PHASE-END PROJECT

Project Agenda: Create an online quiz portal, where users can explore multiple quiz, attempt the quiz, and save their position on the leaderboard

Source code

com.Controller

Validate admin

```
package com.controller;
// Importing lib
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
// Importing other Java files
import com.db.Admins;
import com.model.AdminsDao;
@WebServlet("/com.controller.ValidateAdmin")
public class ValidateAdmin extends HttpServlet {
        private static final long serialVersionUID = 1L;
    public ValidateAdmin() {
    super();
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {


                PrintWriter out = response.getWriter();
                response.setContentType("text/html");

                String username = request.getParameter("uname");
                String password = request.getParameter("pass");
                Admins ad = new Admins(); // Creating object
                ad.setUsername(username);
                ad.setPassword(password);

          boolean status = AdminsDao.doValidate(ad);

          if(status)
          {

                //Logged in  as admin do something (pending)
                 HttpSession session = request.getSession();
                        session.setAttribute("username", ad.getUsername());
                        response.sendRedirect("AdminPanel.jsp");
          }
          else
          {
                String msg2 = "Invalid Username or Password";
                response.sendRedirect("AdminLogin.jsp?msg2="+msg2);
          }


        }

}
```

Validate User

```
package com.controller;
// Importing all libs
import java.io.*;
import javax.servlet.*;
import javax.servlet.annotation.*;
import javax.servlet.http.*;

//importing all packages
//import com.db.Admins;
import com.db.Users;
//import com.model.AdminsDao;
import com.model.Users;

@WebServlet("/com.controller.ValidateUser")
public class ValidateUser extends HttpServlet {
        private static final long serialVersionUID = 1L;
```

```
  public ValidateStudent() {
    super();

  }


       protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {

              PrintWriter out = response.getWriter();
              response.setContentType("text/html");

              String username = request.getParameter("uname");
              String password = request.getParameter("pass");



         Users sd = new Users();
              sd.setUsername(username);
              sd.setPassword(password);

         boolean status = UserDao.doValidate(sd);


         if(status)
         {

                            HttpSession studentsession = request.getSession();
                     studentsession.setAttribute("username", sd.getUsername());
                     studentsession.setAttribute("name",sd.getName() );


         }
         else
         {
              String msg = "Invalid Username or Password";
              response.sendRedirect("UserLogin.jsp?msg="+msg);

       }

}
}
```

QuestionInsert

```
package com.controller;

import java.io.*;
import javax.servlet.*;
import javax.servlet.annotation.*;
import javax.servlet.http.*;
import java.io.PrintWriter;
import com.db.Questions;
import com.model.QuestionsDao;

/**
 * implementation class QuestionInsert
 */
@WebServlet("/com.controller.QuestionInsert")
public class QuestionInsert extends HttpServlet {
        private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public QuestionInsert() {
        super();
        // TODO Auto-generated constructor stub
    }

        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
                // TODO Auto-generated method stub
                PrintWriter out = response.getWriter();
                response.setContentType("text/html");
                String question = request.getParameter("Question");
                String a = request.getParameter("option1");
                String b = request.getParameter("option2");
                String c = request.getParameter("option3");
                String d = request.getParameter("option4");
                String answer = request.getParameter("answer");

                Questions q = new Questions();
                q.setA(a);
                q.setB(b);
                q.setC(c);
                q.setD(d);
                q.setAnswer(answer);
                q.setQuestion(question);

                boolean status = QuestionsDao.insertQuestion(q);

                if(status)
                {
                    String msg1 = "Question Added";
                response.sendRedirect("AddQuestion.jsp?msg1="+msg1);
                }
                else
                {
                    String msg2 = "Question not Added";
                response.sendRedirect("AddQuestion.jsp?msg2="+msg2);
                }



        }

}
```

