

1. 윈도우 도커설치
2. 도커 이미지 만들기
  - A. 우분투설치
  - B. Maria db설치
3. ○

## 1. 도커 설치전 준비사항

처음에는 도커기반에 우분투를 설치하려고 하였으나 쉘만 포함된 최소기능만 올라가는 컨테이너 형이라 서버 및 디비를 상시구동하려면 필요작업이 많이 필요함. 따라서 아래의 방식으로 변경함.

윈도우에 깔아서 해도 되나 추후 사내 리눅스 서버에서 활용하기 위해 로컬윈도우 환경에서 속력의 저하가 있는 가상머신(vmware)보다 윈도우환경에 wsl2 를 활용하여 ubuntu(무료) 설치

## 2. 우분투 설치환경

Intel CPU i7-10700  
64 비트 운영체제

- 윈도우 사양:

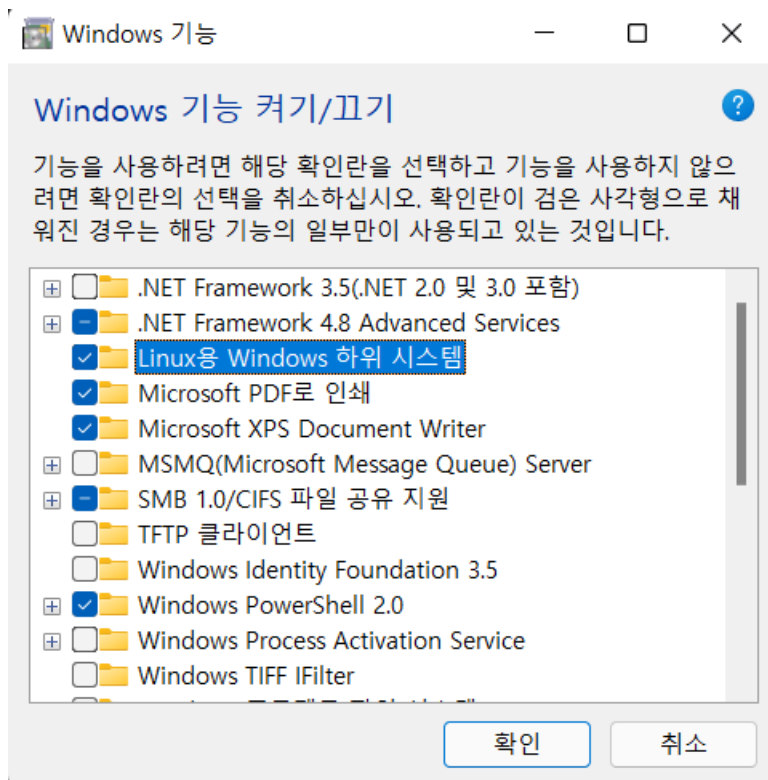
윈도우 10 Home  
버전: 20H2  
OS 빌드: 19042.1165

요구조건

-> x64 시스템의 경우:  
버전 1903 이상, 빌드 18362 이상

\*필요한 경우 버전 업데이트를 합니다.

설정 - 검색 - windows 기능 켜기 - Linux용 Windows 하위 시스템 체크



PowerShell 관리자 권한 실행 – 윈도우 운영체제 관리 인터프리터

WSL설치

```
dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart
```

가상머신 기능 사용

```
dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart
```

완료

```
관리자: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 기능 및 개선 사항에 대 한 최신 PowerShell을 설치 하세요! https://aka.ms/PSWindows
PS C:\Windows\system32> dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart

배포 이미지 서비스 및 관리 도구
버전: 10.0.22000.1

이미지 버전: 10.0.22000.613

기능을 사용하도록 설정하는 중
[=====100.0%=====]
작업을 완료했습니다.
PS C:\Windows\system32> dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart

배포 이미지 서비스 및 관리 도구
버전: 10.0.22000.1

이미지 버전: 10.0.22000.613

기능을 사용하도록 설정하는 중
[=====100.0%=====]
작업을 완료했습니다.
```

최신버전 커널 업데이트 (wsl2 설치)

wsl --set-default-version 2

wsl2 설치확인 – 아래처럼 나오면 설치해야 함

```
PS C:\Windows\system32> wsl --set-default-version 2
WSL 2와의 주요 차이점에 대한 자세한 내용은 https://aka.ms/wsl2를 참조하세요
작업을 완료했습니다.
```

