**MySQL CODE:**

**Creation of database:** create database onlinequizdb;

**Use database:** use onlinequizdb;

**Creation of Users table:**

create table users(user\_id int not null auto\_increment primary key, email varchar(100), password varchar(100));

**Creating admin:**

insert into users(email,password) values("admin@gmail.com","Test123@");

**Creation of Categories table:**

create table categories(category\_id int not null auto\_increment primary key, name varchar(100));

**Creation of quizzes table:**

create table quizes(

quiz\_id int not null auto\_increment primary key,

title varchar(100),

category int,

foreign key(category) references categories(category\_id)

);

**Creation of questions table:**

create table questions(

question\_id int not null auto\_increment primary key,

title varchar(500),

option1 varchar(100),

option2 varchar(100),

option3 varchar(100),

option4 varchar(100),

answer varchar(100),

category int,

foreign key(category) references categories(category\_id)

);

**Creation of scores table:**

create table scores(

score\_id int not null auto\_increment primary key,

score int,

quiz int,

user int,

foreign key(quiz) references quizes(quiz\_id),

foreign key(user) references users(user\_id)

);

**Java Code:**

**index.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**input** {

outline: *none*;

font-size: *18px*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**input**[type="submit"] {

width: *100%*;

padding: *0.75rem 1rem*;

color: *white*;

border: *1px solid green*;

border-radius: *0.3rem*;

cursor: *pointer*;

background-color: *green*;

}

**input**[type="submit"]*:hover* {

background-color: *white*;

color: *green*;

transition: *all 0.3s ease-in-out*;

}

</style>

</head>

<body style="min-height: *100vh*; display: *flex*; flex-direction: *column*; align-items: *center*;">

<%@include file=*"navbar.jsp"* %>

<div style="width: *100%*; padding: *2rem*; display: *flex*; flex-direction: *column*; align-items: *center*;">

<h1>Login</h1>

<form action=*"UserLoginServlet"* method=*"post"* style="width: *250px*; margin-top: *2rem*; display: *flex*; flex-direction: *column*; gap: *1rem*;">

<input type=*"email"* name=*"email"* placeholder=*"Email"*

style="width: *100%*; padding: *0.5rem 1rem*;" required />

<input type=*"password"* name=*"password"* placeholder=*"Password"*

style="width: *100%*; padding: *0.5rem 1rem*;" required />

<input type=*"submit"* value=*"Login"* />

</form>

</div>

</body>

</html>

**adminNavbar.jsp:**

<div style="position: *sticky*; top: *0*; left: *0*; height: *10vh*; width: *100%*; padding: *1rem*; color: *white*; display: *flex*; justify-content: *space-around*; align-items: *center*; background-color: *#27374D*; z-index: *100*;">

<h2>Quiz App</h2>

<ul style="list-style: *none*; display: *flex*; justify-content: *center*; align-items: *center*; gap: *1rem*;">

<li><a href=*"AdminDashboardServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Dashboard</a></li>

<li><a href=*"CategoryServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Category</a></li>

<li><a href=*"QuizServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Quiz</a></li>

<li><a href=*"QuestionServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Questions</a></li>

<li><a href=*"LogoutServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Logout</a></li>

</ul>

</div>

**admin/dashboard.jsp:**

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

**else** **if**(!user.getEmail().equals("admin@gmail.com")) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"adminNavbar.jsp"* %>

<div style="min-height: *100vh*; width: *100%*; display: *flex*; flex-direction: *column*; align-items: *center*; padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Quiz List</h1>

<%

List<Quiz> quizList = (List<Quiz>)request.getAttribute("quizes");

**if**(quizList == **null**) {

out.println("<p style='font-size: 20px;'>No Quiz to display!</p>");

}

**else** {

**if**(quizList.isEmpty()) {

out.println("<p style='font-size: 20px;'>No quiz to display!</p>");

}

**else** { %>

<table style="border-collapse: *collapse*;">

<thead>

<tr>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No.</th>

<th style="width: *40%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Quiz Title</th>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Category</th>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Action</th>

</tr>

</thead>

<tbody>

<% **for**(Quiz quiz : quizList) { %>

<tr>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getQuiz\_id() %>

</td>

<td

style="width: *40%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getTitle() %>

</td>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getCategory().getName() %>

</td>

<td style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<a href=*"#"* style="margin-right: *0.5rem*;">Edit</a>

<a href=*"#"*>Delete</a>

</td>

</tr>

<%} %>

</tbody>

</table>

<%}}%>

</div>

</div>

</body>

</html>

**userNavbar.jsp:**

<div style="position: *sticky*; top: *0*; left: *0*; height: *10vh*; width: *100%*; padding: *1rem*; color: *white*; display: *flex*; justify-content: *space-around*; align-items: *center*; background-color: *#27374D*; z-index: *100*;">

<h2>Quiz App</h2>

<ul style="list-style: *none*; display: *flex*; justify-content: *center*; align-items: *center*; gap: *1rem*;">

<li><a href=*"UserDashboardServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Dashboard</a></li>

<li><a href=*"LogoutServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">Logout</a></li>

</ul>

</div>

**user/dashboard.jsp:**

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

*#startQuiz* {

text-decoration: *none*;

padding: *0.5rem*;

color: *white*;

border: *1px solid green*;

background-color: *green*;

}

*#startQuiz:hover* {

color: *green*;

background-color: *white*;

transition: *all 0.2s ease-in-out*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"userNavbar.jsp"* %>

<div style="min-height: *100vh*; width: *100%*; display: *flex*; flex-direction: *column*; align-items: *center*; padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Ongoing Quizes</h1>

