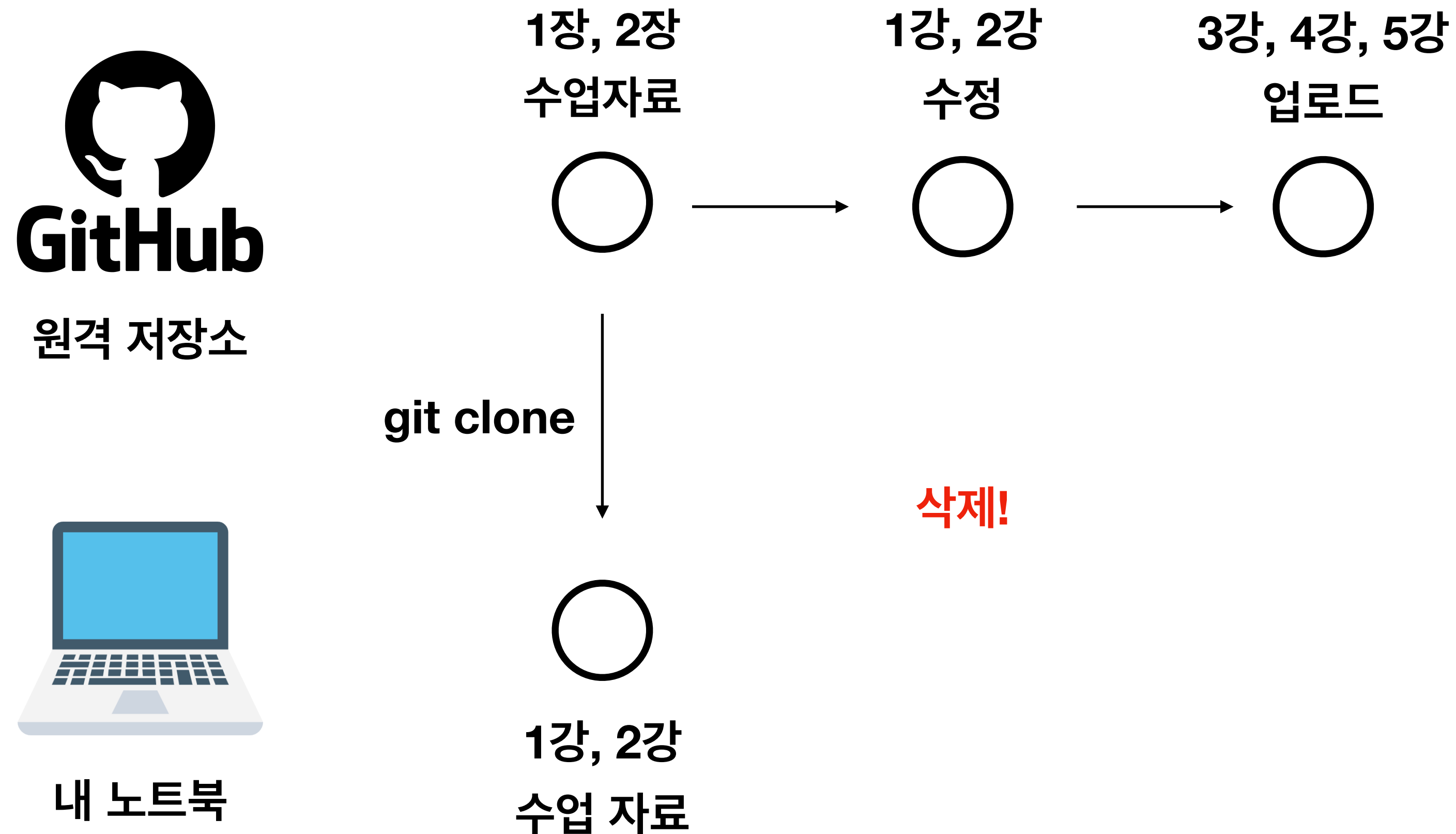
Git pull

이 문제를 해결하는 여러 방법 중에서 가장 쉬운 방법을 소개하겠습니다.
바로 내가 수정한 내역을 날려버리고, 원격 저장소의 코드를 내려받는 방법입니다.



