

Open source Software

오픈소스 소프트웨어

10. 파일 비교 diff





학습 개요

1. 깃 3 영역의 파일을 비교
 - 작업 디렉토리, 스테이징 영역, 깃 저장소
2. 커밋 간의 파일 비교
 - HEAD
 - ➡ 최신 커밋을 가리키는 포인터



학습 목표

1. 파일 비교 표시를 이해할 수 있다.
2. 깃 3 영역의 파일을 비교할 수 있다.
 - 스테이징 영역 기준으로 작업 디렉토리 파일 비교
 - 깃 저장소 기준으로 스테이징 영역 파일 비교
 - 깃 저장소 기준으로 작업 디렉토리 파일 비교
3. 커밋 간의 파일을 비교할 수 있다.

LESSON 01

3 영역 파일 비교 diff



1 3 영역 파일 비교 diff

파일 비교

깃 3 영역의 파일 비교

커밋 간의 파일 비교

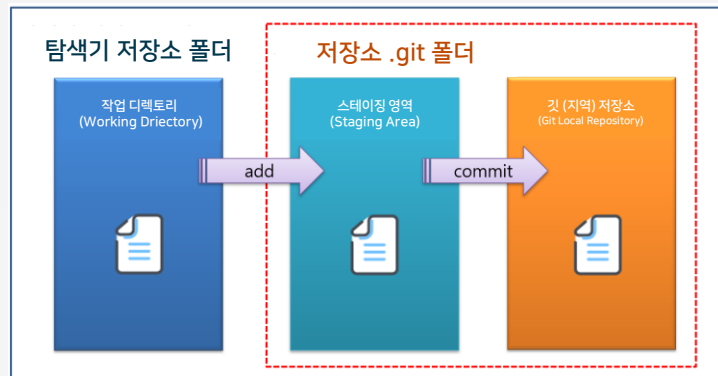
- HEAD
- HEAD~ HEAD^
 - 하나 전 커밋
 - ~ ➡ tilde, 틸드, 물결
 - ^ ➡ caret, 커렛, 삿갓(모자)
- HEAD~~ HEAD~2 HEAD^^ HEAD^~
 - 두개 이전 커밋

브랜치 간의 파일 비교

1 3 영역 파일 비교 diff

기어 아이콘 깃 3 영역

- ✓ PC의 탐색기 저장소 폴더에서 보이는 파일
 - 작업 디렉토리의 파일
- ✓ 저장소 하부의 .git 폴더에 정보로 숨겨진 파일
 - 스테이징 영역
 - 깃 (지역) 저장소
- ✓ 리눅스 명령어 ls로 보이는 파일
 - 작업 디렉토리의 파일
- ✓ 깃 명령 git ls-files로 보이는 파일
 - 스테이징 영역의 파일

