

5주차: Server 와 Database

서울시립대학교 통계데이터사이언스학과
전종준

Contents

- 리눅스의 기초와 서버 원격 접속
- Vscode 에서 Conda 가상환경의 사용

01. 리눅스의 기초와 서버 원격 접속

리눅스의 기초

리눅스

- 정의:

- 리눅스는 유닉스(UNIX) 계열의 운영체제를 기반으로 한 오픈소스 운영체제 커널.
- 처음에는 1991년 Linus Torvalds가 개발
- 전 세계 개발자들이 협업하여 유지 관리

- 구조

- 리눅스는 커널(kernel) + 다양한 유틸리티/라이브러리 + 셸(shell) 로 구성됨
- GUI는 선택사항이며, CLI 환경 중심

리눅스의 기초

리눅스의 기본구조

- 커널 (Kernel)

- 운영체제의 핵심(중앙) 구성요소로, 하드웨어와 소프트웨어 사이를 중재하는 역할
 - ✓ 프로세스 관리: 여러 프로그램이 동시에 실행될 수 있도록 관리
 - ✓ 메모리 관리: RAM을 효율적으로 분배하고 보호
 - ✓ 파일 시스템: 저장장치와 파일 접근 처리
 - ✓ 장치 제어: 키보드, 마우스, 디스크, 네트워크 등 하드웨어 제어
 - ✓ 시스템 호출 처리: 사용자 프로그램이 커널 기능을 사용할 수 있도록 중개

리눅스의 기초

리눅스의 기본구조

- 라이브러리 (Library)

- 사용자 프로그램과 커널 사이에서 공통 기능을 모아둔 코드 집합입니다.

예: 수학 계산, 문자열 처리, 파일 입출력 등

- 프로그램이 직접 커널 기능을 호출하지 않고, 라이브러리를 통해 간접 호출함 → 코드 재사용성 및 안정성 향상

예: printf(), malloc() 등은 라이브러리 함수이며, 내부적으로 커널에 시스템 호출(syscall)을 요청

리눅스의 기초

리눅스의 기본구조

- 셸 (Shell)
 - 사용자가 운영체제와 상호작용할 수 있게 해주는 명령어 해석기(인터페이스)입니다
예: bash (가장 보편적인 리눅스 셸 (Bourne Again SHell))
zsh (확장성과 자동완성 기능이 우수)
sh (전통적인 유닉스 기본 셸)
- 명령 전달 체계
 - 사용자 ↔ 셸 ↔ 라이브러리 ↔ 커널 ↔ 하드웨어

리눅스의 기초

Ubuntu

- 우분투는 리눅스 배포판(Linux distribution) 중 하나로, Debian 기반의 오픈소스 운영체제
- 2004년 Canonical Ltd. 에서 개발하여 사용자 친화적인 데스크탑 및 서버 환경 제공을 목표로 함.

SSH를 이용한 서버접속

SSH란?

