

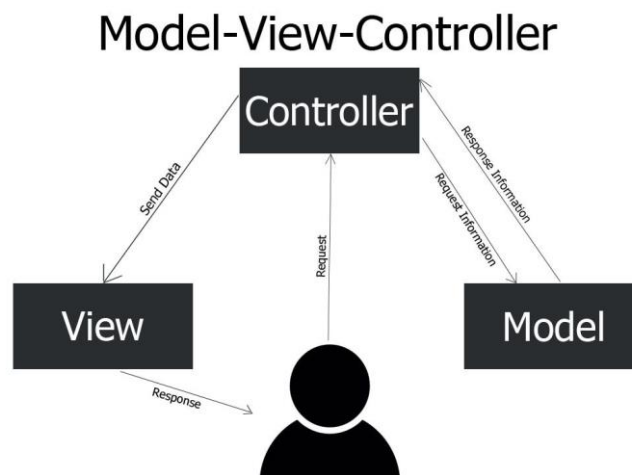
Object Oriented Analysis and Design using Java (UE20CS352)

Self Learning Hands-on Assignment: MVC Framework

Name	Ananya Nigam
SRN	PES1UG20CS044
Section	A
Date	12.03.23

o MVC Architecture pattern

The MVC pattern is designed to separate an application's data model, user interface, and control logic into distinct modules that can be easily maintained and modified.



The Model-View-Controller (MVC) is an architectural design that dissects an application into three primary logical constituents:

1. Model

- It has the data and business logic of an application that the user requires to work on.
- This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data.

2. View

- This is used for all the User Interface logic of the application.

3. Controller

- Controllers act as an interface between Model and View components.
- Handles user input and managing the flow of data between the two.
- It processes all the business logic and incoming requests from user, manipulate data using the Model component and interact with the Views to render the final output which is displayed to the user.

Each of these constituents is tailored to manage specific development aspects of an application. MVC is a widely adopted web development framework that conforms to industry standards and facilitates the development of scalable and expandable projects.

o Advantages of MVC pattern

1. **Separation of concerns:** The MVC pattern separates an application into three distinct components, each with its own responsibilities:
 - the model (for handling data and business logic)
 - the view (for handling user interface)
 - the controller (for handling user input and controlling the flow of data between the model and the view)

This separation allows each component to handle specific concerns, making the code **more readable, organized, maintainable, and flexible**.

2. **Increases Code Reusability:** As each component of the MVC pattern is independent and can be reused in other parts of the application or in other applications entirely so this separation of concerns also promotes code reusability.
For example, a view or controller developed for one application can be reused in another application without modification, as long as the new application uses the same data model.
3. **Testability:** As the model, view, and controller are separate modules in MVC pattern, each of these can be tested independently of the others. This allows developers to write more comprehensive tests and identify bugs more easily.

4. **Scalability:** As each component can be optimized separately to handle high loads or complex use cases, the MVC pattern allows an application to be more scalable.
5. **Adaptability:** It allows developers to adapt an application to changing requirements more easily. As each component is independent, changes to one component do not affect the others, making it easier to modify or extend an application without causing unintended consequences.
6. **Maintenance:** When changes need to be made, developers can modify individual components without affecting the entire application.
7. **Easy development:** The MVC pattern provides a clear structure and organization for an application, making it easier for developers to work on the codebase. By separating concerns and promoting modularity, the MVC pattern makes it easier to develop new features or fix bugs in the code.
8. **Improving Code readability:** The MVC pattern promotes a cleaner and more organized code structure, making the codebase more readable and easier to understand.
9. **Security:** By separating concerns and promoting modularity, it is easier to control access to sensitive data and functionality for the developers.

Overall, the MVC pattern is a powerful tool for designing and building scalable, maintainable, and flexible software applications.

o Features of MVC Framework chosen

Spring Boot is a popular Java-based framework that provides a set of tools and libraries to simplify the development of web applications. Some of the key features of Spring Boot include:

1. **Convention over Configuration:** Spring Boot follows a "convention over configuration" approach, which means that it provides sensible defaults and auto-configuration options for common application components, reducing the amount of boilerplate code required to get started.