QuizInsert

```
package com.controller;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import com.db.Quizs;
import com.model.CreateDao;

/**
 * Servlet implementation class QuizInsert
 */
@WebServlet("/QuizInsert")
public class QuizInsert extends HttpServlet {
            private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public QuizInsert() {
        super();
        // TODO Auto-generated constructor stub
    }

            /**
             * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
             */
            protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
                        // TODO Auto-generated method stub
                        //response.getWriter().append("Served at: ").append(request.getContextPath());
                         int quizId = Integer.parseInt(request.getParameter("quizId"));
                        String quizTitle = request.getParameter("quizTitle");
                    String category = request.getParameter("category");

                    // Create a new Quizs object and set its properties
                    Quizs quiz = new Quizs();
                    quiz.setId(quizId);
                    quiz.setQuizTitle(quizTitle);
                    quiz.setCategory(category);

                    // Save the quiz details to the database using CreateDao
                    if (CreateDao.insertQuiz(quiz))
                    {

                        response.sendRedirect("quizinsert.jsp?status=success");
                    }
```

```
                else
                {
                    response.sendRedirect("quizinsert.jsp?status=fail");
        }

         }

         /**
          * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
          */


}
```

userInsert

```
package com.controller;

import java.io.*;
import javax.servlet.*;
import javax.servlet.annotation.*;
import javax.servlet.http.*;

import com.db.Students;
import com.model.StudentsDao;

/**
 * implementation class UserInsert
 */
@WebServlet("/com.controller.UserInsert")
public class UserInsert extends HttpServlet {
         private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public StudentInsert() {
        super();
        // TODO Auto-generated constructor stub
    }

         /**
          * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
```

```
	 */
	protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
		// TODO Auto-generated method stub
		PrintWriter out = response.getWriter();
		response.setContentType("text/html");

		String username = request.getParameter("uname");
		String password = request.getParameter("pass");
		String name = request.getParameter("name");

		Students st = new Students();
		st.setName(name);
		st.setPassword(password);
		st.setUsername(username);

		boolean status = StudentsDao.insertStudent(st);
		if(status)
		{
			String msg1 = "User Register successfully";
		response.sendRedirect("AddUser.jsp?msg1="+msg1);
		}
		else
		{
			String msg2 = "Not Registerd successfully";
		response.sendRedirect("AddUser.jsp?msg2="+msg2);
		}

	}

}
```

Com.db

Admins

```
package com.db;
//DB Admins
public class Admins {

	public static int precount = 0;
	private String username;
	private String password;
	public String getUsername() {
		return username;
```

```
      }
      public void setUsername(String username) {
            this.username = username;
      }
      public String getPassword() {
            return password;
      }
      public void setPassword(String password) {
            this.password = password;
      }

}
```

Questions

```
package com.db;
//import java.util.*;
public class Questions {
      private int id;
    public int getId() {
            return id;
      }
      public void setId(int id) {
            this.id = id;
      }
      private String question;
    private String a;
    private String b;
    private String c;
    private String d;
    private String answer;
    public String getQuestion() {
        return question;
    }
    public void setQuestion(String question) {
        this.question = question;
    }
    public String getA() {
        return a;
    }
    public void setA(String a) {
        this.a = a;
    }
    public String getB() {
        return b;
    }
    public void setB(String b) {
        this.b = b;
    }
    public String getC() {
        return c;
    }
    public void setC(String c) {
        this.c = c;
    }
```

```java
    public String getD() {
        return d;
    }
    public void setD(String d) {
        this.d = d;
    }
    public String getAnswer() {
        return answer;
    }
    public void setAnswer(String answer) {
        this.answer = answer;
    }
}
```

Quizs

```java
package com.db;

public class Quizs {
        private int id;
    private String quizTitle;
    private String category;




      public int getId() {
             return id;
      }

      public void setId(int id) {
             this.id = id;
      }

      public String getQuizTitle() {
        return quizTitle;
    }

    public void setQuizTitle(String quizTitle) {
        this.quizTitle = quizTitle;
    }

    public String getCategory() {
        return category;
    }

    public void setCategory(String category) {
        this.category = category;
    }


}
```

