

Open source Software

오픈소스 소프트웨어

27. 비주얼스튜디오 코드의 깃 활용



학습 개요

1. 비주얼스튜디오코드 vs code
2. 폴더 생성, 깃 저장소 생성
3. Changes, Staged Changes
4. Git Graph



학습 목표

1. 비주얼스튜디오코드에서 깃 저장소를 생성해 파일을 만들 수 있다.
2. 비주얼스튜디오코드에서 파일을 커밋할 수 있다.
3. 비주얼스튜디오코드에서 로그 이력을 살펴볼 수 있다.
4. 비주얼스튜디오코드에서 파일 비교 diff와 동일한 기능을 수행할 수 있다.



LESSON 01

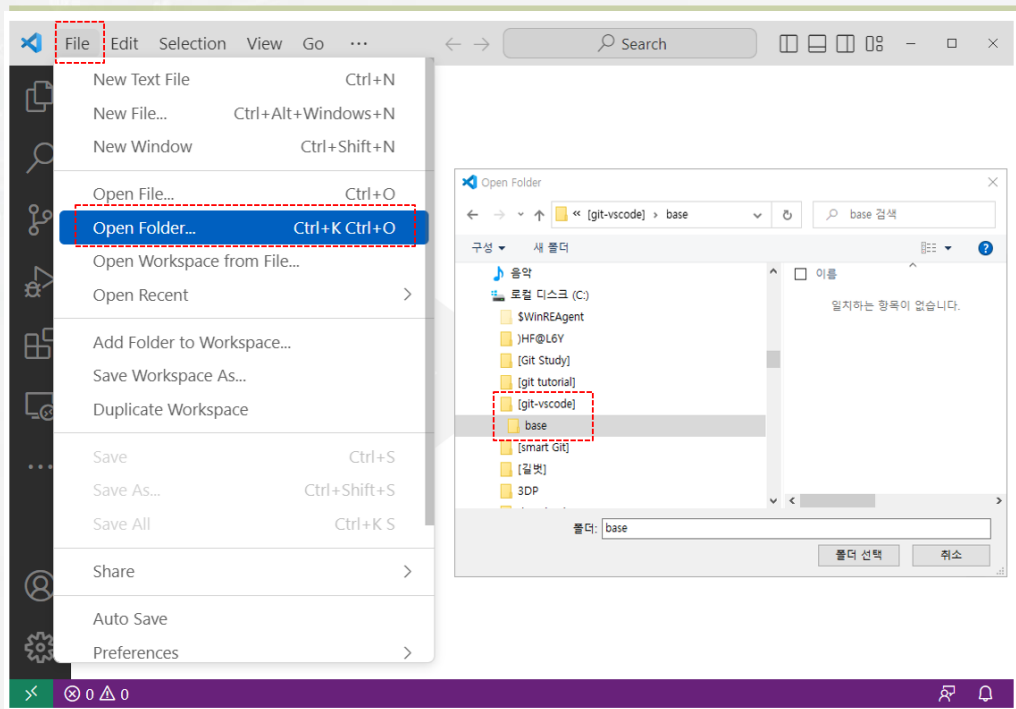
Vscode에서 깃 활용 기초



1 Vscod에서 깃 활용 기초

VSCODE 기초 1/3

폴더 열기

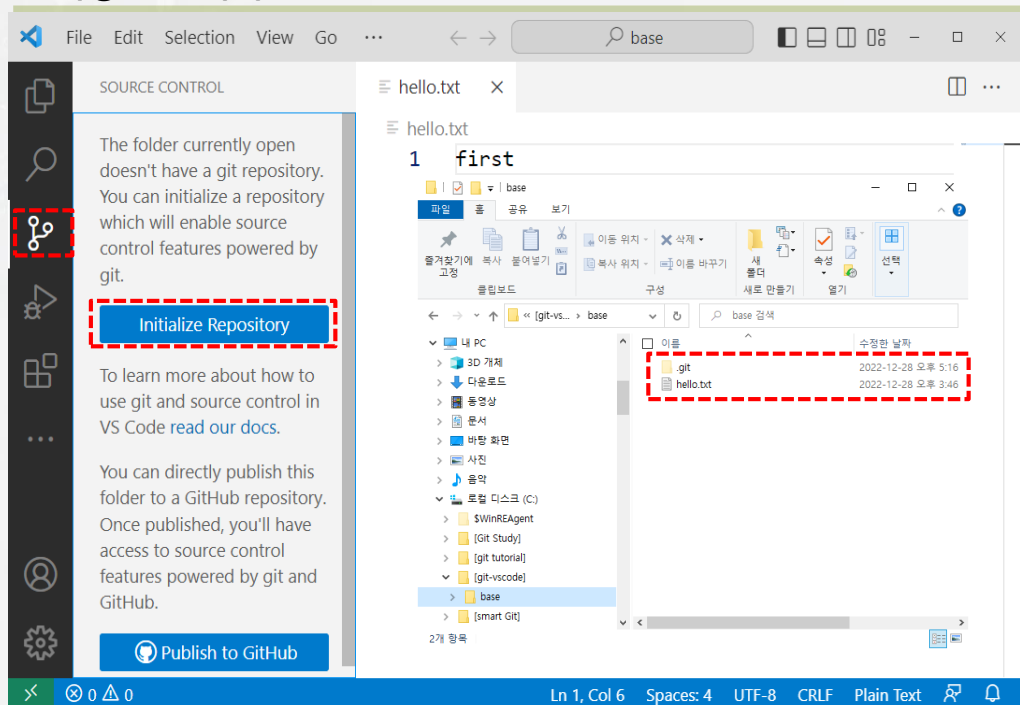


1 Vscod에서 깃 활용 기초

VS CODE 기초 2/3

소스 제어 활동바(activity bar)