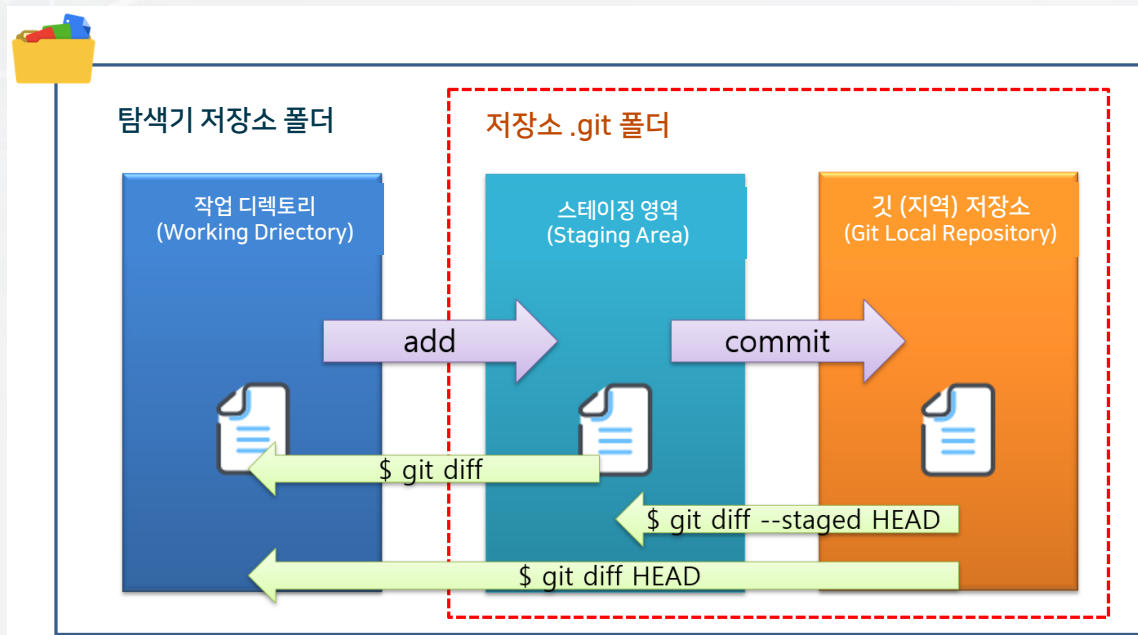


1 3 영역 파일 비교 diff

깃 3 영역의 파일 비교

✓ 기준에서 파일 비교

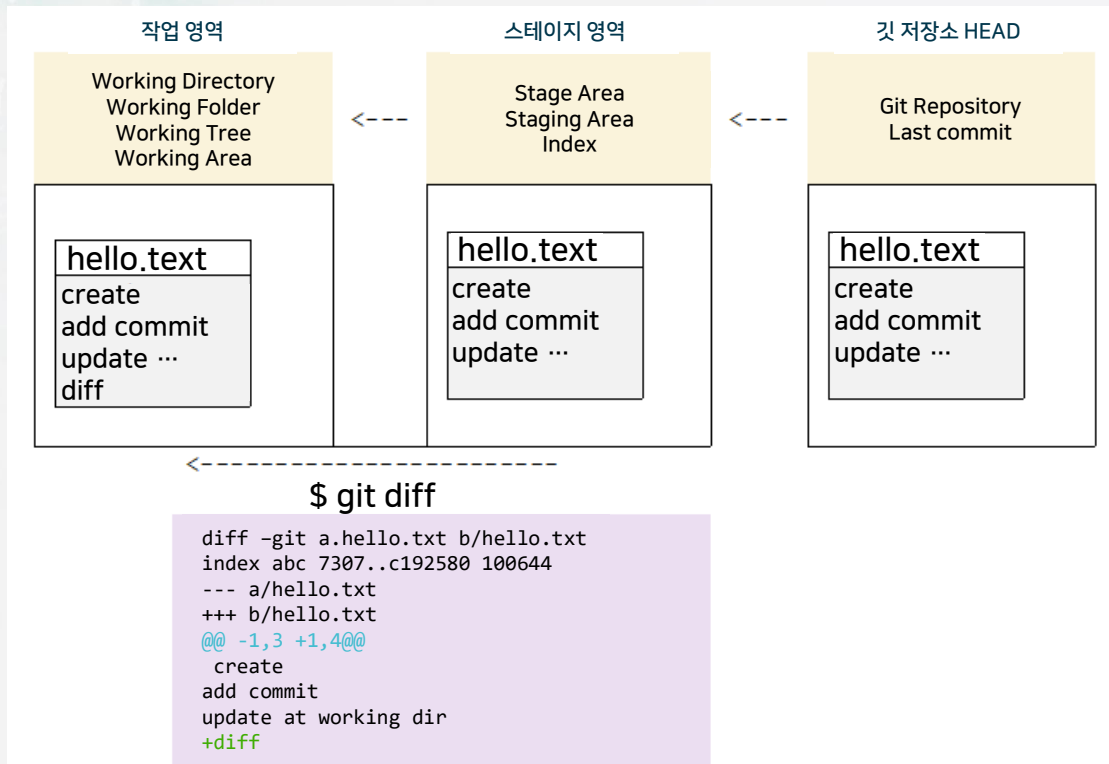
- 다음 그림에서 두 영역을 비교한다면 오른쪽 영역이 기준



1 3 영역 파일 비교 diff

스태이징 영역 기준에서 작업 디렉토리 비교 1/2

✓ \$ git diff





1 3 영역파일 비교 diff

스테이징 영역 기준에서 작업 디렉토리 비교 2/2

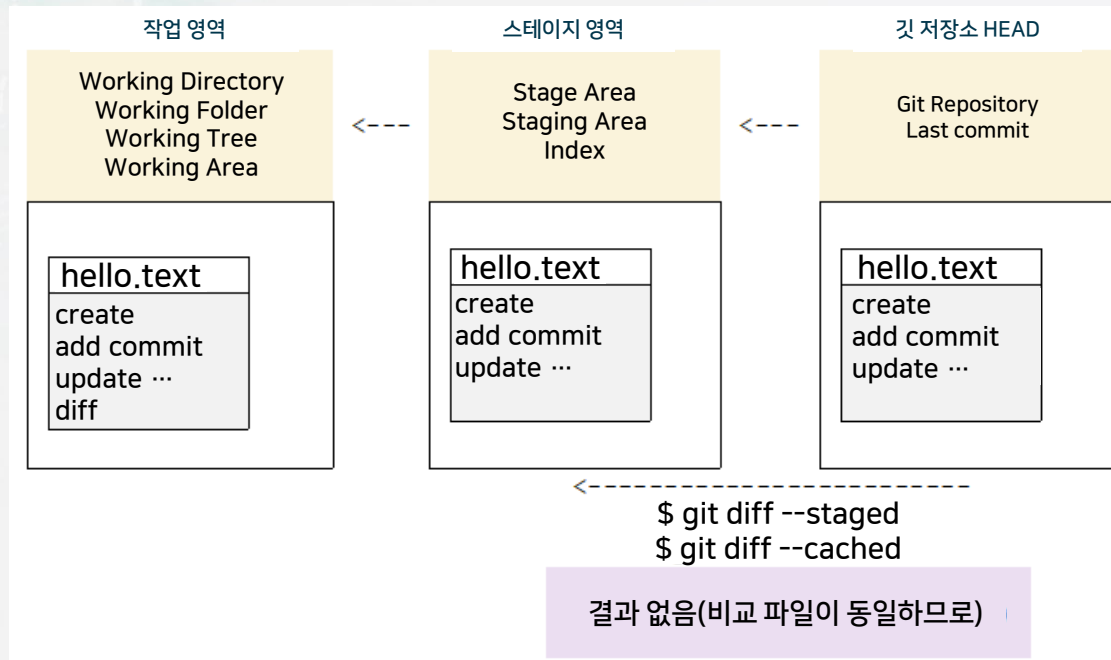
\$ git diff

<div><-----</div>			
비교 위치	명령		기준 위치
Working Directory	\$ git diff		Stage Area Index
<div>hello.txt create add commit update ... diff</div>	결과 표시	해석	<div>hello.txt create add commit update ...</div>
	<div>diff --git a/hello.txt b/hello.txt index abc7307..c192580 100644 --- a/hello.txt +++ b/hello.txt @@ -1,3 +1,4 @@ create add commit update at working dir +diff</div>	<div>비교 기준 파일(스테이지 영역): a/hello.txt 비교 대상 파일(작업 폴더): b/hello.txt a/hello.txt를 ---로 표기 b/hello.txt를 +++로 표기 -1,3: 기준 파일은 첫 번째 줄에서 3개의 줄 비교 +1,4: 대상 파일은 첫 번째 줄에서 4개의 줄 비교 create add commit update at working dir : 양쪽에서 공통인 내용 +diff : 대상 파일은 네 번째 줄에 diff 줄이 있음</div>	

1 3 영역 파일 비교 diff

깃 저장소 HEAD 기준으로 스테이징 영역 비교 1/2

✓ \$ git diff --staged [HEAD]



1 3 영역 파일 비교 diff

깃 저장소 HEAD 기준으로 스테이징 영역 비교 2/2

✓ \$ git diff --staged [HEAD]