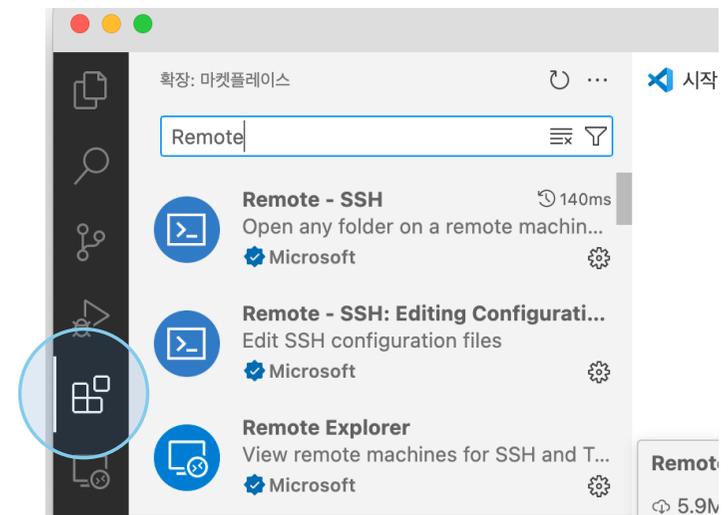
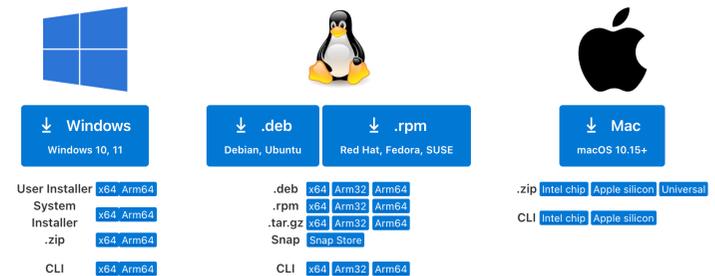
- SSH (Secure Shell): 네트워크 상에서 암호화된 통신을 통해 다른 컴퓨터에 원격 접속할 수 있는 프로토콜
- 주로 리눅스 서버에 접속하거나, 원격 개발환경 구축에 사용됨

사전준비

- VS Code 설치 (<https://code.visualstudio.com/>)
- Remote - SSH 확장 설치
- Ubuntu 서버의 IP 주소 및 로그인 계정 정보

Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.



SSH를 이용한 서버접속

SSH config 파일 설정

- Remote SSH 호스트연결 -> SSH 호스트 구성 -> config 파일 선택
- 기본구조

```
Host my-server
```

```
HostName 123.123.123.123
```

```
User ubuntu
```

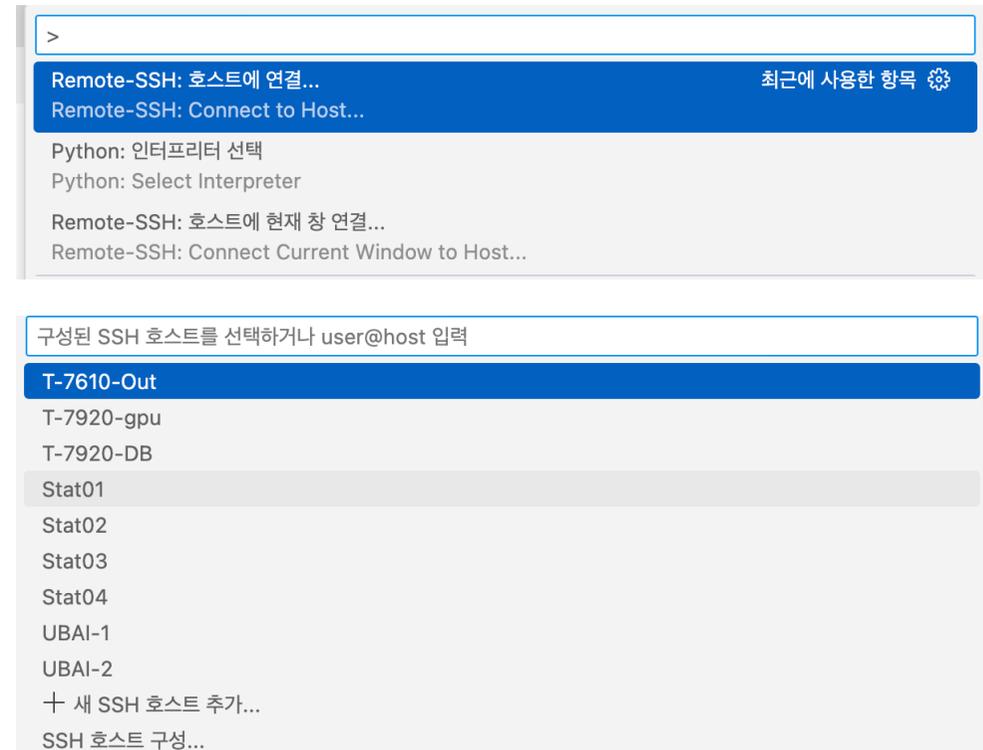
```
IdentityFile ~/.ssh/my_private_key
```

```
Port 22
```

