

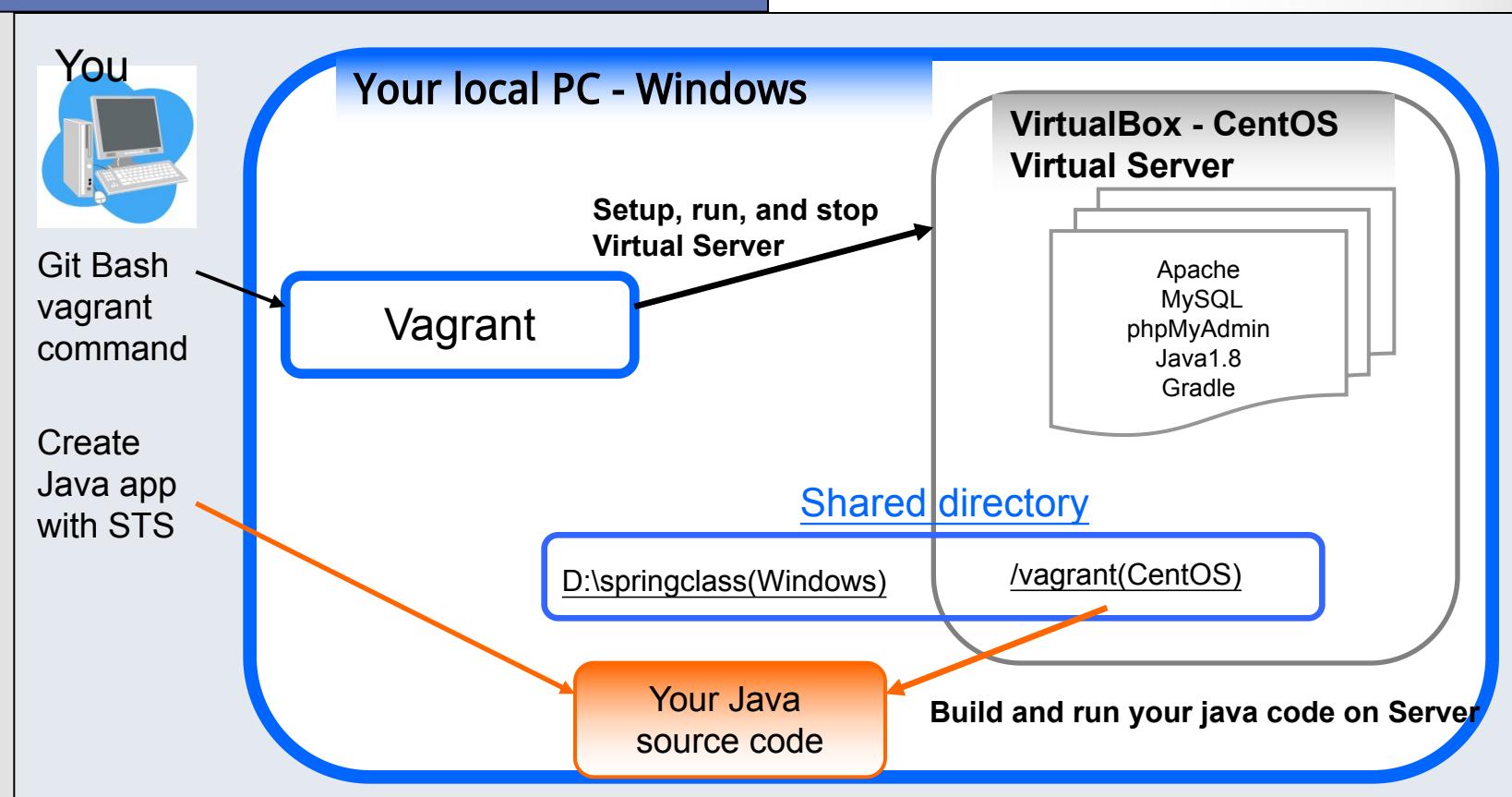


Development environment for SPRING BOOT

Presented by ARMS (THAILAND) Co., Ltd.

Development Environment

Outline for your development environment



Development Environment

Some of the benefit of using this environment

You are not affected by changes in your PC environment such as Java version updates, multiple Java versions, and many build paths. This can provide you with stable development environment.

Not only can you learn how to build and run your Java source in local, but you also can learn how to build and run it on server. This could also work as backup build environment.

You can use both Linux and Windows

Experience cloud environment (although it's locally setup)

Index

1. Download necessary software for the class.

- 1-1. Download
- 1-2. Install VirtualBox
- 1-3 Install vagrant
- 1-4 Install Git Bash

2. Install OS to VM

- 2-1. Download OS image
- 2-2. Initial setup

3. Install IDE in PC

- 3-1. Download STS
- 3-2. local build environment

4. Create a project

- 4-1. Hello in local
- 4-2. Hello in Server

1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-1. Download

- Go to <https://www.virtualbox.org/wiki/Downloads>
and click the link VirtualBox 5.0.24 for Windows hosts [x86/amd64](#)
This VirtualBox-5.0.24-108355-Win.exe will be downloaded.

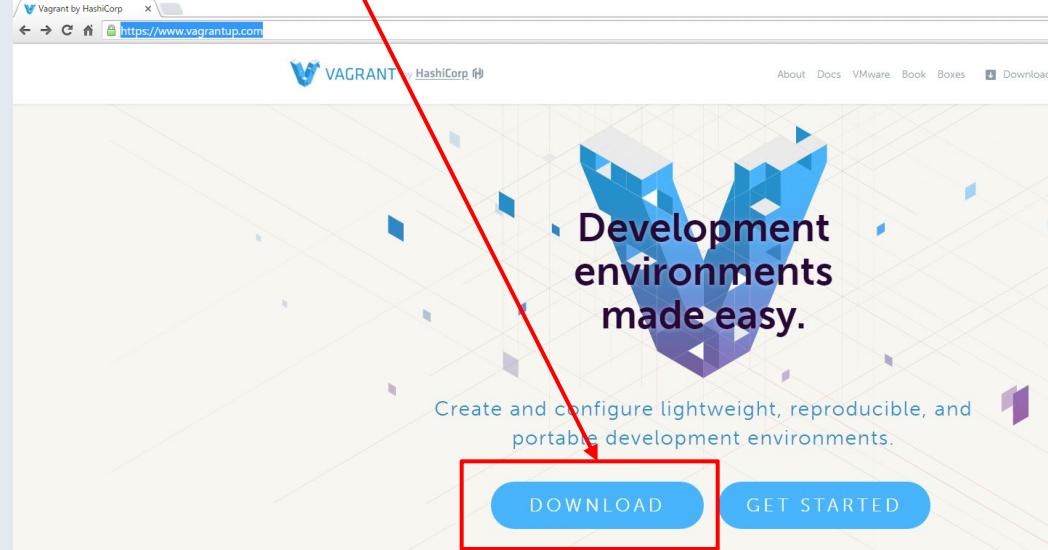


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-1. Download

- Go to <https://www.vagrantup.com/>
and click the DOWNLOAD link on this page

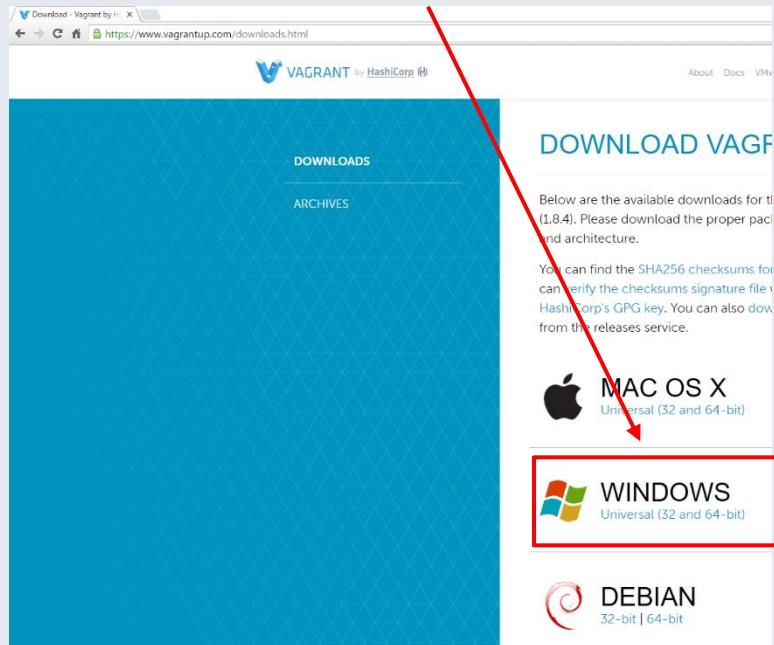


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-1. Download

- Go to <https://www.vagrantup.com/downloads.html>
and click the DOWNLOAD link for WINDOWS on this page



This will download `vagrant_1.8.4.msi`

1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-2. Install VirtualBox

- Double click on VirtualBox-5.0.24-108355-Win.exe



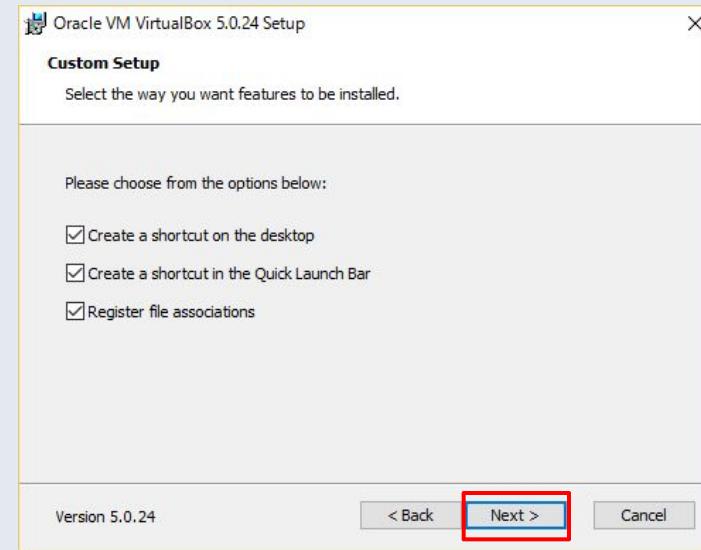
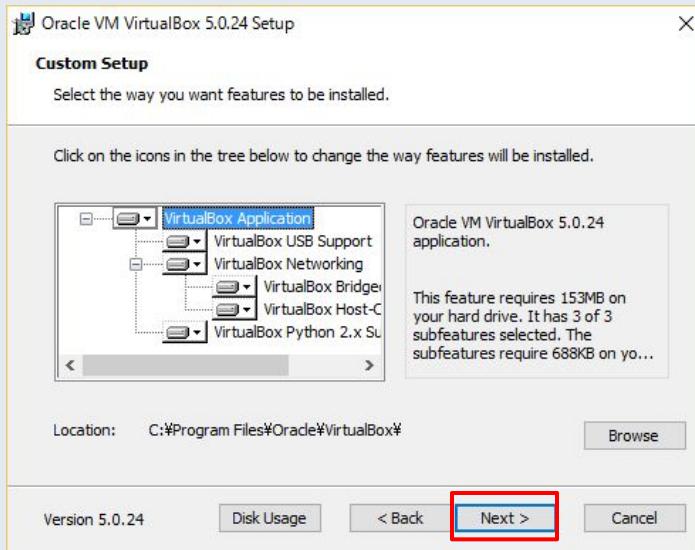
Click Next button

1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-2. Install VirtualBox

- Click Next button to both of the dialogues.

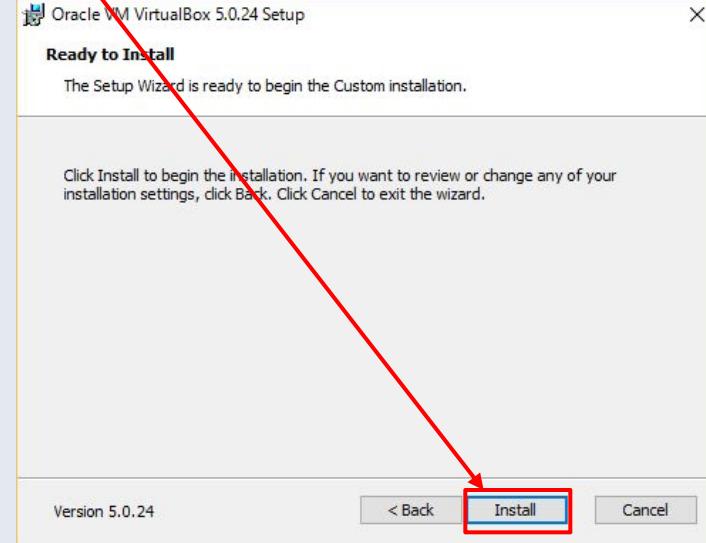
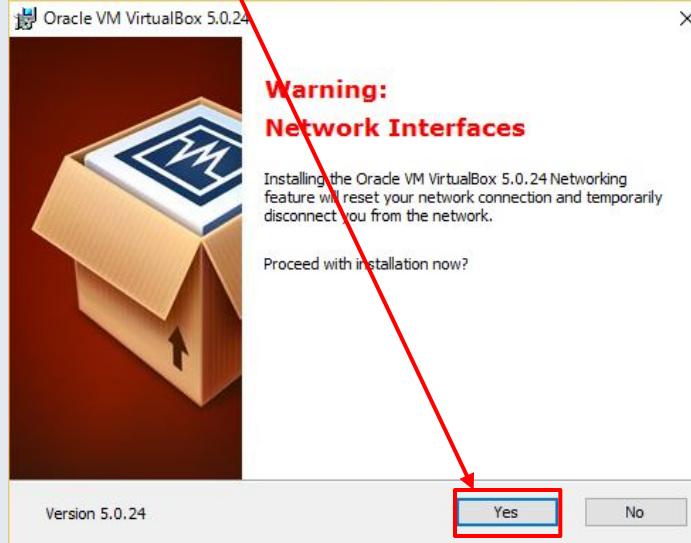


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-2. Install VirtualBox

- Click “Yes” to the Warning and “Install” to the next screen.

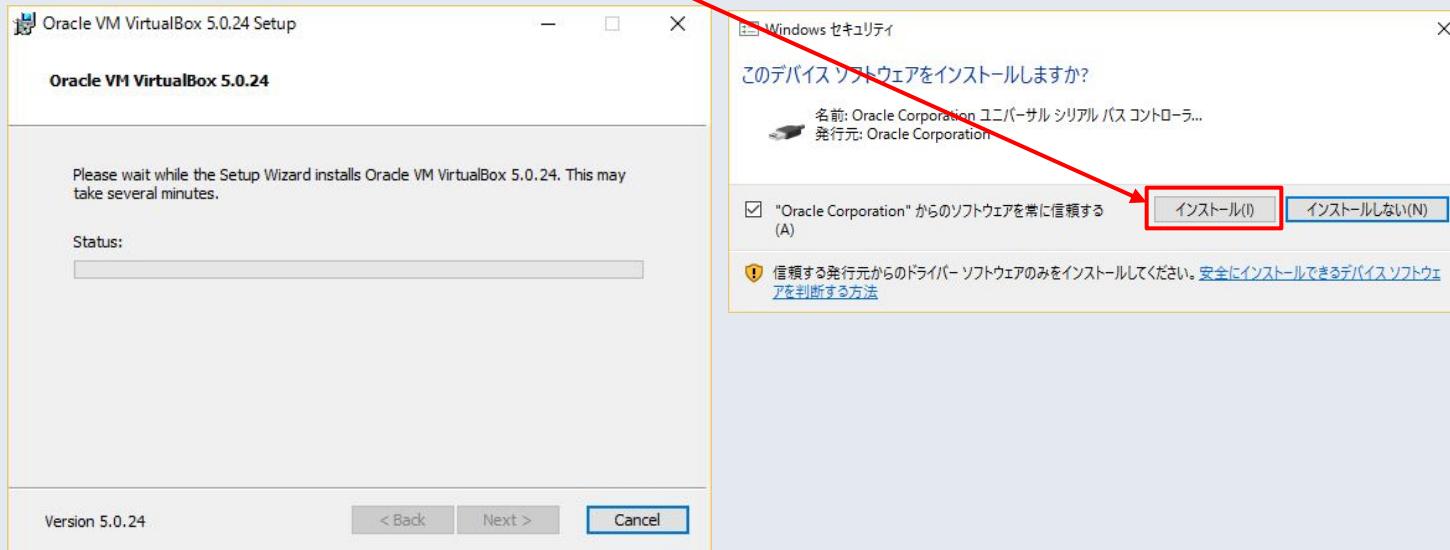


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-2. Install VirtualBox

- Install will start. During the installation, you will be also asked to install USB driver, click “Install” to this dialog.

