



VM 활용 매뉴얼

VM을 활용하기 전, 팀장님은 아래 0번을 먼저 꼭 진행해 주세요.

▼ VM이란?

VM은 Virtual Machine의 약자로, 가상 머신을 의미합니다.

“가상”은 컴퓨터공학 용어인데, 결과적으로 가상 머신은 작은 컴퓨터를 의미합니다.

이번 VM, 즉 작은 컴퓨터에는 윈도우, 맥 대신 리눅스라는 OS가 깔려 있습니다.

마우스 대신 키보드로 유닉스 커맨드만 사용하게 됩니다.

VM은 공유할 수 있는 컴퓨터입니다.

www.naver.com 주소를 통해 누구나 네이버를 접속하듯이, VM에 주어진 도메인 주소를 통해 누구나, 여러분이 구현(npm start)한 웹 서비스를 볼 수 있습니다.

▼ 0. VM 사용 전 팀장이 할 일

1. 아래 1번 매뉴얼을 통해 VM에 접속합니다.
2. 아래 명령어를 통해 업데이트합니다 (시간이 조금 소요됩니다).

```
sudo apt update -y && sudo apt upgrade -y
```

3. 아래 명령어를 차례대로 입력하여 node를 설치합니다

```
alice@team24:~$
```

```
# node 설치 curl -o- https://raw.githubusercontent.com/nvm-  
sh/nvm/v0.37.2/install.sh | bash source ~/.bashrc nvm install 16.14.0 nvm use  
16.14.0 # node 잘 설치됐는지 확인 node # node 나가기 컨트롤 + c (2번 연속 누르기) #  
npm도 함께 잘 설치됐는지 확인 npm
```

4. (선택) 아래 명령어를 입력하여 yarn을 설치합니다.

```
alice@team24:~$
```

```
npm install --global yarn
```

▼ 1. 할당된 VM에 접속하기, 나가기

1. 터미널(git bash 등)을 실행한 후, 아래와 같이 입력합니다.
복사 및 붙여 넣기는 마우스 우클릭을 이용할 수 있습니다.

```
# VM 접속 ssh <아이디>@<배포용 VM IP> # 예시 ssh elice@34.64.XX.XXX
```

2. (아래와 같은 문구가 뜰 경우) yes 를 입력합니다.
처음 한 번만 yes를 입력하면, 다음부터는 안 물어봅니다.

```
The authenticity of host '[YOUR_DNS_NAME] ([YOUR_PUBLIC_IP])' can't be established.  
ECDSA key fingerprint is SHA256:[sha256_string] Are you sure you want to continue  
connecting (yes/no/[fingerprint])? yes
```

3. 비밀번호를 입력합니다. 그러면 VM 접속이 완료됩니다.
복사 및 붙여 넣기는 마우스 우클릭을 이용할 수 있습니다.

```
Warning: Permanently added '[YOUR_DNS_NAME],[YOUR_PUBLIC_IP]' (ECDSA) to the list of  
known hosts. elice@[YOUR_DNS_NAME]'s password:
```

4. VM에서 나가고 싶은 경우, 터미널에 exit을 입력합니다.

```
# VM 나가기 exit
```

▼ 2. 리눅스 커맨드 써보기

VM에서는 마우스를 쓰지 않고 터미널 명령어만으로 수행합니다.

처음에는 어색할지라도, 반복해서 써 보면서 익숙해지면 좋습니다.

```
# 터미널에 문구 출력 (echo) $ echo hello # 현재 폴더 위치 (print working directory) $  
pwd # 폴더 생성 (make directory) $ mkdir front # 숨김 폴더 생성 (앞에 점 붙이면 됨) $  
mkdir .secret # 현재 위치의 파일, 폴더 보기 (list) $ ls # 현재 위치의 모든(앞에 점이 붙  
어서 숨겨진 것도) 파일, 폴더 보기 (list all) $ ls -al # 폴더로 이동 (change directory)  
$ cd front # 상위 폴더로 이동 $ cd .. # 파일 생성 (touch) $ touch User.js # 문구 넣어  
서 파일 생성 $ echo hello >> User.js # 파일 내용 출력 (cat) $ cat User.js # 파일 이  
름 변경 (move) $ mv User.js User2.js # 파일을 상위 폴더로 이동 (띄어쓰기 중요) $ mv  
User2.js .. # 폴더를 상위 폴더로 이동 (띄어쓰기 중요) $ mv front .. # 파일 복사 (copy)  
$ cp User2.js User3.js # 폴더 복사 (copy recursively) $ cp -r front front2 # 파일 삭  
제 (remove) $ rm User3.js # 폴더 삭제 (remove recursively) $ rm -rf 폴더이름 # 참고 #  
리눅스에서 취소, 뒤로가기는 보통 컨트롤+c, 혹은 q(큐) 입니다.
```

아래의 페이지 등, 검색을 통해 여러 명령어를 확인하고 연습해 보세요!