2. **Embedded server:** Spring Boot includes an embedded web server (Tomcat, Jetty or Undertow), so you can run your application as a standalone executable JAR file without the need for a separate web server installation.
3. **Dependency management:** Spring Boot uses a "starter" approach to dependency management, which means that it includes pre-configured dependencies for common application features, such as databases, security, and testing frameworks. This simplifies the process of adding dependencies and reduces the likelihood of version conflicts.
4. **Spring Boot CLI:** Spring Boot includes a Command Line Interface (CLI) that allows you to quickly create and run Spring Boot applications from the command line
5. **Simplified configuration:** Spring Boot provides a simpler configuration model than traditional Spring applications, using properties files or YAML files to configure application settings. This makes it easier to manage and deploy your application across different environments.
6. **Integration with other Spring projects:** Spring Boot integrates seamlessly with other Spring projects, such as Spring Data, Spring Security, etc providing a comprehensive ecosystem of tools for building robust, scalable applications.

Problem Definition:

The task is to develop a Quiz web application using the Spring Boot framework and Java, which will follow the Model-View-Controller (MVC) design pattern. The application will use a MySQL database to store and retrieve data.

The application should allow users to add, delete, and update questions, which are stored in a MySQL database.

Scenario Description:

The quiz web application will allow users to take quiz to test their knowledge. The quizzes will consist of multiple-choice questions.

When the user clicks on **Add Question** button in (<http://localhost:8080/>) i.e. the main home page, he/she is redirected to <http://localhost:8080/shownewQuestionForm> where a form is displayed with various fields needed to add a new question. The user fills all these fields and clicks on Save Question after which he/she is redirected to home page again which now also contains the newly added question along with all the previous questions.

When the user clicks on **Modify Question** for a specific question on the home page then he is redirected to <http://localhost:8080/showFormForUpdate/1> where “1” represents the question number for the corresponding question. This page has a form filled with the original question that user wants to modify. After making changes in these the user clicks on Update Question button and is redirected to the home page and the update question is visible in place of the original question.

When the user clicks on **Delete Question** in row of a question then, that particular question is deleted from webpage and the remaining questions are displayed on the webpage. The application will have three main components:

Model: This component will be responsible for managing the data and the logic associated with it. The model will interact with the MySQL database to store and retrieve quiz questions and answers.

View: This component will be responsible for displaying the user interface. It will present the quiz questions and answers to the user.

Controller: This component will act as an intermediary between the model and the view. It will handle user's action and redirect the user's request to the model accordingly. It will also update the view based on changes in the model.



Complete screenshot of code written by you

quiz.java

```
demo > src > main > java > com > example > SpringBootWebApplication > quiz.java > quiz > setquestion(String)
1 package com.example.SpringBootWebApplication;
2 import jakarta.persistence.Column;
3 import jakarta.persistence.Entity;
4 import jakarta.persistence.Id;
5 import jakarta.persistence.Table;
6 @Entity // This tells Hibernate to make a table out of this class
7 @Table(name = "mcqs")
8 public class quiz{
9     @Id
10    @Column(name="id")
11    public Integer id;
12    @Column(name="question")
13    public String question;
14    @Column(name="option_a")
15    public String option_a;
16    @Column(name="option_b")
17    public String option_b;
18    @Column(name="option_c")
19    public String option_c;
20    @Column(name="option_d")
21    public String option_d;
22    @Column(name="correct_answer")
23    public String correct_answer;
24
25    public Integer getId() {
26        return id;
27    }
28
29    public void setId(Integer id) {
30        this.id = id;
31    }
32
33    public String getquestion() {
34        return question;
35    }
36
```

```
demo > src > main > java > com > example > SpringBootWebApplication > quiz.java > quiz > setquestion(String)
37    public void setquestion(String question) {
38        this.question = question;
39    }
40
41
42    public String getoption_a() {
43        return option_a;
44    }
45
46    public void setoption_a(String option_a) {
47        this.option_a = option_a;
48    }
49
50    public String getoption_b() {
51        return option_b;
52    }
53
54    public void setoption_b(String option_b) {
55        this.option_b = option_b;
56    }
57    public String getoption_c() {
58        return option_c;
59    }
60    public void setoption_c(String option_c) {
61        this.option_c = option_c;
62    }
63
64    public String getoption_d() {
65        return option_d;
66    }
67
68    public void setoption_d(String option_d) {
69        this.option_d = option_d;
70    }
71
72    public String getcorrect_answer() {
73        return correct_answer;
74    }
75
```

```

74     }
75
76     public void setcorrect_answer(String correct_answer) {
77         this.correct_answer = correct_answer;
78     }
79
80
81 }

