



SPRING BOOT INTRODUCTION

GIẢNG VIÊN:

- ❖ GIỚI THIỆU SPRING, SPRING MVC VÀ SPRING BOOT
- ❖ TÍCH HỢP STS
- ❖ TẠO DỰ ÁN SPRING BOOT
 - ❖ TÌM HIỂU FILE CẤU HÌNH ỨNG DỤNG
 - ❖ KHAI BÁO THƯ VIỆN CẦN THIẾT POM.XML
 - ❖ KHAI BÁO VIEW APPLICATION.PROPERTIES
 - ❖ TẠO CONTROLLER
 - ❖ TẠO VIEW
 - ❖ RUN PROJECT
- ❖ TÌM HIỂU QUY TRÌNH XỬ LÝ REQUEST/RESPONSE
- ❖ SỬ DỤNG CÁC THÀNH PHẦN WEB
- ❖ ĐÓNG GÓI VÀ TRIỂN KHAI ỨNG DỤNG WEB





SPRING BOOT INTRODUCTION

- ❑ Spring là framework được giới thiệu 2003 bởi Rod Johnson.
- ❑ Spring là mã nguồn mở sử dụng để phát triển ứng dụng java phổ biến nhất hiện nay.
- ❑ Phiên bản hiện tại là 5.x được giới thiệu cuối 2018
- ❑ Spring MVC được tổ chức một cách rõ ràng theo MVC pattern.
- ❑ Spring hỗ trợ IoC, DI giúp dễ dàng mở rộng ứng dụng
- ❑ Spring rất nhẹ trong việc phát triển và triển khai.
- ❑ Spring cung cấp cơ chế điều khiển transaction

SPRING FRAMEWORK ARCHITECTURE

DAO

*JDBC Transaction
Management*

ORM

Spring Data JPA

JEE

*JMS, EJB,
Remoting...*

WEB

*Spring Web MVC
Integration*

AOP

AspectJ Integration

Core

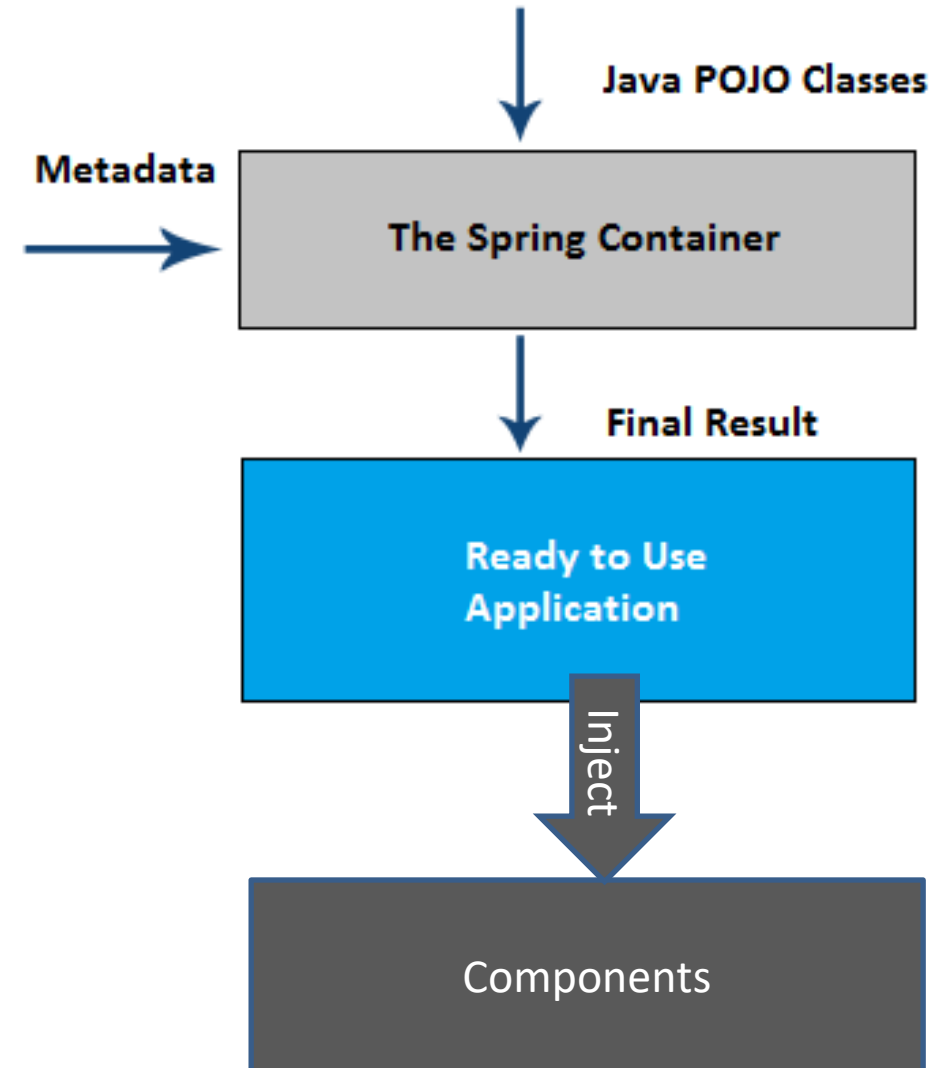
(IoC Container)

❑ Core (IoC Container)

- ❖ Lõi của Spring hỗ trợ IoC (Inversion of Control)

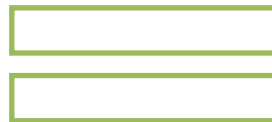
❑ DAO, ORM, AOP, WEB

- ❖ Cho phép tích hợp các công nghệ khác một cách dễ dàng.



- ❑ Spring Boot là một dự án bổ sung của Spring, nhằm đơn giản hóa việc phát triển ứng dụng Spring.
- ❑ Dễ dàng hơn trong việc thiết lập và phát triển ứng dụng.
- ❑ Nguyên tắc “Cấu hình mặc định” giảm thiểu viết mã cấu hình
- ❑ Cho phép nhúng Web Server vào ứng dụng.
- ❑ Có thể chạy ứng dụng web chạy bằng cơ chế dòng lệnh hoặc xuất ra file war để triển khai lên Web Server.
- ❑ Dễ dàng tương tác với hệ sinh thái công nghệ của Spring (JDBC, ORM, Security...)