<%

List<Quiz> quizList = (List<Quiz>)request.getAttribute("quizes");

**if**(quizList == **null**) {

out.println("<p style='font-size: 20px;'>No Quiz to display!</p>");

}

**else** {

**if**(quizList.isEmpty()) {

out.println("<p style='font-size: 20px;'>No Quiz to display!</p>");

}

**else** { %>

<table style="border-collapse: *collapse*;">

<thead>

<tr>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No.</th>

<th style="width: *40%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Quiz Title</th>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Category</th>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Action</th>

</tr>

</thead>

<tbody>

<% **for**(Quiz quiz : quizList) { %>

<% **int** index = quizList.indexOf(quiz)+1; %>

<tr>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= index %>

</td>

<td

style="width: *40%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getTitle() %>

</td>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getCategory().getName() %>

</td>

<td style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<a id=*"startQuiz"* href=*"AttemptQuizServlet?id=*<%= quiz.getQuiz\_id() %>*"*>

Attempt

</a>

</td>

</tr>

<%} %>

</tbody>

</table>

<%}}%>

</div>

</div>

</body>

</html>

**attemptQuiz.jsp:**

<%@page import=*"org.hibernate.internal.build.AllowSysOut"*%>

<%@page import=*"com.to.Question"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**b** {

font-size: *22px*;

}

*#play* {

text-decoration: *none*;

width: *100px*;

padding: *0.5rem*;

font-size: *18px*;

color: *white*;

text-align: *center*;

background-color: *green*;

cursor: *pointer*;

border: *1px solid green*;

}

*#play:hover* {

color: *green*;

background-color: *white*;

transition: *all 0.2s ease-in-out*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"user/userNavbar.jsp"* %>

<%

Quiz quiz = (Quiz)request.getAttribute("quiz");

List<Question> questions = (List<Question>)request.getAttribute("questions");

%>

<div style="min-height: *100vh*; width: *100%*; display: *flex*; flex-direction: *column*; align-items: *center*; gap: *1rem*; padding: *2rem*;">

<h1><%= quiz.getTitle() %></h1>

<b>Total Questions: <%= questions.size() %></b>

<b>Total Marks: <%= questions.size()\*1 %></b>

<a id=*"play"* href=*"StartQuizServlet?id=*<%= quiz.getQuiz\_id() %>*&offset=*<%= 0 %>*"*>Start</a>

</div>

</body>

</html>

**category.jsp:**

<%@page import=*"com.to.Category"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**input** {

outline: *none*;

font-size: *18px*;

}

**input**[type="submit"] {

width: *100px*;

padding: *0.5rem*;

color: *white*;

border: *1px solid green*;

border-radius: *0.3rem*;

background-color: *green*;

cursor: *pointer*;

}

**input**[type="submit"]*:hover* {

color: *green*;

background-color: *white*;

transistion: *all 0.2s ease-in-out*;

}

**label** {

font-size: *18px*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

**else** **if**(!user.getEmail().equals("admin@gmail.com")) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"admin/adminNavbar.jsp"* %>

<div

style="min-height: *100vh*; width: *100%*;

display: *flex*; flex-direction: *column*; align-items: *center*;

padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Create New Category</h1>

<form action=*"AddCategoryServlet"* method=*"post"* style="width: *100%*; display: *flex*; flex-direction: *column*; align-items: *start*; gap: *1rem*;">

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"name"*>Category Name</label>

<input type=*"text"* name=*"name"* id=*"name"* placeholder=*"Enter category name"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<input type=*"submit"* value=*"Submit"* />

</form>

</div>

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Categories</h1>

<%

List<Category> categoryList = (List<Category>)request.getAttribute("categoryList");

**if**(categoryList == **null**) {

out.println("<p style='font-size: 20px;'>No category to display!</p>");

}

**else** {

**if**(categoryList.isEmpty()) {

out.println("<p style='font-size: 20px;'>No category to display!</p>");

}

**else** {%>

<table style="border-collapse: *collapse*;">

<thead>

<tr>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No.</th>

<th style="width: *50%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Category</th>

<th style="width: *30%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Action</th>

</tr>

</thead>

<tbody>

<% **for**(Category category : categoryList) { %>

<% **int** sno = categoryList.indexOf(category)+1; %>

<tr>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= sno %>

</td>

<td

style="width: *50%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= category.getName() %>

</td>

<td style="width: *30%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<a href=*"DisplayCategoryServlet?id=*<%= category.getCategory\_id() %>*"* style="margin-right: *0.5rem*;">Edit</a>

<a href=*"DeleteCategoryServlet?id=*<%= category.getCategory\_id() %>*"*>Delete</a>

</td>

</tr>

<%} %>

</tbody>

</table>

<%}}%>

</div>

</div>

</body>

</html>

**displayCategory.jsp:**

<%@page import=*"com.to.Category"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**input** {

outline: *none*;

font-size: *18px*;

}

**input**[type="submit"] {

width: *100%*;

padding: *0.5rem*;

color: *white*;

border: *1px solid green*;

border-radius: *0.3rem*;

background-color: *green*;

cursor: *pointer*;

}

**input**[type="submit"]*:hover* {

color: *green*;

background-color: *white*;

transistion: *all 0.2s ease-in-out*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

**else** **if**(!user.getEmail().equals("admin@gmail.com")) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"admin/adminNavbar.jsp"* %>