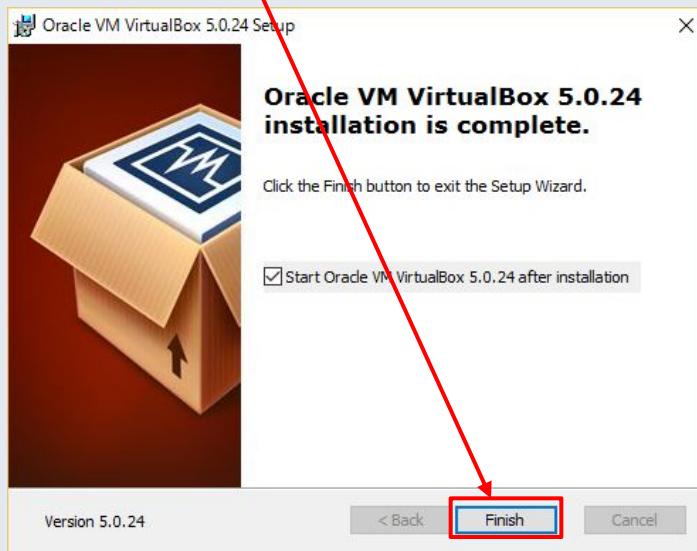


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-2. Install VirtualBox

- Click “Finish” button to exit Setup Wizard.

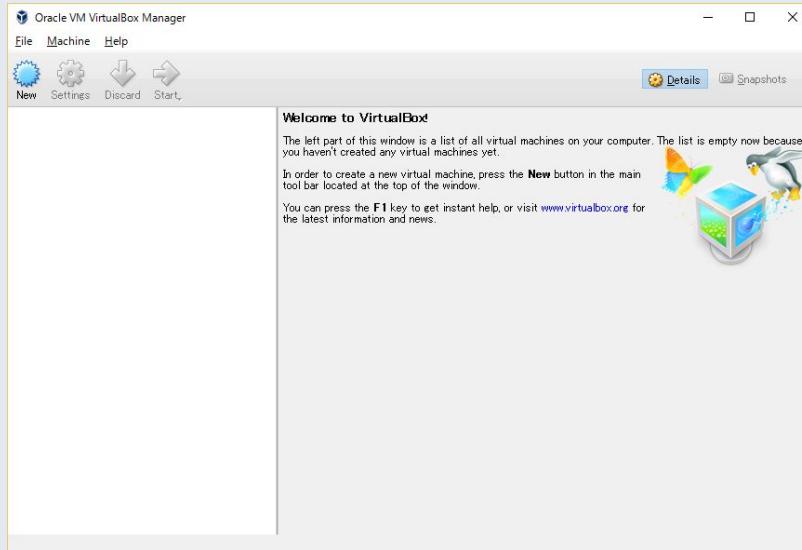


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-2. Install VirtualBox

- If you can see “Oracle VM VirtualBox Manager” window and a shortcut icon to it. Your installation has successfully finished.

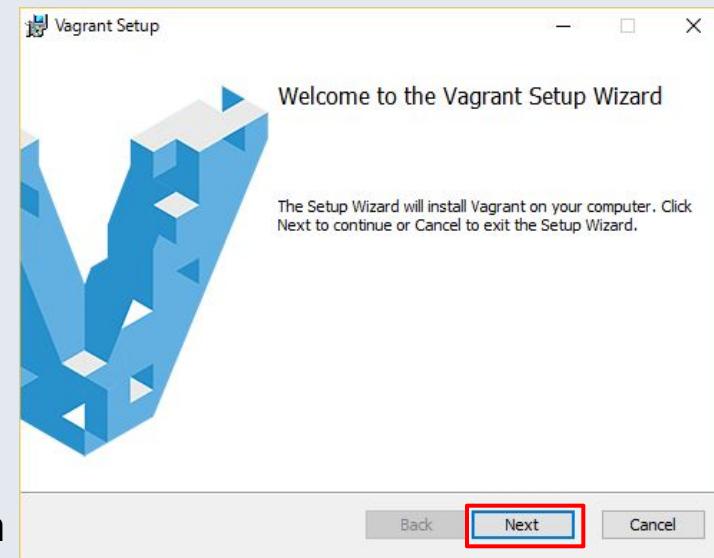
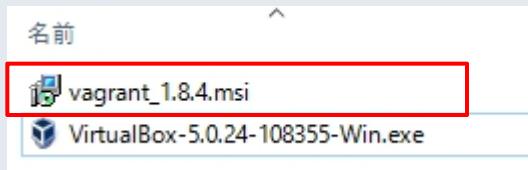


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-3. Install vagrant

- Double click on vagrant_1.8.4.msi



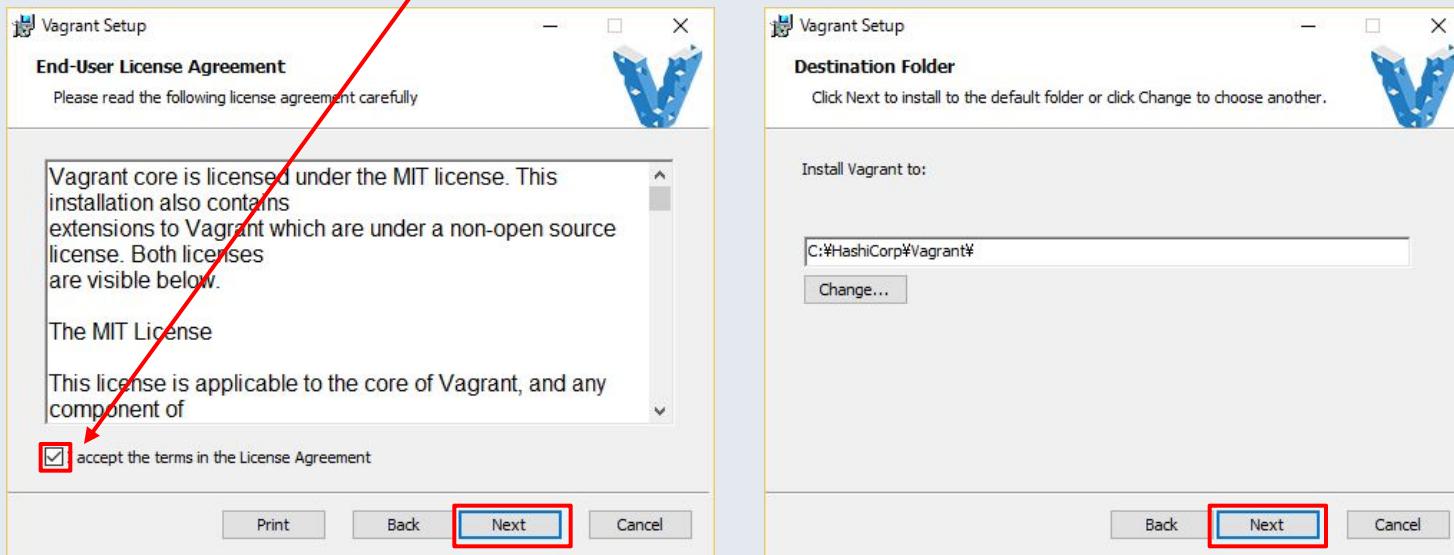
Click Next button

1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-3. Install vagrant

- On EULA dialog, check to accept the LA and click “Next” button, and then to the Destination folder as default, click “Next” button.

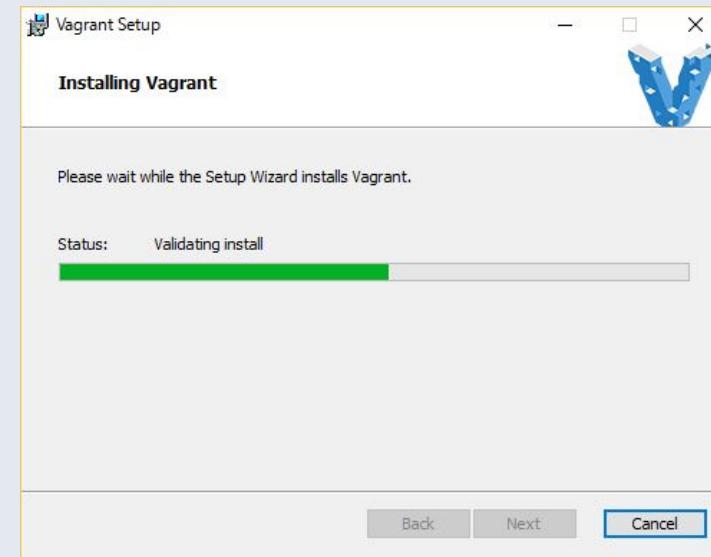
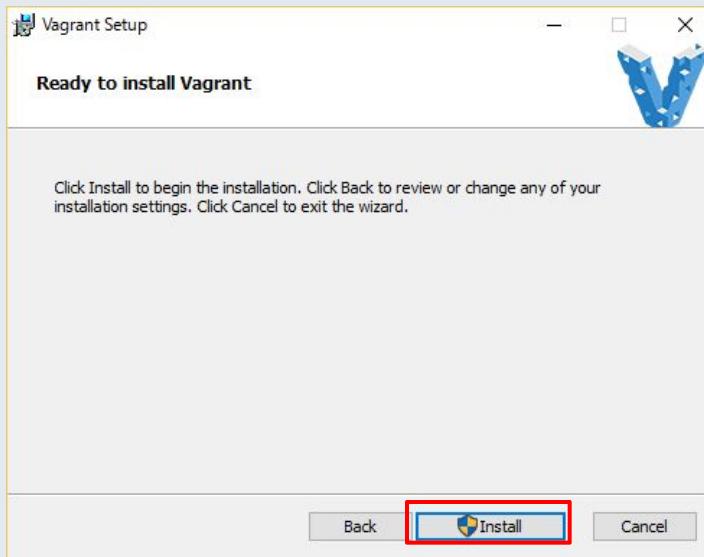


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-3. Install vagrant

- Click “Install” button

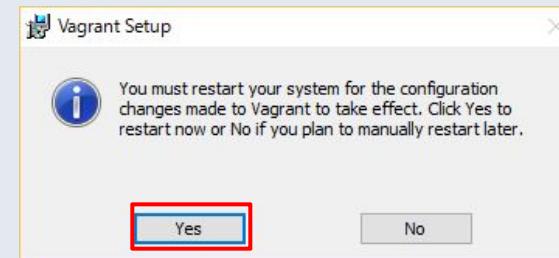
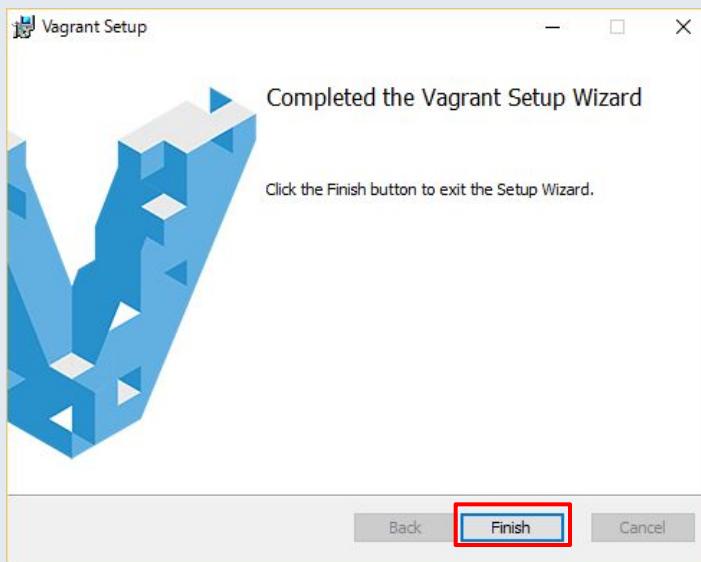


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-3. Install vagrant

- Click “Finish to exit the Setup, but you have to restart your PC, so make to close important documents and click “Yes”

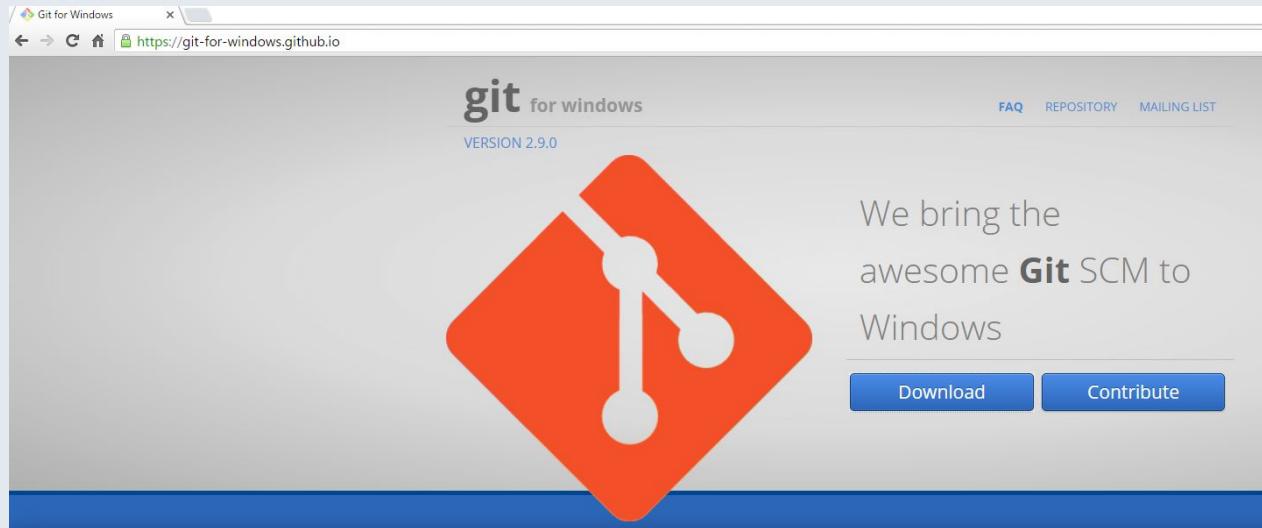


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- Go to <https://git-for-windows.github.io/> and click “Download” button on this page. (If already installed skip 1-4.)



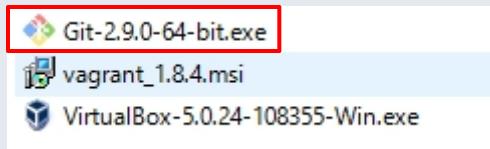
If Git Bash is installed, you can use Linux command from windows.