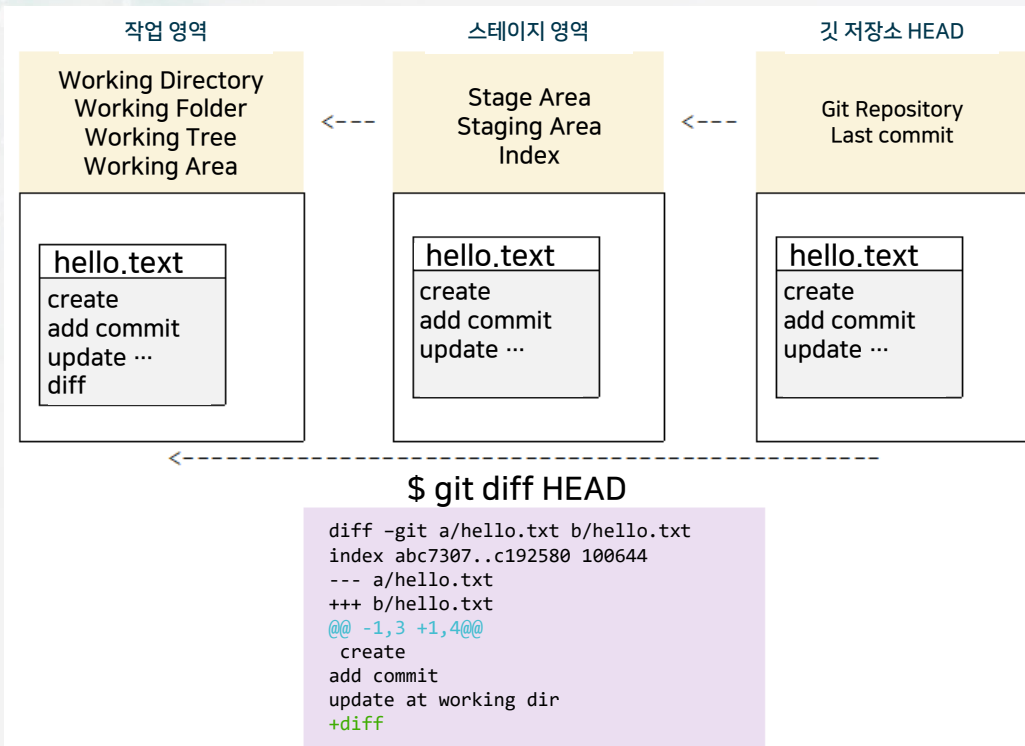
←-----		
비교 위치	명령	기준 위치
Stage Area Index	\$ git diff [--staged --cached]	깃 저장소 마지막 커밋
	결과 표시	해석
<div>hello.txt</div> <div>create</div> <div>add commit</div> <div>update ...</div> <div>diff</div>	<pre>diff --git a/hello.txt b/hello.txt index abc7307..c192580 100644 --- a/hello.txt +++ b/hello.txt @@ -1,3 +1,4 @@ create add commit update at working dir +diff</pre>	<div>hello.txt</div> <div>create</div> <div>add commit</div> <div>update ...</div>

비교 기준 파일(깃 저장소 마지막 커밋): a/hello.txt
비교 대상 파일(스테이징 영역): b/hello.txt
a/hello.txt를 ---로 표기
b/hello.txt를 +++로 표기
-1,3: 기준 파일은 첫 번째 줄에서 3개의 줄 비교
+1,4: 대상 파일은 첫 번째 줄에서 4개의 줄 비교
create
add commit
update at working dir
: 양쪽에서 공통인 내용
+diff: 대상 파일은 네 번째 줄에 diff 줄이 있음

1 3 영역 파일 비교 diff

깃 저장소 HEAD 기준으로 작업 디렉토리 비교 1/2

✓ \$ git diff HEAD



1 3 영역 파일 비교 diff

깃 저장소 HEAD 기준으로 작업 디렉토리 비교 2/2

\$ git diff HEAD

←-----			
비교 위치	명령		기준 위치
Working Directory	\$ git diff HEAD		깃 저장소 마지막 커밋
<div>hello.txt</div> <div>create add commit update ... diff</div>	결과 표시	해석	<div>hello.txt</div> <div>create add commit update ...</div>
	diff --git a/hello.txt b/hello.txt index abc7307..c192580 100644 --- a/hello.txt +++ b/hello.txt @@ -1,3 +1,4 @@ create add commit update at working dir +diff	비교 기준 파일(깃 저장소 마지막 커밋): a/hello.txt 비교 대상 파일(작업 디렉토리): b/hello.txt a/hello.txt를 ---로 표기 b/hello.txt를 +++로 표기 -1,3: 기준 파일은 첫 번째 줄에서 3개의 줄 비교 +1,4: 대상 파일은 첫 번째 줄에서 4개의 줄 비교 create add commit update at working dir : 양쪽에서 공통인 내용 +diff: 대상 파일은 네 번째 줄에 diff 줄이 있음	