```

quizcontroller.java

```

quizcontroller.java X
demo > src > main > java > com > example > SpringBootApplication > quizcontroller.java > quizcontroller > saveQuest
1  package com.example.SpringBootApplication;
2  import org.springframework.beans.factory.annotation.Autowired;
3  import org.springframework.stereotype.Controller;
4  import org.springframework.ui.Model;
5  import org.springframework.web.bind.annotation.GetMapping;
6  import org.springframework.web.bind.annotation.ModelAttribute;
7  import org.springframework.web.bind.annotation.PathVariable;
8  import org.springframework.web.bind.annotation.PostMapping;
9
10 @Controller
11 public class quizcontroller {
12     @Autowired
13     public QuizService quizservice;
14
15     //display List of questions
16
17     @GetMapping("/")
18     public String viewHomePage(Model model){
19         model.addAttribute(attributeName:"listQuestions",quizservice.getAllQuestions());
20         return "index";
21     }
22     @GetMapping("/shownewQuestionForm")
23     public String shownewQuestionForm(Model model){
24
25         quiz Quiz=new quiz();
26         model.addAttribute(attributeName:"Quiz",Quiz);
27         return "new_question";
28     }
29
30     @PostMapping("/saveQuestion")
31     public String saveQuestion(@ModelAttribute("Quiz") quiz Quiz){
32         quizservice.saveQuestion(Quiz);
33         return "redirect:/";
34     }
35 }
36

```

```

37
38     @GetMapping("/showFormForUpdate/{id}")
39     public String showFormForUpdate(@PathVariable("id")int id,Model model){
40         //get question from service
41         quiz Quiz=quizservice.getQuestionById(id);
42         //set question as a model attr to pre-populate the form
43         model.addAttribute(attributeName:"Quiz", Quiz);
44         return "update_question";
45     }
46
47
48     @GetMapping("/deleteQuestion/{id}")
49     public String deleteQuestion(@PathVariable (value = "id") int id){
50         //call delete question method
51
52         this.quizservice.deleteQuestionById(id);
53         return "redirect:/";
54     }
55
56
57 }
58
59

```

QuizRepository.java

```

Go Run Terminal Help QuizRepository.java - Untitled (Workspace) - Visual Studio Code
QuizRepository.java X
demo > src > main > java > com > example > SpringBootWebApplication > QuizRepository.java > ...
1 package com.example.SpringBootWebApplication;
2 import org.springframework.data.repository.CrudRepository;
3
4 // import java.util.List;
5
6 import org.springframework.stereotype.Repository;
7
8 @Repository
9 public interface QuizRepository extends CrudRepository<quiz, Integer> {
10
11 }
12
13
14

```

QuizService.java

```

Go Run Terminal Help QuizService.java - Untitled (Workspace) - Visual Studio Code
QuizService.java X
demo > src > main > java > com > example > SpringBootWebApplication > QuizService.java > QuizService
1 package com.example.SpringBootWebApplication;
2 // import java.util.*;
3
4 import java.util.List;
5
6 public interface QuizService {
7
8     List<quiz> getAllQuestions();
9     void saveQuestion(quiz Quiz);
10    quiz getQuestionById(int id);
11    void deleteQuestionById(int id);
12
13 }
14

```


quizserviceImpl.java

```
demo > src > main > java > com > example > SpringBootWebApplication > quizserviceImpl.java > ...
1  package com.example.SpringBootWebApplication;
2  import java.util.List;
3  import java.util.Optional;
4  import org.springframework.beans.factory.annotation.Autowired;
5  import org.springframework.stereotype.Service;
6
7  @Service
8  public class quizserviceImpl implements QuizService{
9      @Autowired
10
11      public QuizRepository quizRepository;
12
13      @Override
14      public List<quiz> getAllQuestions(){
15
16          return (List<quiz>) quizRepository.findAll();
17      }
18
19      @Override
20      public void saveQuestion(quiz Quiz){
21          this.quizRepository.save(Quiz);
22      }
23
24      @Override
25      public quiz getQuestionById(int id){
26          Optional<quiz> optional=quizRepository.findById(id);
27          quiz Quiz=null;
28          if(optional.isPresent()){
29              Quiz=optional.get();
30          }
31          else{
32              throw new RuntimeException("Question Not Found for Question Number : " + id);
33          }
34
35          return Quiz;
36      }
37
38
39      @Override
40      public void deleteQuestionById(int id){
41          this.quizRepository.deleteById(id);
42      }
43  }
44
45
46
```