저장소 초기화

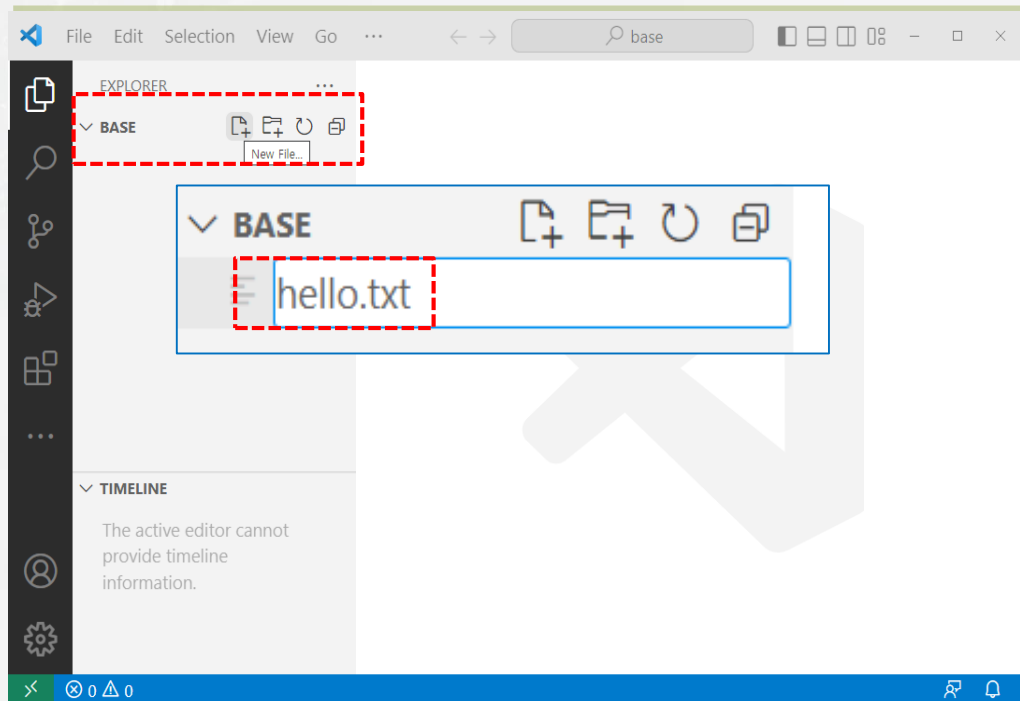


1 Vscod에서 깃 활용 기초

VS CODE 기초 3/3

✓ 파일 생성

• 메뉴 아이콘 New File

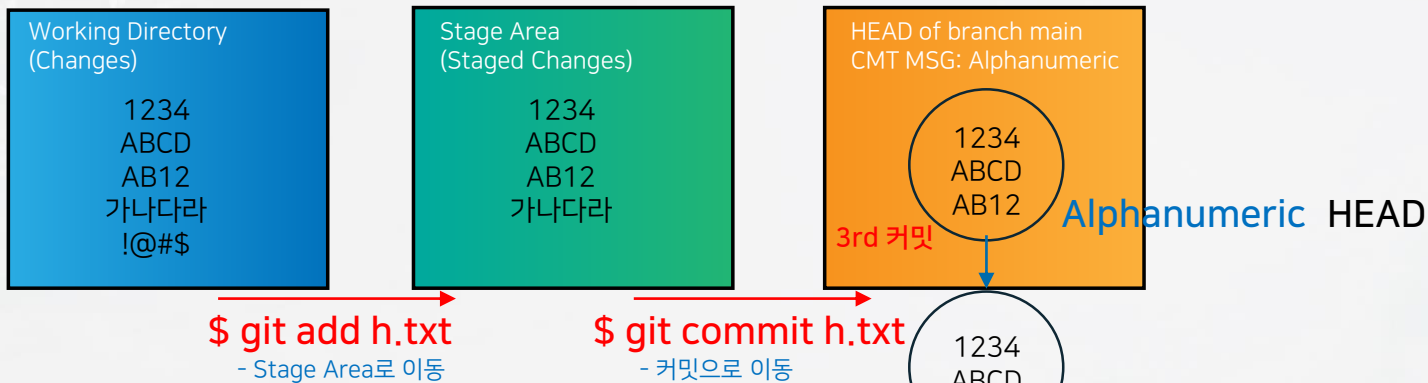


1 Vscode에서 깃 활용 기초

다음까지 add, commit

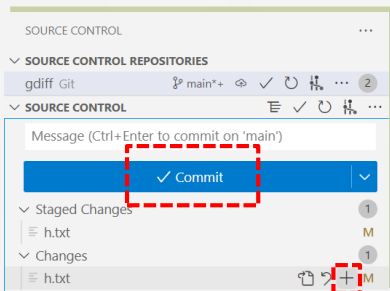
✓ 항상 add와 commit을 함께 가능

• 구분해서도 가능



✓ Vscode 메뉴

- + Commit

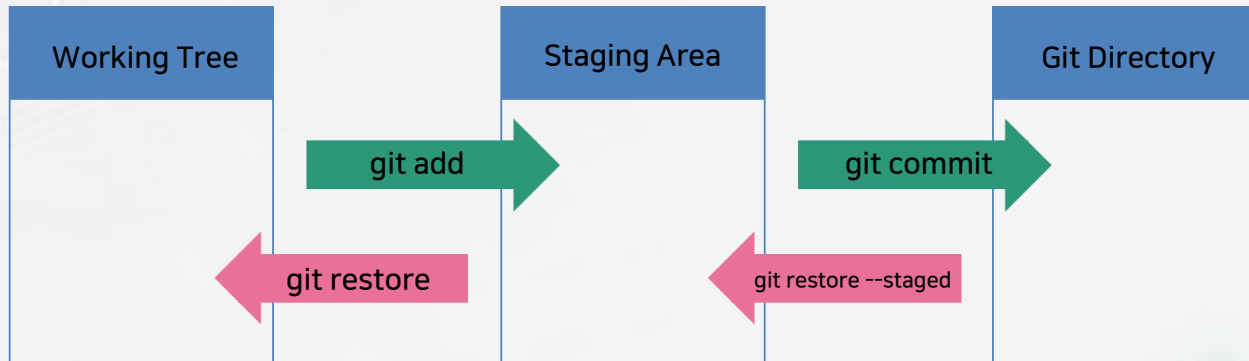