Users

```
package com.db;

public class Users {
	private String username;
	private String password;
	private String name;
	public String getName() {
		return name;
	}
	public void setName(String name) {
		this.name = name;
	}
	public String getUsername() {
		return username;
	}
	public void setUsername(String username) {
		this.username = username;
	}
	public String getPassword() {
		return password;
	}
	public void setPassword(String password) {
		this.password = password;
	}

}
```

Com.model

AdminDao

```
package com.model;
//Lib import
import java.sql.*;
// Packages import
import com.db.Admins;



public class AdminsDao {
   public static boolean doValidate(Admins ad) {

      boolean status = false;
      try
      {
              Connection con=DBConnection.getConnection();
            String sql = "select * from admintable where userid=? and password=?"; // SQL Query
                  PreparedStatement pst = con.prepareStatement(sql);
                  pst.setString(1, ad.getUsername());
                  pst.setString(2, ad.getPassword());
                  ResultSet rs = pst.executeQuery();
                  if(rs.next())
            {
                status = true;
                  }
                  else
            {
                status = false;
            }
        }catch(Exception e){System.out.println(e);}
              return status;
    }
}
```

CraeteDao

```
package com.model;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;

import com.db.Quizs;

import com.db.Questions;

public class CreateDao {

    public static boolean insertQuiz(Quizs quiz) {
        boolean status = false;
        try {
            Connection con = DBConnection.getConnection();
            String sql = "INSERT INTO quiz (id,title, category) VALUES (?, ?, ?)";
            PreparedStatement pst = con.prepareStatement(sql);
            pst.setInt(1, quiz.getId());
            pst.setString(2, quiz.getQuizTitle());
            pst.setString(3, quiz.getCategory());

            int val = pst.executeUpdate();
            if (val > 0) {
                status = true;
            } else {
                status = false;
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
        return status;
    }



    public static ArrayList<Quizs> getAllQuizzes() {
        ArrayList<Quizs> quizzesList = new ArrayList<>();
        try {
            Connection con = DBConnection.getConnection();
            String sql = "SELECT * FROM quiz";
            PreparedStatement pst = con.prepareStatement(sql);

            ResultSet rs = pst.executeQuery();
            while (rs.next()) {
                Quizs quiz = new Quizs();
                quiz.setId(rs.getInt("id"));
                quiz.setQuizTitle(rs.getString("title"));
                quiz.setCategory(rs.getString("category"));
                quizzesList.add(quiz);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
        return quizzesList;
    }


    public static int updateQuiz(int OriginalId, int newId,String title,String category) {
              int status = 0;
    try {
        Connection con = DBConnection.getConnection();
        String sql = "UPDATE quiz SET title=?, category=? where id=?";
        PreparedStatement pst = con.prepareStatement(sql);
        pst.setInt(1, newId);
        pst.setString(2, title);
        pst.setString(3, category);



        int val = pst.executeUpdate();
                            if(val > 0)
                            {
```

```
                       status = 1;
               }
               else
               {
                       status = -1;
               }
               }
               catch(Exception f)
               {
                       status = 2;
                       f.printStackTrace();
               }


               return status;
}
  public static int deleteQuiz(Quizs q) {
        int status = 0;
    try {
      Connection con = DBConnection.getConnection();
      String sql = "DELETE FROM quiz WHERE id=?";
      PreparedStatement pst = con.prepareStatement(sql);
      pst.setInt(1, q.getId());

      int val = pst.executeUpdate();
          if(val > 0)
          {
               status =1;
          }
          else
          {
               status = 0;
          }

               }
               catch(Exception e)
               {
                       e.printStackTrace();
               }

               return status;
  }
  public static Quizs getQuizById(int quizId) {
    Quizs quiz = null;
    try {
      Connection con = DBConnection.getConnection();
      String sql = "SELECT * FROM quiz WHERE id=?";
      PreparedStatement pst = con.prepareStatement(sql);
```

```
            pst.setInt(1, quizId);

            ResultSet rs = pst.executeQuery();
            if (rs.next()) {
                quiz = new Quizs();
                quiz.setId(rs.getInt("id"));
                quiz.setQuizTitle(rs.getString("title"));
                quiz.setCategory(rs.getString("category"));
            }
            con.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return quiz;
    }

    // Method to get questions for a specific quiz based on its ID
    public static ArrayList<Questions> getQuestionsByQuizId(int quizId) {
        ArrayList<Questions> questionsList = new ArrayList<>();
        try {
            Connection con = DBConnection.getConnection();
            String sql = "SELECT * FROM questiontable WHERE question=?";
            PreparedStatement pst = con.prepareStatement(sql);
            pst.setInt(1, quizId);

            ResultSet rs = pst.executeQuery();
            while (rs.next()) {
                Questions question = new Questions();

                question.setQuestion(rs.getString("question"));
                question.setA(rs.getString("A"));
                question.setB(rs.getString("B"));
                question.setC(rs.getString("C"));
                question.setD(rs.getString("D"));
                question.setAnswer(rs.getString("answer"));
                questionsList.add(question);
            }
            con.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return questionsList;
    }
}
```

