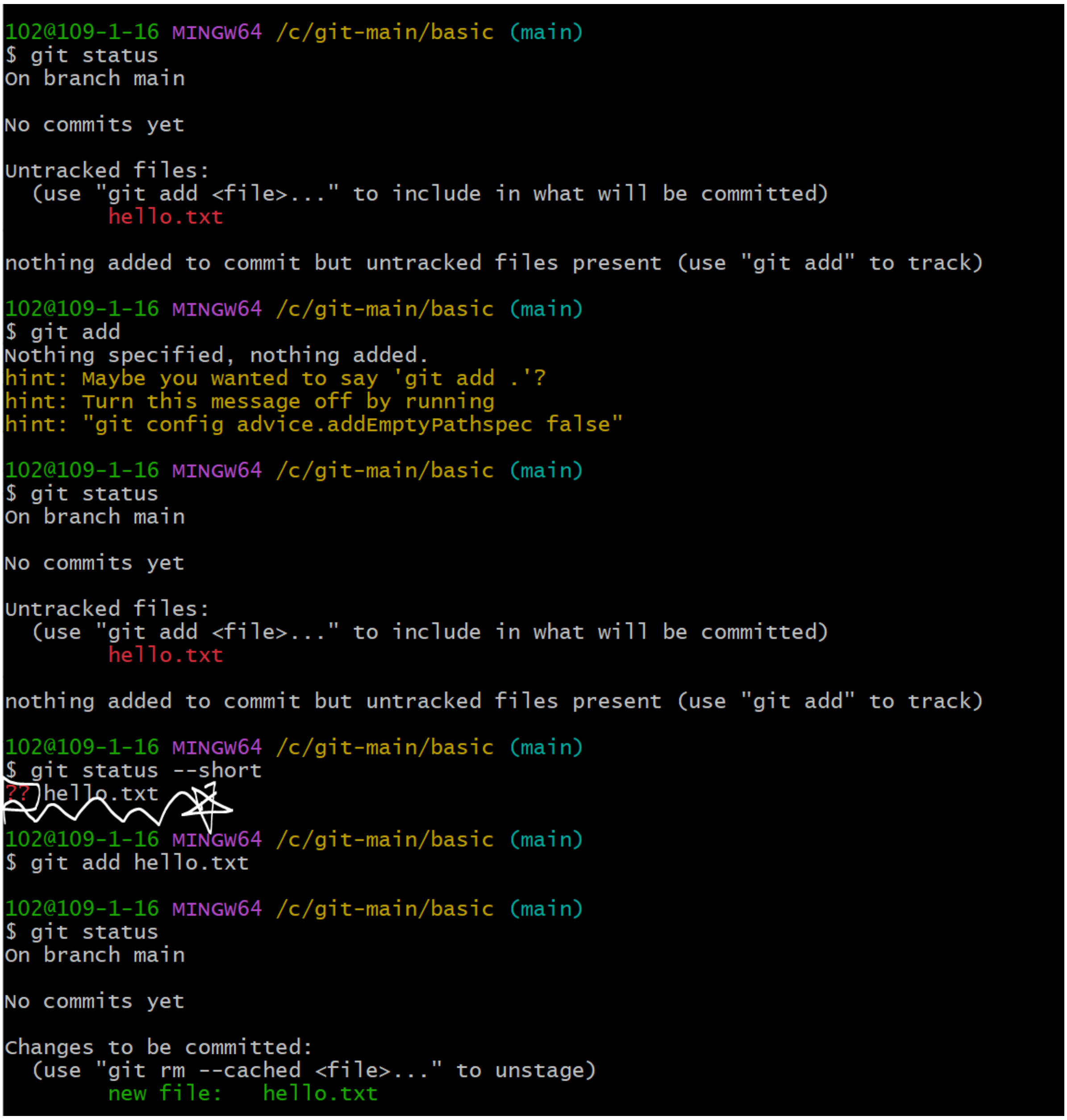
📍

중요

4주차 내용

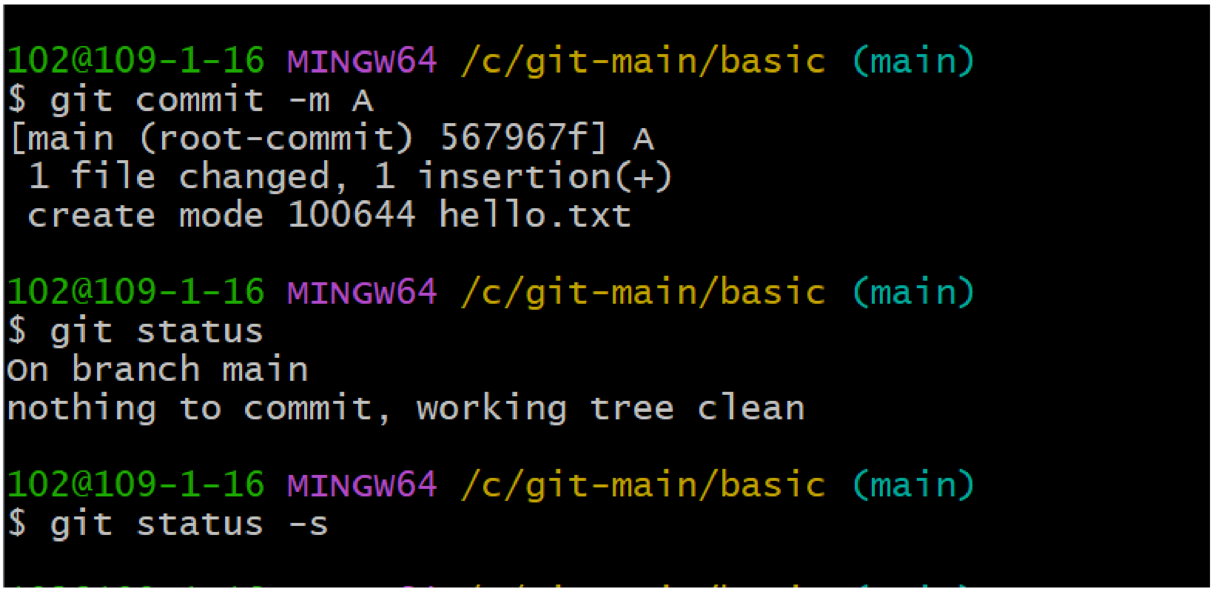
09. 3-3 [실습] 커밋과 로그 이력, 과거 여행.pdf



```
102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status
On branch main

No commits yet

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

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

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git add
Nothing specified, nothing added.
hint: Maybe you wanted to say 'git add .'?
hint: Turn this message off by running
hint: "git config advice.addEmptyPathspec false"

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status
On branch main

No commits yet

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

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

