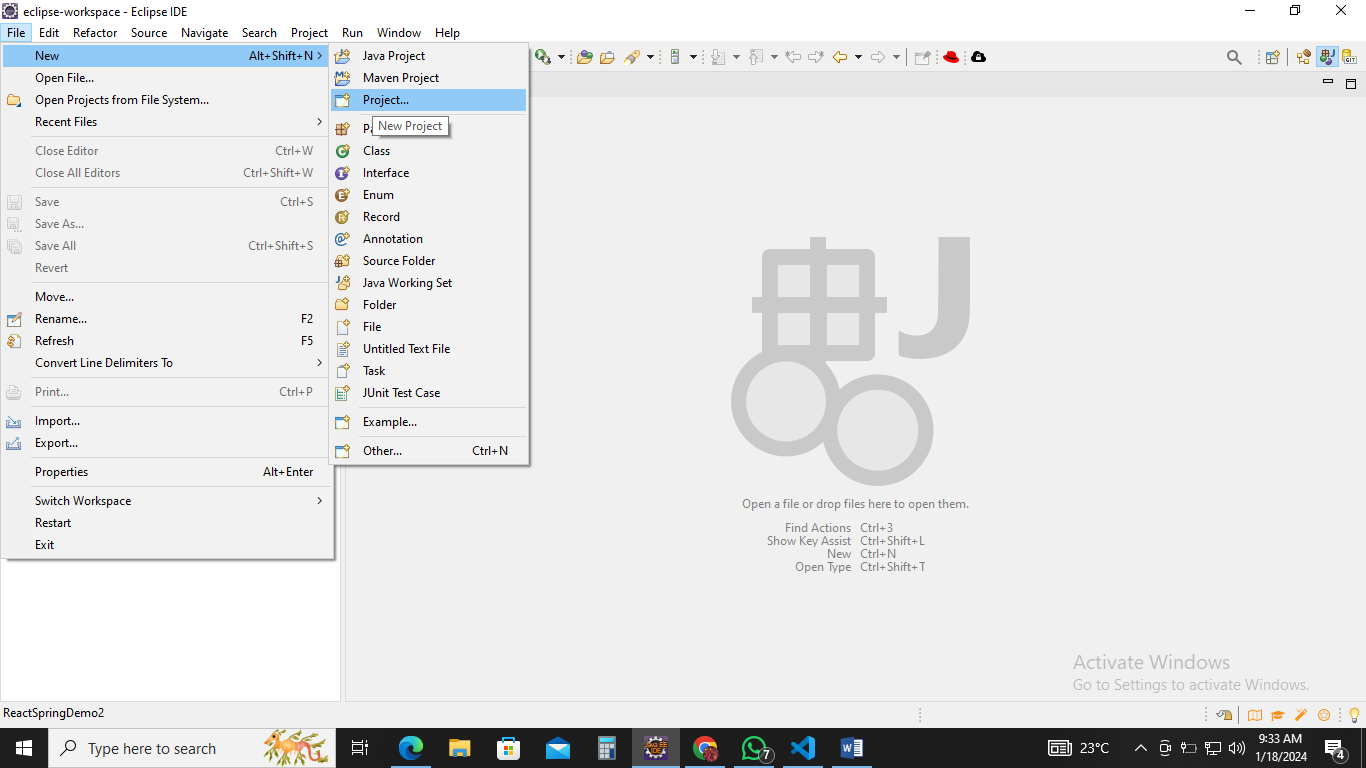
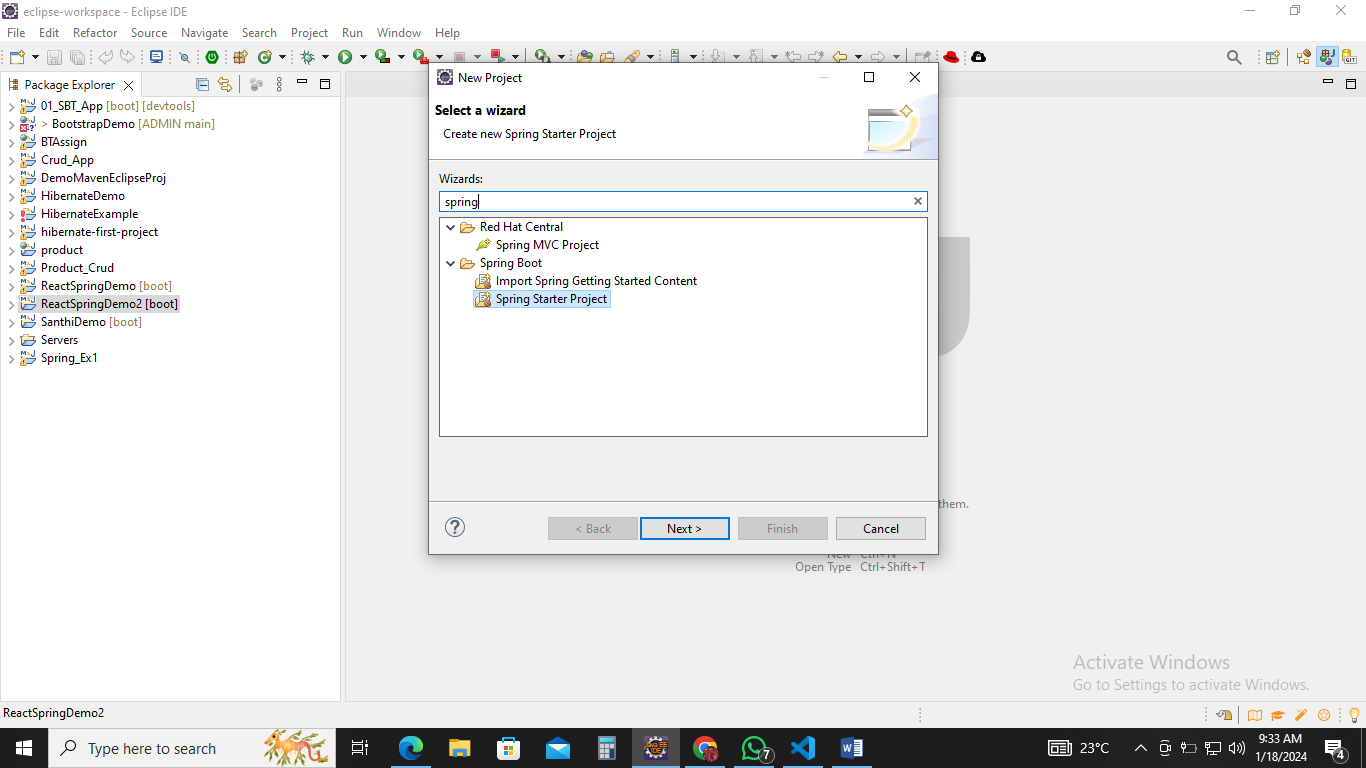