Spring Boot



Spring Framework



Web
Server



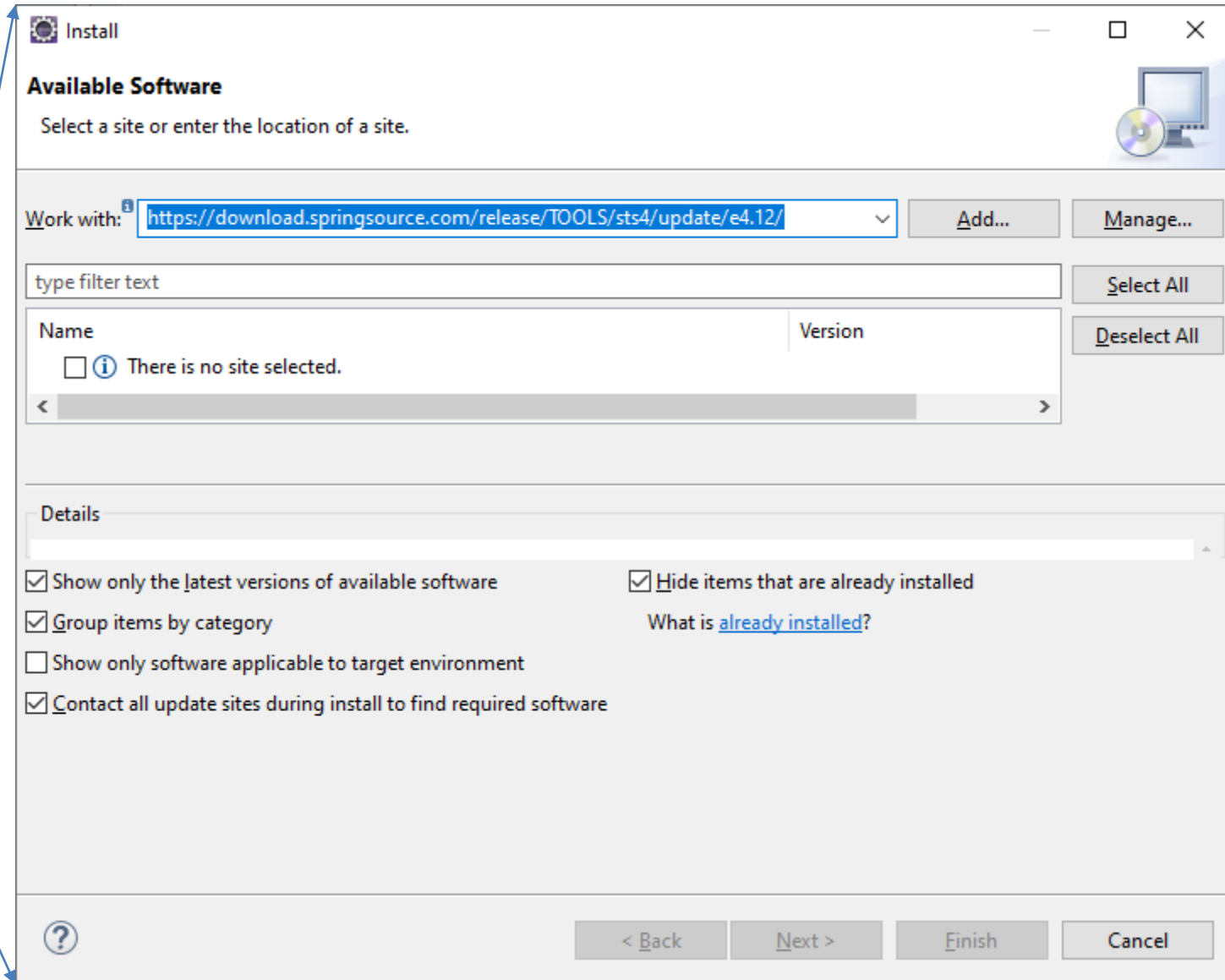
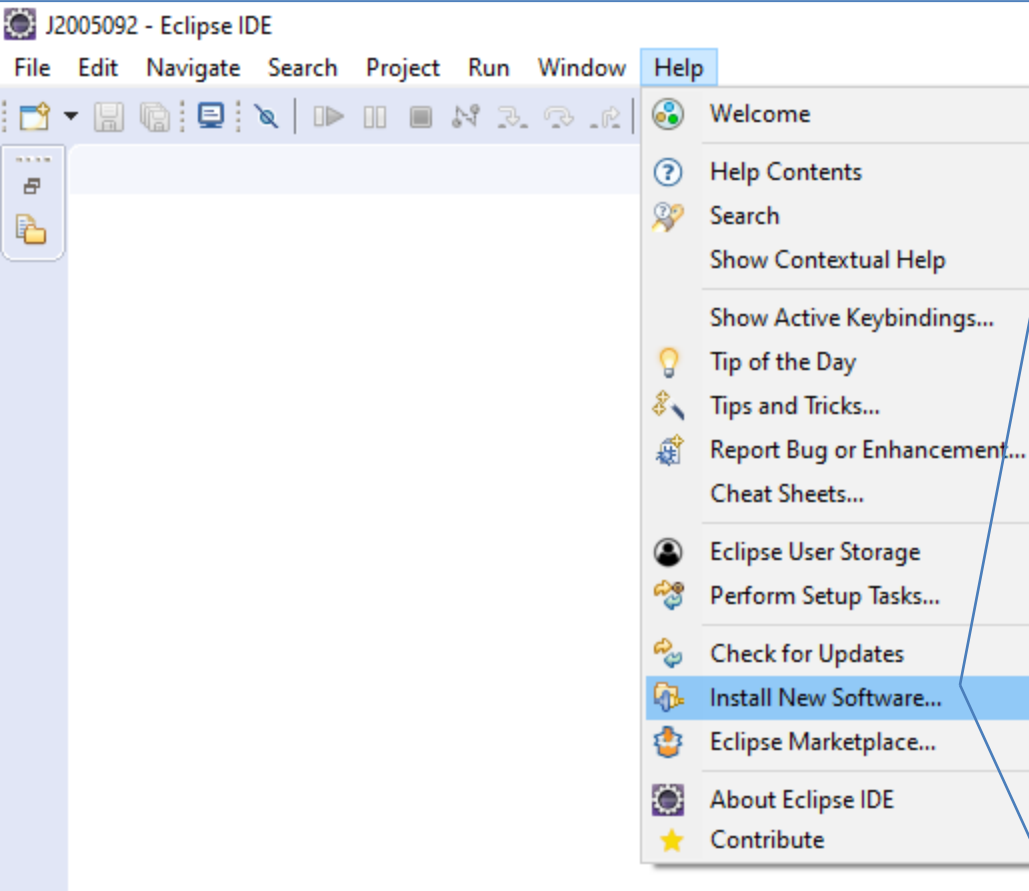
XML
Annotation
Configuration



