

응용 SW 기초기술 활용

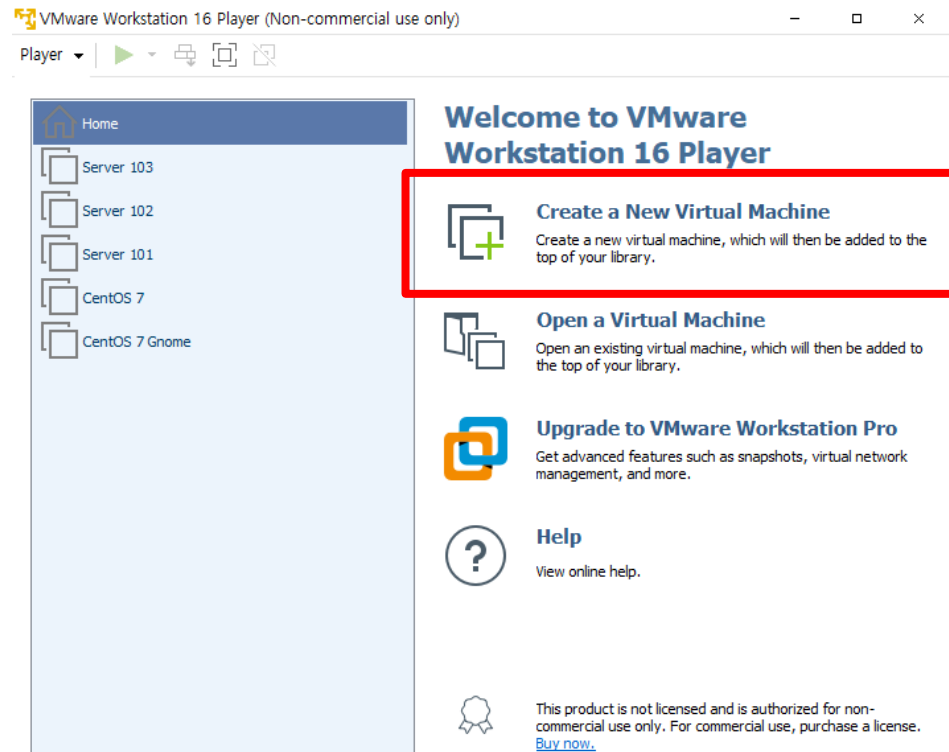
기본 개발환경 구축하기

수행 순서

1. 새로운 가상머신 생성 (가상머신 이름은 Linux Server)
2. 가상머신에 리눅스(CentOS 7) 설치 (최소 설치 버전으로 설치)
3. 쉘 환경에서 Java "HelloWorld" 예제 출력

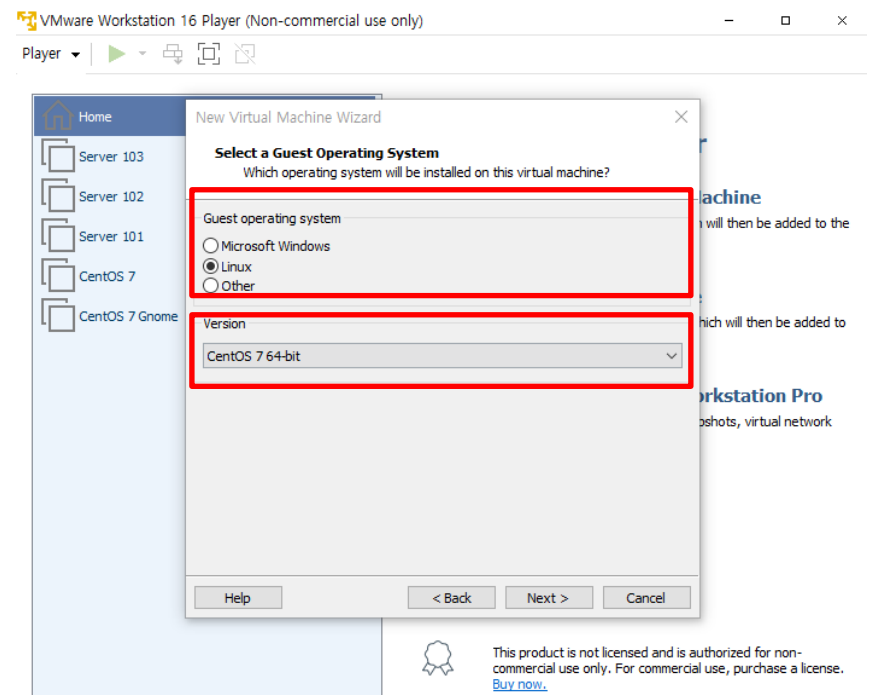
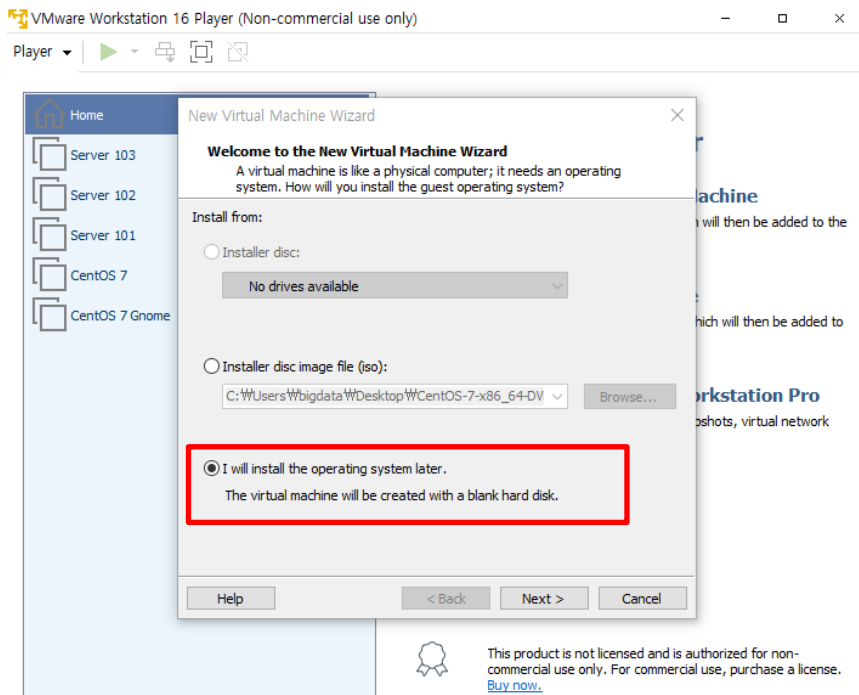
1) 새로운 가상머신 생성