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status --short
?? hello.txt

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git add hello.txt

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status
On branch main

No commits yet

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

git status —short

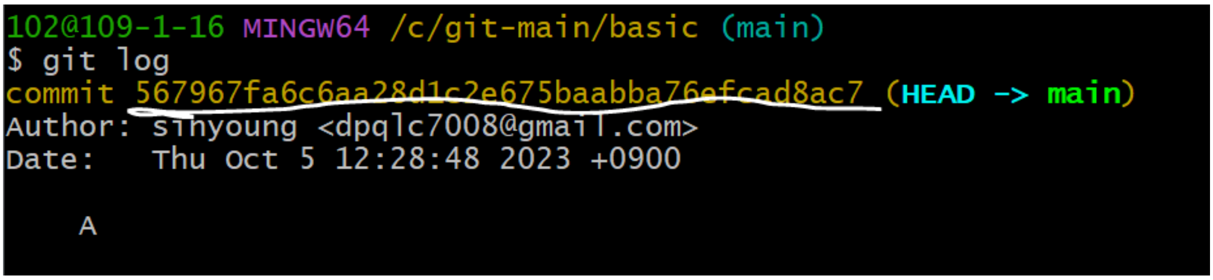
아직 add를 안 했을 때 git status —short 를 하였을 때 untracked file표현을 ??hello.txt(파일명)으로 나오는 것을 시험에 낸데



```
102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git commit -m A
[main (root-commit) 567967f] A
 1 file changed, 1 insertion(+)
 create mode 100644 hello.txt

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status
On branch main
nothing to commit, working tree clean