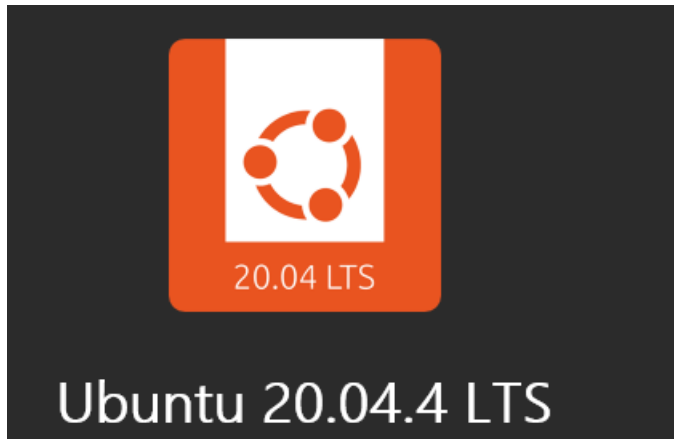
설치 doc

<https://docs.microsoft.com/ko-kr/windows/wsl/install#step-4---download-the-linux-kernel-update-package>

설치 직접다운로드

[https://wslstorestorage.blob.core.windows.net/wslblob/wsl\\_update\\_x64.msi](https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi)

Microsoft store - Ubuntu 20.04.4 LTS 설치



파워셸에서 `wsl -l -v`

```
PS C:\Windows\system32> wsl -l -v
  NAME                STATE      VERSION
*  docker-desktop      Running    2
   Ubuntu-20.04        Running    2
   docker-desktop-data Running    2
```

우분투 apt 서버를 카카오미러페이지로 변경

```
sudo vim /etc/apt/sources.list
```

:입력후

```
%s/archive.ubuntu.com/mirror.kakao.com
%s/security.ubuntu.com/mirror.kakao.com
```

패키지 및 기존 설치된 소프트웨어 업데이트

```
sudo apt update
```

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

```
sudo passwd
```

root 권한 패스워드 변경

htop

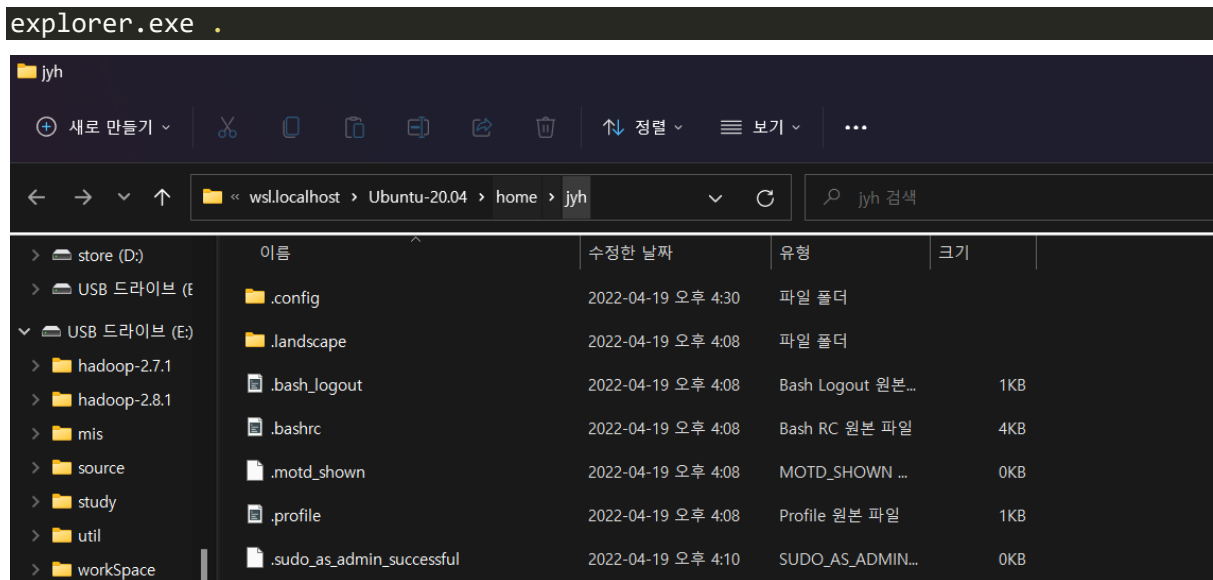
## 자원공유사항

1 [ 0.0%] 5 [ 0.0%]  
2 [ 0.0%] 6 [ 0.0%]  
3 [ 0.7%] 7 [ 0.0%]  
4 [ 0.0%] 8 [ 0.0%]  
Mem [|||||] 961M/15.5G Tasks: 5, 2 thr: 1 running  
Swp [ ] 1.01M/4.00G Load average: 0.00 0.02 0.00  
Uptime: 07:21:04

