

MariaDB를 이용한 데이터베이스 연결 프로그램

작성자: 정인혜



INDEX

01 / CentOS / VirtualBox

02 / VirtualBox Install

03 / CentOS Install

04 / IP Address Static (고정아이피 설정)

05 / SoftWare Install

06 / CRUD 사용 가이드



1.

CentOs / VirtualBox





CentOS

여러가지 리눅스 os종류가 있지만 그 중 서버가 안정적이게 구동되는 CentOS를 사용한다.

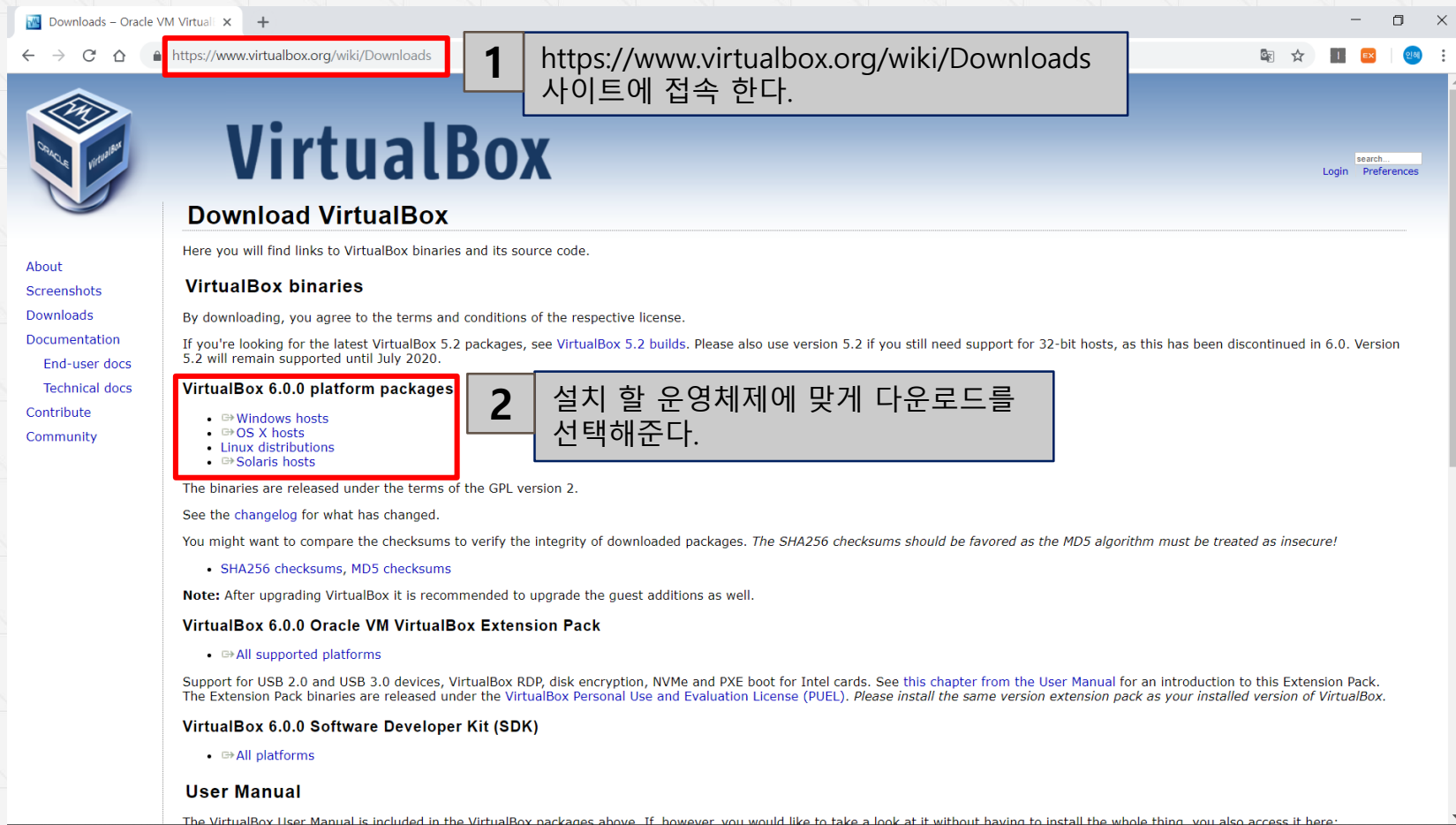


가상머신은 VirtualBox를 사용 할 것이다.



2. **VirtualBox Install**





1 <https://www.virtualbox.org/wiki/Downloads> 사이트에 접속 한다.

VirtualBox

Download VirtualBox

Here you will find links to VirtualBox binaries and its source code.

VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 5.2 packages, see [VirtualBox 5.2 builds](#). Please also use version 5.2 if you still need support for 32-bit hosts, as this has been discontinued in 6.0. Version 5.2 will remain supported until July 2020.

2 설치 할 운영체제에 맞게 다운로드를 선택해준다.

VirtualBox 6.0.0 platform packages

- Windows hosts
- OS X hosts
- Linux distributions
- Solaris hosts

The binaries are released under the terms of the GPL version 2.

See the [changelog](#) for what has changed.

You might want to compare the checksums to verify the integrity of downloaded packages. *The SHA256 checksums should be favored as the MD5 algorithm must be treated as insecure!*

- SHA256 checksums, MD5 checksums

Note: After upgrading VirtualBox it is recommended to upgrade the guest additions as well.

VirtualBox 6.0.0 Oracle VM VirtualBox Extension Pack

- All supported platforms

Support for USB 2.0 and USB 3.0 devices, VirtualBox RDP, disk encryption, NVMe and PXE boot for Intel cards. See [this chapter from the User Manual](#) for an introduction to this Extension Pack. The Extension Pack binaries are released under the [VirtualBox Personal Use and Evaluation License \(PUEL\)](#). Please install the same version extension pack as your installed version of VirtualBox.

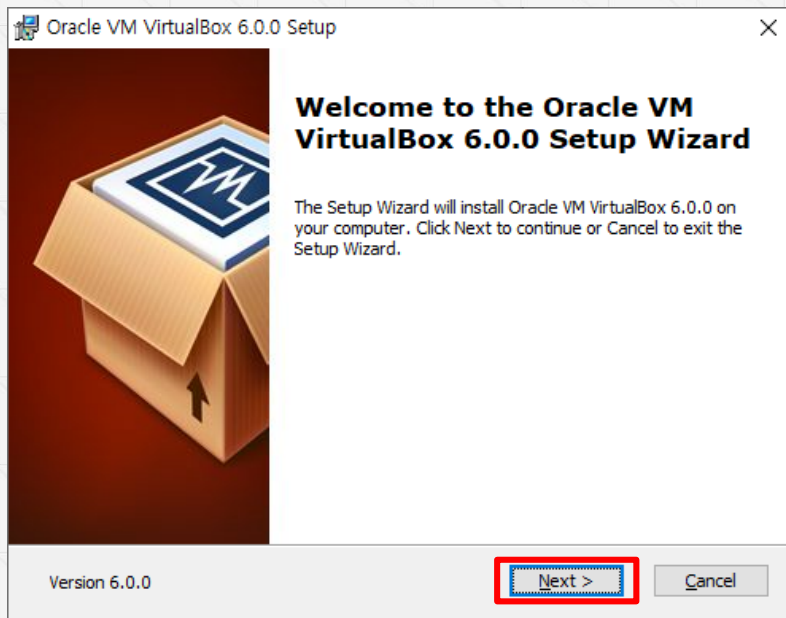
VirtualBox 6.0.0 Software Developer Kit (SDK)

- All platforms

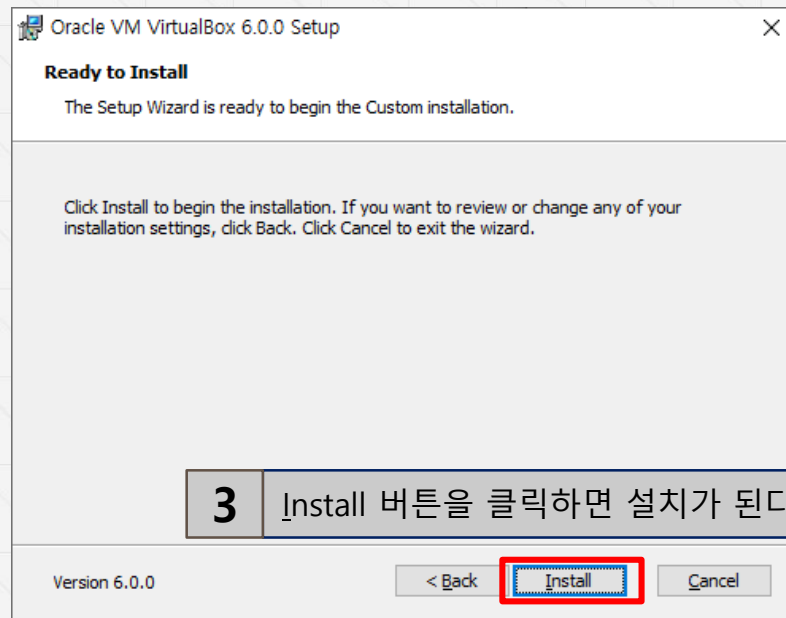
User Manual

The VirtualBox User Manual is included in the VirtualBox packages above. If, however, you would like to take a look at it without having to install the whole thing, you also access it here:

1 다운로드 된 파일을 실행시킨다.



2 Next > 를 계속 클릭해준다.



3 Install 버튼을 클릭하면 설치가 된다.



3.

CentOs Install



The screenshot shows the CentOS download page in a web browser. The address bar is highlighted with a red box and labeled '1'. The 'DVD ISO' button is highlighted with a red box and labeled '2'. The page content includes a welcome message, download options, and links to cloud/container images.

1 <https://www.centos.org/download/> 사이트에 접속 한다.

Download CentOS

As you download and use CentOS Linux, the CentOS Project invites you to [be a part of the community as a contributor](#). There are many ways to contribute to the project, from documentation, QA, and testing to coding changes for [SIGs](#), providing mirroring or hosting, and helping other users.

2 **DVD ISO** CentOS 를 VrtualBox 에 올리기 위해서는 광학 디스크의 압축파일인 iso가 필요하므로 DVD ISO 파일을 다운받아준다.

ISOs are also available [via Torrent](#)

How to [verify](#) your iso

If the above is not for you, [alternative downloads](#) might be.

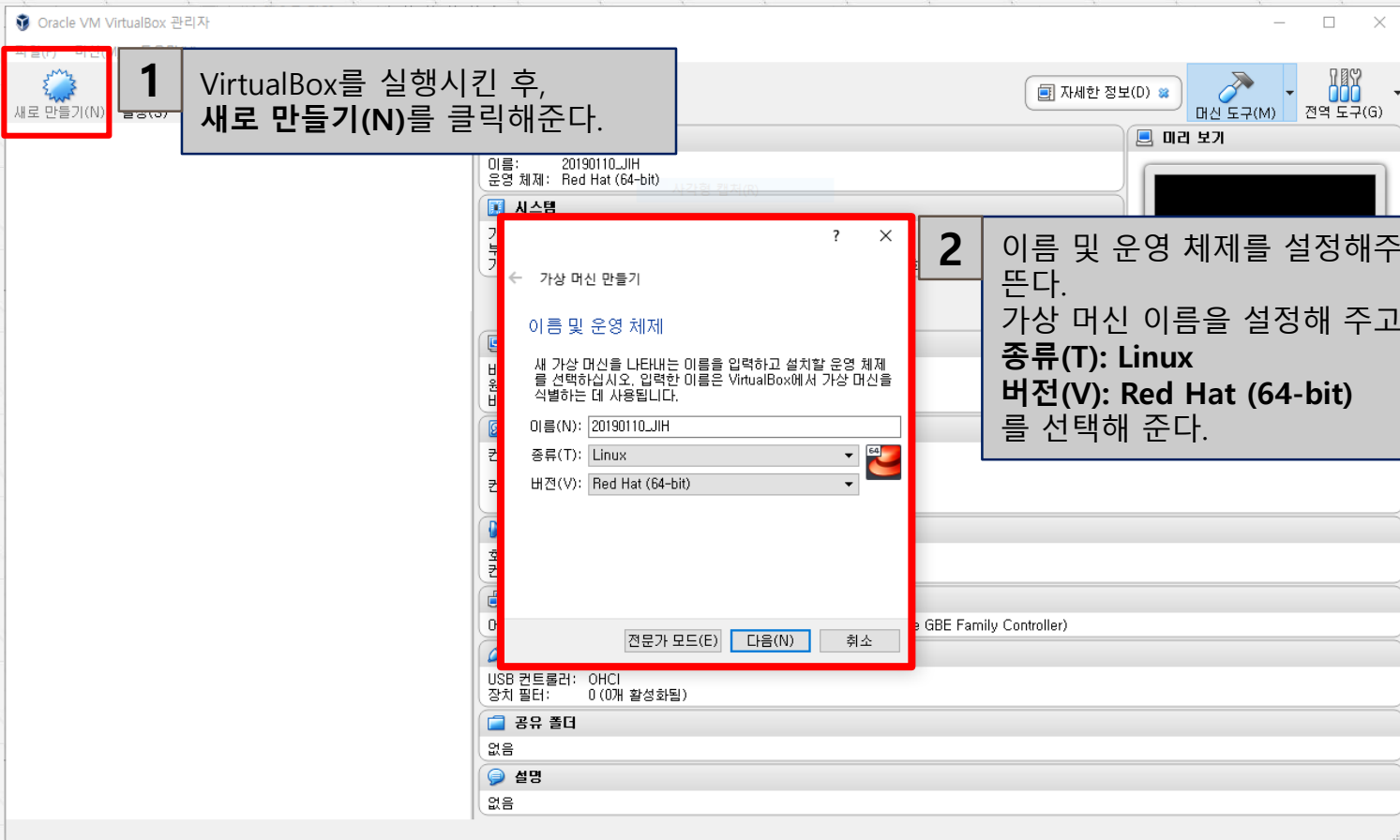
The [release notes](#) are continuously updated to include issues and incorporate feedback from users.

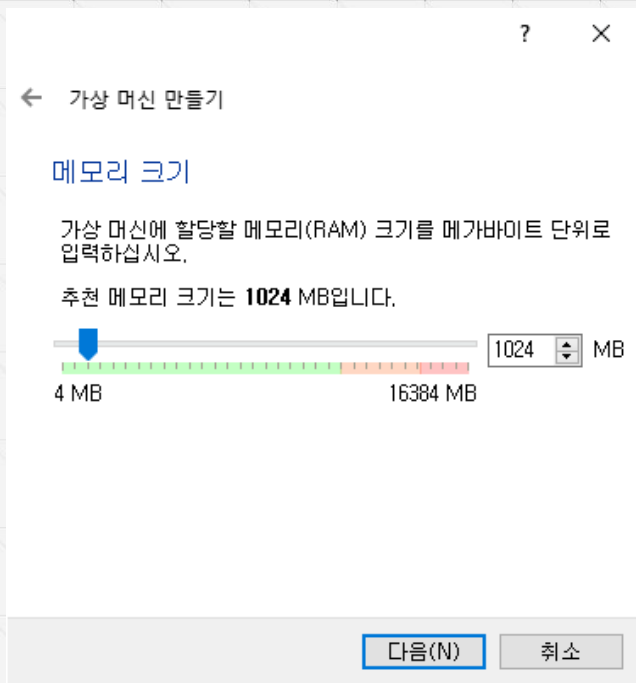
Need a Cloud or Container Image?

[Amazon Web Services](#)

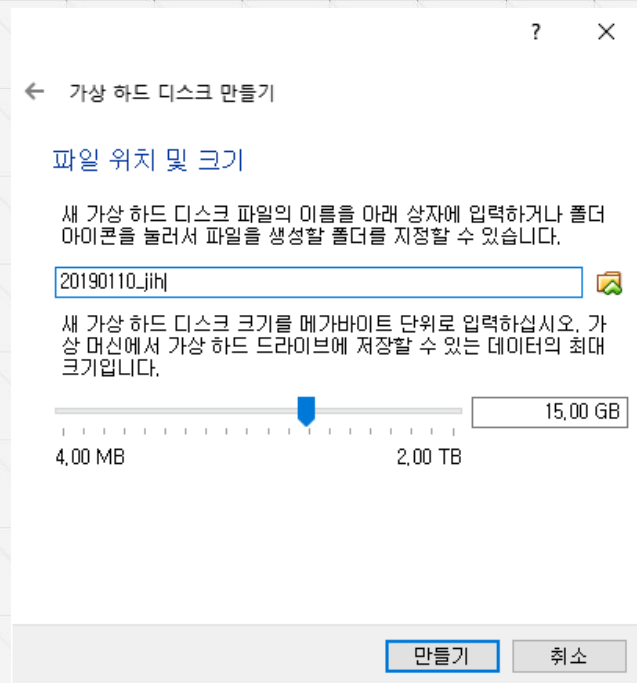
[Docker registry](#)

[More download choices](#)