1 Vscode에서 깃 활용 기초

수정 작업 취소

✓ \$ git restore

✓ \$ git restore --staged



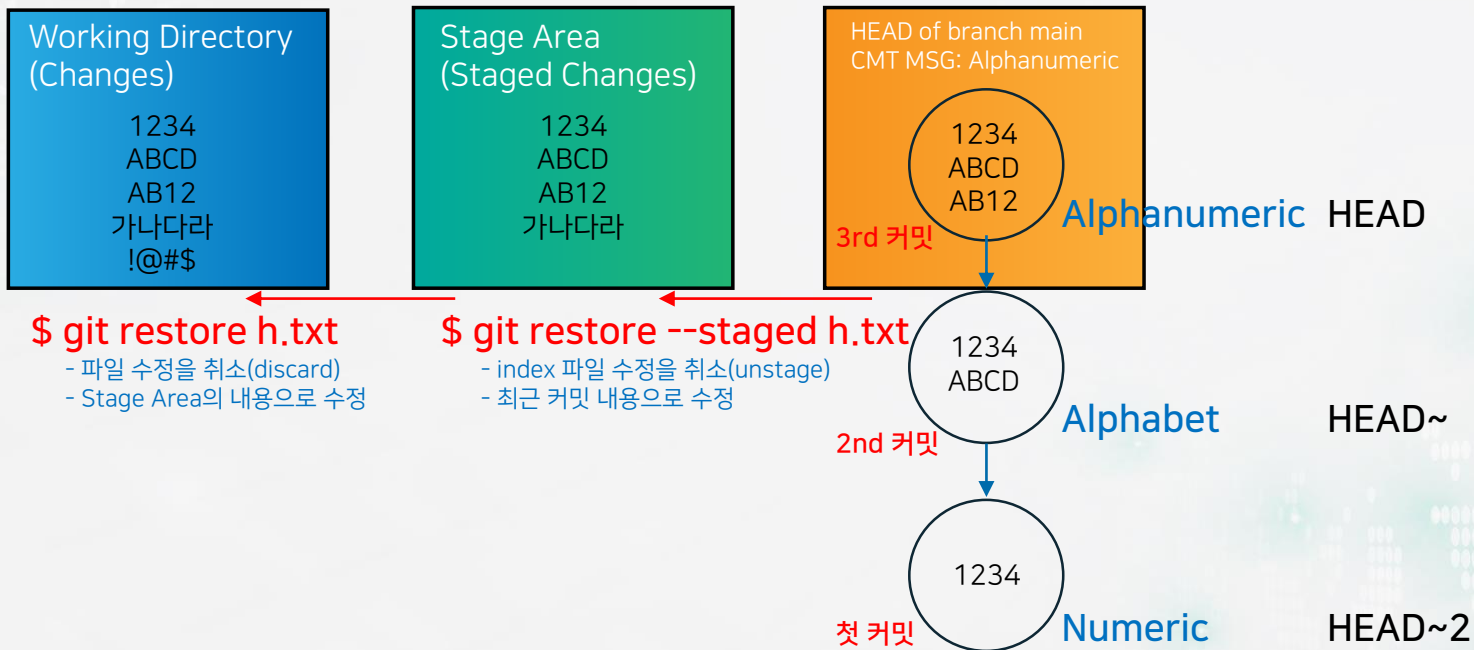
<출처> : <https://bufferings.hatenablog.com/entry/2020/05/01/013451>

1 Vscod에서 깃 활용 기초

작업 영역과 스테이지 영역 파일 수정 취소

명령어 git restore

- 옵션 --staged : index 파일 unstage(index를 최근 커밋으로 수정[복사])

