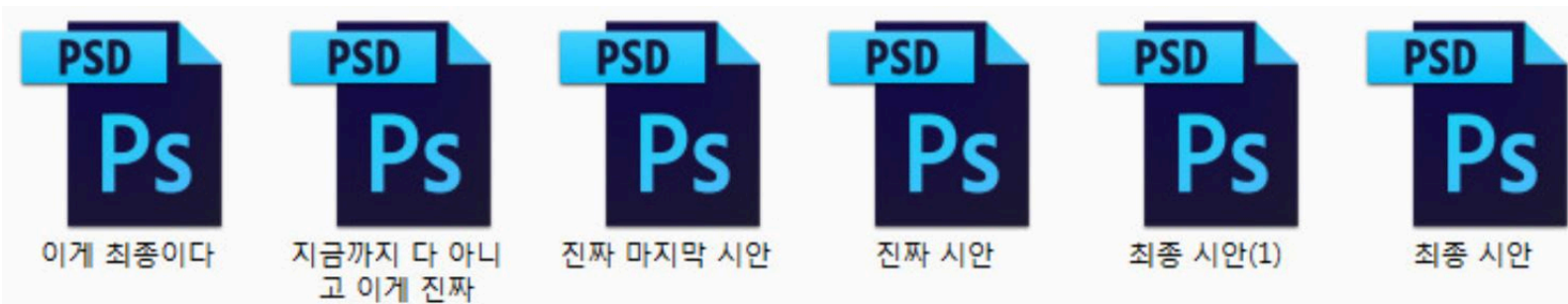


# LLM 활용 인공지능 서비스 개발자 양성과정

도봉 SeSAC 캠퍼스 X **Saltlux**

강사 최동혁

# Git은 왜 필요할까?



- 우리는 일을 하면서 수많은 선택의 기로에 놓인다.
- 특히 컴퓨터로 작업을 할 때에가 유독 그렇다.
- 그럴 때마다 우리는 늘 새로운 "버전"을 만들고 파일이름에 그 버전의 특징을 기록해 두곤 했다.
- (감상문\_최종.hwp, 감상문\_최종\_latest.hwp, ...)

# Git은 왜 필요할까?

- 그리고 예전에 작업했던 내용을 다시 찾으려면 무수히 많은 비슷한 파일들을 하나 하나 열어보면서 확인해 본 경험이 있을 것이다.
- 혹은 작업을 하다가 **“아 이게 아닌데? 옛날 꺼가 더 좋았던 것 같은데”** 라고 생각할 때가 많다.
- 혹은 협업을 하다 서로 작업한 내용을 합칠 때, 일일이 내용을 확인해 가면서 작업한 내용을 합치곤 한다.
- 이런 문제들을 해결해 주려고 만든 것이 버전관리 시스템이다.
- 잠깐! Git이 버전관리 시스템 중 가장 많이 쓰이지만, 이것만 있는 것은 아닙니다.

# Commit

- Git은 파일을 **복붙**해서 여러 개 만들어 관리하는 것이 아니다.
- Git은 구 버전의 파일(**파일 A**)과 새 버전의 파일 (**파일 A`**)의 **딱 바뀐 만큼만** 기록해 둔다.
- 그렇다면 언제를 기준으로 **파일 A**와 **파일 A`** 을 정할 것인가? 그 기준을 정해주는 작업이 바로 "commit"이다. 그리고 commit 시점을 기준으로 확정된 버전을 "snapshot"이라고 부른다.

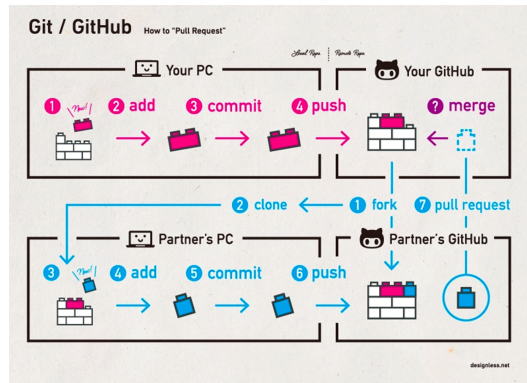
# Commit

- 개발자들의 작업은 보통 commit을 기준으로 하기 때문에 굉장히 중요하다.



- (참고) add는 "이제부터 너를 commit 할 거야"라는 뜻이다.

```
git add 감상문.hwp  
git commit -m "감상문 초안" ==> snapshot 생성!
```



## 참고자료

- <https://git-scm.com/book/ko/v2/시작하기-Git-기초>
- <https://wikidocs.net/149672>

# Git과 Github의 차이는?

Git은 내 컴퓨터 안(Local)에서 버전관리를 도와주는 프로그램 이에요.

Github은 Git을 조금 더 편리하게 사용할 수 있게 도와주는 플랫폼(Remote, Server) 이에요.

- 여러 사람들이 함께 Git을 사용할 수 있어요.
- • 나의 Git을 백업할 수 있어요. (북극에도!)

간단히 말하면, Git은 도구이고 GitHub는 Git을 사용하는 플랫폼이에요.

# Github에서 내가 심은 잔디 구경하기

github.com/{내 아이디} 클릭해보세요. 여러분이 얼마나 많은 기여(commit)을 했는지 알 수 있어요.