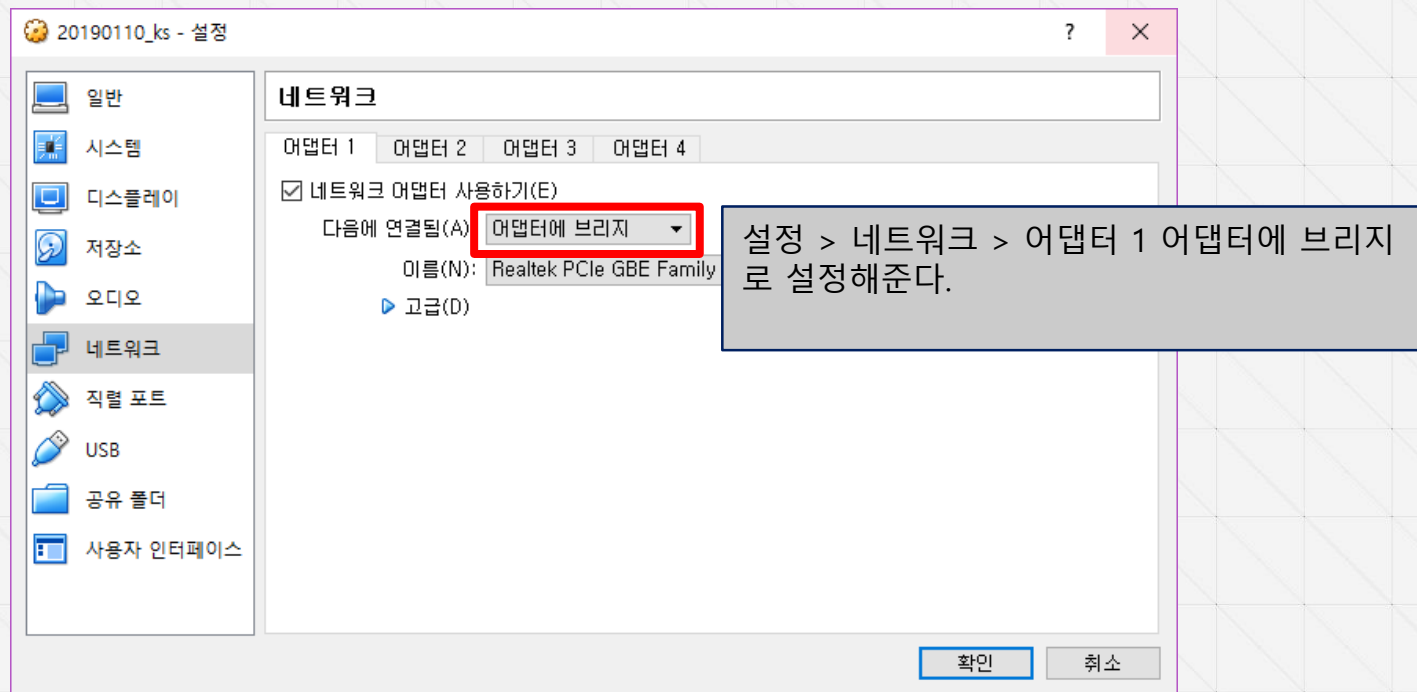


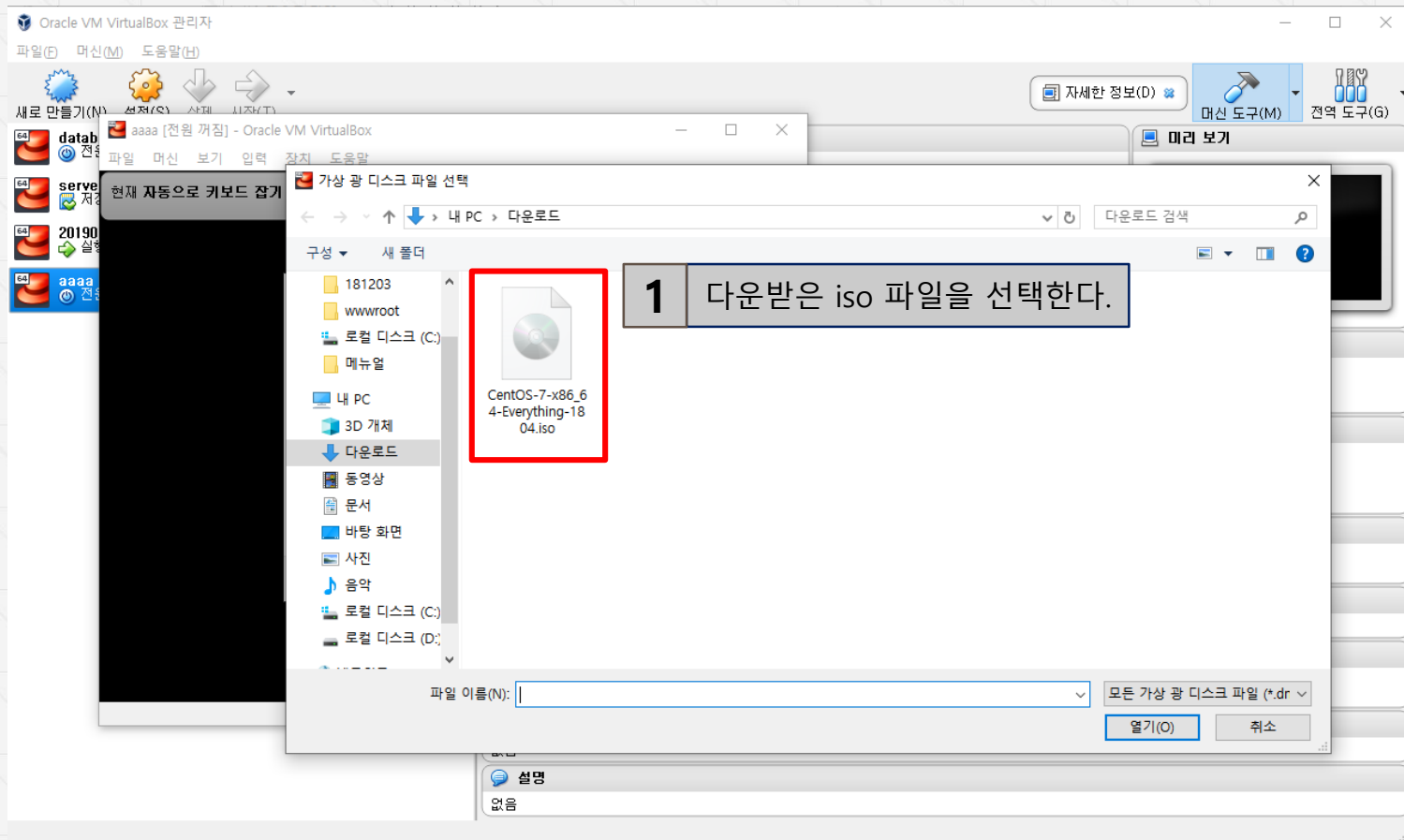


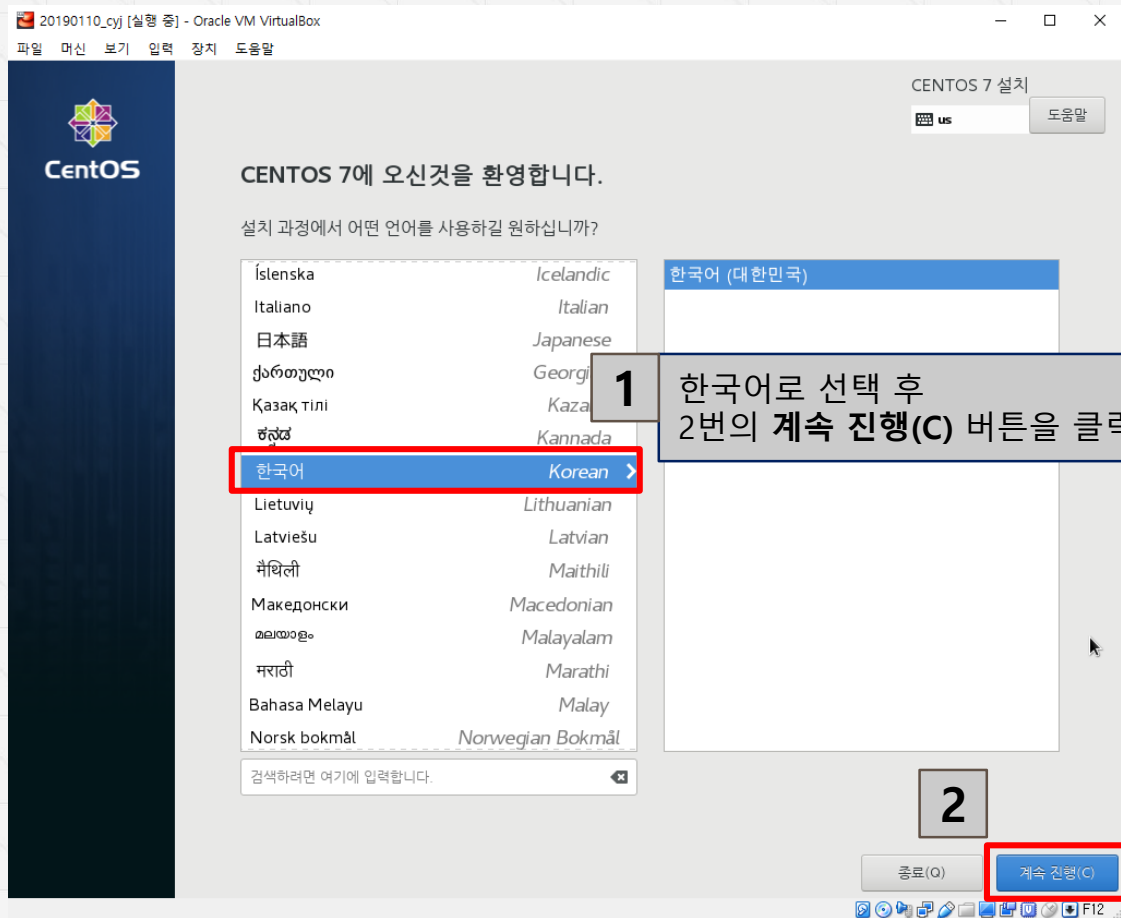
원하는 메모리 크기(RAM) 크기를 설정해준다.

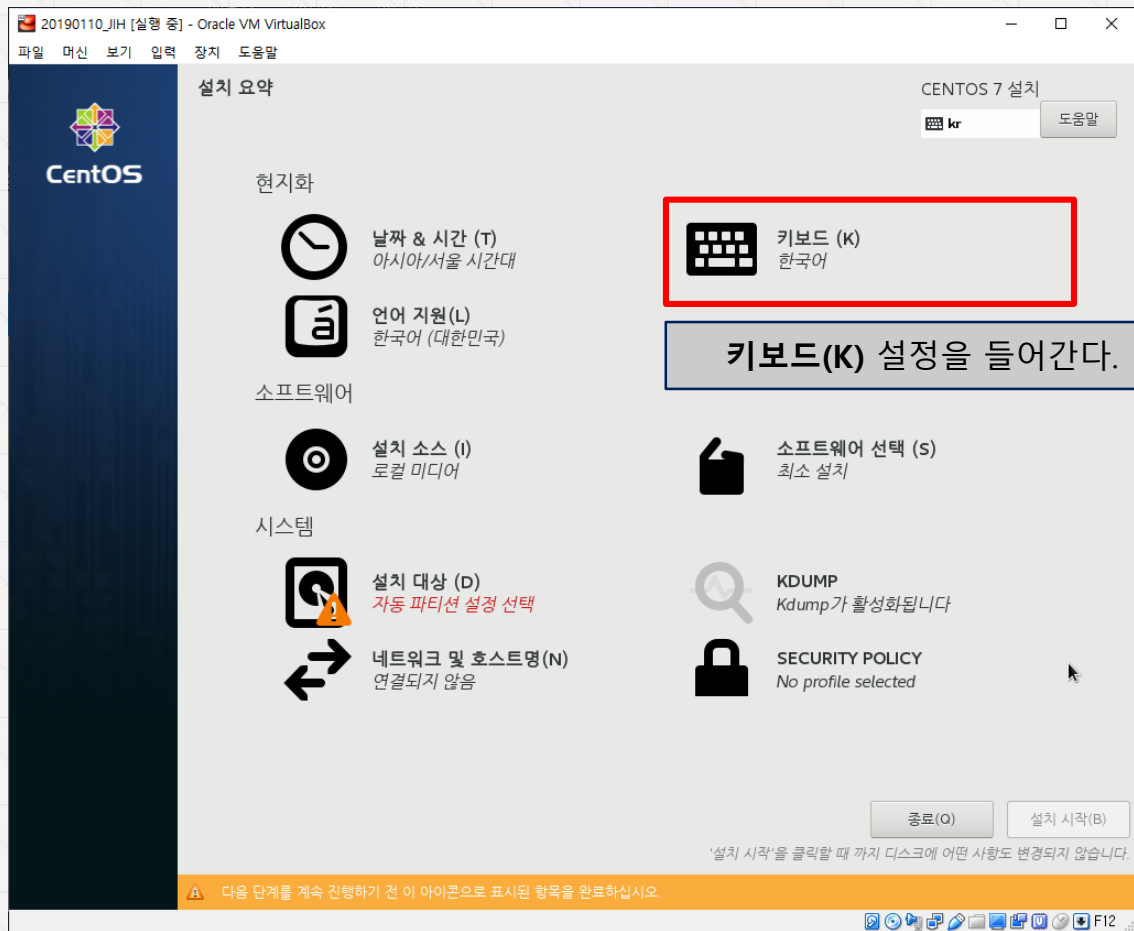


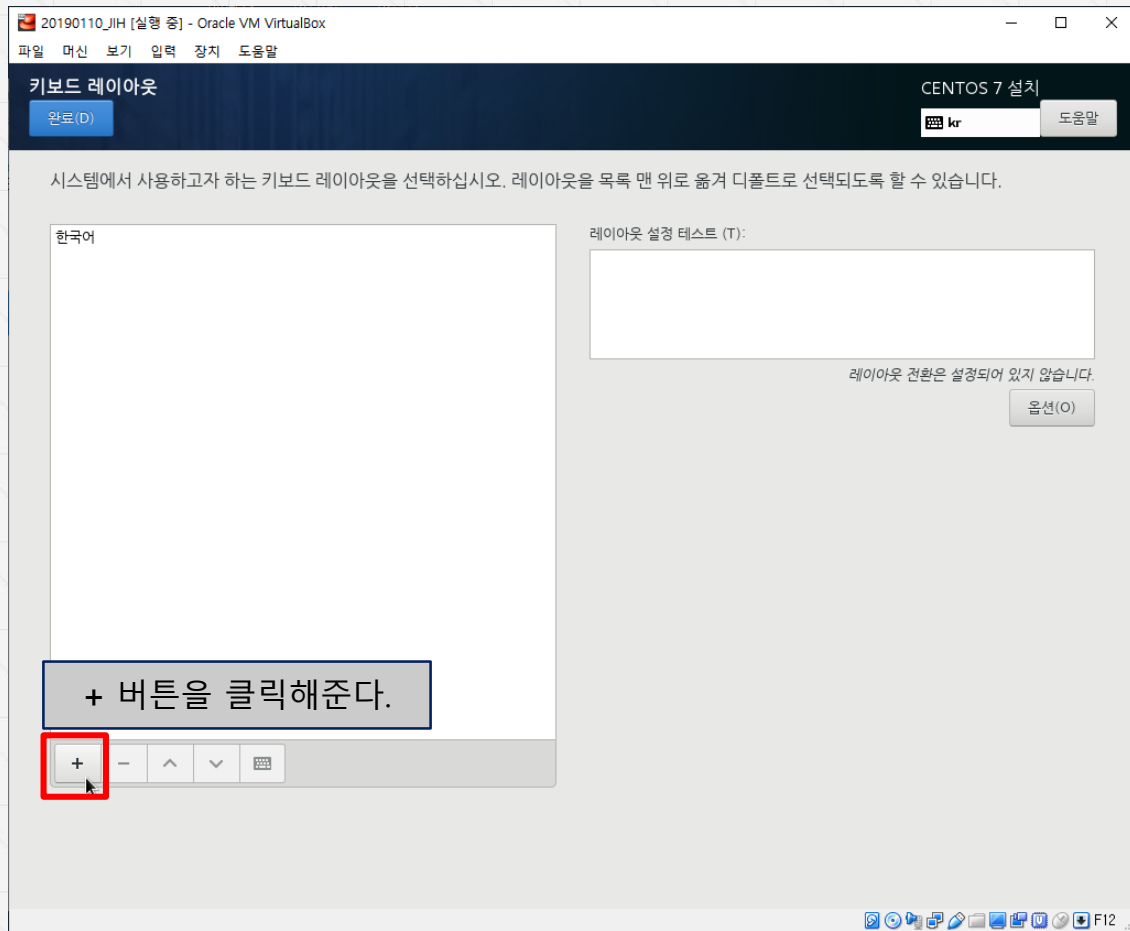
전체 가상 하드 디스크의 크기를 설정해준다.