SpringBootThymeleafWebApplication.java

```
SpringBootThymeleafWebApplication.java X
demo > src > main > java > com > example > SpringBootApplication > SpringBootThymeleafWebApplication.java > ...
1 package com.example.SpringBootApplication;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5
6 @SpringBootApplication
7 public class SpringBootThymeleafWebApplication{
8
9     public static void main(String[] args) {
10         SpringApplication.run(SpringBootThymeleafWebApplication.class, args);
11     }
12 }
13
14
15
16
```

index.html

```
index.html X
demo > src > main > resources > templates > index.html > html > body > div.container.my-2 > a.btn.btn-primary.btn-sm.mb-3
1 <html lang="en" xmlns:th="http://www.thymeleaf.org">
2 <head>
3 <title>Quiz Web Application</title>
4
5 <link
6 rel="stylesheet"
7 href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/css/bootstrap.min.css"
8 integrity="sha384-B0vP5xmATw1+K9KRQjQERjvTumQW0nPEzvF6L/Z6nronJ3oUOFUfPcJEUQouq2+1"
9 crossorigin="anonymous"
10 />
11 </head>
12
13 <body>
14 <style>
15 body {
16 background-color: #84fdb3;
17 }
18 </style>
19
20 <div class="container my-2">
21 <h1 align="center" style="color: rgb(0, 0, 0)">
22 🤖 Quizzz Yourself ! 🤖
23 </h1>
24 <h3 style="color: rgb(233, 67, 92)">Multiple Choice Questions</h3>
25 <a th:href="@{/shownewQuestionForm}" class="btn btn-primary btn-sm mb-3"
26 >Add Question</a>
27 >
28
29 <hr />
30
31 <form th:action="@{/submit}" method="post">
32 <table
33 class="table table-bordered table-striped"
34 style="background-color: rgb(240, 240, 195)"
35 >
36 <thead>
37 <tr>
```

index.html X

demo > src > main > resources > templates > index.html > html > body > div.container.my

```
38         <th>Question Number</th>
39         <th>Question</th>
40         <th>Choose an Option</th>
41         <th>Actions</th>
42     </tr>
43 </thead>
44 <tbody>
45     <tr th:each="mcq: ${listQuestions}">
46         <td th:text="${mcq.id}" />
47         <td th:text="${mcq.question}" />
48         <!-- <td th:text="${mcq.option_a}" />
49         <td th:text="${mcq.option_b}" />
50         <td th:text="${mcq.option_c}" />
51         <td th:text="${mcq.option_d}" /> -->
52     <td>
53         <div class="form-check">
54             <input
55                 class="form-check-input"
56                 type="radio"
57                 name="answer[${mcq.id}]"
58                 id="option_a[${mcq.id}]"
59                 value="a"
60             />
61             <label
62                 class="form-check-label"
63                 for="option_a[${mcq.id}]"
64                 th:text="${mcq.option_a}"
65             />
66         </div>
67         <div class="form-check">
68             <input
69                 class="form-check-input"
70                 type="radio"
71                 name="answer[${mcq.id}]"
72                 id="option_b[${mcq.id}]"
73                 value="b"
74             />
```

```

index.html X
demo > src > main > resources > templates > index.html > html > b
75         <label
76             class="form-check-label"
77             for="option_b[{{mcq.id}}]"
78             th:text="{{mcq.option_b}}"
79         />
80     </div>
81     <div class="form-check">
82         <input
83             class="form-check-input"
84             type="radio"
85             name="answer[{{mcq.id}}]"
86             id="option_c[{{mcq.id}}]"
87             value="c"
88         />
89         <label
90             class="form-check-label"
91             for="option_c[{{mcq.id}}]"
92             th:text="{{mcq.option_c}}"
93         />
94     </div>
95     <div class="form-check">
96         <input
97             class="form-check-input"
98             type="radio"
99             name="answer[{{mcq.id}}]"
100            id="option_d[{{mcq.id}}]"
101            value="d"
102        />
103        <label
104            class="form-check-label"
105            for="option_d[{{mcq.id}}]"
106            th:text="{{mcq.option_d}}"
107        />
108    </div>
109 </td>
110 <td>
111     <a

```

```

112         th:href="@{/showFormForUpdate/{id}(id={{mcq.id}})"
113         class="btn btn-primary"
114     >Modify Question</a>
115     >
116     <a
117         th:href="@{/deleteQuestion/{id}(id={{mcq.id}})"
118         class="btn btn-danger"
119     >Delete Question</a>
120     >
121 </td>
122 </tr>
123 </tbody>
124 </table>
125
126     <button type="submit" class="btn btn-primary">Submit</button>
127 </form>
128 </div>
129 </body>
130 </html>
131