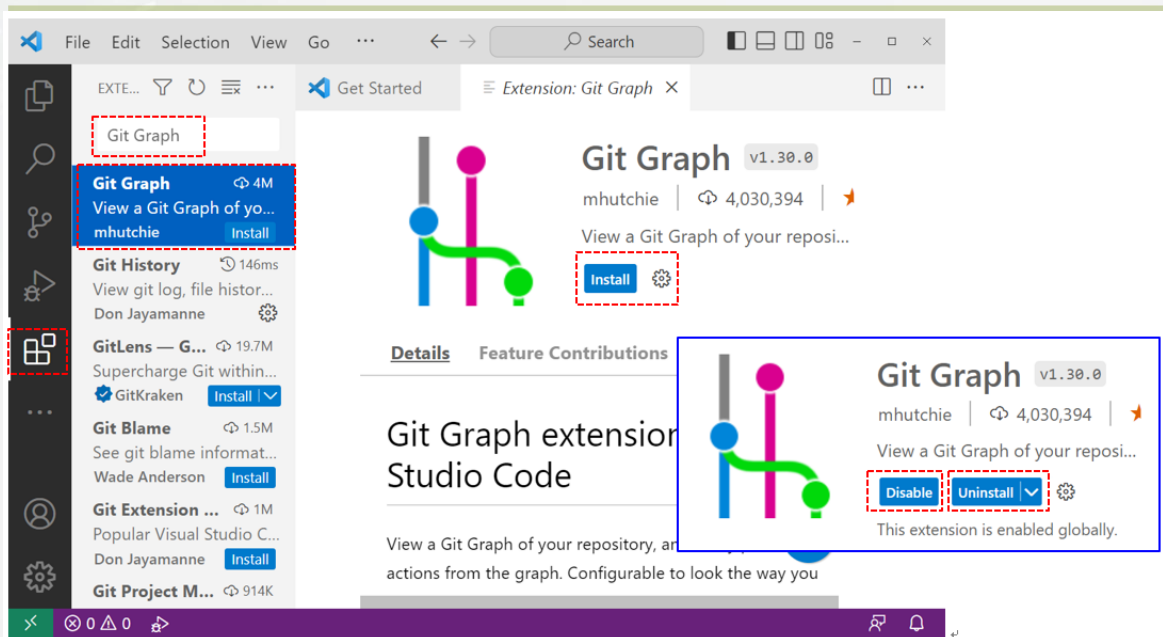


1 Vscode에서 깃 활용 기초

확장 Git Graph 설치

활동바의 Extensions(단축키 Ctrl+Shift+X)

- 사이드바에서 Git Graph로 검색
 - 검색된 확장에서 install로 설치

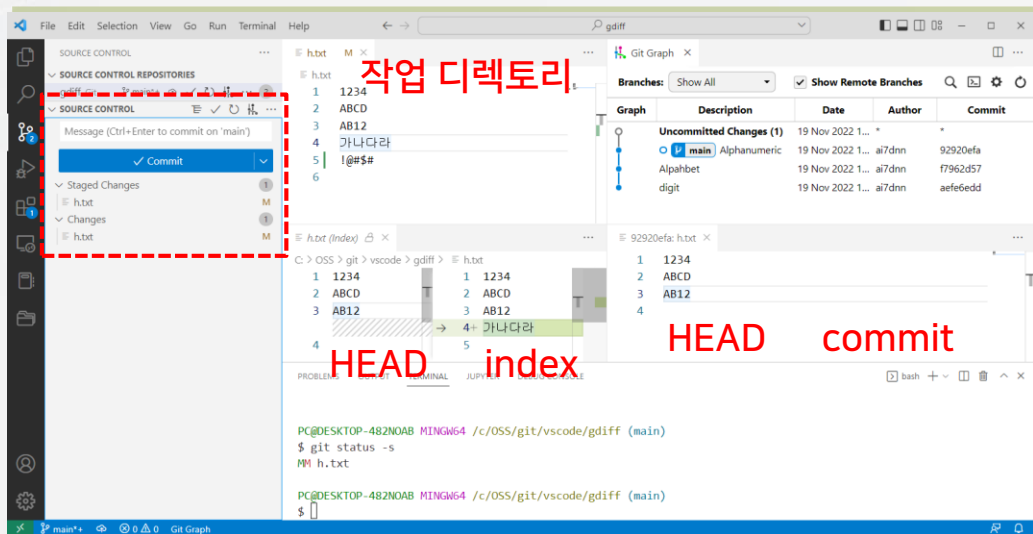


1 Vscode에서 깃 활용 기초

현재 상태

Vscode로 파악

- Staged Changes
 - 스테이지 영역
- Changes
 - 작업 디렉토리



1 Vscode에서 깃 활용 기초

⚙️ 작업 영역 수정 취소

☑️ 현재의 index 파일로 수정

• \$ git restore h.txt

```
PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
```

```
$ git status
```

```
On branch main
```

```
Changes to be committed:
```

```
(use "git restore --staged <file>..." to unstage)
```

```
modified: h.txt
```

```
Changes not staged for commit:
```