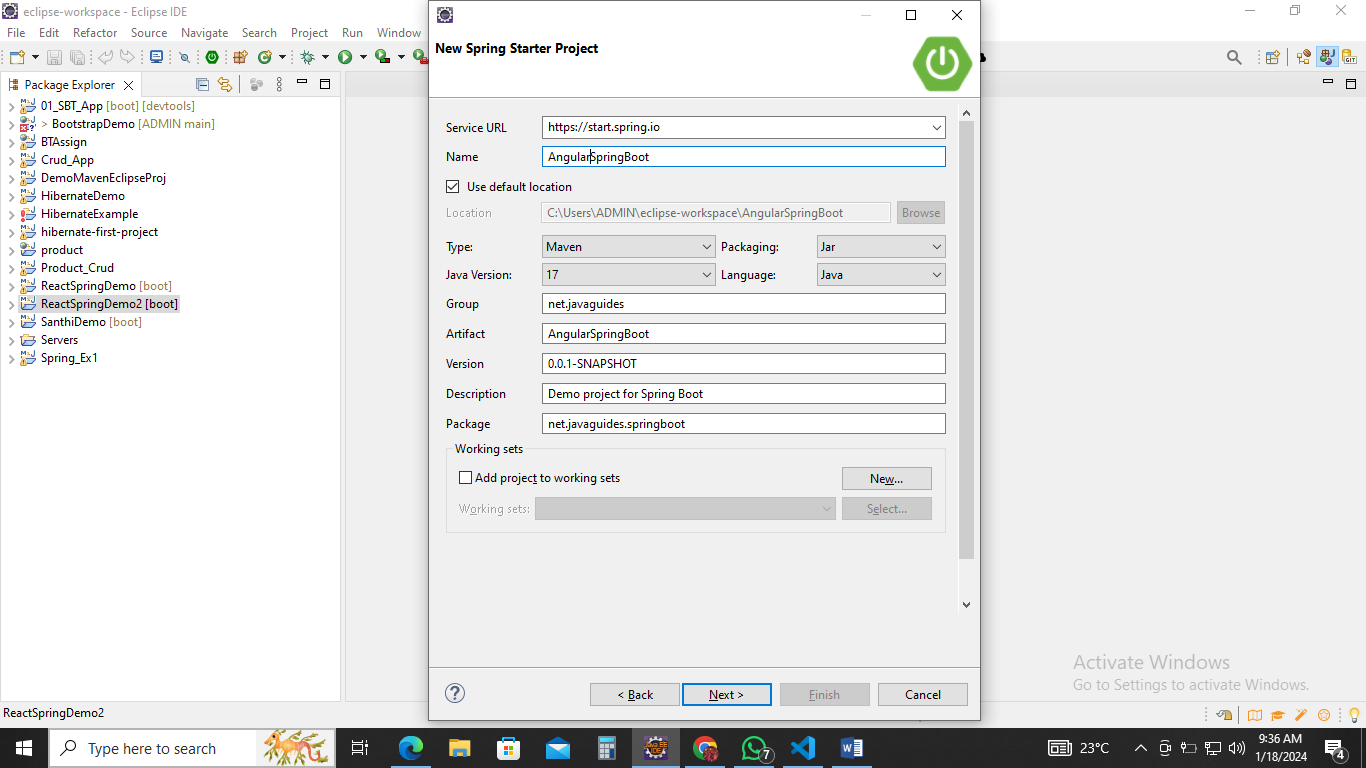
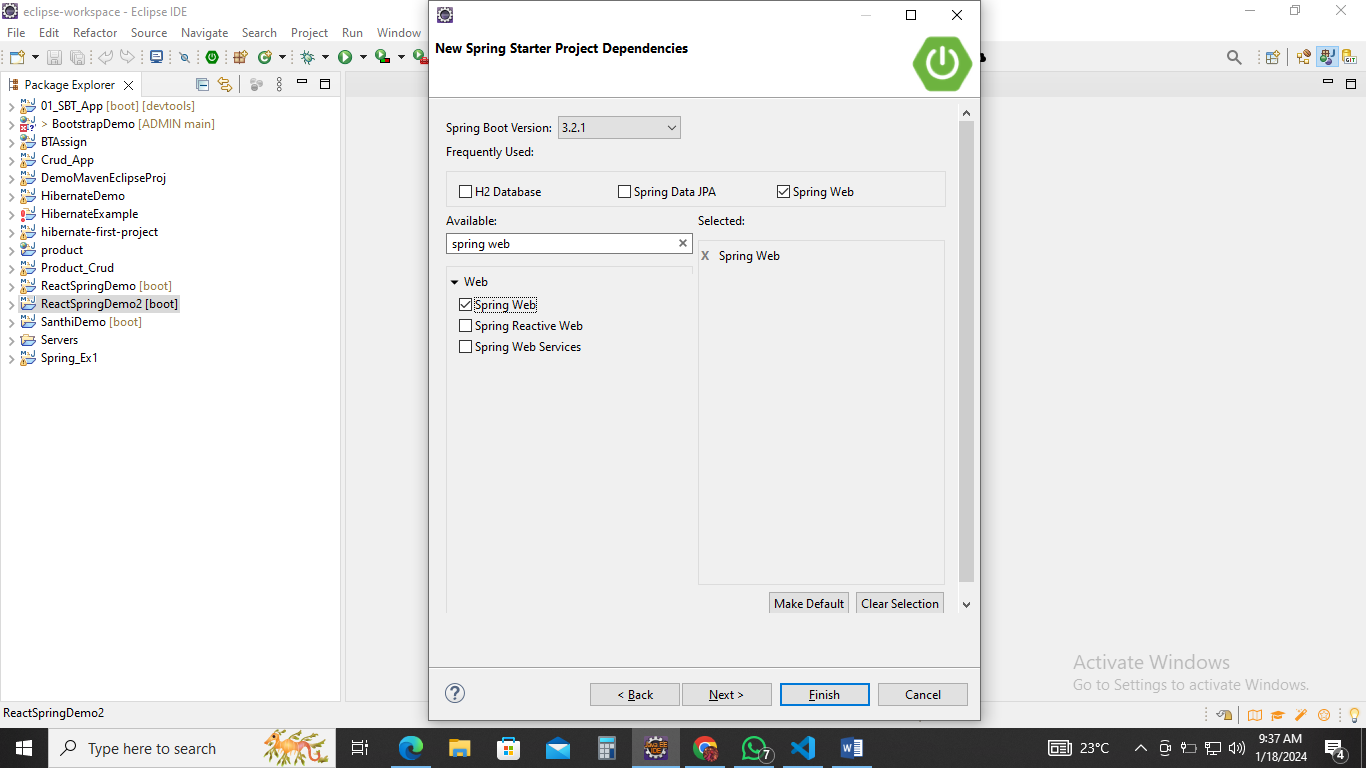
Angular Springboot with REST API