1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- Double click on Git-2.9.0-64-bit.exe, and click “Next” button

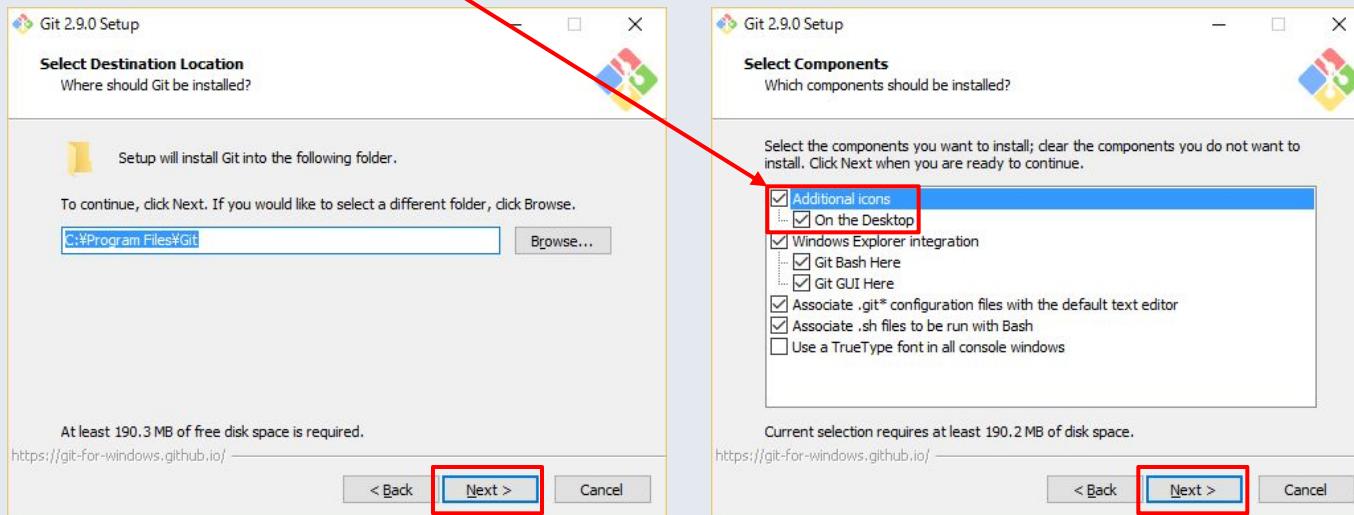


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- Install location as default, click “Next” button. On the next screen, check Additional icons and click “Next” button.

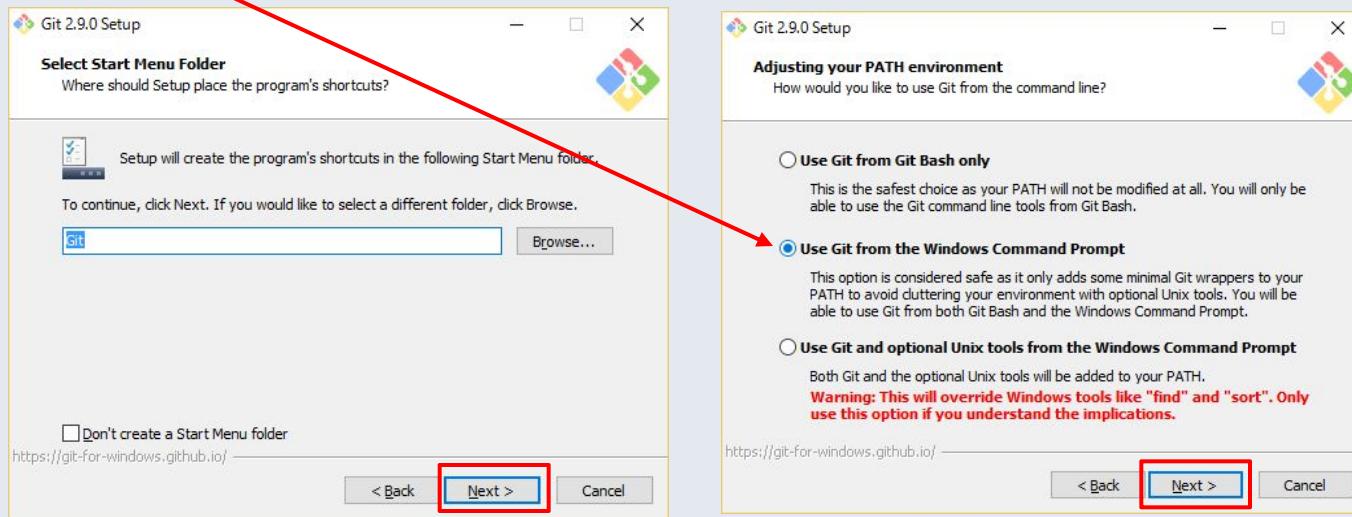


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- Click “Next” button to add Git to your Start Menu. Select the middle option on the next screen and click “Next” button

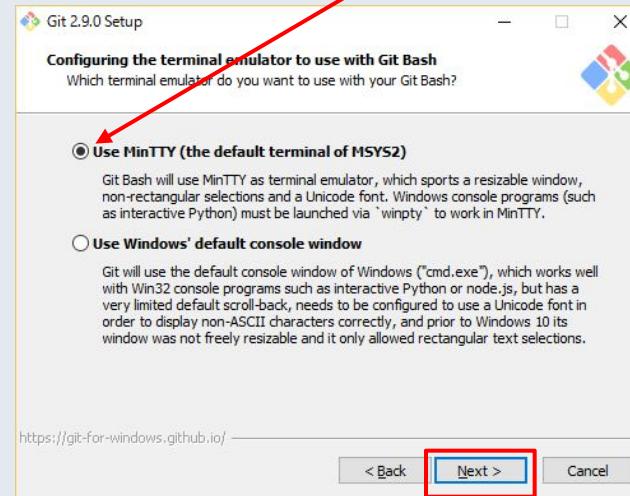
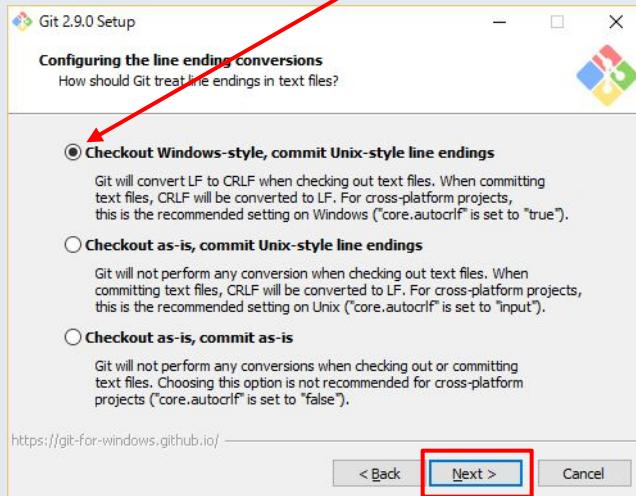


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- On this screen, select the first option (recommended for windows) and click “Next” button. For terminal emulator, select the first option(not window cmd.exe) and click “Next” button

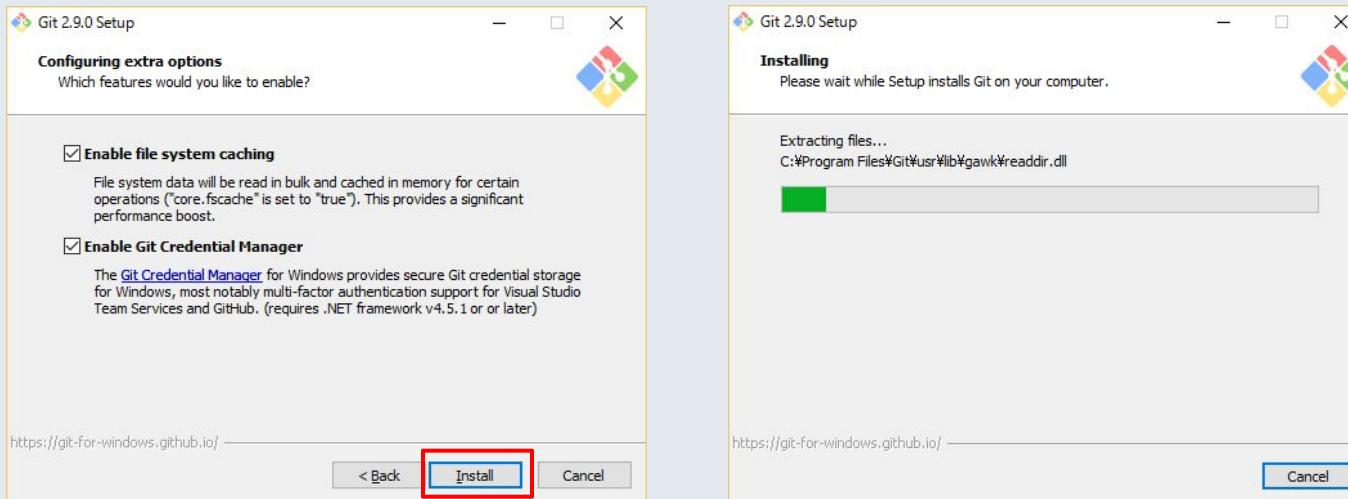


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- For extra options, check both of them. The first check is to increase response speed of git, the second check is for additional credential function when using Github.

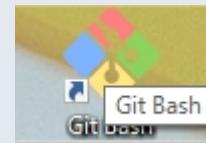
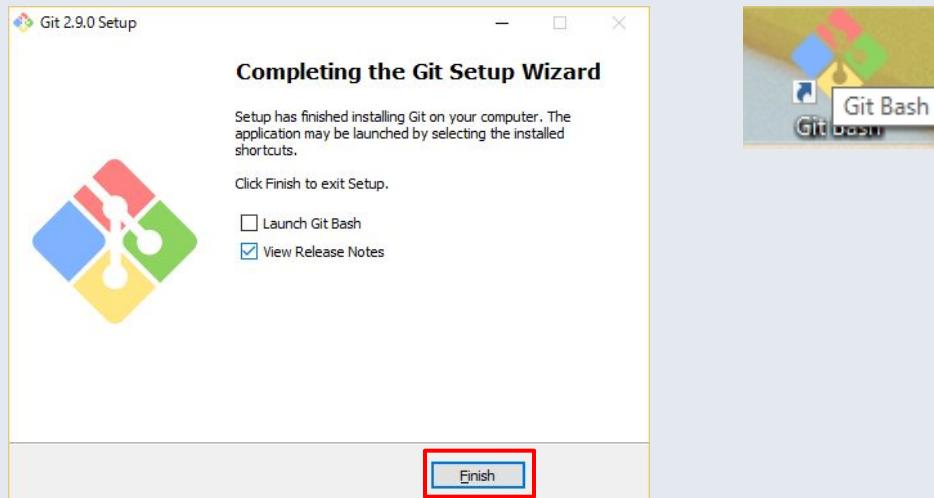


1. Download necessary software for the class

Practical Web Development with Spring Boot Development environment

1-4. Install Git Bash

- Click “Finish” to exit Setup. And you can see Git Bash icon on your Desktop.



2. Install OS to VM

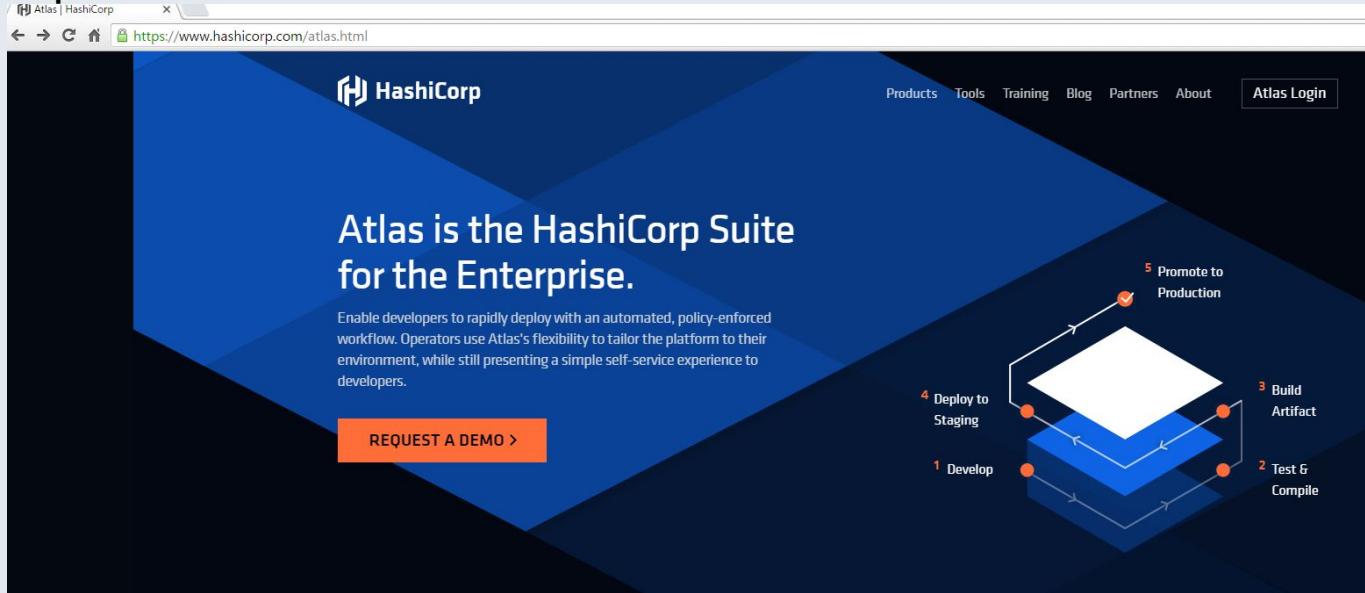
Practical Web Development with Spring Boot
Development environment

2-1. Download OS image

- Various OS images are provided by hashicorp atlas.

<https://www.hashicorp.com/atlas.html>

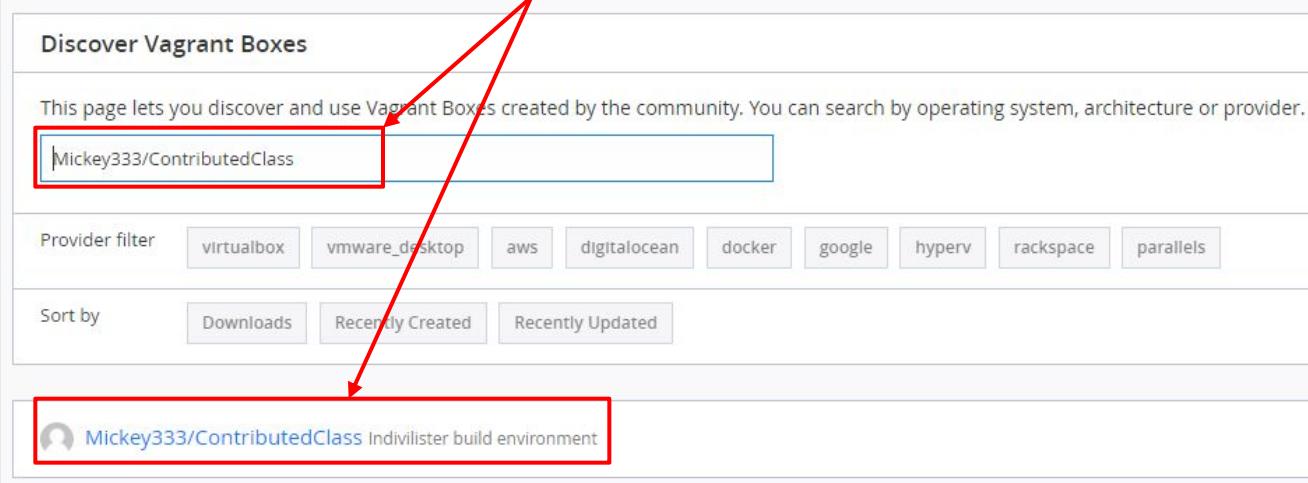
If you want to upload your own created OS image, you should sign up for Atlas account.



2. Install OS to VM

2-1. Download OS image

- In this class, go to <https://atlas.hashicorp.com/boxes/search>, and Discover vagrant boxes. Type Mickey333/ContributedClass and the box will be found. Click the link to the box.

