SpringBoot RestAPI Crud Example:

Using Spring initializer, create the Maven Project and then write the Following code.

**package** com.example.RestAPI;

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

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

**import** org.springframework.context.annotation.ComponentScan;

**import** org.springframework.data.jpa.repository.config.EnableJpaRepositories;

@ComponentScan({"main.controllers", "main.repositories"})

@EnableJpaRepositories("main.repositories")

@SpringBootApplication

**public** **class** RestApiApplication {

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

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

}

}

**package** com.example.RestAPI.controller;

**import** java.util.List;

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

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

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

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

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

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

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

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

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

**import** com.example.RestAPI.entity.User;

**import** com.example.RestAPI.service.UserService;

@RestController

@RequestMapping("/api")

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

@Autowired

**private** UserService userService;

@GetMapping("/users")

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

**return** userService.getUser();

}

@PostMapping("/save")

**public** User saveUser(@RequestBody User user) {

userService.save(user);

**return** user;

}

@GetMapping("/user/{id}")

**public** User getUserById(@PathVariable Integer id) {

User user = userService.findById(id);

**return** user;

}

@DeleteMapping("/delete/{id}")

**public** User deleteUser(@PathVariable Integer id) {

User user = userService.findById(id);

userService.delete(user);

**return** user;

}

@PutMapping("/update/{id}")

**public** User updateUser(@PathVariable Integer id, @RequestBody User userDetail) {

User user = userService.findById(id);

user.setUserName(userDetail.getUserName());

user.setMobileNo(userDetail.getMobileNo());

user.setEmailId(userDetail.getEmailId());

user.setCity(userDetail.getCity());

user.setPassword(userDetail.getPassword());

userService.save(user);

**return** user;

}

}

**package** com.example.RestAPI.dao;

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

**import** com.example.RestAPI.entity.User;

**public** **interface** UserRespository **extends** JpaRepository<User, Integer> {

}

**package** com.example.RestAPI.entity;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

//@Entity

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

//@Id

//#@GeneratedValue(strategy = GenerationType.IDENTITY)

**private** Integer id;

**private** String userName;

**private** String mobileNo;

**private** String emailId;

**private** String city;

**private** String password;

**public** User() {

}

**public** Integer getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getUserName() {

**return** userName;

}

**public** **void** setUserName(String userName) {

**this**.userName = userName;

}

**public** String getMobileNo() {

**return** mobileNo;

}

**public** **void** setMobileNo(String mobileNo) {

**this**.mobileNo = mobileNo;

}

**public** String getEmailId() {

**return** emailId;

}

**public** **void** setEmailId(String emailId) {

**this**.emailId = emailId;

}

**public** String getCity() {

**return** city;

}

**public** **void** setCity(String city) {

**this**.city = city;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

}

**package** com.example.RestAPI.service;

**import** java.util.List;

**import** com.example.RestAPI.entity.User;

**public** **interface** UserService {

**public** List<User> getUser();

**public** **void** save(User user);

**public** User findById(Integer id);

**public** **void** delete(User user);

}

**package** com.example.RestAPI.service;

**import** java.util.List;

**import** java.util.Optional;

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

**import** org.springframework.stereotype.Service;

**import** com.example.RestAPI.dao.UserRespository;

**import** com.example.RestAPI.entity.User;

@Service

**public** **class** UserServiceImpl **implements** UserService {

@Autowired

**private** UserRespository userRepo;

@Override

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

**return** userRepo.findAll();

}

@Override

**public** **void** save(User user) {

userRepo.save(user);

}

@Override

**public** User findById(Integer id) {

Optional<User> userResult = userRepo.findById(id);

User user = **null**;

**if** (userResult.isPresent()) {

user = userResult.get();

}

**return** user;

}

@Override

**public** **void** delete(User user) {

userRepo.delete(user);

}

}