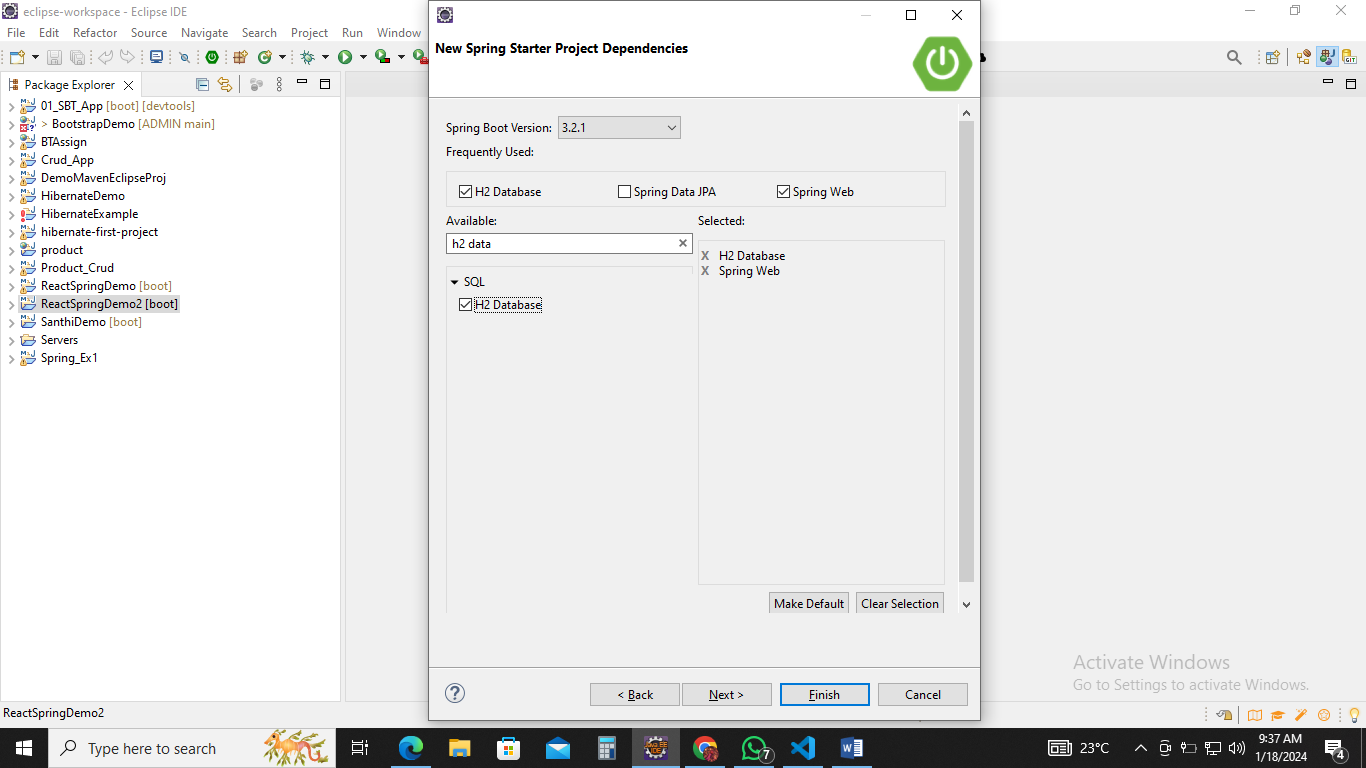
Eclipse

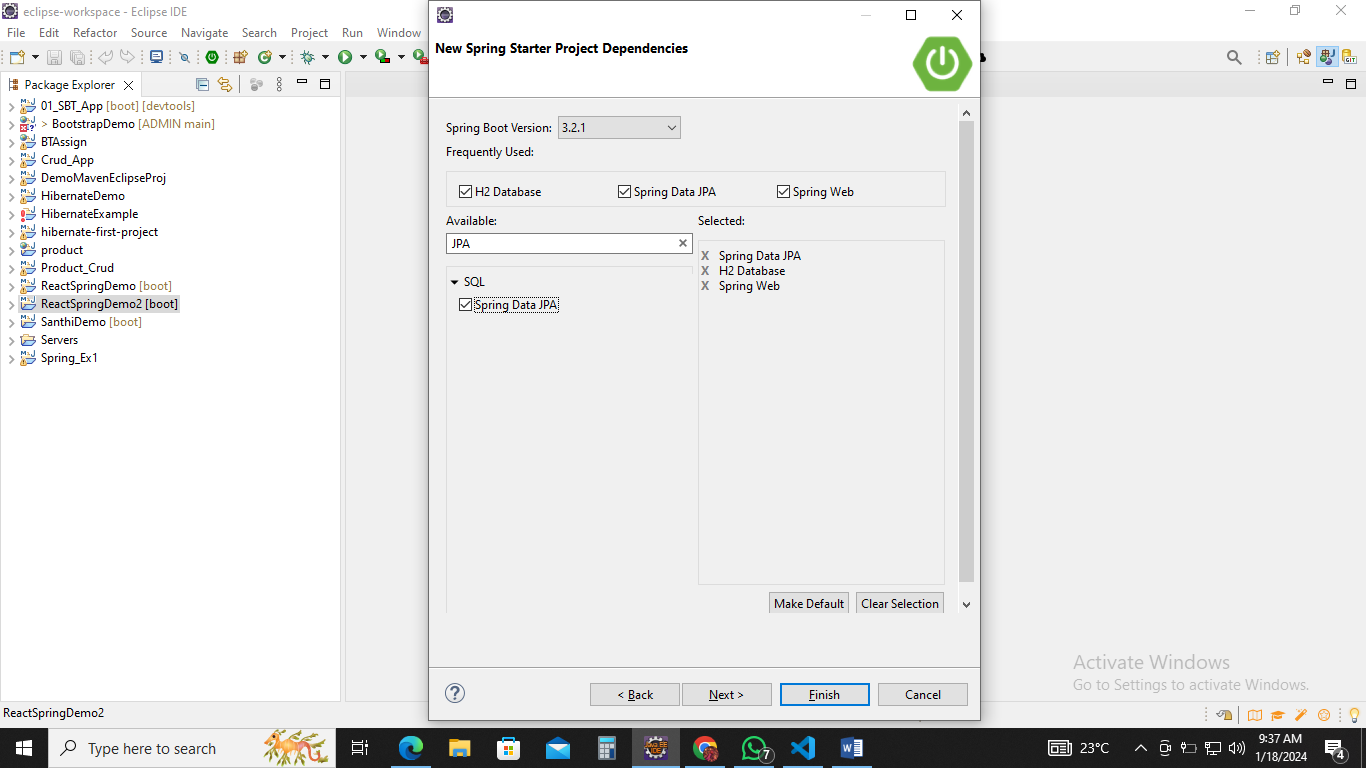












Click -> finish

Then go and explore the project folder

Src/main/java

-🡪net.javaguides.springboot

Create 3 packages

1. net.javaguides.springboot.model
2. net.javaguides.springboot.repository
3. net.javaguides.springboot.controller