QuestionDao

```
package com.model;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;


import com.db.Questions;
//import com.db.Students;

public class QuestionsDao {

        public static boolean insertQuestion(Questions q) {
                // TODO Auto-generated method stub
                boolean status = false;
                try
                {

                        Connection con=DBConnection.getConnection();
                String sql = "insert into questiontable values(?,?,?,?,?,?)";
                PreparedStatement pst = con.prepareStatement(sql);
          pst.setString(1, q.getQuestion().trim());
          pst.setString(2, q.getA());
          pst.setString(3, q.getB());
          pst.setString(4, q.getC());
          pst.setString(5, q.getD());
          pst.setString(6, q.getAnswer());
         // pst.setString(7,"e");

          int val = pst.executeUpdate();
          if(val > 0)
          {
                status =true;
          }
          else
          {
                status = false;
          }

                }
                catch(Exception e)
                {
                        e.printStackTrace();
                }
                return status;
```

```
    }
    public static ArrayList<Questions> getAllRecords()
    {
ArrayList<Questions>  samp =new ArrayList<Questions>();
             try
             {

               samp.clear();
               Connection con=DBConnection.getConnection();
               String sql = "select * from questiontable";
               PreparedStatement pst = con.prepareStatement(sql);

               ResultSet rs = pst.executeQuery(sql);
               while(rs.next())
               {
                       Questions  s = new Questions();
                       s.setQuestion(rs.getString(1));
                       s.setAnswer(rs.getString(6));
                       s.setA(rs.getString(2));
                       s.setB(rs.getString(3));
                       s.setC(rs.getString(4));
                       s.setD(rs.getString(5));
                        samp.add(s);



               }


             }
             catch(Exception e2)
             {
                      e2.printStackTrace();
             }
             return samp;
    }
    public static int deleteRecord(Questions q)
    {
             int status = 0;
             try
             {

                      Connection con=DBConnection.getConnection();
             String sql = "delete from questiontable where question=?";
             PreparedStatement pst = con.prepareStatement(sql);
      pst.setString(1, q.getQuestion());
```

```
          int val = pst.executeUpdate();
          if(val > 0)
          {
                status =1;
          }
          else
          {
                status = 0;
          }

                }
                catch(Exception e)
                {
                        e.printStackTrace();
                }

                return status;

        }
        public static int doUpdateNowRecord(String OriginalQuestion,String newQuestion,String
opta,String optb,String optc,String optd,String ans)
        {
                int status = 0;
                try
                {

                        Connection con=DBConnection.getConnection();
                String sql = "update questiontable set question=?,a=?,b=?,c=?,d=?,answer=? where
question=?";
                PreparedStatement pst = con.prepareStatement(sql);
                pst.setString(1,newQuestion);
                pst.setString(2, opta);
                pst.setString(3, optb);
                pst.setString(4, optc);
                pst.setString(5, optd);
                pst.setString(6, ans);
                pst.setString(7, OriginalQuestion);

                int val = pst.executeUpdate();
                if(val > 0)
                {
                        status = 1;
                }
                else
                {
                        status = -1;
                }
                }
```

```
                    catch(Exception f)
                    {
                              status = 2;
                              f.printStackTrace();
                    }


                    return status;
          }


}
```