1 3 영역파일 비교 diff

기어 깃 저장소 HEAD~2 기준으로 작업 디렉토리 비교

✓ \$ git diff HEAD~2

<div>←-----</div>			
비교 위치		명령	기준 위치
Working Directory		\$ git diff HEAD~2	깃 저장소 HEAD~2 (마지막 전전 커밋)
		결과 표시	해석
<div>hello.txt</div> <div>create</div> <div>add commit</div> <div>update ...</div> <div>diff</div>	<div>diff --git a/hello.txt b/hello.txt</div> <div>index abc7307..c192580</div> <div>100644</div> <div>--- a/hello.txt</div> <div>+++ b/hello.txt</div> <div>@@@ -1,3 +1,4 @@@</div> <div>create</div> <div>+add commit</div> <div>+update at working dir</div> <div>+diff</div>	<div>비교 기준 파일(깃 저장소 마지막 커밋): a/hello.txt</div> <div>비교 대상 파일(작업 디렉토리): b/hello.txt</div> <div>a/hello.txt를 ---로 표기</div> <div>b/hello.txt를 +++로 표기</div> <div><div>-1,3: 기준 파일은 첫 번째 줄에서 3개의 줄 비교</div><div>+1,4: 대상 파일은 첫 번째 줄에서 4개의 줄 비교</div></div> <div>create : 양쪽에서 공통인 내용</div> <div>+add commit</div> <div>+update at working dir</div> <div>+diff : 대상 파일은 3개 줄이 추가됨</div>	<div>hello.txt</div> <div>create</div>