net.javaguides.springboot.model

User.java

Type inside the main class

**public** **class** User {

**private** **long** id;

**private** String firstname;

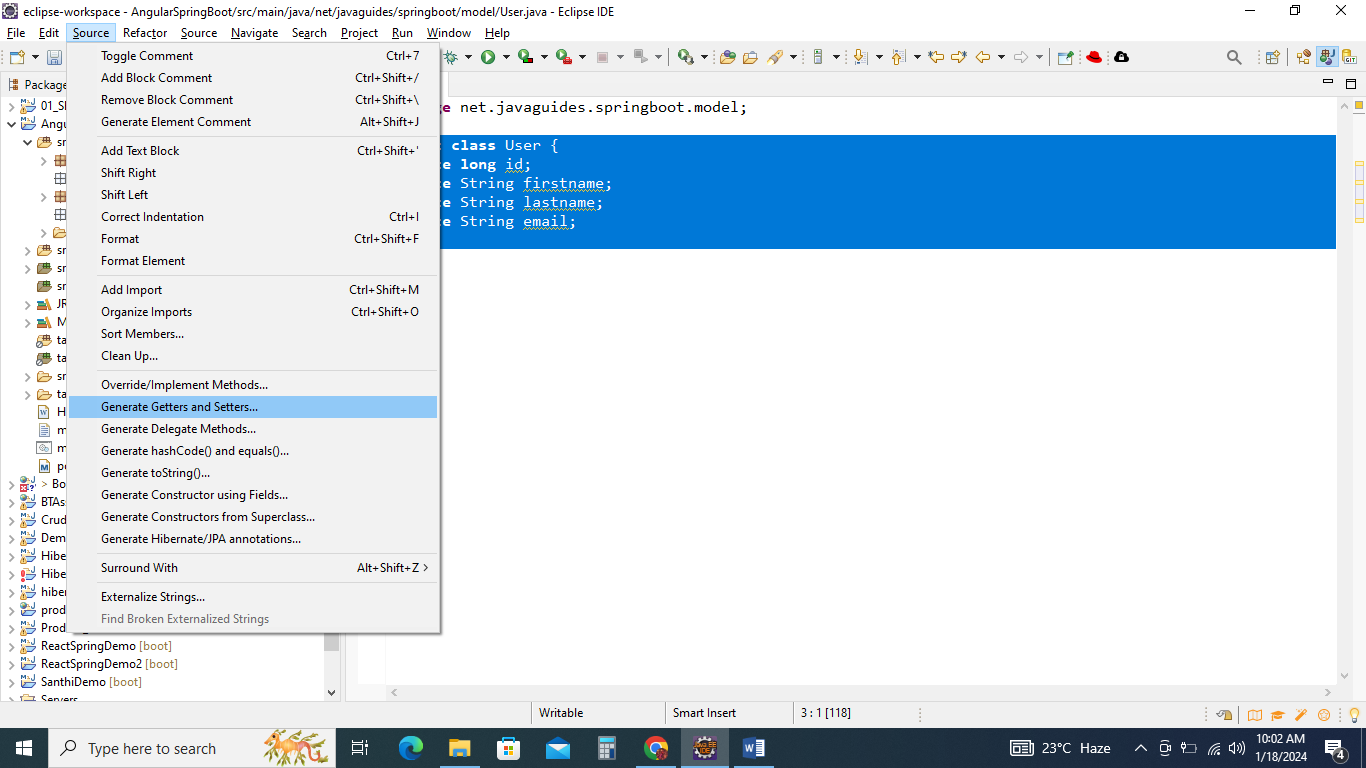
**private** String lastname;

**private** String email;

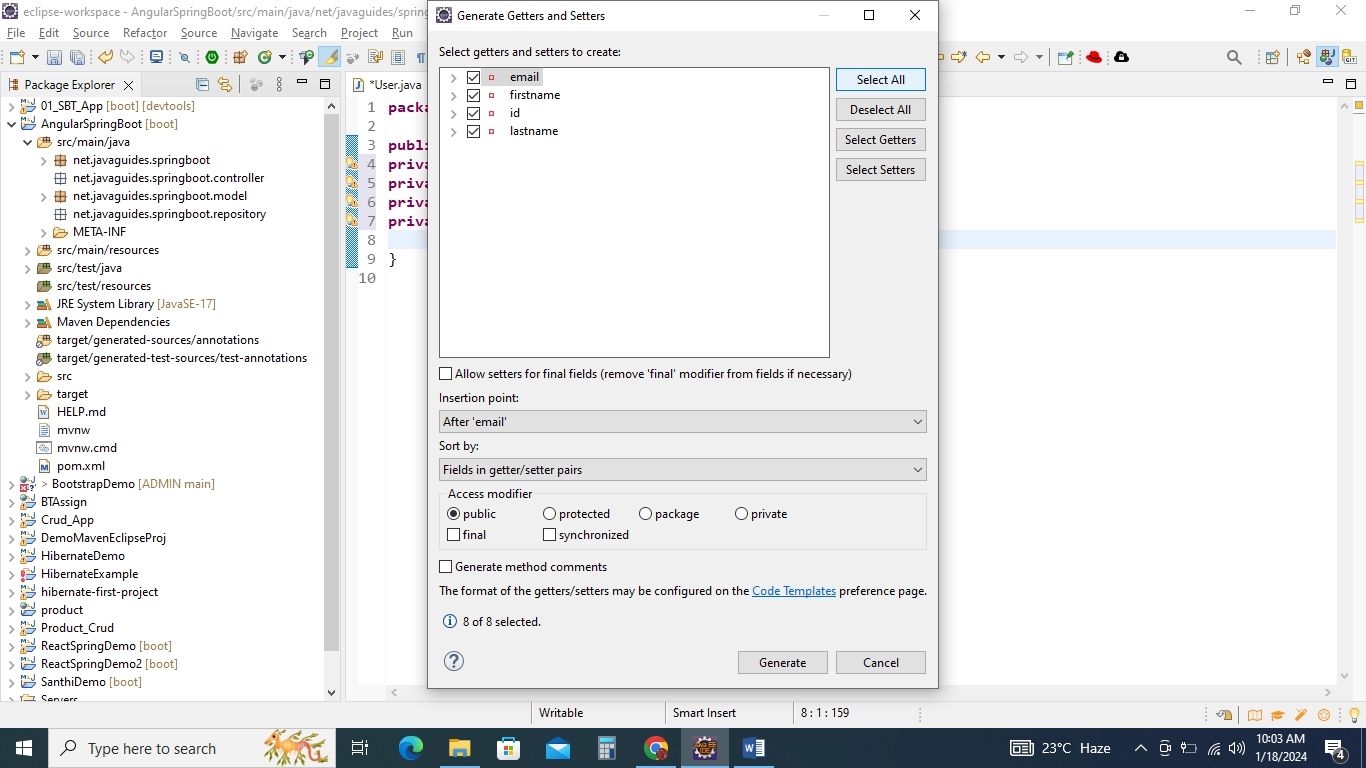
}

Click the inside the class and

In Source (Tab) ->Generate getters and setters



Select All



And click generate.