<%Category category = (Category)request.getAttribute("category"); %>

<div

style="min-height: *100vh*; width: *100%*;

display: *flex*; flex-direction: *column*; align-items: *center*;

padding: *2rem*;">

<h1>Edit Category</h1>

<form action=*"UpdateCategoryServlet"* method=*"post"*

style="width: *250px*; margin-top: *1rem*; display: *flex*; flex-direction: *column*; gap: *1rem*;"

>

<input type=*"hidden"* name=*"id"* value=*"*<%= category.getCategory\_id() %>*"* />

<input type=*"text"* name=*"name"* value=*"*<%= category.getName() %>*"* placeholder=*"Category Name"*

style="width: *100%*; padding:*0.5rem 1rem*;" required

/>

<input type=*"submit"* value=*"Edit"* />

</form>

</div>

</body>

</html>

**leaderboard.jsp:**

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"com.to.Score"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"user/userNavbar.jsp"* %>

<%

List<Score> scores = (List<Score>)request.getAttribute("scores");

Quiz quiz = (Quiz)request.getAttribute("quiz");

%>

<div style="min-height: *100vh*; width: *100%*; display: *flex*; flex-direction: *column*; align-items: *center*; padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Quiz Name:<%= quiz.getTitle() %></h1>

<h1>Category:<%= quiz.getCategory().getName() %></h1>

<% **if**(scores.isEmpty()) { %>

<p style="font-size: *20px*;">No score to display!</p>

<% } **else** { %>

<table style="width: *100%*; border-collapse: *collapse*;">

<thead>

<tr>

<th style="padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No</th>

<th style="padding: *0.7rem*; border: *1px solid lightgrey*;">User</th>

<th style="padding: *0.7rem*; border: *1px solid lightgrey*;">Score</th>

</tr>

</thead>

<tbody>

<% **for**(Score score:scores) { %>

<tr>

<td style="text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= scores.indexOf(score)+1 %></td>

<td style="text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= score.getUser().getEmail() %></td>

<td style="text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= score.getScore() %></td>

</tr>

<% } %>

</tbody>

</table>

<% } %>

</div>

</div>

</body>

</html>

**navbar.jsp:**

<div style="position: *sticky*; top: *0*; left: *0*; height: *10vh*; width: *100%*; padding: *1rem*; color: *white*; display: *flex*; justify-content: *space-around*; align-items: *center*; background-color: *#27374D*; z-index: *100*;">

<h2>Quiz App</h2>

<ul>

<li style="list-style: *none*;">

<a href=*"UserLoginServlet"* style="text-decoration: *none*; color: *white*; font-size: *20px*;">

Register/Login

</a>

</li>

</ul>

</div>

**playQuiz.jsp:**

<%@page import=*"org.hibernate.internal.build.AllowSysOut"*%>

<%@page import=*"com.services.QuizServices"*%>

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"com.to.User"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.Question"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**input** {

outline: *none*;

font-size: *18px*;

}

**label** {

font-size: *18px*;

}

*#submit* {

text-decoration: *none*;

width: *100px*;

padding: *0.5rem*;

font-size: *18px*;

color: *green*;

text-align: *center*;

background-color: *white*;

cursor: *pointer*;

border: *1px solid green*;

}

*#submit:hover* {

color: *white*;

background-color: *green*;

transition: *all 0.2s ease-in-out*;

}

*#next* {

text-decoration: *none*;

width: *100px*;

padding: *0.5rem*;

font-size: *18px*;

color: *white*;

text-align: *center*;

background-color: *green*;

cursor: *pointer*;

border: *1px solid green*;

}

*#next:hover* {

color: *green*;

background-color: *white*;

transition: *all 0.2s ease-in-out*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"user/userNavbar.jsp"* %>

<%

Question question = (Question)request.getAttribute("question");

Quiz quiz = (Quiz)request.getAttribute("quiz");

List<Question> questions = (List<Question>)request.getAttribute("questions");

**int** index = -1;

**for**(Question q:questions) {

**if**(q.getQuestion\_id() == question.getQuestion\_id()) {

index = questions.indexOf(q)+1;

}

}

%>

<div style="min-height: *100vh*; width: *100%*; display: *flex*; flex-direction: *column*; align-items: *center*; padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Q<%= index %>. <%= question.getTitle() %></h1>

<b style="font-size: *20px*;">Options</b>

<form action=*"ValidateAnswerServlet"* method=*"post"* style=" display: *flex*; flex-direction: *column*; gap: *1rem*;">

<div style="width: *100%*; display: *flex*; justify-content: *start*; align-items: *center*; gap: *0.3rem*;">

<input type=*"checkbox"* name=*"option1"* id=*"option1"* value=*"*<%= question.getOption1() %>*"* />

<label for=*"option1"*><%= question.getOption1() %></label>

</div>

<input type=*"hidden"* name=*"quiz\_id"* value=*"*<%= quiz.getQuiz\_id() %>*"* />

<input type=*"hidden"* name=*"questionid"* value=*"*<%= question.getQuestion\_id() %>*"* />

<div style="width: *100%*; display: *flex*; justify-content: *start*; align-items: *center*; gap: *0.3rem*;">

<input type=*"checkbox"* name=*"option2"* id=*"option2"* value=*"*<%= question.getOption2() %>*"* />

<label for=*"option2"*><%= question.getOption2() %></label>

</div>

<div style="width: *100%*; display: *flex*; justify-content: *start*; align-items: *center*; gap: *0.3rem*;">

<input type=*"checkbox"* name=*"option3"* id=*"option3"* value=*"*<%= question.getOption3() %>*"* />

<label for=*"option3"*><%= question.getOption3() %></label>

</div>

<div style="width: *100%*; display: *flex*; justify-content: *start*; align-items: *center*; gap: *0.3rem*;">