- VMware 웹사이트를 통해 무료 버전인 VMware Workstation Player를 다운받는다. (다운로드 경로: <https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html>)
- VMware를 통해 가상머신을 생성한다. Home에 Create a New Virtual Machine을 클릭한다.



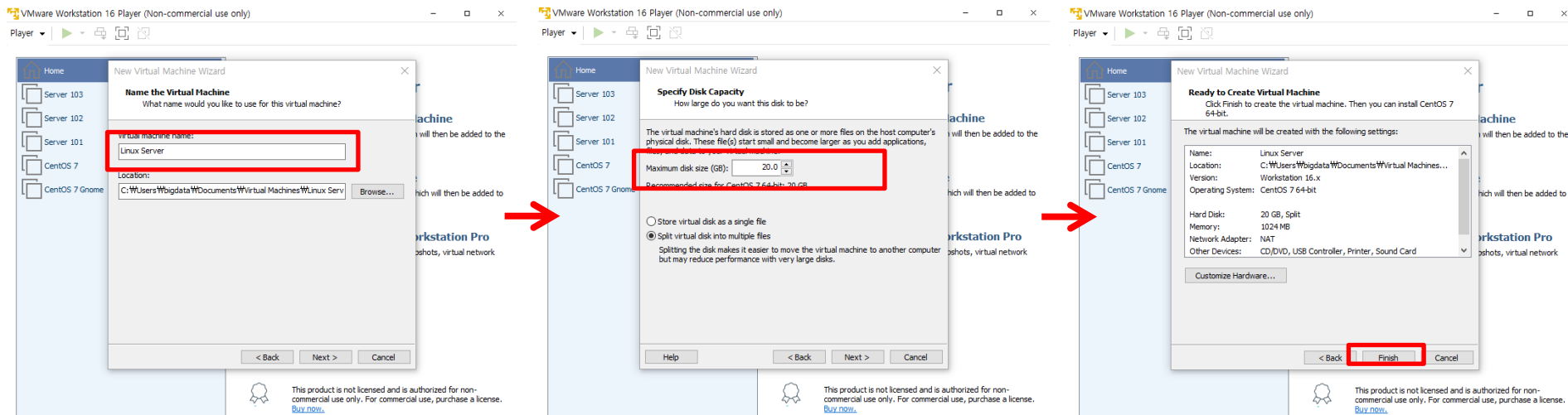
1) 새로운 가상머신 생성

- 가상머신 생성 뒤 iso파일을 이용해 OS설치하기 위해 먼저 마지막 옵션인 'I will install the operating system later.'를 클릭하고 OS는 Linux 를 클릭하고 설치할 버전인 CentOS 7 64bit 버전을 선택한다.