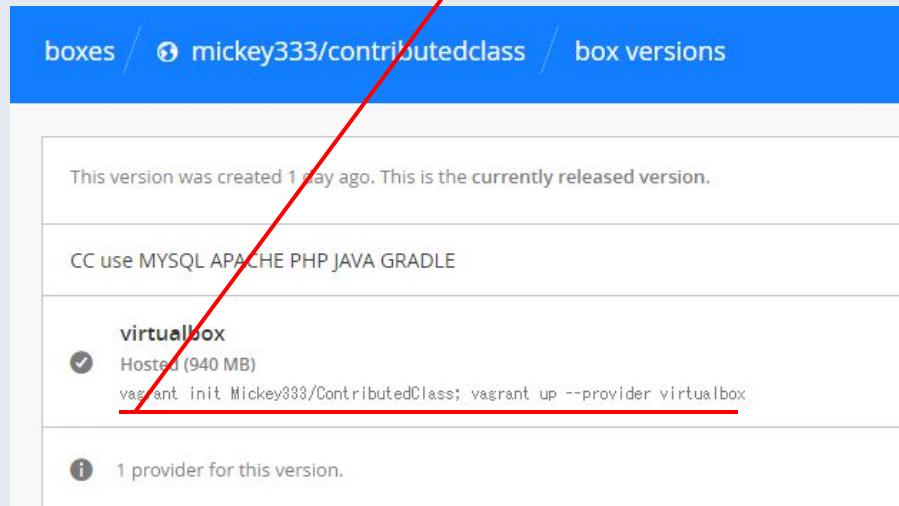


2. Install OS to VM

2-1. Download OS image

- This page provides you with vagrant command to install this OS image into VirtualBox.

```
vagrant init Mickey333/ContributedClass; vagrant up --provider virtualbox
```



This image has Linux
CentOS, Apache, MYSQL,
PHP, JAVA and Gradle
installed.

2. Install OS to VM

2-1. Download OS image

- Double click on Git Bash shortcut icon, and type some command to see if it's working properly.

vagrant –v : this will display vagrant version

vagrant box list : so far you haven't installed boxes yet.



A screenshot of a terminal window titled 'MINGW64:/c/Users/MICK'. The window shows the following text:

```
MICK@PCWIN8-001 MINGW64 ~
$ vagrant -v
Vagrant 1.8.4

MICK@PCWIN8-001 MINGW64 ~
$ vagrant box list
There are no installed boxes! Use `vagrant box add` to add some

MICK@PCWIN8-001 MINGW64 ~
$ |
```

2. Install OS to VM

2-1. Download OS image

Jump to 32 page if you use locally downloaded image

- Let's create a directory to install OS image.

```
cd /drive name
```

```
mkdir springboot
```

```
cd springboot
```

```
MICK@PCWIN8-001 MINGW64 ~
$ cd /h
MICK@PCWIN8-001 MINGW64 /h
$ mkdir springboot
MICK@PCWIN8-001 MINGW64 /h
$ cd springboot
```

Windows
Development tool
(eclipse, STS)

Create source code from windows

H:\springboot
Source files

CentOS
Java and Gradle
to build source files

Build and run it on CentOS in VM

2. Install OS to VM

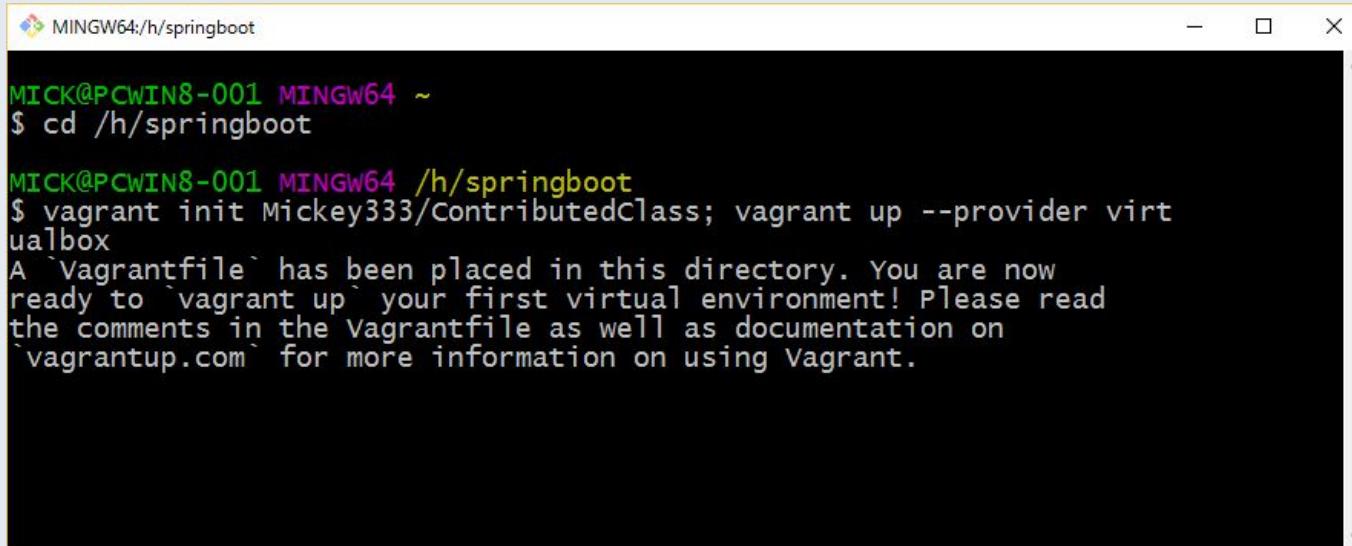
2-1. Download OS image

- Install CentOS by using vagrant command provided on the website

Copy and paste the following command to gitbash.

vagrant init Mickey333/ContributedClass; vagrant up --provider virtualbox

This will install CentOS into VirtualBox.



```
MICK@PCWIN8-001 MINGW64 ~
$ cd /h/springboot

MICK@PCWIN8-001 MINGW64 /h/springboot
$ vagrant init Mickey333/ContributedClass; vagrant up --provider virtualbox
A `Vagrantfile` has been placed in this directory. You are now
ready to `vagrant up` your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
`vagrantup.com` for more information on using Vagrant.
```

2. Install OS to VM

2-1. Download OS image

- After the install finished, type “ls” command to see if “Vagrantfile” exists.

```
MICK@PCWIN8-001 MINGW64 /h/springboot
$ ls
Vagrantfile
```

Whenever you want to run this CentOS, use “vagrant up” command in this directory where Vagrantfile is located.

2. Install OS to VM

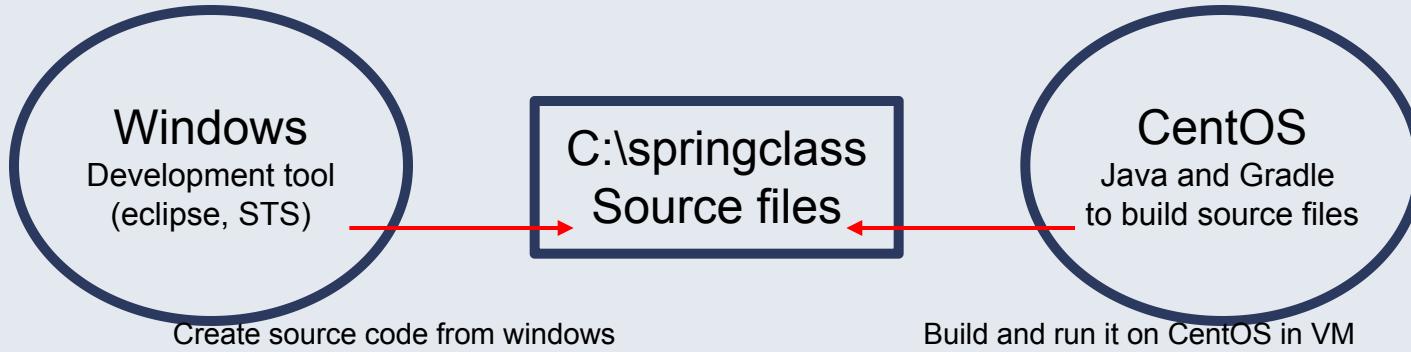
2-1. Download OS image

- Let's create a directory "springclass" under C drive to install OS image.

Open gitbash
and move to the directory

cd /c/springclass

```
MIKIO@win8mk MINGW64 ~
$ cd /c/springclass
MIKIO@win8mk MINGW64 /c/springclass
$ -
```



2. Install OS to VM

Practical Web Development with Spring Boot
Development environment

2-1. Download OS image

- Install OS by using locally downloaded file.

```
MIKIO@win8mk MINGW64 /c/springclass
$ vagrant box add Mickey333/ContributedClass C:/vm/contributedClass.box
```

↓
\$ vagrant box add Mickey333/ContributedClass C:/vm/contributedClass.box

```
MIKIO@win8mk MINGW64 /c/springclass
$ vagrant init Mickey333/ContributedClass
```

↓
\$ vagrant init Mickey333/ContributedClass

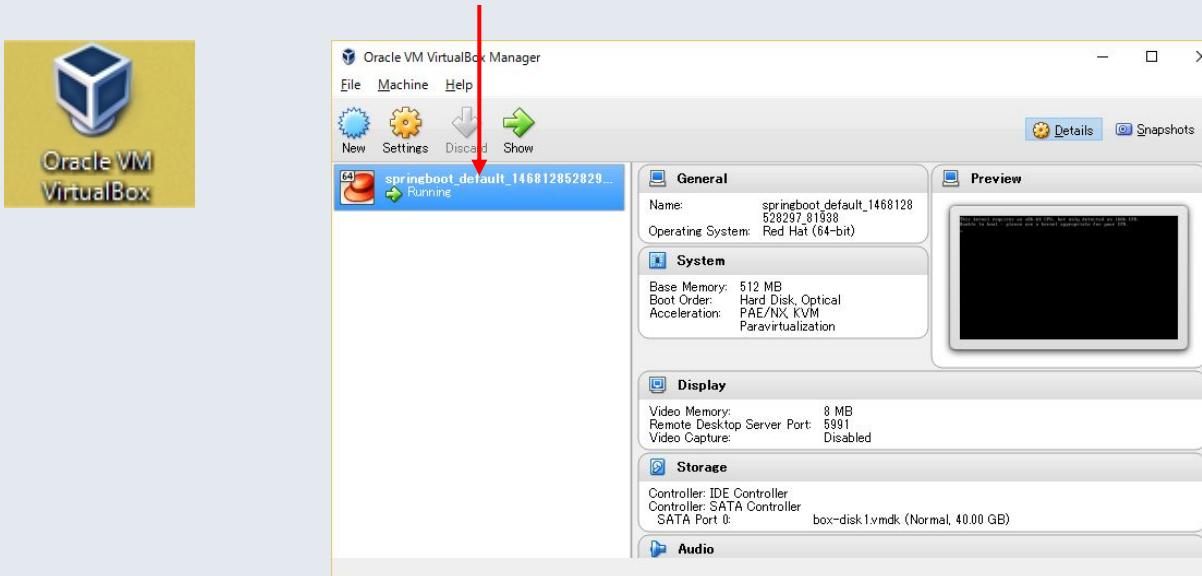
```
MIKIO@win8mk MINGW64 /c/springclass
$ vagrant up
```

↓
\$ vagrant up

2. Install OS to VM

2-1. Download OS image

- Double click on “Oracle VM VirtualBox” icon, and you will see the CentOS server is running on VM.



2. Install OS to VM

2-2. initial setup

- Set IP address for this server. Open Vagrantfile with vi editor, and uncomment the line with ip. Press Esc key and type “:wq” then Enter

```
MICK@PCWIN8-001 MINGW64 /h/springboot
$ vi Vagrantfile
```

```
# Create a private network, which allows host-only access to the machine
# using a specific IP.
# config.vm.network "private_network", ip: "192.168.33.10"
```

```
# Create a private network, which allows host-only access to the machine
# using a specific IP.
config.vm.network "private_network", ip: "192.168.33.10"
```

```
"/h/springboot/Vagrantfile[+][dos] (12:21 10/07/2016)
```

```
:wq
```

2. Install OS to VM

2-2. initial setup

- \$vagrant reload command to apply the change in IP address.

```
MICK@PCWIN8-001 MINGW64 /h/springboot
$ vagrant reload
```

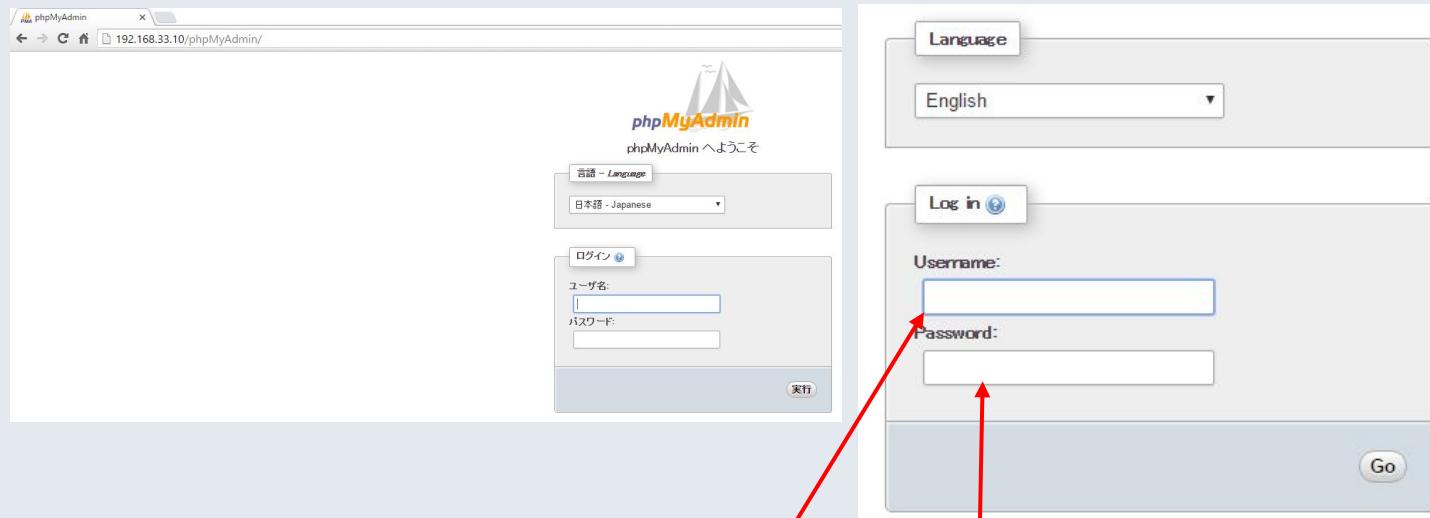
After reloading, open up your browser and access 192.168.33.10, you will see the apache default screen.



2. Install OS to VM

2-2. initial setup

- Access <http://192.168.33.10/phpMyAdmin/> to see if you can access phpMyAdmin.