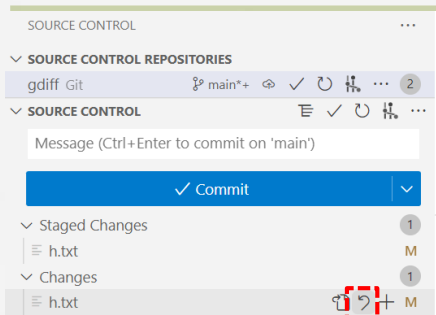
```
(use "git add <file>..." to update what will be committed)
```

```
(use "git restore <file>..." to discard changes in working directory)
```

```
modified: h.txt
```

☑️ 메뉴

• Discard Changes



1 Vscod에서 깃 활용 기초

스테이지 영역 수정 취소

현재의 최근 커밋 파일로 수정, unstage

- \$ git restore --staged h.txt

```
PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
```

```
$ git status
```

```
On branch main
```

```
Changes to be committed:
```

```
(use "git restore --staged <file>..." to unstage)
```

```
modified: h.txt
```

```
Changes not staged for commit:
```

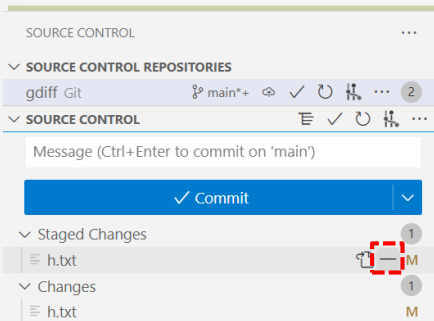
```
(use "git add <file>..." to update what will be committed)
```

```
(use "git restore <file>..." to discard changes in working directory)
```

```
modified: h.txt
```