ENVIRONMENT SETUP

- ❑ Để phát triển được dự án Spring Boot trong eclipse bạn cần công cụ hỗ trợ STS (Spring Tool Suite)
- ❑ STS là một plugin cho eclipse nó cung cấp một số template hỗ trợ phát triển ứng dụng
 - ❖ Chuẩn tắc, giảm lỗi
 - ❖ Giảm thời gian viết mã
- ❑ Sử dụng liên kết sau để tích hợp STS vào eclipse (slide sau)
 - ❖ <https://download.springsource.com/release/TOOLS/sts4/update/e4.12/>

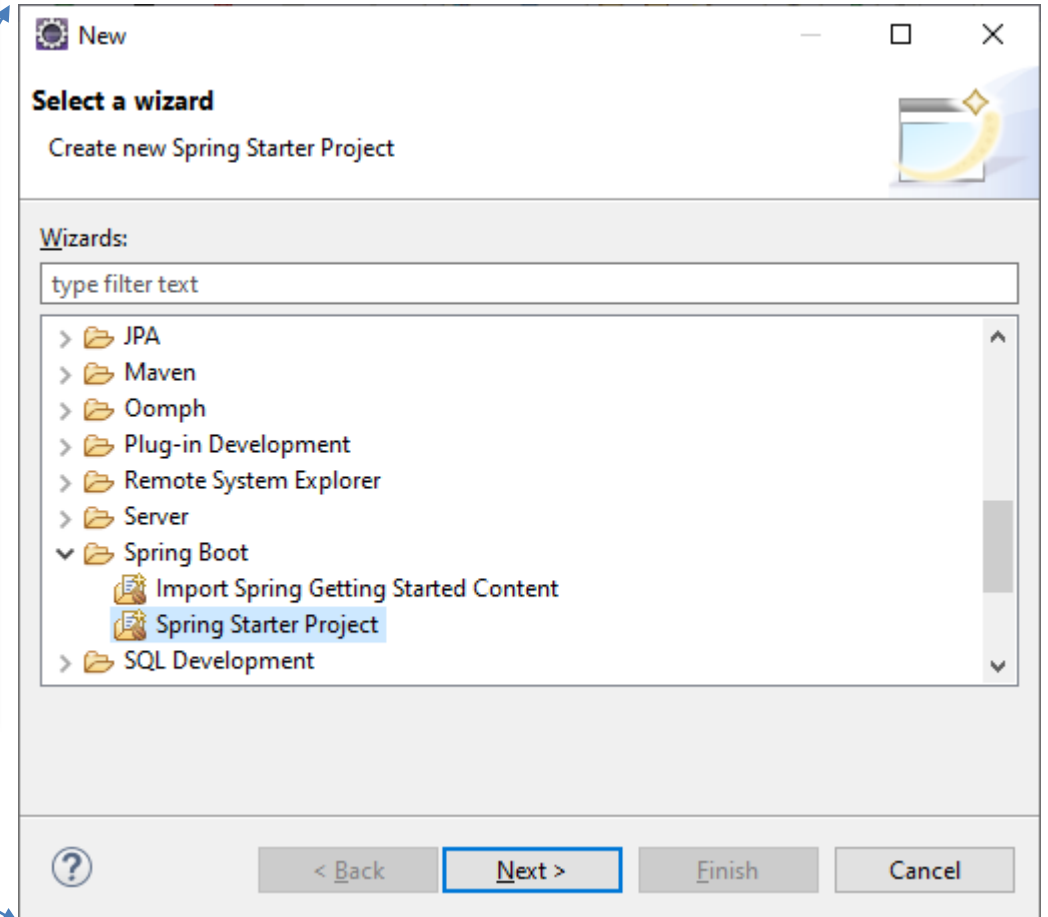
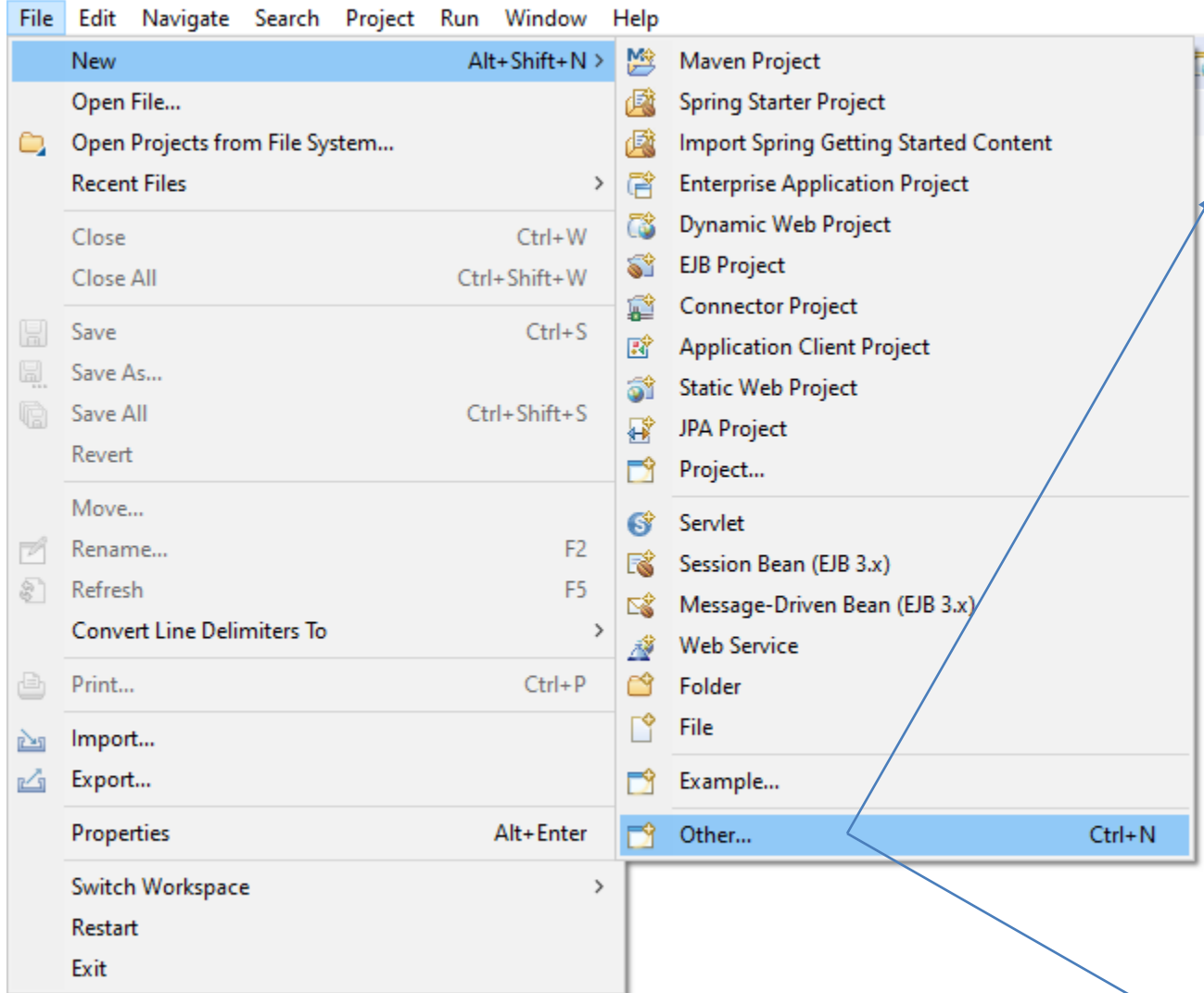
TÍCH HỢP STS VÀO ECLIPSE