Check if you can also login with Username: root Password: manager

2. Install OS to VM

2-2. initial setup

- Connect to CentOS with ssh and try some command

```
MICK@PCWIN8-001 MINGW64 /h/springboot
$ vagrant ssh
Last Login: Fri Jul  8 10:10:44 2016 from 10.0.2.2
[vagrant@localhost ~]$ java -version
java version "1.8.0_91"
Java(TM) SE Runtime Environment (build 1.8.0_91-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.91-b14, mixed mode)
[vagrant@localhost ~]$ javac -version
javac 1.8.0_91
[vagrant@localhost ~]$ gradle -v
-----
Gradle 2.13
-----
Build time:  2016-04-25 04:10:10 UTC
Build number: none
Revision:   3b427b1481e46232107303c90be7b05079b05b1c

Groovy:      2.4.4
Ant:         Apache Ant(TM) version 1.9.6 compiled on June 29 2015
JVM:         1.8.0_91 (Oracle Corporation 25.91-b14)
OS:          Linux 2.6.32-573.el6.x86_64 amd64
```

\$vagrant ssh
Connect to CentOS

[---]\$ java -version

[---]\$ javac -version

[---]\$ gradle -v

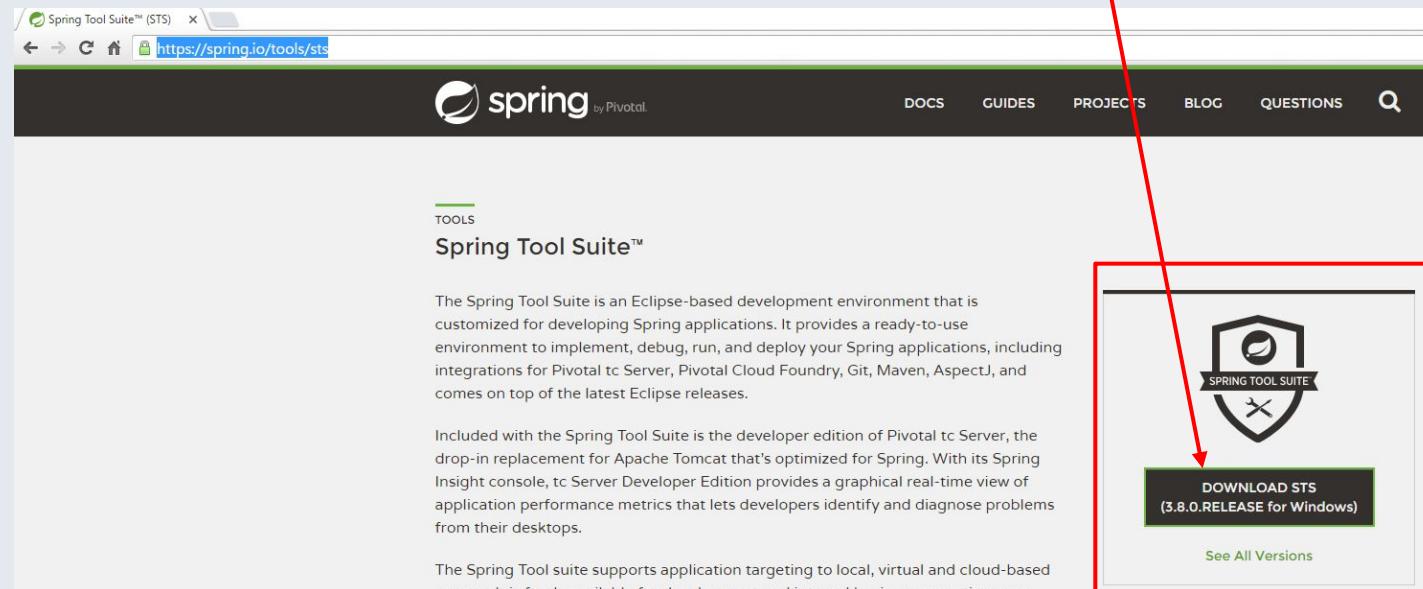
You can build and run Java program on this CentOS server

3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-1. Download STS

- Go to <https://spring.io/tools/sts> and click DOWNLOAD STS.
Download will start. STS = SPRING TOOL SUITE



STS is an eclipse based IDE specialized for Spring boot

3. Install IDE in PC

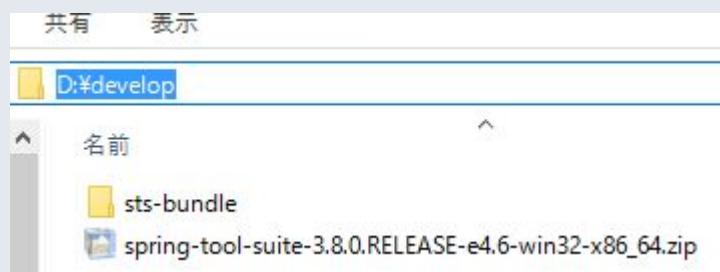
Practical Web Development with Spring Boot
Development environment

3-1. Download STS

- After downloading, copy spring-tool-suite-3.8.0.RELEASE-e4.6-win32-x86_64.zip to D:\develop, and unzip the file to the directory.

Example:

1. Create a folder “develop” under D drive
2. Copy spring-tool-suite-3.8.0.RELEASE-e4.6-win32-x86_64.zip
3. Unzip the file
4. You will see “sts-bundle” folder.



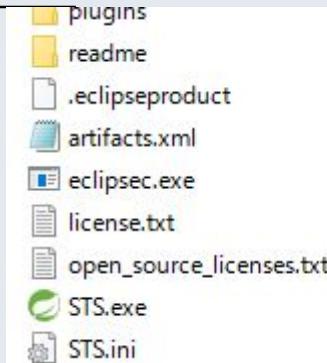
3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

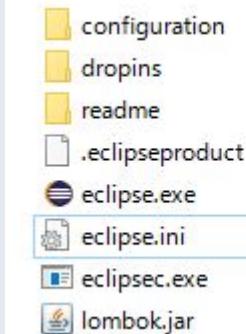
3-1. Download STS

- Move to D:\develop\sts-bundle\sts-3.8.0.RELEASE directory, and compare it with eclipse.

STS



Eclipse



STS.exe = eclipse.exe double click STS.exe to use STS

STS.ini = eclipse.ini configure initial setting with this file.

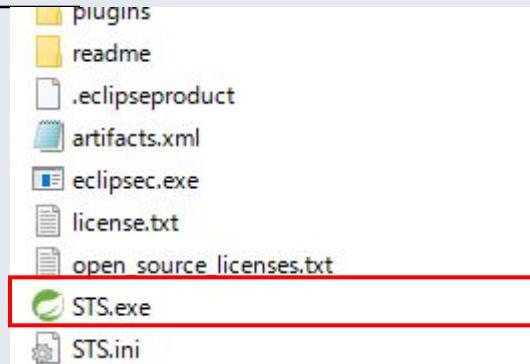
3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-1. Download STS

- Copy STS.exe and create a shortcut on your Desktop

STS

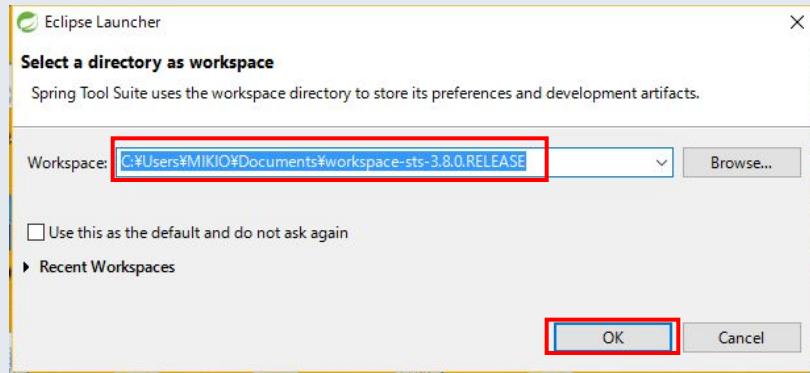


Double-click on the STS short
cut on your Desktop to start
STS.

3. Install IDE in PC

3-1. Download STS

- Select a workspace and press “OK” button



For the time being, select the default workspace until the setup finishes.

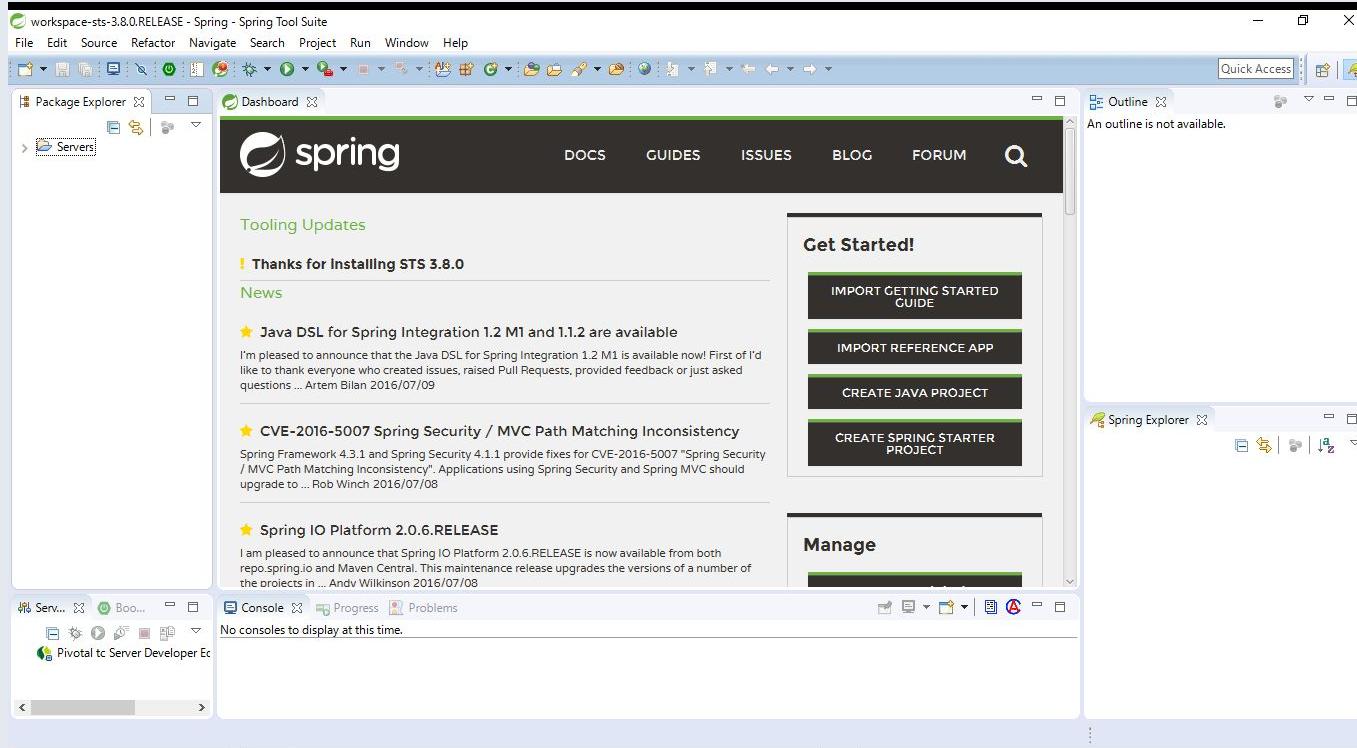
Important: when you actually create an application, you must select the directory where Vagrantfile exists. Otherwise you can not build and run the application on the CentOS server

3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-1. Download STS

- You can now see STS screen, pretty much the same as Eclipse!!

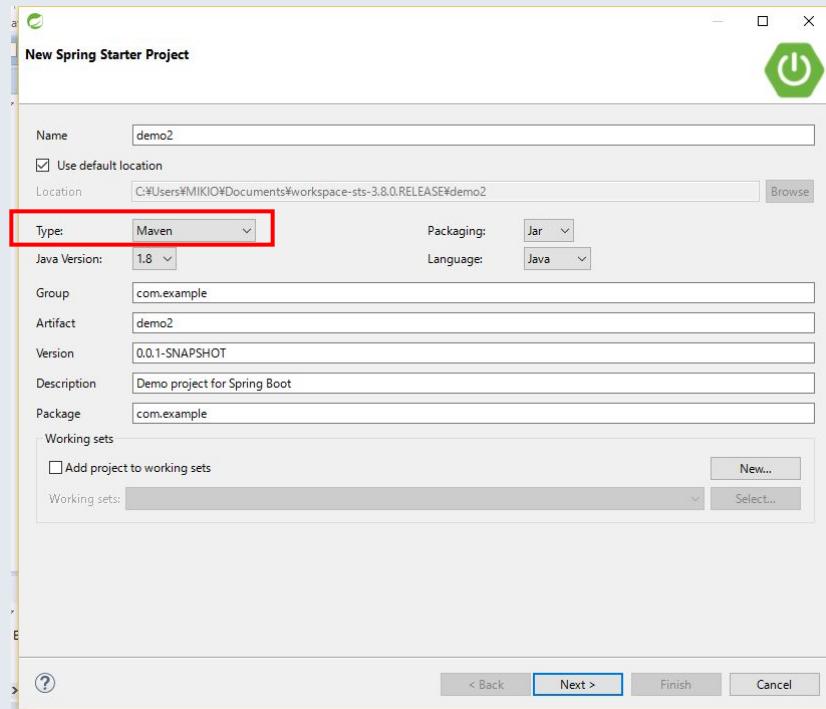


3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-2. local build environment

- Lets create a project. File- New – Spring Starter Project



Now Type: field, you can only choose Maven.
If you select Gradle, you can not press Next> button

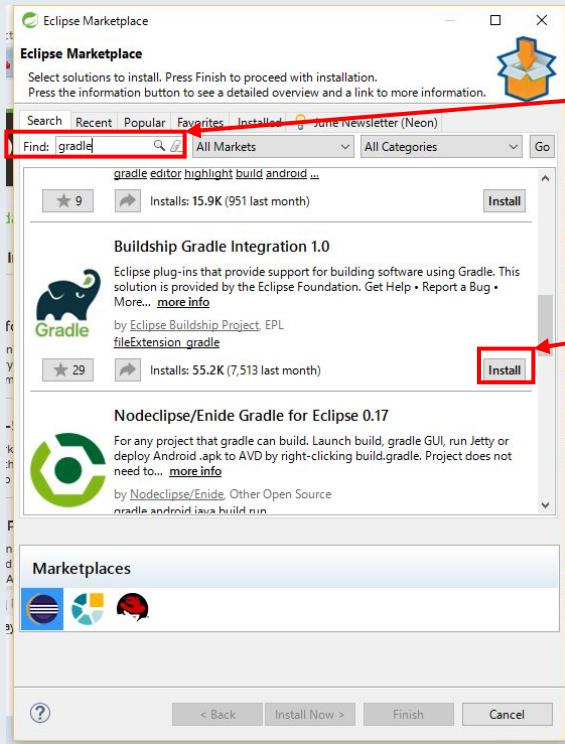
Let's install Gradle build environment.

3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-2. local build environment

- Help – Eclipse Marketplace



Find: gradle
Buildship Gradle Integration 1.0

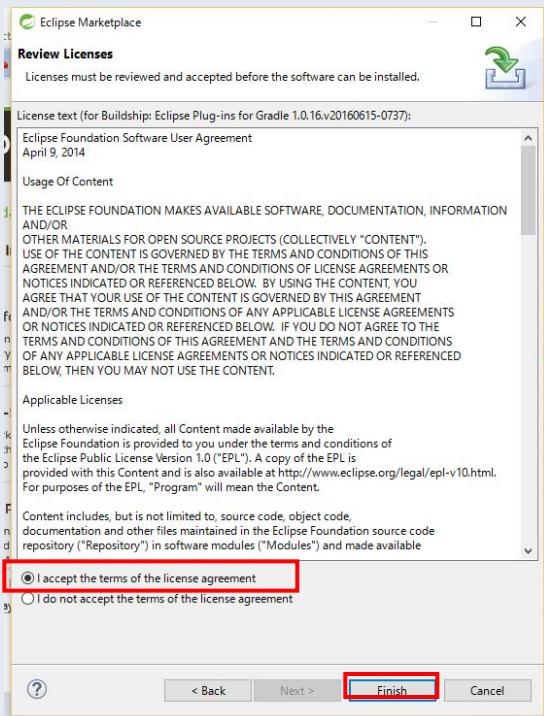
Click “Install” button

3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

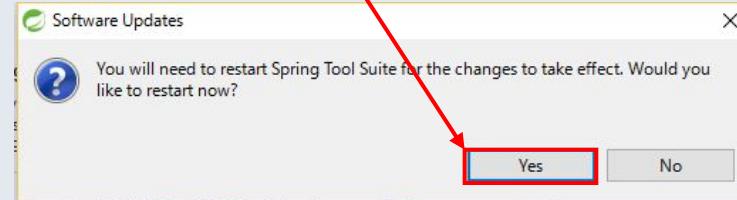
3-2. local build environment

- Select “I accept the terms of the license agreement” and press “Finish” button



Install will start, and it takes a while to finish.

Then press “Yes” to restart STS.