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status -s
```

commit

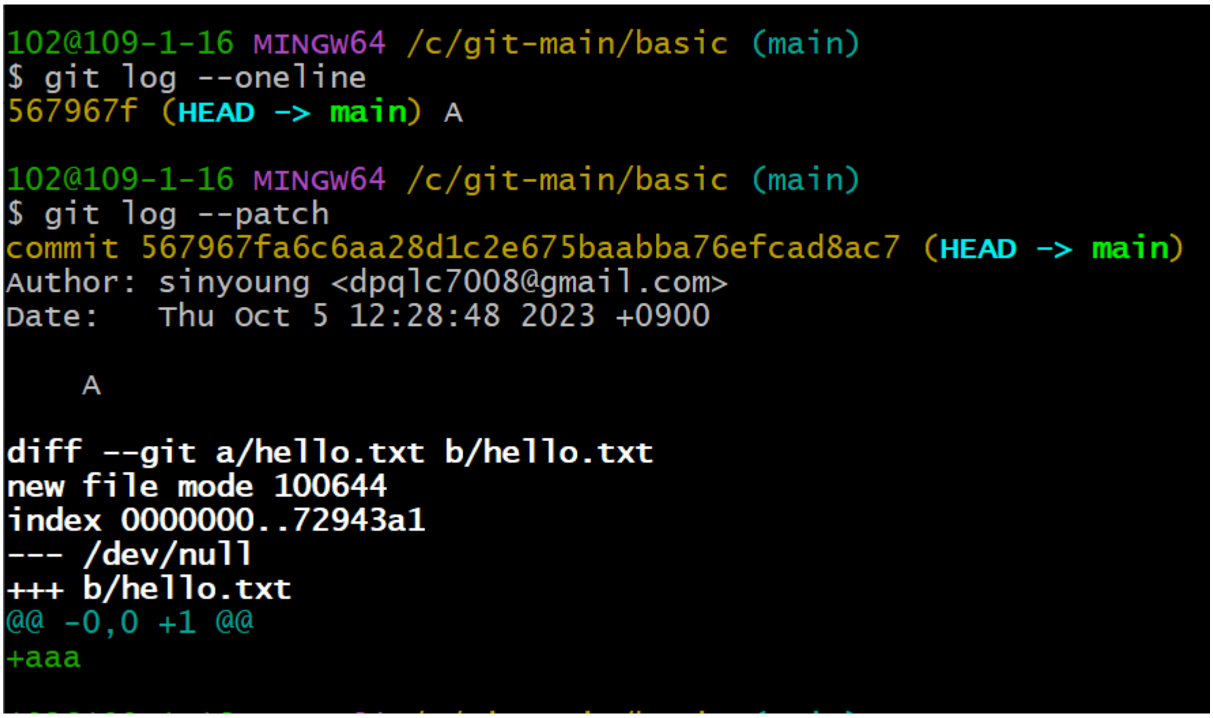
commit까지 하고 git status를 하면 아무것도 안나오는 clean이 나온다. —short를 하면 아예 아무것도 안나온다.



```
102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git log
commit 567967fa6c6aa28d1c2e675baabba76efcad8ac7 (HEAD -> main)
Author: sihyoung <dpqlc7008@gmail.com>
Date:   Thu Oct 5 12:28:48 2023 +0900

    A
```

git log

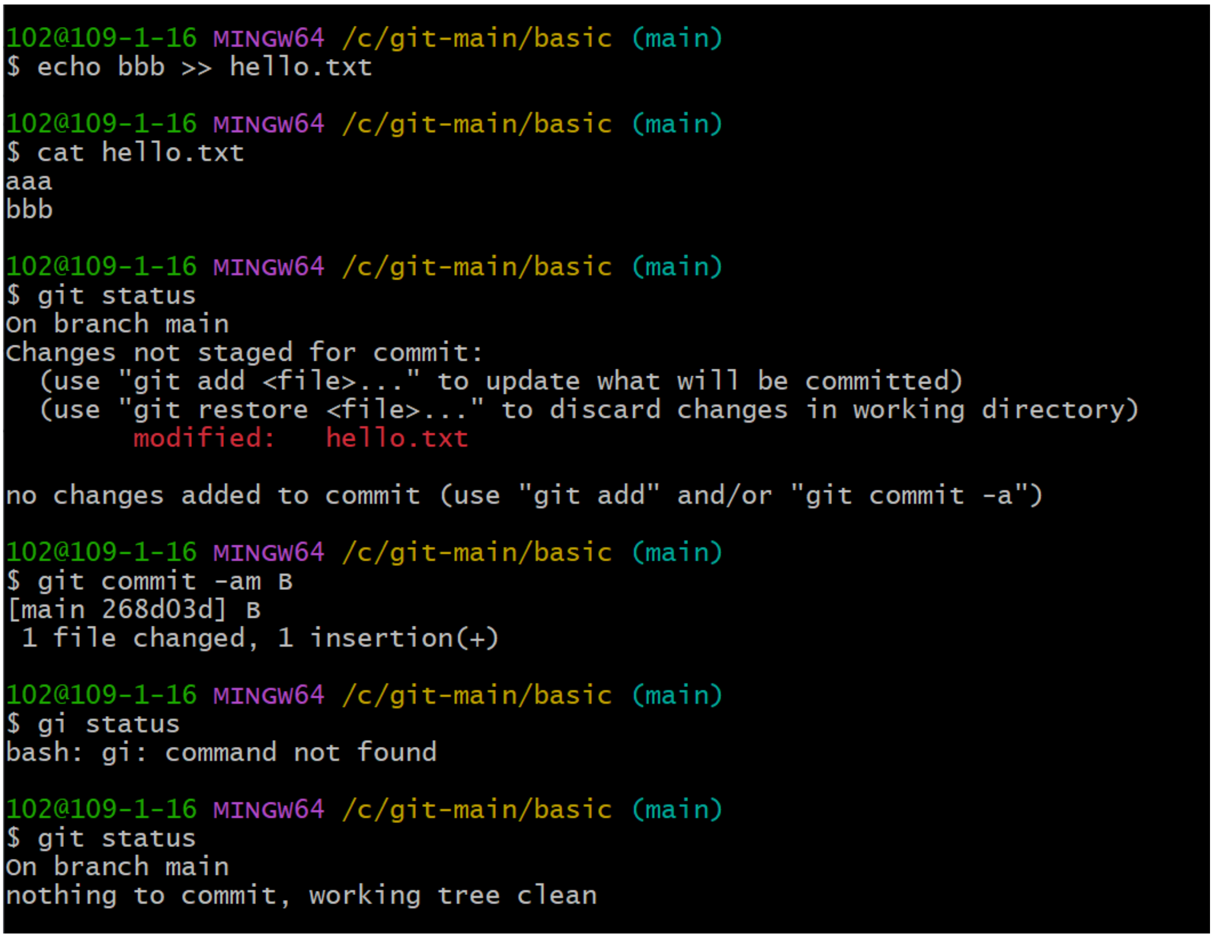
git log 를 치면 commit ID가 나온다..