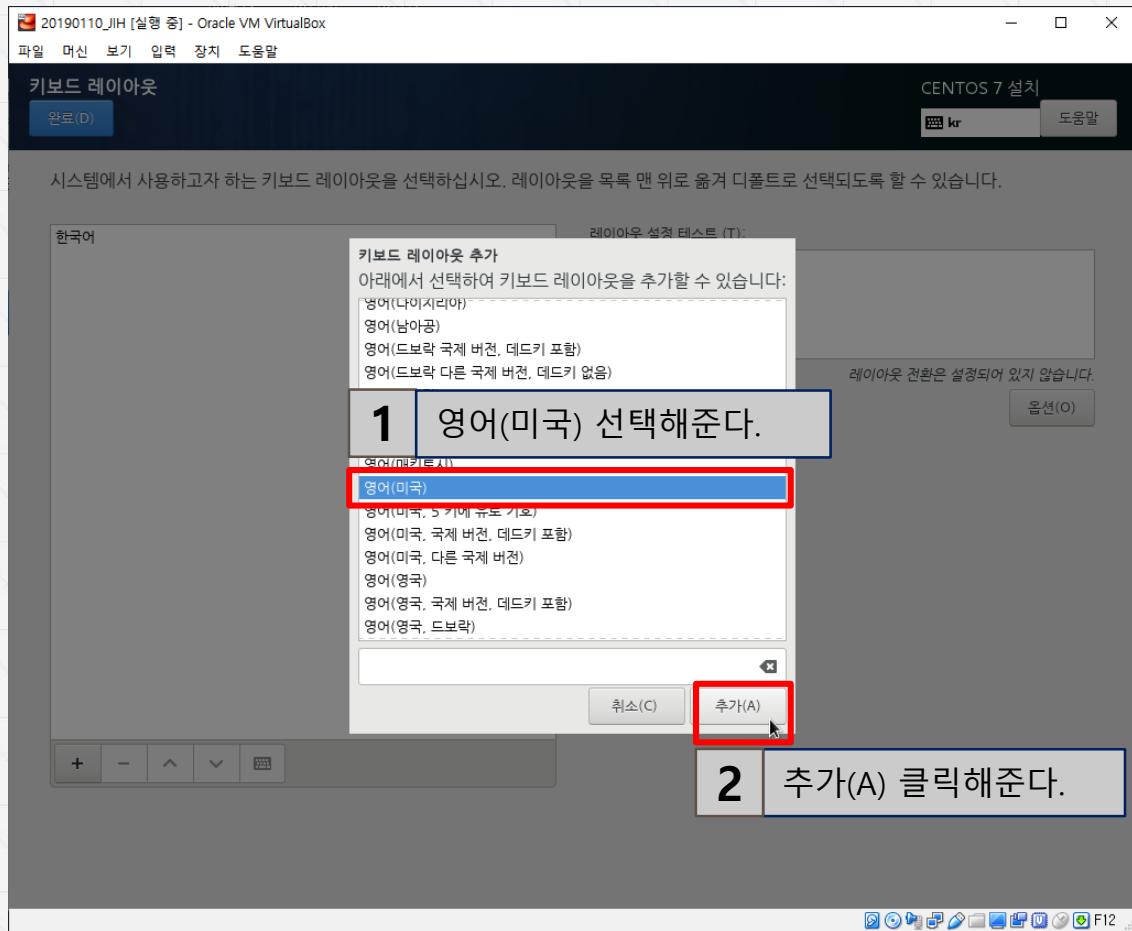


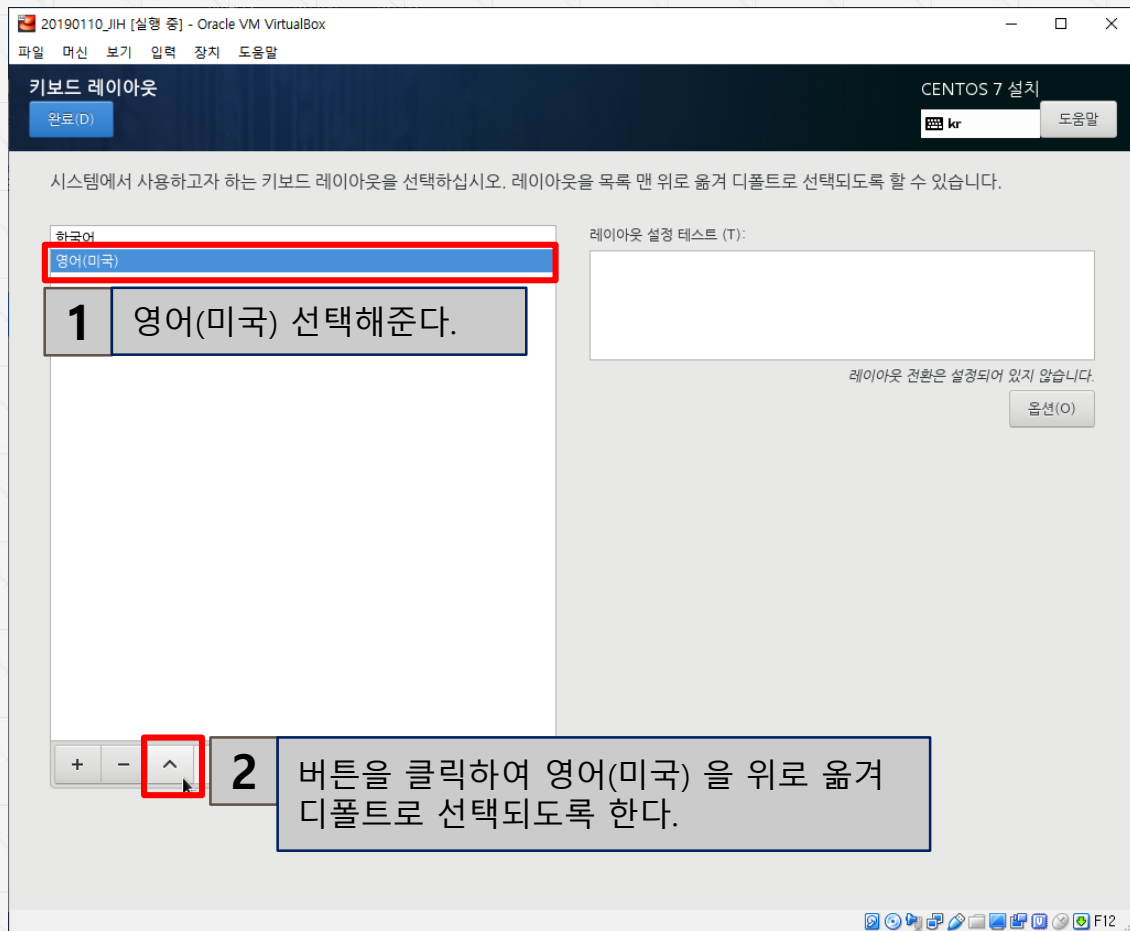


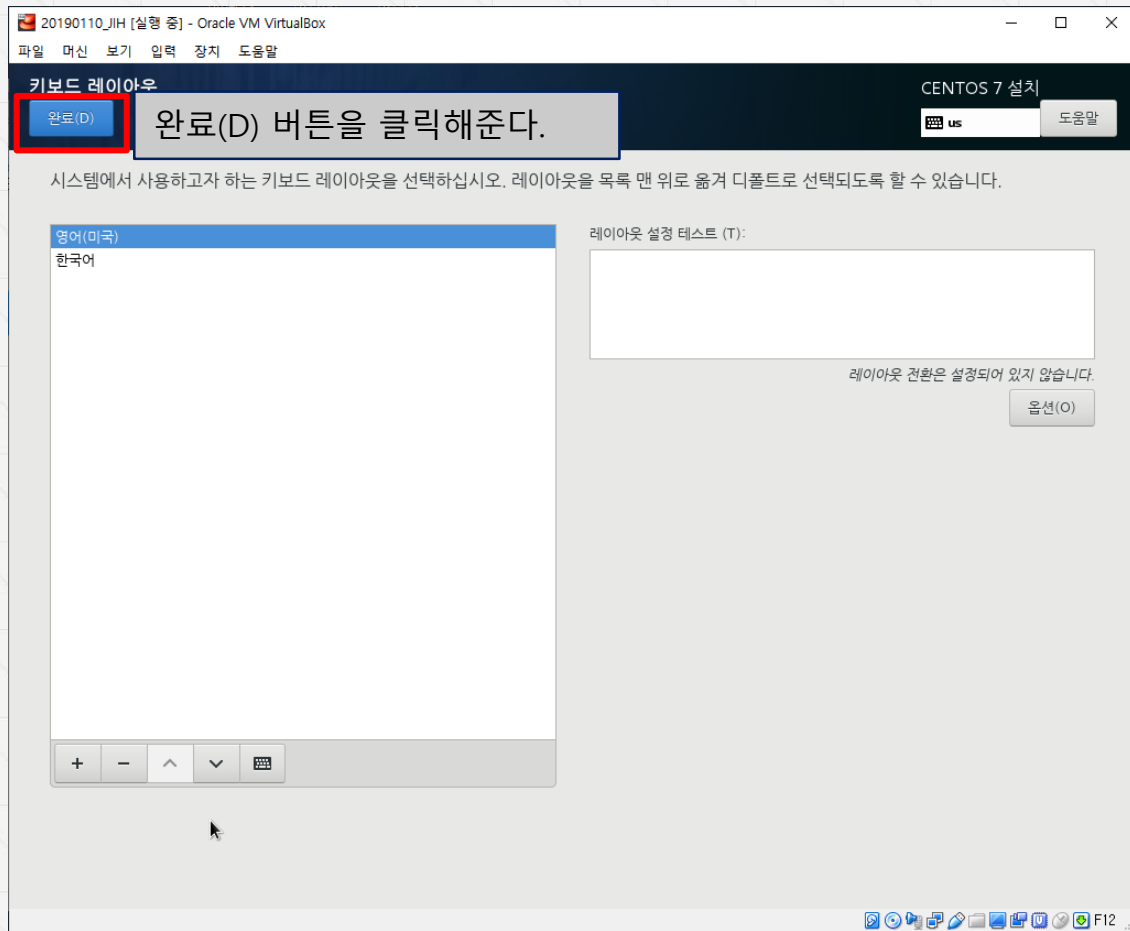


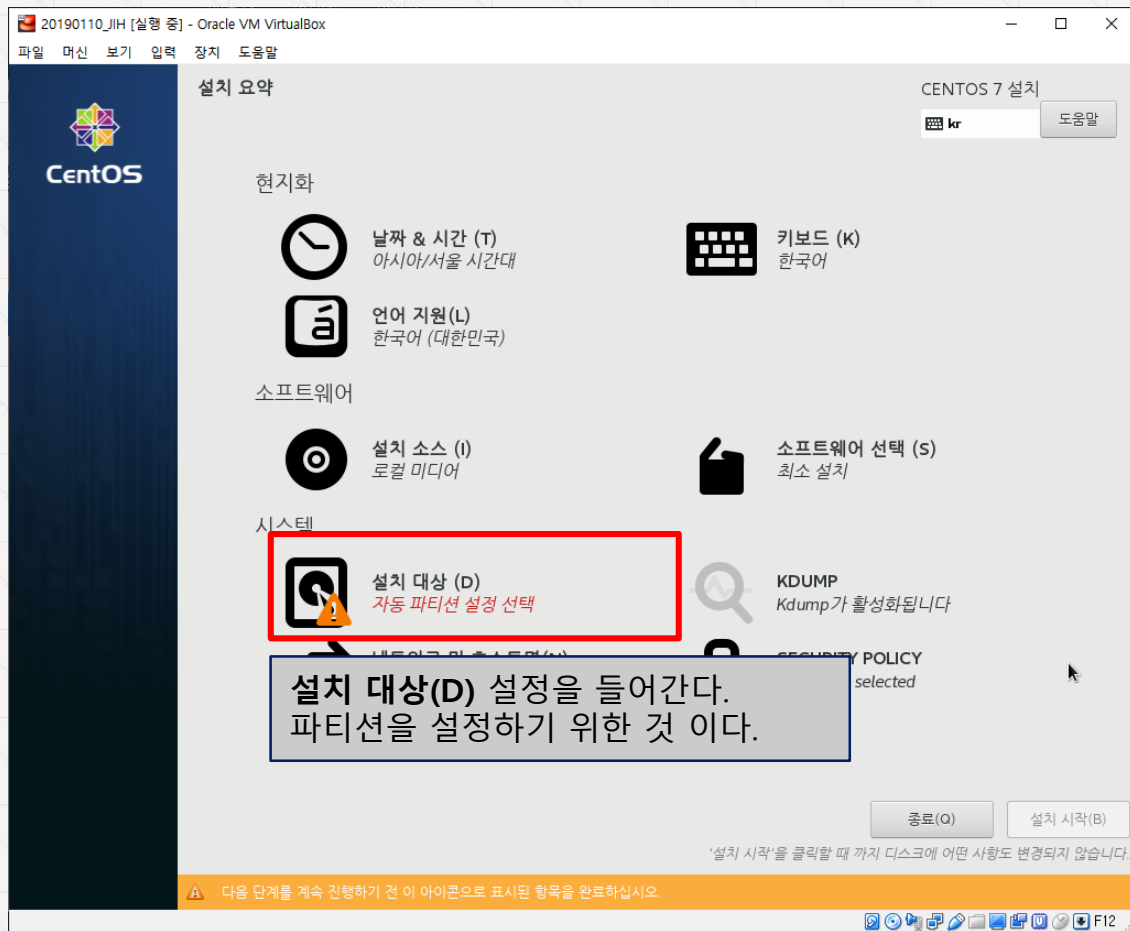




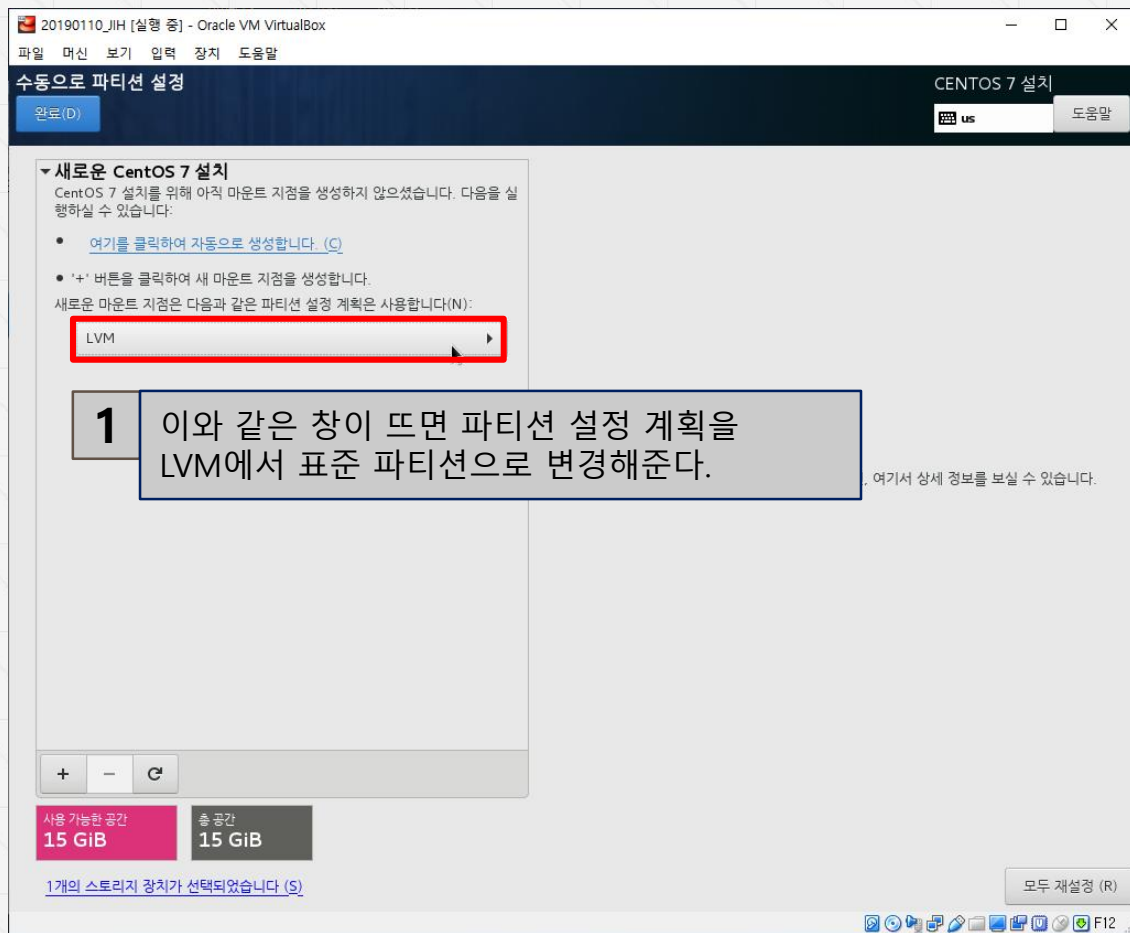


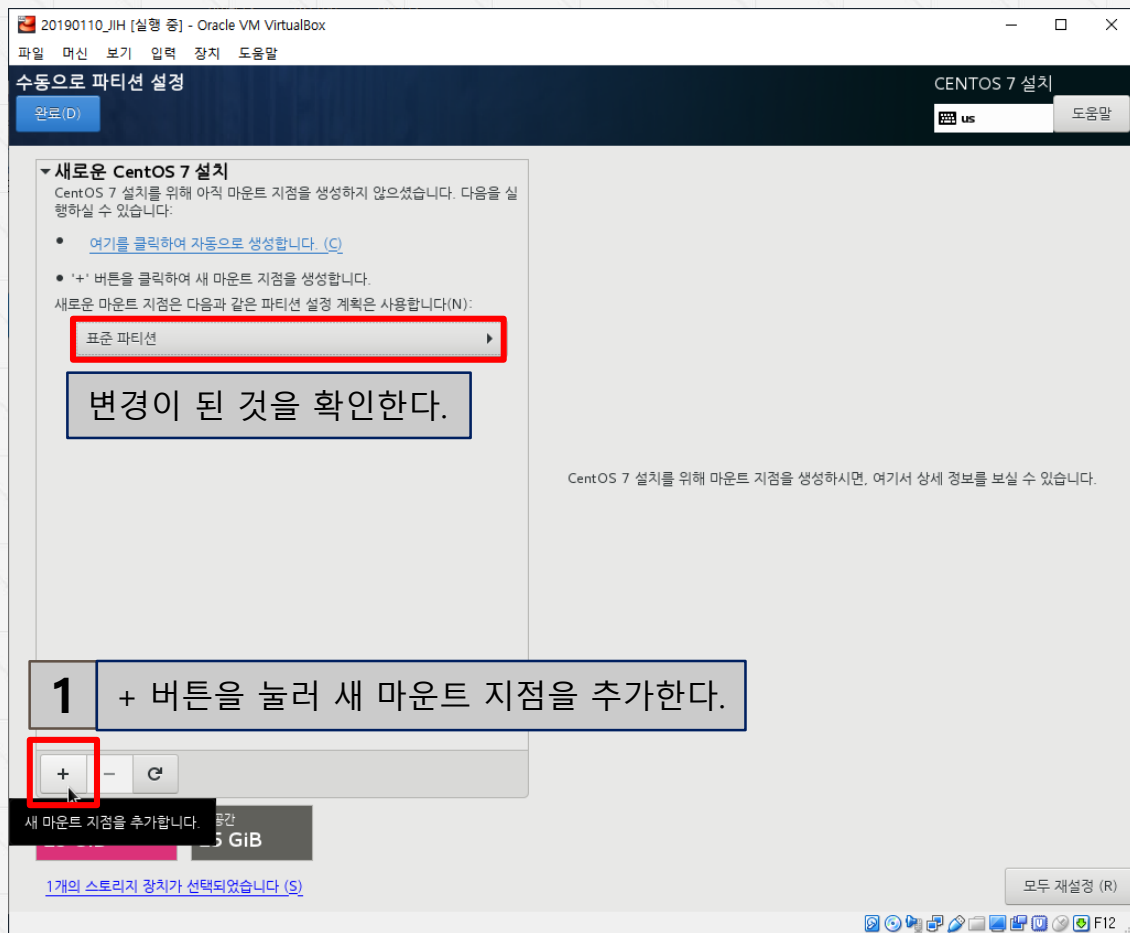


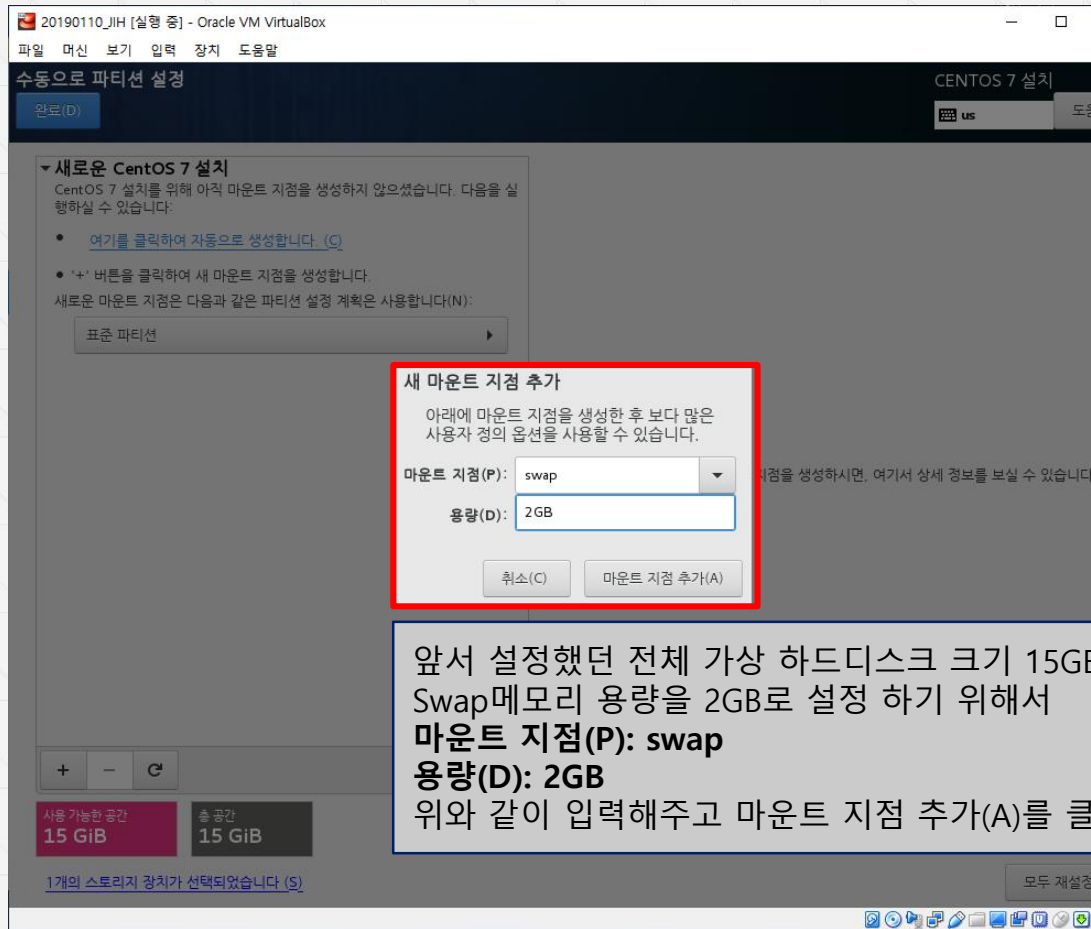


