ÔN THI CUỐI KỲ.

Nội dung thi: Spring MVC, Spring Rest, Hibernate, SQL Server, JSP

## PHẦN 1: ÔN LẠI KIẾN THỨC 1 CHÚT

Sring MVC: là một framework hỗ trợ xây dựng website theo mô hình MVC

Spring REST: dùng để tạo server cung cấp xử lí data cho nhiều client app khác.

REST (Representational State Transfer) api sử dụng các giao thức HTTP để gửi request cho server, không đòi hỏi ngôn ngữ lập trình hay bất công nghệ nào. Dùng đúng cú pháp là gọi được dữ liệu từ REST.

Dữ liệu được trả về từ REST app thường là xml hoặc JSON. (JSON phổ biến hơn).

JSON (JavaScript Object Notaion):

* Vd: {

“id”:14,

“fisrtName”: “Mario”,

“sex”: true,

“favoriteFood”: [“Chicken”, “BBQ”],

“address”: {

“street”: “12 au co, p 10 , q tan binh”,

“city”: “Ho Chi Minh”

}

}

Java JSON Data Binding: Là quá trình chuyển đổi qua lại JSON sang Java POJO (Class Java chỉ có thuộc và get set).

* Sử dụng thư viện Jackson để làm điều đó.

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-databind</artifactId>

<version>2.9.5</version>

</dependency>

REST trên HTTP:

* POST : tạo mới entity
* GET : đọc danh sách của đối tượng
* PUT : cập nhật dữ liệu cho 1 mẫu tin nào đó
* DELETE : xóa 1 dữ liệu nào đó.

Mỗi HTTP Messages request tới server sẽ được trả về dưới dạng http messages response

Cấu trúc HTTP Request Message gồm 3 phần:

* Request line: phương thức
* Header variables: metadata (cho biết định dạng của dữ liệu)
* Message body: nội dung gửi đi

Cấu trúc HTTP Response Message gồm 3 phần:

* Response line: giao thức và trạng thái.
* Header variables: metadata.
* Message body: nội dung trả về.

Một số trạng thái respone line:

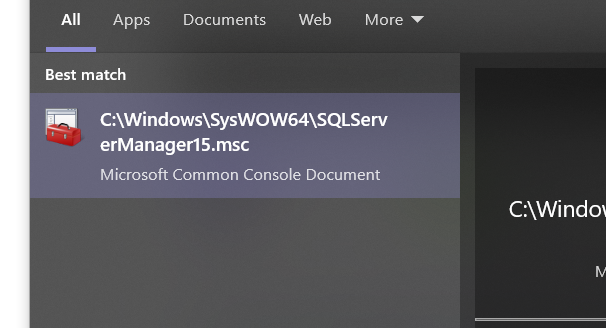
* 100-199: Infomational.
* 200-299: Successful.
* 300-399: Redirection.
* 400 – 499: Client error.
* 500 – 599: Server error.

Spring REST api design: được thiết kế xoay quanh một đối tượng như customer có dạng api/cusmers/{id} và các phương thức post put delete get

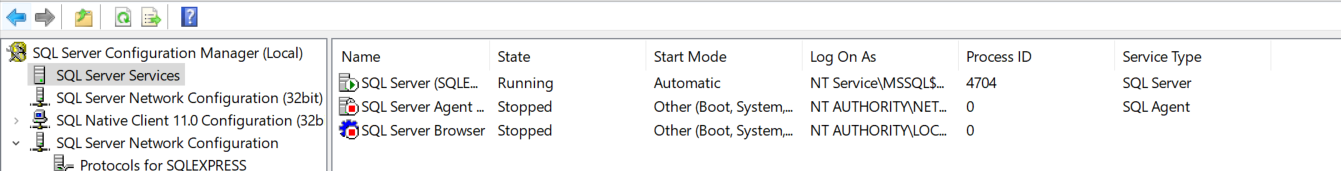
## PHẦN 2: CHIẾN THÔI

### TẠO DATABASE VÀ TẢI APACHE TOMCAT SERVER ĐƠN GIẢN.

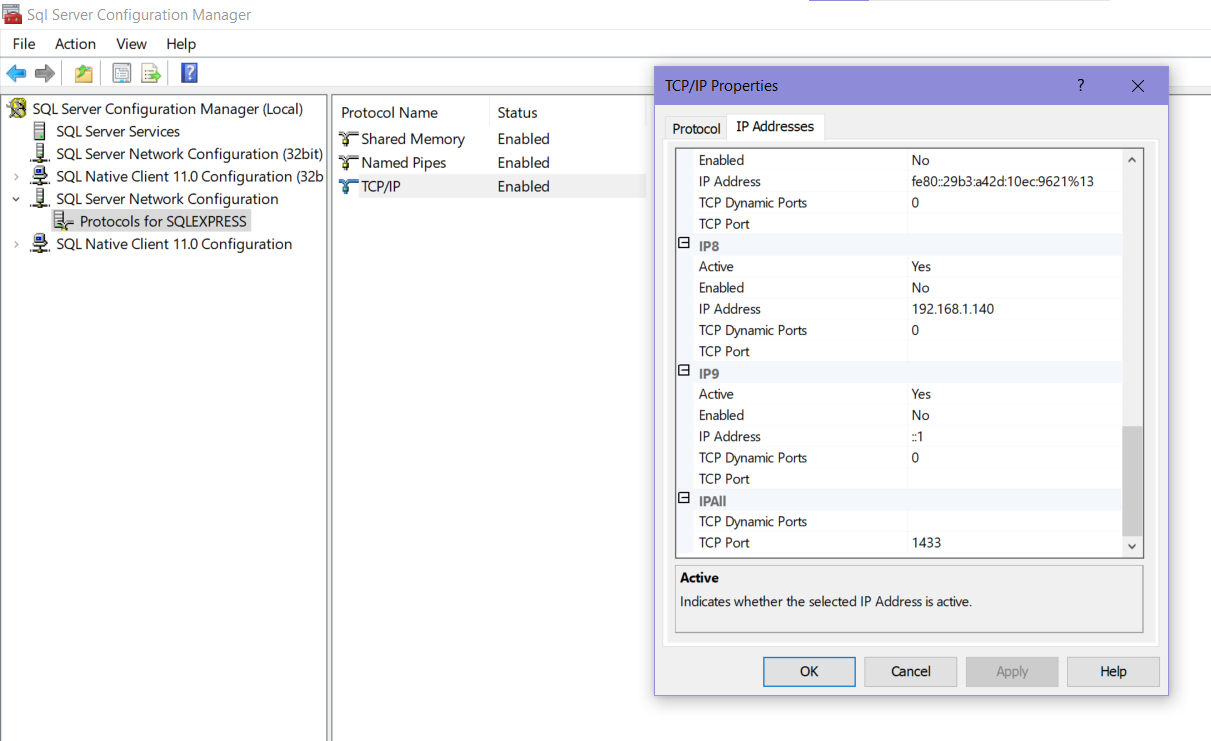
Config SQL Server:

Bước 1: mở 

Bước 2: Đảm báo dịch vụ đang chạy như hình



Bước 3: Chỉnh port 1433:



CREATE DATABASE web-customer-tracker

create table customer (

id INT IDENTITY(1,1) PRIMARY KEY,

first\_name VARCHAR(50),

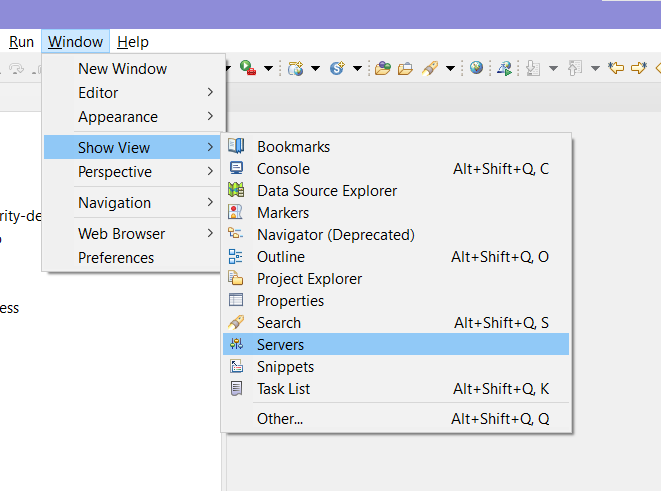
last\_name VARCHAR(50),

email VARCHAR(50),

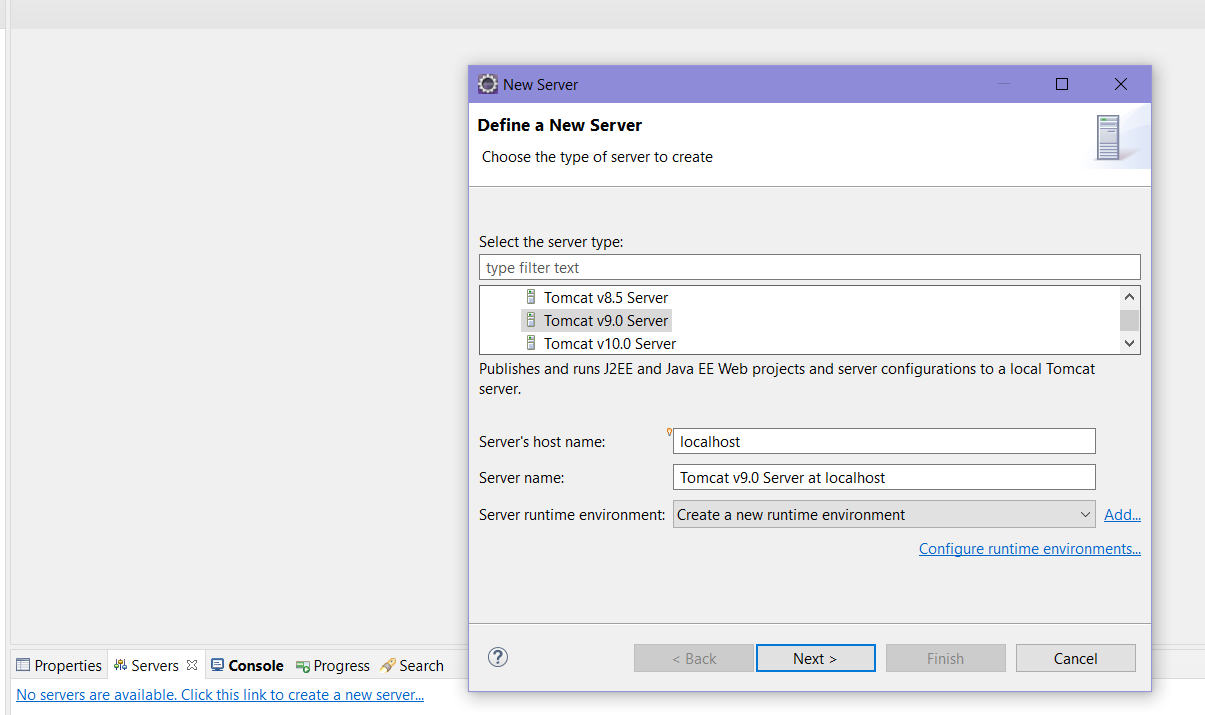
)

Cách tải tomcat server:

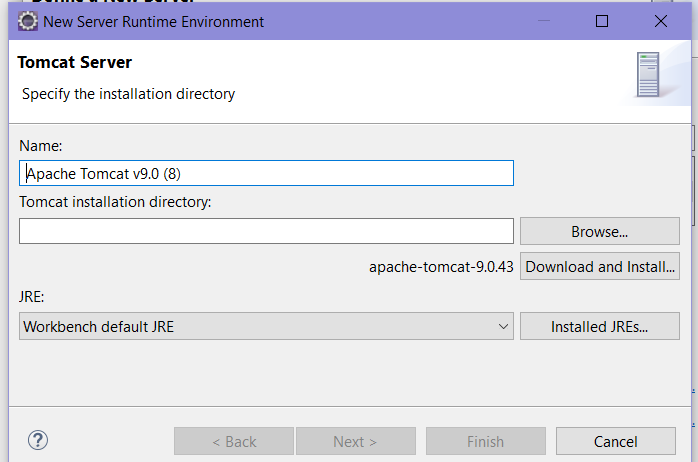
Bước 1:



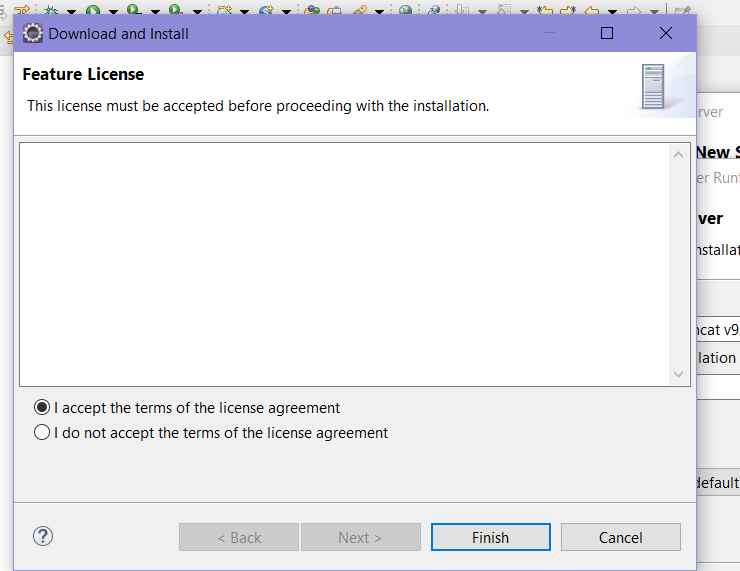
Bước 2: Click tạo mới. Chọn TomCat 9.



Bước 3: Bấm Add hiện lên bảng sau rồi bấm download and install:

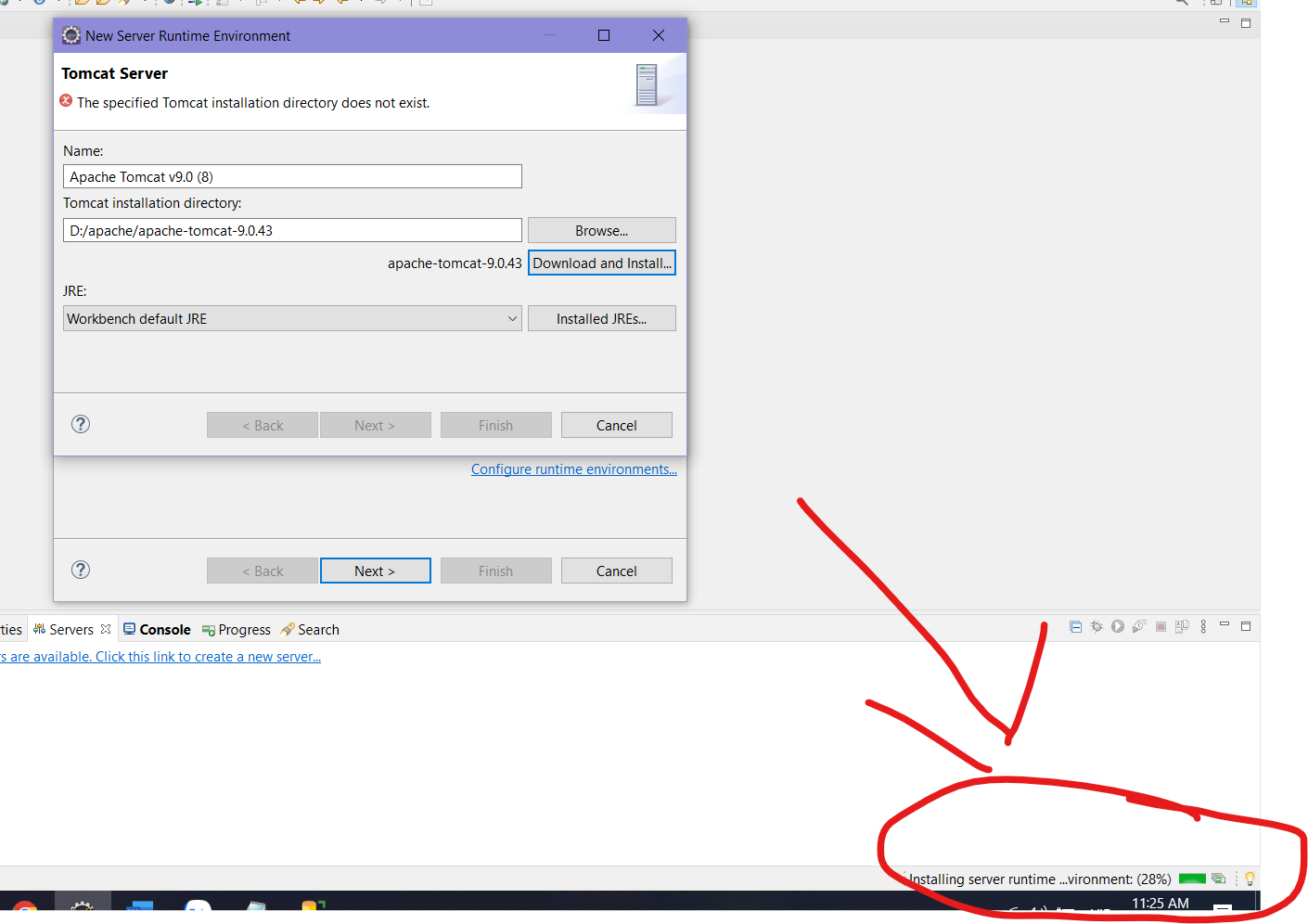


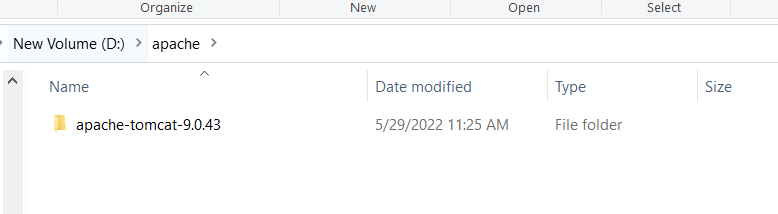
Bước 4: Bấm accept và finish



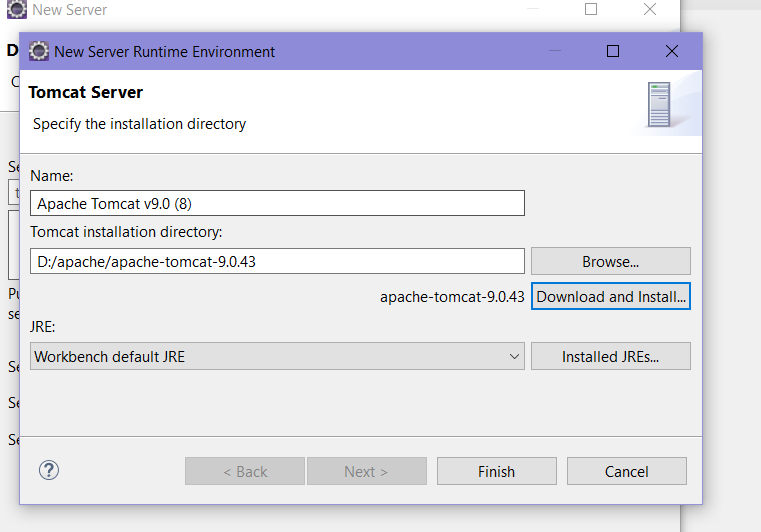
Bước 5: Chọn folder chứa server.

Bước 6: Chờ tải về:



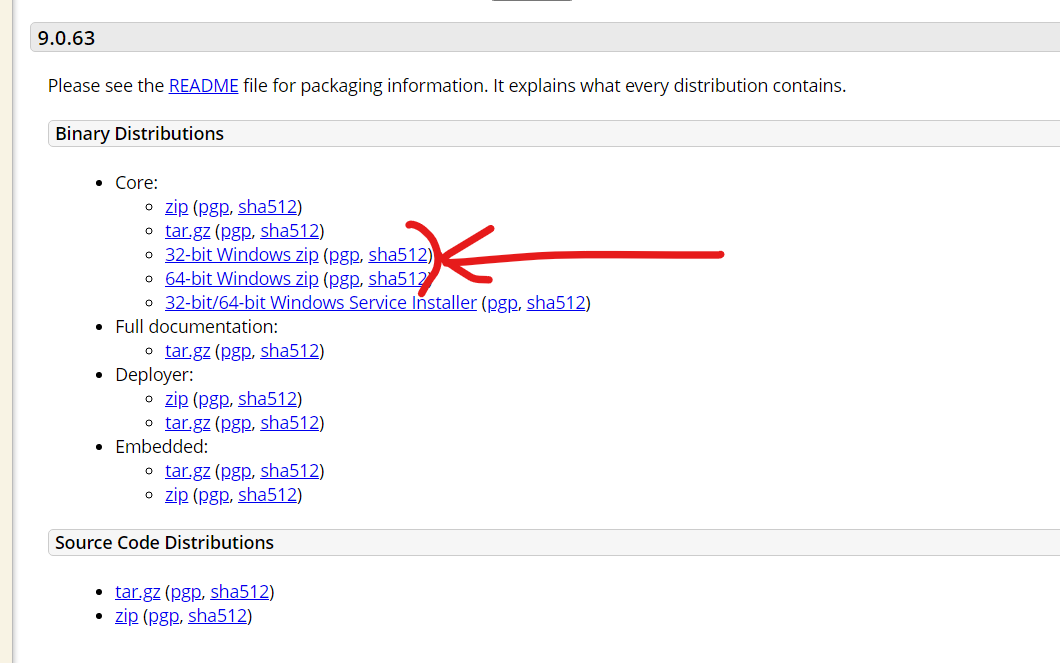


Bước 7: Bấm finish kết thúc:

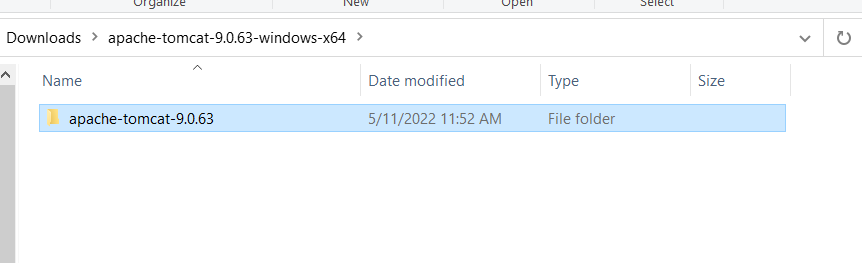


Cách 2:

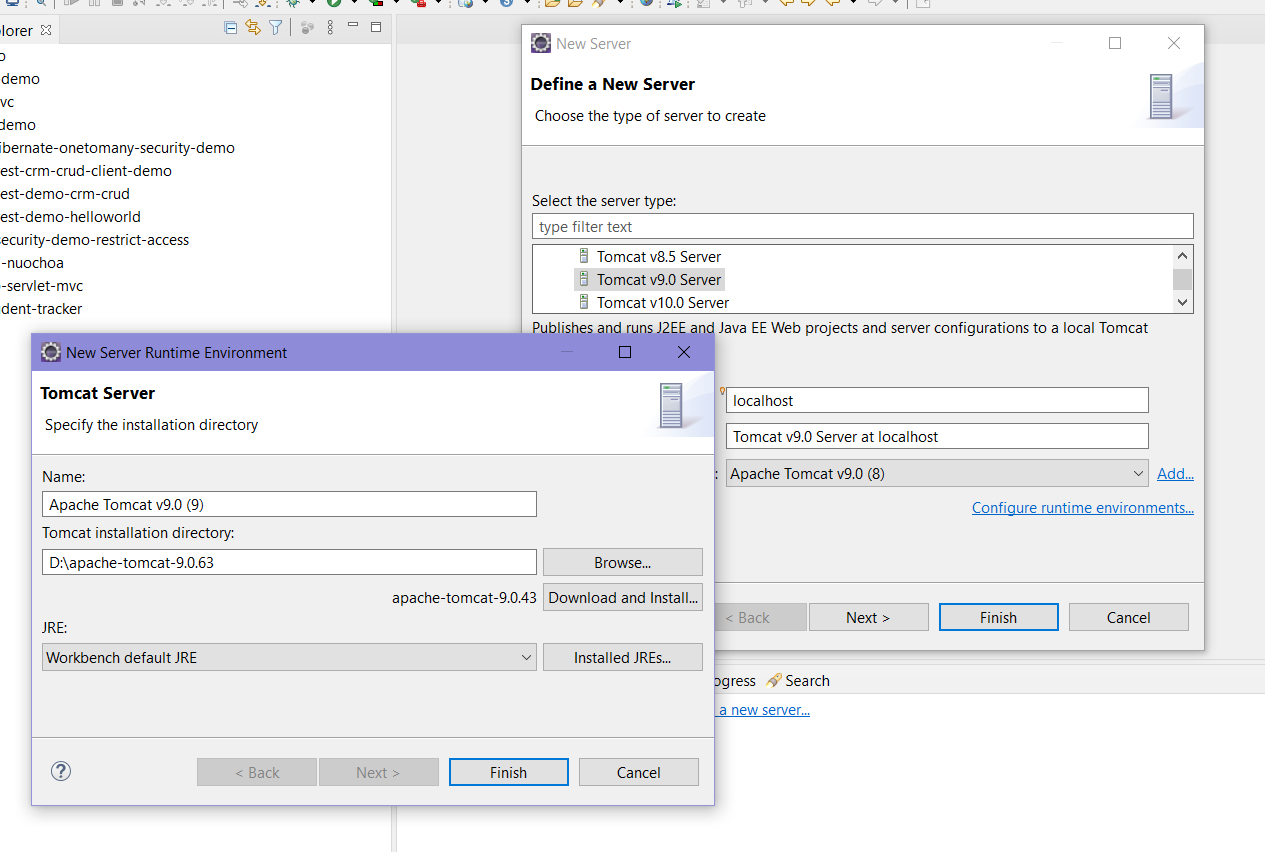
Bước1: Lên web <https://tomcat.apache.org/download-90.cgi> tải về.



Bước 2: Giải nén



Bước 3: Mở create server lên bấm add tomcat 9, browser để dẫn tới folder vừa giải nén.



Bước 4: bấm finish kết thúc.

### SERVER

Đề bài: tạo một ứng dụng quản lí customer và có các chức năng thêm sửa xóa.

Đảm bảo đã thêm server và chạy sql trước khi làm.

Bước 1: Tạo Maven Project (War)

Bước 2: Add thêm dữ liệu vào file pom và update project:

<build>

<plugins>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.1</version>

<configuration>

<source>11</source>

<target>11</target>

</configuration>

</plugin>

<plugin>

<artifactId>maven-war-plugin</artifactId>

<version>3.3.1</version>

<configuration>

<failOnMissingWebXml>false</failOnMissingWebXml>

</configuration>

</plugin>

</plugins>

</build>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<springframework.version>5.2.16.RELEASE</springframework.version>

<maven.compiler.source>11</maven.compiler.source>

<maven.compiler.target>11</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>${springframework.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-tx</artifactId>

<version>${springframework.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>${springframework.version}</version>

</dependency>

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-databind</artifactId>

<version>2.9.5</version>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.4.0.Final</version>

</dependency>

<dependency>

<groupId>com.microsoft.sqlserver</groupId>

<artifactId>mssql-jdbc</artifactId>

<version>8.4.1.jre11</version>

</dependency>

<dependency>

<groupId>com.mchange</groupId>

<artifactId>c3p0</artifactId>

<version>0.9.5.2</version>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>3.1.0</version>

</dependency>

<dependency>

<groupId>javax.servlet.jsp</groupId>

<artifactId>javax.servlet.jsp-api</artifactId>

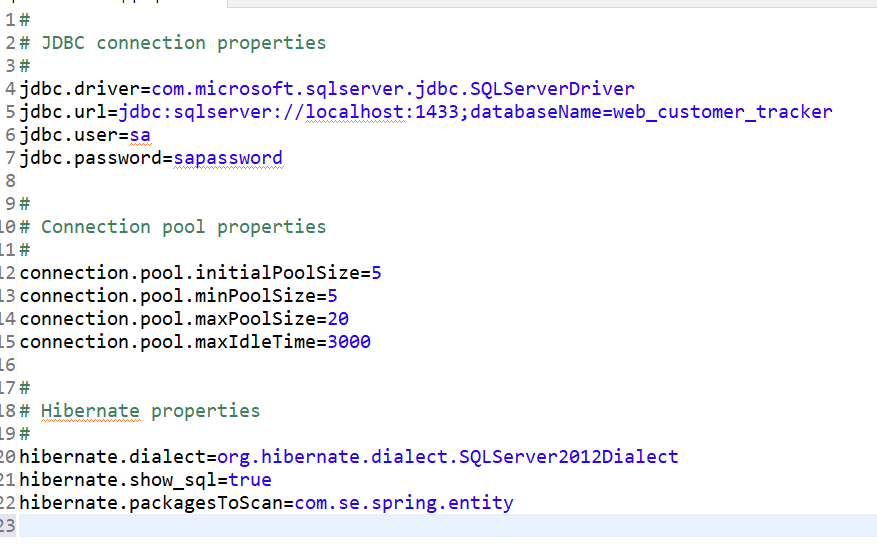
<version>2.3.1</version>

</dependency>

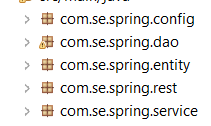
</dependencies>

Bước 3: Tạo file properties ở resoucse:

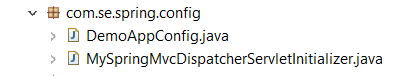




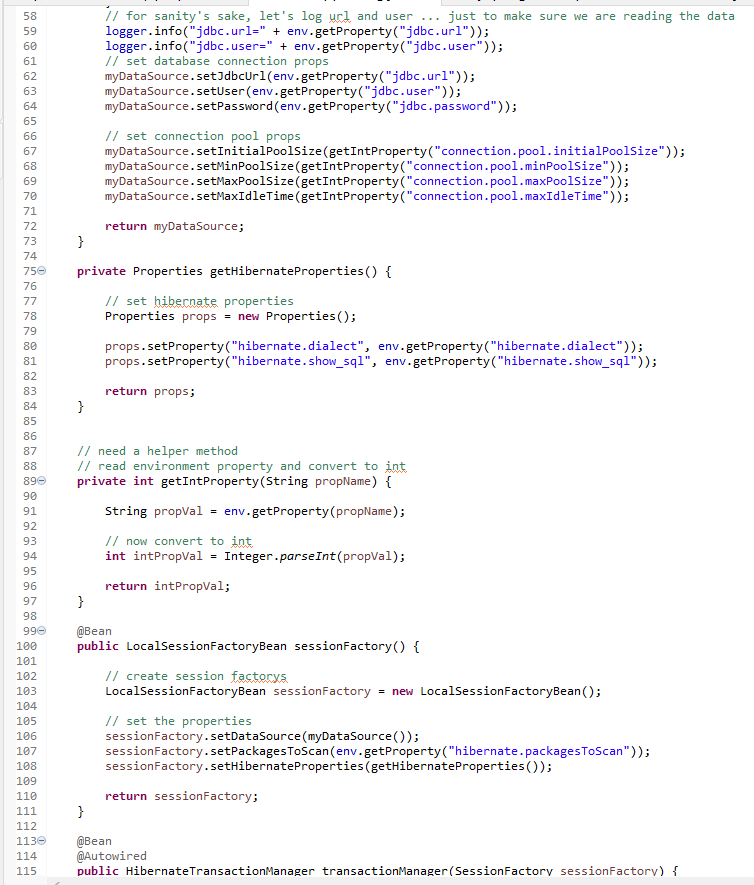
Bước 4: Tạo cấu trúc package như sau:



Bước 5: Tạo các class config:

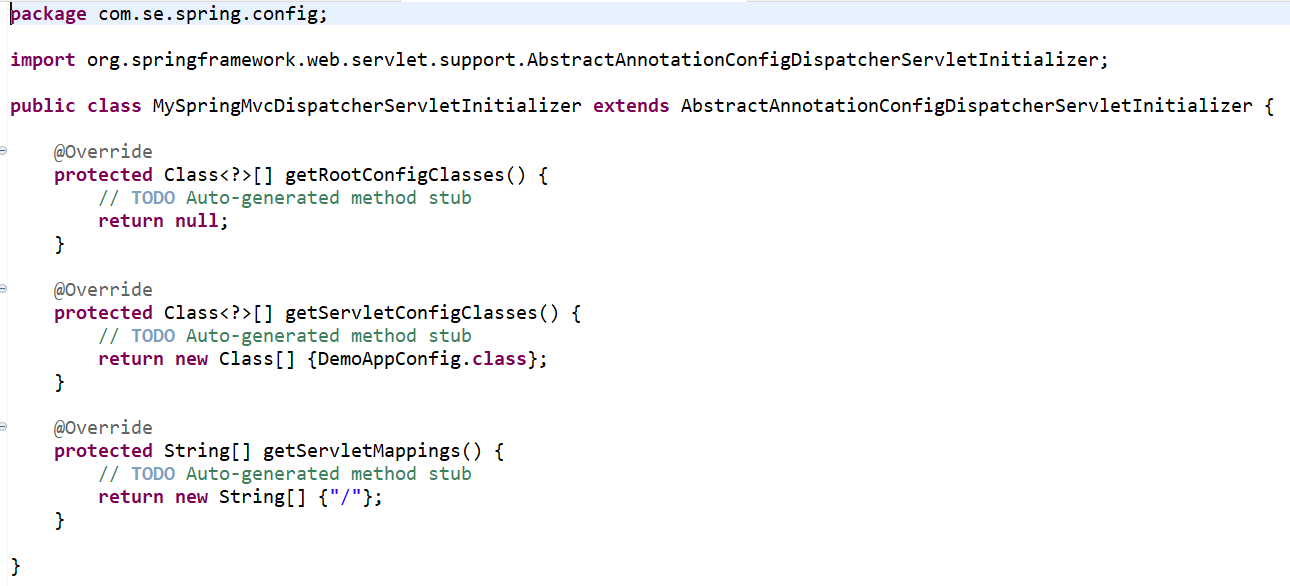


Bỏ phần logger đi.

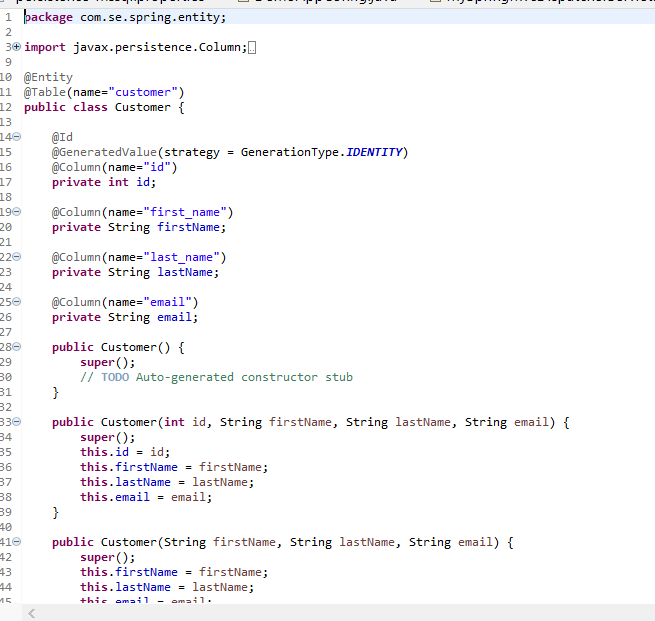


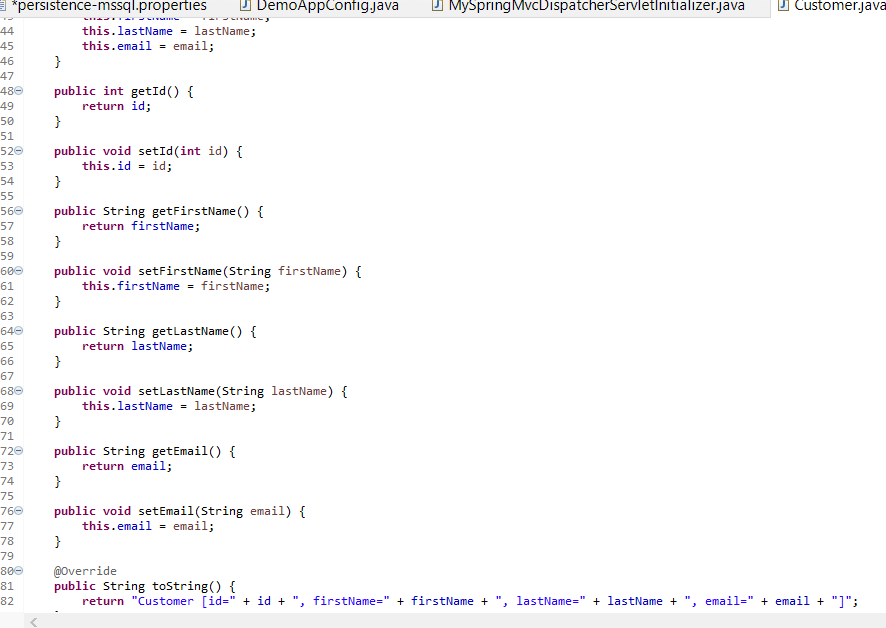
File Initailizer



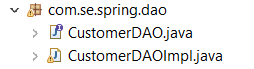
Bước 6: Tạo các POJO Class customer:

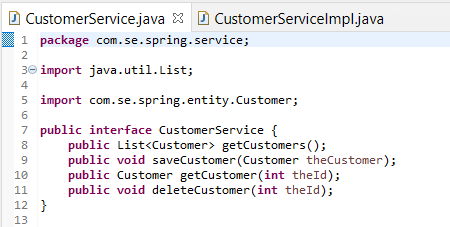


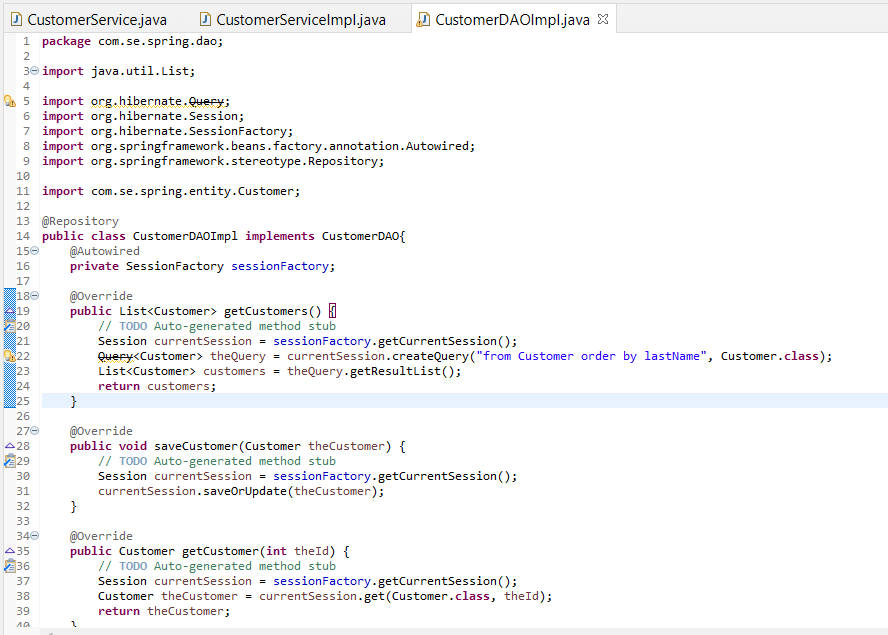


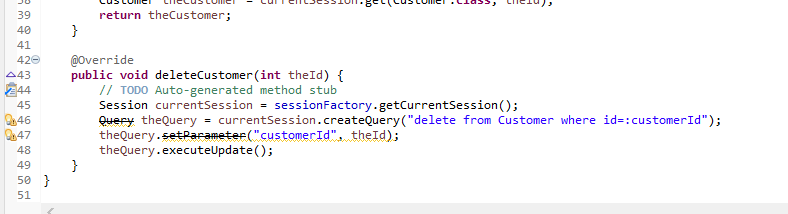


Bước 7: Tạo DAO:

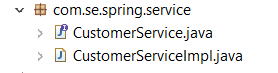


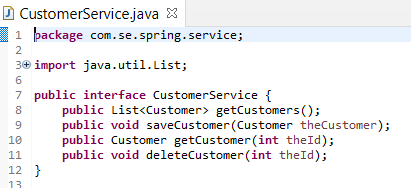


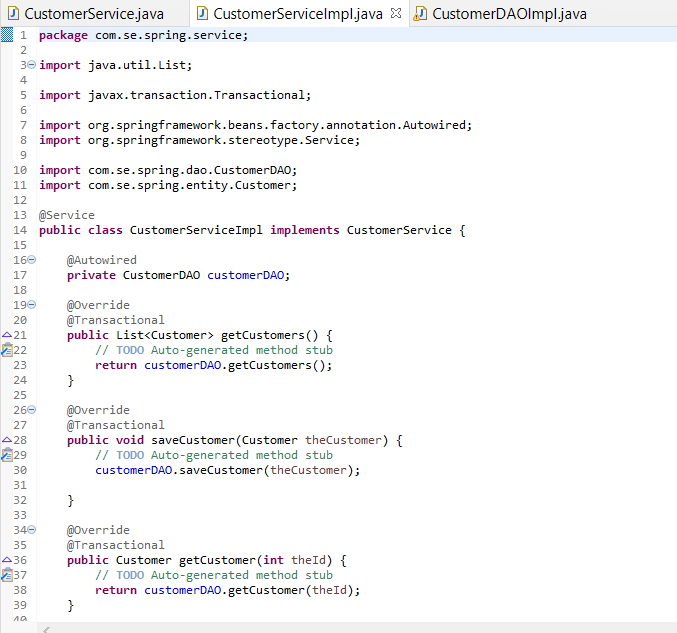


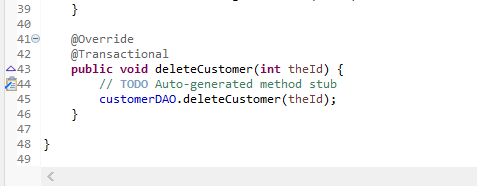


Bước 8: Tạo service:



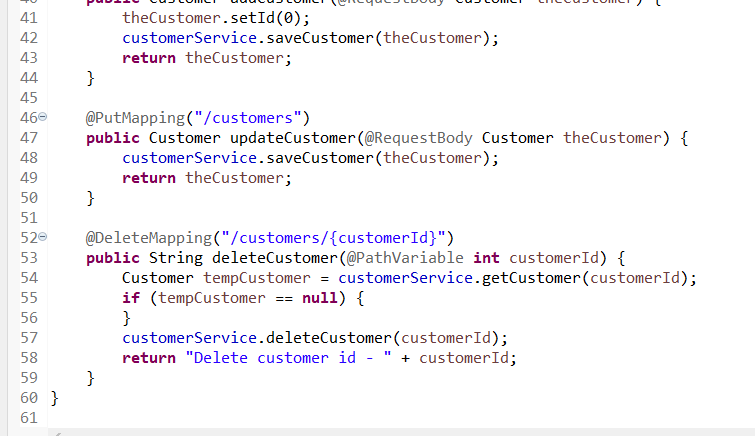


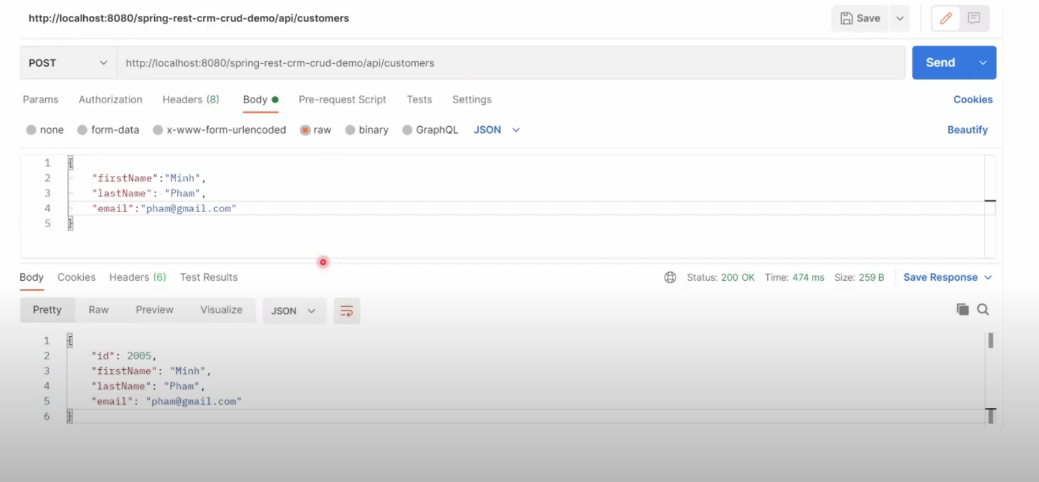




Bước 9: Tạo RestController.