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
5	root	20	0	1756	1084	1020	S	0.0	0.0	0:00.00	/init
6	root	20	0	1756	1084	1020	S	0.0	0.0	0:00.00	/init
1	root	20	0	1756	1084	1020	S	0.0	0.0	0:00.00	/init
48	root	20	0	2112	348	0	S	0.0	0.0	0:00.00	/init
49	root	20	0	2112	356	0	S	0.0	0.0	0:00.10	/init
50	jyh	20	0	10036	5036	3324	S	0.0	0.0	0:00.02	-bash
3838	jyh	20	0	8156	3696	3040	R	0.0	0.0	0:00.00	htop

## F10 빠져나오기

## 윈도우 탐색기연결 - 파일관리 쉽게 가능



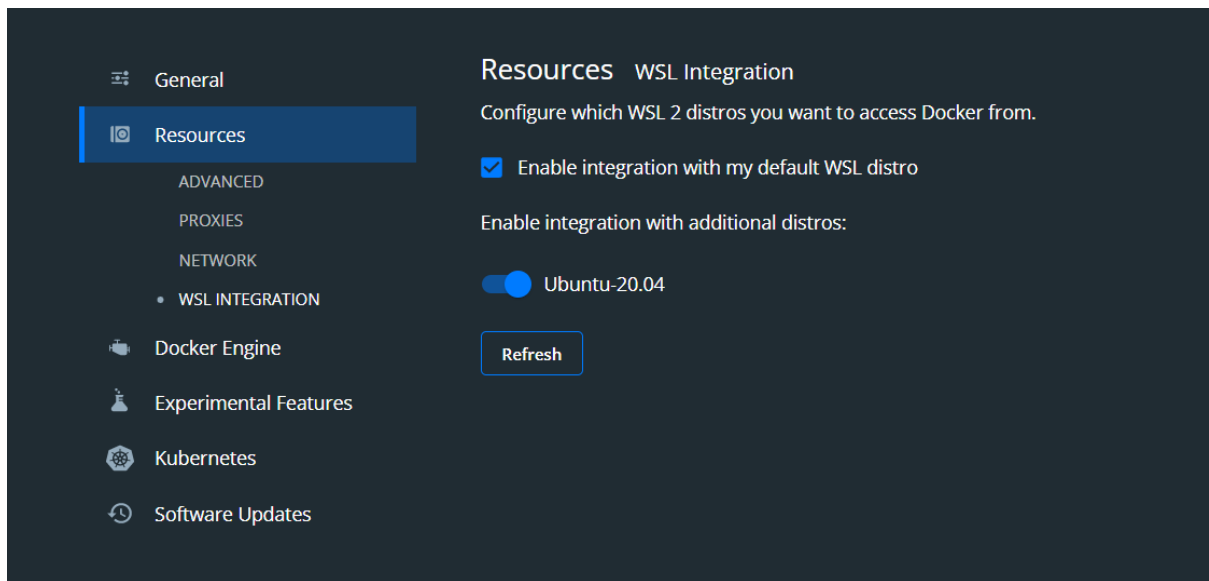
## Vscode 설치

code .

## 도커설치 Docker Desktop 4.7.0 (77141)

윈도우용 도커 먼저 설치후 wsl2 우분투를 설치하여 환경설정이나 옵션이 다르게 체크 되어 있을수 있음.

<https://docs.docker.com/desktop/windows/install/>



```
yh@Thingspire-jyh:~$ docker --version
Docker version 20.10.14, build a224086
yh@Thingspire-jyh:~$ docker-compose --version
docker-compose version 1.29.2, build 5becea4c
```

윈도우에 설치된 wsl2 의 cpu 자원 및 ram 소모량 제한걸기

<https://docs.microsoft.com/en-us/windows/wsl/release-notes#build-18945>

탐색기 - %UserProfile%

.wslconfig 파일생성 - 아래 내용 입력

[wsl2]

