**PRACTICAL SUBMISSION RECORD- A.Y. 2024-25**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class: FYMCA Div: A**  **Semester: II** | **Course Code:**  MCA01554  **Course Name: Java Programming Laboratory** | | **Batch:**  **F1** |
| **Name: Anurag Dalal** | | **Roll No: 51009** | |
| **CO No: CO515.5** | | **Assignment No: 16** | |

**Title : Create a CRUD application using Spring MVC**

**Code :**

**Student.java**

**package org.websparrow.model;**

**public class Student {**

**// Generate Getter and Setters...**

**private int id;**

**private String name;**

**private String email;**

**private String course;**

**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 getEmail() {**

**return email;**

**}**

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

**this.email = email;**

**}**

**public String getCourse() {**

**return course;**

**}**

**public void setCourse(String course) {**

**this.course = course;**

**}**

**}**

**CreateController**

**import org.springframework.web.servlet.ModelAndView;**

**import org.websparrow.dao.StudentDao;**

**import org.websparrow.model.Student;**

**@Controller**

**public class CreateController {**

**@Autowired  
private StudentDao studentDao;**

**@RequestMapping(value = &quot;/create&quot;, method = RequestMethod.POST)**

**public ModelAndView createStudent(@RequestParam(&quot;name&quot;) String name, @RequestParam(&quot;email&quot;)**

**String email,**

**@RequestParam(&quot;course&quot;) String course, ModelAndView mv) {**

**Student student = new Student();**

**student.setName(name);**

**student.setEmail(email);**

**student.setCourse(course);**

**int counter = studentDao.create(student);**

**if (counter &gt; 0) {**

**mv.addObject(&quot;msg&quot;, &quot;Student registration successful.&quot;);**

**} else {**

**mv.addObject(&quot;msg&quot;, &quot;Error- check the console log.&quot;);**

**}**

**mv.setViewName(&quot;create&quot;);**

**return mv;**

**}**

**}**

**@Controller**

**public class DeleteController**

**@Autowired**

**private StudentDao studentDao;**

**@RequestMapping(value = &quot;/delete/{studentId}&quot;)**

**public ModelAndView deleteStudentById(ModelAndView mv, @PathVariable(&quot;studentId&quot;) int studentId)**

**throws IOException**

**int counter = studentDao.delete(studentId);**

**if (counter &gt; 0) {**

**mv.addObject(&quot;msg&quot;, &quot;Student records deleted against student id: &quot; + studentId);**

**} else {**

**mv.addObject(&quot;msg&quot;, &quot;Error- check the console log.&quot;);**

**}**

**mv.setViewName(&quot;delete&quot;);**

**return mv;**

**}**

**}**

**package org.websparrow.controller**

**@Controller**

**public class ReadController {**

**@Autowired**

**private StudentDao studentDao;**

**@RequestMapping(value = &quot;/read&quot;)**

**public ModelAndView readStudent(ModelAndView model) throws IOException {**

**List&lt;Student&gt; listStudent = studentDao.read();**

**model.addObject(&quot;listStudent&quot;, listStudent);**

**model.setViewName(&quot;read&quot;);**

**return model;**

**}**

**}**

**package org.websparrow.controller;**

**@Controller**

**public class UpdateController {**

**@Autowired**

**private StudentDao studentDao;**

**@RequestMapping(value = &quot;/update/{studentId}&quot;)**

**public ModelAndView findStudentById(ModelAndView model, @PathVariable(&quot;studentId&quot;) int studentId)**

**throws IOException {**

**List&lt;Student&gt; listStudent = studentDao.findStudentById(studentId);**

**model.addObject(&quot;listStudent&quot;, listStudent);**

**model.setViewName(&quot;update&quot;);**

**return model;**

**}**

**@RequestMapping(value = &quot;/update&quot;, method = RequestMethod.POST)**

**public ModelAndView updateStudent(@RequestParam(&quot;id&quot;) int id, @RequestParam(&quot;name&quot;) String**

**name,**

**@RequestParam(&quot;email&quot;) String email, @RequestParam(&quot;course&quot;) String course,**

**ModelAndView mv) {**

**Student student = new Student();**

**student.setId(id);**

**student.setName(name);**

**student.setEmail(email);**

**student.setCourse(course);**

**int counter = studentDao.update(student);**

**if (counter &gt; 0) {**

**mv.addObject(&quot;msg&quot;, &quot;Student records updated against student id: &quot; + student.getId());**

**} else { mv.addObject(&quot;msg&quot;, &quot;Error- check the console log.&quot;);}**

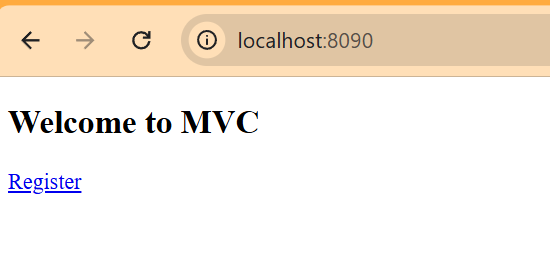
**mv.setViewName(&quot;update&quot;);**

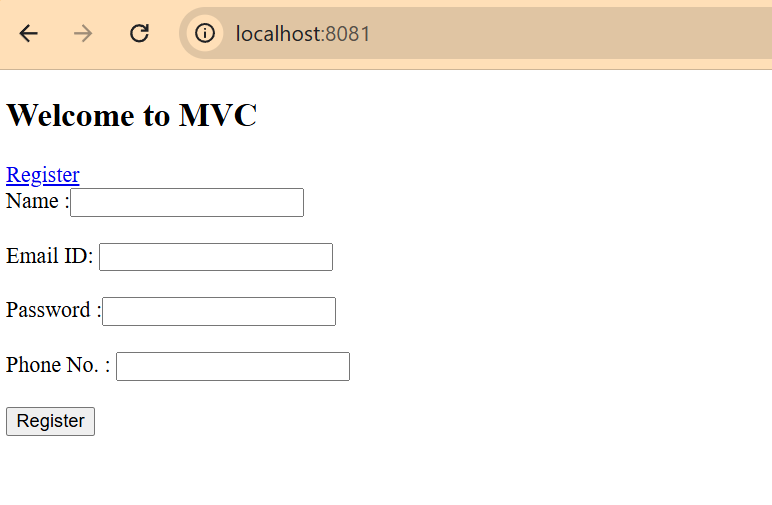
**return mv;**

**}**

**}**

**Output :**

****

****

****

****

****