메뉴

- Unstage Changes



LESSON 02

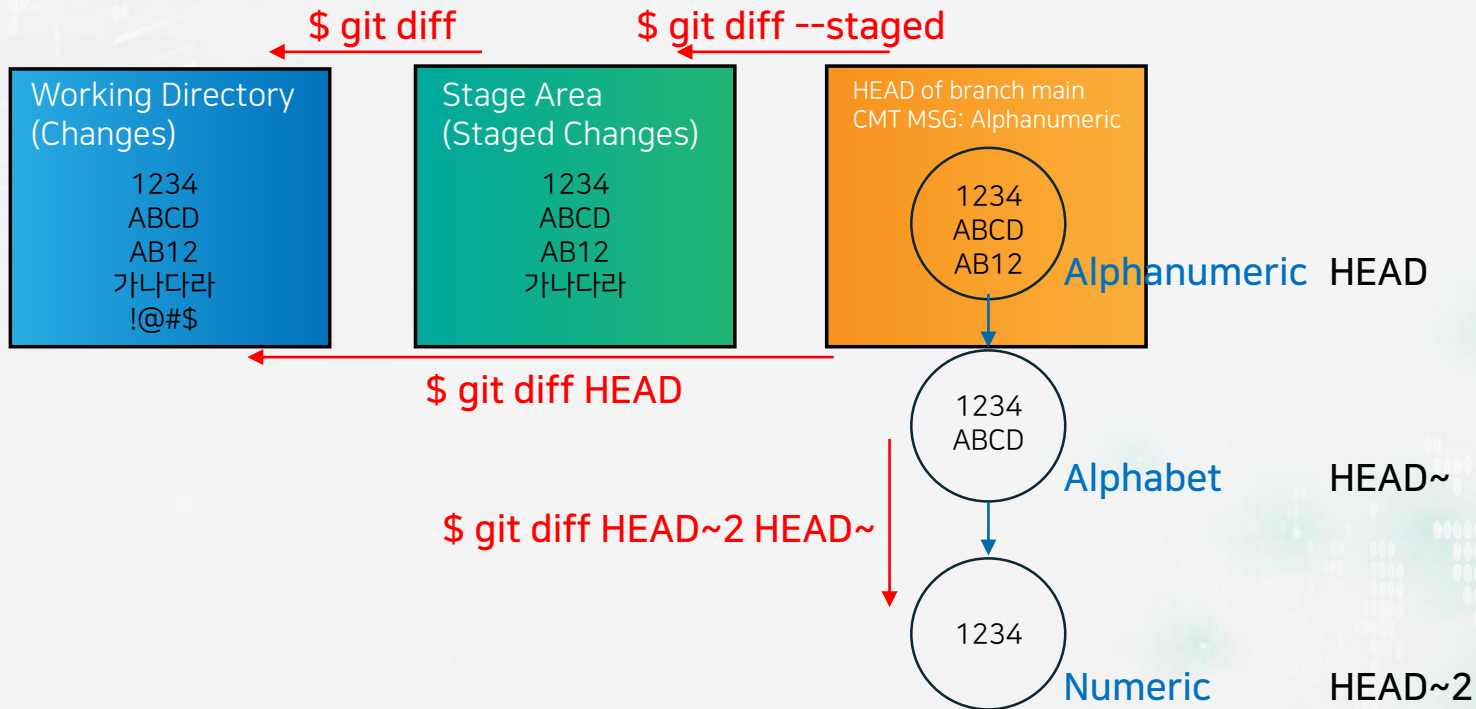
Vscode에서 파일 비교 git diff



2 Vscod에서 파일 비교 git diff

현재 상태에서의 다양한 파일 비교

현재의 작업 영역과 스테이지 영역, 그리고 커밋 로그



2 Vscod에서 파일 비교 git diff

VSCODE에서 GUI와 CLI 동시에 활용

깃 배시로도 작업 가능

The screenshot shows the VS Code interface with the following components:

- Left Sidebar:** Source Control view showing a repository named 'gdiff'. The file 'h.txt' is staged for commit. The commit message is '가나다라'.
- Editor:** The file 'h.txt' is open, showing the following content:

```
1 1234
2 ABCD
3 AB12
4 가나다라
5 !@#$
6
```
- Right Sidebar:** Git Graph view showing the commit history. The 'main' branch is selected, and the commit '92920ef' is highlighted. The commit message is 'Alphanumeric'.
- Terminal:** The terminal shows the following commands and output:

```
PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
$ git log --oneline --graph --all
* 92920ef (HEAD -> main) Alphanumeric
* f7962d5 Alphabeb
* ae6e6ed digit

PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
$ git status -s
MM h.txt

PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
$
```


2 Vscod에서 파일 비교 git diff

\$ git diff

Index(스테이징 영역)와 Working Tree 비교

- Changes 하부 파일 h.txt 클릭

VS Code interface showing the git diff command output in the terminal and the file explorer.