Thực hiện một số bước, đợi cho đến khi eclipse tự restart lại

TẠO DỰ ÁN SPRING STARTER PROJECT

J2005092 - Eclipse IDE



TẠO DỰ ÁN SPRING STARTER

New Spring Starter Project

Service URL:

Name:

☒ Use default location

Location:

Type: Packaging:

Java Version: Language:

Group:

Artifact:

Version:

Description:

Package:

New Spring Starter Project Dependencies

Spring Boot Version:

Available:

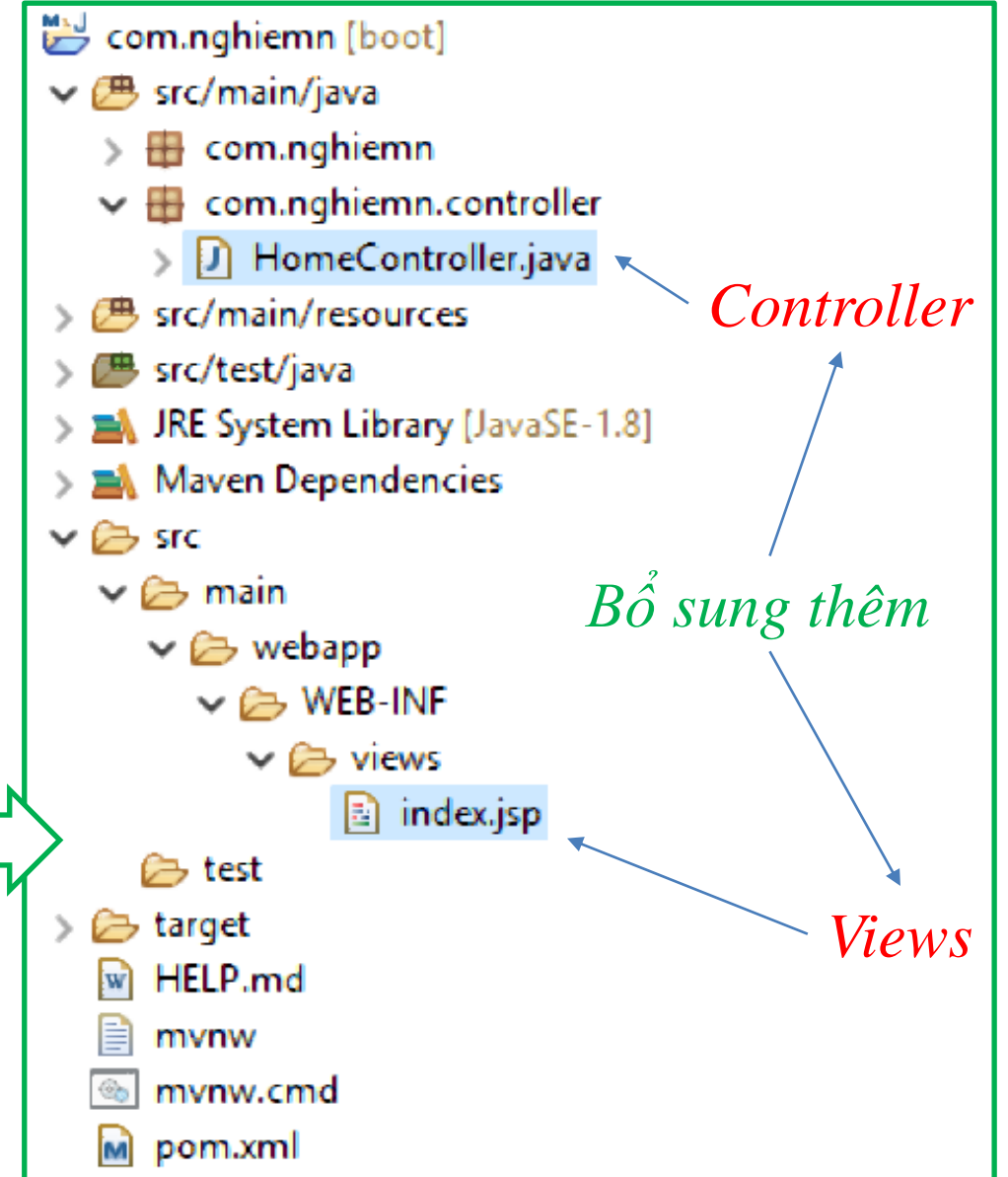
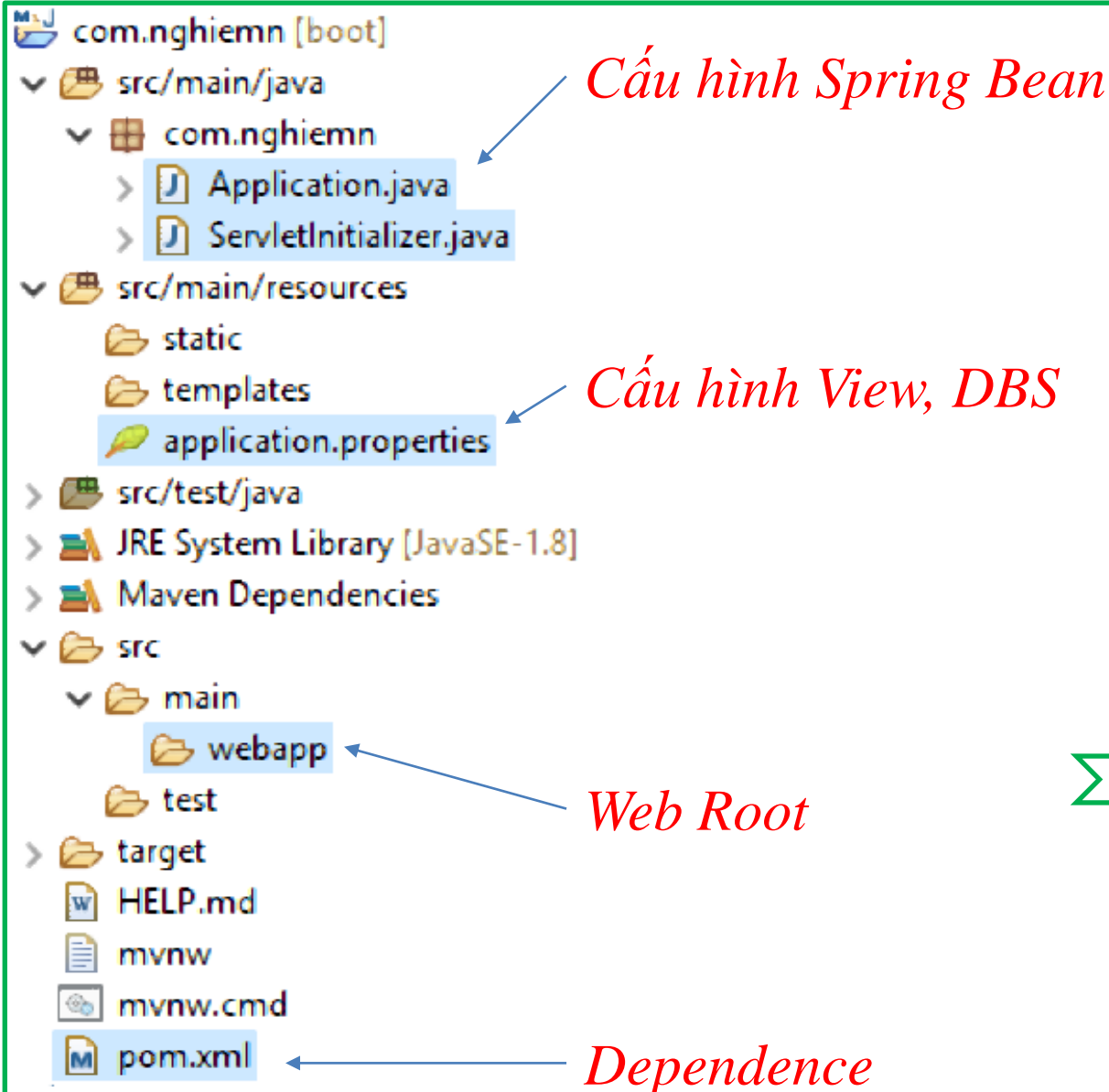
Type to search dependencies

Selected:

X Spring Web

- Spring Cloud Discovery
- Spring Cloud Messaging
- Spring Cloud Routing
- Spring Cloud Security
- Spring Cloud Tools
- Spring Cloud Tracing
- Template Engines
- Testing
- ☒ Web
 - ☒ Spring Web
 - ☐ Spring Reactive Web

CẤU TRÚC DỰ ÁN SPRING STARTER



1. CẤU HÌNH ỨNG DỤNG

@SpringBootApplication

```
public class Application {  
    public static void main(String[] args) {  
        SpringApplication.run(Application.class, args);  
    }  
}
```

Cấu hình app chạy bằng dòng lệnh

```
public class ServletInitializer extends SpringBootServletInitializer {  
    @Override  
    protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {  
        return application.sources(Application.class);  
    }  
}
```

Cấu hình app chạy trên web server

- ☐ pom.xml chứa các khai báo thư viện cần thiết (dependence) được sử dụng trong dự án
- ☐ Các thư viện này được Maven quản lý (download và update) tự động mỗi khi có thay đổi.
- ☐ Một số dependence được khai báo sẵn do lựa chọn trong quá trình tạo dự án.
- ☐ Trong quá trình phát triển, bạn cần phải khai báo thêm những thư viện cần thiết khác.


```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-tomcat</artifactId>
    <scope>provided</scope>
</dependency>
```















Nhúng tomcat web server vào dự án

```
<dependency>
    <groupId>org.apache.tomcat.embed</groupId>
    <artifactId>tomcat-embed-jasper</artifactId>
</dependency>
```