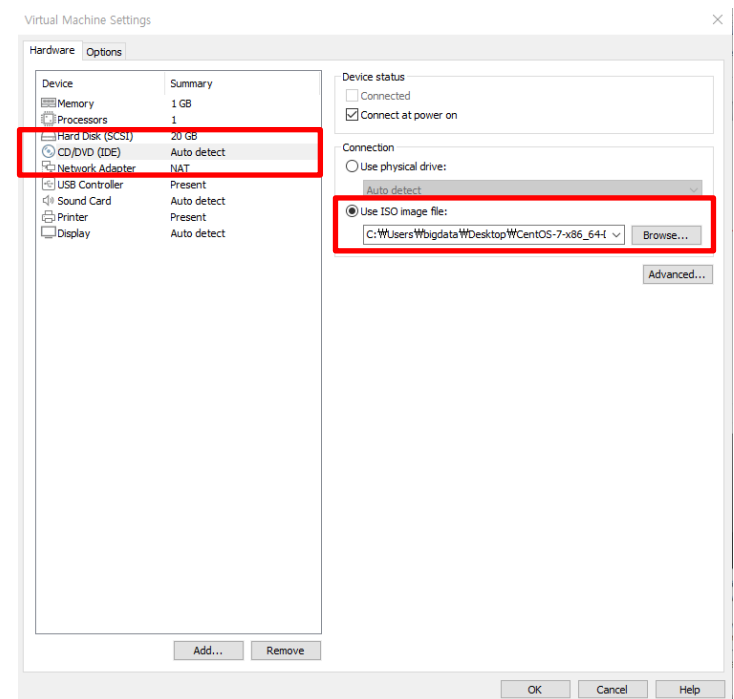
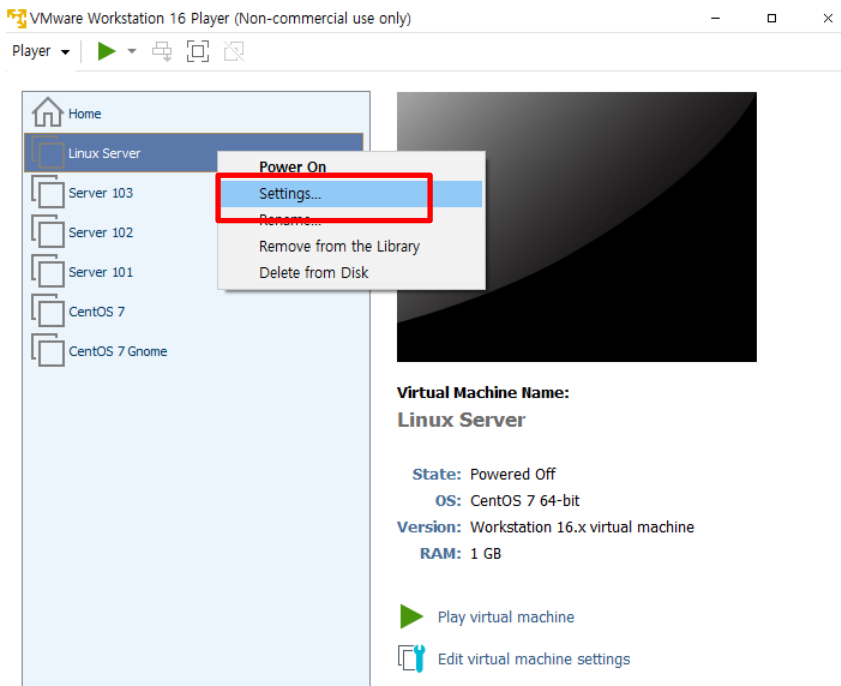
1) 새로운 가상머신 생성

- 가상머신 이름은 'Linux Server' 라고 입력한 후 Next 버튼을 클릭한다.
- 디스크 용량 설정으로 최소 설치버전이기 때문에 추천 용량 사이즈인 20GB로 설정 후 파일형태를 설정하고 Next 버튼을 클릭한다.
- 마지막으로 설정 내용들을 확인하고 Finish 버튼을 클릭하면 가상의 CentOS가 생성된다.



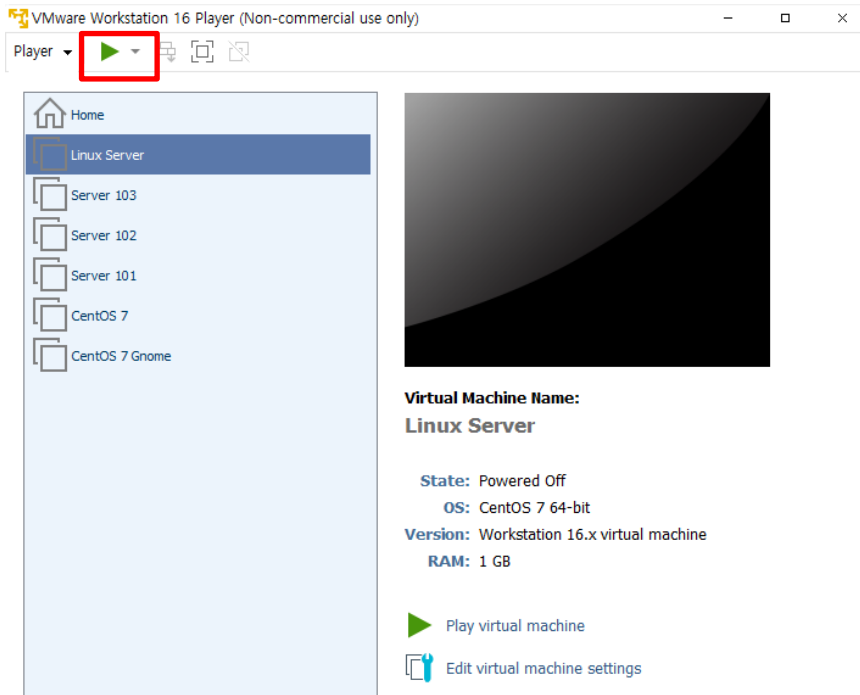
1) 새로운 가상머신 생성

- 가상머신 'Linux Server' 마우스 오른쪽 클릭하여 Settings를 클릭한다.
- 디스크 이미지를 설정하기 위해 CD/DVD(IDE) 를 클릭한 후 Use ISO image file에 다운로드 받은 CentOS 7 ISO 파일을 설정한다. (CentOS 7 ISO 다운로드 경로:
<https://www.centos.org/download/>)



2) 가상머신에 리눅스(CentOS 7) 설치

- Vmware 설정 완료 후 Power on 버튼을 클릭해서 CentOS 7 설치를 진행한다.