**public** **long** getId() {

**return** id;

}

**public** **void** setId(**long** id) {

**this**.id = id;

}

**public** String getFirstname() {

**return** firstname;

}

**public** **void** setFirstname(String firstname) {

**this**.firstname = firstname;

}

**public** String getLastname() {

**return** lastname;

}

**public** **void** setLastname(String lastname) {

**this**.lastname = lastname;

}

**public** String getEmail() {

**return** email;

}

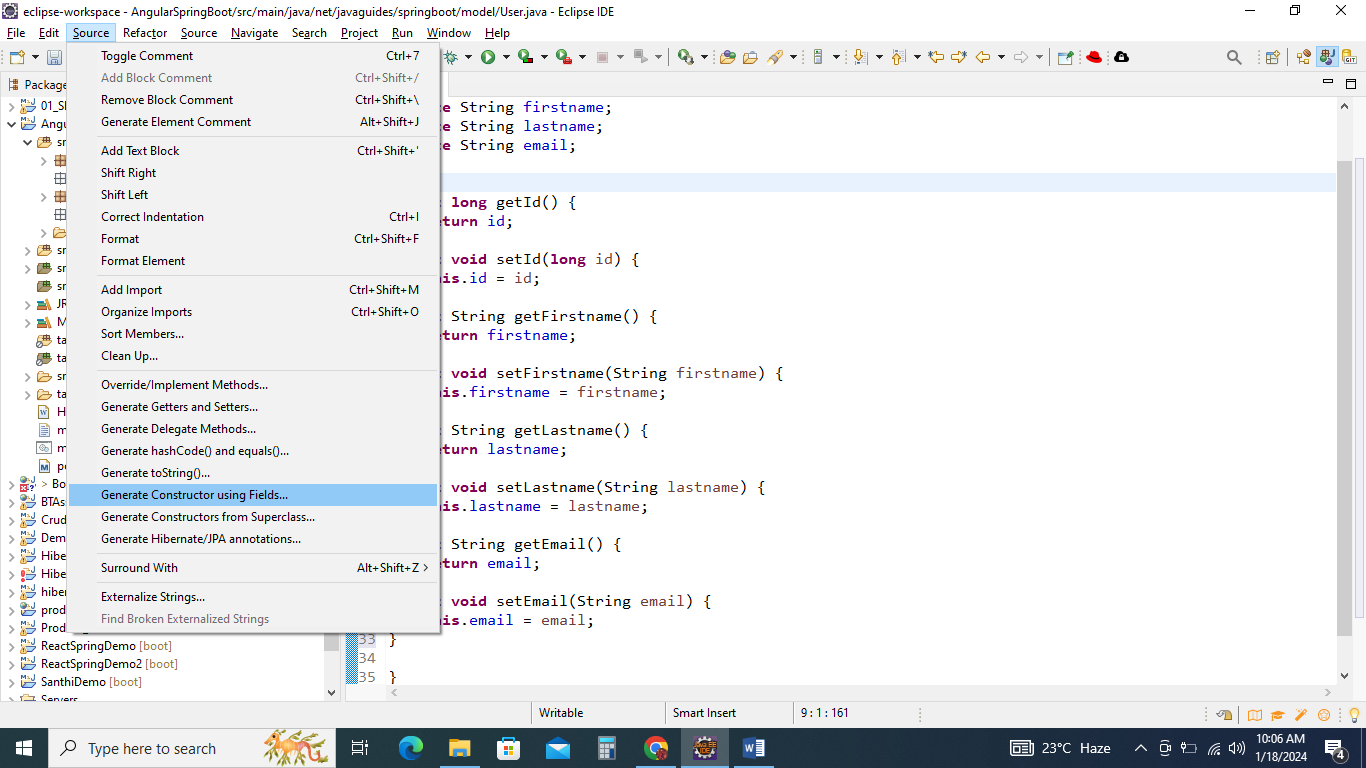
**public** **void** setEmail(String email) {