DBConnection

```
package com.model;

import java.sql.Connection;
import java.sql.DriverManager;

public class DBConnection {

           private static Connection con;
          public static Connection getConnection() {
                    try {
                    //Step 1: load driver in memory
                    Class.forName("com.mysql.cj.jdbc.Driver");
                    //Step 2: Connection with database

          con=DriverManager.getConnection("jdbc:mysql://localhost:3306/OnlineQuiz","root","Anusuya 
manju@123");
                    }
                    catch(Exception e) {
                              e.printStackTrace();
                    }
                    return con;
          }

          }
```

JSP PAGES

AdminLogin

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
	li {
		float: left;
	}
	ul {
		list-style-type: none;
		margin: 0;
		padding: 0;
		overflow: hidden;
		background-color: #333;
	}
	li a {
		display: inline-block;
		color: white;
		text-align: center;
		padding: 14px 16px;
		text-decoration: none;
	}
	li a:hover {
		background-color: aqua;
	}
	.center-content {
		text-align: center;
		color: blue;
	}
	.error-msg {
		color: red;
	}
	.success-msg {
		color: green;
	}
</style>
<title>Admin Login</title>
</head>
<body>
	<div class="center-content">
		<h1>Admin Login</h1>
	</div>
	<div class="error-msg">
		<%
			if(request.getParameter("msg2") != null)
				out.print(request.getParameter("msg2"));
		%>
	</div>
	<div class="success-msg">
```

```
             <%
                    if(request.getParameter("msg1") != null)
                          out.print(request.getParameter("msg1"));
             %>
       </div>
       <form action="com.controller.ValidateAdmin" method="post">
             <pre>
                                                              Username : <input
type="text" name="uname">

                                                              Password : <input
type="password" name="pass">

                                                                    <input
onclick="location.href='index.html'" type="button" value="Exit" class="btn btn-
outline-primary">    <input type="submit" value="Login" class="btn btn-outline-
primary">
             </pre>
       </form>
</body>
</html>
```

AdminPanel

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Admin Panel</title>
    <style type="text/css">
        /* Center the content */
        body {
            text-align: center;
        }
        /* Center the container */
        .container {
            display: inline-block;
            text-align: left;
        }
        /* Set the height and background color for the top section */

        }
        /* Add padding for the buttons */
        .container input[type="button"] {
            margin: 10px;
        }
    </style>
</head>
<body>
<%
if (session.getAttribute("username") != null) {
} else {
    String msg2 = "Please Login as Admin to Continue";
    response.sendRedirect("AdminLogin.jsp?msg2=" + msg2);
}
%>
<div class="top-section">
  <marquee behavior="alternate">
     <h1> <font color="green" ><center>Welcome <%= session.getAttribute("username")
%></center></font></h1>
  </marquee>
</div>

<pre>
</pre>
<input onclick="location.href='Quiz Manager.jsp'" type="button" value="Quiz Manager">
<input onclick="location.href='QuestionList.jsp'" type="button" value="Question
Manager">          <input onclick="location.href='com.controller.LogoutAdmin'"
type="button" value="Logout">

</body>
</html>
```

QuizList

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" %>
<%@ page import="com.model.CreateDao" %>
<%@ page import="com.db.Quizs" %>
<%@page import="java.util.ArrayList"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Quiz List</title>
</head>
<body>
    <h1>Quiz List</h1>
    <table border=2>
        <tr>
        <th>Quiz Id</th>