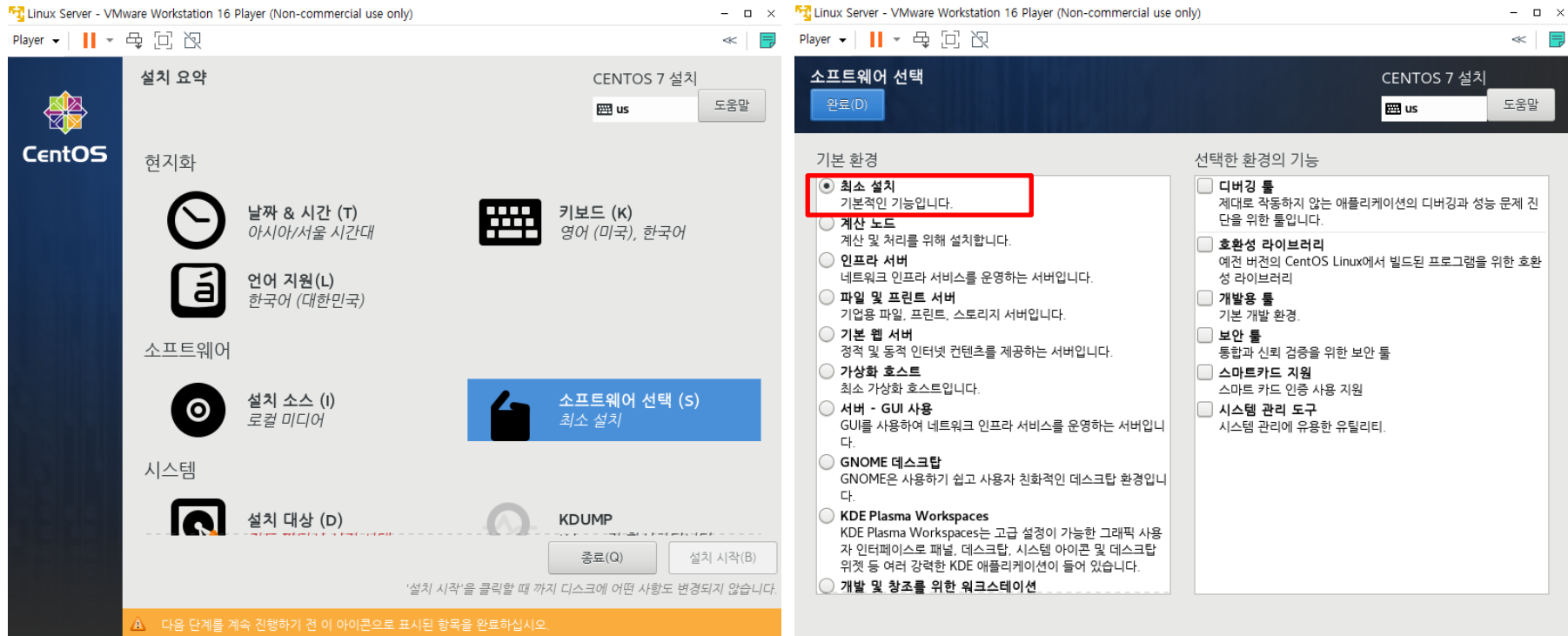
2) 가상머신에 리눅스(CentOS 7) 설치

- 한국어 언어를 선택 후 키보드 설정, 소프트웨어 선택, 설치 대상, KDUMP, 네트워크 및 호스트명을 설정한다.
- 키보드 설정은 영어(미국), 한국어를 선택한다.
- 설치 대상은 자동 파티션으로 설정 후 KDUMP는 비활성화시킨다.
- 네트워크 및 호스트명을 클릭 후 이더넷을 켜다.



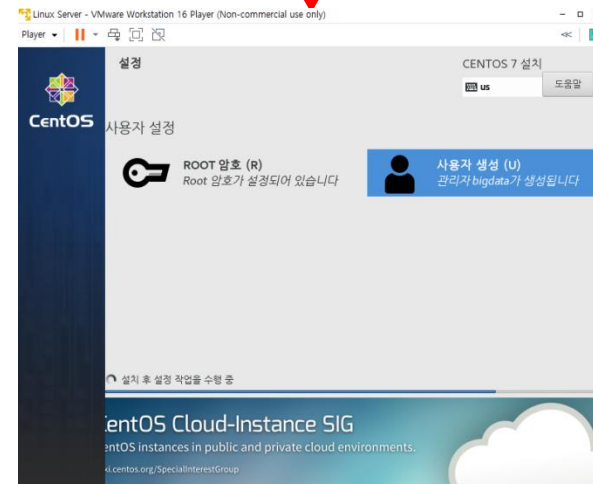
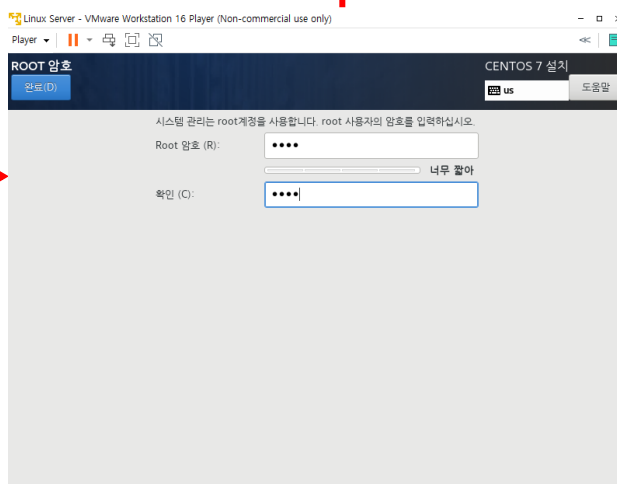
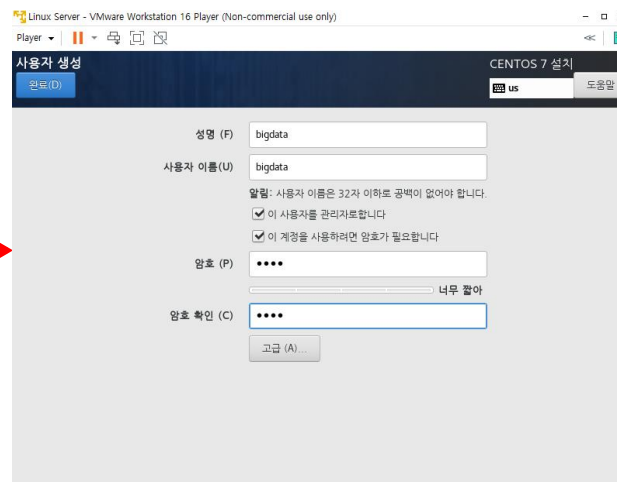
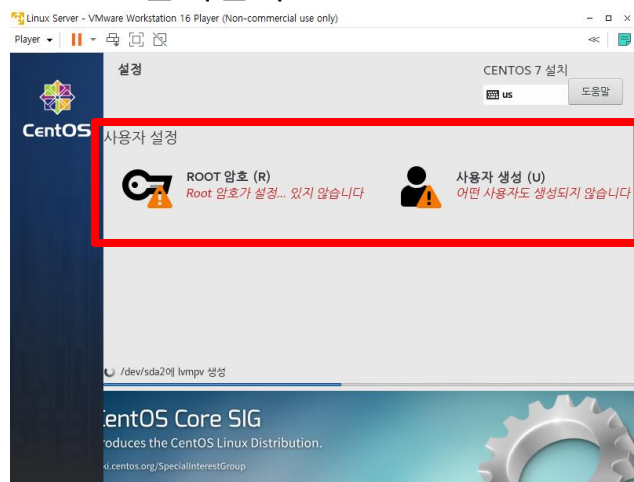
2) 가상머신에 리눅스(CentOS 7) 설치