```

new_question.html

```
<html lang="en" xmlns:th="http://www.thymeleaf.org">
  <head>
    <title>Quiz Web Application</title>

    <link
      rel="stylesheet"
      href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/css/bootstrap.min.css"
      integrity="sha384-B0vP5xmATw1+K9KRQjQERJvTumQW0nPEzvf6L/Z6nronJ3oUOFUFpCjEUQouq2+1"
      crossorigin="anonymous"
    />
  </head>

  <body>
    <div class="container">
      <h1>Quiz Web Application</h1>
      <hr></hr>
      <h2>Save Question</h2>

      <form
        action="#"
        th:action="@{/saveQuestion}"
        th:object="${Quiz}"
        method="POST"
      >
        <input
          type="number"
          th:field="*{id}"
          placeholder="Question Number"
          class="form-control mb-4 col-4"
        />
        <input
          type="text"
          th:field="*{question}"
          placeholder="Question"
          class="form-control mb-4 col-4"
        />
        <input
          type="text"
          th:field="*{option_a}"
          placeholder="Option A"
          class="form-control mb-4 col-4"
        />
        <input
          type="text"
          th:field="*{option_b}"
```

```

        placeholder="Option B"
        class="form-control mb-4 col-4"
    />
    <input
        type="text"
        th:field="*{option_c}"
        placeholder="Option C"
        class="form-control mb-4 col-4"
    />
    <input
        type="text"
        th:field="*{option_d}"
        placeholder="Option D"
        class="form-control mb-4 col-4"
    />
    <input
        type="text"
        th:field="*{correct_answer}"
        placeholder="Correct Answer"
        class="form-control mb-4 col-4"
    />

    <button type="submit" class="btn btn-info col-2">Save
Question</button>

    </form>
</div>
</body>
</html>

```

update_question.html

```

<html Lang="en" xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>Quiz Web Application</title>

        <link
            rel="stylesheet"
            href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/css/bootstrap.min.css"
            integrity="sha384-
B0vP5xmATw1+K9KRQjQERJvTumQW0nPEzvF6L/Z6nronJ3oU0FUFPcJEUQouq2+1"
            crossorigin="anonymous"
        />
    </head>

```

```
<body>
  <div class="container">
    <h1>Quiz Web Application</h1>
    <hr />
    <h2>Modify Question</h2>

    <form
      action="#"
      th:action="@{/saveQuestion}"
      th:object="${Quiz}"
      method="POST"
    >
      <!-- Add hidden field to handle update -->

      <input type="hidden" th:field="*{id}" />
      <input
        type="number"
        th:field="*{id}"
        placeholder="Question Number"
        class="form-control mb-4 col-4"
      />
      <input
        type="text"
        th:field="*{question}"
        placeholder="Question"
        class="form-control mb-4 col-4"
      />
      <input
        type="text"
        th:field="*{option_a}"
        placeholder="Option A"
        class="form-control mb-4 col-4"
      />
      <input
        type="text"
        th:field="*{option_b}"
        placeholder="Option B"
        class="form-control mb-4 col-4"
      />
      <input
        type="text"
        th:field="*{option_c}"
        placeholder="Option C"
        class="form-control mb-4 col-4"
      />
    </form>
  </div>
</body>
```

```

        <input
            type="text"
            th:field="*{option_d}"
            placeholder="Option D"
            class="form-control mb-4 col-4"
        />
        <input
            type="text"
            th:field="*{correct_answer}"
            placeholder="Correct Answer"
            class="form-control mb-4 col-4"
        />

        <button type="submit" class="btn btn-info col-2">
            Update Question
        </button>
    </form>

    <hr />
    <a th:href="@{/}"> Back to MCQ List</a>
</div>
</body>
</html>

```

application.properties

```

demo > src > main > resources > application.properties
1  spring.jpa.hibernate.ddl-auto=update
2  spring.datasource.url=jdbc:mysql://${MYSQL_HOST:localhost}:3306/springbootquiz
3  spring.datasource.username=root
4  spring.datasource.password=Ananya@123
5  spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
6  #spring.jpa.show-sql: true
7

