**RestApiTestApplication**

**package** com.test.example;

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

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

@SpringBootApplication

**public** **class** RestApiTestApplication {

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

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

}

}

**Entity Class**

**package** com.test.example;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

@Entity

**public** **class** ProductEntity {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

@Column(name = "id", updatable = **false**, nullable = **false**)

**private** **int** id;

@Column

**private** String name;

@Column

**private** String description;

**public** ProductEntity() {

**super**();

}

**public** ProductEntity(**int** id, String name, String description) {

**super**();

**this**.id = id;

**this**.name = name;

**this**.description = description;

}

**public** **int** getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getDescription() {

**return** description;

}

**public** **void** setDescription(String description) {

**this**.description = description;

}

}

**Repository Class**

**package** com.test.example;

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

**public** **interface** ProductRepository **extends** JpaRepository<ProductEntity, Integer>{

}

**Service Class**

**package** com.test.example;

**import** java.util.List;

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

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

@Service

**public** **class** ProductService {

@Autowired

ProductRepository productRepository;

**public** List<ProductEntity> getAllProduct(){

**return** productRepository.findAll();

}

**public** ProductEntity getProduct(**int** id){

**return** productRepository.findById(id).get();

}

**public** **void** addProduct(ProductEntity pe){

productRepository.save(pe);

}

**public** **void** updateProduct(**int** id, ProductEntity pe){

**if**(productRepository.findById(id).isPresent()) {

ProductEntity oldProductEntity=productRepository.findById(id).get();

oldProductEntity.setName(pe.getName());

oldProductEntity.setDescription(pe.getDescription());

productRepository.save(oldProductEntity);

}

}

**public** **void** deleteProduct(**int** id){

productRepository.deleteById(id);

}

}

**Controller Class**

**package** com.test.example;

**import** java.util.List;

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

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

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

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

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

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

@RestController

@RequestMapping(path = "/webapi")

**public** **class** ProductController {

@Autowired

ProductService productService;

@RequestMapping("/allproduct")

**public** List<ProductEntity> getAllProduct(){

**return** productService.getAllProduct();

}

@RequestMapping("/product/{id}")

**public** ProductEntity getProduct(@PathVariable **int** id){

**return** productService.getProduct(id);

}

@RequestMapping(method=RequestMethod.POST, value="/product")

**public** **void** addProduct(@RequestBody ProductEntity pe ) {

productService.addProduct(pe);

}

@RequestMapping(method=RequestMethod.PUT, value="/product/{id}")

**public** **void** updateProduct(@PathVariable **int** id, @RequestBody ProductEntity pe ) {

productService.updateProduct(id, pe);

}

@RequestMapping(method=RequestMethod.DELETE, value="/product/{id}")

**public** **void** deleteProduct(@PathVariable **int** id) {

productService.deleteProduct(id);

}

}