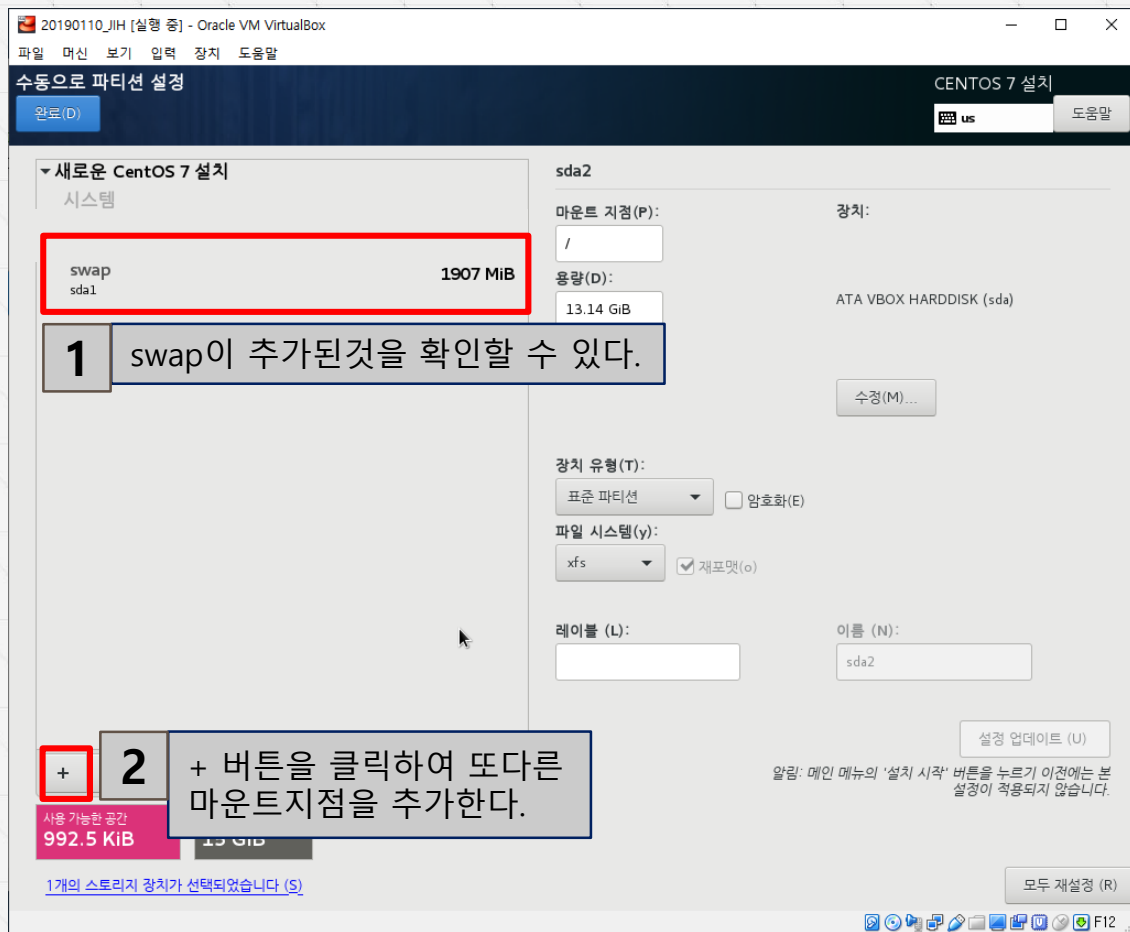


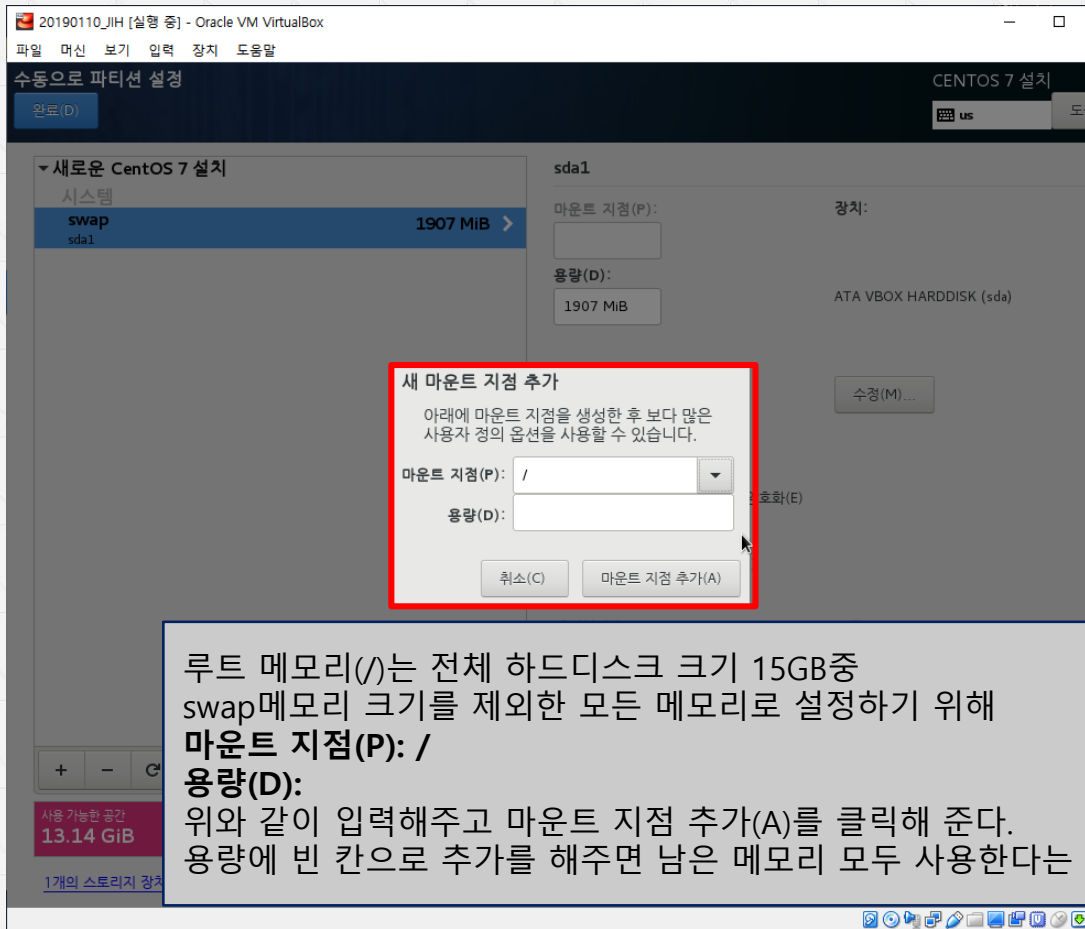


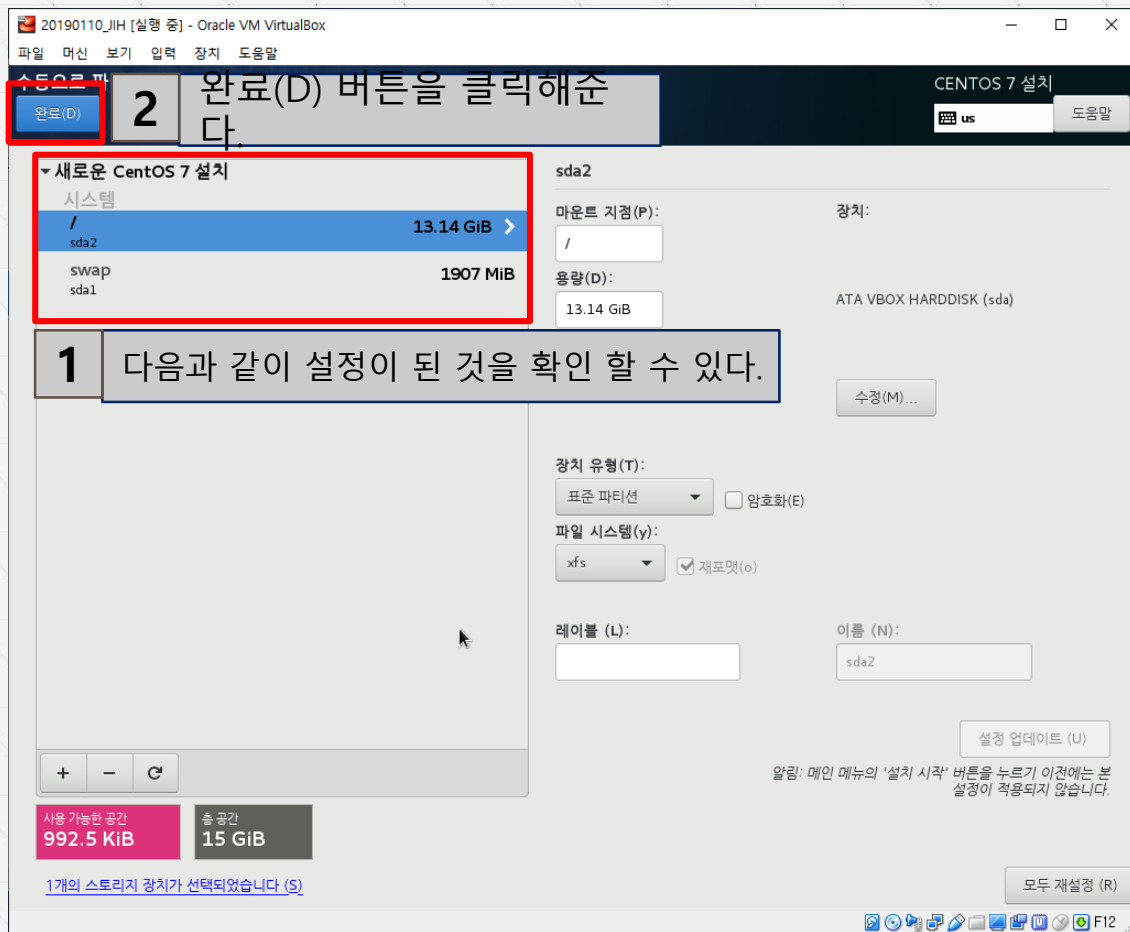


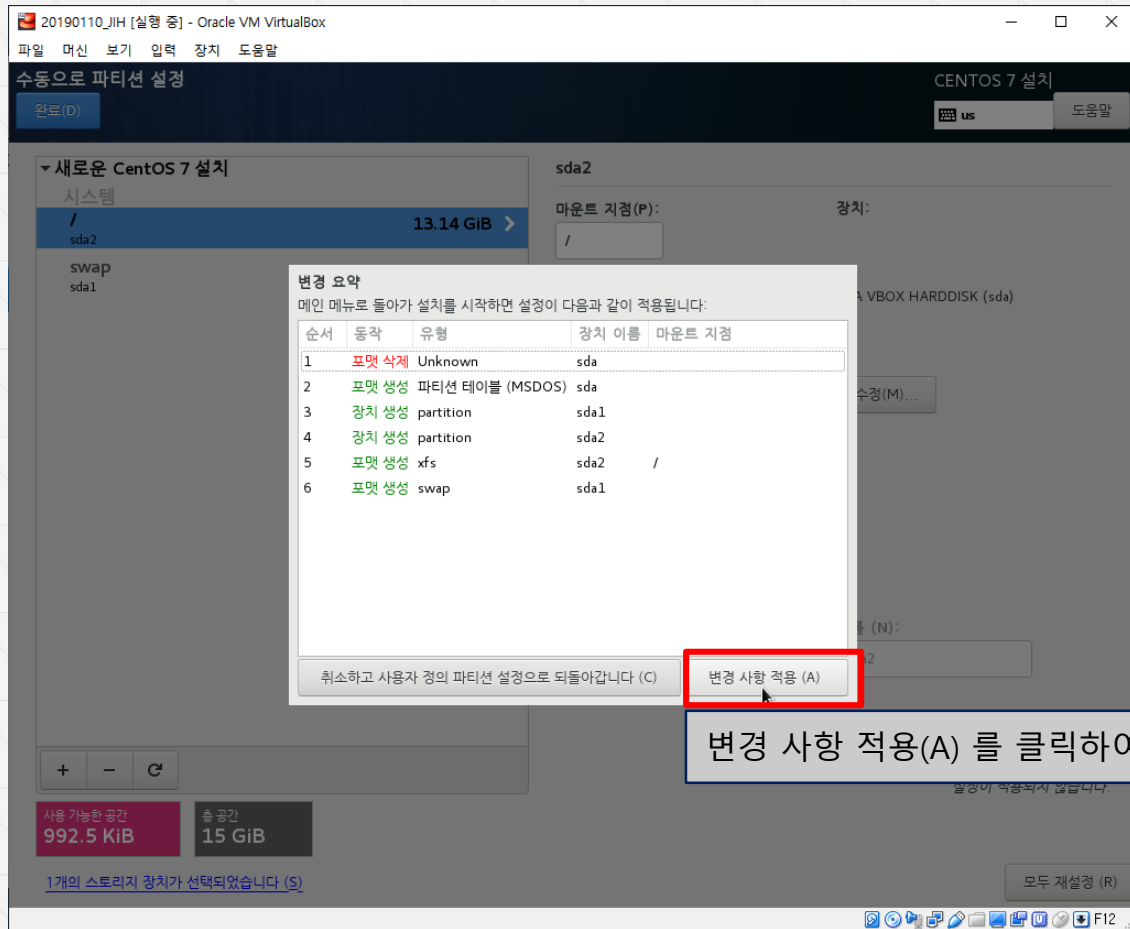


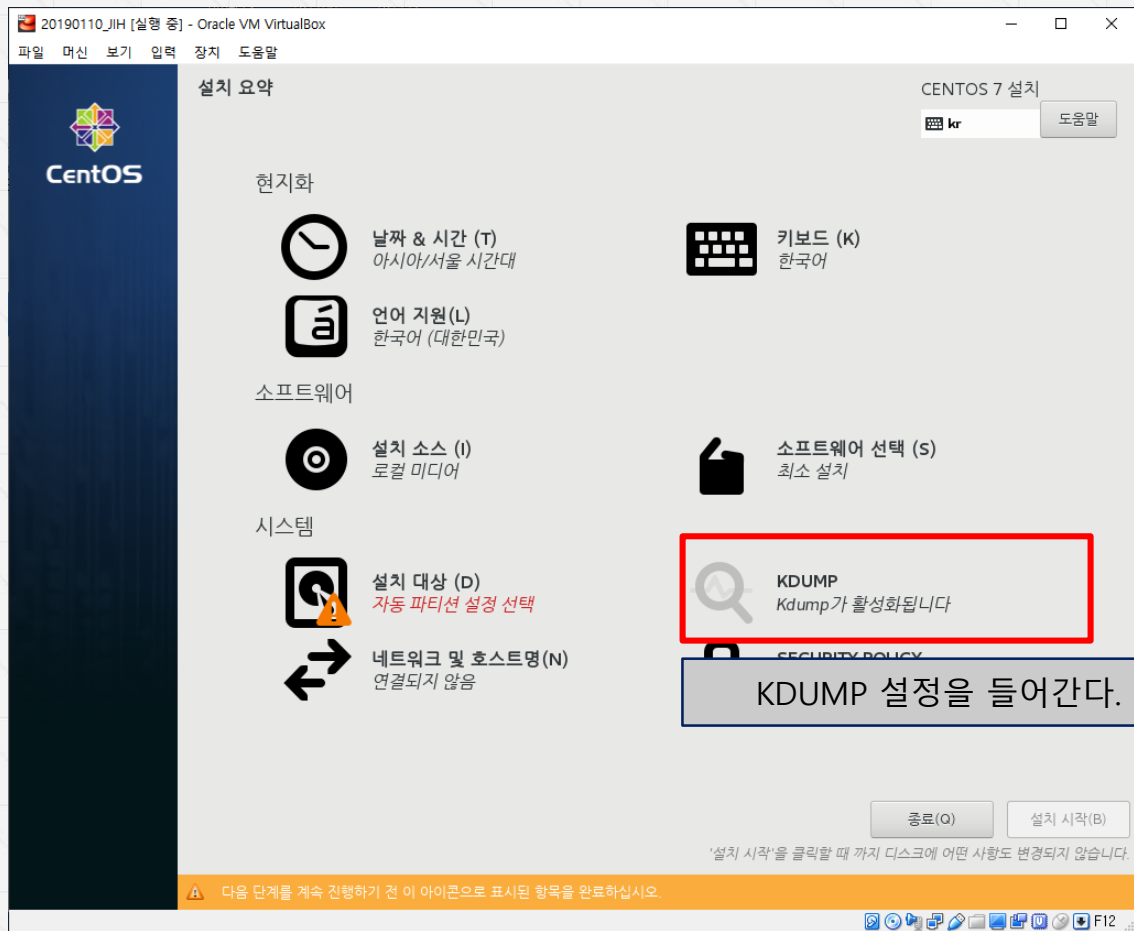


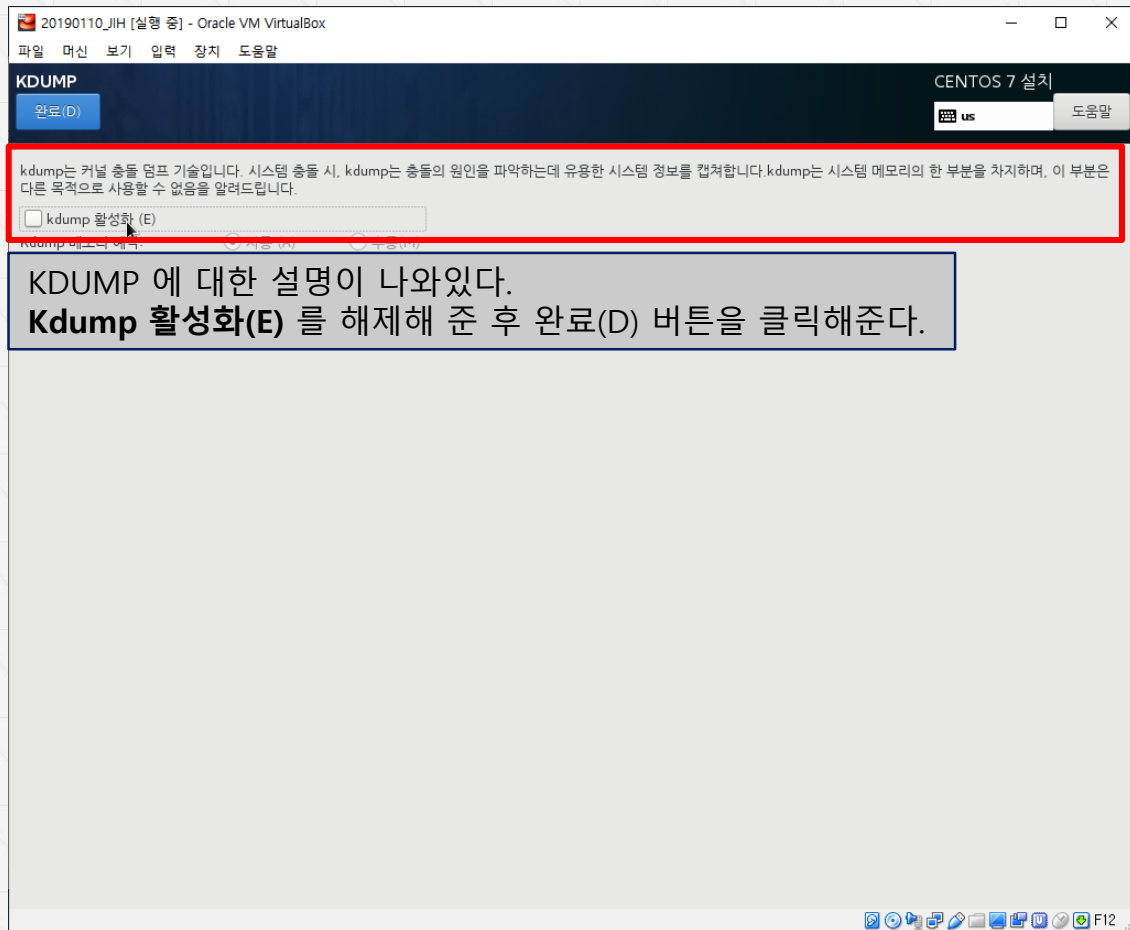


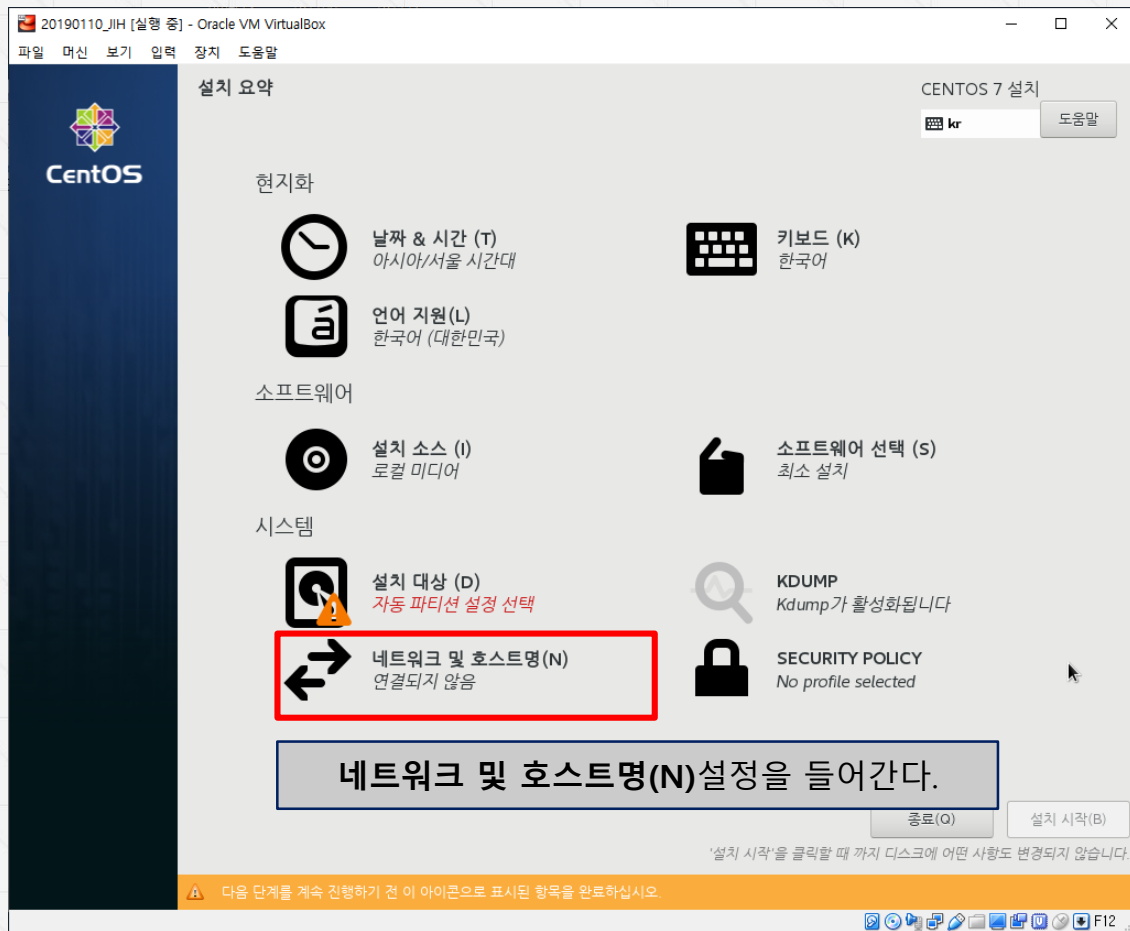