memory=4GB

processors=4

swap=4GB

파워셸 실행 window+x a

wsl --shutdown

docker restart

파워셸에서 wsl 입력하면 자동우로 우분투에 접속가능

htop

```
jyh@Thingspire-jyh: ~  
  
1 [ 0.0%] 5 [ 0.0%]  
2 [ 0.0%] 6 [ 0.0%]  
3 [ 0.0%] 7 [ 0.0%]  
4 [ 0.0%] 8 [ 0.0%]  
Mem[|||||] 991M/15.5G Tasks: 10, 26 thr; 1 running  
Swp[|] 1.01M/4.00G Load average: 0.00 0.00 0.00  
Uptime: 07:51:15  
  
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command  
4068 jyh 20 0 8156 3680 3016 R 0.0 0.0 0:00.09 htop  
49 root 20 0 2112 356 0 S 0.0 0.0 0:00.17 /init  
4012 jyh 20 0 747M 39040 28768 S 0.0 0.2 0:00.17 docker serve --address unix:///home/jyh/.docker/run/docke  
4017 jyh 20 0 747M 39040 28768 S 0.0 0.2 0:00.01 docker serve --address unix:///home/jyh/.docker/run/docke  
5 root 20 0 1892 1180 1020 S 0.0 0.0 0:00.05 /init  
6 root 20 0 1892 1180 1020 S 0.0 0.0 0:00.00 /init  
1 root 20 0 1892 1180 1020 S 0.0 0.0 0:00.15 /init  
48 root 20 0 2112 348 0 S 0.0 0.0 0:00.00 /init  
50 jyh 20 0 10036 5036 3324 S 0.0 0.0 0:00.06 -bash  
3996 root 20 0 1892 220 0 S 0.0 0.0 0:00.00 /init  
3997 root 20 0 1892 220 0 S 0.0 0.0 0:00.00 /init  
3999 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4000 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4001 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4002 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4003 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4004 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4005 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
4006 root 20 0 1569M 24204 13136 S 0.0 0.1 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit
```

```
jyh@Thingspire-jyh: ~  
  
1 [ 0.0%] 3 [ 0.0%]  
2 [ 0.0%] 4 [ 0.0%]  
Mem[|||||] 743M/3.84G Tasks: 10, 20 thr; 1 running  
Swp[|] 0K/4.00G Load average: 0.01 0.00 0.00  
Uptime: 00:02:30  
  
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command  
179 jyh 20 0 8156 3664 3008 R 0.0 0.1 0:00.01 htop  
154 jyh 20 0 746M 38452 27872 S 0.0 1.0 0:00.06 docker serve --address unix:///home/jyh/.docker/run/docke  
157 jyh 20 0 746M 38452 27872 S 0.0 1.0 0:00.01 docker serve --address unix:///home/jyh/.docker/run/docke  
143 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.05 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
5 root 20 0 1844 1092 1020 S 0.0 0.0 0:00.00 /init  
6 root 20 0 1844 1092 1020 S 0.0 0.0 0:00.00 /init  
1 root 20 0 1844 1092 1020 S 0.0 0.0 0:00.13 /init  
141 root 20 0 1752 72 0 S 0.0 0.0 0:00.00 /init  
142 root 20 0 1752 80 0 S 0.0 0.0 0:00.00 /init  
144 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
145 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
147 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
146 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
148 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
149 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
150 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
151 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.02 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
152 root 20 0 1352M 33008 12996 S 0.0 0.8 0:00.00 /mnt/wsl/docker-desktop/docker-desktop-user-distro proxy  
153 root 20 0 1752 80 0 S 0.0 0.0 0:00.00 /init  
155 jyh 20 0 746M 38452 27872 S 0.0 1.0 0:00.00 docker serve --address unix:///home/jyh/.docker/run/docke  
156 jyh 20 0 746M 38452 27872 S 0.0 1.0 0:00.00 docker serve --address unix:///home/jyh/.docker/run/docke  
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit
```

<http://dveamer.github.io/backend/DockerImageDirectory.html>

<https://bitlog.tistory.com/48>



<https://www.popit.kr/%EA%B0%9C%EB%B0%9C%EC%9E%90%EA%B0%80-%EC%B2%98%EC%9D%8C->

[docker-%EC%A0%91%ED%95%A0%EB%95%8C-%EC%98%A4%EB%8A%94-%EB%A9%98%EB%B6%95-%EB%AA%87%EA%B0%80%EC%A7%80/](#)

## Docker 는 Virtual machine 이 아니다!

이 문구로 유추해보면 우분투 컨테이너를 실행하면 우분투 서버가 실행되는 것이 아니라 `/bin/bash` 가 실행되는 것 뿐이다. 이것이 **Virtual machine** 의 컨테이너와 **Docker** 컨테이너의 가장 큰 차이점이다. 일반적으로 **Linux** 서버(**Ubuntu** or **CentOS** 등)나 **Windows** 와 같은 운영 체제를 실행한다는 의미에는 많은 것을 내포하고 있다. 대략 다음과 같은 기능들이 실행될 것이다.

- 프로그램 실행 기능
  - **Memory, CPU** 등의 하드웨어 자원을 이용하여 프로그램을 실행할 수 있는 환경 제공
- 네트워크 서비스 제공
  - **NIC** 등을 하드웨어 자원을 인식해서 네트워크 처리가 가능한 환경을 제공
- 키보드, 모니터, 마우스 등과 같은 주변 장치의 입출력에 대한 처리
  - 사용자로부터의 입력과 결과를 출력해주는 기능 제공
- 외부에서 접속할 수 있는 환경
  - **sshd** 등과 같은 데몬을 실행하여 서버 외부에서 네트워크를 이용하여 원격에서 접속할 수 있는 기능 제공

그리고 이 모든 것은 사용자가 임의로 전원을 끄기 전에는 지속적으로 동작하는 특징을 가지고 있다. 흔히 **Virtual machine** 이라고 하는 컨테이너 들은 이런 속성을 가지고 있다. 다만 여러 하드웨어 자원을 **Host OS** 로 부터 할당 받은 것만 사용하도록 되어 있는 것이다.