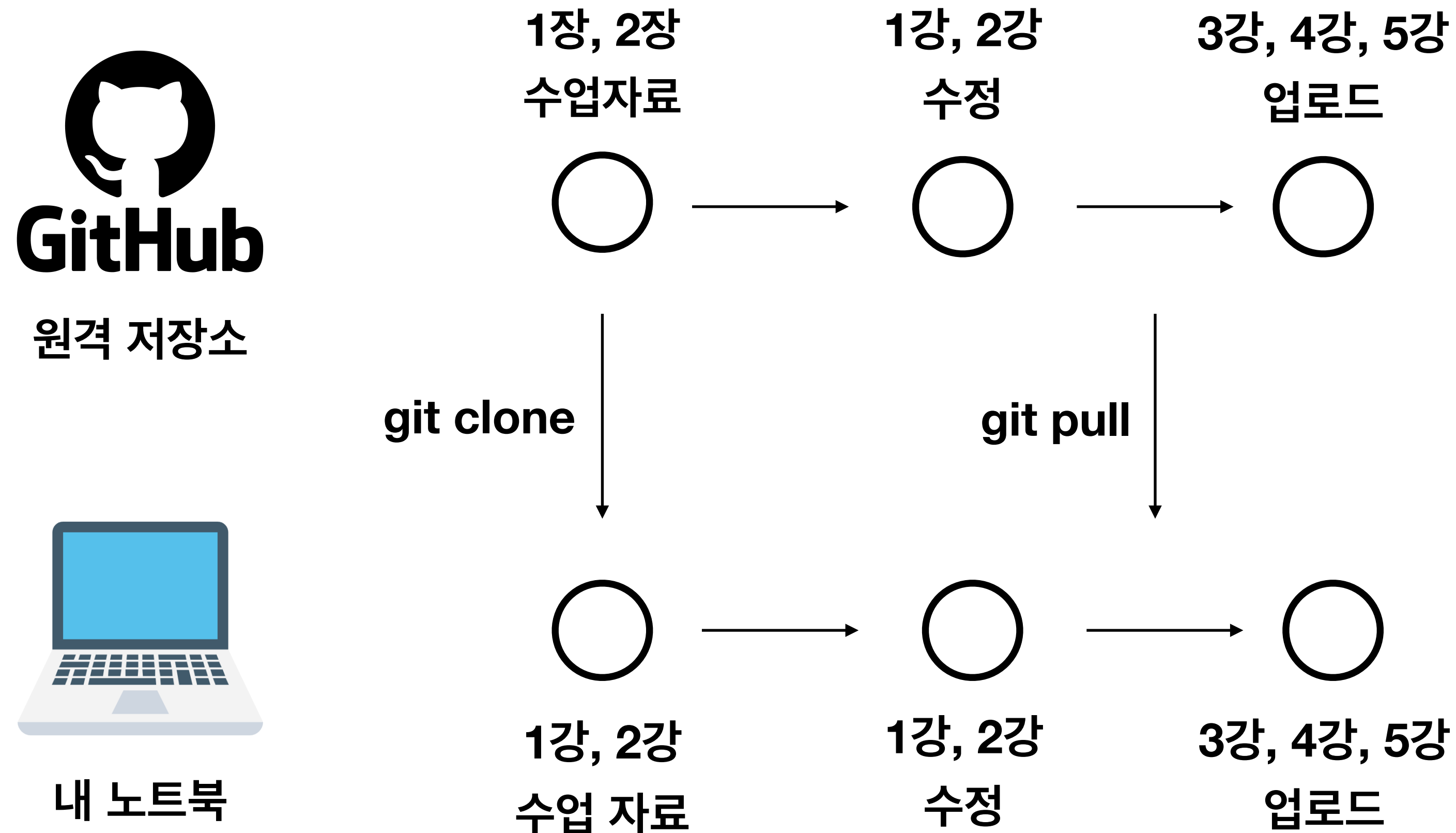
Git pull

이 문제를 해결하는 여러 방법 중에서 가장 쉬운 방법을 소개하겠습니다.
바로 내가 수정한 내역을 날려버리고, 원격 저장소의 코드를 내려받는 방법입니다.