SSH를 이용한 서버접속

접속하기

- Ctrl+Shift+P → Remote-SSH: Connect to Host 입력
- Host 별칭 선택 (my-server 등)
- 접속 성공 시, 하단에 SSH: my-server 표시됨
- 서버의 파일 탐색기, 터미널, 코드 편집 등이 가능



SSH를 이용한 서버접속

Password 변경

- passwd

사용안내

- 기한: ~ 2025. 5. 27 이후 계정 및 데이터 삭제
- 서버로 웹크롤링 금지
- 파일 업로드 100G 미만

SSH를 이용한 서버접속

루트 디렉토리(/)란?

- 리눅스 파일 시스템의 최상위 디렉토리
- 모든 경로는 /로부터 시작됨 (Windows에서 C:\와 유사한 개념)
- 하위 디렉토리들이 / 아래에 계층 구조로 존재함

SSH를 이용한 서버접속

/home

- 사용자별 홈 디렉토리가 위치하는 곳
예: 사용자 alice의 홈 디렉토리는 /home/alice
- 각 사용자는 이 경로 안에서 개인 파일과 설정을 저장
- 권한 설정으로 사용자 간 접근이 제한됨

/etc

- 시스템 설정 파일(configuration files) 이 모여 있는 디렉토리
예: 네트워크 설정 /etc/network/interfaces, 사용자 정보 /etc/passwd
- 대부분 텍스트 파일로 구성되며, 수동으로 편집 가능
- 일반 사용자 권한으로는 변경 불가 (보안상 중요)

SSH를 이용한 서버접속

/var

- 변동 데이터(variable data) 저장 디렉토리
- 로그 파일: /var/log/
- 메일, 캐시, spool, 잠금 파일(lock files) 등 운영 중 생성되는 데이터 위치
- 시스템이 계속 동작하면서 파일 크기가 커질 수 있음

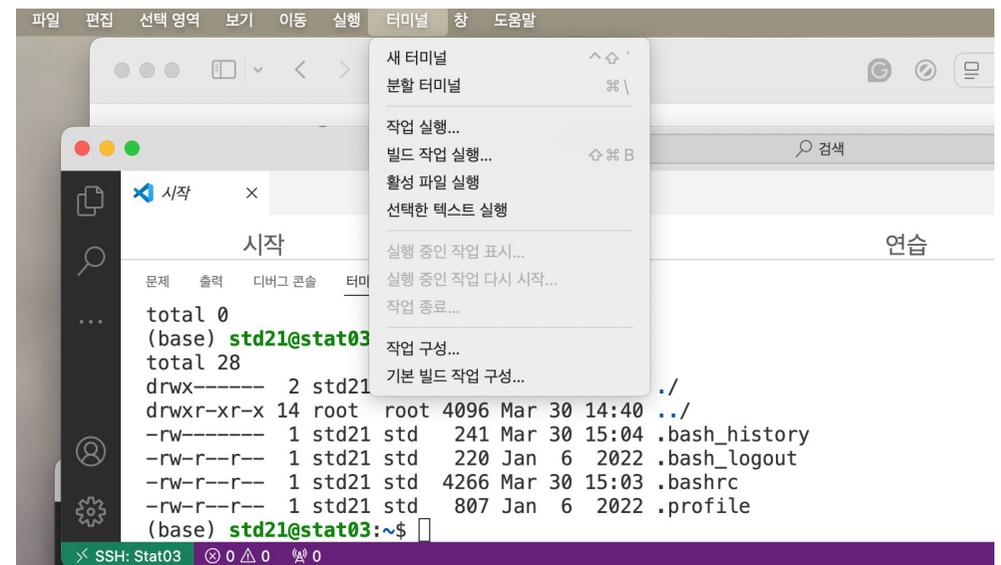
/bin

- 시스템 기본 명령어 실행 파일(binary files)이 저장된 디렉토리
예: ls, cp, mv, rm 등 사용자 및 시스템 필수 명령어
- 부팅 시 꼭 필요한 명령어들이 위치