```


Screenshot of console with application running

```
C:\Windows\System32\cmd.exe - mvn spring-boot:run
C:\Users\anany\Quiz\demo>mvn spring-boot:run
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.example:demo >-----
[INFO] Building demo 0.0.1-SNAPSHOT
[INFO] from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] >>> spring-boot:3.0.5:run (default-cli) > test-compile @ demo >>>
[INFO]
[INFO] --- resources:3.3.0:resources (default-resources) @ demo ---
[INFO] Copying 1 resource
[INFO] Copying 3 resources
[INFO]
[INFO] --- compiler:3.10.1:compile (default-compile) @ demo ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- resources:3.3.0:testResources (default-testResources) @ demo ---
[INFO] skip non existing resourceDirectory C:\Users\anany\Quiz\demo\src\test\resources
[INFO]
[INFO] --- compiler:3.10.1:testCompile (default-testCompile) @ demo ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] <<< spring-boot:3.0.5:run (default-cli) < test-compile @ demo <<<
[INFO]
[INFO]
[INFO] --- spring-boot:3.0.5:run (default-cli) @ demo ---
```

```
C:\Windows\System32\cmd.exe - mvn spring-boot:run
[INFO] Attaching agents: []

  ____ _
 / ___ \ | |
/ /___ \ | |
 \___ \_|_|
  =====|=====
  :: Spring Boot ::                (v3.0.5)

2023-04-13T10:38:05.307+05:30 INFO 19844 --- [ restartedMain] c.e.S.SpringBootThymeleafWebApplication : Starting Sp
ringBootThymeleafWebApplication using Java 19.0.1 with PID 19844 (C:\Users\anany\Quiz\demo\target\classes started by A
NANYA in C:\Users\anany\Quiz\demo)
2023-04-13T10:38:05.310+05:30 INFO 19844 --- [ restartedMain] c.e.S.SpringBootThymeleafWebApplication : No active p
rofile set, falling back to 1 default profile: "default"
2023-04-13T10:38:05.363+05:30 INFO 19844 --- [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools pr
operty defaults active! Set 'spring.devtools.add-properties' to 'false' to disable
2023-04-13T10:38:05.364+05:30 INFO 19844 --- [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additio
nal web related logging consider setting the 'logging.level.web' property to 'DEBUG'
2023-04-13T10:38:05.851+05:30 INFO 19844 --- [ restartedMain] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrappi
ng Spring Data JPA repositories in DEFAULT mode.
2023-04-13T10:38:05.899+05:30 INFO 19844 --- [ restartedMain] .s.d.r.c.RepositoryConfigurationDelegate : Finished Sp
ring Data repository scanning in 40 ms. Found 1 JPA repository interfaces.
2023-04-13T10:38:06.523+05:30 INFO 19844 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat init
ialized with port(s): 8080 (http)
2023-04-13T10:38:06.533+05:30 INFO 19844 --- [ restartedMain] o.apache.catalina.core.StandardService : Starting se
rvice [Tomcat]
```

```

2023-04-13T10:38:06.534+05:30 INFO 19844 --- [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Se
rvlet engine: [Apache Tomcat/10.1.7]
2023-04-13T10:38:06.611+05:30 INFO 19844 --- [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializin
g Spring embedded WebApplicationContext
2023-04-13T10:38:06.614+05:30 INFO 19844 --- [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApp
licationContext: initialization completed in 1247 ms
2023-04-13T10:38:06.768+05:30 INFO 19844 --- [ restartedMain] o.hibernate.jpa.internal.util.LogHelper : HHH000204:
Processing PersistenceUnitInfo [name: default]
2023-04-13T10:38:06.801+05:30 INFO 19844 --- [ restartedMain] org.hibernate.Version : HHH000412:
Hibernate ORM core version 6.1.7.Final
2023-04-13T10:38:07.022+05:30 INFO 19844 --- [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-
1 - Starting...
2023-04-13T10:38:07.342+05:30 INFO 19844 --- [ restartedMain] com.zaxxer.hikari.pool.HikariPool : HikariPool-
1 - Added connection com.mysql.cj.jdbc.ConnectionImpl@14f70b1
2023-04-13T10:38:07.344+05:30 INFO 19844 --- [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-
1 - Start completed.
2023-04-13T10:38:07.406+05:30 INFO 19844 --- [ restartedMain] SQL dialect : HHH000400:
Using dialect: org.hibernate.dialect.MySQLDialect
2023-04-13T10:38:08.030+05:30 INFO 19844 --- [ restartedMain] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490:
Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2023-04-13T10:38:08.038+05:30 INFO 19844 --- [ restartedMain] j.LocalContainerEntityManagerFactoryBean : Initialized
JPA EntityManagerFactory for persistence unit 'default'
2023-04-13T10:38:08.256+05:30 WARN 19844 --- [ restartedMain] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.
open-in-view is enabled by default. Therefore, database queries may be performed during view rendering. Explicitly con
figure spring.jpa.open-in-view to disable this warning
2023-04-13T10:38:08.394+05:30 INFO 19844 --- [ restartedMain] o.s.b.a.w.s.WelcomePageHandlerMapping : Adding welc
ome page template: index
2023-04-13T10:38:08.590+05:30 INFO 19844 --- [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload
server is running on port 35729
2023-04-13T10:38:08.636+05:30 INFO 19844 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat star
ted on port(s): 8080 (http) with context path ''
2023-04-13T10:38:08.648+05:30 INFO 19844 --- [ restartedMain] c.e.s.SpringBootThymeleafWebApplication : Started Spr
ingBootThymeleafWebApplication in 3.663 seconds (process running for 4.007)
2023-04-13T10:38:24.376+05:30 INFO 19844 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializin
g Spring DispatcherServlet 'dispatcherServlet'
2023-04-13T10:38:24.376+05:30 INFO 19844 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initializin
g Servlet 'dispatcherServlet'
2023-04-13T10:38:24.379+05:30 INFO 19844 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Completed i
nitialization in 2 ms