- ❑ application.properties là file cấu hình tài nguyên mặc định của ứng dụng.
- ❑ Ví dụ: Cấu hình JSTL View Resolver
 - ❖ Mặc định Spring Boot sử dụng Thymeleaf View Resolver để làm giao diện. Để sử dụng JSP bạn cần khai báo JSTL View Resolver như sau:

```
spring.mvc.view.prefix=/WEB-INF/views/  
spring.mvc.view.suffix=.jsp
```
 - ❖ Với khai báo này bạn phải có thư mục webapp/WEB-INF/views để chứa các file JSP

4.1 CONTROLLER & VIEW

- ▼  src/main/java
 - >  com.nghiemn
 - ▼  com.nghiemn.controller
 - >  HomeController.java
 - >  src/main/resources
 - >  src/test/java
 - >  JRE System Library [JavaSE-1.8]
 - >  Maven Dependencies
- ▼  src
 - ▼  main
 - ▼  webapp
 - ▼  WEB-INF
 - ▼  views
 -  index.jsp

Theo nguyên tắc “**Cấu hình mặc định**” thì **Controller** và các **Component** phải có tên package bắt đầu bởi tên package của cấu hình bean.

Nếu đặt trong package khác thì phải khai báo đường dẫn cụ thể

application.properties

```
spring.mvc.view.prefix=/WEB-INF/views/  
spring.mvc.view.suffix=.jsp
```

4.2 CONTROLLER & VIEW

@Controller

```
public class HomeController {  
    @RequestMapping("/index.php")  
    public String index(Model model) {  
        model.addAttribute("message", "Welcome to Spring MVC");  
        return "index";  
    }  
}
```

@Controller chứa Web Method

/index.php <=> index()

Model chứa dữ liệu chia sẻ với View

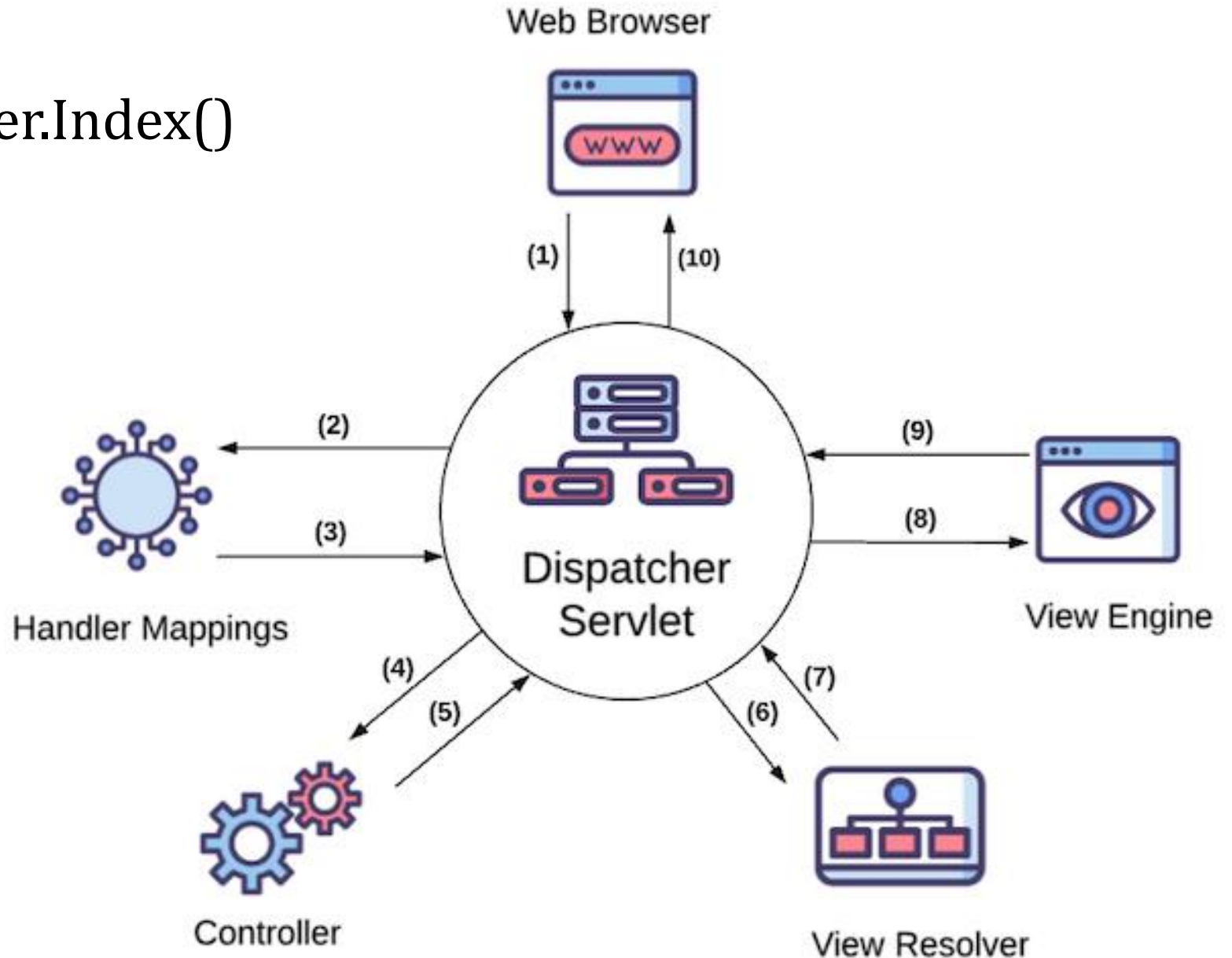
/WEB-INF/views/index.jsp

View Resolver

```
<%@ page pageEncoding="utf-8"%>  
<!DOCTYPE html>  
<html>  
<head><title>Spring Boot</title></head>  
<body>  
    <h1>Home Page</h1>  
    <h3>${message}</h3>  
</body>  
</html>
```

QUY TRÌNH XỬ LÝ REQUEST & RESPONSE

- ❑ 1 & 2: .../index.php
- ❑ 3 & 4: HelloController.Index()
- ❑ 5 & 6: "home/index"
- ❑ 7 & 8: ".../index.jsp"
- ❑ 9 & 10: HTML





SPRING STATER PROJECT

GIẢNG VIÊN:



HTTP COMPONENTS



HttpServletRequest

- Parameters, Cookies, Header, URL, Attributes...