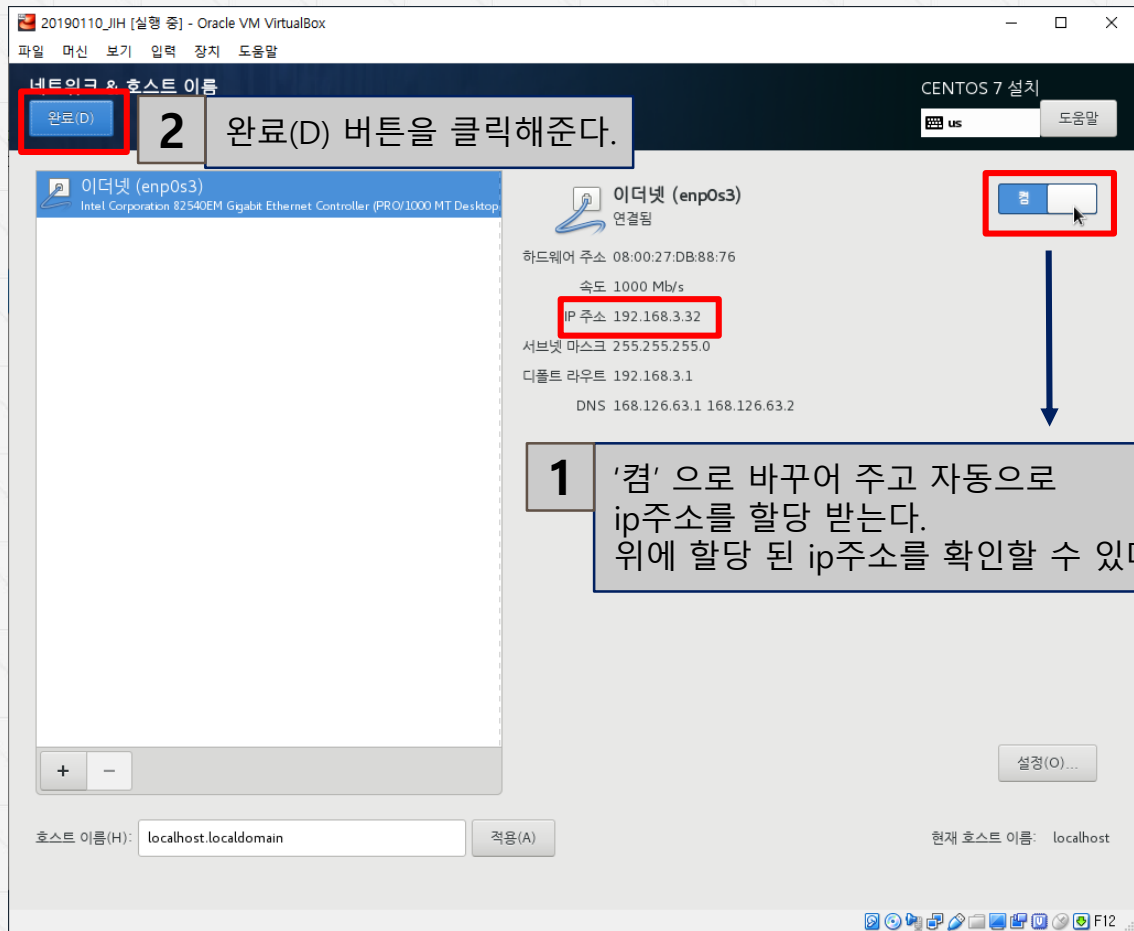


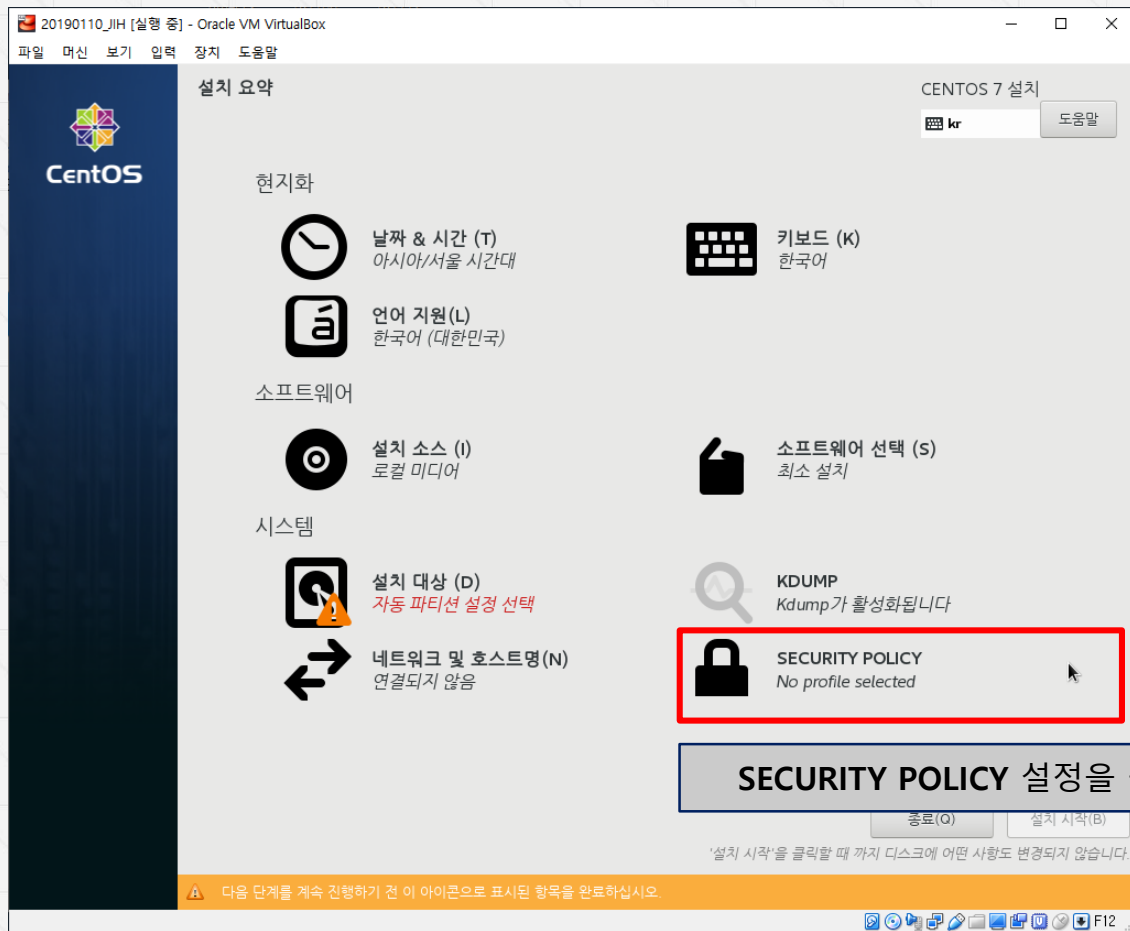




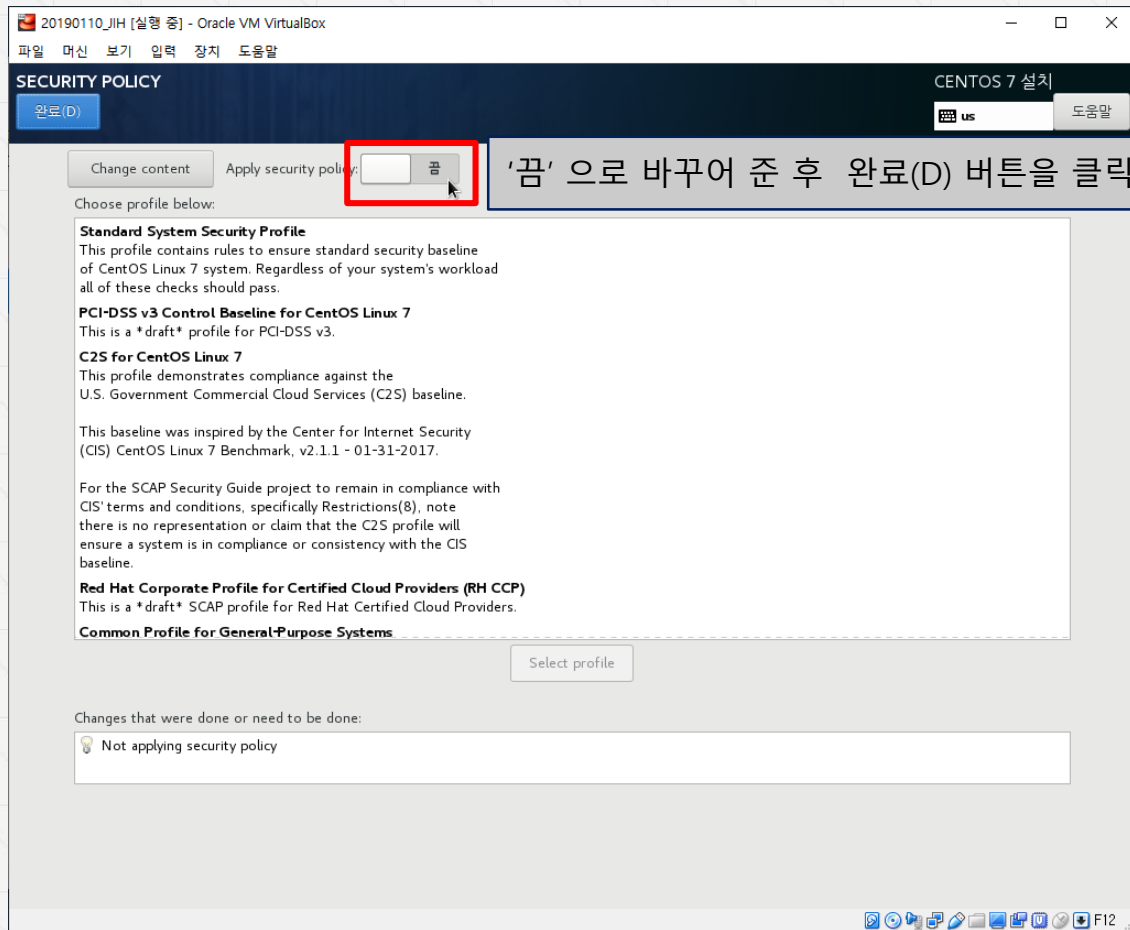


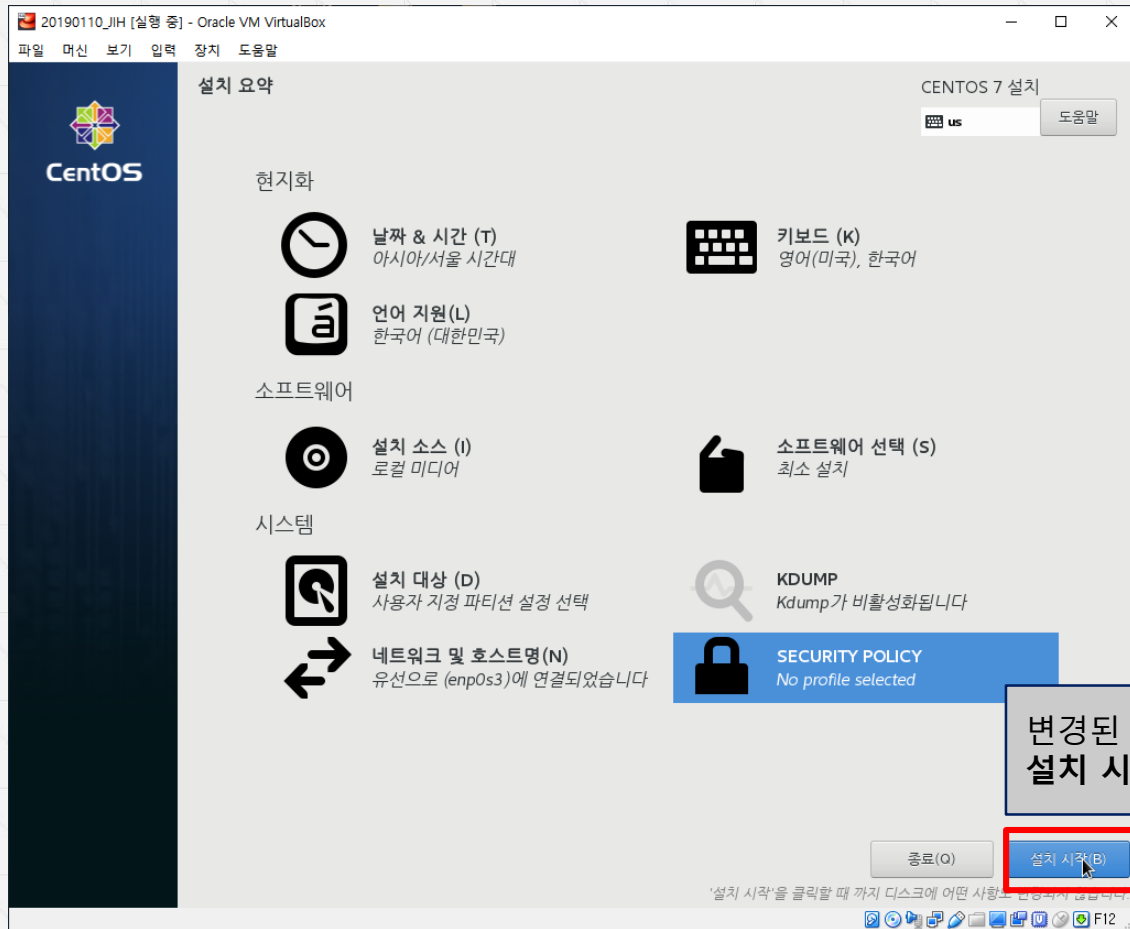


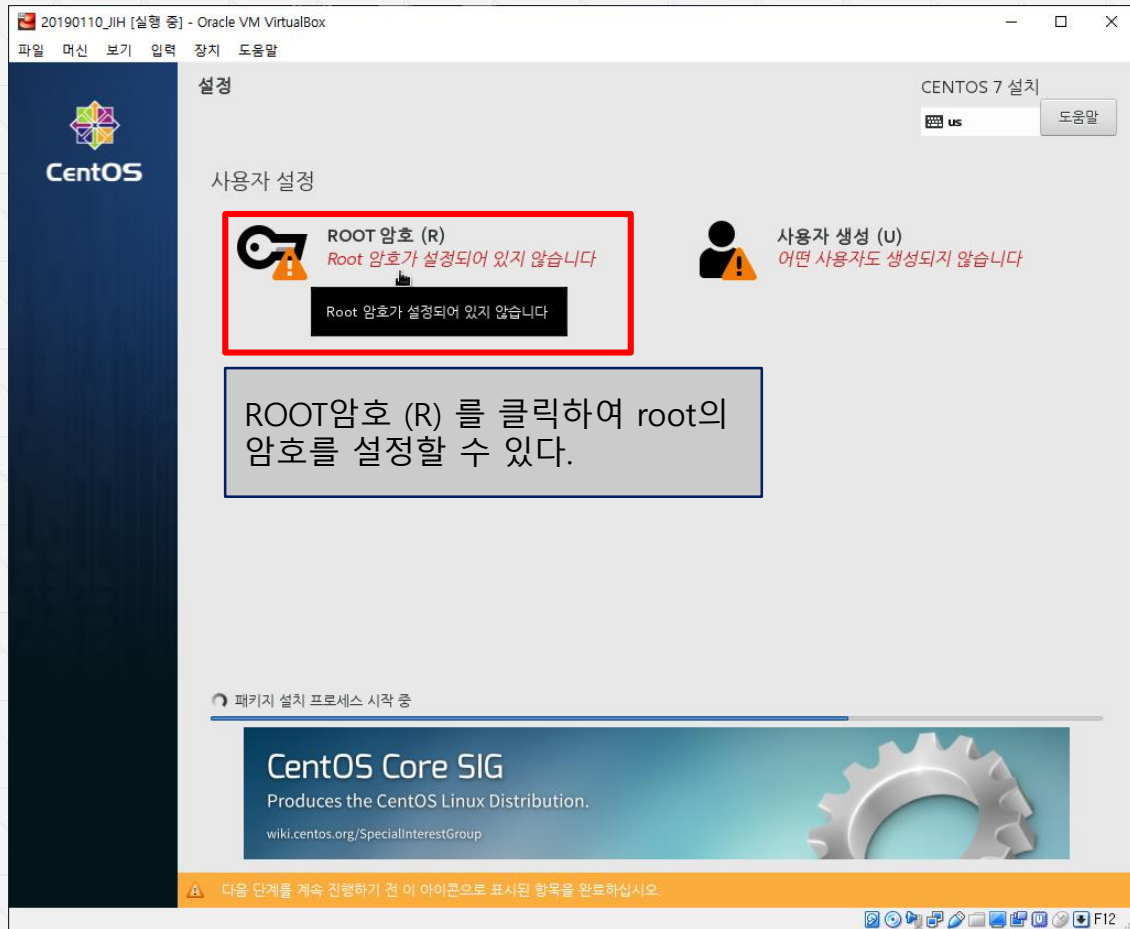


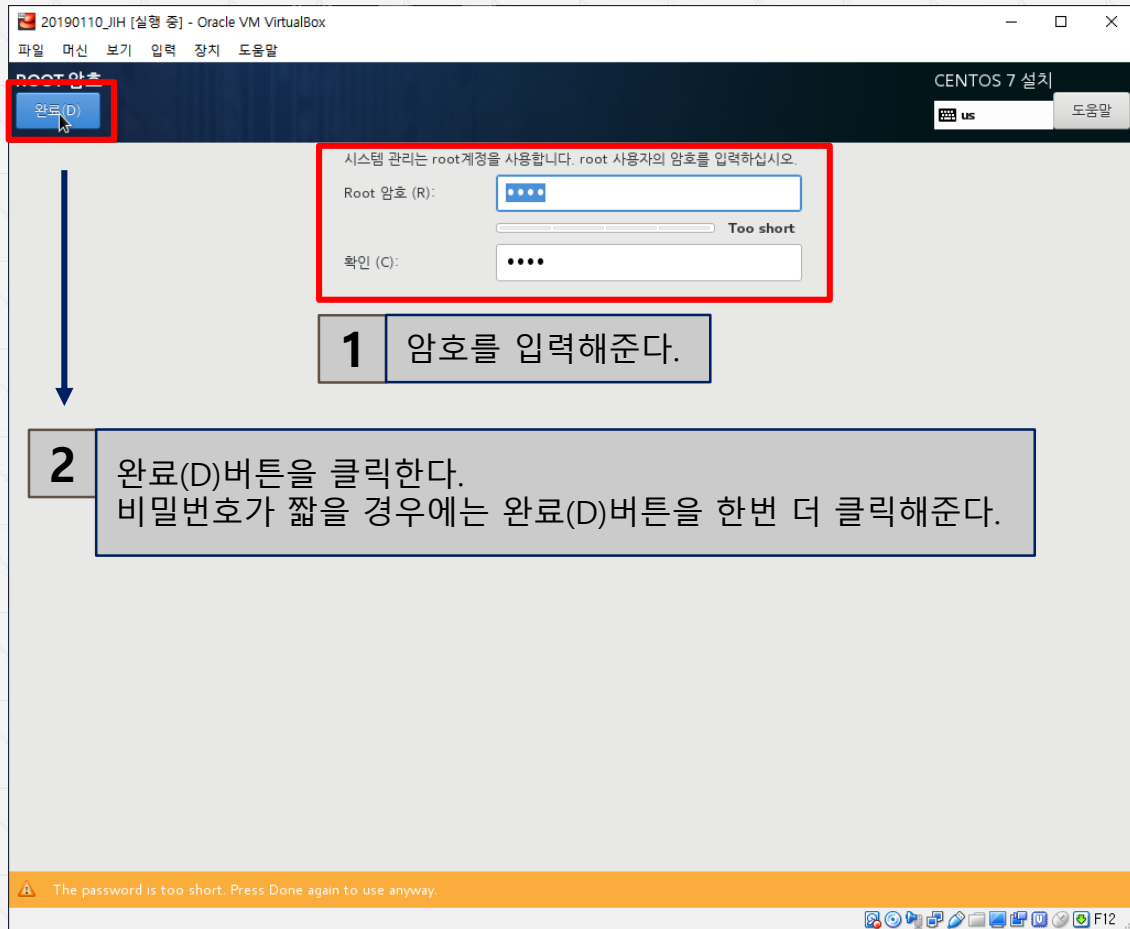


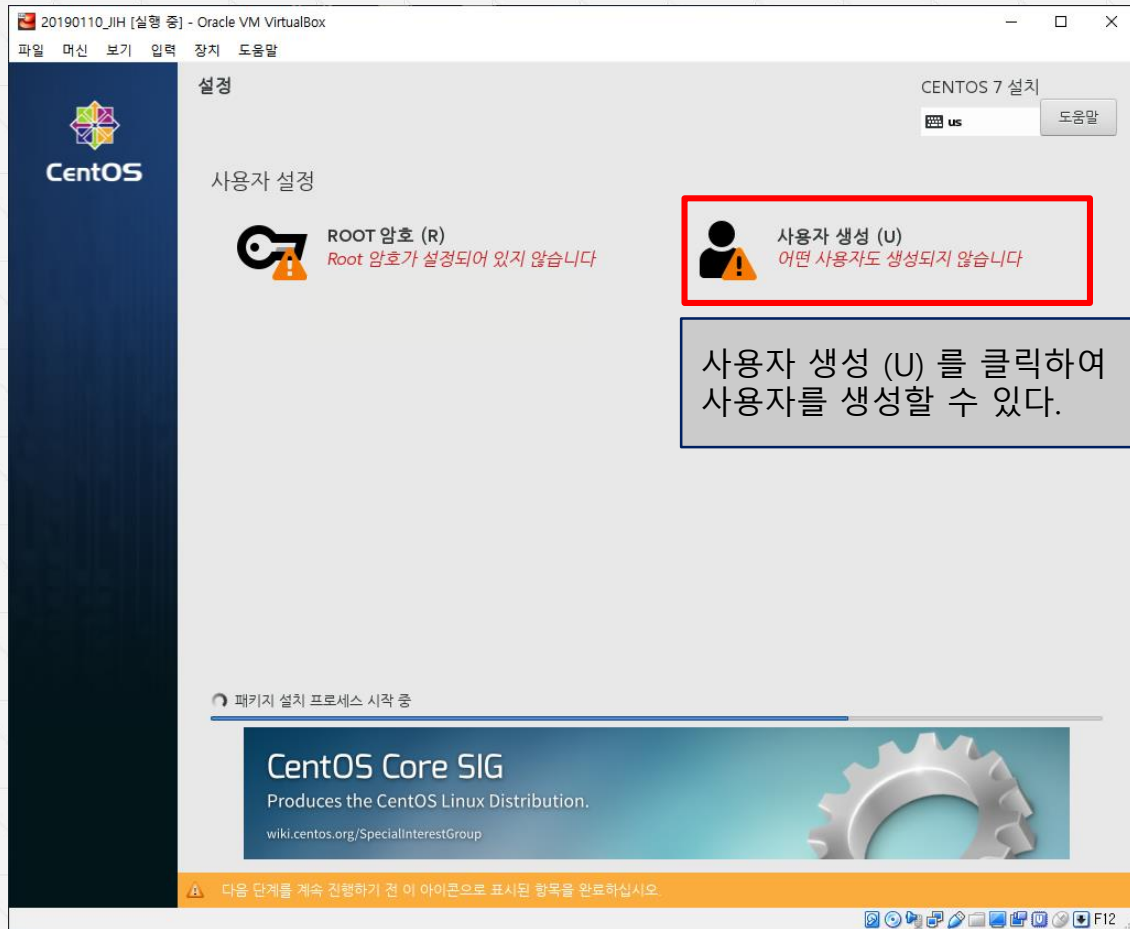
SECURITY POLICY 설정을 들어간다.