<input type=*"checkbox"* name=*"option4"* id=*"option4"* value=*"*<%= question.getOption4() %>*"* />

<label for=*"option4"*><%= question.getOption4() %></label>

</div>

<div style="width: *100%*; display: *flex*; justify-content: *start*; align-items: *center*; gap: *1rem*;">

<input type=*"submit"* id=*"submit"* name=*"submit"* value=*"Submit"* />

</div>

</form>

</div>

</div>

</body>

</html>

**questions.jsp:**

<%@page import=*"com.to.Question"*%>

<%@page import=*"com.to.Category"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**input** {

outline: *none*;

font-size: *18px*;

}

**input**[type="submit"] {

width: *100px*;

padding: *0.5rem*;

color: *white*;

border: *1px solid green*;

border-radius: *0.3rem*;

background-color: *green*;

cursor: *pointer*;

}

**input**[type="submit"]*:hover* {

color: *green*;

background-color: *white*;

transistion: *all 0.2s ease-in-out*;

}

**label** {

font-size: *18px*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

**else** **if**(!user.getEmail().equals("admin@gmail.com")) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"admin/adminNavbar.jsp"* %>

<%

List<Category> categoryList = (List<Category>)request.getAttribute("categoryList");

List<Question> questions = (List<Question>)request.getAttribute("questions");

%>

<div

style="min-height: *100vh*; width: *100%*;

display: *flex*; flex-direction: *column*; align-items: *center*;

padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Create New Question</h1>

<form action=*"AddQuestionServlet"* method=*"post"* style="width: *100%*; display: *flex*; flex-direction: *column*; align-items: *start*; gap: *1rem*;">

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"title"*>Question</label>

<input type=*"text"* name=*"title"* id=*"title"* placeholder=*"Enter question"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"option1"*>Option-1</label>

<input type=*"text"* name=*"option1"* id=*"option1"* placeholder=*"Option 1"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"option2"*>Option-2</label>

<input type=*"text"* name=*"option2"* id=*"option2"* placeholder=*"Option 2"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"option3"*>Option-3</label>

<input type=*"text"* name=*"option3"* id=*"option3"* placeholder=*"Option 3"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"option4"*>Option-4</label>

<input type=*"text"* name=*"option4"* id=*"option4"* placeholder=*"Option 4"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"answer"*>Answer</label>

<input type=*"text"* name=*"answer"* id=*"answer"* placeholder=*"Answer"*

style="padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="width: *100%*; display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"category"*>Category</label>

<select name=*"category"* id=*"category"* style="padding: *0.5rem*;">

<% **if**(categoryList.size() == 0) { %>

<option>No category to display</option>

<%}%>

<% **for**(Category category : categoryList) { %>

<option value=*"*<%= category.getCategory\_id() %>*"*><%= category.getName() %></option>

<%} %>

</select>

</div>

<input type=*"submit"* value=*"Submit"* />

</form>

</div>

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Questions</h1>

<%

**if**(categoryList.isEmpty()) {

out.println("<p style='font-size: 20px;'>No category to display!</p>");

}

**else** {%>

<table style="width: *100%*; border-collapse: *collapse*;">

<thead>

<tr>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No.</th>

<th style="width: *30%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Question</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-1</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-2</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-3</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-4</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Answer</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Action</th>

</tr>

</thead>

<tbody>

<% **for**(Question question : questions) { %>

<tr>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= questions.indexOf(question)+1 %></td>

<td style="width: *30%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getTitle() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption1() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption2() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption3() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption4() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getAnswer() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;">

<a href=*"#"*>Edit</a>

<a href=*"#"*>Delete</a>

</td>

</tr>

<%}%>

</tbody>

</table>

<%}%>

</div>

</div>

</body>

</html>

**quiz.jsp:**

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.Category"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

**input** {

outline: *none*;

font-size: *18px*;

}

**input**[type="submit"] {

width: *100px*;

padding: *0.5rem*;

color: *white*;

border: *1px solid green*;

border-radius: *0.3rem*;

background-color: *green*;

cursor: *pointer*;

}

**input**[type="submit"]*:hover* {

color: *green*;

background-color: *white*;

transistion: *all 0.2s ease-in-out*;

}

**label** {

font-size: *18px*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

**else** **if**(!user.getEmail().equals("admin@gmail.com")) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"admin/adminNavbar.jsp"* %>

<%

List<Quiz> quizes = (List<Quiz>)request.getAttribute("quizes");

List<Category> categoryList = (List<Category>)request.getAttribute("categoryList");

%>

<div

style="min-height: *100vh*; width: *100%*;

display: *flex*; flex-direction: *column*; align-items: *center*;

padding: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Create New Quiz</h1>

<form action=*"AddQuizServlet"* method=*"post"* style="width: *30%*; display: *flex*; flex-direction: *column*; align-items: *start*; gap: *1rem*;">

<div style="width: *100%*; display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"title"*>Quiz Title</label>

<input type=*"text"* name=*"title"* id=*"title"* placeholder=*"Quiz title"*

style="width: *100%*; padding: *0.5rem 1rem*;"

required

/>

</div>

<div style="width: *100%*; display: *flex*; flex-direction: *column*; align-items: *start*; gap: *0.3rem*;">

<label for=*"category"*>Category</label>

<select name=*"category"* id=*"category"* style="width: *100%*; padding: *0.5rem*;">

<% **if**(categoryList.size() == 0) { %>

<option>No category to display</option>

<%}%>

<% **for**(Category category : categoryList) { %>

<option value=*"*<%= category.getCategory\_id() %>*"*><%= category.getName() %></option>