**this**.email = email;

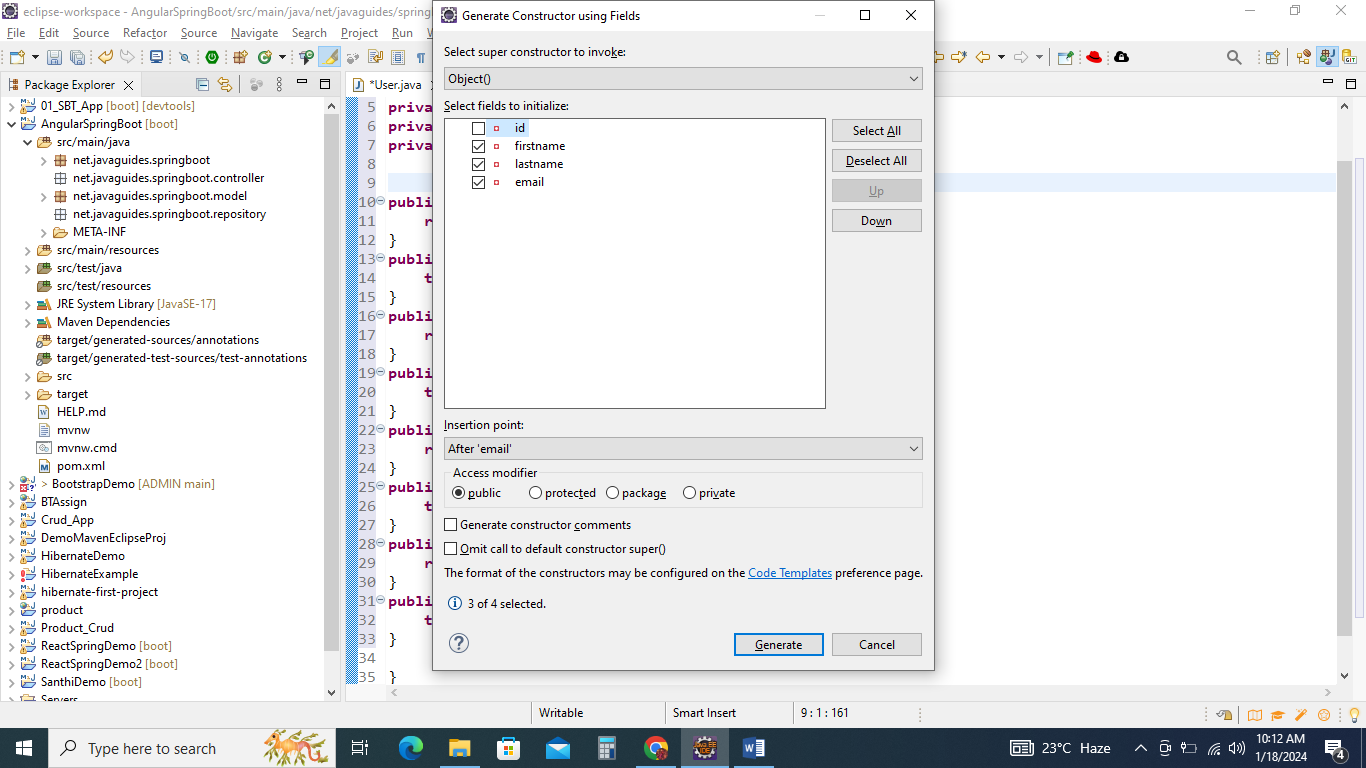
}

Before getters and setters

In Source -> Generate Constructor using Fields



Uncheck id and generate



**public** User(String firstname, String lastname, String email) {

**super**();

**this**.firstname = firstname;

**this**.lastname = lastname;

**this**.email = email;

}

Before that you have to update

**public** User() {

}

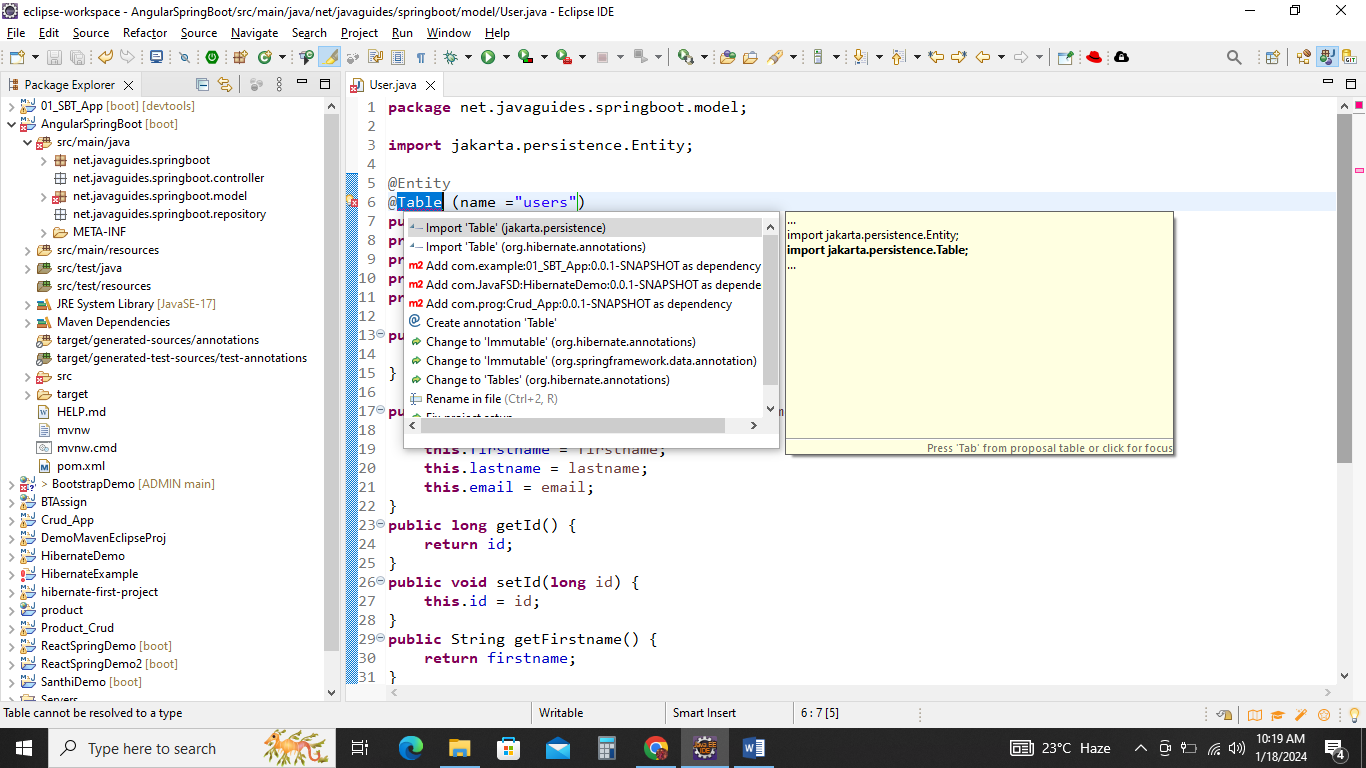
Create Annotations

@Entity

**import** jakarta.persistence.Entity;

@Table(name=”users”)

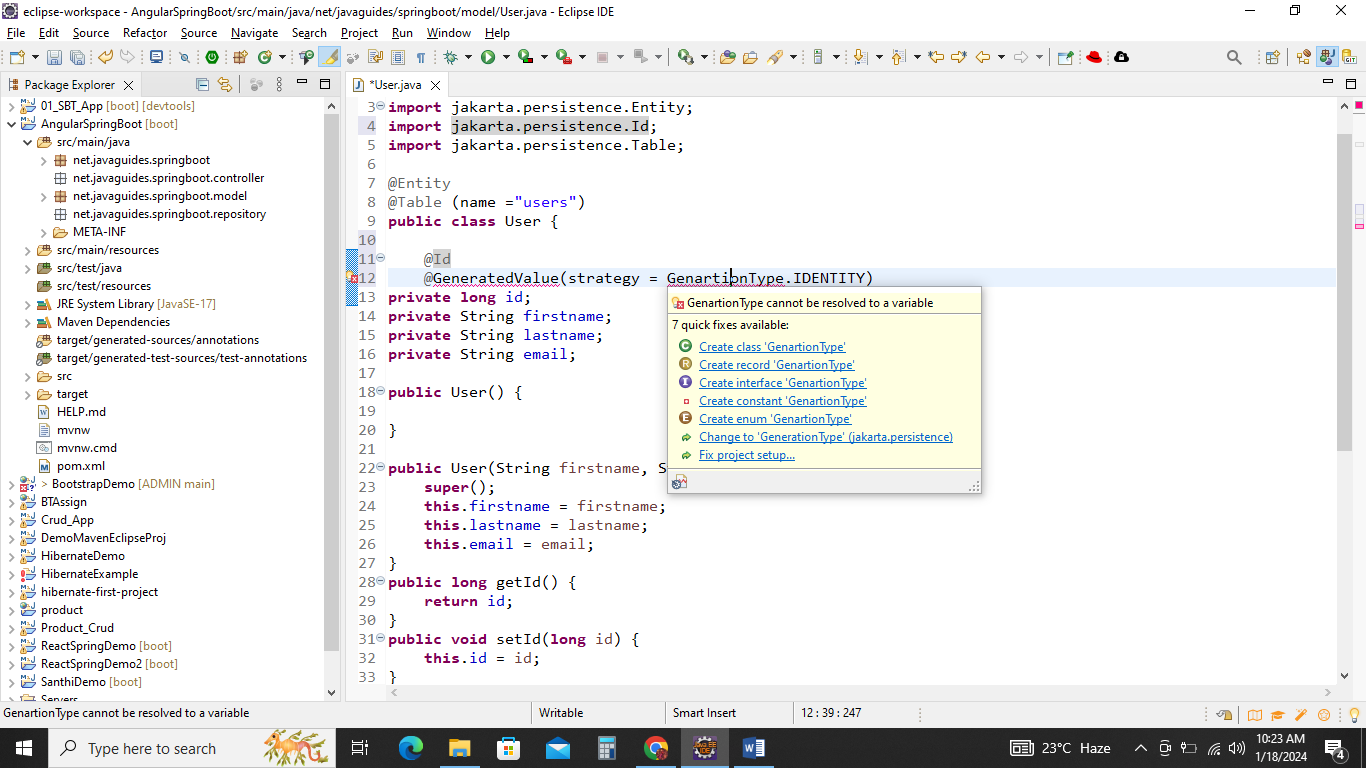
**import** jakarta.persistence.Table;



@Id

**import** jakarta.persistence.Id;

@GeratedValue(strategy=GenerationType.IDENTITY)



**import** jakarta.persistence.GeneratedValue;

**import** jakarta.persistence.GenerationType;

@Column(name="first\_name")

**import** jakarta.persistence.Column;

@Column(name="last\_name")

User.java

**package** net.javaguides.springboot.model;

**import** jakarta.persistence.Column;

**import** jakarta.persistence.Entity;

**import** jakarta.persistence.GeneratedValue;

**import** jakarta.persistence.GenerationType;

**import** jakarta.persistence.Id;

**import** jakarta.persistence.Table;

@Entity

@Table (name ="users")