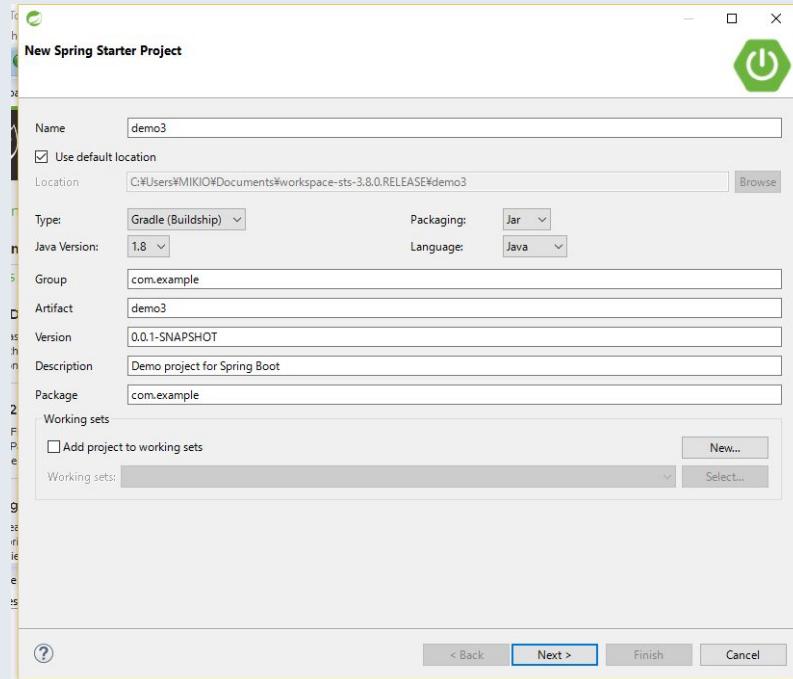


3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-2. local build environment

- After restarting STS, File – New – Spring Start Project.



Now you can choose Type:
Gradle(Buildship) , and Next
>button is enabled.

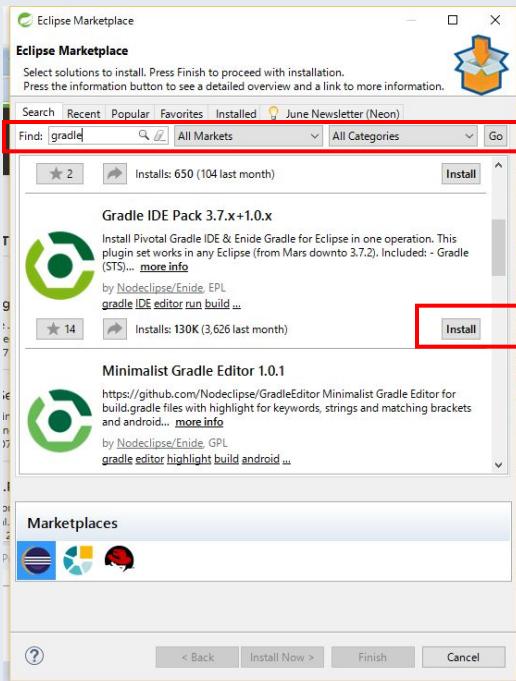
Cancel this project.

3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-2. local build environment

- Go to Eclipse Marketplace again and install Gradle IDE Pack 3.7.x + 1.0.x



Type: gradle

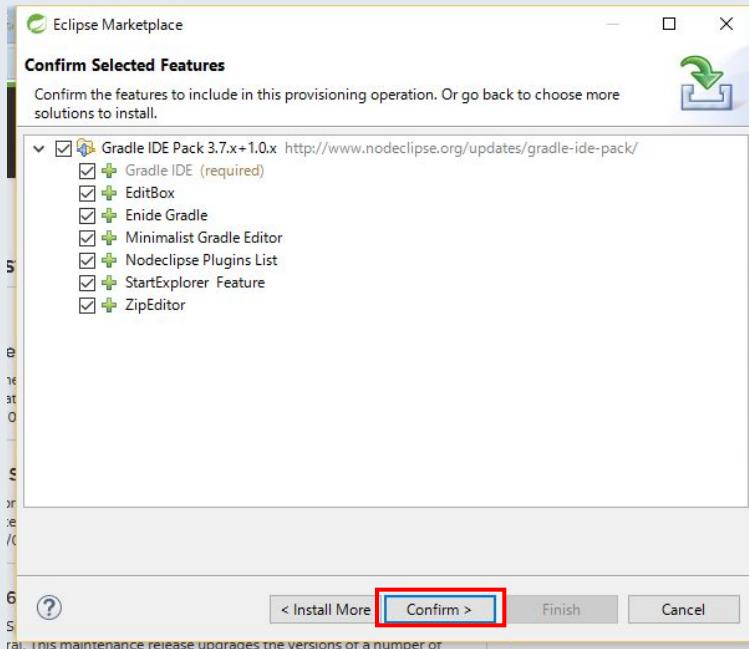
Press “install” button

3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-2. local build environment

- Press “Confirm>” button to what is installed.

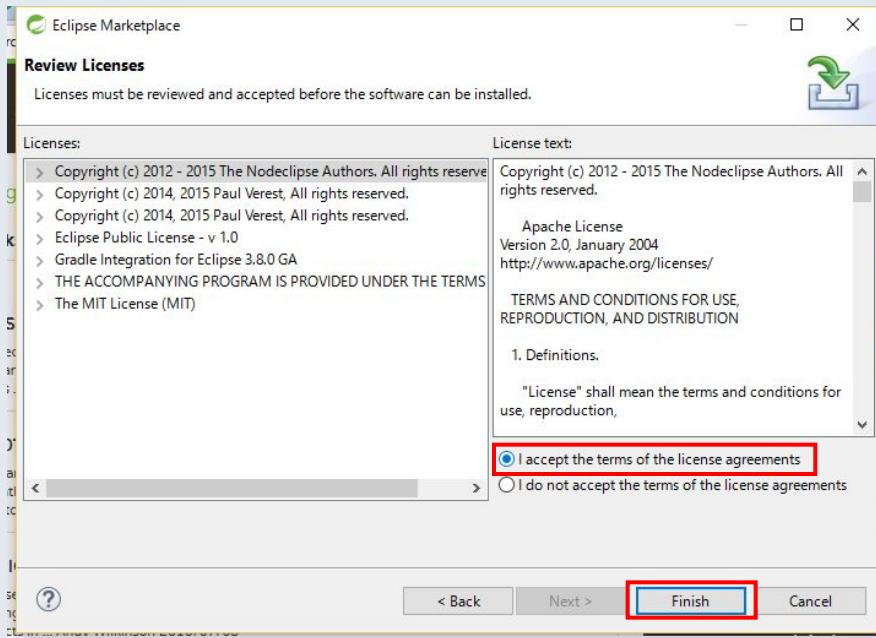


3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

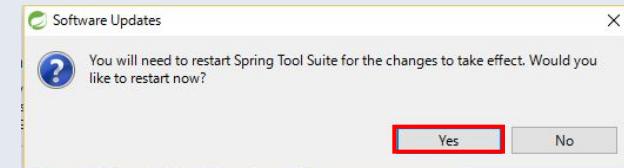
3-2. local build environment

- Select “I accept the terms of the license agreements” and Press “Finish” button



Install will start, it takes a while to finish.

Press “Yes” to restart STS

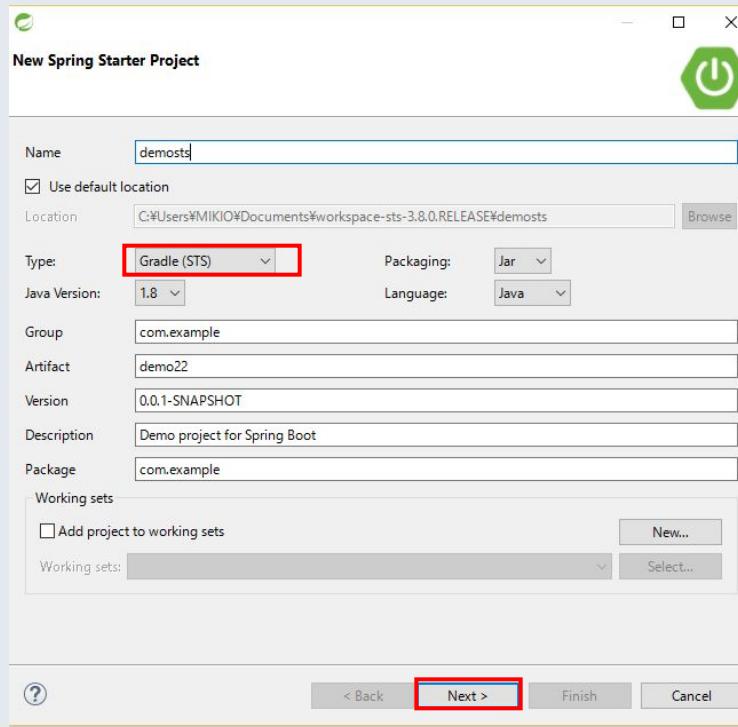


3. Install IDE in PC

Practical Web Development with Spring Boot
Development environment

3-2. local build environment

- File – New – Spring Starter Project



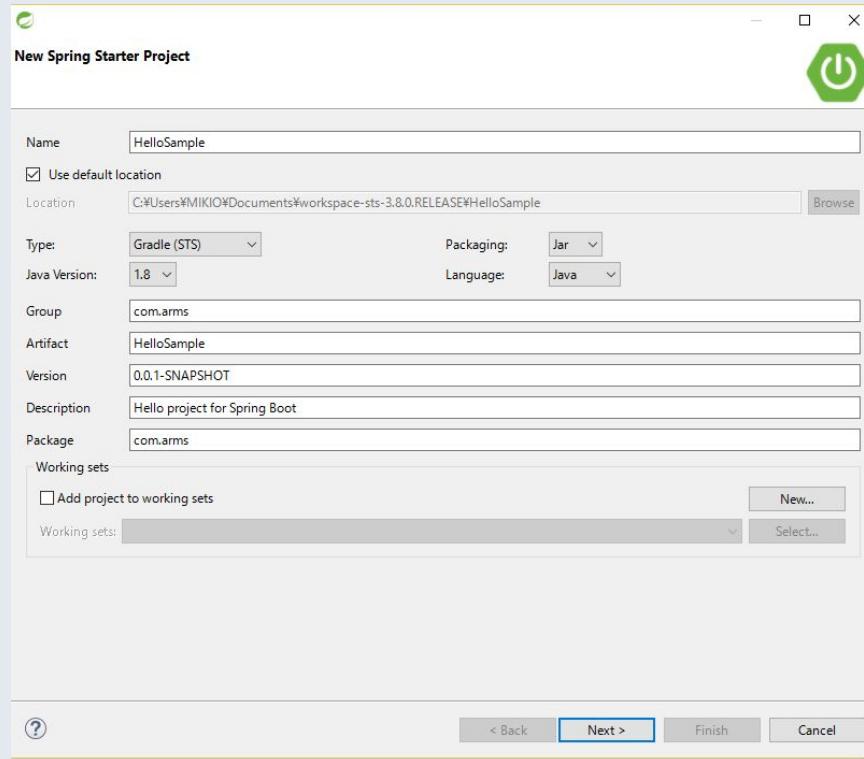
Now you can choose Type:
Gradle(STS), and “Next>”
button is enabled

4. Create a project

Practical Web Development with Spring Boot
Development environment

4-1. Hello in local

- File – New – Spring Starter Project



Create a project with the following parameter.

Name: HelloSample

Type: Gradle(STS)

Group: com.arms

Artifact: HelloSample

Description:

Hello project for Spring boot

Package: com.arms

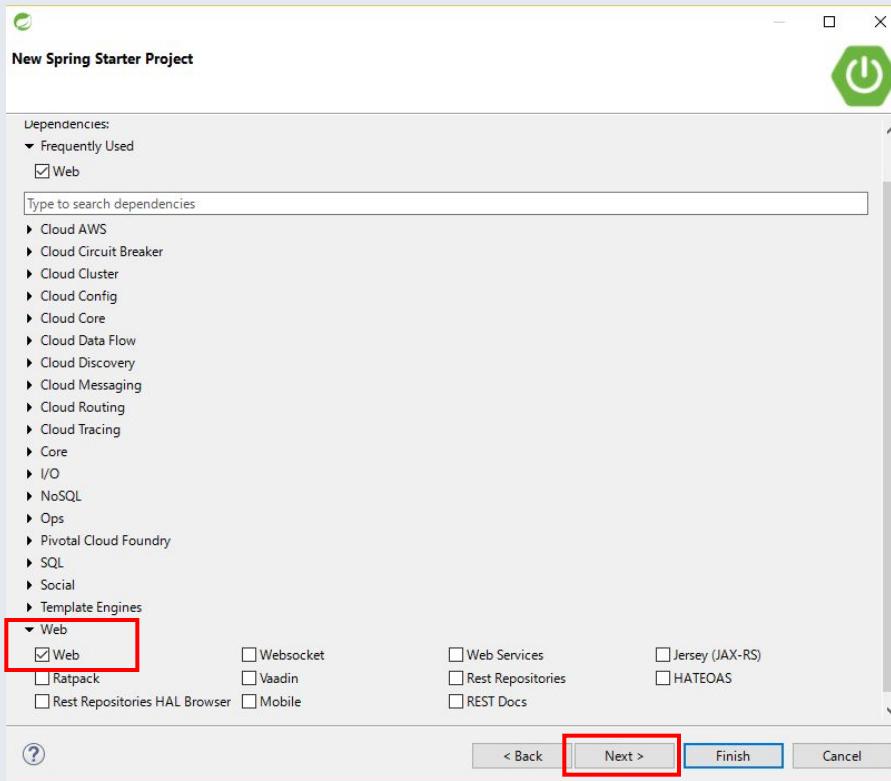
Press “Next >” button

4. Create a project

Practical Web Development with Spring Boot Development environment

4-1. Hello in local

- On the New Spring Starter Project, check “Web” and click “Next>” button

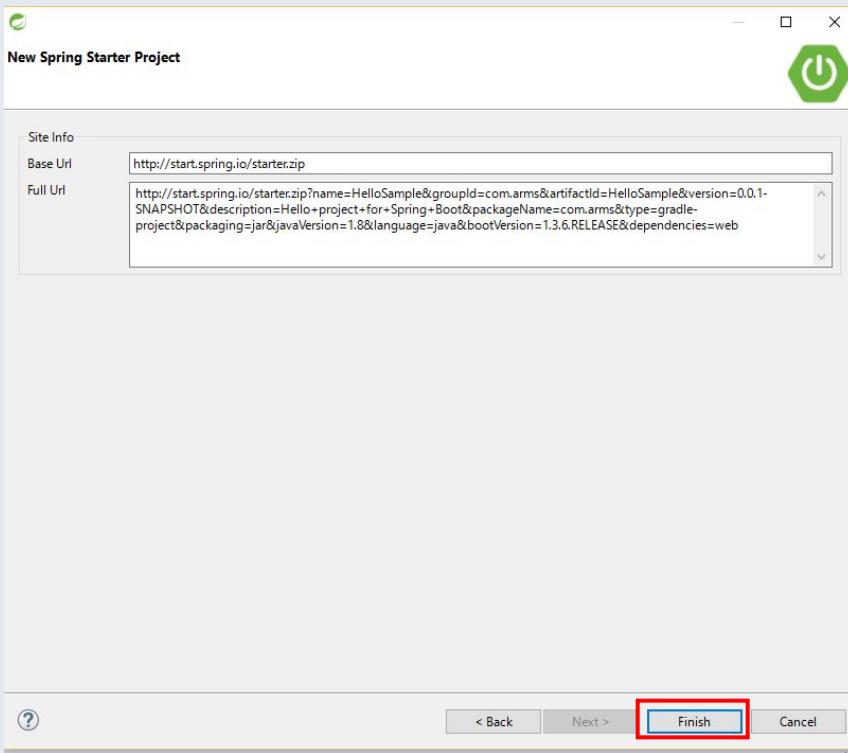


4. Create a project

Practical Web Development with Spring Boot
Development environment

4-1. Hello in local

- Click “Finish”



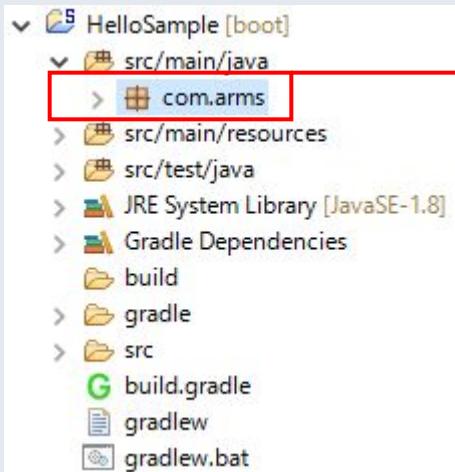
You can visit the following site
<https://start.spring.io/>
and also create the same zip file.