*Docker 컨테이너는 단지 명령만 실행하고 그 결과만 보여주는 기능을 수행한다.<sup>[1]</sup>*

## 우분투 maria db 설치

### docker search maria

```
C:\Users\salva>docker search maria
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
mariadb	MariaDB Server is a high performing open sou...	4790	[OK]	
phpmyadmin	phpMyAdmin - A web interface for MySQL and M...	509	[OK]	
linuxserver/mariadb	A Mariadb container, brought to you by Linux...	297		
bitnami/mariadb	Bitnami MariaDB Docker Image	156		[OK]
mariadb/server	MariaDB Server is a modern database for mode...	40		[OK]
jic21/mariadb-aria	Extension of the mariadb image that forces a...	26		
mariadb/maxscale	MariaDB MaxScale - The world's most advanced...	19		[OK]
bitnami/mariadb-galera		17		
centos/mariadb-101-centos7	MariaDB 10.1 SQL database server	13		
marian/rebrow	Web UI for redis content	12		[OK]
centos/mariadb-102-centos7	MariaDB 10.2 SQL database server	6		
circleci/mariadb	CircleCI images for MariaDB	4		[OK]
mariadb/columnstore	Official MariaDB ColumnStore community image	3		
ibmcom/mariadb	Docker Image for IBM Cloud Private-CE (Commu...	1		
ibmcom/mariadb-ppc64le	Docker Image for IBM Cloud Private-CE (Commu...	1		
vmware/mariadb-photon		0		
intel/nmt_marian_framework_demo	Docker Image for open-source Neural Machine ...	0		

To get more help with docker, check out our guides at <https://docs.docker.com/go/guides/>

```
jyh@Thingspire-jyh:~$ docker search maria
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
mariadb	MariaDB Server is a high performing open sou...	4790	[OK]	
phpmyadmin	phpMyAdmin - A web interface for MySQL and M...	509	[OK]	
linuxserver/mariadb	A Mariadb container, brought to you by Linux...	297		
bitnami/mariadb	Bitnami MariaDB Docker Image	156		[OK]
mariadb/server	MariaDB Server is a modern database for mode...	40		[OK]
jic21/mariadb-aria	Extension of the mariadb image that forces a...	26		
mariadb/maxscale	MariaDB MaxScale - The world's most advanced...	19		[OK]
bitnami/mariadb-galera		17		
centos/mariadb-101-centos7	MariaDB 10.1 SQL database server	13		
marian/rebrow	Web UI for redis content	12		[OK]
centos/mariadb-102-centos7	MariaDB 10.2 SQL database server	6		
circleci/mariadb	CircleCI images for MariaDB	4		[OK]
mariadb/columnstore	Official MariaDB ColumnStore community image	3		
ibmcom/mariadb	Docker Image for IBM Cloud Private-CE (Commu...	1		
ibmcom/mariadb-ppc64le	Docker Image for IBM Cloud Private-CE (Commu...	1		
vmware/mariadb-photon		0		
intel/nmt_marian_framework_demo	Docker Image for open-source Neural Machine ...	0		
ibmcom/mariadb-centos-ppc64le		0		
ibmcom/mariadb-centos-s390x		0		
mariadb/docker-test		0		

### docker pull mariadb

### sudo su

```
docker run -d --name mariadb -p 3306:3306 -v Desktop:/var/lib/mysql -e MARIADB_ROOT_PASSWORD=vup -e MARIADB_DATABASE=vup mariadb
```

### run : 실행

-d : 백그라운드 계속 실행

--name mariadb : 컨테이너 이름

-p 3306:3306 : 호스트포트 와 컨테이너 포트 연결

-v Desktop:/var/lib/mysql : bind volume , 호스트폴더:컨테이너폴더

MARIADB\_ROOT\_PASSWORD=vup : 패스워드 vup

-e MARIADB\_DATABASE=vup : vup 라는 이름의 database

Mariadb : 이 이미지를 이용하여 컨테이너 생성

Docker ps

Docker start mariadb

docker exec -it mariadb bash

구동중인 컨테이너에 접속 bash shell 을 사용하여

docker exec -it mariadb bash

mariadb -uroot -pvup

```
root@76779f267d9d: /
Microsoft Windows [Version 10.0.22000.613]
(c) Microsoft Corporation. All rights reserved.

C:\Users\salva>docker exec -it mariadb /bin/bash
root@76779f267d9d:/# mysql -u root -p ***
```

