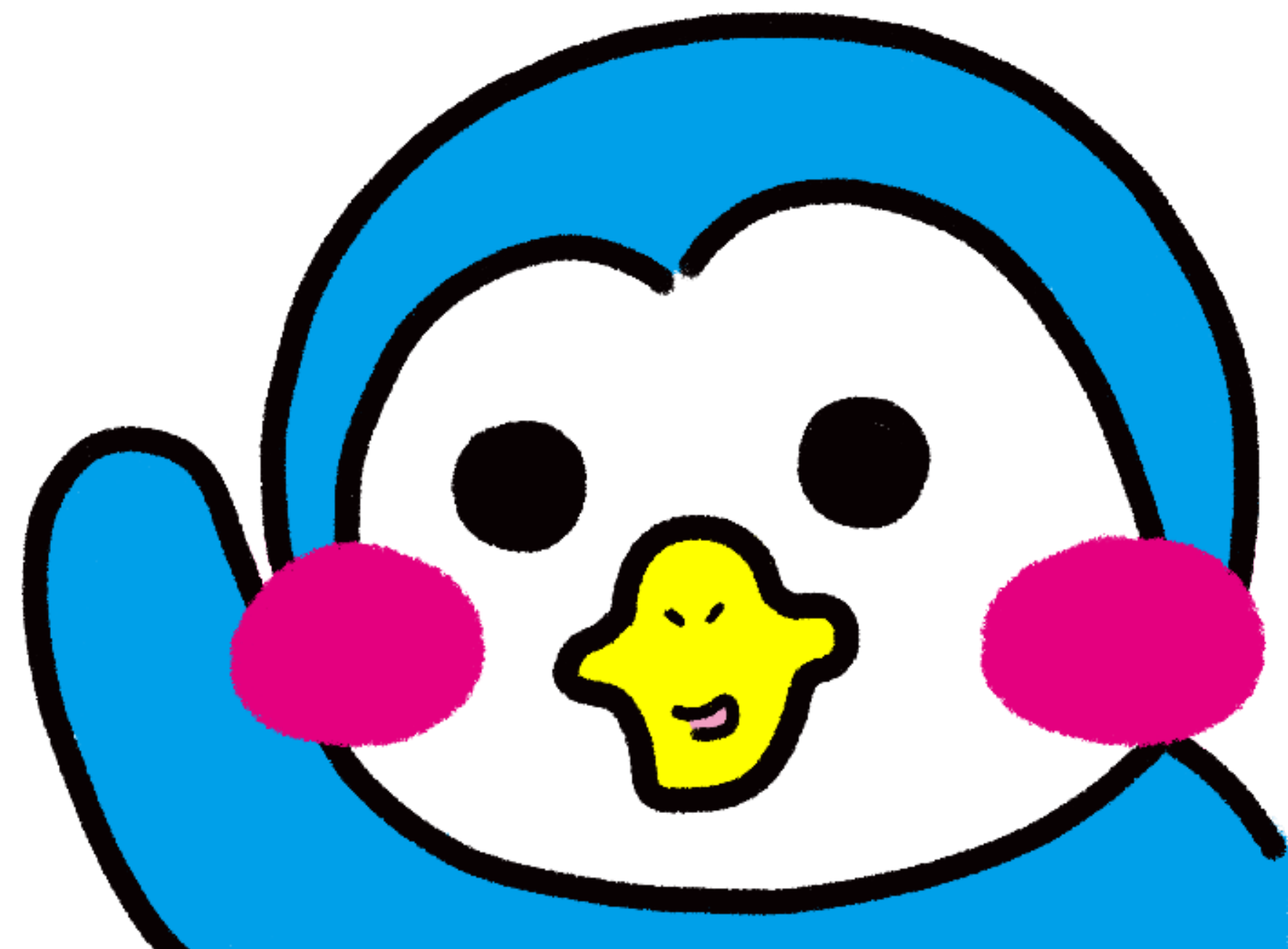


# 깃과 깃허브가 처음인 당신에게

커밋 이력 자세히 보기





## 커밋 이력 살펴보기

명령행에 `git log`를 입력하면 커밋 이력이 최신순으로 출력된다. 이때의 구성 요소로는 커밋 해시, 작업자 정보, 날짜와 시간, 커밋 메시지가 출력된다. 또한 특정 커밋에 대해서는 브랜치명과 HEAD 참조자가 표시된다.

```
MINGW64:/c/GGFB-PRACTICE/git-sample
DELL@DESKTOP-R7DF4U7 MINGW64 /c/GGFB-PRACTICE/git-sample (master)
$ git log
commit 168c3b94fc508c831c8d6a31a8a97c4c082f5feb (HEAD -> master)
Author: beansdrawer <beansdrawer@naver.com>
Date: Tue Jul 11 10:11:14 2023 +0900

    두 번째 커밋

commit cbe55a4eb93c4d14e08be2b921fc7837e115b0f0
Author: beansdrawer <beansdrawer@naver.com>
Date: Tue Jul 11 10:10:45 2023 +0900

    첫 번째 커밋
```

```
DELL@DESKTOP-R7DF4U7 MINGW64 /c/GGFB-PRACTICE/git-sample (master)
$ git log
commit 168c3b94fc508c831c8d6a31a8a97c4c082f5feb (HEAD -> master)
Author: beansdrawer <beansdrawer@naver.com>
Date: Tue Jul 11 10:11:14 2023 +0900
```

두 번째 커밋

```
commit cbe55a4eb93c4d14e08be2b921fc7837e115b0f0
Author: beansdrawer <beansdrawer@naver.com>
Date: Tue Jul 11 10:10:45 2023 +0900
```

첫 번째 커밋

DELL@DESKTOP-R7DF4U7 MINGW64 /c/GGFB-PRACTICE/git-sample (master)

\$ git log

commit 168c3b94fc508c831c8d6a31a8a97c4c082f5feb (HEAD -> master)