- 소프트웨어를 선택해서 리눅스 최소 설치버전으로 설치한다.



2) 가상머신에 리눅스(CentOS 7) 설치

- CentOS7 설치가 되는 동안 root 암호 설정과 사용자를 생성한다. 설치가 완료되면 재부팅 버튼을 클릭한다.



3) 쉘 환경에서 Java "HelloWorld" 예제 출력

- 슈퍼 관리자인 root로 로그인 한 후 먼저 #yum update를 통해 패키지 업데이트를 한다.

```
Linux Server - VMware Workstation 16 Player (Non-commercial use only)
Player
CentOS Linux 7 (Core)
Kernel 3.10.0-1160.el7.x86_64 on an x86_64
localhost login: root
Password:
root@localhost ~]# yum update_

Verifying : rsyslog-8.24.0-55.el7.x86_64 65/83
Verifying : systemd-219-78.el7.x86_64 66/83
Verifying : firewallld-0.6.3-11.el7.noarch 67/83
Verifying : device-mapper-persistent-data-0.8.5-3.el7.x86_64 68/83
Verifying : nss-softokn-3.44.0-8.el7_7.x86_64 69/83
Verifying : open-vm-tools-11.0.5-3.el7.x86_64 70/83
Verifying : coreutils-8.22-24.el7.x86_64 71/83
Verifying : tzdata-2020a-1.el7.noarch 72/83
Verifying : libcurl-7.29.0-59.el7.x86_64 73/83
Verifying : kernel-tools-libs-3.10.0-1160.el7.x86_64 74/83
Verifying : curl-7.29.0-59.el7.x86_64 75/83
Verifying : initscripts-9.49.53-1.el7.x86_64 76/83
Verifying : kexec-tools-2.0.15-51.el7.x86_64 77/83
Verifying : systemd-sysv-219-78.el7.x86_64 78/83
Verifying : nss-util-3.44.0-4.el7_7.x86_64 79/83
Verifying : selinux-policy-3.13.1-268.el7.noarch 80/83
Verifying : 1:NetworkManager-libnm-1.18.8-1.el7.x86_64 81/83
Verifying : python-perf-3.10.0-1160.el7.x86_64 82/83
Verifying : iprutils-2.4.17.1-3.el7.x86_64 83/83

Installed:
kernel.x86_64 0:3.10.0-1160.6.1.el7

Updated:
NetworkManager.x86_64 1:1.18.8-2.el7_9
NetworkManager-tui.x86_64 1:1.18.8-2.el7_9
curl.x86_64 0:7.29.0-59.el7_9.1
firewallld-filessystem.noarch 0:0.6.3-12.el7
initscripts.x86_64 0:9.49.53-1.el7_9.1
kernel-tools-libs.x86_64 0:3.10.0-1160.6.1.el7
libcroco.x86_64 0:0.6.12-6.el7_9
nspr.x86_64 0:4.25.0-2.el7_9
nss-softokn-freebl.x86_64 0:3.53.1-6.el7_9
nss-util.x86_64 0:3.53.1-1.el7_9
python-firewall.noarch 0:0.6.3-12.el7
rsyslog.x86_64 0:8.24.0-57.el7_9
systemd.x86_64 0:219-78.el7_9.2
tuned.noarch 0:2.11.0-10.el7
NetworkManager-libnm.x86_64 1:1.18.8-2.el7_9
bind-export-libs.x86_64 32:9.11.4-26.P2.el7_9.2
device-mapper-persistent-data.x86_64 0:0.8.5-3.el7_9.2
freetype.x86_64 0:2.8-14.el7_9.1
iprutils.x86_64 0:2.4.17.1-3.el7_9
kexec-tools.x86_64 0:2.0.15-51.el7_9.1
libcurl.x86_64 0:7.29.0-59.el7_9.1
nss.x86_64 0:3.53.1-3.el7_9
nss-sysinit.x86_64 0:3.53.1-3.el7_9
open-vm-tools.x86_64 0:11.0.5-3.el7_9.1
python-libs.x86_64 0:2.7.5-90.el7
selinux-policy.noarch 0:3.13.1-268.el7_9.2
systemd-libs.x86_64 0:219-78.el7_9.2
tzdata.noarch 0:2020d-2.el7
NetworkManager-team.x86_64 1:1.18.8-2.el7_9
coreutils.x86_64 0:8.22-24.el7_9.2
firewallld.noarch 0:0.6.3-12.el7
glib2.x86_64 0:2.56.1-8.el7
kernel-tools.x86_64 0:3.10.0-1160.6.1.el7
kpartx.x86_64 0:0.4.9-134.el7_9
microcode_ctl.x86_64 2:2.1-73.2.el7_9
nss-softokn.x86_64 0:3.53.1-6.el7_9
nss-tools.x86_64 0:3.53.1-3.el7_9
python.x86_64 0:2.7.5-90.el7
python-perf.x86_64 0:3.10.0-1160.6.1.el7
selinux-policy-targeted.noarch 0:3.13.1-268.el7_9.2
systemd-sysv.x86_64 0:219-78.el7_9.2

Complete!
root@localhost ~]#
```