            <th>Quiz Title</th>
            <th>Category</th>
            <th>Action</th>
        </tr>
        <%
            ArrayList<Quizs> allQuizzes = CreateDao.getAllQuizzes();
            for (Quizs quiz : allQuizzes) {
        %>
        <tr>

         <td><%= quiz.getId() %></td>
```

```
            <td><%= quiz.getQuizTitle() %></td>
            <td><%= quiz.getCategory() %></td>
            <td><a href="updatequiz.jsp?id=<%= quiz.getQuizTitle()
%>">Update</a></td>
            <td><a href="deletequiz.jsp?id=<%= quiz.getQuizTitle()
%>">Delete</a></td>

        </tr>
        <%
            }
        %>
    </table>
</body>
</html>
```

updateQuiz

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"
%>
<%@ page import="com.model.CreateDao" %>
<%@ page import="com.db.Quizs" %>
<%@ page import="java.util.ArrayList" %>

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Quiz List</title>
</head>
<body>
    <h1>Quiz List</h1>

     <table border=2>
        <tr class="table-dark">
            <th>Quiz ID</th>
            <th>Quiz Title</th>
            <th>Category</th>
            <th>Update</th>
        </tr>
        <%
            ArrayList<Quizs> allQuizzes = CreateDao.getAllQuizzes();
            for (Quizs quiz : allQuizzes) {
              if (request.getParameter("id").equals(quiz.getId())) {
        %>
         <tr class="table-primary">
        <form action="updatequiznow.jsp">
        <td><input type="hidden" value="<%=quiz.getId()%>" type="text"
name="idoriginal"><%=quiz.getId()%></td>


          <td> <input type="text" value="<%=quiz.getId()%>"
name="idmodified"></input> </td>
```

```
          <td> <input type="text" value="<%=quiz.getQuizTitle()%>" name="na"></input>
</td>
          <td><input type="text" value="<%=quiz.getCategory()%>" name="na1"></td>
           <td><a href="deletequiz.jsp?quz=<%=quiz.getId()%>">Delete</a></td>
          </form>
          </tr>
           <%
      } else {
  %>
        <tr>

            <td><%= quiz.getId() %></td>
            <td><%= quiz.getQuizTitle() %></td>
            <td><%= quiz.getCategory() %></td>
            <td><a href="updatequiz.jsp?id=<%= quiz.getId() %>">Update</a></td>
            <td><a href="deletequiz.jsp?id=<%= quiz.getId() %>">delete</a></td>
        </tr>
       <%
      }
    }
  %>
</table>
<div>

    <input type="button" class="btn btn-outline-danger" value="Back"
onclick="location.href='quizlist.jsp'">
    </div>
</body>
</html>
```

updateQuiznow

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ page import="com.model.CreateDao" %>
<%@ page import="com.db.Quizs" %>

<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Updating Quiz</title>
</head>
<body>
<%
    String newQuizTitle = request.getParameter("quizTitle");
    String newCategory = request.getParameter("category");
    String originalQuizId = request.getParameter("quizId");

    // Create a new Quizs object and set its properties
    Quizs quiz = new Quizs();
    quiz.setId(Integer.parseInt(originalQuizId));
    quiz.setQuizTitle(newQuizTitle);
    quiz.setCategory(newCategory);

    int status = CreateDao.updateQuiz(quiz);
    if (status > 0) {
        response.sendRedirect("quizlist.jsp");
    } else {
        out.print("Failed to update quiz information.");
    }
%>
</body>
</html>
```

Deletionquiz

```
<%-- Document: deletequiz --%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ page import="com.db.*" %>
<%@ page import="com.model.*" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Delete Quiz</title>
</head>
<body>
<%
    Quizs quiz = new Quizs();
    quiz.setId(Integer.parseInt(request.getParameter("quizId")));

    int status = CreateDao.deleteQuiz(quiz.getId());
    if (status > 0) {
        response.sendRedirect("quizlist.jsp");
    } else {
        out.print("Error deleting quiz.");
    }
%>
</body>
</html>
```

Questionlist