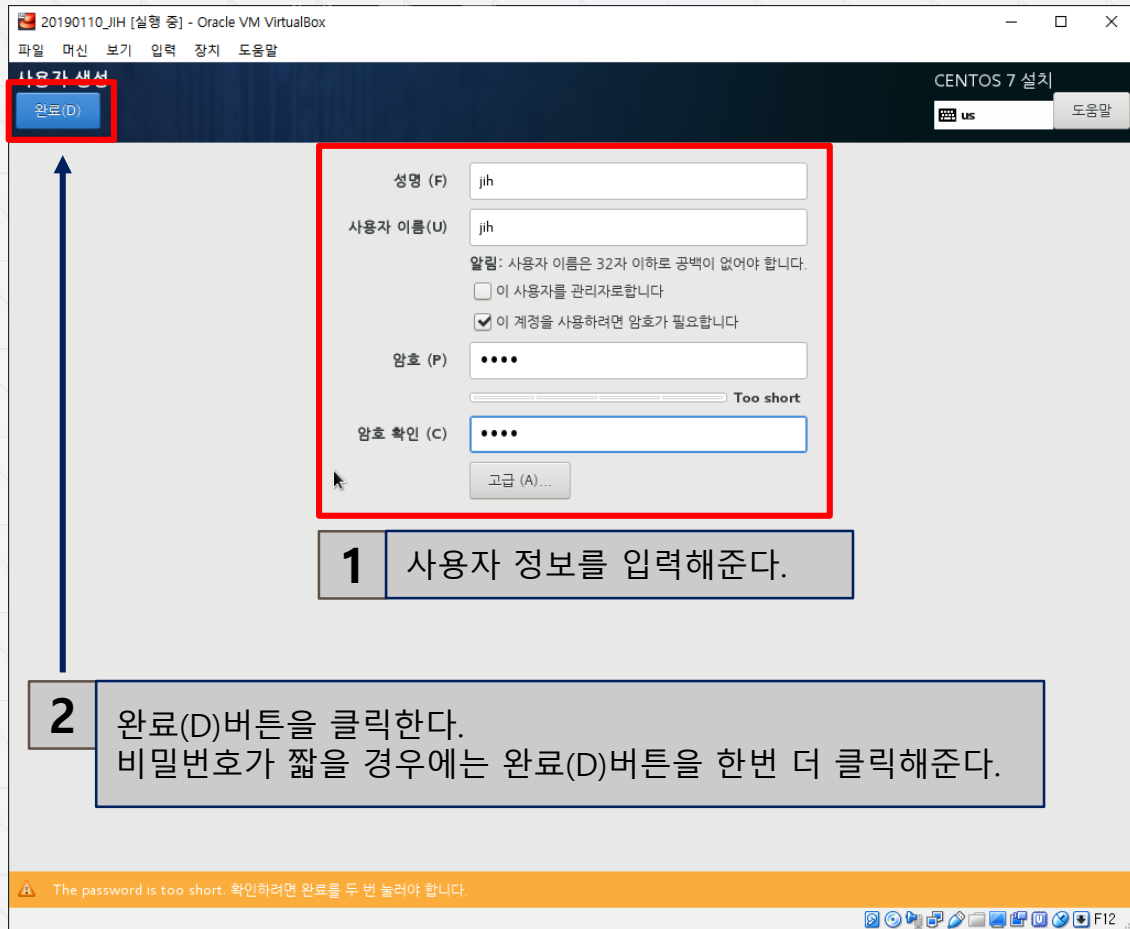


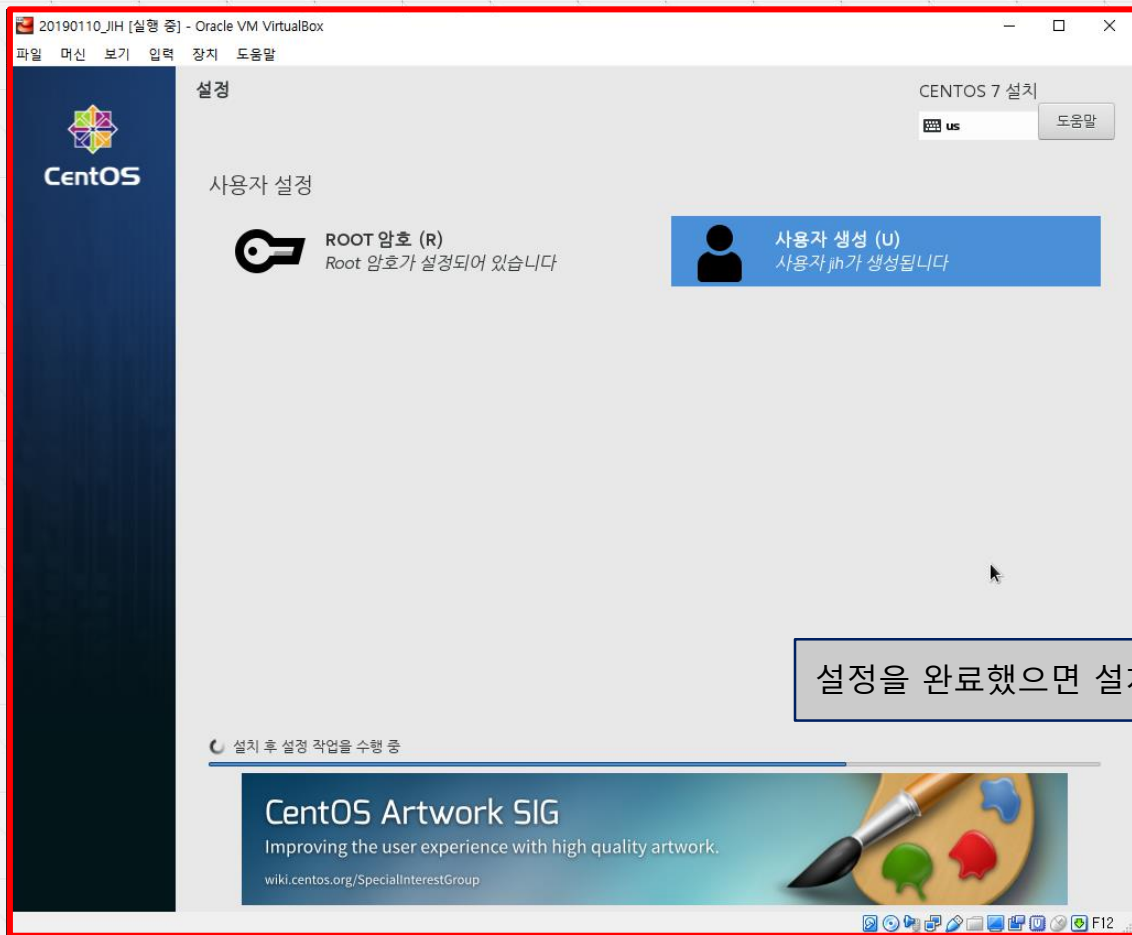


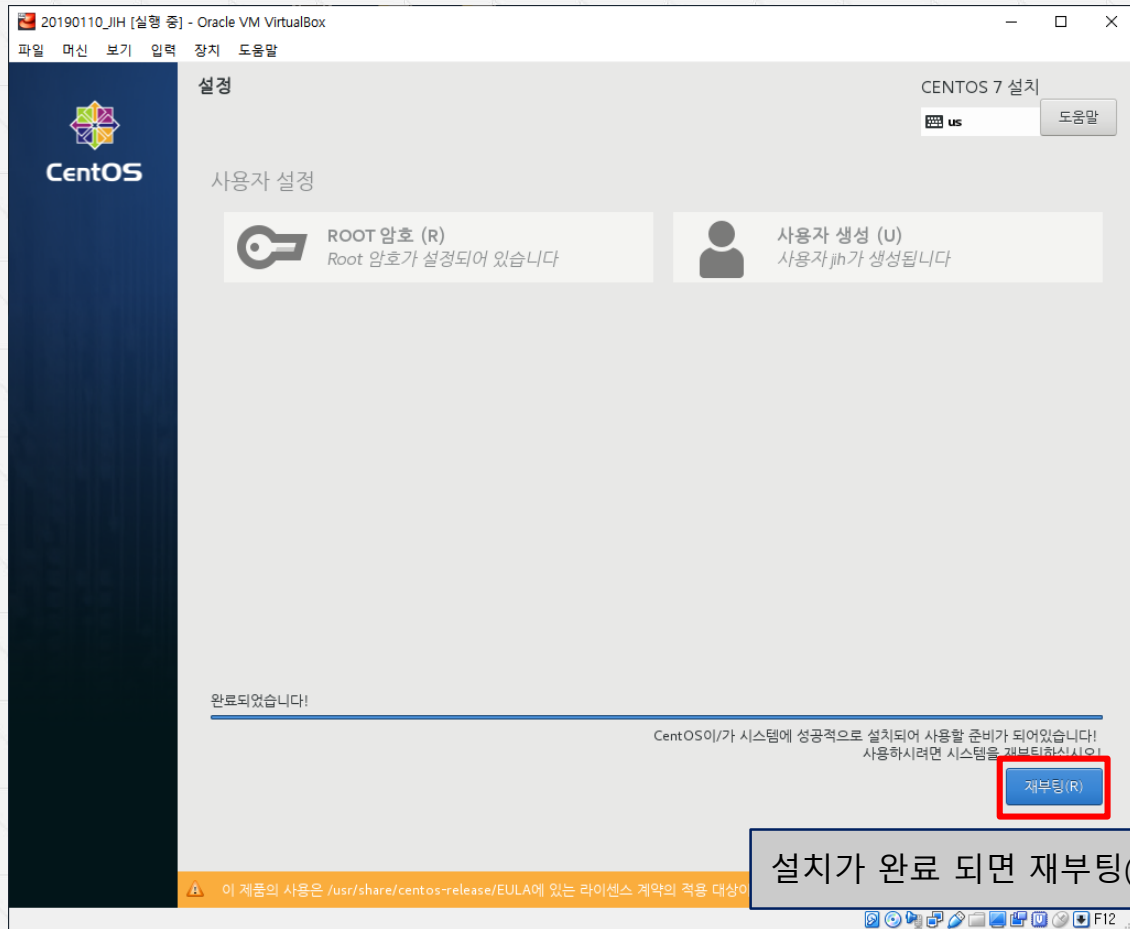




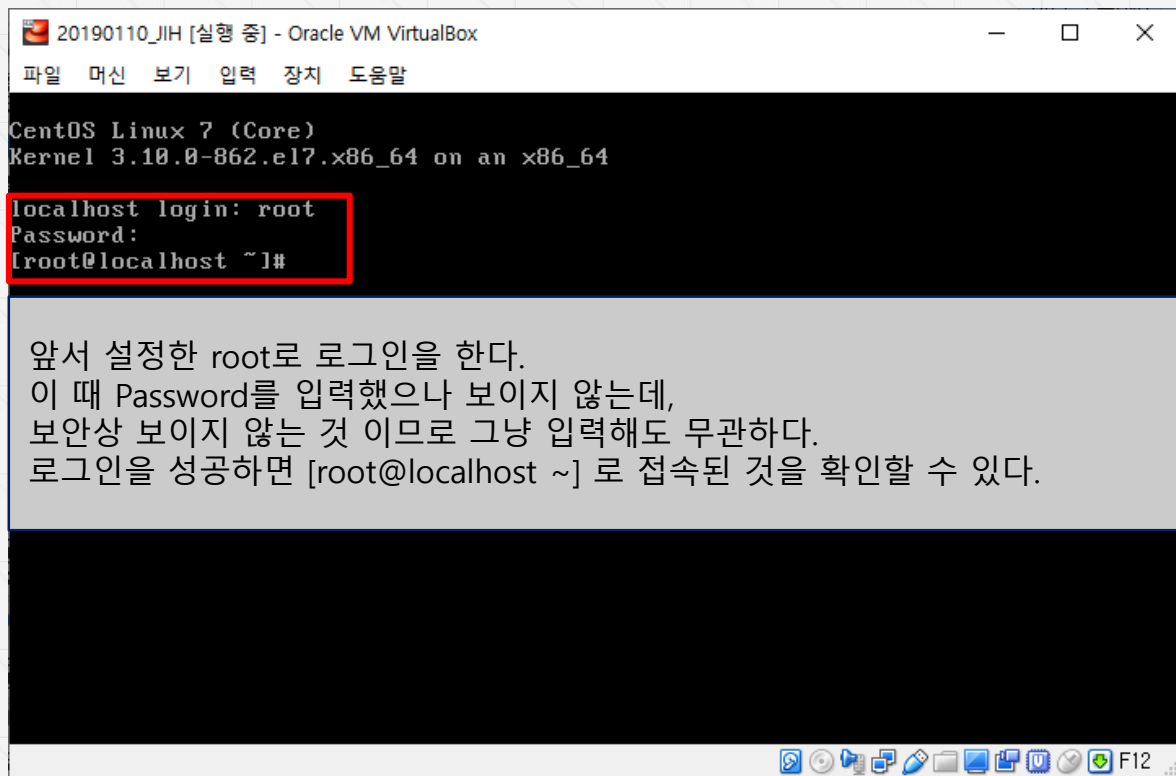






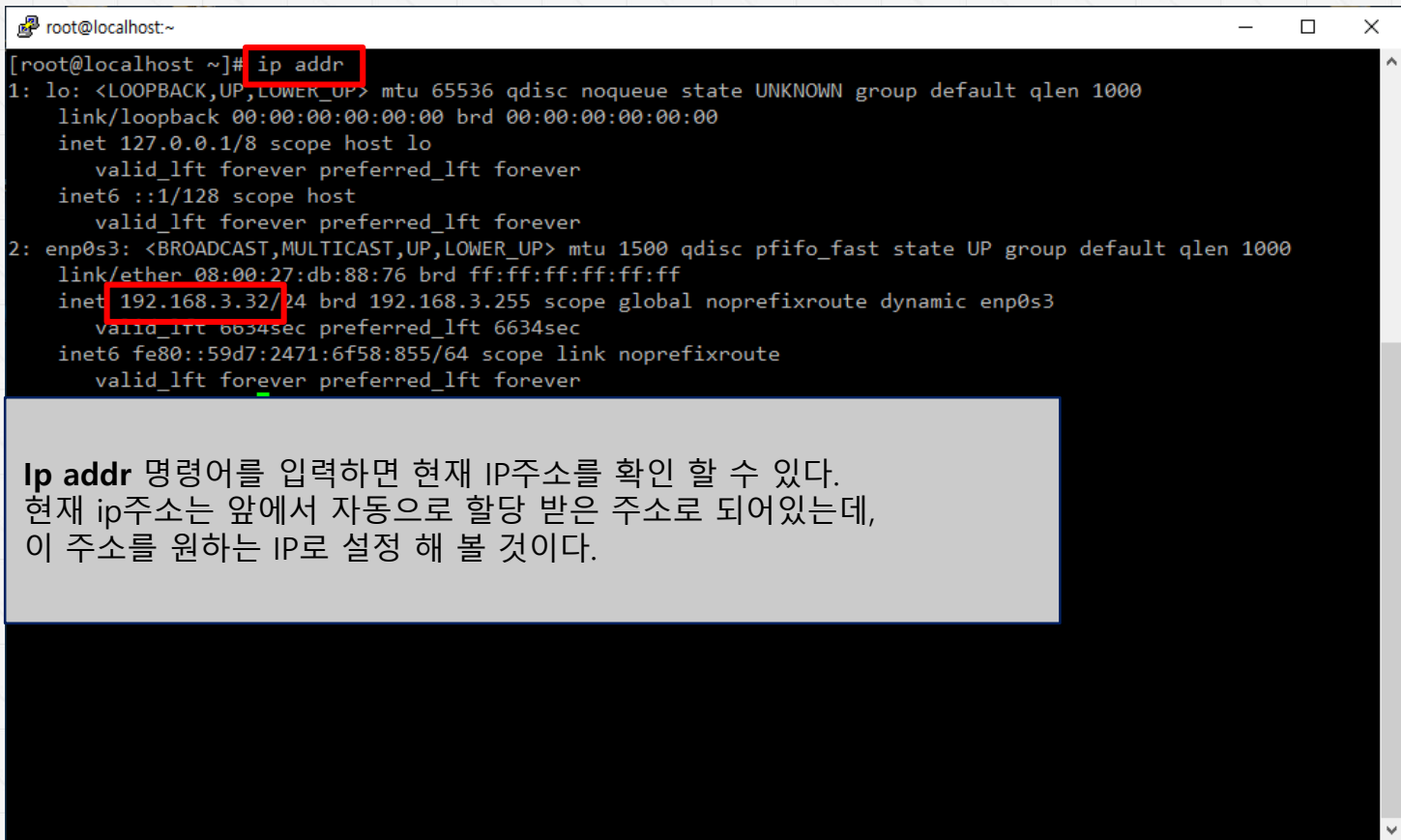


설치가 완료 되면 재부팅(R) 을 클릭해 준다.



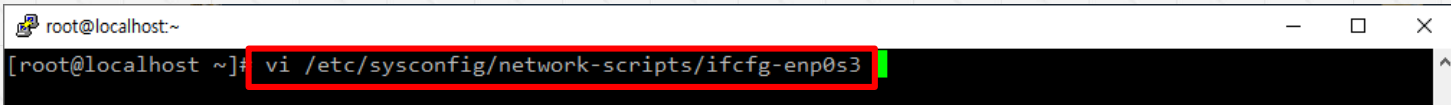
4.

IP Address Static (고정아이피 설정)



```
root@localhost:~  
[root@localhost ~]# ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000  
    link/ether 08:00:27:db:88:76 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.3.32/24 brd 192.168.3.255 scope global noprefixroute dynamic enp0s3  
        valid_lft 6634sec preferred_lft 6634sec  
    inet6 fe80::59d7:2471:6f58:855/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever
```

ip addr 명령어를 입력하면 현재 IP주소를 확인 할 수 있다.
현재 ip주소는 앞에서 자동으로 할당 받은 주소로 되어있는데,
이 주소를 원하는 IP로 설정 해 볼 것이다.

A terminal window with a title bar showing 'root@localhost:~'. The command prompt is '[root@localhost ~]#'. The command 'vi /etc/sysconfig/network-scripts/ifcfg-enp0s3' is entered and highlighted with a red box. The terminal background is black with white text.

```
root@localhost:~  
[root@localhost ~]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s3
```

고정 IP로 설정하기 위해서 네트워크 설정파일을 vi편집기로 연다.
경로는 `/etc/sysconfig/network-scripts/ifcfg-enp0s3` 이며 이 경로는 다를 수 있다.

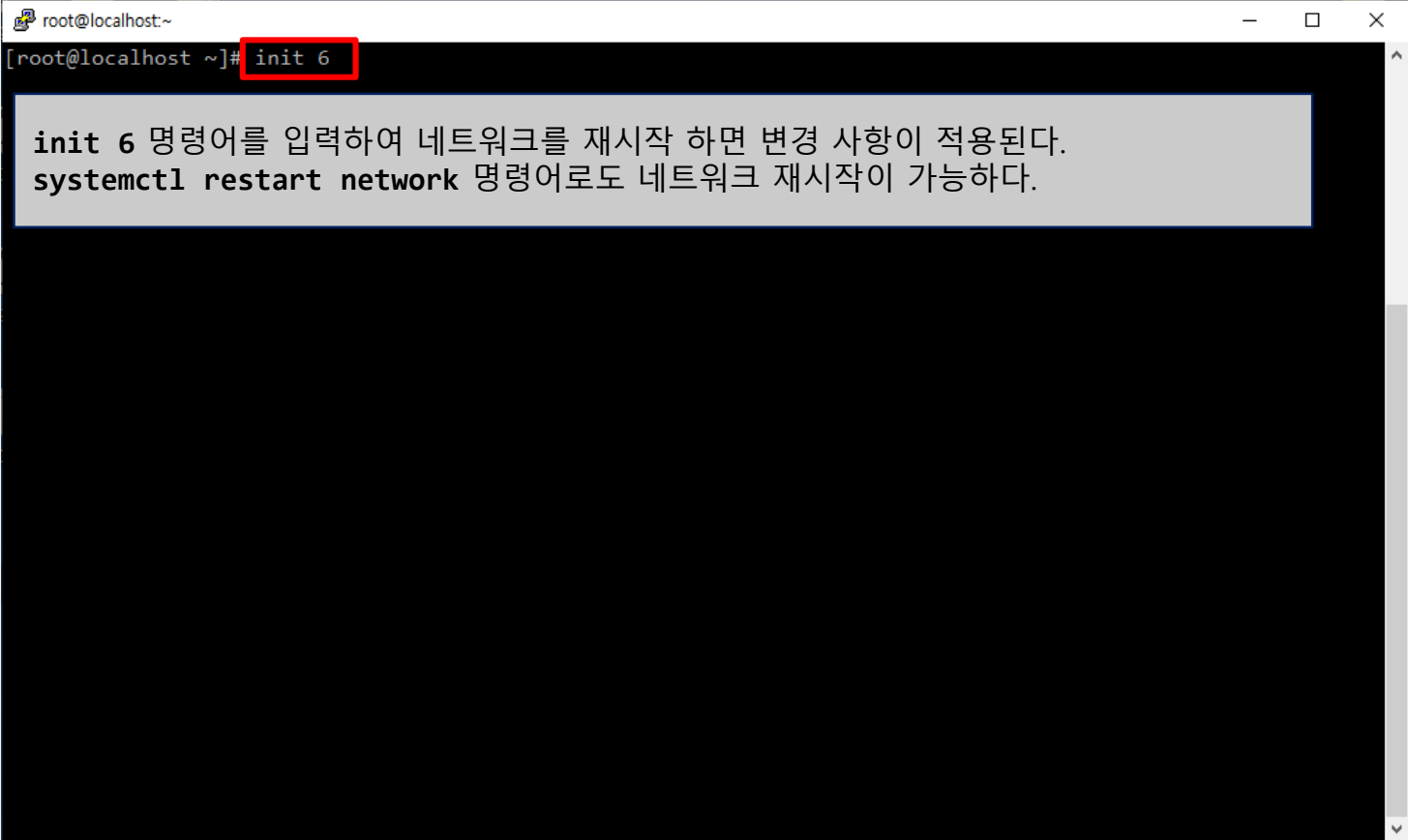
```
root@localhost:~#  
TYPE="Ethernet"  
PROXY_METHOD="none"  
BROWSER_ONLY="no"  
#BOOTPROTO="dhcp"  
DEFROUTE="yes"  
IPV4_FAILURE_FATAL="no"  
IPV6INIT="yes"  
IPV6_AUTOCONF="yes"  
IPV6_DEFROUTE="yes"  
IPV6_FAILURE_FATAL="no"  
IPV6_ADDR_GEN_MODE="stable-privacy"  
NAME="enp0s3"  
UUID="f00fb742-0074-49ec-b5d4-210beac766cc"  
DEVICE="enp0s3"  
ONBOOT="yes"  
BOOTPROTO=static  
IPADDR=192.168.3.107  
NETMASK=255.255.255.0  
GATEWAY=192.168.3.1  
DNS1=168.126.63.1  
DNS2=168.126.63.2
```

BOOTPROTO="dhcp" 를 제일 앞에 #을 붙여 주석 처리한다.

BOOTPROTO=static
IPADDR=원하는 고정IP
NETMASK=넷마스크
GATEWAY=게이트웨이 주소
DNS1=DNS주소1
DNS2=DNS주소2

네트워크 정보를 아래에 추가해준다.

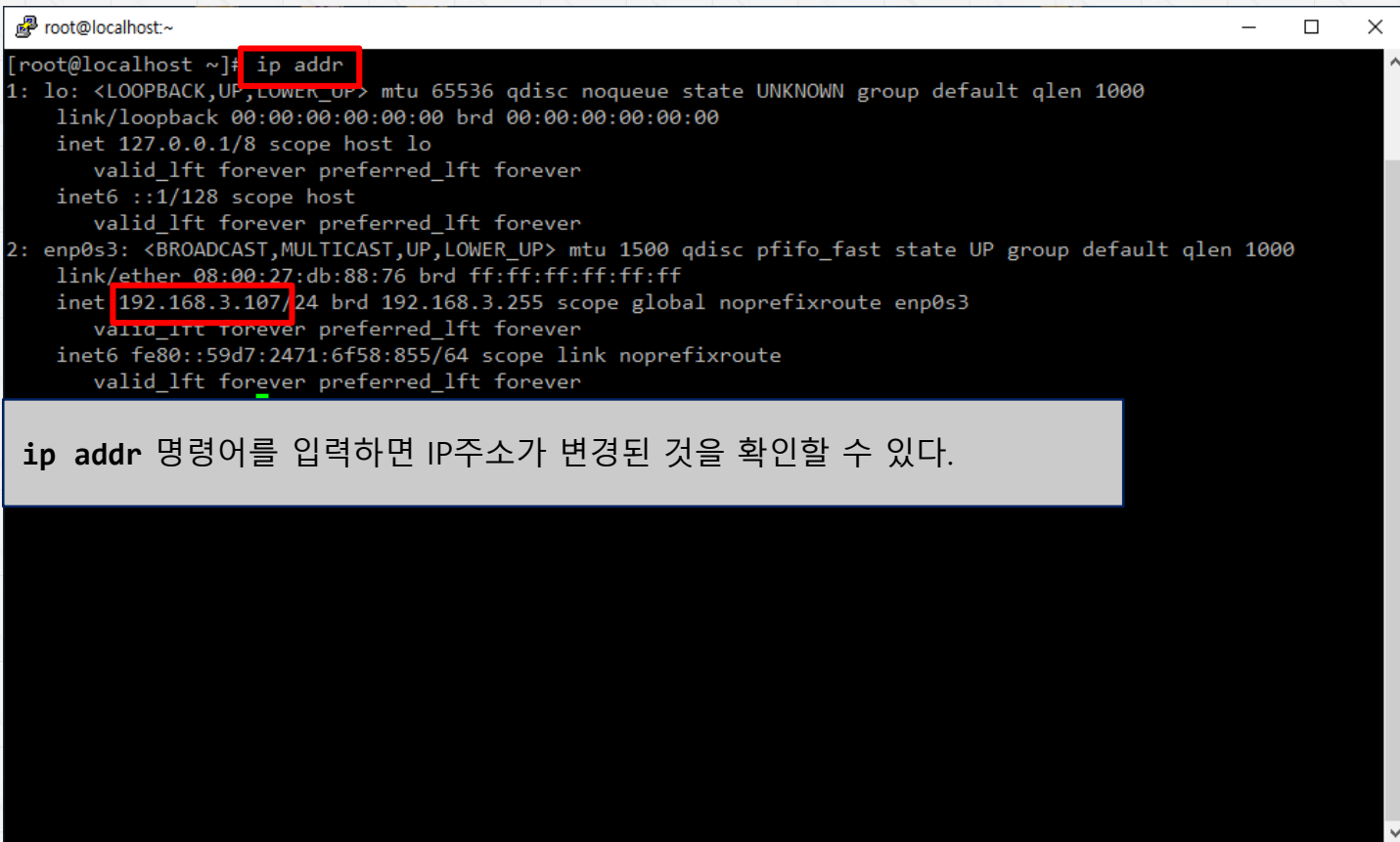
```
: wq
```

A terminal window titled 'root@localhost~' with standard window controls. The prompt is '[root@localhost ~]#'. The command 'init 6' is entered and highlighted with a red box. A light gray text box is overlaid on the terminal, containing Korean text explaining the command.

```
root@localhost~
```

```
[root@localhost ~]# init 6
```

init 6 명령어를 입력하여 네트워크를 재시작 하면 변경 사항이 적용된다.
systemctl restart network 명령어로도 네트워크 재시작이 가능하다.

A terminal window titled 'root@localhost~' with standard window controls. The command 'ip addr' has been entered and its output is displayed. Two red boxes highlight the command and the IP address '192.168.3.107' in the output.

```
root@localhost ~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:db:88:76 brd ff:ff:ff:ff:ff:ff
    inet 192.168.3.107/24 brd 192.168.3.255 scope global noprefixroute enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::59d7:2471:6f58:855/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

`ip addr` 명령어를 입력하면 IP주소가 변경된 것을 확인할 수 있다.

5-1.

Software Install_git

05-1

Software Install_git

```
root@localhost:~  
[root@localhost ~]# yum install -y git  
Verifying : perl-Filter-1.49-3.el7.x86_64 29/32  
Verifying : perl 30/32  
Verifying : perl 31/32  
Verifying : 4:perl-lib 32/32  
  
Installed:  
git.x86_64 0:1.8.3.1-20.el7  
  
Dependency Installed:  
perl.x86_64 4:5.16.3-293.el7  
perl-Encode.x86_64 0:2.51-7.el7  
perl-Exporter.noarch 0:5.68-3.el7  
perl-File-Temp.noarch 0:0.23.01-3.el7  
perl-Getopt-Long.noarch 0:2.40-3.el7  
perl-HTTP-Tiny.noarch 0:0.033-3.el7  
perl-Pod-Escapes.noarch 1:1.04-293.el7  
perl-Pod-Simple.noarch 1:3.28-4.el7  
perl-Scalar-List-Utils.x86_64 0:1.27-248.el7  
perl-Storable.x86_64 0:2.45-3.el7  
perl-Text-ParseWords.noarch 0:3.29-4.el7  
perl-Time-Local.noarch 0:1.2300-2.el7  
perl-libs.x86_64 4:5.16.3-293.el7  
perl-parent.noarch 1:0.225-244.el7  
perl-threads.x86_64 0:1.87-4.el7  
rsync.x86_64 0:3.1.2-4.el7  
perl-Carp.noarch 0:1.26-244.el7  
perl-Error.noarch 1:0.17020-2.el7  
perl-File-Path.noarch 0:2.09-2.el7  
perl-Filter.x86_64 0:1.49-3.el7  
perl-Git.noarch 0:1.8.3.1-20.el7  
perl-PathTools.x86_64 0:3.40-5.el7  
perl-Pod-Perldoc.noarch 0:3.20-4.el7  
perl-Pod-Usage.noarch 0:1.63-3.el7  
perl-Socket.x86_64 0:2.010-4.el7  
perl-TermReadKey.x86_64 0:2.30-20.el7  
perl-Time-HiRes.x86_64 4:1.9725-3.el7  
perl-constant.noarch 0:1.27-2.el7  
perl-macros.x86_64 4:5.16.3-293.el7  
perl-podlators.noarch 0:2.5.1-3.el7  
perl-threads-shared.x86_64 0:1.43-6.el7  
  
Complete!  
[root@localhost ~]# git --version  
git version 1.8.3.1
```

1 Git을 설치하기 위해서는
yum install -y git 명령어를 입력해주면 된다.

2 git --version 명령어를 입력한다.
git이 설치되었다면 git의 version정보를 알 수 있다.

5-2.

Software Install_dotnet

The screenshot shows the Microsoft .NET Downloads page for Linux, macOS, and Windows. The page has a blue header with the Microsoft logo and navigation links. Below the header, there's a section titled ".NET Downloads" with a blue background. A note says: "Not sure where to start? See [Hello World in 10 minutes](#) to install .NET and build your first app."

Below the note, there are two buttons: "Windows" and "Linux". The "Linux" button is highlighted with a red box.

Below the buttons, the ".NET Core" logo is displayed. Underneath, it says ".NET Core 2.2" and ".NET Core is a cross-platform version of .NET for building websites, services, and console apps."

At the bottom, there are two rows of buttons. The first row has "Build Apps" and "Install .NET Core SDK". The "Install .NET Core SDK" button is highlighted with a red box. The second row has "Run Apps" and "Install .NET Core Runtime".

Annotations:

1. <https://dotnet.microsoft.com/download> 사이트에 접속 한다.
2. CentOS 환경에 설치해야하기 때문에 Linux를 선택해준다.
3. Install .Net Core SDK 를 클릭해준다.

Install .NET Core SDK on Linux x

← → ↻ ⌂ 🔒 https://dotnet.microsoft.com/download/linux-package-manager/centos/sdk-current

Microsoft | .NET About Learn Architecture Docs Downloads Community All Microsoft

Install .NET Core SDK on Linux CentOS / Oracle x64

🕒 Not sure where to start? See [Hello World in 10 minutes](#) to install .NET and build your first app.

Linux Distribution

CentOS / Oracle

1 운영체제를 선택할 수 있는데, CentOS / Oracle 을 선택해준다.

Add the dotnet product feed

Before installing .NET, you'll need to register the Microsoft key, register the product repository, and install required dependencies. This only needs to be done once per machine.

Open a command prompt and run the following commands:

```
$ sudo rpm -Uvh https://packages.microsoft.com/config/rhel/7/packages-microsoft-prod.rpm
```

2 명령어를 순서대로 터미널창에 입력해준다.

Install the .NET SDK

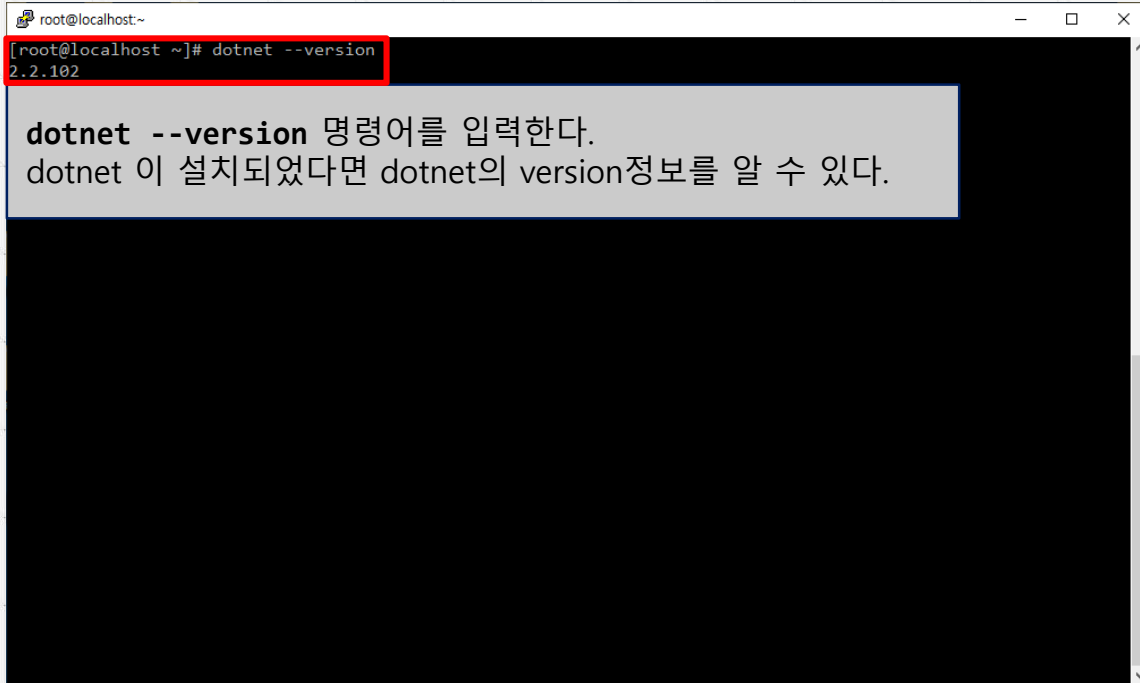
Update the products available for installation, then install the .NET SDK.

In your command prompt, run the following commands:

```
~$ sudo yum update  
~$ sudo yum install dotnet-sdk-2.2
```

3

Feedback



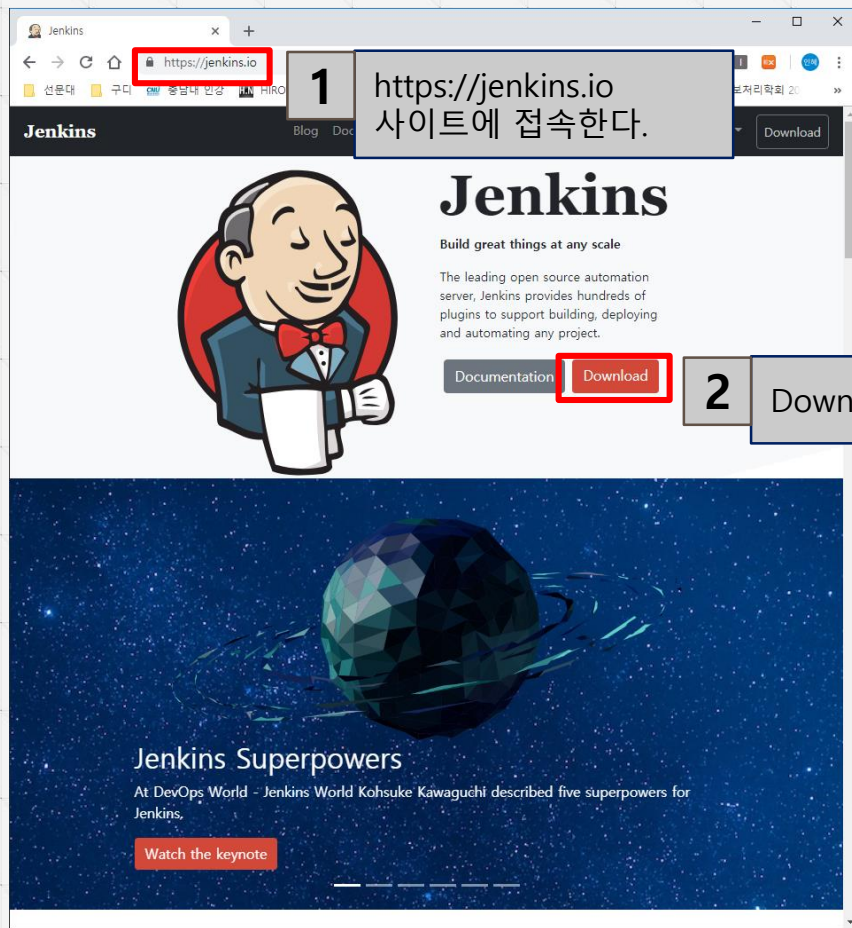
A terminal window titled 'root@localhost:~' with standard window controls. The command '[root@localhost ~]# dotnet --version' is entered and highlighted with a red box. The output '2.2.102' is displayed below the command. A grey text box is overlaid on the terminal, containing Korean text explaining the command.