The file explorer shows the 'Changes' section with 'h.txt' selected. The terminal shows the output of the 'git diff' command, which compares the index and the working tree. The output shows that 'h.txt' has been modified, with changes highlighted in green.

The file explorer also shows the 'h.txt' file in the 'Working Tree' section, with the same changes highlighted in green.

Terminal output:

```
$ git diff
diff --git a/h.txt b/h.txt
index e4b016b..cf740f7 100644
--- a/h.txt
+++ b/h.txt
@@ -2,3 +2,4 @@
ABCD
AB12
가나다라
+!@#$
```

Git Graph view shows the commit history:

Graph	Description	Date	Author	Commit
Uncommitted Chang...		19 Nov ...	*	*
main	Alphanu...	19 Nov ...	ai7dnn	92920efa
	Alphabeb	19 Nov ...	ai7dnn	f7962d57
	digit	19 Nov ...	ai7dnn	aefe6edd

2 Vscod에서 파일 비교 git diff

파일 비교 창에서 F7 클릭

파일 비교가 보이는 창에서 F7 클릭

- Git diff 명령 결과와 비슷한 결과를 보이며 파일 비교를 보여줌

→ X를 누르면 제거

```
1 1234 1 1234
2 ABCD 2 ABCD
3 AB12 3 AB12
4 가나다라 4 가나다라
5 5+ !@@$
6 6
```

```
1/1: @@ -2,4 +2,5 @@
2 2 ABCD
3 3 AB12
4 4 가나다라
5 5+ !@@$
6 6
```

```
PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vs
$ git diff
diff --git a/h.txt b/h.txt
index e4b016b..cf740f7 100644
--- a/h.txt
+++ b/h.txt
@@ -2,3 +2,4 @@
ABCD
AB12
가나다라
+!@@$

PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
$
```