3) 쉘 환경에서 Java "HelloWorld" 예제 출력

- Java 패키지를 설치할 위해 명령어 `#yum install java-1.8.*` 를 입력 후 Enter를 눌러 설치를 실행한다.

Linux Server - VMware Workstation 16 Player (Non-commercial use only)

Player ▾ | || ▢ ↻ 🔍

- □ ×

◀ ▶

```
root@localhost ~]# yum install java-1.8.*_
```

Installed:

java-1.8.0-openjdk.x86_64 1:1.8.0.272.b10-1.el7_9	java-1.8.0-openjdk-accessibility.x86_64 1:1.8.0.272.b10-1.el7_9
java-1.8.0-openjdk-demo.x86_64 1:1.8.0.272.b10-1.el7_9	java-1.8.0-openjdk-devel.x86_64 1:1.8.0.272.b10-1.el7_9
java-1.8.0-openjdk-headless.x86_64 1:1.8.0.272.b10-1.el7_9	java-1.8.0-openjdk-javadoc.noarch 1:1.8.0.272.b10-1.el7_9
java-1.8.0-openjdk-javadoc-zip.noarch 1:1.8.0.272.b10-1.el7_9	java-1.8.0-openjdk-src.x86_64 1:1.8.0.272.b10-1.el7_9

Dependency Installed:

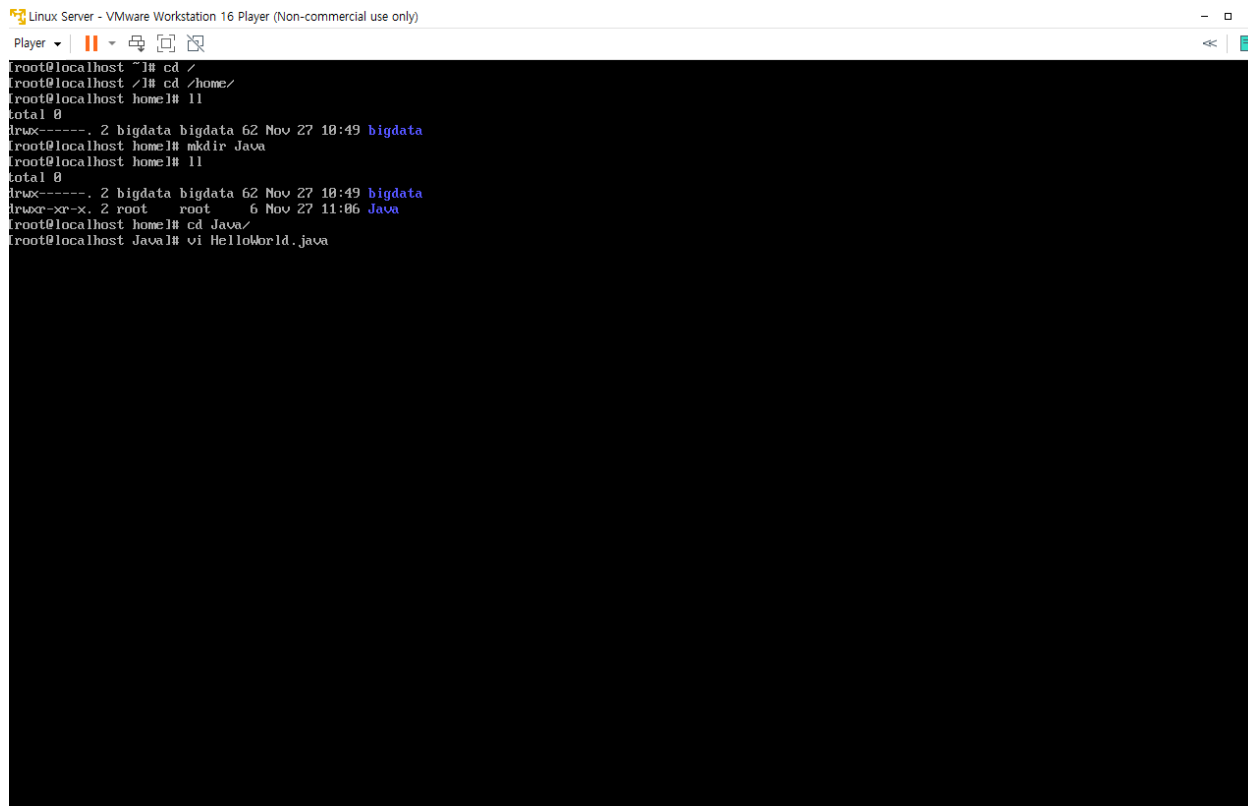
GConf2.x86_64 0:3.2.6-8.el7	atk.x86_64 0:2.28.1-2.el7	avahi-libs.x86_64 0:0.6.31-20.el7
cairo.x86_64 0:1.15.12-4.el7	copy-jdk-configs.noarch 0:3.3-10.el7_5	cups-libs.x86_64 1:1.6.3-51.el7
dejavu-fonts-common.noarch 0:2.33-6.el7	dejavu-sans-fonts.noarch 0:2.33-6.el7	fontconfig.x86_64 0:2.13.0-4.3.el7
fontpackages-filesystem.noarch 0:1.44-8.el7	fribidi.x86_64 0:1.0.2-1.el7_7.1	gdk-pixbuf2.x86_64 0:2.36.12-3.el7
glib.x86_64 0:4.1.6-9.el7	graphite2.x86_64 0:1.3.10-1.el7_3	gtk-update-icon-cache.x86_64 0:3.22.30-6.el7
gtk2.x86_64 0:2.24.31-1.el7	harfbuzz.x86_64 0:1.7.5-2.el7	hicolor-icon-theme.noarch 0:0.12-7.el7
jasper-libs.x86_64 0:1.900.1-33.el7	java-atk-wrapper.x86_64 0:0.30.4-5.el7	javapackages-tools.noarch 0:3.4.1-11.el7
jbigkit-libs.x86_64 0:2.0-11.el7	libICE.x86_64 0:1.0.9-9.el7	libSM.x86_64 0:1.2.2-2.el7
libX11.x86_64 0:1.6.7-3.el7_9	libX11-common.noarch 0:1.6.7-3.el7_9	libXau.x86_64 0:1.0.8-2.1.el7
libXcomposite.x86_64 0:0.4.4-4.1.el7	libXcursor.x86_64 0:1.1.15-1.el7	libXdamage.x86_64 0:1.1.4-4.1.el7
libXext.x86_64 0:1.3.3-3.el7	libXfixes.x86_64 0:5.0.3-1.el7	libXft.x86_64 0:2.3.2-2.el7
libXi.x86_64 0:1.7.9-1.el7	libXinerama.x86_64 0:1.1.3-2.1.el7	libXrandr.x86_64 0:1.5.1-2.el7
libXrender.x86_64 0:0.9.10-1.el7	libXtst.x86_64 0:1.2.3-1.el7	libXv.x86_64 0:1.0.11-1.el7
libXxf86dga.x86_64 0:1.1.4-2.1.el7	libXxf86misc.x86_64 0:1.0.3-7.1.el7	libXxf86vm.x86_64 0:1.1.4-1.el7
libdmx.x86_64 0:1.1.3-3.el7	libfontenc.x86_64 0:1.1.3-3.el7	libglvnd.x86_64 1:1.0.1-0.8.git5baae5.el7
libglvnd-egl.x86_64 1:1.0.1-0.8.git5baae5.el7	libglvnd-glx.x86_64 1:1.0.1-0.8.git5baae5.el7	libjpeg-turbo.x86_64 0:1.2.90-8.el7
libthai.x86_64 0:0.1.14-9.el7	libtiff.x86_64 0:4.0.3-35.el7	libwayland-client.x86_64 0:1.15.0-1.el7
libwayland-server.x86_64 0:1.15.0-1.el7	libxcb.x86_64 0:1.13-1.el7	libxshmfence.x86_64 0:1.2-1.el7
lksctp-tools.x86_64 0:1.0.17-2.el7	mesa-libEGL.x86_64 0:18.3.4-12.el7_9	mesa-libGL.x86_64 0:18.3.4-12.el7_9
mesa-libgbm.x86_64 0:18.3.4-12.el7_9	mesa-libglapi.x86_64 0:18.3.4-12.el7_9	pango.x86_64 0:1.42.4-4.el7_7
pcsc-lite-libs.x86_64 0:1.8.8-8.el7	pixman.x86_64 0:0.34.0-1.el7	psmisc.x86_64 0:22.20-17.el7
python-javapackages.noarch 0:3.4.1-11.el7	python-lxml.x86_64 0:3.2.1-4.el7	ttmkfdir.x86_64 0:3.0.9-42.el7
tzdata-java.noarch 0:2020d-2.el7	xorg-x11-font-utils.x86_64 1:7.5-21.el7	xorg-x11-fonts-Type1.noarch 0:7.5-9.el7
xorg-x11-utils.x86_64 0:7.5-23.el7		