```
[root@localhost ~]# dotnet --version
2.2.102
```

dotnet --version 명령어를 입력한다.
dotnet 이 설치되었다면 dotnet의 version정보를 알 수 있다.

5-3.

Software Install_Jenkins



The screenshot shows the Jenkins download page at <https://jenkins.io/download/>. The page features a navigation bar with links to Blog, Documentation, Plugins, Community, Sub-projects, and About, along with a Download button. The main content area is divided into two columns. The left column, titled 'Long-term support (LTS)', explains that LTS releases are chosen every 12 weeks from the stream of regular releases. It includes links to Changelog, Upgrade Guide, and Past Releases. Below this, there is a 'Deploy Jenkins 2.150.1' button and a 'Download Jenkins 2.150.1 for:' section with a list of operating systems. The 'Red Hat/Fedora/CentOS' option is highlighted with a red rectangular box. The right column, titled 'weekly', states that a new release is produced weekly to deliver bug fixes and features. It also includes links to Changelog and Past Releases, and a 'Download Jenkins 2.157 for:' section with a list of operating systems. A grey callout box with a blue border points to the 'Red Hat/Fedora/CentOS' option in the left column, containing the text 'CentOS 를 클릭해준다.' (Click CentOS).

Jenkins Installation and setup x +

https://jenkins.io/download/

선문대 구디 홍남대 인강 HIRO'S.NET Home - Pottermore ppt템플릿 시나공 수연수 한국정보처리학회 20

Jenkins Blog Documentation Plugins Community Sub-projects About Download

Long-term support (LTS)

LTS (Long-Term Support) releases are chosen every 12 weeks from the stream of regular releases as the stable release for that time period. [Learn more...](#)

[Changelog](#) | [Upgrade Guide](#) | [Past Releases](#)

Deploy Jenkins 2.150.1

[Deploy to Azure](#)

Download Jenkins 2.150.1 for:

- Docker
- FreeBSD
- Gentoo
- Mac OS X
- OpenBSD
- openSUSE
- Red Hat/Fedora/CentOS**
- Ubuntu/Debian
- Windows
- Generic Java package (.war)

weekly

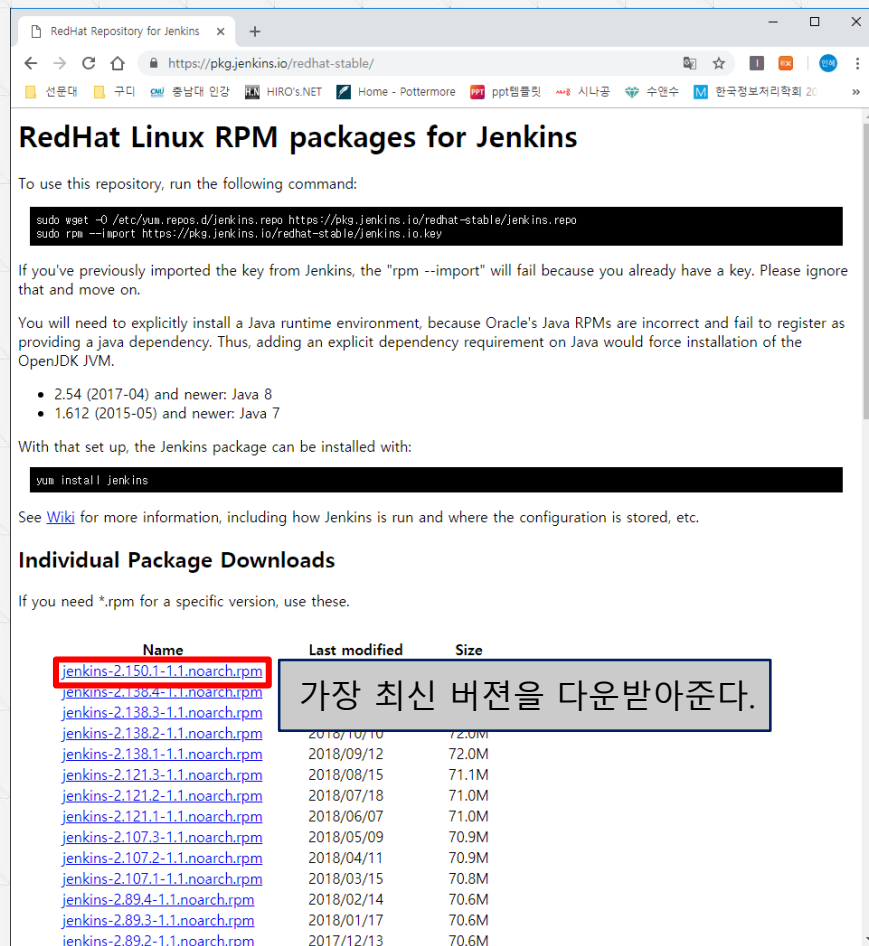
A new release is produced weekly to deliver bug fixes and features to users and plugin developers.

[Changelog](#) | [Past Releases](#)

Download Jenkins 2.157 for:

- Arch Linux
- Docker
- FreeBSD
- Gentoo
- Mac OS X
- OpenBSD
- openSUSE
- Red Hat/Fedora/CentOS
- Generic Java package (.war)

Once a Jenkins package has been downloaded, proceed to the [Installing Jenkins](#) section of the [User Handbook](#).



RedHat Linux RPM packages for Jenkins

To use this repository, run the following command:

```
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
```

If you've previously imported the key from Jenkins, the "rpm --import" will fail because you already have a key. Please ignore that and move on.

You will need to explicitly install a Java runtime environment, because Oracle's Java RPMs are incorrect and fail to register as providing a java dependency. Thus, adding an explicit dependency requirement on Java would force installation of the OpenJDK JVM.

- 2.54 (2017-04) and newer: Java 8
- 1.612 (2015-05) and newer: Java 7

With that set up, the Jenkins package can be installed with:

```
yum install jenkins
```

See [Wiki](#) for more information, including how Jenkins is run and where the configuration is stored, etc.

Individual Package Downloads

If you need *.rpm for a specific version, use these.

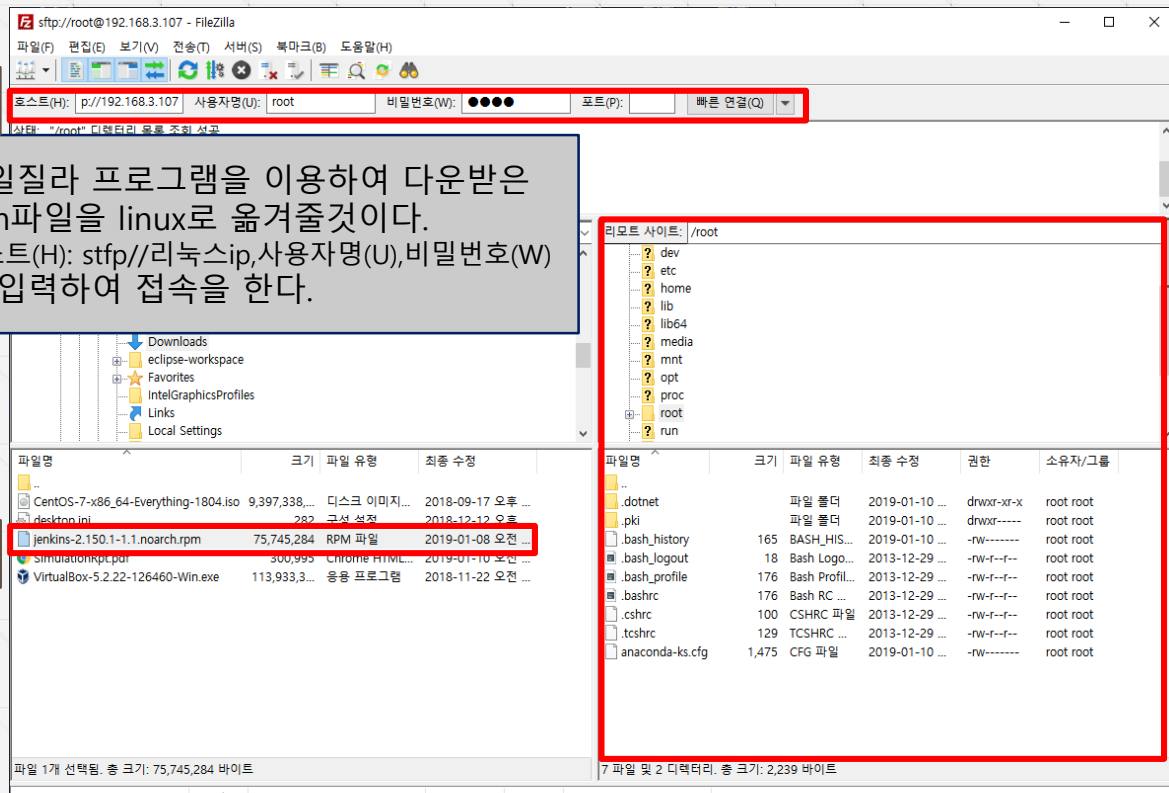
Name	Last modified	Size
jenkins-2.150.1-1.1.noarch.rpm		
jenkins-2.138.4-1.1.noarch.rpm		
jenkins-2.138.3-1.1.noarch.rpm		
jenkins-2.138.2-1.1.noarch.rpm	2018/10/10	72.0M
jenkins-2.138.1-1.1.noarch.rpm	2018/09/12	72.0M
jenkins-2.121.3-1.1.noarch.rpm	2018/08/15	71.1M
jenkins-2.121.2-1.1.noarch.rpm	2018/07/18	71.0M
jenkins-2.121.1-1.1.noarch.rpm	2018/06/07	71.0M
jenkins-2.107.3-1.1.noarch.rpm	2018/05/09	70.9M
jenkins-2.107.2-1.1.noarch.rpm	2018/04/11	70.9M
jenkins-2.107.1-1.1.noarch.rpm	2018/03/15	70.8M
jenkins-2.89.4-1.1.noarch.rpm	2018/02/14	70.6M
jenkins-2.89.3-1.1.noarch.rpm	2018/01/17	70.6M
jenkins-2.89.2-1.1.noarch.rpm	2017/12/13	70.6M

가장 최신 버전을 다운받아준다.

1

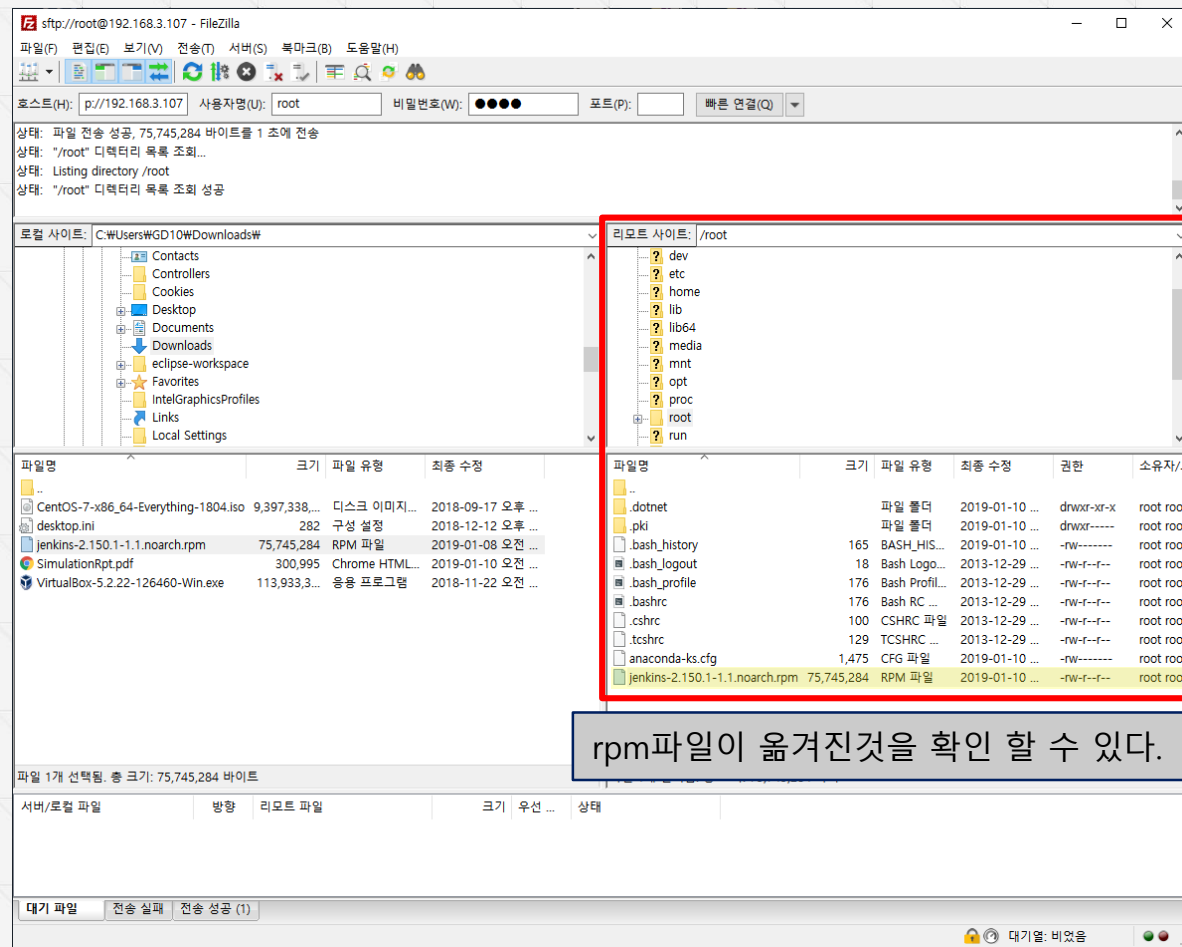
파일질라 프로그램을 이용하여 다운받은 rpm파일을 linux로 옮겨줄것이다.
호스트(H): stfp//리눅스ip, 사용자명(U), 비밀번호(W)를 입력하여 접속을 한다.

2



3

왼쪽이 window에 있는 파일이고, 오른쪽이 linux에 있는 파일이다.
왼쪽에 다운받은 rpm파일을 오른쪽의 /root 경로에 드래그 앤드 드롭으로 끌어다 놓는다.



rpm파일이 옮겨진것을 확인 할 수 있다.

```
root@localhost~  
[root@localhost ~]# ls  
anaconda-ks.cfg  jenkins-2.150.1-1.1.noarch.rpm  
[root@localhost ~]# yum localinstall -y jenkins-2.150.1-1.1.noarch.rpm
```

1 `ls` 명령어를 통해 파일이 있는지 확인하고,
`yum localinstall -y [파일명]` 명령어를 이용하여 설치를 해준다.

```
=====
Size
=====
noarch 72 M

Transaction Summary
=====
Install 1 Package

Total size: 72 M
Installed size: 72 M
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : jenkins-2.150.1-1.1.noarch 1/1
  Verifying : jenkins-2.150.1-1.1.noarch 1/1

Installed:
  jenkins.noarch 0:2.150.1-1.1

Complete!
[root@localhost ~]# rpm -qa | grep jenkins
jenkins-2.150.1-1.1.noarch
[root@localhost ~]#
```

2 `rpm -qa | grep jenkins` 명령어를 입력해준다.
jenkins가 설치되었다면 아래와 같이 출력이 된다.

05-3 Software Install_Jenkins

```
root@localhost:~  
[root@localhost ~]# yum install -y java  
java-1.8.0-openjdk.x86_64 1:1.8.0.191.b12-1.el7_6  
Dependency Installed:  
copy-jdk-configs.noarch 0:3.3-10.el7_5  
dejavu-fonts-common.noarch 0:2.33-6.el7  
fontconfig.x86_64 0:2.13.0-4.3.el7  
fontpackages-filesystem.noarch 0:1.44-8.el7  
java-1.8.0-openjdk-headless.x86_64 1:1.8.0.191.b12-1.el7_6  
libICE.x86_64 0:1.0.9-9.el7  
libX11.x86_64 0:1.6.5-2.el7  
libXau.x86_64 0:1.0.8-2.1.el7  
libXext.x86_64 0:1.3.3-3.el7  
libXrender.x86_64 0:0.9.10-1.el7  
libfontenc.x86_64 0:1.1.3-3.el7  
libxcb.x86_64 0:1.13-1.el7  
lksctp-tools.x86_64 0:1.0.17-2.el7  
python-lxml.x86_64 0:3.2.1-4.el7  
tzdata-java.noarch 0:2018g-1.el7  
xorg-x11-fonts-Type1.noarch 0:7.5-9.el7  
dejavu-fonts-common.noarch 0:2.33-6.el7  
fontconfig.x86_64 0:2.13.0-4.3.el7  
giflib.x86_64 0:4.1.6-9.el7  
javapackages-tools.noarch 0:3.4.1-11.el7  
libSM.x86_64 0:1.2.2-2.el7  
libX11-common.noarch 0:1.6.5-2.el7  
libXcomposite.x86_64 0:0.4.4-4.1.el7  
libXi.x86_64 0:1.7.9-1.el7  
libXtst.x86_64 0:1.2.3-1.el7  
libjpeg-turbo.x86_64 0:1.2.90-6.el7  
libxslt.x86_64 0:1.1.28-5.el7  
python-javapackages.noarch 0:3.4.1-11.el7  
ttmkfdir.x86_64 0:3.0.9-42.el7  
xorg-x11-font-utils.x86_64 1:7.5-21.el7  
Complete!  
[root@localhost ~]# rpm -qa | grep java  
python-javapackages-3.4.1-11.el7.noarch  
tzdata-java-2018g-1.el7.noarch  
java-1.8.0-openjdk-1.8.0.191.b12-1.el7_6.x86_64  
javapackages-tools-3.4.1-11.el7.noarch  
java-1.8.0-openjdk-headless-1.8.0.191.b12-1.el7_6.x86_64  
[root@localhost ~]#
```

1

jenkins는 java기반으로 운영되기 때문에 java설치도 필요하다.
yum install -y java 명령어를 이용하여 설치를 해준다.

2

rpm -qa | grep java 명령어를 입력해준다.
java가 설치되었다면 이와 같이 출력이 된다.


```
root@localhost:~  
[root@localhost ~]# firewall-cmd --zone=public --add-port=8080/tcp --permanent  
success  
[root@localhost ~]# firewall-cmd --reload  
success  
[root@localhost ~]# systemctl start jenkins  
[root@localhost ~]# systemctl status jenkins  
● jenkins.service - LSB: Jenkins Automation Server  
   Loaded: loaded (/etc/rc.d/init.d/jenkins; bad; vendor preset: disabled)  
   Active: active (running) since 2019-01-10 14:49:08 KST; 6s ago  
     Docs: man:systemd-sysv-generator(8)  
  Process: 3521 ExecStart=/etc/rc.d/init.d/jenkins start (code=exited, status=0/SUCCESS)  
    CGroup: /system.slice/jenkins.service  
            └─3540 /etc/alternatives/java -Dcom.sun.akuma.Daemon=daemonized -Djava.awt.headless=true -DJENKI...  
  