그외 mysql 컨테이너 생성방법

docker-compose.yml 파일 생성

```
version: "3" # 파일 규격 버전
services: # 이 항목 밑에 실행하려는 컨테이너 들을 정의
  db: # 서비스 명
    image: mysql # 사용할 이미지
    container_name: custom_mysql # 컨테이너 이름 설정
    ports:
      - "3306:3306" # 접근 포트 설정 (컨테이너 외부:컨테이너 내부)
    environment: # -e 옵션
      MYSQL_ROOT_PASSWORD: "password" # MYSQL 패스워드 설정 옵션
    command: # 명령어 실행
      - --character-set-server=utf8mb4
      - --collation-server=utf8mb4_unicode_ci
    volumes:
      - /Users/jmlim/datadir:/var/lib/mysql # -v 옵션 (다렉토리 마운트 설정)
```

실행 docker-compose 파일 실행

- docker-compose.yml 작성한 위치에서 실행

docker-compose up -d

## 5. 컨테이너 접속

`docker exec -it custom_mysql bash`

## 6. mysql 접속 및 계정 생성하기

6-1. mysql 접속 : `mysql -u root -p 비밀번호`

6-2. 계정 생성 : `CREATE USER '계정명'@'%' IDENTIFIED BY '비밀번호';`

6-3. 계정 권한 부여 : `GRANT ALL PRIVILEGES ON *.* IDENTIFIED BY '비밀번호';`

6-4. 변경 내용 적용 : `FLUSH PRIVILEGES;`

6-5. mysql 종료 : `QUIT`

6-6. 컨테이너 나가기 : `exit`

```
MariaDB [(none)]> create user 'vup'@'%' identified by 'vup';
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]> grant all privileges on vup.* to 'vup'@'%;
Query OK, 0 rows affected (0.009 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.000 sec)
```

`create user 'vup'@'%' identified by 'vup';`

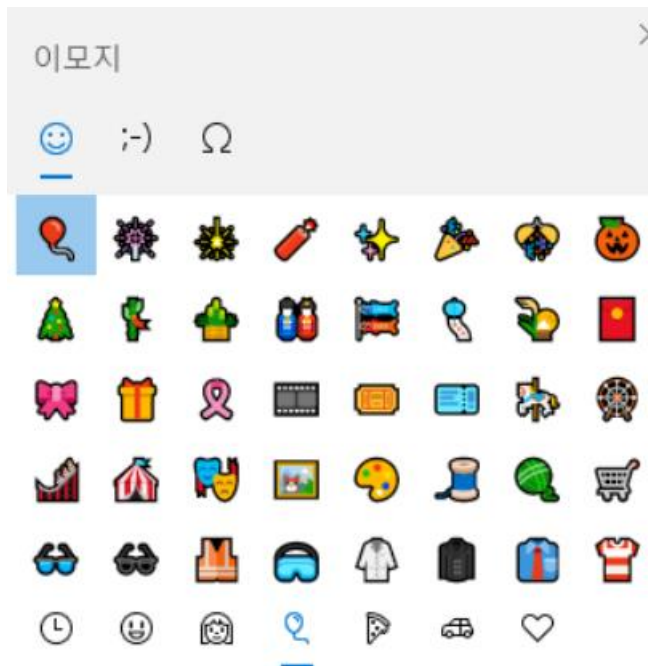
`grant all privileges on vup.* to 'vup'@'%;`

`flush privileges;`

## character set 설정

UTF-8의 경우 4바이트의 인코딩인데, MySQL, MariaDB의 경우 utf8이 3바이트로 구현

유니코드 U+10000 이상의 문자는 UTF-8로 인코딩을 하려면 4바이트가 필요한데, 최근 많이 사용되고 있는 이모지가 그 유형



status

```
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> status
mysql Ver 15.1 Distrib 10.7.3-MariaDB, for debian-linux-gnu (x86_64) using readline 5.2

Connection id:          10
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:           ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.7.3-MariaDB-1:10.7.3+maria-focal mariadb.org binary distribution
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    utf8mb4
Db characterset:        utf8mb4
Client characterset:    latin1
Conn. characterset:     latin1
UNIX socket:            /run/mysqld/mysqld.sock
Uptime:                 28 min 14 sec

Threads: 4  Questions: 110  Slow queries: 0  Opens: 18  Open tables: 11  Queries per second avg: 0.064

MariaDB [(none)]>
```

exit

vim

vim 설치

cat /etc/\*-release

```

root@76779f267d9d:/etc/mysql# nano my.cnf
bash: nano: command not found
root@76779f267d9d:/etc/mysql# cat /etc/*-release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=20.04
DISTRIB_CODENAME=focal
DISTRIB_DESCRIPTION="Ubuntu 20.04.4 LTS"
NAME="Ubuntu"
VERSION="20.04.4 LTS (Focal Fossa)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 20.04.4 LTS"
VERSION_ID="20.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=focal
UBUNTU_CODENAME=focal
root@76779f267d9d:/etc/mysql#

