



Git 기초 강의



Func gossip

git이란 단어는 영국에서 사용되는 속어로 '고집센 사람', '항상 자기자신이 옳다고 생각하며 따지기 좋아하는 사람'을 의미

기분 좋을 때는 "Global Information Tracker" 의 약자

제대로 동작하지 않을 때는 "Goddamn idiotic truckload of shit" 망할 멍청한 똥덩어리의 약자

Git은 소프트웨어 개발에서 버전 관리를 위해 사용되는 분산 버전 관리 시스템입니다. 이제 Git의 기초적인 개념과 사용 방법을 알아보겠습니다.

1. Git 기본 용어

- 저장소(Repository): Git에서 버전 관리 대상이 되는 디렉토리 또는 파일들의 모음입니다.
- 커밋(Commit): Git에서 변경 사항을 저장하는 작업을 의미합니다.
- 브랜치(Branch): 독립적인 작업 영역을 의미합니다. 여러 작업자가 같은 저장소에서 동시에 작업할 때, 각자의 작업 내용을 서로 간섭하지 않게끔 할 수 있습니다.
- 푸시(Push): 로컬 저장소에 저장된 변경 사항을 원격 저장소에 업로드하는 작업을 의미합니다.
- 풀(Pull): 원격 저장소에서 변경된 사항을 로컬 저장소에 다운로드하는 작업을 의미합니다.

2. Git 설치

- Git을 사용하기 위해서는 운영체제에 맞는 Git 설치 파일을 다운로드하여 설치해야 합니다. Git 공식 홈페이지에서 다운로드 및 설치 가이드를 제공합니다.

3. Git 저장소 생성 및 초기화

- Git 저장소를 생성하기 위해 먼저 작업 디렉토리를 선택합니다.
- 작업 디렉토리 내에서 `git init` 명령어를 사용하여 Git 저장소를 초기화합니다.

4. 커밋 작성 및 로그 확인

- 파일을 추가, 수정 또는 삭제한 후, `git add` 명령어를 사용하여 변경 사항을 스테이징 영역에 추가합니다.
- 변경 사항이 스테이징 영역에 추가된 후, `git commit` 명령어를 사용하여 변경 사항을 로컬 저장소에 저장합니다.
- `git log` 명령어를 사용하여 커밋 로그를 확인할 수 있습니다.

5. 브랜치 생성 및 전환

- `git branch` 명령어를 사용하여 브랜치를 생성합니다.
- `git checkout` 명령어를 사용하여 브랜치를 전환합니다.

6. 원격 저장소 생성 및 푸시

- GitHub, GitLab 등의 원격 저장소를 이용하여 Git 저장소를 생성합니다.
- `git remote add` 명령어를 사용하여 원격 저장소를 추가합니다.
- `git push` 명령어를 사용하여 로컬 저장소에 저장된 변경 사항을 원격 저장소에 업로드합니다.

7. 병합(Merge)

- `git merge` 명령어를 사용하여 브랜치를 병합합니다.

8. 충돌 해결

- 여러 작업자가 같은 파일을 수정하고 커밋할 경우, 충돌이 발생할 수 있습니다.
- `git status` 명령어를 사용하여 충돌이 발생한 파일을 확인합니다.
- 충돌이 발생한 파일을 수정하고, `git add` 명령어를 사용하여 스테이징 영역에 추가합니다.

- 변경 사항이 스테이징 영역에 추가된 후, `git commit` 명령어를 사용하여 변경 사항을 로컬 저장소에 저장합니다.

9. .gitignore 파일 작성

- `.gitignore` 파일을 생성하여 Git에서 추적하지 않을 파일을 설정합니다.
- 이 파일에 등록된 파일들은 로컬 저장소 및 원격 저장소에서 제외됩니다.

10. Git의 협업 기능

- Git은 여러 명의 작업자가 함께 작업할 때 효율적으로 협업할 수 있는 기능을 제공합니다.
- 풀 리퀘스트(Pull Request)를 이용하여 브랜치를 비교하고 변경 사항을 검토하고 병합할 수 있습니다.
- 이슈 트래킹(Issue Tracking)을 이용하여 개발 과정에서 발생한 이슈를 관리하고 해결할 수 있습니다.

위와 같은 Git의 기초적인 개념과 사용 방법을 익히고, 실제로 프로젝트를 진행하며 Git을 활용하면서 더욱 능숙하게 사용할 수 있을 것입니다.