<%} %>

</select>

</div>

<input type=*"submit"* value=*"Submit"* />

</form>

</div>

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1.5rem*;">

<h1>Quiz List</h1>

<%

**if**(quizes.isEmpty()) {

out.println("<p style='font-size: 20px;'>No quiz to display!</p>");

}

**else** {%>

<table style="border-collapse: *collapse*;">

<thead>

<tr>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No.</th>

<th style="width: *40%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Quiz Title</th>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Category</th>

<th style="width: *20%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Action</th>

</tr>

</thead>

<tbody>

<% **for**(Quiz quiz : quizes) { %>

<% **int** sno = quizes.indexOf(quiz)+1; %>

<tr>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= sno %>

</td>

<td

style="width: *40%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getTitle() %>

</td>

<td

style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<%= quiz.getCategory().getName() %>

</td>

<td style="width: *20%*; padding: *0.7rem*;

text-align: *center*; border: *1px solid lightgrey*;">

<a href=*"DisplayCategoryServlet?id=*<%= quiz.getQuiz\_id() %>*"* style="margin-right: *0.5rem*;">Edit</a>

<a href=*"DeleteCategoryServlet?id=*<%= quiz.getQuiz\_id() %>*"*>Delete</a>

</td>

</tr>

<%} %>

</tbody>

</table>

<%}%>

</div>

</div>

</body>

</html>

**quizDetail.jsp:**

<%@page import=*"com.to.Category"*%>

<%@page import=*"com.to.Question"*%>

<%@page import=*"java.util.List"*%>

<%@page import=*"com.to.Quiz"*%>

<%@page import=*"com.to.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Quiz App</title>

<style>

\* {

margin: *0*;

padding: *0*;

box-sizing: *border-box*;

}

**html** {

position: *relative*;

min-height: *100vh*;

width: *100%*;

}

</style>

</head>

<body>

<%

User user = (User)session.getAttribute("user");

**if**(user == **null**) {

response.sendRedirect("index.jsp");

}

**else** **if**(!user.getEmail().equals("admin@gmail.com")) {

response.sendRedirect("index.jsp");

}

%>

<%@include file=*"admin/adminNavbar.jsp"* %>

<%

Quiz quiz = (Quiz)request.getAttribute("quiz");

Category category = quiz.getCategory();

List<Question> questions = (List<Question>)request.getAttribute("questions");

%>

<div style="width: *100%*; padding:*2rem*; display: *flex*; flex-direction: *column*; align-items: *center*; gap: *2rem*;">

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1rem*;">

<h1>Quiz Details</h1>

<table style="width: *100%*; border-collapse: *collapse*;">

<thead>

<tr>

<th style="padding: *0.7rem*; border: *1px solid lightgrey*;">Quiz Title</th>

<th style="padding: *0.7rem*; border: *1px solid lightgrey*;">Category</th>

</tr>

</thead>

<tbody>

<tr>

<td style="text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= quiz.getTitle() %></td>

<td style="text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= category.getName() %></td>

</tr>

</tbody>

</table>

</div>

<div style="width: *75%*; padding: *1rem*; display: *flex*; flex-direction: *column*; gap: *1rem*;">

<h1>Questions List</h1>

<% **if**(questions.isEmpty()) { %>

<p style="font-size: *18px*;">No questions to display!</p>

<%} **else** {%>

<table style="width: *100%*; border-collapse: *collapse*;">

<thead>

<tr>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Sl No.</th>

<th style="width: *30%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Question</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-1</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-2</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-3</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Option-4</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Answer</th>

<th style="width: *10%*; padding: *0.7rem*; border: *1px solid lightgrey*;">Action</th>

</tr>

</thead>

<tbody>

<% **for**(Question question : questions) { %>

<tr>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= questions.indexOf(question)+1 %></td>

<td style="width: *30%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getTitle() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption1() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption2() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption3() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getOption4() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;"><%= question.getAnswer() %></td>

<td style="width: *10%*; text-align: *center*; padding: *0.7rem*; border: *1px solid lightgrey*;">

<a href=*"#"*>Edit</a>

<a href=*"#"*>Delete</a>

</td>

</tr>

<%}%>

</tbody>

</table>

<% } %>

</div>

</div>

</body>

</html>

**config.properties:**

driverName=com.mysql.cj.jdbc.Driver

connectionString=jdbc:mysql://localhost:3306/onlinequizdb

username=root

password=password

**DBConnection.java:**

package com.connection;

import java.io.FileReader;

import java.sql.Connection;

import java.sql.DriverManager;

import java.util.Properties;

public class DBConnection {

private static String driverName, connectionString, username, password;

public static Connection connectToDB() {

Connection conObj = null;

try {

String app\_path = "C:\\Users\\tusha\\eclipse-workspace\\OnlineQuizApp";

app\_path = app\_path + "\\src\\config.properties";

FileReader file = new FileReader(app\_path);

Properties properties = new Properties();

properties.load(file);

driverName = properties.getProperty("driverName");

connectionString = properties.getProperty("connectionString");

username = properties.getProperty("username");

password = properties.getProperty("password");

Class.forName(driverName);

conObj = DriverManager.getConnection(connectionString, username, password);

if(conObj != null) {

System.out.println("Connected to DB......");

}

} catch (Exception e) {

e.printStackTrace();

}

return conObj;