```

apt update && apt install nano && apt install vim

nano /etc/mysql/my.cnf

[client]

default-character-set = utf8mb4

[mysql]

default-character-set = utf8mb4

[mysqldump]

default-character-set = utf8mb4

[mysqld]

character-set-server=utf8mb4

collation-server=utf8mb4\_unicode\_ci

skip-character-set-client-handshake

저장후

exit

docker restart mariadb

docker exec -it mariadb /bin/bash

mysql -u root -p

status

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> status
-----
mysql Ver 15.1 Distrib 10.7.3-MariaDB, for debian-linux-gnu (x86_64) using readline 5.2

Connection id:          5
Current database:
Current user:            root@localhost
SSL:                     Not in use
Current pager:           stdout
Using outfile:           ''
Using delimiter:         ;
Server:                  MariaDB
Server version:          10.7.3-MariaDB-1:10.7.3+maria~focal mariadb.org binary distribution
Protocol version:        10
Connection:              Localhost via UNIX socket
Server characterset:     utf8mb4
Db characterset:         utf8mb4
Client characterset:     utf8mb4
Conn. characterset:      utf8mb4
UNIX socket:              /run/mysqld/mysqld.sock
Uptime:                  2 min 16 sec

Threads: 1 Questions: 4 Slow queries: 0 Opens: 17 Open tables: 10 Queries per second avg: 0.029
-----
```

원격접속 ssh 터널 생성

```
ssh -N -L 13306:127.0.0.1:3306 vup@175.123.142.155
```

ssh: connect to host 175.123.142.155 port 22: Connection refused

175.123.142.155:13306

```
mysql -h 175.123.142.155 -u vup -p
```

```
ssh -CNf -L3306:127.0.0.1:3306 vup@175.123.142.155
```

외부에서 3306 으로 접속시 내부 175.123.142.155:13306 으로 접속한다는 의미