2 Vscod에서 파일 비교 git diff

\$ git diff -staged

깃 저장소와 index 비교

- Staged Changes 하부 파일 h.txt 클릭

→ F7으로 보기와 x로 삭제

The screenshot shows the VS Code interface with the Source Control panel on the left. The 'Staged Changes' section shows a file named 'h.txt'. The main editor displays a diff comparison between the staged file and the index. The diff shows a change on line 4 from '가나다라' to '가나다라'. A red box highlights the diff view, showing the 'h.txt (Index)' tab and the diff content.

```

C: > OSS > git > vscode > gdiff > h.txt
1 1234
2 ABCD
3 AB12
4
4+ 가나다라
5

C: > OSS > git > vscode > gdiff > h.txt
1/1: @@ -1,4 +1,5 @@
1 1 1234
2 2 ABCD
3 3 AB12
4 4 +가나다라
4 5
  
```

2 Vscod에서 파일 비교 git diff

⚙️ \$ git diff HEAD

✓ Git Graph의 최상단 Uncommitted Changes를 눌러 파일 h.txt를 클릭

■ HEAD와 changes 비교

➡ F7으로 보기와 x로 삭제

The screenshot shows the VS Code interface with the following components:

- Source Control Panel (Left):** Shows 'SOURCE CONTROL REPOSITORIES' with 'gdiff Git' selected. Below it, 'SOURCE CONTROL' shows 'h.txt' with a message '가나다라' and a commit hash '1@#\$. The 'Commit' button is highlighted.
- Git Graph (Middle):** Displays a table of changes for 'h.txt'. The table has columns: G..., D..., Date, Author, Commit. The first row is 'Unc... 19 Nov ... *'. The message 'Displaying all uncommitted changes.' is shown.
- Diff View (Right):** Shows the diff for 'h.txt (Uncommitted)'. The diff shows changes to 'h.txt' with a message '가나다라' and a commit hash '1@#\$. The diff is highlighted with a red dashed box.
- Terminal (Bottom):** Shows the command prompt output for the 'git diff HEAD' command. The output is:


```
PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
$ git diff HEAD
diff --git a/h.txt b/h.txt
index 0d7fe64..cf740f7 100644
--- a/h.txt
+++ b/h.txt
@@ -1,3 +1,5 @@
1234
ABCD
AB12
+가나다라
+!@#$
```

2 Vscod에서 파일 비교 git diff

\$ git diff HEAD~2 HEAD^

Git Graph의 Alphabet 커밋을 눌러 파일 h.txt를 클릭

- 클릭한 커밋의 이전(HEAD~2)과 클릭한 커밋(HEAD^)을 비교

→ F7으로 보기와 x로 삭제

The screenshot shows the VS Code Git Graph interface. The left sidebar shows the 'SOURCE CONTROL' view with 'h.txt' selected. The main area shows the 'Graph' view with a commit history. The 'HEAD~2' commit is highlighted, and the 'HEAD^' commit is selected. The diff view shows the changes between these two commits for the file 'h.txt'. The diff shows a change from '1234' to 'ABCD'. The terminal window at the bottom shows the command 'git diff HEAD~2 HEAD^' and its output:

```
PC@DESKTOP-482NOAB MINGW64 /c/OSS/git/vscode/gdiff (main)
$ git diff HEAD~2 HEAD^
diff --git a/h.txt b/h.txt
index 81c545e..51c03ff 100644
--- a/h.txt
+++ b/h.txt
@@ -1,2 @@
1234
+ABCD
```

Summary

» 비주얼스튜디오코드에서 깃 저장소를 생성하고 파일 생성

- ◆ 탐색기 활동바, 버전 컨트롤 활동바

» 비주얼스튜디오코드에서 파일을 추가, 커밋

- ◆ Changes, Staged Changes

» Changes, Staged Changes

- ◆ 확장 Git Graph 설치와 활용

» 비주얼스튜디오코드에서 파일 비교 기능 수행

- ◆ Changes, Staged Changes
- ◆ Git Graph
 - 다음 하부의 파일을 클릭
 - Uncommitted changes
 - 각 커밋