```
<%--Document : QuestionList--%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@page import="com.model.QuestionsDao"%>
<%@page import="com.db.*"%>
<%@page import="java.util.ArrayList"%>
<!DOCTYPE html>
<html>
<head>
```

```
    <meta charset="ISO-8859-1">
    <title>Question List</title>
</head>
<body>
    <h3 style="color: blue; text-align: center;">List of all questions</h3>
    <table border=2>
        <tr class="table-dark">
            <th>s no.</th>
            <th>Question</th>
            <th>Option A</th>
            <th>Option B</th>
            <th>Option C</th>
            <th>Option D</th>
            <th>Correct Answer</th>
            <th>Update</th>
            <th>Delete</th>
        </tr>
        <%-- Start loop to display questions --%>
        <%
            int i = 1;
            ArrayList<Questions> allQuestions = QuestionsDao.getAllRecords();
            for(Questions e : allQuestions) {
        %>
        <tr>
            <td><%= i++ %></td>
            <td><%= e.getQuestion() %></td>
            <td><%= e.getA() %></td>
            <td><%= e.getB() %></td>
            <td><%= e.getC() %></td>
            <td><%= e.getD() %></td>
            <td><%= e.getAnswer() %></td>
            <td><a href="updatequestion.jsp?ques=<%= e.getQuestion()
%>">Update</a></td>
            <td><a href="deletequestion.jsp?ques=<%= e.getQuestion()
%>">Delete</a></td>
        </tr>
        <%
            }
        %>
        <%-- End loop --%>
 </table>
        <tr>
            <th colspan="4" style="text-align: center;"><input type="button"
value="Add Question" class="btn btn-outline-success"
onclick="location.href='AddQuestion.jsp'"></th>
            <th><input type="hidden"></th>
            <th colspan="4" style="text-align: center;"><input type="button"
class="btn btn-outline-danger" value="Back to Panel"
onclick="location.href='AdminPanel.jsp'"></th>
        </tr>

</body>
</html>
```

updateQuestion

```
<%-- Document   : updatequestion --%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ page import="com.model.*" %>
<%@ page import="com.db.*" %>
<%@ page import="java.util.ArrayList" %>
<!DOCTYPE html>
<html>
<head>

  <meta charset="ISO-8859-1">
  <title>Update Questions</title>
</head>
<body>
<h3><font color="blue">List of all questions</font></h3>
<table border=2>
  <tr class="table-dark">
    <th>s no.</th>
    <th>Ques Id</th>
    <th>Question</th>
    <th>Option A</th>
    <th>Option B</th>
    <th>Option C</th>
    <th>Option D</th>
    <th>Correct Answer</th>
    <th>Update</th>
    <th>Delete</th>
  </tr>
  <%
    int i = 0;
    ArrayList<Questions> allQuestions = QuestionsDao.getAllRecords();
    for (Questions e : allQuestions) {
      if (request.getParameter("ques").equals(e.getQuestion())) {
  %>
        <tr class="table-primary">
        <form action="updatequestionnow.jsp">

          <td><input type="hidden" value="<%=e.getQuestion()%>" type="text"
name="quesoriginal"><%=e.getQuestion()%></td>
           <td><%=i++%></td>