HttpServletResponse

- Cookies, Headers, TEXT/HTML...

HttpSession

- Attributes

ServletContext

- Attributes, ContextPath...

- ❑ Các các đối tượng http (request, response, session, application) được tham chiếu và sử dụng trong Spring MVC theo 2 cách:

```
@Controller
public class MyController {
    @Autowired
    ServletContext application;
    @Autowired
    HttpSession session;
    @Autowired
    HttpServletRequest request;
    @Autowired
    HttpServletResponse response;
    @RequestMapping("/url.php")
    public String sayHello() {
        String fullname = request.getParameter("hoten");
        request.setAttribute("message", "Http Components");
        return "hello";
    }
}
```

```
@Controller
public class MyController {
    @Autowired
    ServletContext application;
    @RequestMapping("/url.php")
    public String sayHello(HttpSession session,
        HttpServletRequest request, HttpServletResponse response;) {
        String fullname = request.getParameter("hoten");
        request.setAttribute("message", "Http Components");
        return "hello";
    }
}
```



PACKAGING & DEPLOYING

PACKAGING WEB APPLICATION

The screenshot illustrates the steps to package a web application in Eclipse:

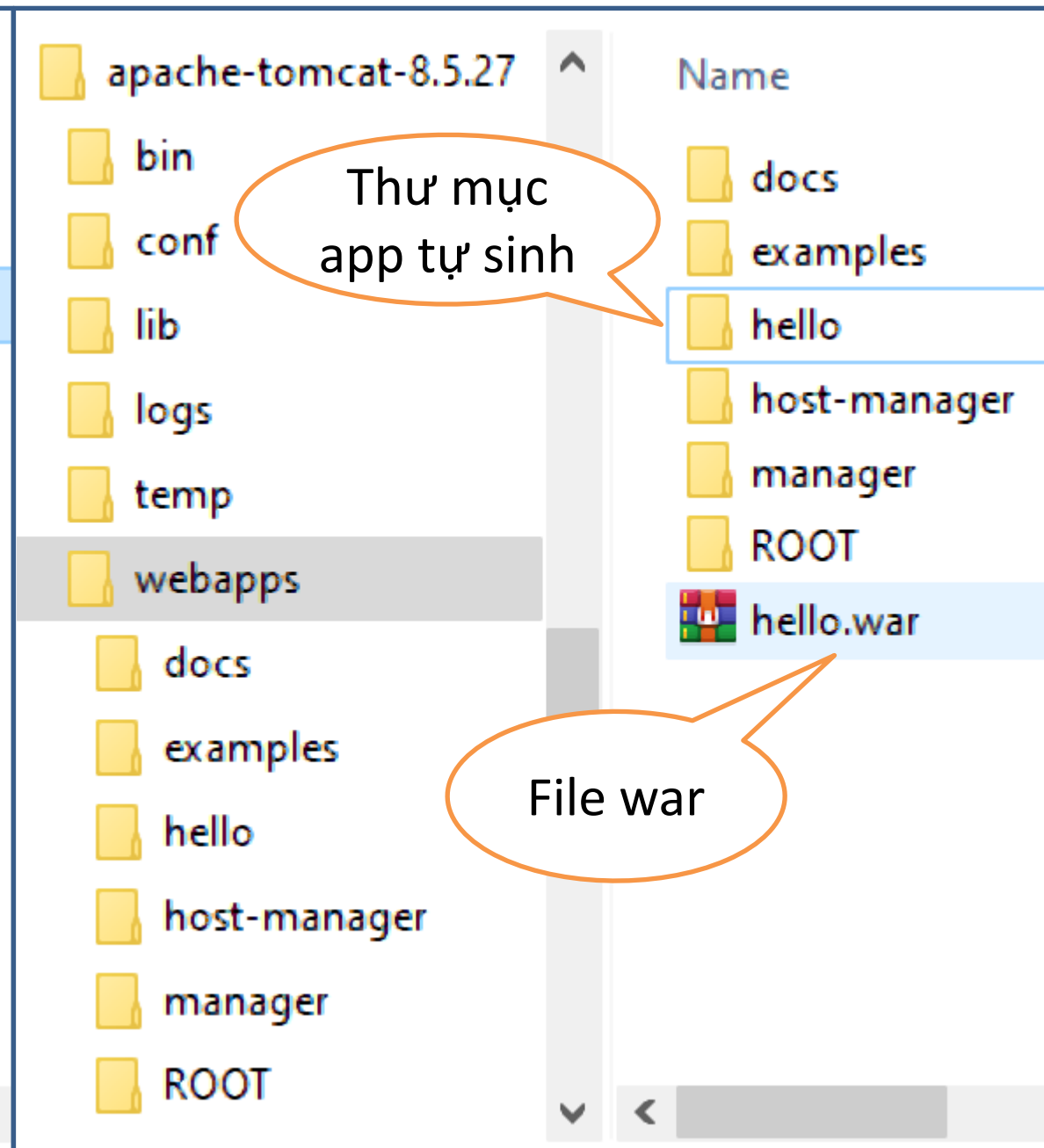
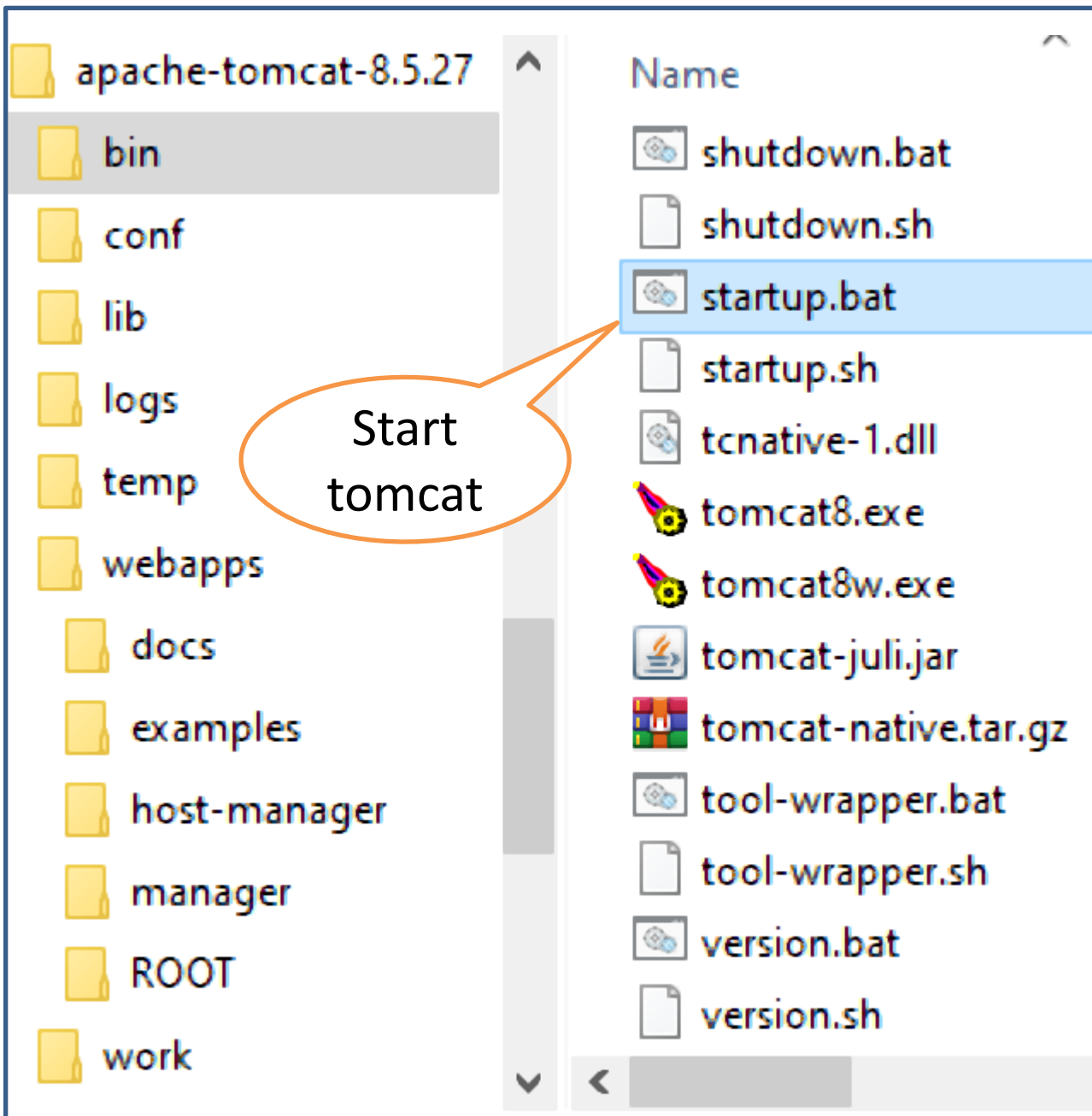
- Project Structure:** The left sidebar shows the project 'com.nghiemn [boot]' with folders for 'src/main/java', 'src/main/resources', 'src/test/java', 'JRE System Library', 'Maven Dependencies', 'src', and 'target'. Files include 'HELP.md', 'mvnw', 'mvnw.cmd', and 'pom.xml'.
- Export Menu:** The 'Export...' option is selected from the context menu.
- WAR Export Wizard:**
 - Web project:** com.nghiemn
 - Destination:** C:\Users\Admin\Desktop\com.nghiemn.war
 - Buttons:** Back, Next >, Finish, Cancel.
- Progress:** The 'Next >' button is highlighted, indicating the next step in the wizard.

