Quiz portal

Answer.java

package com.simplilearn.demo;

import java.util.ArrayList;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Answers {

@Id

@Column(name="id")

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private String name;

private String answer1;

private String answer2;

private String answer3;

private int quizid

public Answers(int id, String name, String answer1, String answer2, String answer3, int quizid) {

super();

this.id = id;

this.name = name;

this.answer1 = answer1;

this.answer2 = answer2;

this.answer3 = answer3;

this.quizid = quizid;

}

public Answers() {

super();

// TODO Auto-generated constructor stub

}

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 getAnswer1() {

return answer1;

}

public void setAnswer1(String answer1) {

this.answer1 = answer1;

}

public String getAnswer2() {

return answer2;

}

public void setAnswer2(String answer2) {

this.answer2 = answer2;

}

public String getAnswer3() {

return answer3;

}

public void setAnswer3(String answer3) {

this.answer3 = answer3;

}

public int getQuizid() {

return quizid;

}

public void setQuizid(int quizid) {

this.quizid = quizid;

}

@Override

public String toString() {

return "Answers [id=" + id + ", name=" + name + ", answer1=" + answer1 + ", answer2=" + answer2 + ", answer3="

+ answer3 + ", quizid=" + quizid + "]";

}

}

AnswersRepo.java

package com.simplilearn.demo;

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

import org.springframework.stereotype.Repository;

@Repository

public interface AnswersRepo extends JpaRepository<Answers, Integer>{

}

Questions.java

package com.simplilearn.demo;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Questions {

@Id

@Column(name="id")

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private String que;

private String opA;

private String opB;

private String opC;

private String opD;

private String answer;

public Questions(int id, String que, String opA, String opB, String opC, String opD, String answer) {

super();

this.id = id;

this.que = que;

this.opA = opA;

this.opB = opB;

this.opC = opC;

this.opD = opD;

this.answer = answer;

}

public Questions() {

super();

// TODO Auto-generated constructor stub

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getQue() {

return que;

}

public void setQue(String que) {

this.que = que;

}

public String getOpA() {

return opA;

}

public void setOpA(String opA) {

this.opA = opA;

}

public String getOpB() {

return opB;

}

public void setOpB(String opB) {

this.opB = opB;

}

public String getOpC() {

return opC;

}

public void setOpC(String opC) {

this.opC = opC;

}

public String getOpD() {

return opD;

}

public void setOpD(String opD) {

this.opD = opD;

}

public String getAnswer() {

return answer;

}

public void setAnswer(String answer) {

this.answer = answer;

}

@Override

public String toString() {

return "Questions [id=" + id + ", que=" + que + ", opA=" + opA + ", opB=" + opB + ", opC=" + opC + ", opD="

+ opD + ", answer=" + answer + "]";

}

}

QuestionsController.java

package com.simplilearn.demo;

import java.util.List;

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

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

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.RequestBody;

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

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

@RestController

@RequestMapping("/admin")

public class QuestionsController {

@Autowired

private QuestionsService service;

///create new record

@PostMapping("/")

public ResponseEntity<Questions> addQuestion(@RequestBody Questions q){

Questions question= service.addQuestion(q);

if(question!=null)

return new ResponseEntity<Questions>(question,HttpStatus.CREATED);

else

return new ResponseEntity<Questions>(question, HttpStatus.INTERNAL\_SERVER\_ERROR);

}

@GetMapping("/")

public List<Questions> getAllQuestion(){

return service.getAllQuestion();

}

@DeleteMapping("/{id}")

public ResponseEntity<Object> deleteQuestion(@PathVariable int id ){

if(service.deleteQuestion(id))

return new ResponseEntity<Object>("Question Deleted", HttpStatus.OK);

else

return new ResponseEntity<Object>("No User Found",HttpStatus.NOT\_FOUND);

}

@PostMapping("/createquizz")

public ResponseEntity<Quizz> addQuiz(@RequestBody Quizz q){

Quizz qui= service.addQuiz(q);

if(qui!=null)

return new ResponseEntity<Quizz>(qui,HttpStatus.CREATED);

else

return new ResponseEntity<Quizz>(qui, HttpStatus.INTERNAL\_SERVER\_ERROR);

}

}

QuestionsRepo.java

package com.simplilearn.demo;

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

import org.springframework.stereotype.Repository;

@Repository

public interface QuestionsRepo extends JpaRepository<Questions, Integer> {

}

QuestionsService.java

package com.simplilearn.demo;

import java.util.List;

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

import org.springframework.stereotype.Service;

@Service

public class QuestionsService {

@Autowired

private QuestionsRepo repo;

@Autowired

private QuizRepo repo1;

//add method or CREATE RECORD

public Questions addQuestion(Questions q) {

// TODO Auto-generated method stub

return repo.save(q);

}

//get all questions

public List<Questions> getAllQuestion() {

// TODO Auto-generated method stub

return repo.findAll();

}

//delete Question

public boolean deleteQuestion(int id) {

if(repo.findById(id).isPresent())

{

repo.deleteById(id);

return true;

}

else

return false;