1월 10 14:49:06 localhost.localdomain systemd[1]: Starting LSB: Jenkins Automation Server...  
1월 10 14:49:06 localhost.localdomain runuser[3526]: pam_unix(runuser:session): session opened for use...=0)  
1월 10 14:49:08 localhost.localdomain jenkins[3521]: Starting Jenkins [ OK ]  
1월 10 14:49:08 localhost.localdomain systemd[1]: Started LSB: Jenkins Automation Server.  
Hint: Some lines were ellipsized, use -l to show in full.
```

jenkins는 8080 의 방화벽을 사용한다.

따라서 **firewall-cmd --zone=public --add-port=8080/tcp --permanent** 를 입력하여 방화벽을 오픈시켜준다.

firewall-cmd --reload 방화벽을 재로드 시킨다.

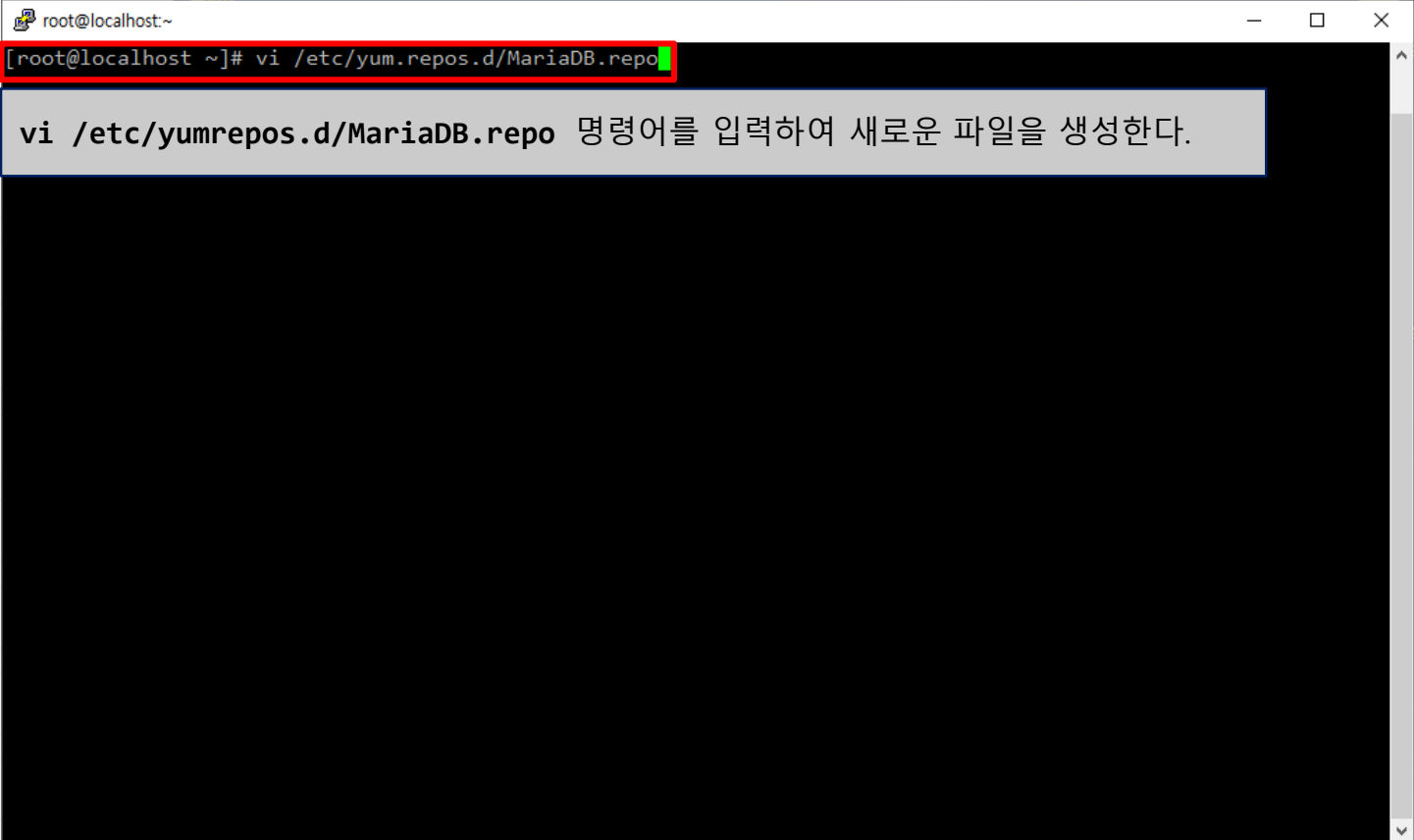
systemctl start jenkins 명령어를 이용하여 jenkins를 시작시켜준다.

systemctl status jenkins 명령어를 이용하여 jenkins의 상태를 확인한다.

현재 jenkins를 시작했으므로 active (running)이 뜬것을 확인 할 수 있다.

5-4.

Software Install_Mariadb

A terminal window titled 'root@localhost~' with standard window controls. The command '[root@localhost ~]# vi /etc/yum.repos.d/MariaDB.repo' is entered and highlighted with a red box. A grey tooltip box contains the Korean text 'vi /etc/yumrepos.d/MariaDB.repo 명령어를 입력하여 새로운 파일을 생성한다.'.

```
root@localhost~
```

```
[root@localhost ~]# vi /etc/yum.repos.d/MariaDB.repo
```

vi /etc/yumrepos.d/MariaDB.repo 명령어를 입력하여 새로운 파일을 생성한다.

2

파일을 생성하면 아무것도 없는 빈 파일이 뜬다.
위와같이 입력해주고 저장해주고 나가면 된다.

2

2

1

1

2

2

2

1

2

2

2

2

1

2

2

2

1

1

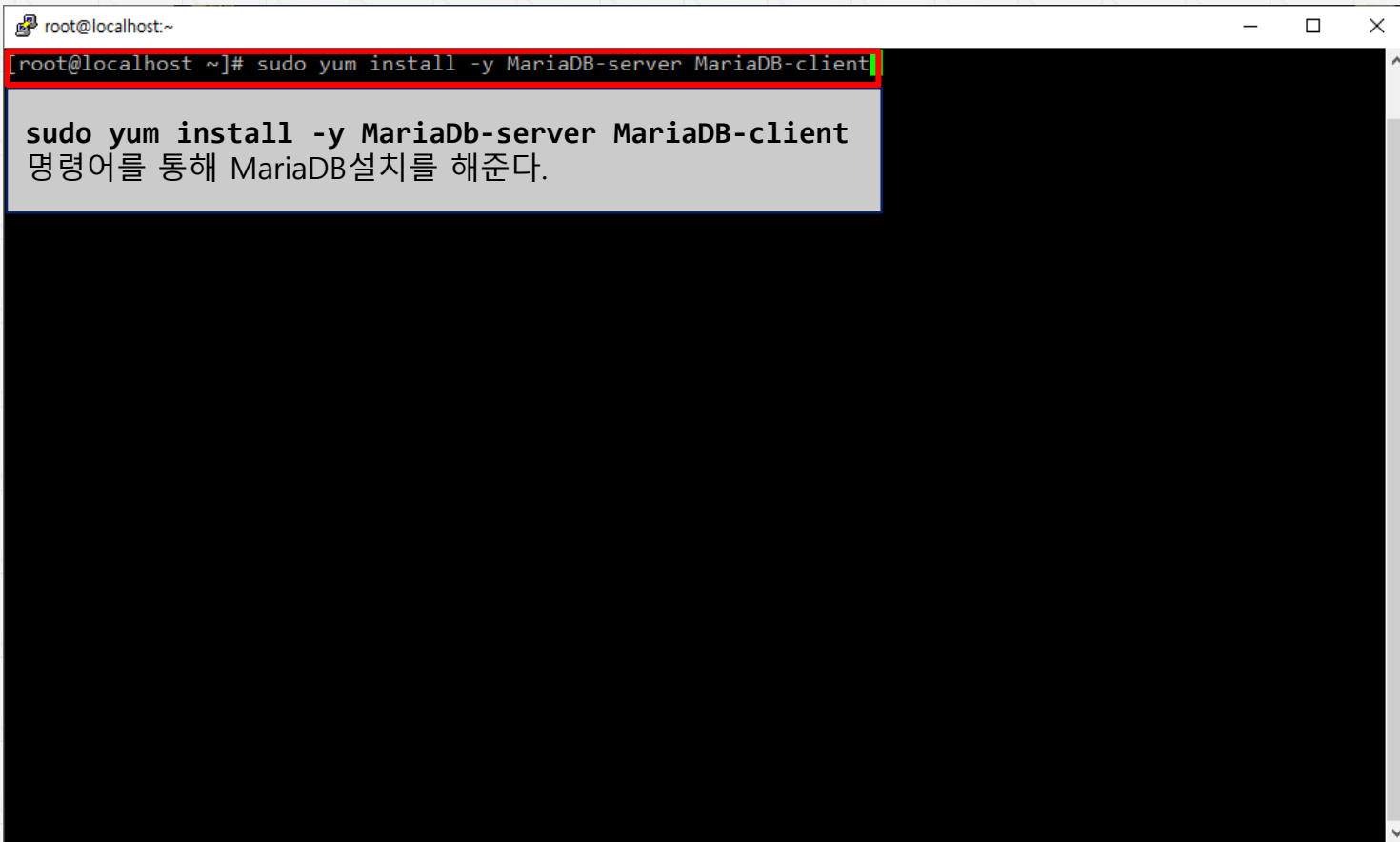
2

2

2



1

1



A terminal window titled 'root@localhost:~' with standard window controls. The command `sudo yum install -y MariaDB-server MariaDB-client` is entered and highlighted with a red background. Below it, the same command is repeated on a light gray background, followed by the Korean text '명령어를 통해 MariaDB설치를 해준다.' (Install MariaDB using the command).

```
root@localhost:~  
[root@localhost ~]# sudo yum install -y MariaDB-server MariaDB-client  
  
sudo yum install -y MariaDb-server MariaDB-client  
명령어를 통해 MariaDB설치를 해준다.
```

 root@localhost:~
[root@localhost ~]# vi /etc/my.cnf

Mariadb를 사용할 때 UTF-8 설정을 해야지만 한국어 사용이 가능하다.
vi /etc/my.cnf 명령어를 통해 my.cnf라는 파일을 편집한다.

원래 있던 내용을 지운 다음 위와 같이 저장을 해주고 나간다.

~~~~~

```

root@localhost:~
root@localhost ~]# systemctl start mariadb
root@localhost ~]# systemctl status mariadb
● mariadb.service - MariaDB 10.3.12 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/mariadb.service.d
            └─migrated-from-my.cnf-settings.conf
   Active: active (running) since 목 2019-01-10 15:08:55 KST; 9s ago
     Docs: man:mysqld(1)

systemctl start mariadb 명령어를 이용하여 mariadb를 시작한다.
systemctl status mariadb 명령어를 이용하여 mariadb의 상태를 확인 할 수 있다.
mariadb가 실행되었다면 active(running) 라는 표시가 뜬다.

=0/SUCCESS)
Main PID: 4049 (mysqld)
  Status: "Taking your SQL requests now..."
   CGroup: /system.slice/mariadb.service
           └─4049 /usr/sbin/mysqld

1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] InnoDB: 10.3.12 start... 21
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] InnoDB: Loading buffe...ool
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] InnoDB: Buffer pool(s...:55
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] Plugin 'FEEDBACK' is ...ed.
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] Server socket created...'.
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] Reading of all Master...ded
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] Added new Master_info...ble
1월 10 15:08:55 localhost.localdomain mysqld[4049]: 2019-01-10 15:08:55 0 [Note] /usr/sbin/mysqld: rea...ns.
1월 10 15:08:55 localhost.localdomain mysqld[4049]: Version: '10.3.12-MariaDB' socket: '/var/lib/mysq...ver

```

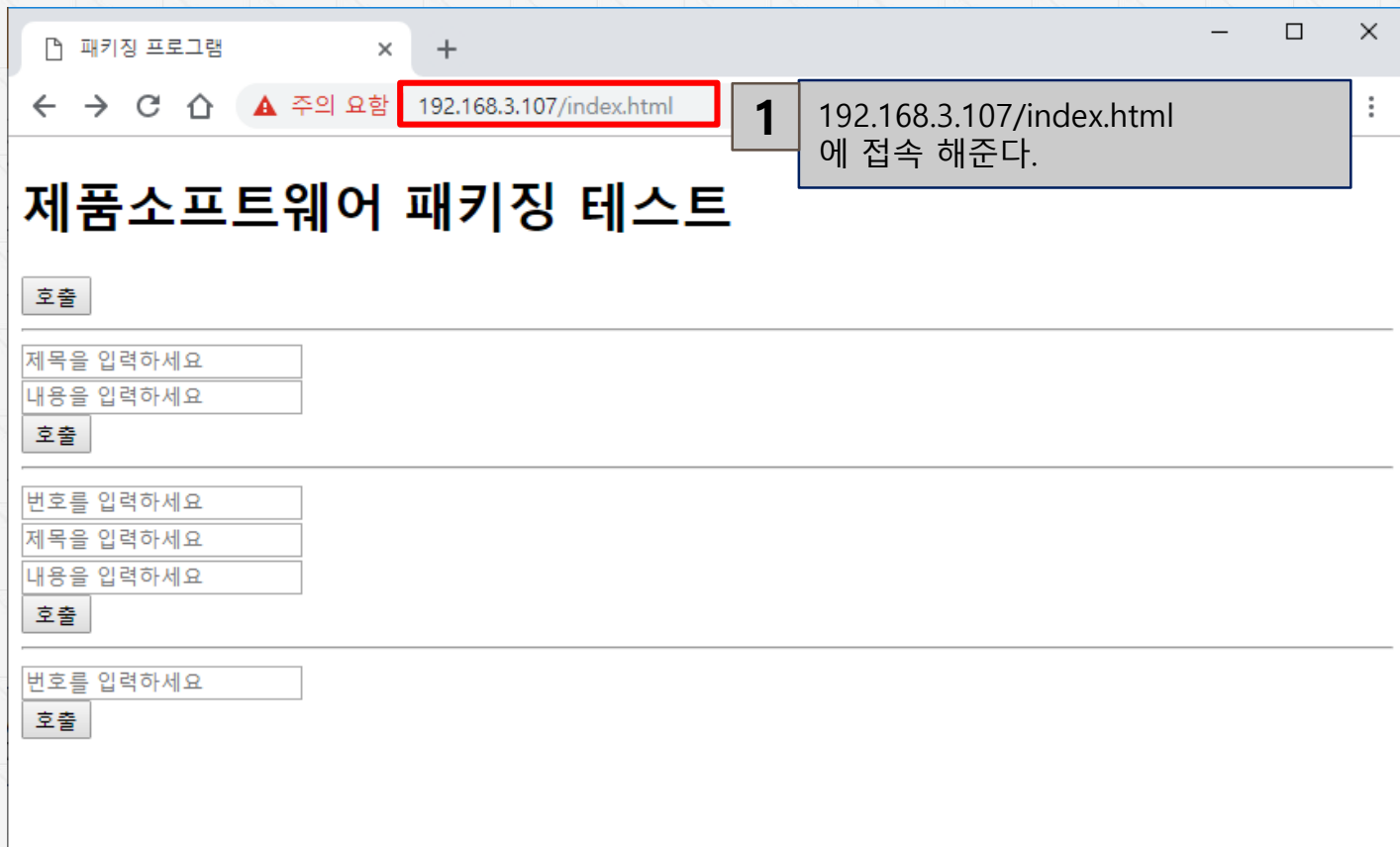


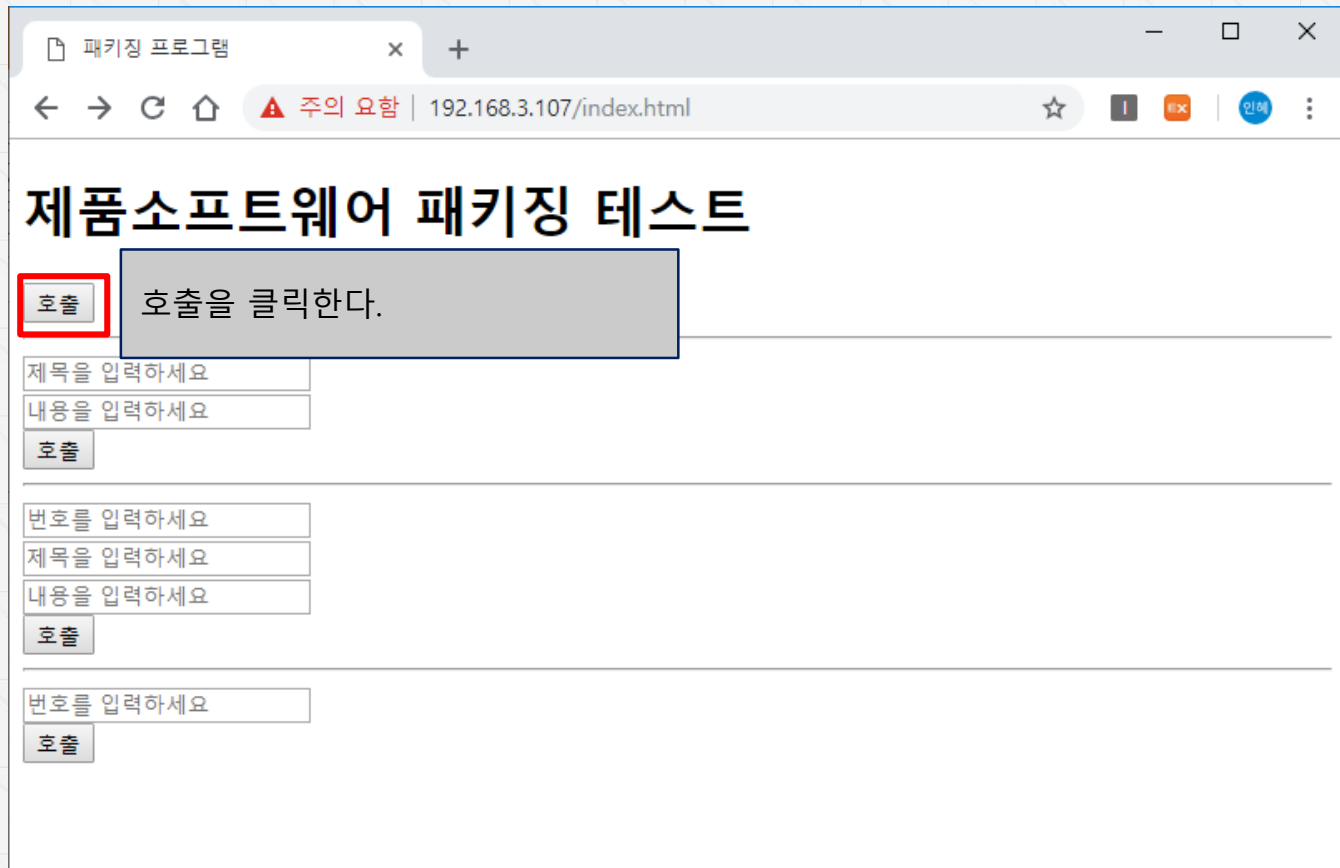
6.

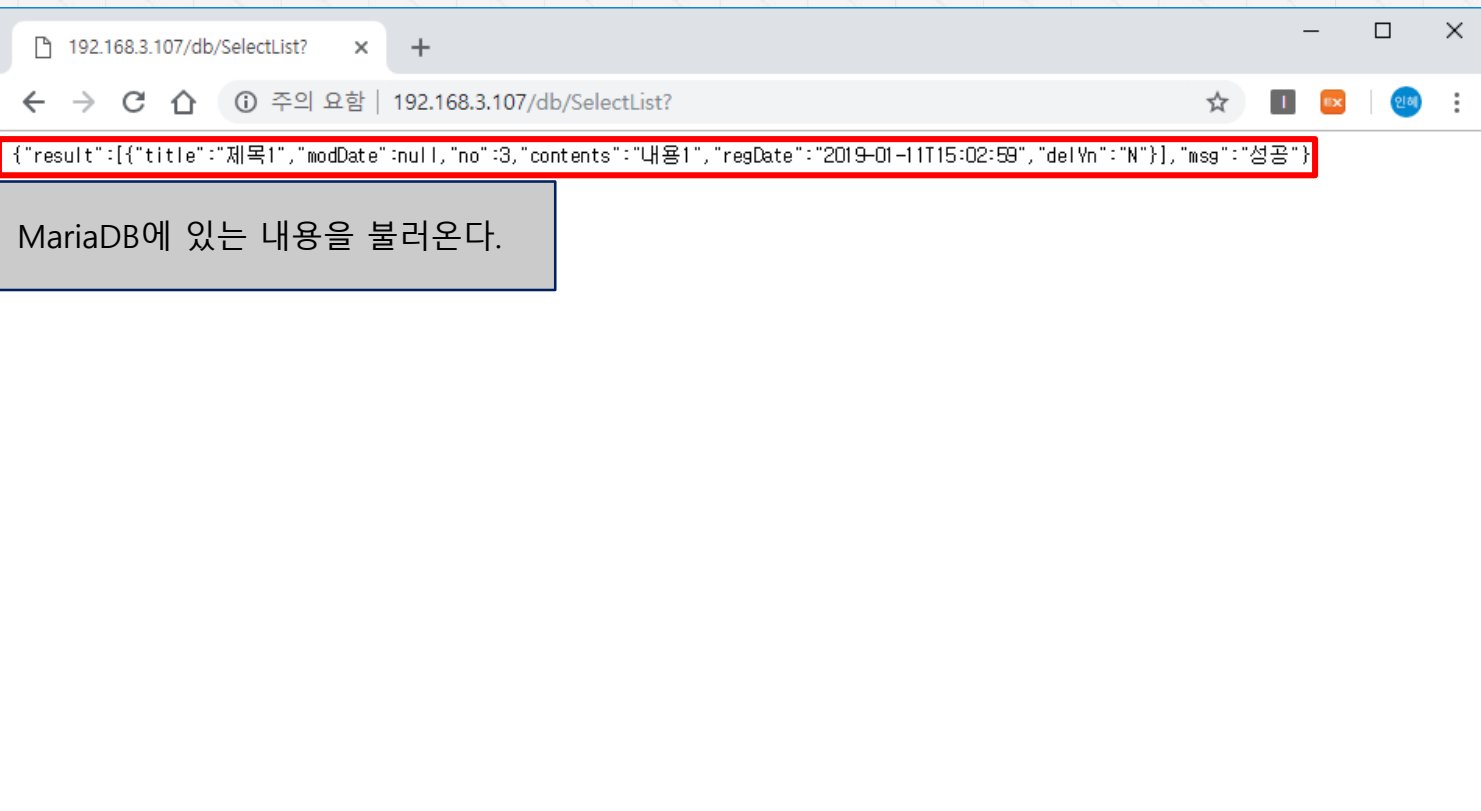
## CRUD 사용 가이드











패키징 프로그램 x +

← → ↺ ⬆ ⚠ 주의 요함 | 192.168.3.107/index.html ☆ ⓘ Ⓜ 인맥 ⋮

## 제품소프트웨어 패키징 테스트

호출

제목1

내용1

호출

번호를 입력하세요

제목을 입력하세요

내용을 입력하세요

호출

번호를 입력하세요

호출

1

제목과 내용을 입력해주고 호출을 클릭하면  
성공 메시지가 뜨면서 작성한 내용이  
MariaDB에 들어가게 된다.

2

192.168.3.107/db/Insert x +

← → ↺ ⬆ ⓘ 주의 요함 | 192.168.3.107/db/Insert

```
{"result":1,"msg":"성공"}
```

패키징 프로그램

← → ↺ ⌂ ⚠ 주의 요함 | 192.168.3.107/index.html ☆ ⓘ 🔍 인헤 ⋮

## 제품소프트웨어 패키징 테스트

호출

제목을 입력하세요

내용을 입력하세요

호출

번호를 입력하세요

제목을 입력하세요

내용을 입력하세요

호출

번호를 입력하세요

호출

수정할 번호 입력하고,  
수정할 제목과 수정할 내용을 입력하고  
호출을 클릭하면 번호에 해당하는 내용에 대한것이 수정된다.

패키징 프로그램 x +

← → ↻ 🏠 ⚠ 주의 요함 | 192.168.3.107/index.html ☆ ⓘ 📄 인제

## 제품소프트웨어 패키징 테스트

호출

제목을 입력하세요

내용을 입력하세요

호출

번호를 입력하세요

제목을 입력하세요

내용을 입력하세요

호출

번호를 입력하세요

호출

삭제하고 싶은 번호를 입력하고 호출을 클릭하면 그에 해당하는 번호가 삭제된다.

## Contact Infromation

---

**Inhye Jeong**

**스마트팩토리 양성과정**

**Email    inhyea36@naver.com**

**Mobile   010 4157 6017**

**Office    02 2108 5900**

---