Bước 10: Run as server và test trên postman

### CLIENT

Bước 1: Tạo Maven Project (War).

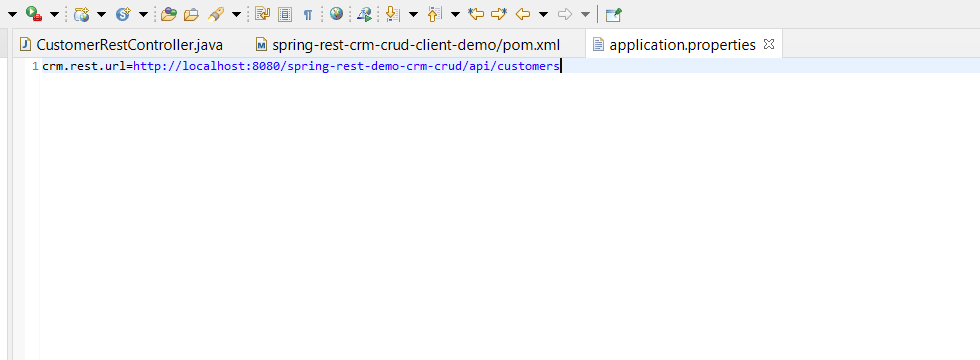
Bước 2: config file pom.xml.



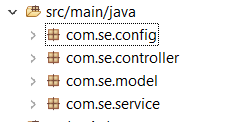


Bước 3: Tạo file kết nối:

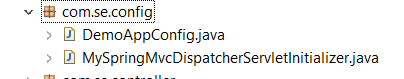


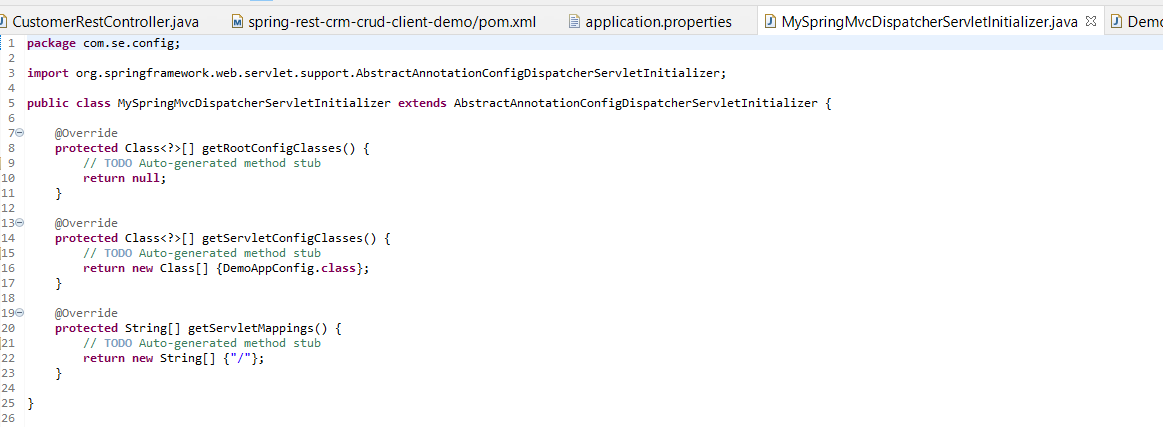


Bước 4: Tạo cấu trúc package như sau:



Bước 5: Tạo 2 file config:

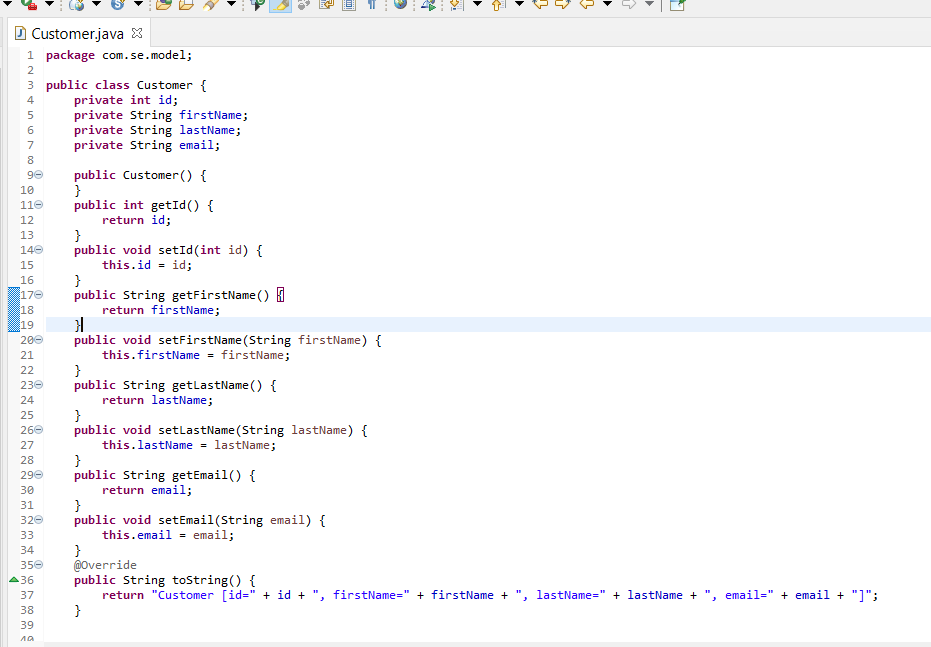




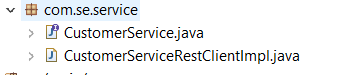


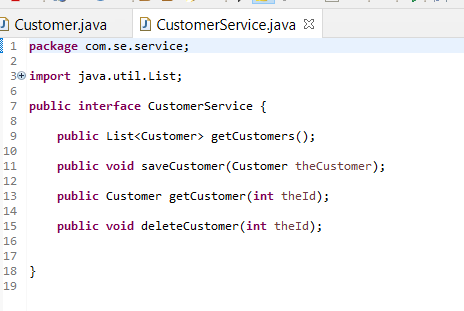
Bước 6: Tạo Model:

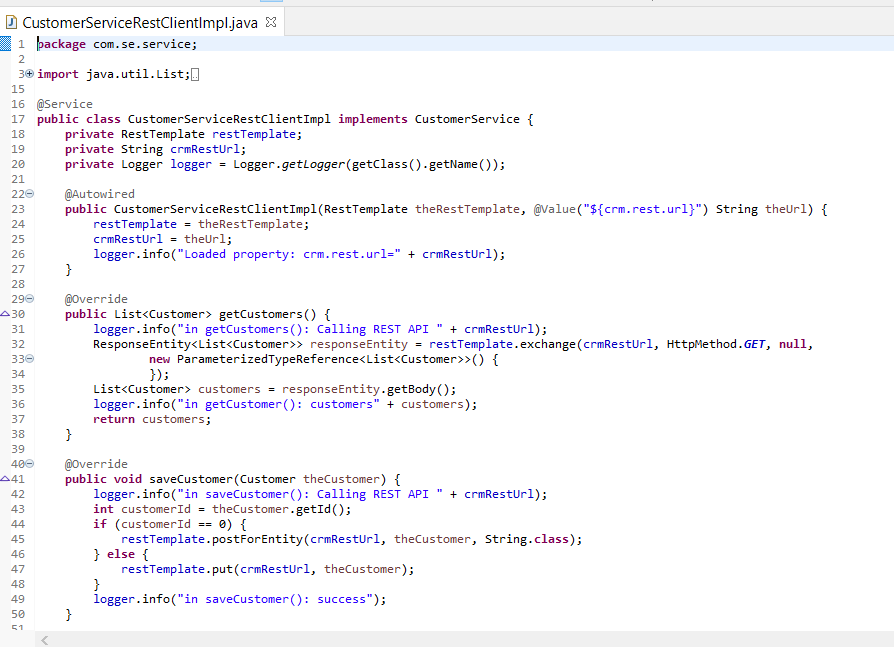


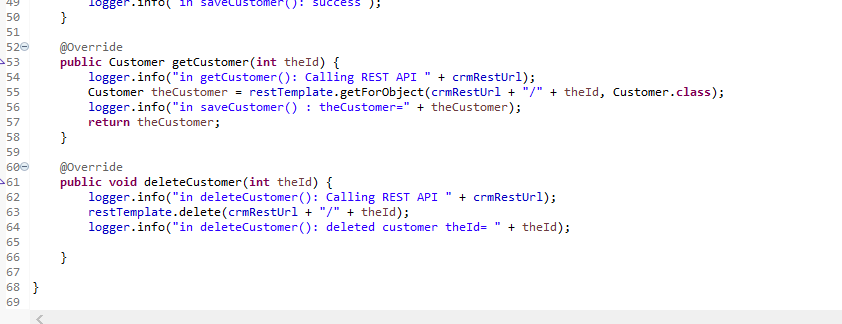


Bước 7: Tạo service:

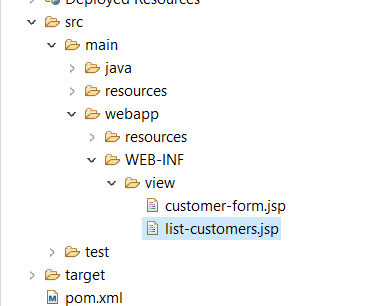


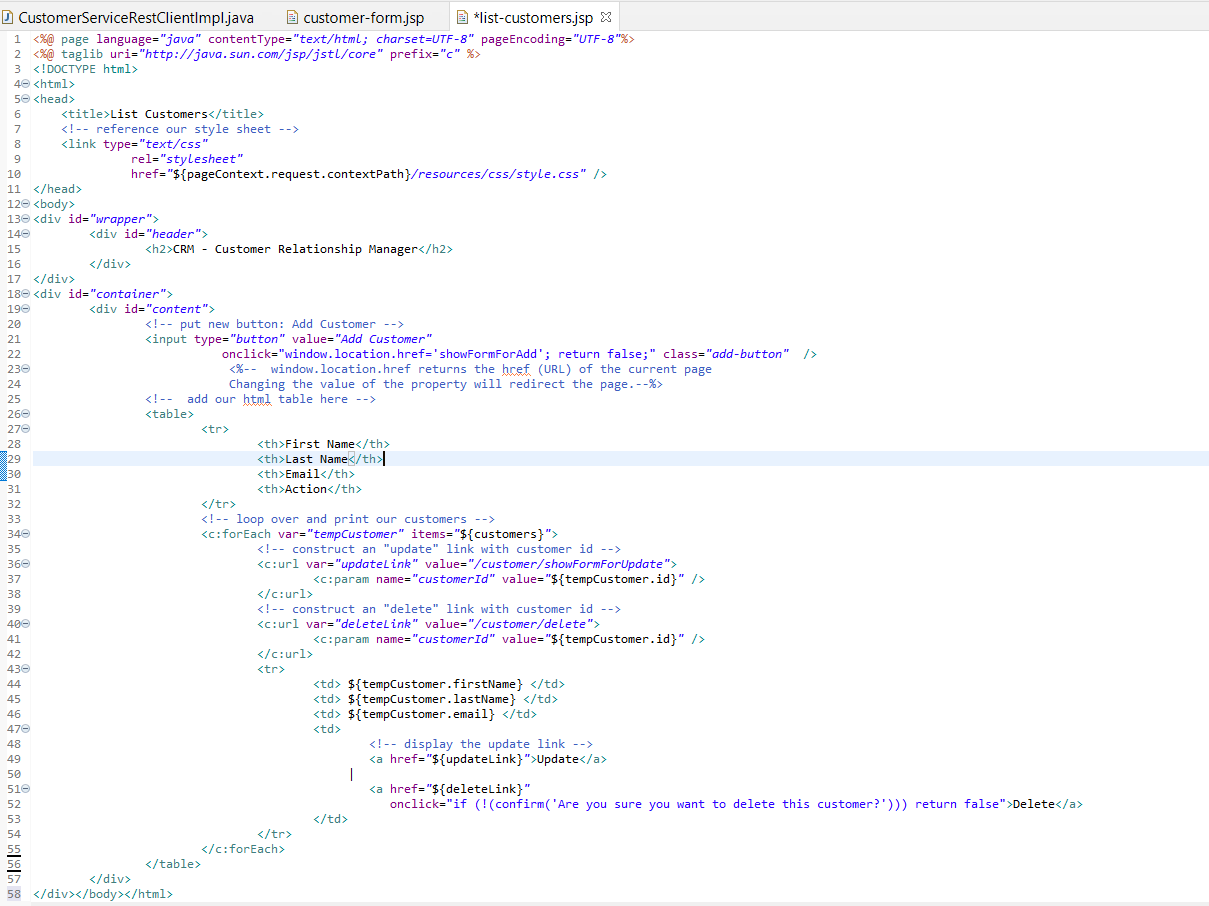


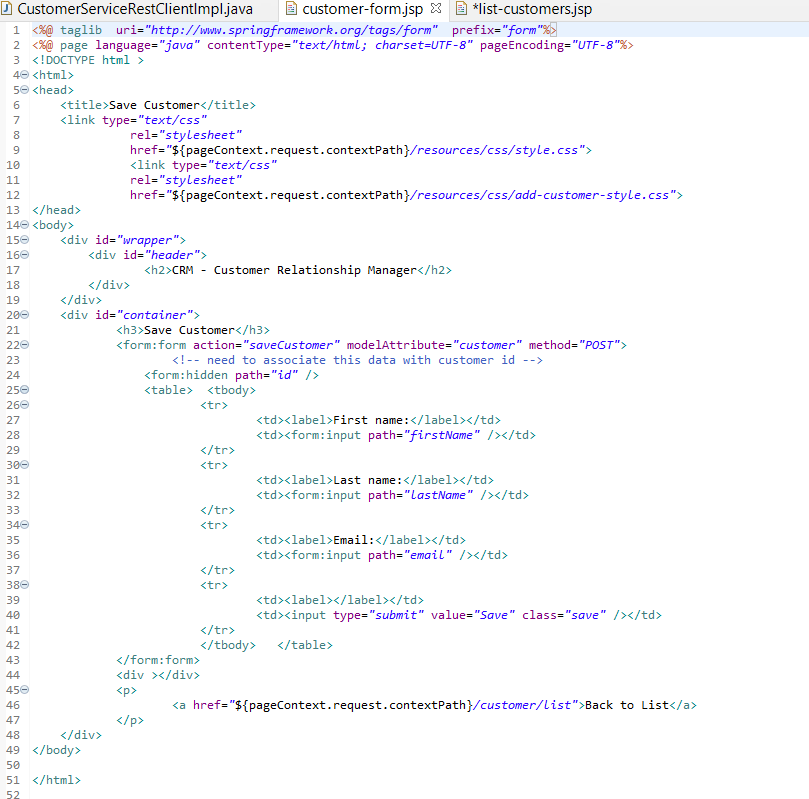




Bước 8: Tạo JSP







Bước 10: Tạo Controller