자주 사용하는 리눅스 기본 명령어 모음

리눅스에서 자주 사용하는 기본 명령어들을 모아 보았습니다. 리눅스 명령어
야 굉장히 많지만 금방 모두 익히기는 쉽지 않습니다. 하지만 기본 명령어를

 <https://shanepark.tistory.com/196>



▼ 3. 프론트엔드-react 앱 실행해보기

1. (앞 과정 생략하였음) yarn start로 실행하고, 포트번호를 확인합니다.

```
alice@team24:~/portfolio-share-service-racer/front$
```

2. 도메인 + : + 포트번호를 주소창에 입력합니다.

← → C Google에서 검색하거나 URL을 입력하세요.

Google

Q

↓

Google Search

I'm Feeling Lucky

Google offered in: 한국어

3. 만약 포트번호가 이미 사용되고 있다면, react가 알아서 다른 포트번호를 할당합니다.

컨트롤+y를 눌러주면 됩니다.

예시) 3000번이 사용중 → 자동으로 3001번 할당

```
alice@team24:~/portfolio-share-service-racer/front$
```

4. 메모리 사용량을 항상 주의합니다.

react 앱을 여러 개 동시에 키면 VM(컴퓨터)에 부하가 갑니다.

갑자기 화면이 멈추거나, 느려집니다.

react 앱은 안 쓸 때는 바로 컨트롤+c로 끄도록 합니다.

▼ 4. 백엔드-express 앱 실행해보기

1. (앞 과정 생략하였음) .env 파일에서 포트번호를 확인합니다.

```
alice@team24:~/portfolio-share-service-racer/back$
```

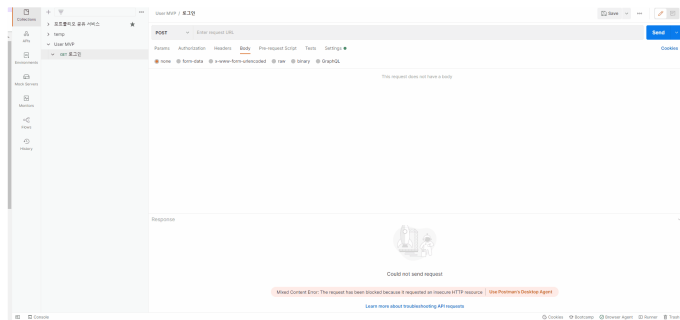
2. yarn start로 실행합니다.

(mongodb 아틀라스를 쓰는 경우, 모든 IP 허용을 해 두어야 데이터베이스가 연결됩니다)

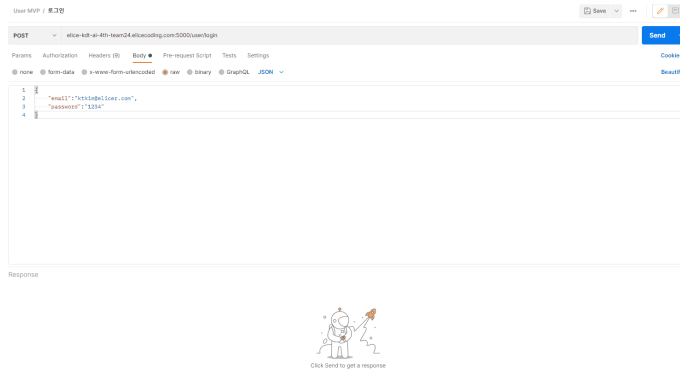
```
alice@team24:~/portfolio-share-service-racer/back$
```

3. 포스트맨에서 도메인 + : + 포트번호를 주소창에 입력하고, 요청합니다.

오류가 날 시, Desktop Agent를 설치합니다.



4. 다음과 같이 SSL 설정을 끄고, 요청합니다.



5. 만약 포트번호가 이미 사용되고 있다면, .env를 수정합니다.

vim으로 .env 파일 열기 → 키보드의 i → 포트번호 고치기 → 키보드의 esc → 키보드의 :x → 키보드의 엔터

예시) 5000번이 사용중 → 5001번으로 수정

```
alice@team24:~/portfolio-share-service-racer/back$
```

6. 메모리 사용량을 항상 주의합니다.

express 앱을 여러 개 동시에 키면 VM(컴퓨터)에 부하가 갑니다.

갑자기 화면이 멈추거나, 느려집니다.

express 앱은 안 쓸 때는 바로 컨트롤+c로 끄도록 합니다.

▼ 5. 프로젝트 완성 후 배포하기

react+express 웹 앱의 배포를 하는 전체적인 흐름은 아래와 같습니다.

자신의 팀에 할당된 VM에 접속하여 진행해야 합니다.