**public** **class** User {

@Id

@GeneratedValue(strategy = GenerationType.***IDENTITY***)

**private** **long** id;

@Column(name="first\_name")

**private** String firstname;

@Column(name="last\_name")

**private** String lastname;

**private** String email;

**public** User() {

}

**public** User(String firstname, String lastname, String email) {

**super**();

**this**.firstname = firstname;

**this**.lastname = lastname;

**this**.email = email;

}

**public** **long** getId() {

**return** id;

}

**public** **void** setId(**long** id) {

**this**.id = id;

}

**public** String getFirstname() {

**return** firstname;

}

**public** **void** setFirstname(String firstname) {

**this**.firstname = firstname;

}

**public** String getLastname() {

**return** lastname;

}

**public** **void** setLastname(String lastname) {

**this**.lastname = lastname;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

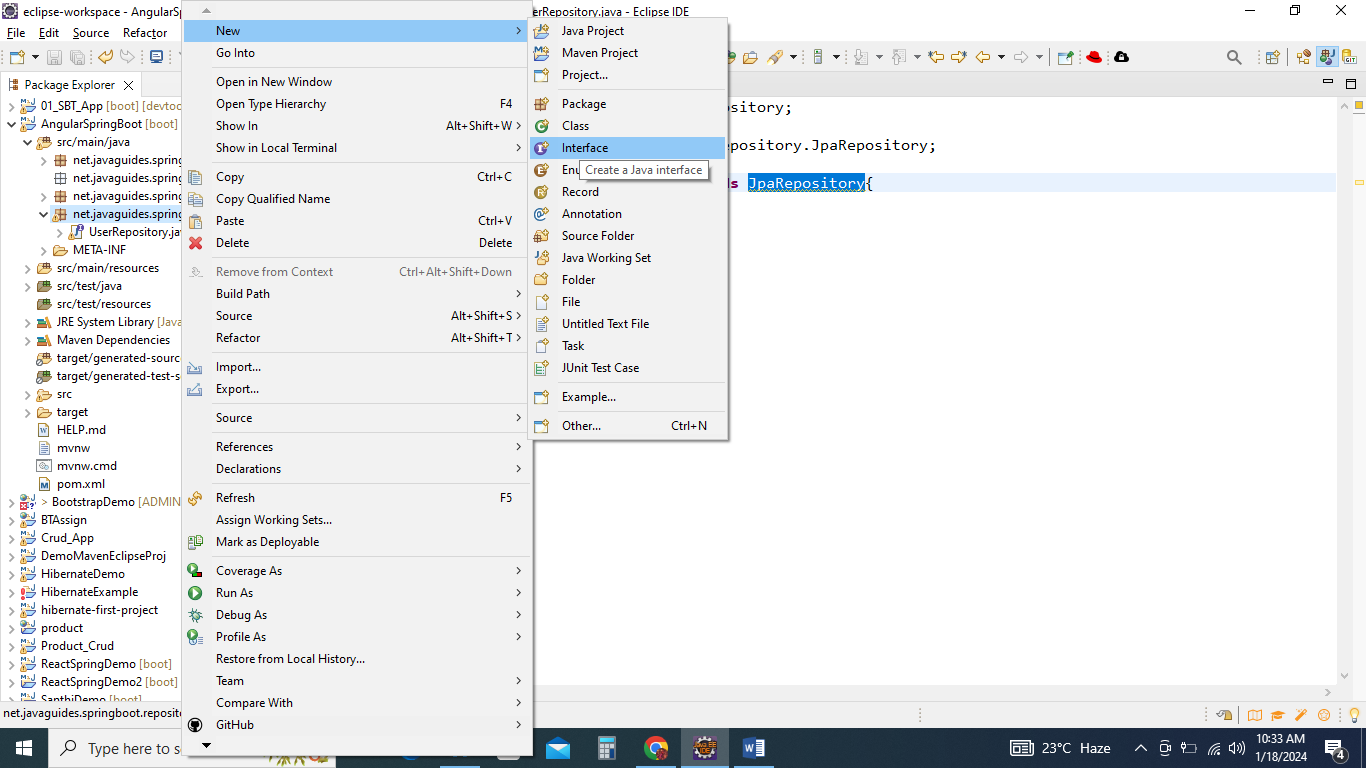
**this**.email = email;

}

}

Net.javaguides.springBoot.repository

Create Interface



Name->UserRepository

extends JpaRepository

**import** org.springframework.data.jpa.repository.JpaRepository;

**public** **interface** UserRepository **extends** JpaRepository{

}

Next Step

**package** net.javaguides.springboot.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

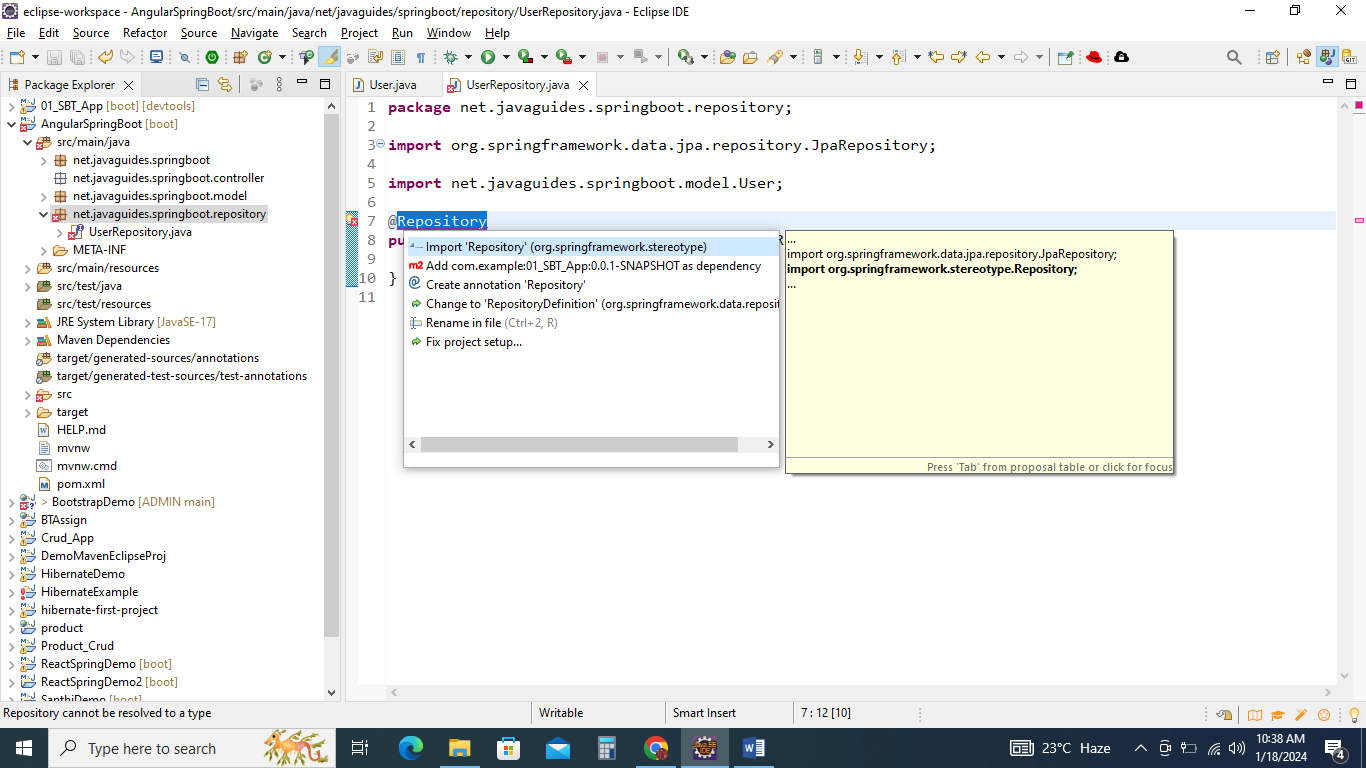
**import** net.javaguides.springboot.model.User;

