

Git ?? ?? (Basic)

1. Git? ?? ??? ??

Git은 "diff"를 추적, 저장하는 도구(??)를 관리하는 도구.

“?? ?? = ?? ?? ?? ?? (??, ?? ?? ??) ”

2. Git? 3? ??

?? ??	→	?? ??	→	?? ??
(Working Directory)		(Staging/Index)		(Repository/.git)
?? ??		git add		git commit

??	??
?? ??	?? ?? ?? /?? ??
?? ?? (Index)	?? ?? ?? ?? (git add)
?? ??	?? ?? .git ??

3. ?? Git ?? ??

Git은 4?? ?? ?? ?? :

??	??	??
?? Blob	?? ?? ?? (binary large object)	hello.java ??
?? Tree	?? ?? ?? ??	?? ??
?? Commit	?? ?? + ?? ?? + tree	"?? ?? "
?? Tag	?? ?? ?? ??	v1.0.0

□□ □□ tree□ □□□ , tree□ blob□ □□□□ .

```
[commit]
  |
  v
[tree]
├─ [blob] hello.txt
└─ [blob] app.js
```

4. HEAD??

HEAD → or

- `HEAD` → `main`
- `git checkout` `main` `HEAD` `main` `main`
- `git reset` `HEAD` `main` `main`

5. ?????

 .














```
A---B---C ← main (□□□)
      ↑
    HEAD
```

- \rightarrow
- \rightarrow

6. Reflog??

```
git reflog
```

→ HEAD

- `reset`  `checkout`   
-     `Git` `HEAD`      

7. .git ????? ?? ??

```
.git/
├─ objects/      ← 00, 00, 00, 00 00 000
├─ refs/         ← 0000, 00 00 000
├─ HEAD          ← 00 0000 000
├─ index         ← 0000 00
├─ logs/         ← reflog 00 00
└─ config        ← Git 00
```

? ??

??	??
Commit	00 000 + 00 00 00
Blob	00 00 00
Tree	00 00
Branch	000 0000 000
HEAD	00 000 or 00
Staging Area	00 000 0000 00
Reflog	HEAD 00 00 00 00 (000 00)