LESSON 02

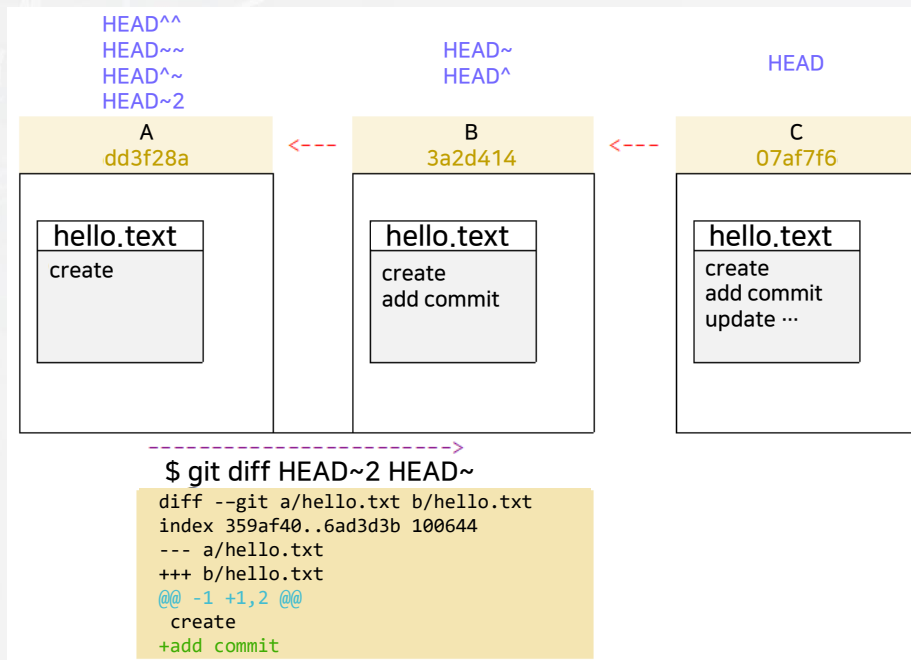
두 버전과의 파일 비교 diff



2 두 버전과의 파일 비교 diff

⚙️ \$ git diff c1D1 c1D2

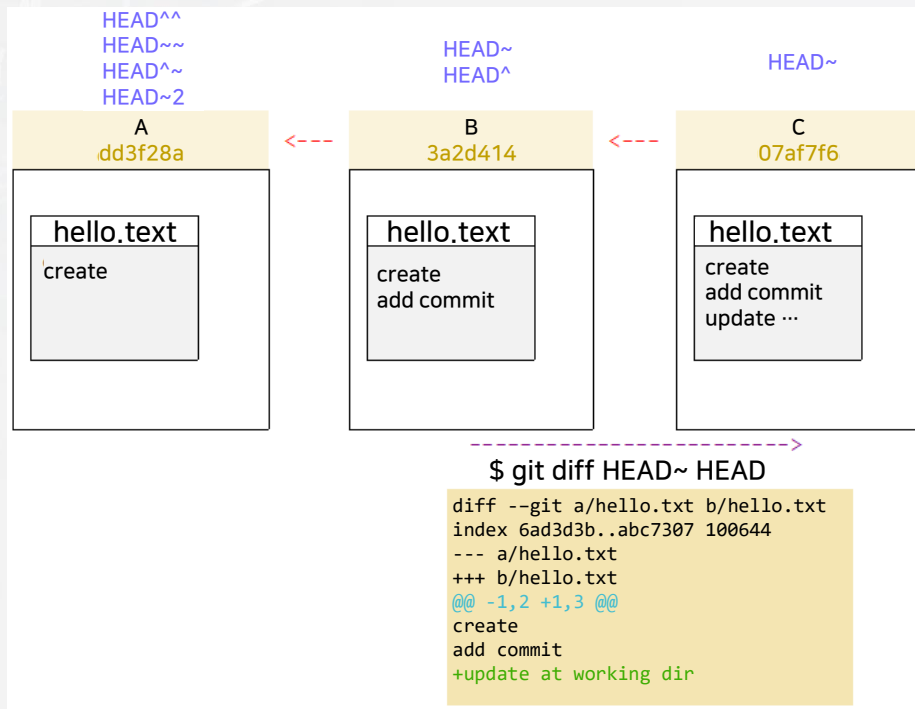
✅ \$ git diff HEAD~2 HEAD~



2 두 버전과의 파일 비교 diff

⚙️ \$ git diff c101 c102

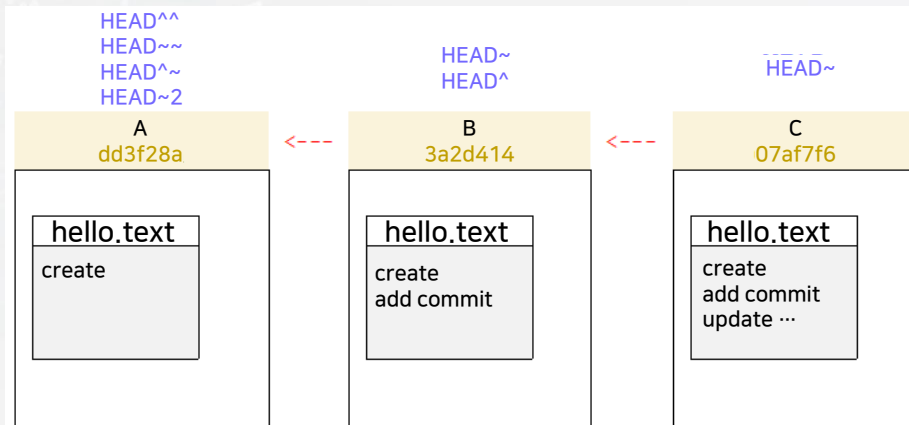
✅ \$ git diff HEAD~ HEAD



2 두 버전과의 파일 비교 diff

⚙️ \$ git diff c101 c102

✅ \$ git diff HEAD~2 HEAD



\$ git diff HEAD~2 HEAD
\$ git diff dd3f28a 07af7f6

```
diff --git a/hello.txt b/hello.txt
index 395af40..abc7307 100644
--- a/hello.txt
+++ b/hello.txt
@@ -1,3 @@
create
+add commit
+update at working dir
```

Summary

» 스테이징 영역 기준으로 작업 디렉토리 파일 비교

- ◆ \$ git diff

» 깃 저장소 기준으로 스테이징 영역 파일 비교

- ◆ \$ git diff --staged HEAD
- ◆ \$ git diff --staged HEAD~

» 깃 저장소 기준으로 작업 디렉토리 파일 비교

- ◆ \$ git diff HEAD
- ◆ \$ git diff HEAD~

» 커밋 간의 파일 비교

- ◆ git diff [비교할commit해쉬1] [비교할commit해쉬2]
 - ➡ git diff 048171 0c747d
 - ➡ git diff HEAD^ HEAD