```
Mysql -u vup -p -h 127.0.0.1 -p 3306
```

```
netstat -nltp
```

127.0.0.1:3306 포트 열려있음

```
mysql -u vup -p -h 127.0.0.1 -p 3306
```

Enter password:

ERROR 1044 (42000): Access denied for user 'vup'@'%' to database '3306'

ERROR 1396 (HY000): Operation CREATE USER failed for

drop user 'vup'@'%';

create user vup@'%' identified by 'vup';

grant all privileges on vup.\* to vup@'localhost';

grant all privileges on vup.\* to vup@'%';


flush privileges;

Connection "vup\_gis" configuration

**Connection settings**

MariaDB connection settings

— □ ×



▼ Connection settings

- Initialization
- Shell Command
- Client identification
- Transactions
- General
- Metadata
- Errors and timeouts
- > Data editor
- > SQL Editor

Main Driver properties SSH Proxy SSL

Server

Server Host: localhost Port: 3306

Database: vup

Authentication (Database Native)

Username: vup

Password: ●●● ☒ Save password locally

Advanced

Server Time Zone: Auto-detect

Local Client: MySQL Binaries

① You can use variables in connection parameters.

Driver name: MariaDB [Edit Driver Settings](#)

Test Connection ...

확인 취소



## Connection settings

MariaDB connection settings



✓ Connection settings

Initialization

Shell Command

Client identification

Transactions

General

Metadata

Errors and timeouts

> Data editor


> SQL Editor

MainDriver propertiesSSHProxySSL

☒ Use SSH 터널

Profile: 

▼



Settings

Host/IP:175.123.142.155Port:2222

User Name:yhjung

Authentication Method:Password ▼

Password

●●●●●●●●

☒ Save Password

▶ Jump server settings

▶ Advanced settings

Test tunnel configuration

You can use parameters. [SSH Documentation](#)

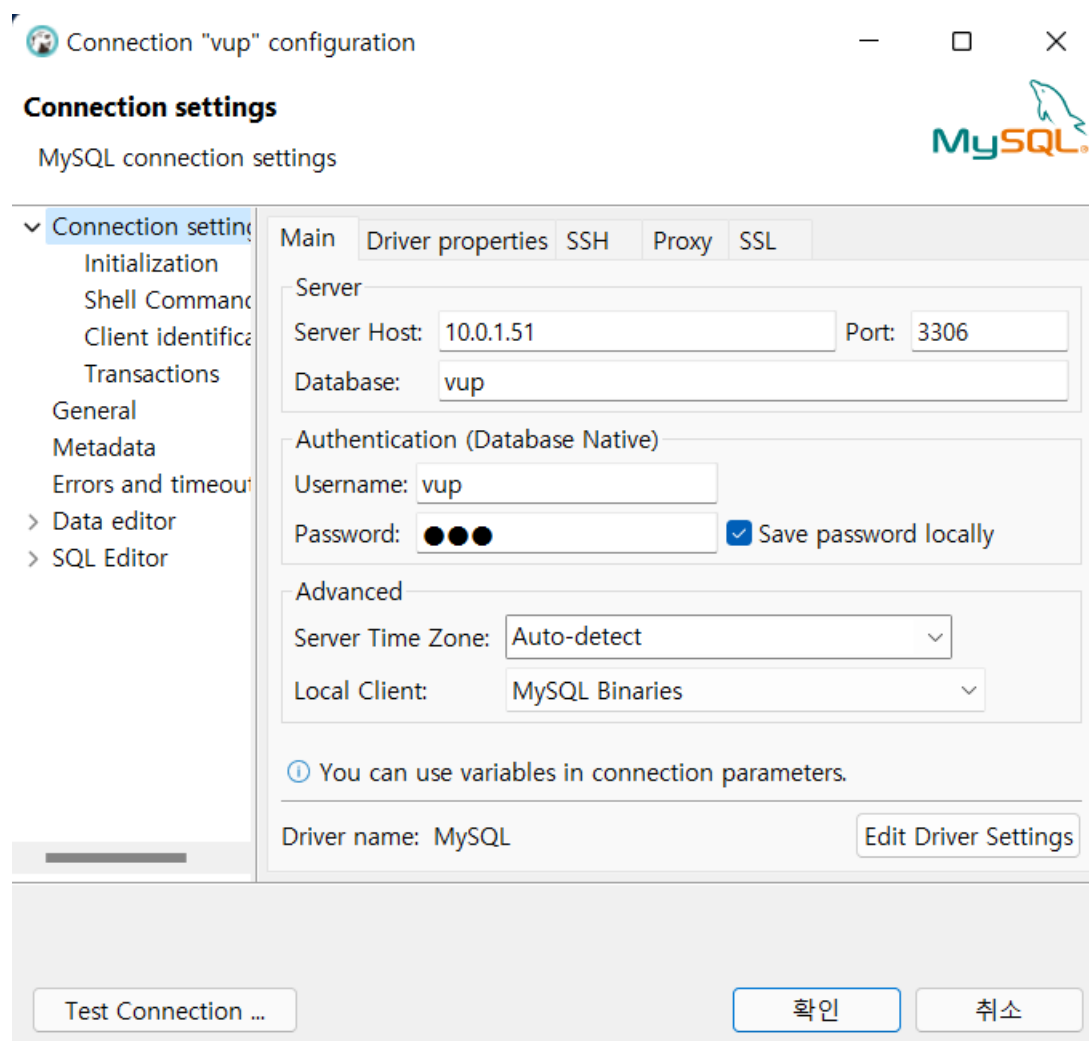
Test Connection ...

확인

취소



bash shell 빠져나올때 ctrl+p , ctrl+q 이해야 컨테이너가 종료안됨, 단 계속 실행 옵션을 미리 해놔기 때문에 필요없음



docker version

설치확인 - 도커 버전

```
D:\study\docker>docker version
Client:
 Cloud integration: v1.0.23
 Version:          20.10.14
 API version:      1.41
 Go version:       go1.16.15
 Git commit:       a224086
 Built:            Thu Mar 24 01:53:11 2022
 OS/Arch:          windows/amd64
 Context:          default
 Experimental:     true

Server: Docker Desktop 4.7.0 (77141)
Engine:
 Version:          20.10.14
 API version:      1.41 (minimum version 1.12)
 Go version:       go1.16.15
 Git commit:       87a90dc
 Built:            Thu Mar 24 01:46:14 2022
 OS/Arch:          linux/amd64
 Experimental:     false
containerd:
 Version:          1.5.11
 GitCommit:        3df54a852345ae127d1fa3092b95168e4a88e2f8
```

docker pull ubuntu

도커허브에서 ubuntu이미지 가져오기

```
D:\study\docker>docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
e0b25ef51634: Already exists
Digest: sha256:9101220a875cee98b016668342c489ff0674f247f6ca20dfc91b91c0f28581ae
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

docker images

현재 설치된 이미지 확인

```
D:\study\docker>docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest    825d55fb6340   13 days ago   72.8MB
```

docker run --name ubuntu-server ubuntu

ubuntu 이미지로 ubuntu-server 의 이름을 가지는 이미지 생성

```
D:\study\docker>docker run --name ubuntu-server ubuntu
D:\study\docker>
```

docker start ubuntu-server

컨테이너 실행

```
D:\study\docker>docker start ubuntu-server
ubuntu-server
```