Complete!

```
root@localhost ~]#
```

3) 쉘 환경에서 Java "HelloWorld" 예제 출력

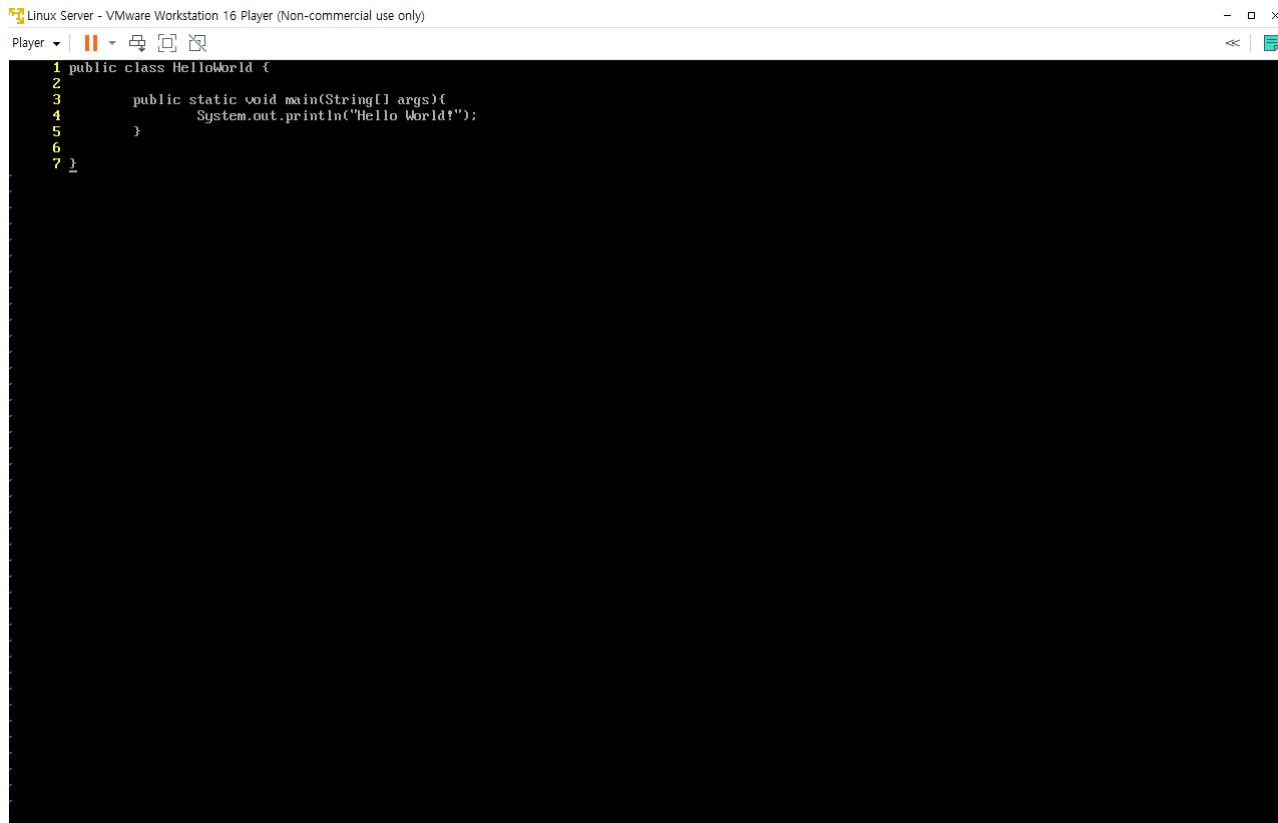
- Home에서 명령어 mkdir를 이용해 새로운 디렉터리 Java를 생성한다.
- Java 디렉터리에서 vi 에디터를 이용해 .java 확장자 파일을 생성한다. #vi HelloWorld.java 를 입력한다.



```
Linux Server - VMware Workstation 16 Player (Non-commercial use only)
Player
root@localhost ~# cd /
root@localhost /# cd /home/
root@localhost home# cd ~
total 0
drwxr-xr-x. 2 bigdata bigdata 62 Nov 27 10:49 bigdata
root@localhost home# mkdir Java
root@localhost home# cd Java/
total 0
drwxr-xr-x. 2 root root 6 Nov 27 11:06 Java
root@localhost home# cd Java/
root@localhost Java# vi HelloWorld.java
```

3) 쉘 환경에서 Java "HelloWorld" 예제 출력

- Insert 버튼을 누르면 '- INSERT -' 로 바뀌고 입력모드로 전환 된다. 편집을 하고 Esc를 누르면 명령 모드로 돌아간다. 다음과 같이 Java 예제를 입력한다. 저장을 하기 위해 명령어 #wq 를 입력한다.



The screenshot shows a terminal window titled "Linux Server - VMware Workstation 16 Player (Non-commercial use only)". The terminal has a black background with yellow text. The text entered is a Java class definition for HelloWorld. The cursor is at the end of the last line. The window title bar includes standard VMware controls and a "Player" dropdown menu.

```
1 public class HelloWorld {  
2  
3     public static void main(String[] args){  
4         System.out.println("Hello World!");  
5     }  
6  
7 }
```

3) 쉘 환경에서 Java "HelloWorld" 예제 출력

- 파일을 저장한 후 Java 컴파일을 위해 `#javac HelloWorld.java` 를 명령어로 입력해 컴파일을 진행한다.
- 클래스 파일이 만들어진 걸 확인 후 `#java HelloWorld` 를 명령어로 입력해 쉘 환경에서 출력이 되는 지 확인한다.

```
"HelloWorld.java" [New] 7L, 112C written
[root@localhost Java]#
[root@localhost Java]#
[root@localhost Java]# javac HelloWorld.java
[root@localhost Java]# ll
total 8
-rw-r--r--. 1 root root 426 Nov 27 11:15 HelloWorld.class
-rw-r--r--. 1 root root 112 Nov 27 11:14 HelloWorld.java
[root@localhost Java]# java HelloWorld
Hello World!
[root@localhost Java]#
```