}

}

**User.java:**

**package** com.to;

**public** **class** User {

**private** String email;

**public** String getEmail() {

**return** email;

}

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

**this**.email = email;

}

@Override

**public** String toString() {

**return** "User [email=" + email +"]";

}

}

**Category.java:**

**package** com.to;

**public** **class** Category {

**private** **int** category\_id;

**private** String name;

**public** **int** getCategory\_id() {

**return** category\_id;

}

**public** **void** setCategory\_id(**int** category\_id) {

**this**.category\_id = category\_id;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

@Override

**public** String toString() {

**return** "Category [category\_id=" + category\_id + ", name=" + name + "]";

}

}

**Question.java:**

**package** com.to;

**public** **class** Question {

**private** **int** question\_id;

**private** String title;

**private** String option1;

**private** String option2;

**private** String option3;

**private** String option4;

**private** String answer;

**private** Category category;

**public** **int** getQuestion\_id() {

**return** question\_id;

}

**public** **void** setQuestion\_id(**int** question\_id) {

**this**.question\_id = question\_id;

}

**public** String getTitle() {

**return** title;

}

**public** **void** setTitle(String title) {

**this**.title = title;

}

**public** String getOption1() {

**return** option1;

}

**public** **void** setOption1(String option1) {

**this**.option1 = option1;

}

**public** String getOption2() {

**return** option2;

}

**public** **void** setOption2(String option2) {

**this**.option2 = option2;

}

**public** String getOption3() {

**return** option3;

}

**public** **void** setOption3(String option3) {

**this**.option3 = option3;

}

**public** String getOption4() {

**return** option4;

}

**public** **void** setOption4(String option4) {

**this**.option4 = option4;

}

**public** String getAnswer() {

**return** answer;

}

**public** **void** setAnswer(String answer) {

**this**.answer = answer;

}

**public** Category getCategory() {

**return** category;

}

**public** **void** setCategory(Category category) {

**this**.category = category;

}

@Override

**public** String toString() {

**return** "Question [question\_id=" + question\_id + ", title=" + title + ", option1=" + option1 + ", option2="

+ option2 + ", option3=" + option3 + ", option4=" + option4 + ", answer=" + answer + ", category="

+ category + "]";

}

}

**Quiz.java:**

**package** com.to;

**public** **class** Quiz {

**private** **int** quiz\_id;

**private** String title;

**private** Category category;

**public** **int** getQuiz\_id() {

**return** quiz\_id;

}

**public** **void** setQuiz\_id(**int** quiz\_id) {

**this**.quiz\_id = quiz\_id;

}

**public** String getTitle() {

**return** title;

}

**public** **void** setTitle(String title) {

**this**.title = title;

}

**public** Category getCategory() {

**return** category;

}

**public** **void** setCategory(Category category) {

**this**.category = category;

}

@Override

**public** String toString() {

**return** "Quiz [quiz\_id=" + quiz\_id + ", title=" + title + ", category=" + category + "]";

}

}

**Score.java:**

**package** com.to;

**public** **class** Score {

**private** **int** score\_id;

**private** **int** score;

**private** Quiz quiz;

**private** User user;

**public** **int** getScore\_id() {

**return** score\_id;

}

**public** **void** setScore\_id(**int** score\_id) {

**this**.score\_id = score\_id;

}

**public** **int** getScore() {

**return** score;

}

**public** **void** setScore(**int** score) {

**this**.score = score;

}

**public** Quiz getQuiz() {

**return** quiz;

}

**public** **void** setQuiz(Quiz quiz) {

**this**.quiz = quiz;

}

**public** User getUser() {

**return** user;

}

**public** **void** setUser(User user) {

**this**.user = user;

}

@Override

**public** String toString() {

**return** "Score [score\_id=" + score\_id + ", score=" + score + ", quiz=" + quiz + ", user=" + user + "]";

}

}

**QuizServices.java:**

package com.services;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

import java.util.List;

import com.connection.DBConnection;

import com.to.Category;

import com.to.Question;

import com.to.Quiz;

import com.to.Score;

import com.to.User;

public class QuizServices {

private Connection conObj = null;

public QuizServices() {

this.conObj = DBConnection.connectToDB();

}

public List<Quiz> fetchQuizes() {

List<Quiz> quizList = new ArrayList<Quiz>();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from quizes");

ResultSet quizRes = ps.executeQuery();

while(quizRes.next()) {

Quiz quiz = new Quiz();

quiz.setQuiz\_id(quizRes.getInt("quiz\_id"));

quiz.setTitle(quizRes.getString("title"));

ps = conObj.prepareStatement("select \* from categories where category\_id=?");

ps.setInt(1, quizRes.getInt("category"));

ResultSet isCategory = ps.executeQuery();

if(isCategory.next()) {

Category category = new Category();

category.setCategory\_id(isCategory.getInt("category\_id"));

category.setName(isCategory.getString("name"));

quiz.setCategory(category);

quizList.add(quiz);

}

else {

System.out.println("Invalid category");

}

}

} catch (Exception e) {

e.printStackTrace();

}

return quizList;

}

public List<Category> fetchCategories() {

List<Category> categoryList = new ArrayList<Category>();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from categories");

ResultSet categoryRes = ps.executeQuery();

while(categoryRes.next()) {

Category category = new Category();

category.setCategory\_id(categoryRes.getInt("category\_id"));

category.setName(categoryRes.getString("name"));

categoryList.add(category);

}

} catch (Exception e) {

e.printStackTrace();

}

return categoryList;

}

public boolean addCategory(String name) {

boolean result = false;

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("insert into categories(name) values(?)");

ps.setString(1, name);

int res = ps.executeUpdate();

if(res >= 1) {

result = true;

}

} catch (Exception e) {

e.printStackTrace();

}

return result;