Chạy bin/startup.bat để Start Tomcat

Chép file *.war vào thư mục webapps

Chạy trình duyệt và gõ URL phù hợp

Chạy bin/shutdown.bat để Shutdown Tomcat





LOMBOK & DEVTOOL

❑ Lombok là API công cụ hỗ trợ các IDE sinh code java (getters, setters và constructors) tự động dựa vào các annotation.

❖ @Getter, @Setter, @Data: sinh getters/setters

❖ @NoArgsConstructor và @AllArgsConstructor: sinh constructors

❑ Để sử dụng Lombok trong eclipse cần thực hiện 2 bước

❖ Bước 1: Cài đặt Lombok cho eclipse (tải và chạy file sau)

➤ <https://projectlombok.org/downloads/lombok.jar>

❖ Bước 2: Khai báo thư viện cần thiết

```
<dependency>  
  <groupId>org.projectlombok</groupId>  
  <artifactId>lombok</artifactId>  
  <optional>true</optional>  
</dependency>
```

@Getter @Setter

@NoArgsConstructor @AllArgsConstructor

@Table(name = "Categories")

@Entity

```
public class Category implements Serializable {  
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)  
    Integer id;  
    String name;  
    @Column(name = "Namevn")  
    String nameVN;  
    @OneToMany(mappedBy = "category")  
    List<Product> products;  
}
```


- ❑ DevTool là Tool API giúp eclipse tự dịch khi code thay đổi bên trong dự án.
- ❑ Cần phải khai báo thư viện cần thiết cho dự án

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-devtools</artifactId>  
  <scope>runtime</scope>  
  <optional>true</optional>  
</dependency>
```

- ❑ GIỚI THIỆU SPRING, SPRING MVC VÀ SPRING BOOT
- ❑ TÍCH HỢP STS
- ❑ TẠO DỰ ÁN SPRING BOOT
 - ❑ TÌM HIỂU FILE CẤU HÌNH ỨNG DỤNG
 - ❑ KHAI BÁO THƯ VIỆN CẦN THIẾT POM.XML
 - ❑ KHAI BÁO VIEW APPLICATION.PROPERTIES
 - ❑ TẠO CONTROLLER
 - ❑ TẠO VIEW
 - ❑ RUN PROJECT
- ❑ TÌM HIỂU QUY TRÌNH XỬ LÝ REQUEST/RESPONSE
- ❑ SỬ DỤNG CÁC THÀNH PHẦN WEB
- ❑ ĐÓNG GÓI VÀ TRIỂN KHAI





FPT POLYTECHNIC

Thank you