```

Screenshot of UIs related to the two scenarios


Quizzz Yourself !



Multiple Choice Questions

[Add Question](#)

Question Number	Question	Choose an Option	Actions
1	What is the capital of France?	<input type="radio"/> Paris <input type="radio"/> London <input type="radio"/> Madrid <input type="radio"/> San Diego	Modify Question Delete Question
2	What is the largest country in the world?	<input type="radio"/> Russia <input type="radio"/> Canada <input type="radio"/> China <input type="radio"/> USA	Modify Question Delete Question
3	What is Capital of Haryana?	<input type="radio"/> Pune <input type="radio"/> Mumbai <input type="radio"/> Chandigarh <input type="radio"/> Bengaluru	Modify Question Delete Question
4	What company was originally called "Cadabra"?	<input type="radio"/> Flipkart <input type="radio"/> Amazon <input type="radio"/> Google <input type="radio"/> Microsoft	Modify Question Delete Question

[Submit](#)

➤ Create data items

Quiz Web Application

Save Question

5

Which of the following is NOT written by Munshi

Gaban

Godan

Guide

Manasorovar

Guide

Save Question

➤ Read the newly created data items

Add Question			
Question Number	Question	Choose an Option	Actions
1	What is the capital of France?	<input type="radio"/> Paris <input type="radio"/> London <input type="radio"/> Madrid <input type="radio"/> San Diego	Modify Question Delete Question
2	What is the largest country in the world?	<input type="radio"/> Russia <input type="radio"/> Canada <input type="radio"/> China <input type="radio"/> USA	Modify Question Delete Question
3	What is Capital of Haryana?	<input type="radio"/> Pune <input type="radio"/> Mumbai <input type="radio"/> Chandigarh <input type="radio"/> Bengaluru	Modify Question Delete Question
4	What company was originally called "Cadabra"?	<input type="radio"/> Flipkart <input type="radio"/> Amazon <input type="radio"/> Google <input type="radio"/> Microsoft	Modify Question Delete Question
5	Which of the following is NOT written by Munshi Premchand?	<input type="radio"/> Gaban <input type="radio"/> Godan <input type="radio"/> Guide <input type="radio"/> Manasorovar	Modify Question Delete Question

➤ Update item

- Before Modifying:

2	What is the largest country in the world?	<input type="radio"/> Russia <input type="radio"/> Canada <input type="radio"/> China <input type="radio"/> USA	Modify Question	Delete Question
---	---	--	---------------------------------	---------------------------------

Quiz Web Application

Modify Question

2
What is the largest country in the world?
Russia
Canada
China
USA
Russia
Update Question

[Back to MCQ List](#)

- Modifying:

Quiz Web Application

Modify Question

2
What is the smallest country in the world?
Russia
Vatican City
China
USA
Vatican City
Update Question

[Back to MCQ List](#)