}

public List<Category> deleteCategory(int id) {

List<Category> categoryList = fetchCategories();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("delete from categories where category\_id=?");

ps.setInt(1, id);

int res = ps.executeUpdate();

if(res >= 1) {

categoryList = fetchCategories();

}

} catch (Exception e) {

e.printStackTrace();

}

return categoryList;

}

public Category getCategory(int id) {

Category category = new Category();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from categories where category\_id=?");

ps.setInt(1, id);

ResultSet isCategory = ps.executeQuery();

if(isCategory.next()) {

category.setCategory\_id(isCategory.getInt("category\_id"));

category.setName(isCategory.getString("name"));

}

else {

category = null;

}

} catch (Exception e) {

e.printStackTrace();

}

return category;

}

public List<Category> editCategory(int id, String name) {

List<Category> categoryList = fetchCategories();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("update categories set name=? where category\_id=?");

ps.setString(1, name);

ps.setInt(2, id);

int res = ps.executeUpdate();

if(res >= 1) {

categoryList = fetchCategories();

}

} catch (Exception e) {

e.printStackTrace();

}

return categoryList;

}

public Quiz addQuiz(String title, int category) {

Quiz quiz = new Quiz();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("insert into quizes(title,category) values(?,?)");

ps.setString(1, title);

ps.setInt(2, category);

int res = ps.executeUpdate();

if(res >= 1) {

ps = conObj.prepareStatement("select \* from quizes order by quiz\_id desc limit 1");

ResultSet isQuiz = ps.executeQuery();

if(isQuiz.next()) {

quiz.setQuiz\_id(isQuiz.getInt("quiz\_id"));

quiz.setTitle(isQuiz.getString("title"));

Category cat = new Category();

ps= conObj.prepareStatement("select \* from categories where category\_id=?");

ps.setInt(1, category);

ResultSet isCategory = ps.executeQuery();

if(isCategory.next()) {

cat.setCategory\_id(isCategory.getInt("category\_id"));

cat.setName(isCategory.getString("name"));

quiz.setCategory(cat);

}

}

}

else {

quiz = null;

}

} catch (Exception e) {

e.printStackTrace();

}

return quiz;

}

public List<Question> fetchQuestionsByCategory(int category) {

List<Question> questions = new ArrayList<Question>();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from questions where category=?");

ps.setInt(1, category);

ResultSet isQuestions = ps.executeQuery();

while(isQuestions.next()) {

Question question = new Question();

question.setQuestion\_id(isQuestions.getInt("question\_id"));

question.setTitle(isQuestions.getString("title"));

question.setOption1(isQuestions.getString("option1"));

question.setOption2(isQuestions.getString("option2"));

question.setOption3(isQuestions.getString("option3"));

question.setOption4(isQuestions.getString("option4"));

question.setAnswer(isQuestions.getString("answer"));

ps = conObj.prepareStatement("select \* from categories where category\_id=?");

ps.setInt(1, category);

ResultSet isCategory = ps.executeQuery();

if(isCategory.next()) {

Category cat = new Category();

cat.setCategory\_id(isCategory.getInt("category\_id"));

cat.setName(isCategory.getString("name"));

question.setCategory(cat);

questions.add(question);

}

}

} catch (Exception e) {

e.printStackTrace();

}

return questions;

}

public List<Question> fetchAllQuestions() {

List<Question> questions = new ArrayList<Question>();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from questions");

ResultSet isQuestions = ps.executeQuery();

while(isQuestions.next()) {

Question question = new Question();

question.setQuestion\_id(isQuestions.getInt("question\_id"));

question.setTitle(isQuestions.getString("title"));

question.setOption1(isQuestions.getString("option1"));

question.setOption2(isQuestions.getString("option2"));

question.setOption3(isQuestions.getString("option3"));

question.setOption4(isQuestions.getString("option4"));

question.setAnswer(isQuestions.getString("answer"));

ps = conObj.prepareStatement("select \* from categories where category\_id=?");

ps.setInt(1, isQuestions.getInt("category"));

ResultSet isCategory = ps.executeQuery();

if(isCategory.next()) {

Category cat = new Category();

cat.setCategory\_id(isCategory.getInt("category\_id"));

cat.setName(isCategory.getString("name"));

question.setCategory(cat);

questions.add(question);

}

}

} catch (Exception e) {

e.printStackTrace();

}

return questions;

}

public List<Question> addQuestion(Question question) {

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("insert into questions(title,option1,option2,option3,option4,answer,category) values(?,?,?,?,?,?,?)");

ps.setString(1, question.getTitle());

ps.setString(2, question.getOption1());

ps.setString(3, question.getOption2());

ps.setString(4, question.getOption3());

ps.setString(5, question.getOption4());

ps.setString(6, question.getAnswer());

ps.setInt(7, question.getCategory().getCategory\_id());

int res = ps.executeUpdate();

if(res >= 1) {

System.out.println("Question added......");

}

} catch (Exception e) {

e.printStackTrace();

}

List<Question> questions = fetchAllQuestions();

return questions;

}

public Quiz fetchQuizById(int id) {

Quiz quiz = new Quiz();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from quizes where quiz\_id=?");

ps.setInt(1, id);

ResultSet isQuiz = ps.executeQuery();

if(isQuiz.next()) {

quiz.setQuiz\_id(isQuiz.getInt("quiz\_id"));

quiz.setTitle(isQuiz.getString("title"));

Category category = getCategory(isQuiz.getInt("category"));

quiz.setCategory(category);

}

} catch (Exception e) {

e.printStackTrace();

}

System.out.println("question service");

return quiz;

}

public Question getQuestionByCategory(int id, int offset) {

Question question = new Question();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from questions where category=? limit 1 offset ?");

ps.setInt(1, id);

ps.setInt(2, offset);