**public** **interface** UserRepository **extends** JpaRepository<User, Long>{

}

Create Annotations

@Repository



UserRepository.java

**package** net.javaguides.springboot.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** net.javaguides.springboot.model.User;

@Repository

**public** **interface** UserRepository **extends** JpaRepository<User, Long>{

}

Net.javaguides.springboot.controllere

Create a new class

UserController.java

Create Annotations

@RestController

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

@RequestMapping("api/")

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

inside the class

**public** List<User> getUsers(){

}

**import** java.util.List;

**import** net.javaguides.springboot.model.User;

Before Public List

@GetMapping("users")

**import** org.springframework.web.bind.annotation.GetMapping;

Before @GetMapping

@Autowired

**import** org.springframework.beans.factory.annotation.Autowired;

After Autowired

**private** UserRepository userRepository;

**import** net.javaguides.springboot.repository.UserRepository;

Inside the public <List>

**return** **this**.userRepository.findAll();

UserController.java

**package** net.javaguides.springboot.controller;

**import** java.util.List;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.web.bind.annotation.GetMapping;

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

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

**import** net.javaguides.springboot.model.User;

**import** net.javaguides.springboot.repository.UserRepository;

@RestController

@RequestMapping("api/")

**public** **class** UserController {

@Autowired

**private** UserRepository userRepository;

@GetMapping("users")

**public** List<User> getUsers(){

**return** **this**.userRepository.findAll();

}

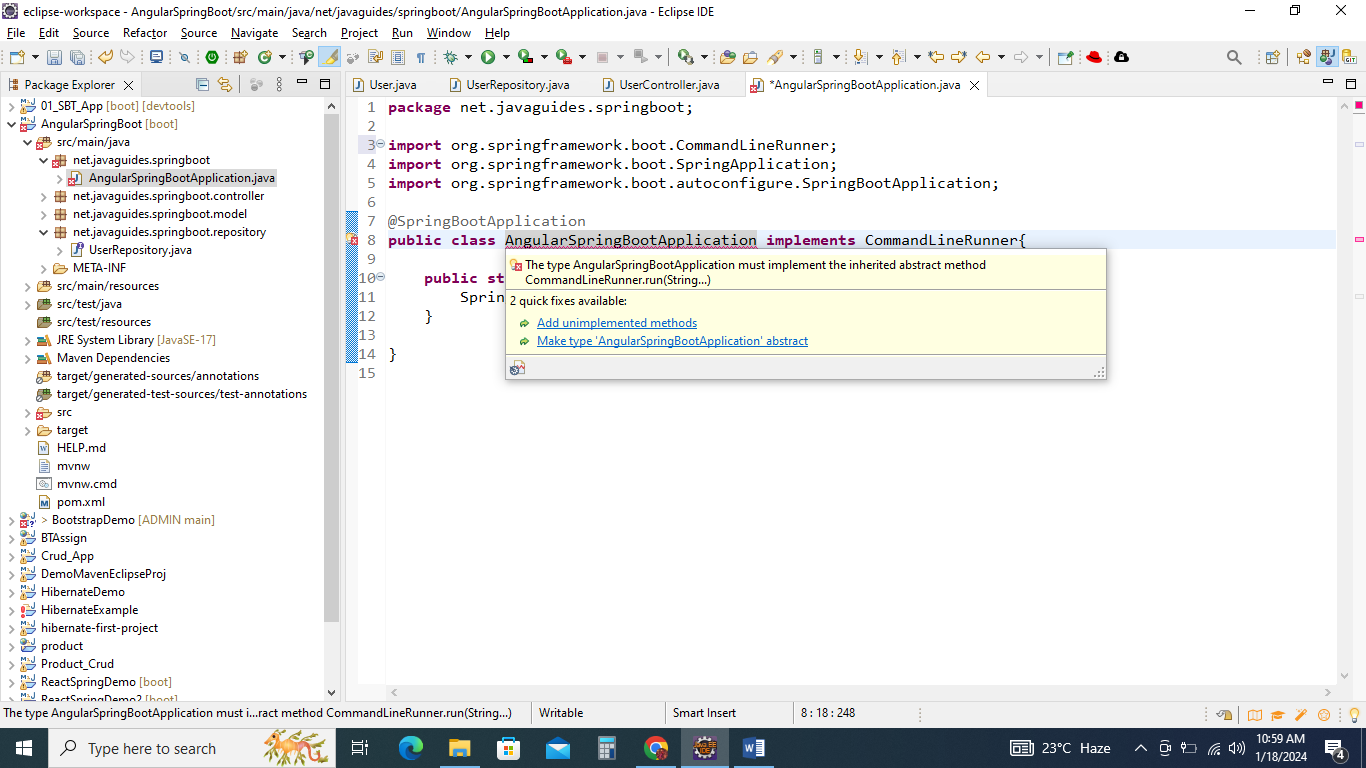
}

Inside net.javaguides.springBoot package class

AngularSpringBootApplication

**import** org.springframework.boot.CommandLineRunner;

**public** **class** AngularSpringBootApplication **implements** CommandLineRunner



Click ->Add unimplemented

Before @override

@Autowired

**private** UserRepository userRepository;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** net.javaguides.springboot.repository.UserRepository;

inside the public void run()

**this**.userRepository.save(**new** User("Santhi Priya","B","santhi@gmail.com"));

**import** net.javaguides.springboot.model.User;

AngularSpringBootApplication.java

**package** net.javaguides.springboot;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.boot.CommandLineRunner;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

**import** net.javaguides.springboot.model.User;

**import** net.javaguides.springboot.repository.UserRepository;

@SpringBootApplication

**public** **class** AngularSpringBootApplication **implements** CommandLineRunner{

**public** **static** **void** main(String[] args) {

SpringApplication.*run*(AngularSpringBootApplication.**class**, args);

}

@Autowired

**private** UserRepository userRepository;

@Override

**public** **void** run(String... args) **throws** Exception {

**this**.userRepository.save(**new** User("Santhi Priya","B","santhi@gmail.com"));

**this**.userRepository.save(**new** User("Leelarani","K","leela@gmail.com"));

**this**.userRepository.save(**new** User("Muthukrishna Vinayagam","B","bmkv@gmail.com"));

}

}

Run the project ->Run As Spring Boot App

Go to browser

[localhost:8080/api/users](http://localhost:8080/api/users)