- After Modifying

Question Number	Question	Choose an Option	Actions
1	What is the capital of France?	<input type="radio"/> Paris <input type="radio"/> London <input type="radio"/> Madrid <input type="radio"/> San Diego	Modify Question Delete Question
2	What is the smallest country in the world?	<input type="radio"/> Russia <input type="radio"/> Vatican City <input type="radio"/> China <input type="radio"/> USA	Modify Question Delete Question
3	What is Capital of Haryana?	<input type="radio"/> Pune <input type="radio"/> Mumbai <input type="radio"/> Chandigarh <input type="radio"/> Bengaluru	Modify Question Delete Question
4	What company was originally called "Cadabra"?	<input type="radio"/> Flipkart <input type="radio"/> Amazon <input type="radio"/> Google <input type="radio"/> Microsoft	Modify Question Delete Question
5	Which of the following is NOT written by Munshi Premchand?	<input type="radio"/> Gaban <input type="radio"/> Godan <input type="radio"/> Guide <input type="radio"/> Manasorovar	Modify Question Delete Question

[Submit](#)

➤ Deleting Item

Before deletion:

[Add Question](#)

Question Number	Question	Choose an Option	Actions
1	What is the capital of France?	<input type="radio"/> Paris <input type="radio"/> London <input type="radio"/> Madrid <input type="radio"/> San Diego	Modify Question Delete Question
2	What is the smallest country in the world?	<input type="radio"/> Russia <input type="radio"/> Vatican City <input type="radio"/> China <input type="radio"/> USA	Modify Question Delete Question
3	What is Capital of Haryana?	<input type="radio"/> Pune <input type="radio"/> Mumbai <input type="radio"/> Chandigarh <input type="radio"/> Bengaluru	Modify Question Delete Question
4	What company was originally called "Cadabra"?	<input type="radio"/> Flipkart <input type="radio"/> Amazon <input type="radio"/> Google <input type="radio"/> Microsoft	Modify Question Delete Question
5	Which of the following is NOT written by Munshi Premchand?	<input type="radio"/> Gaban <input type="radio"/> Godan <input type="radio"/> Guide <input type="radio"/> Manasorovar	Modify Question Delete Question

After Deletion of one question:

 **Quizzz Yourself !**  

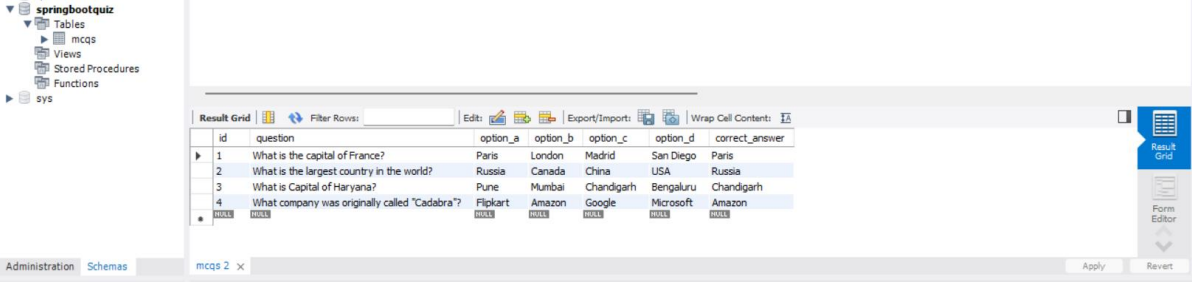
Multiple Choice Questions

[Add Question](#)

Question Number	Question	Choose an Option	Actions
1	What is the capital of France?	<input type="radio"/> Paris <input type="radio"/> London <input type="radio"/> Madrid <input type="radio"/> San Diego	Modify Question Delete Question
2	What is the smallest country in the world?	<input type="radio"/> Russia <input type="radio"/> Vatican City <input type="radio"/> China <input type="radio"/> USA	Modify Question Delete Question
4	What company was originally called "Cadabra"?	<input type="radio"/> Flipkart <input type="radio"/> Amazon <input type="radio"/> Google <input type="radio"/> Microsoft	Modify Question Delete Question
5	Which of the following is NOT written by Munshi Premchand?	<input type="radio"/> Gaban <input type="radio"/> Godan <input type="radio"/> Guide <input type="radio"/> Manasorovar	Modify Question Delete Question

[Submit](#)

 Screenshot of database with data items



id	question	option_a	option_b	option_c	option_d	correct_answer
1	What is the capital of France?	Paris	London	Madrid	San Diego	Paris
2	What is the largest country in the world?	Russia	Canada	China	USA	Russia
3	What is Capital of Haryana?	Pune	Mumbai	Chandigarh	Bengaluru	Chandigarh
4	What company was originally called "Cadabra"?	Flipkart	Amazon	Google	Microsoft	Amazon