```
102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git log --oneline
567967f (HEAD -> main) A

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git log --patch
commit 567967fa6c6aa28d1c2e675baabba76efcad8ac7 (HEAD -> main)
Author: sinyoung <dpqlc7008@gmail.com>
Date:   Thu Oct 5 12:28:48 2023 +0900

    A

diff --git a/hello.txt b/hello.txt
new file mode 100644
index 0000000..72943a1
--- /dev/null
+++ b/hello.txt
@@ -0,0 +1 @@
+aaa
```

git log —oneline && git log —patch

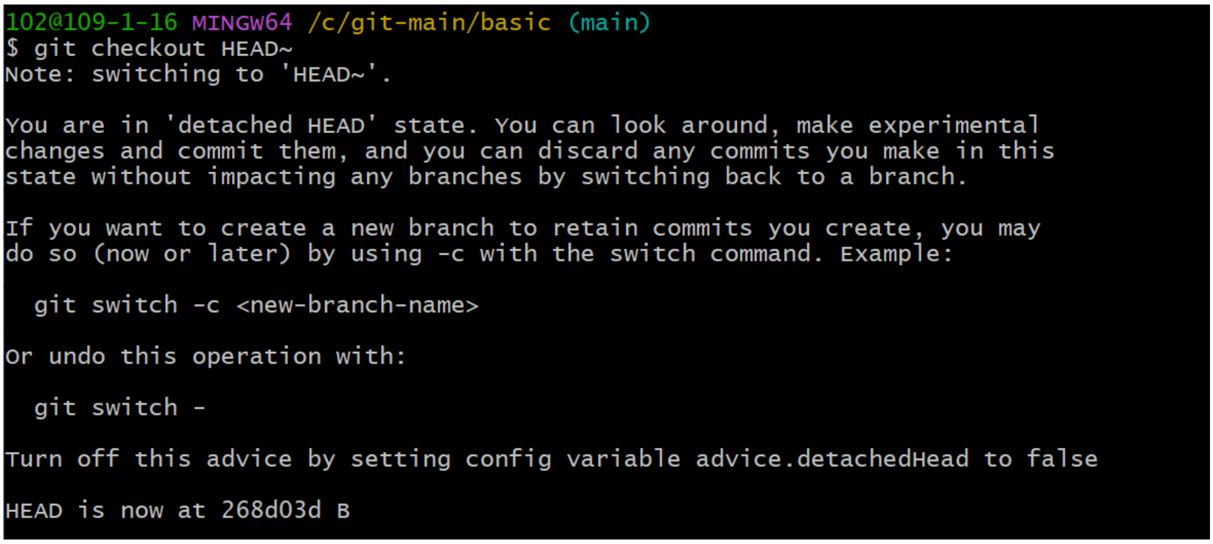
git log 의 —oneline은 간략히 나오며 commit ID는 7자리로 나오고 —patch는 지금까지 commit한 history tree(?)가 나온다/