       <td> <input type="text" value="<%=e.getQuestion()%>"
name="quesmodified"></input> </td>
          <td><input type="text" value="<%=e.getA()%>" name="opta"></td>
          <td><input type="text" value="<%=e.getB()%>" name="optb"></td>
          <td><input type="text" value="<%=e.getC()%>" name="optc"></td>
          <td><input type="text" value="<%=e.getD()%>" name="optd"></td>
          <td><input type="text" value="<%=e.getAnswer()%>" name="ans"></td>
          <td><input type="submit" value="update"></td>
          <td><a href="deletequestion.jsp?ques=<%=e.getQuestion()%>">Delete</a></td>
          </form>
        </tr>
  <%
      } else {
  %>
        <tr>
          <td><%=i++%></td>
          <td><%=e.getQuestion()%></td>
          <td><%=e.getA()%></td>
          <td><%=e.getB()%></td>
          <td><%=e.getC()%></td>
          <td><%=e.getD()%></td>
          <td><%=e.getAnswer()%></td>
          <td><a href="updatequestion.jsp?ques=<%=e.getQuestion()%>">Update</a></td>
          <td><a href="deletequestion.jsp?ques=<%=e.getQuestion()%>">Delete</a></td>
        </tr>
  <%
      }
    }
  %>
</table>
<div>
  <input type="button" class="btn btn-outline-success" value="Add Question"
onclick="location.href='AddQuestion.jsp'">
  <input type="hidden">
  <input type="button" class="btn btn-outline-danger" value="Back to Panel"
onclick="location.href='AdminPanel.jsp'">
</div>

</body>
</html>
```

updateQuestionnow

```
<%--Document   : updatequestionnow--%>
<%@page import="com.model.QuestionsDao"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Updating Questions</title>
</head>
<body>
<%
        String OriginalQuestion = request.getParameter("quesoriginal");
             String newQuestion = request.getParameter("quesmodified");
        String opta = request.getParameter("opta");
        String optb = request.getParameter("optb");
        String optc = request.getParameter("optc");
        String optd = request.getParameter("optd");
        String ans = request.getParameter("ans");
```

```
             int status
=QuestionsDao.doUpdateNowRecord(OriginalQuestion,newQuestion,opta,optb,optc,optd,ans)
;
             if(status > 0)
             {
                    response.sendRedirect("QuestionList.jsp");
             }
             else if(status == -1)
             {
                    out.print("error from databse");

             }
             else
             {
                    out.print("excepion occured");
             }

      %>
</body>
</html>
```

Deletionquestion

```
<%--Document : deletequestion--%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@page import="com.db.*" %>
    <%@page import="com.model.*" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Delete Questions</title>
</head>
<body>
<%
             Questions q = new Questions();
             q.setQuestion(request.getParameter("ques"));
             int status = QuestionsDao.deleteRecord(q);
             if(status > 0)
                    response.sendRedirect("QuestionList.jsp");
             else
                    out.print("Error");

      %>
</body>
</html>
```

UserLogin

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<style type="text/css">
       li {
             float: left;
       }
       ul {
             list-style-type: none;
             margin: 0;
             padding: 0;
             overflow: hidden;
             background-color:#333;
       }
       li a {
             display: inline-block;
             color: white;
             text-align: center;
             padding: 14px 16px;
             text-decoration: none;
       }
       li a:HOVER {
             background-color: aqua;
       }
</style>
<title>Student Login</title>
</head>
<body>
              <h1 style="color: blue; text-align: center;">User Login</h1>
               <font color="red">
             <%
                   if(request.getParameter("msg") != null)
                         out.print(request.getParameter("msg"));
             %>
       </font>
       <font color="green">
             <%
                   if(request.getParameter("msg1") != null)
                         out.print(request.getParameter("msg1"));
             %>
       </font>
       <form action="com.controller.ValidateUser" method="post">
             <pre>
                                                    User Id         :  <input
type="text" name="uname">  <br>
                                                    Password         : <input
type="password" name="pass">     <br>

                                                           <input
onclick="location.href='index.html'" type="button" value="Exit" class="btn btn-
outline-primary">    <input type="submit" value="Login" class="btn btn-outline-
primary">    <input type="button" onclick="location.href='AddUser.jsp'" value="Sign
Up" class="btn btn-outline-primary">
                                                           </pre>success">
```

```
        </form>
</body>
</html>
```