But STS has this feature

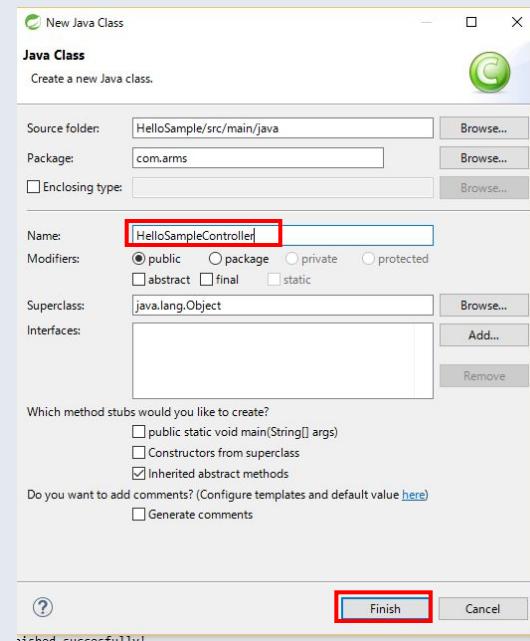
4. Create a project

4-1. Hello in local

- You base project will be created as below. Create a sample class



Let's create a sample class!!!
Right click on "com.arms" New - Class



Type in the name
Name:
HelloSampleController

and Press "Finish"
button

4. Create a project

4-1. Hello in local

- Add the following code to HelloSampleController.java

```
package com.arms;

import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HelloSampleController {

    @RequestMapping("/")
    public String index() {
        return "Hello Sample for Spring boot";
    }
}
```

4. Create a project

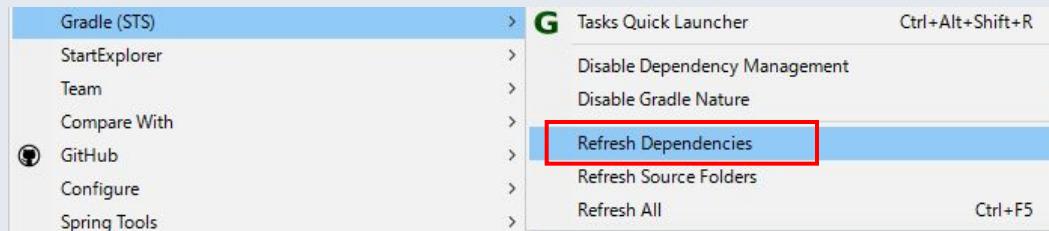
Practical Web Development with Spring Boot
Development environment

4-1. Hello in local

- Right click on your project.

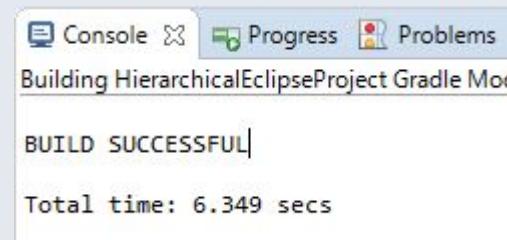
Gradle(STS) – Enable Dependency Management

This will enable some menu



If there are dependencies to be resolved, use this menu.

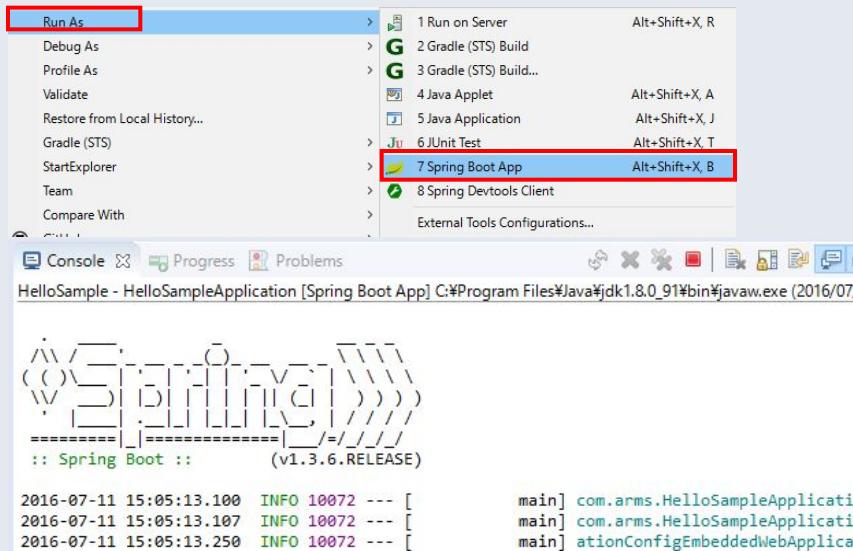
After that, you will see “BUILD SUCCESSFUL” message in the Console.



4. Create a project

4-1. Hello in local

- Let's run your project!! Right click on your project,
Run as – Spring Boot App

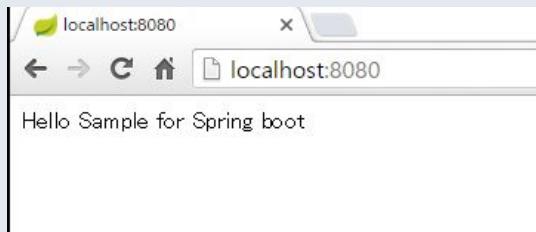


You will see the Console screen like the left figure.

4. Create a project

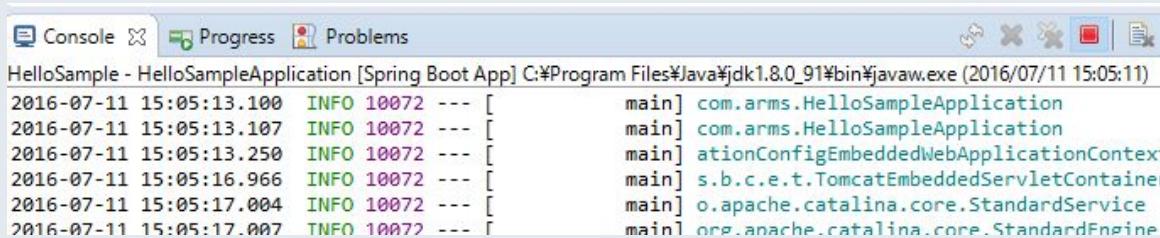
4-1. Hello in local

- Open your browser and access localhost:8080



You will see the screen like the left figure.

To stop your application click the red square on the Console screen.



```
2016-07-11 15:05:13.100  INFO 10072 --- [           main] com.emarsys.sample.HelloSampleApplication
2016-07-11 15:05:13.107  INFO 10072 --- [           main] com.emarsys.sample.HelloSampleApplication
2016-07-11 15:05:13.250  INFO 10072 --- [           main] s.b.c.e.t.TomcatEmbeddedServletContainer
2016-07-11 15:05:16.966  INFO 10072 --- [           main] o.apache.catalina.core.StandardService
2016-07-11 15:05:17.004  INFO 10072 --- [           main] o.apache.catalina.core.StandardEngine
```

If you access localhost:8080 again, you will see an error

4. Create a project

4-1. Hello in local

- About the source code

@RestController

This signifies that this class is a controller for RESTful application.

Receive http request and return response body

@RequestMapping("/")

This maps the method to a specific URL. In this case, localhost:8080/

@RequestMapping("/project")

localhost:8080/project

We are going to learn more about it in the future class.

4. Create a project

4-2. Hello in server

- In 4-1. section, everything is done in your local. In this section, create a source code in local, but build and run it on virtual server.

1. Open gitbash, change the directory to where Vagrantfile exists.

```
$ cd /drive/folder ex.($cd /h/springboot)
```

2. Use ls command to see Vagrantfile exists.

```
$ ls  
Vagrantfile
```

3. vagrant up command to start the server. It takes a while....
(you can also check this on VirtualBox)

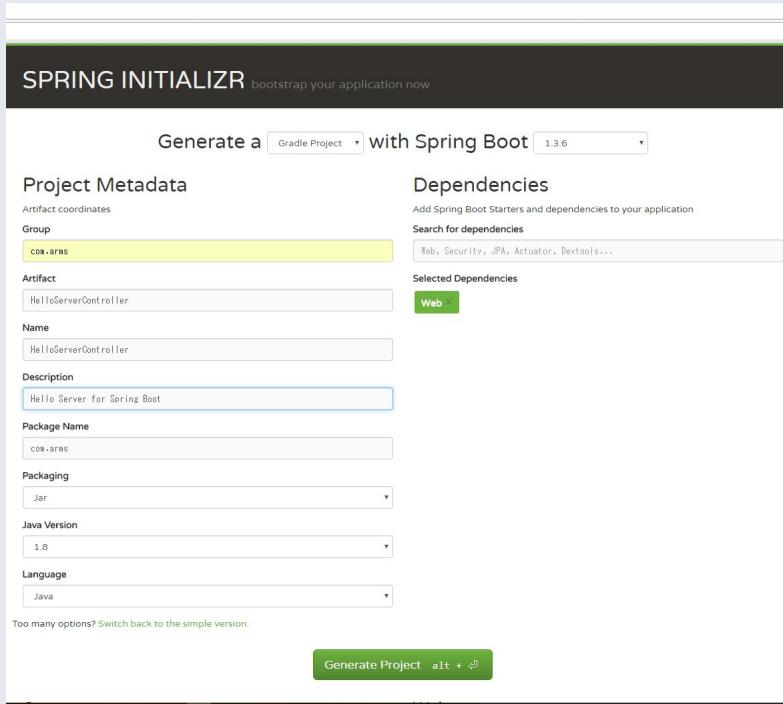
```
$ vagrant up
```

4. Create a project

Practical Web Development with Spring Boot
Development environment

4-2. Hello in server

- Go to <https://start.spring.io/> SPRING INITIALZER, and create a spring project with the following parameters.



Generate a
Gradle Project
With Spring boot
1.3.6

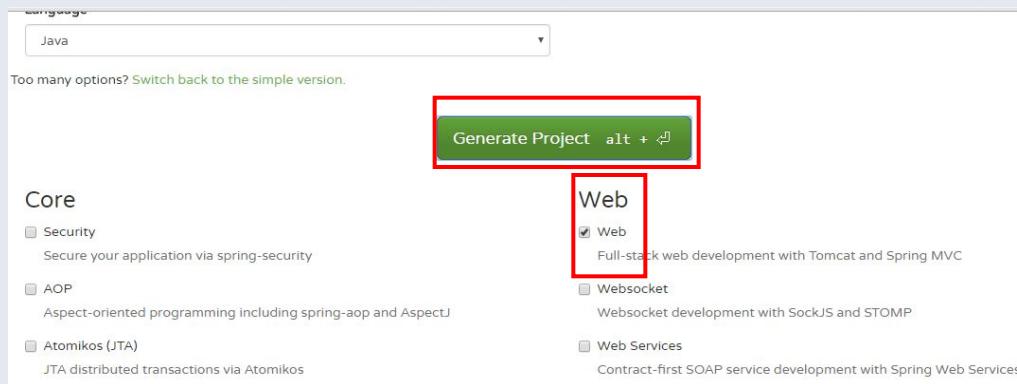
Group: com.arms
Artifact: HelloServerController
Name: HelloServer
Description: Hello Server for Spring Boot
Package Name: com.arms
Packaging: Jar
Language: Java

4. Create a project

4-2. Hello in server

- Go to <https://start.spring.io/> SPRING INITIALZER, and create a spring project with the following parameters.

Check “web” and click “Generate Project” button

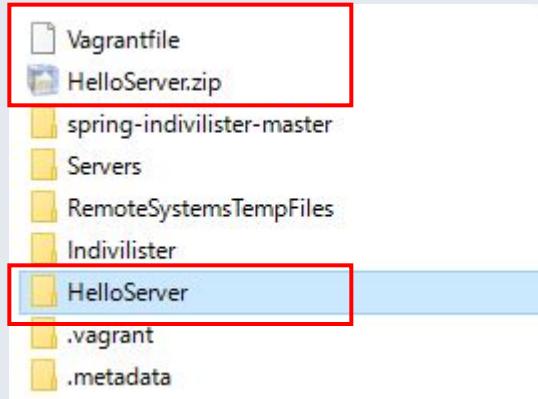


HelloServer.zip will be downloaded to your PC

4. Create a project

4-2. Hello in server

- Copy HelloServer.zip to the directory where Vagrantfile exists, and unzip it in the directory.



[Back to gitbash](#)

Use vagrant ssh command to connect to CentOS server.

```
$ vagrant ssh
```

SSH = Secure SHell

4. Create a project

4-2. Hello in server

- After vagrant ssh, command line will change to linux style as below.

```
[vagrant@-----]$
```

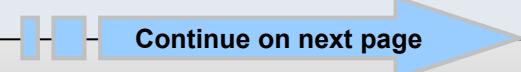
Use pwd command to know the current directory.

```
[vagrant@-----~]$ pwd  
/home/vagrant
```

Move to the directory where HelloServer project exists

```
[vagrant@-----~]$ cd ../../vagrant  
[vagrant@-----vagrant]$ pwd  
/vagrant
```

Please remember how to move to this directory, we are going to use it a lot.



4. Create a project

4-2. Hello in server

Use ls command to list files and directories under the current directory

```
[vagrant@-----vagrant]$ ls  
HelloServer  HelloServer.zip  Vagrantfile
```

Move to HelloServer

```
[vagrant@-----vagrant]$ cd HelloServer
```

Use gradle eclipse command to get ready to import this project into STS(eclipse)

```
[vagrant@-----HelloServer]$ gradle eclipse
```

If you see “BUILD SUCCESSFUL” message, it’s ready to import into STS(eclipse)

You can also see “.gradle” “.settings” “.classpath” “.project” under HelloServer for eclipse import.