```
102@109-1-16 MINGW64 /c/git-main/basic (main)
$ echo bbb >> hello.txt

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ cat hello.txt
aaa
bbb

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ 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:   hello.txt

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

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git commit -am B
[main 268d03d] B
 1 file changed, 1 insertion(+)

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ gi status
bash: gi: command not found

102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git status
On branch main
nothing to commit, working tree clean
```

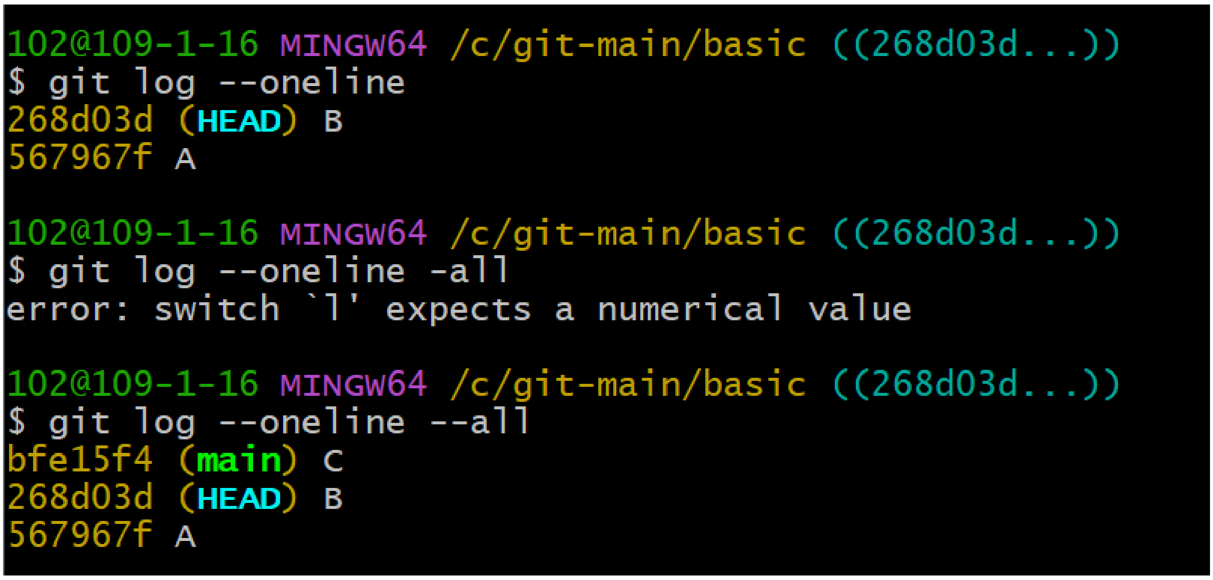
>> vs >   && git commit -am B\

>> 를 두개를 하면 원래 있는 파일에 내용을 추가하고 > 를 하면 원래 있던 걸 삭제 후 아예 새로 추가한 내용만 존재했으면 할 때 사용한다. 또한 git add를 하지 않고 git commit -am 을 하면 추가와 커밋을 동시에 할 수 있다.



```
102@109-1-16 MINGW64 /c/git-main/basic (main)
$ git checkout HEAD~
Note: switching to 'HEAD~'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 268d03d B
```

checkout을 하고 원하는 점으로 지정을하면(HEAD~) 거기로 간다. 밑에 로그를 확인하면 알 수 있다.



```
102@109-1-16 MINGW64 /c/git-main/basic ((268d03d...))
$ git log --oneline
268d03d (HEAD) B
567967f A

102@109-1-16 MINGW64 /c/git-main/basic ((268d03d...))
$ git log --oneline -all
error: switch `l' expects a numerical value

102@109-1-16 MINGW64 /c/git-main/basic ((268d03d...))
$ git log --oneline --all
bfe15f4 (main) C
268d03d (HEAD) B
567967f A
```

과제

01주 1. 2023 OSS 교과목 및 학과 소개.pdf

문제

1번 0x 3개

2번 괄호 채우기 문제 3개

3번 객관식 문항 6~8개

4~5번 단답식 주관형 문제 ex) 이런걸 쳤을 때 어떤 답이 나올까? 같은거 or 이런 답이 나올 때 어떤 걸 쳐야할까?