Author: beansdrawer <beansdrawer@naver.com> 작업자 정보

Date: Tue Jul 11 10:11:14 2023 +0900

날짜 및 시간

두 번째 커밋

커밋 메시지

commit cbe55a4eb93c4d14e08be2b921fc7837e115b0f0

Author: beansdrawer <beansdrawer@naver.com>

Date: Tue Jul 11 10:10:45 2023 +0900

첫 번째 커밋



DELL@DESKTOP-R7DF4U7 MINGW64 /c/GGFB-PRACTICE/git-sample (master)

\$ git log

commit 168c3b94fc508c831c8d6a31a8a97c4c082f5feb (HEAD -> master)

Author: beansdrawer <beansdrawer@naver.com> 커밋 해시 HEAD 참조자 브랜치명

Date: Tue Jul 11 10:11:14 2023 +0900

두 번째 커밋

commit cbe55a4eb93c4d14e08be2b921fc7837e115b0f0

Author: beansdrawer <beansdrawer@naver.com>

Date: Tue Jul 11 10:10:45 2023 +0900

첫 번째 커밋



갑자기 뭐야 어려운 단어나 들이대고

커밋 이력에 표시되는 각종 정보들은 각각 다음과 같은 의미와 역할을 지닌다.

- 커밋 해시 : 커밋 기록에 대한 고유 식별자
- 브랜치명 : 기존 저장소에서 분기된 저장소의 복사본인 ‘브랜치’의 이름
- HEAD : 현재 작업 중인 브랜치를 가리키는 포인터(참조자)

=> 브랜치에 대해서는 추후 영상에서 따로 다룰 예정입니다 :)



## git log 옵션 & 커밋 이동 명령어

```
$ git log -p
```

커밋의 변경사항까지 함께 출력하는 옵션

```
$ git log --oneline
```

각 커밋을 요약해 한 줄 씩 출력하는 옵션

```
$ git checkout 커밋해시
```

문서 내용을 특정 커밋의 내용으로 되돌리는 git 명령어

```
$ git reflog
```

HEAD 포인터의 참조 이력을 출력하는 git 명령어



수고하셨습니다 :)