

Git 활용하기 및 개요

<Git 활용>

git init: 새로운 Git 저장소를 초기화합니다.

git clone <repository URL>: 원격 저장소를 로컬에 복제합니다.

기본 작업 흐름:

git add <파일>: 변경된 파일을 스테이징 영역에 추가합니다.

git commit -m "커밋 메시지": 스테이징 영역에 있는 파일을 커밋하여 변경 내용을 저장합니다.

상태 확인:

git status: 작업 디렉토리와 스테이징 영역의 상태를 확인합니다.

브랜치:

git branch: 로컬 브랜치 목록을 보여줍니다.

git checkout <브랜치>: 특정 브랜치로 이동합니다.

git checkout -b <새 브랜치>: 새로운 브랜치를 생성하고 해당 브랜치로 이동합니다.

git merge <브랜치>: 다른 브랜치를 현재 브랜치에 병합합니다.

원격 저장소 관리:

git remote add <원격 저장소 이름> <URL>: 원격 저장소를 추가합니다.

git push <원격 저장소 이름> <브랜치>: 로컬 변경 사항을 원격 저장소로 푸시합니다.

git pull <원격 저장소 이름> <브랜치>: 원격 저장소에서 변경 사항을 가져와 로컬에 병합합니다.

로그 및 이력:

git log: 커밋 로그를 확인합니다.

git log --oneline: 각 커밋을 한 줄로 요약해서 표시합니다.

변경 내용 비교:

git diff: 작업 디렉토리와 스테이징 영역의 변경 사항을 비교합니다.

git diff <commit1> <commit2>: 두 커밋 간의 변경 사항을 비교합니다.

태그:

git tag: 태그 목록을 표시합니다.

git tag -a <태그 이름> -m "태그 메시지": 주석과 함께 태그를 생성합니다.

git push origin --tags: 모든 태그를 원격 저장소로 푸시합니다.

gitk는 Git 저장소의 히스토리를 시각적으로 보여주는 그래픽 인터페이스를 제공하는 Git 커맨

드라인 도구입니다. gitk를 실행하면 Git 저장소의 커밋 그래프, 브랜치, 태그 등을 쉽게 확인할 수 있습니다.

[UNIX 계열에서의 명령어]

cat [파일명] : 파일의 내용을 확인할때 쓰는 UNIX계열 운영체제(리눅스) 명령어

mkdir [디렉토리명] : 디렉토리 만들기

touch [파일명] : 파일을 바로 만들기

env : 환경변수

env | less 검색 : 환경변수에서 찾을 문자를 페이지별로 스크롤

ls : 현재 디렉토리의 파일과 디렉토리 목록 나타내기

주로 세부 옵션으로 ls -l ls -a를 많이 씀(ls -al)

<nano 편집기>

nano는 터미널에서 사용할 수 있는 텍스트 편집기이다. 사용하기 간편하며, 기본적인 기능들을 제공하여 텍스트 파일을 편집할 수 있는 특징이 있다.

<기본 사용법>

[파일 열기 또는 생성하기]

nano filename.txt

파일이 없으면 파일을 생성하고 편집기로 넘어감.

파일이 있는경우 파일을 열고 편집가능

[텍스트 편집]

Ctrl + O: 저장 (파일에 쓰기)

Ctrl + X: 종료 (나가기)

Ctrl + G: 도움말 보기 (명령어 목록)

[커서 이동]

화살표 키를 사용하여 커서를 이동할 수 있습니다.

[텍스트 입력 및 삭제]

키보드를 사용하여 텍스트를 입력하고, Backspace 키를 사용하여 삭제할 수 있습니다.

[복사 및 붙여넣기]

Ctrl + Shift + 6: 시작 부분을 선택합니다.

Ctrl + K: 선택한 부분을 잘라냅니다.

Ctrl + U: 잘라낸 내용을 붙여넣습니다.

[검색 및 바꾸기]

Ctrl + W: 검색

Ctrl + \: 바꾸기

[파일 열기]

Ctrl + R: 다른 파일을 열기

[종료 시 저장 여부 묻기]

Nano를 종료할 때, 수정된 내용이 있는 경우 저장 여부를 묻습니다.

세부적인 편집을 할 때는 Vim 같은 텍스트 편집기를 고려할 수 있습니다.

<git SSH 통신>

SSH란?

SSH는 Secure Shell Protocol의 의미로, 네트워크 상 다른 컴퓨터의 셸을 사용할 수 있게 해 주는 프로그램 혹은 그 프로토콜을 의미한다. SSH를 사용하면 원격에서 네트워크 상의 컴퓨터에 접속할 수 있다. SSH의 특징으로 강력한 보안을 제공하기에 안전하지 못한, 개방된 네트워크에서도 안전하게 통신할 수 있다.

RSA란?

RSA 암호는 공개키 암호시스템의 하나로, 암호화뿐만 아니라 전자서명이 가능한 최초의 알고리즘이다.

RSA는 두 개의 키(공개키와 비밀키)를 사용하여 데이터를 암호화하고 복호화하는 데 사용된다.

<생성 방법>

ssh-keygen 엔터 엔터

cd .ssh

notepad id_rsa.pub

RSA 3072가 생성됨.

[내용 추가 정리중]