SSH를 이용한 서버접속

터미널 켜기

- 터미널 > 새 터미널
아래쪽에 bash shell 을 통해 명령어를
입력할 수 있음



SSH를 이용한 서버접속

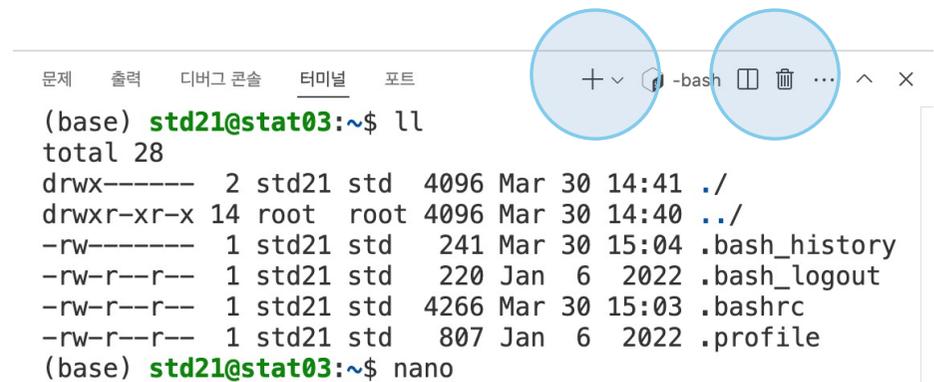
명령어의 기초

- ls -a : 디렉토리 안 하위디렉토리 출력 (상세정보를 확인시 ll 이용)
- cd : change directory
 - cd /home/std01 : 특정 디렉토리로 이동
 - cd /Document : 상대경로로서 이동
 - cd ~ : 홈 디렉토리로 이동
 - cd .. : 상위디렉토리로 이동
- mkdir: make a directory
 - mkdir work : 현재 디렉토리내 work 디렉토리 만들기

SSH를 이용한 서버접속

우분투 편집기

- nano
 - 실행: 터미널에서 nano 입력후 엔터
 - 텍스트 입력 후 저장: ctrl + o -> 파일명 입력 후 엔터
(file.txt 저장)
 - 나가기: ctrl + x
- 참고: 터미널을 새로 시작하고 싶은 경우 vscode의 휴지통 클릭 후 더하기 클릭



A terminal window screenshot showing the execution of the 'll' command and the start of the 'nano' editor. The terminal output lists files and directories with their permissions, owners, sizes, and timestamps. The 'nano' command is entered at the prompt.

```
(base) std21@stat03:~$ ll
total 28
drwx----- 2 std21 std 4096 Mar 30 14:41 ./
drwxr-xr-x 14 root  root 4096 Mar 30 14:40 ../
-rw----- 1 std21 std  241 Mar 30 15:04 .bash_history
-rw-r--r-- 1 std21 std  220 Jan  6  2022 .bash_logout
-rw-r--r-- 1 std21 std 4266 Mar 30 15:03 .bashrc
-rw-r--r-- 1 std21 std  807 Jan  6  2022 .profile
(base) std21@stat03:~$ nano
```

SSH를 이용한 서버접속

파일 또는 디렉토리 복사

- cp file.txt copy.txt # 파일 복사
- cp -r dir1 dir2 # 디렉토리 복사

이동

- mv a.txt b.txt # 파일 이름 변경
- mv a.txt /home/username/ # 다른 위치로 이동

SSH를 이용한 서버접속

파일 내용을 출력

- `cat file.txt` # 모두 출력
- `head file.txt` # 앞 줄 출력
- `head -n 5 file.txt` # 앞 5줄 출력
- `tail file.txt` # 뒷줄 출력
- `tail -n 20 file.txt` # 뒤 5줄 출력
- `tail -f file.txt` # 파일이 갱신될 때 실시간으로 보기

SSH를 이용한 서버접속

실습

- `mkdir -p ~/linux_practice/day1` # -p 는 하위 디렉토리까지 만들기
- `cd ~/linux_practice/day1`
- `touch note.txt memo.txt` # 파일 만들기
- `mkdir temp`

(linux_practice/day1 폴더를 만들고, 그 안에 note.txt, memo.txt 파일과 temp 디렉토리를 생성함.)