Git pull

이 문제를 해결하는 여러 방법 중에서 가장 쉬운 방법을 소개하겠습니다.
바로 내가 수정한 내역을 날려버리고, 원격 저장소의 코드를 내려받는 방법입니다.



Git pull

먼저 수업 자료 코드들이 들어있는 디렉터리에서 `git status`를 체크합니다.
커밋하도록 정하지 않은 변경 사항들이 보이면 성공입니다.

```
> git status
현재 브랜치 main
브랜치가 'origin/main'에 맞게 업데이트된 상태입니다.

커밋하도록 정하지 않은 변경 사항 :
(무엇을 커밋할지 바꾸려면 "git add <파일>..."을 사용하십시오)
(use "git restore <file>..." to discard changes in working directory)
수정함 : week1\_python\_basics/ch1-1\_variables.ipynb
```

Git pull

먼저 수업 자료 코드들이 들어있는 디렉터리에서 `git add` 명령어로 수정한 1강과 2강 수업 자료들을 커밋할 예정이라는 `stage` 상태로 변경해주세요
초록색으로 파일명이 변경되면 됩니다.

```
> git add week1_python_basics/ch1-1_variables.ipynb
> git status
현재 브랜치 main
브랜치가 'origin/main'에 맞게 업데이트된 상태입니다.

커밋할 변경 사항 :
  (use "git restore --staged <file>..." to unstage)
    수정함 :      week1_python_basics/ch1-1_variables.ipynb
```

Git pull

이 상태에서 우리는 커밋을 하지 않고, 내가 수정한 내용들을 날려버릴겁니다.

`git stash` 명령어로 수정한 내용들을 임시 저장하세요.

커밋할 사항이 없음, 깨끗함이라는 상태를 확인하면 됩니다.

```
> git stash
Saved working directory and index state WIP on main: 51dfae3 Add python basic if
else, loop codes
> git status
현재 브랜치 main
브랜치가 'origin/main'에 맞게 업데이트된 상태입니다.

커밋할 사항 없음, 작업 폴더 깨끗함
```

Git pull

Jupyter lab을 켜서 코드의 상태를 확인해보세요

여러분들의 수정 내역들이 초기화됐을 겁니다.

만약 내가 작업한 내용들을 다시 꺼내오고 싶다면, `git stash pop`을 해주면 됩니다.

```
> git stash pop
현재 브랜치 main
브랜치가 'origin/main'에 맞게 업데이트된 상태입니다.

커밋하도록 정하지 않은 변경 사항 :
(무엇을 커밋할지 바꾸려면 "git add <파일>..."을 사용하십시오)
(use "git restore <file>..." to discard changes in working directory)
수정함 :      week1_python_basics/ch1-1_variables.ipynb
```

Git pull

우리는 내가 수정한 내역을 날려버리고 싶기 때문에
다시 git add와 git stash를 해준 뒤, git stash clear를 해주세요.

```
> git stash pop
현재 브랜치 main
브랜치가 'origin/main'에 맞게 업데이트된 상태입니다.

커밋하도록 정하지 않은 변경 사항 :
(무엇을 커밋할지 바꾸려면 "git add <파일>..."을 사용하십시오)
(use "git restore <file>..." to discard changes in working directory)
수정함 :      week1_python_basics/ch1-1_variables.ipynb
```

Git pull

그러면 내가 수정한 내역을 깔끔하게 지우고,
처음 git clone으로 원격 저장소 파일을 내려 받았던 상태가 됩니다.
이 상태에서 git pull을 수행하면 원격 저장소에 이후 푸쉬된 커밋들을 내려받습니다.