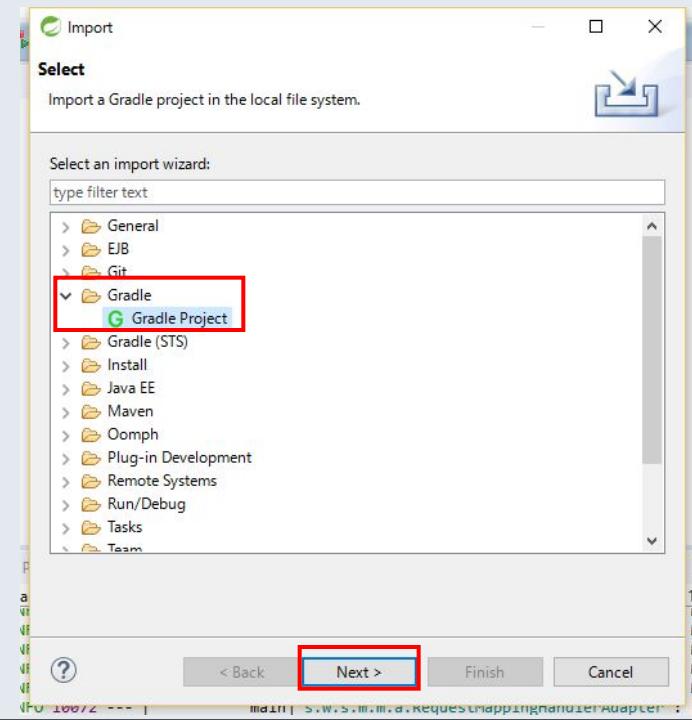
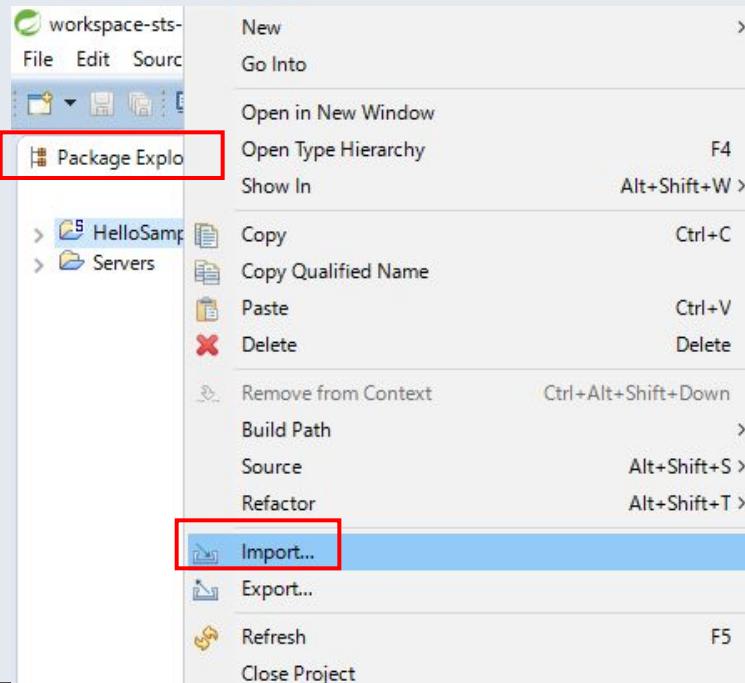
4. Create a project

Practical Web Development with Spring Boot Development environment

4-2. Hello in server

- Open STS and right-click on “Package Explorer” and select “import”. Select “Gradle” – “G Gradle Project”

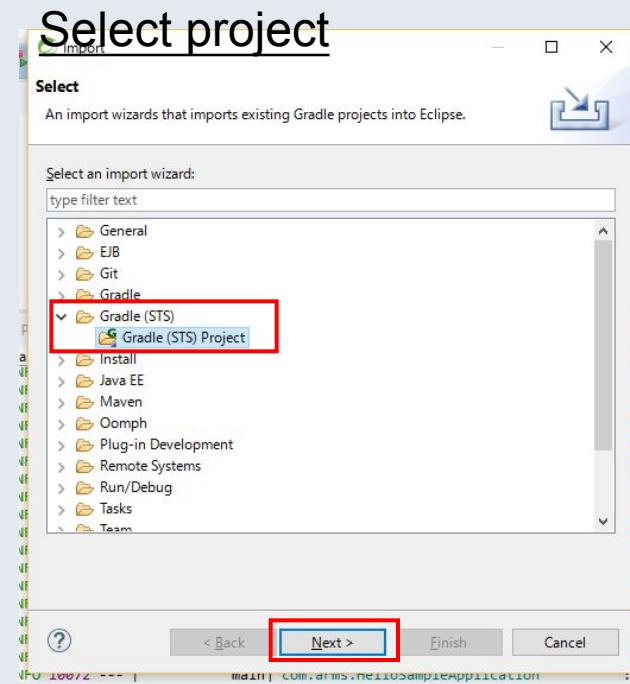
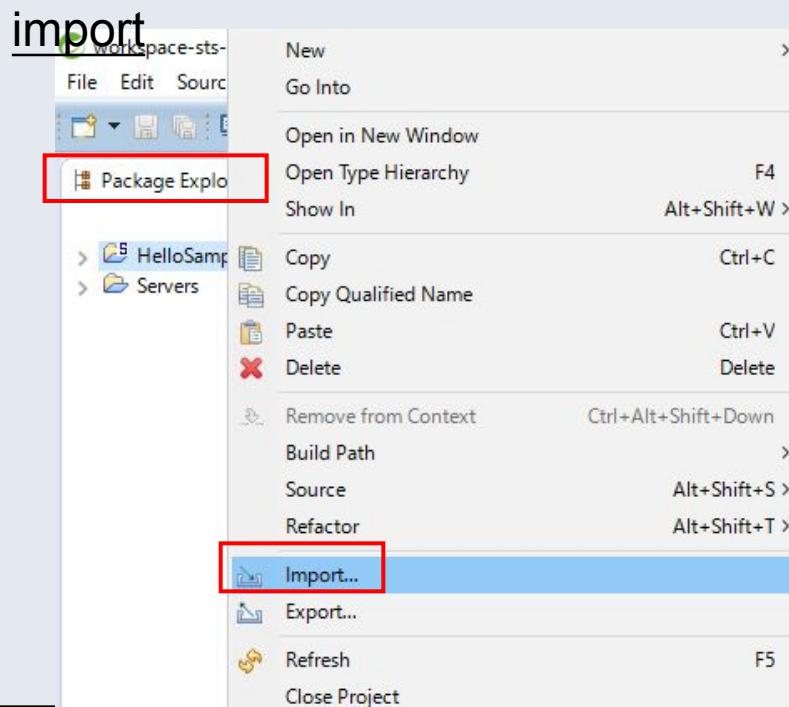
import



4. Create a project

4-2. Hello in server

- Open STS and right-click on “Package Explorer” and select “import”. Select “Gradle(STS)” – “Gradle(STS) Project” and press Next> button

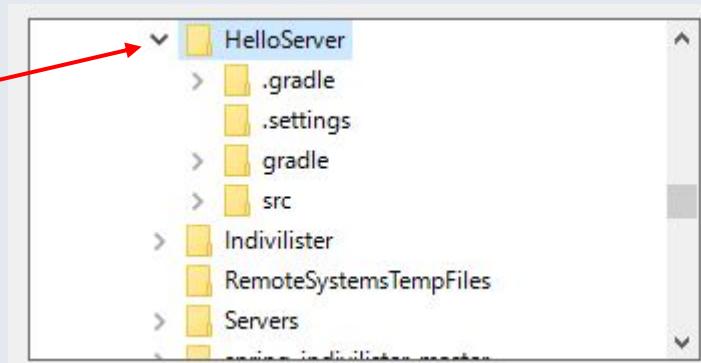
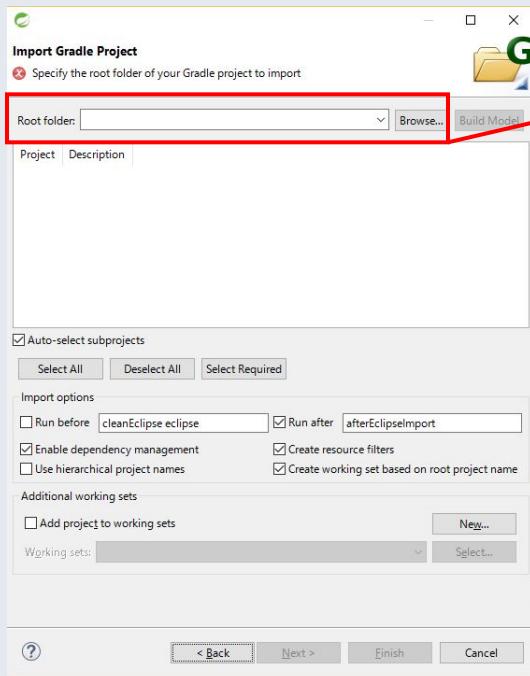


4. Create a project

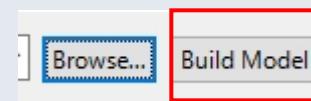
Practical Web Development with Spring Boot Development environment

4-2. Hello in server

- Click “Browse...” button to select the HelloServer project folder.



Then press “Build Model” button

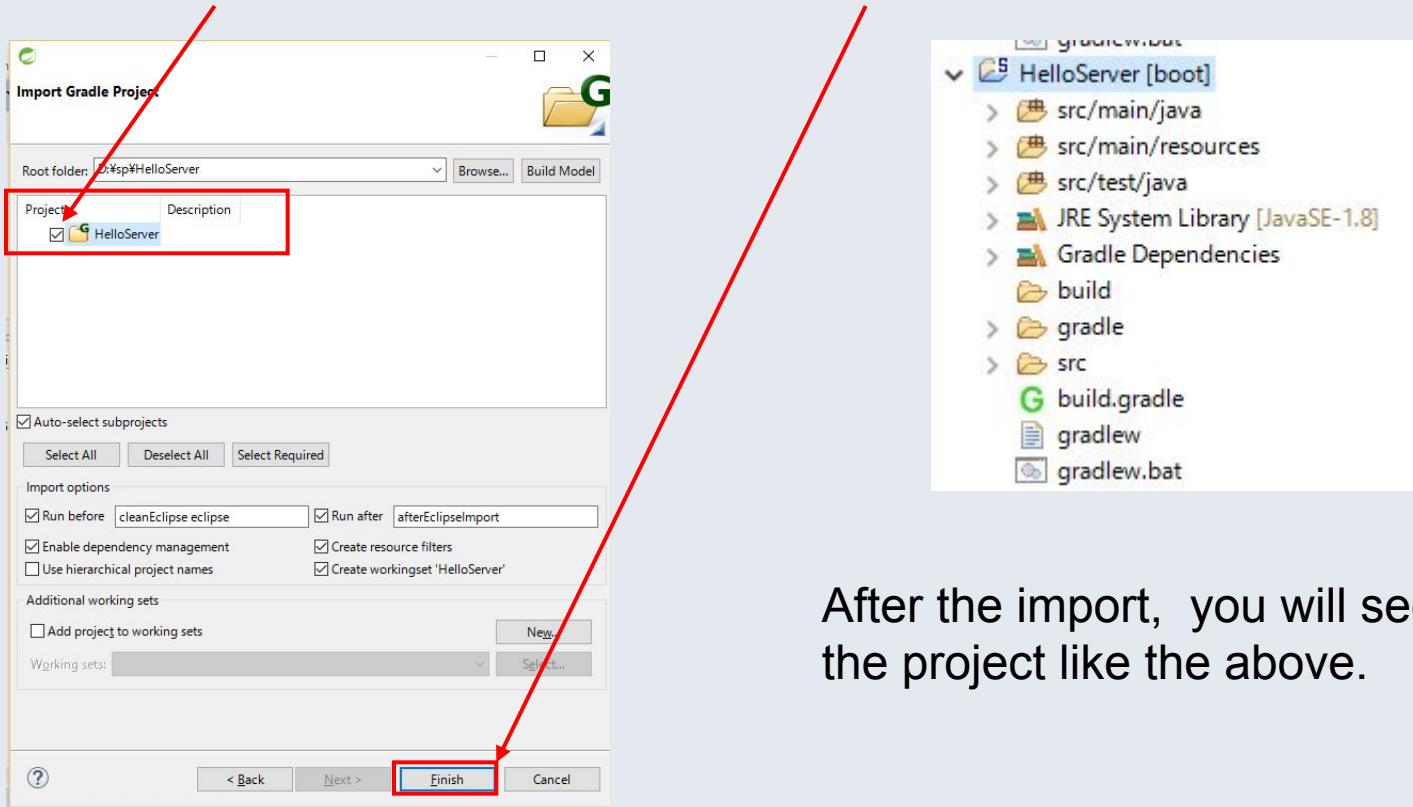


4. Create a project

Practical Web Development with Spring Boot
Development environment

4-2. Hello in server

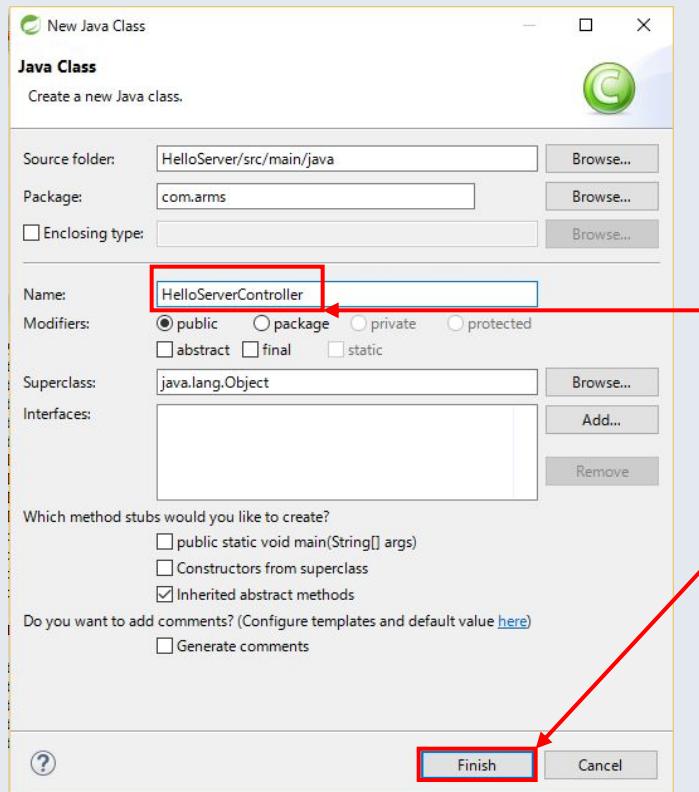
- Check “HelloServer” folder and press “Finish” button



4. Create a project

4-2. Hello in server

- Right-click on “com.arms” New - Class



Type HelloServerController in the Name field and press “Finish”.

4. Create a project

4-2. Hello in server

- Add the following code to HelloServerController.java, and save.

```
package com.arms;

import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HelloServerController {

    @RequestMapping("/hello")
    public String index() {
        return "Hello Server Sample for Spring boot";
    }

}
```

4. Create a project

Practical Web Development with Spring Boot
Development environment

4-2. Hello in server

- This time you will build your project with gradle on the CentOS server.

Use gradle build command to your project.

```
[vagrant@----HelloServer]$ gradle build
```

If you see “BUILD SUCCESSFUL”, ready to run your project

Then use java – jar command to run your project.

```
[vagrant@----HelloServer]$ java -jar build/libs>HelloServer-0.0.1-SNAPSHOT.jar
```



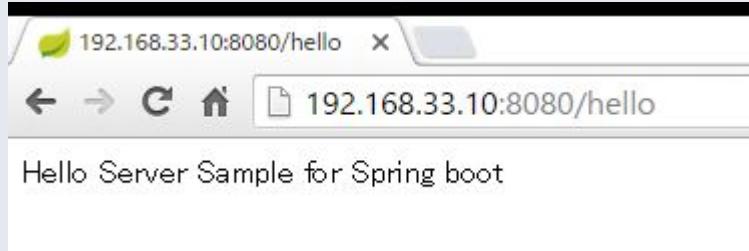
Spring Boot :: (v1.3.6.RELEASE)

```
2016-07-11 11:38:57.657 INFO 2911 --- [           main]
  ing HelloServerApplication on localhost.localdomain with
HelloServer-0.0.1-SNAPSHOT.jar started by vagrant in /vagrant
2016-07-11 11:38:57.692 INFO 2911 --- [           main]
active profile set, falling back to default profiles: defa
2016-07-11 11:38:58.272 INFO 2911 --- [           main]
eshing org.springframework.boot.context.embedded.Annotat
startup date [Mon Jul 11 11:38:58 UTC 2016]; root of cor
```

4. Create a project

4-2. Hello in server

- Access 192.168.33.10:8080/hello and if you can see the following screen, so far so good. Congratulations!!!!!!!



You can now build and run your project both on local and server.

192.168.33.10 is the CentOS server address on VirtualBox.

/hello is @RequestMapping("/hello")



ARMS

Your Idea Leads Your
Ideals

homepage: <http://arms-asia.com/>

facebook: <https://www.facebook.com/arms.asia?ref=ts>