SSH를 이용한 서버접속

파일 복사와 이름 변경

- cp note.txt temp/note_copy.txt
- mv memo.txt memo_old.txt

(note.txt 파일을 temp 디렉토리에 복사하고, memo.txt의 이름을 memo_old.txt로 변경)

파일 내용 확인하기

- echo "This is a note." > note.txt
- cat note.txt
- head -n 1 note.txt

02. Vscode 에서 Conda 가상환경의 사용

miniconda

conda 명령어

- conda init
 - conda activate 같은 명령어가 제대로 작동하도록 셸 초기 설정을 해주는 명령어
- conda create -n 환경이름 python=3.10
- conda env list # conda 환경확인
- conda activate 환경이름 # 환경 활성화
- conda deactivate 환경이름 # 환경 비활성화
- conda env remove -n 환경이름 # 환경 삭제

miniconda

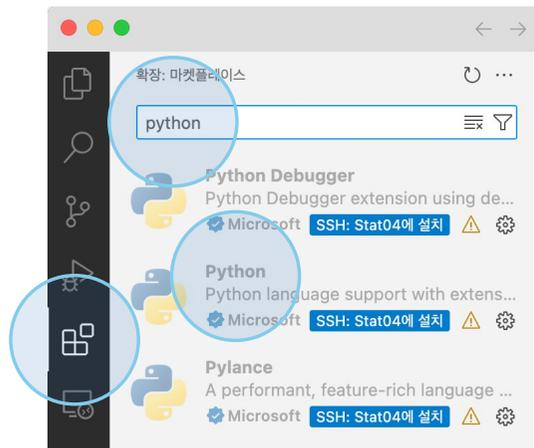
conda 명령어

- 패키지 설치

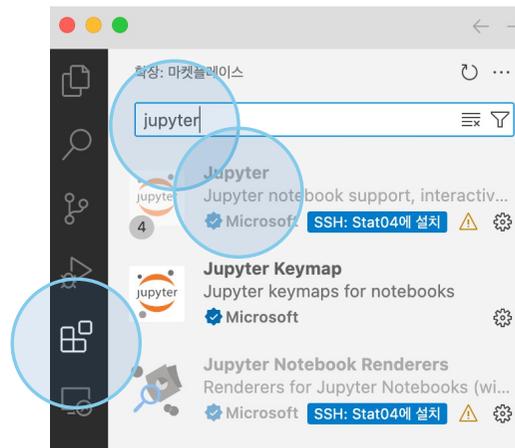
- `conda install numpy # numpy 설치`
- `conda install -c conda-forge seaborn # -c 는 채널 옵션임`

파이썬 실행

파이썬 설치 (vscode)



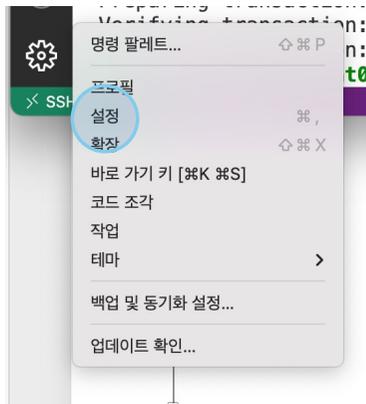
Jupyter 설치



파이썬 실행

설정

- conda install ipykernel. # 주피터의 interactive 실행을 위해 설치
- jupyter 옵션 설정



파이썬 실행

폴더 지정 및 코드 작성