}

public Questions getQuestionById(int id) {

if(repo.findById(id).isPresent())

return repo.findById(id).get();

else

return null;

}

//creating quizz

public Quizz addQuiz(Quizz q) {

// TODO Auto-generated method stub

return repo1.save(q);

}

}

QuizAnswers.java

package com.simplilearn.demo;

import java.util.ArrayList;

import java.util.List;

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

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

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

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

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

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

@RestController

@RequestMapping("/answers")

public class QuizAnswers {

@Autowired

private AnswersRepo repo;

@Autowired

private QuizService service;

@Autowired

private QuestionsRepo repo1;

int qid;

int count=0;

ArrayList<String> finalresult = new ArrayList<String>();

@PostMapping("/")

public ResponseEntity<Object> addAnswers(@RequestBody Answers a) {

Answers answer = repo.save(a);

qid = answer.getQuizid();

if (answer != null)

{

ArrayList<String> uans = new ArrayList<String>();

uans.add(answer.getAnswer1());

uans.add(answer.getAnswer2());

uans.add(answer.getAnswer3());

List<Quizz> resp = service.findque(qid);

// created an arraylist to store which questions comes under the given quiz id

ArrayList<Integer> questionsid = new ArrayList<Integer>();

// saveing all the question id's in the arraylist

resp.forEach(e -> questionsid.add(e.getQuestionid()));

System.out.println(questionsid);

List<Questions> findall = repo1.findAllById(questionsid);

ArrayList<String> ans = new ArrayList<String>();

findall.forEach((e)->ans.add(e.getAnswer()));

for(int i=0; i<ans.size(); i++)

{

if(ans.get(i).equalsIgnoreCase(uans.get(i)))

{

count++;

}

}

finalresult.clear();

finalresult.add("Your Final result is: "+count+" Out of "+questionsid.size());

finalresult.add("Correct Ans: "+ans);

System.out.println(ans);

count=0;

}

return new ResponseEntity<Object>(finalresult, HttpStatus.CREATED);

}

}

QuizFetch.java

package com.simplilearn.demo;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

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

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

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

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

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

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

@RestController

@RequestMapping("/fetchQuiz")

public class QuizFetch {

@Autowired

private QuizService service;

@Autowired

private QuestionsRepo repo;

@GetMapping("/{quizid}")

public ResponseEntity<Object> findque(@PathVariable int quizid)

{

//getting all the questions data attached with the given quiz id

List<Quizz> resp= service.findque(quizid);

System.out.println(resp);

//created an arraylist to store which questions comes under the given quiz id

ArrayList<Integer> questionsid = new ArrayList<Integer>();

//saveing all the question id's in the arraylist

resp.forEach(e->questionsid.add(e.getQuestionid()));

System.out.println(questionsid);

//findinal all the questions with the given question ids

List<Questions> findall = repo.findAllById(questionsid);

ArrayList<String> question = new ArrayList<String>();

//storing all the questions and their options in the new arraylist

findall.forEach((e)->{

question.add("Que. "+e.getQue());

question.add("(a) "+e.getOpA());

question.add("(b) "+e.getOpB());

question.add("(c) "+e.getOpC());

question.add("(d) "+e.getOpD());

question.add("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

});

System.out.println(question);

if(questionsid.size()!=0)

return new ResponseEntity<Object>(question,HttpStatus.FOUND);

else

return new ResponseEntity<Object>("Quiz not Available",HttpStatus.NOT\_FOUND);

}

}

QuizPortalApplication.java

package com.simplilearn.demo;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.annotation.ComponentScan;

@SpringBootApplication

@ComponentScan("com.simplilearn.demo")

public class QuizPortalApplication {

public static void main(String[] args) {

SpringApplication.run(QuizPortalApplication.class, args);

}

}

QuizRepo.java

package com.simplilearn.demo;

import java.util.List;

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

import org.springframework.data.jpa.repository.Query;

import org.springframework.stereotype.Repository;

@Repository

public interface QuizRepo extends JpaRepository<Quizz, Integer> {

public List<Quizz> findByQuizid(int quizid);

}

QuizService.java

package com.simplilearn.demo;

import java.util.List;

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

import org.springframework.stereotype.Service;

@Service

public class QuizService<questionsid> {

@Autowired

private QuizRepo repo;

public List<Quizz> findque(int quizid) {

List<Quizz> questionss = repo.findByQuizid(quizid);

return questionss;

}

Quizz.java

package com.simplilearn.demo;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Quizz {

@Id

@Column(name="id")

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private int quizid;

private Integer questionid;

public Quizz(int id, int quizid, Integer questionid) {

super();

this.id = id;

this.quizid = quizid;

this.questionid = questionid;

}

public Quizz() {

super();

// TODO Auto-generated constructor stub

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public Integer getQuizid() {

return quizid;

}

public void setQuizid(int quizid) {

this.quizid = quizid;

}

public Integer getQuestionid() {

return questionid;

}

public void setQuestionid(Integer questionid) {

this.questionid = questionid;

}

@Override

public String toString() {

return "Quiz [id=" + id + ", quizid=" + quizid + ", questionid=" + questionid + "]";

}

}