ResultSet isQuestion = ps.executeQuery();

if(isQuestion.next()) {

question.setQuestion\_id(isQuestion.getInt("question\_id"));

question.setTitle(isQuestion.getString("title"));

question.setOption1(isQuestion.getString("option1"));

question.setOption2(isQuestion.getString("option2"));

question.setOption3(isQuestion.getString("option3"));

question.setOption4(isQuestion.getString("option4"));

question.setAnswer(isQuestion.getString("answer"));

ps = conObj.prepareStatement("select \* from categories where category\_id=?");

ps.setInt(1, isQuestion.getInt("category"));

ResultSet isCategory = ps.executeQuery();

if(isCategory.next()) {

Category cat = new Category();

cat.setCategory\_id(isCategory.getInt("category\_id"));

cat.setName(isCategory.getString("name"));

question.setCategory(cat);

}

}

} catch (Exception e) {

e.printStackTrace();

}

return question;

}

public boolean submitAnswer(int quizId, int qid, String answer, String email) {

int userId = 0;

boolean result = false;

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from users where email=?");

ps.setString(1, email);

ResultSet isUser = ps.executeQuery();

if(isUser.next()) {

userId = isUser.getInt("user\_id");

}

Question question = getQuestion(qid);

if(question != null) {

System.out.println("question in service: "+question);

// System.out.println("isCorrect service: "+answer.equals(question.getAnswer()));

if(answer.equals(question.getAnswer())) {

ps = conObj.prepareStatement("select \* from scores where user=? and quiz=?");

ps.setInt(1, userId);

ps.setInt(2, quizId);

ResultSet isScore = ps.executeQuery();

if(isScore.next()) {

ps = conObj.prepareStatement("update scores set score=? where score\_id=?");

ps.setInt(1, isScore.getInt("score")+1);

ps.setInt(2, isScore.getInt("score\_id"));

int res = ps.executeUpdate();

if(res >= 1) {

result = true;

System.out.println("Correct answer! Score updated......");

}

}

else {

System.out.println("yes");

PreparedStatement ps1 = conObj.prepareStatement("insert into scores(score,quiz,user) values(?,?,?)");

ps1.setInt(1, 1);

ps1.setInt(2, quizId);

ps1.setInt(3, userId);

int res = ps1.executeUpdate();

if(res >= 1) {

result = true;

System.out.println("Correct answer! Score updated......");

}

}

}

}

} catch (Exception e) {

e.printStackTrace();

}

return result;

}

public List<Score> getScores(int qid) {

List<Score> scores = new ArrayList<Score>();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from scores where quiz=? order by score desc");

ps.setInt(1, qid);

ResultSet isScores = ps.executeQuery();

while(isScores.next()) {

Score score = new Score();

score.setScore\_id(isScores.getInt("score\_id"));

score.setScore(isScores.getInt("score"));

Quiz quiz = fetchQuizById(isScores.getInt("quiz"));

score.setQuiz(quiz);

ps = conObj.prepareStatement("select \* from users where user\_id=?");

ps.setInt(1, isScores.getInt("user"));

User user = new User();

ResultSet isUser = ps.executeQuery();

if(isUser.next()) {

user.setEmail(isUser.getString("email"));

}

score.setUser(user);

scores.add(score);

}

} catch (Exception e) {

e.printStackTrace();

}

return scores;

}

public Question getQuestion(int qid) {

Question question = new Question();

PreparedStatement ps = null;

try {

ps = conObj.prepareStatement("select \* from questions where question\_id=?");

ps.setInt(1, qid);

ResultSet isQuestion = ps.executeQuery();

if(isQuestion.next()) {

question.setQuestion\_id(isQuestion.getInt("question\_id"));

question.setTitle(isQuestion.getString("title"));

question.setOption1(isQuestion.getString("option1"));

question.setOption2(isQuestion.getString("option2"));

question.setOption3(isQuestion.getString("option3"));

question.setOption4(isQuestion.getString("option4"));

question.setAnswer(isQuestion.getString("answer"));

}

else {

question = null;

}

} catch (Exception e) {

e.printStackTrace();

}

return question;

}

}

**UserServices.java:**

package com.services;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import com.connection.DBConnection;

import com.to.User;

public class UserServices {

private Connection conObj = null;

public UserServices() {

conObj = DBConnection.connectToDB();

}

public User checkUser(String email, String password) {

PreparedStatement ps;

User user = new User();

try {

ps = conObj.prepareStatement("select \* from users where email=?");

ps.setString(1, email);

ResultSet isUser = ps.executeQuery();

if(isUser.next()) {

if(password.equals(isUser.getString("password"))) {

user.setEmail(email);

}

else {

user = null;

}

}

else {

ps = conObj.prepareStatement("insert into users(email,password) values(?,?)");

ps.setString(1, email);

ps.setString(2, password);

int res = ps.executeUpdate();

if(res >=1 ) {

user.setEmail(email);

}

}

} catch (Exception e) {

e.printStackTrace();

}

return user;

}

}

**AddCategoryServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import java.util.ArrayList;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

/\*\*

\* Servlet implementation class AddCategoryServlet

\*/

@WebServlet("/AddCategoryServlet")

public class AddCategoryServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public AddCategoryServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

RequestDispatcher rd = null;

QuizServices qs = new QuizServices();

String name = request.getParameter("name");

List<Category> categoryList = new ArrayList<Category>();

if(!name.replaceAll("\\s", "").equals("")) {

boolean res = qs.addCategory(name);

if(!res) {

categoryList = qs.fetchCategories();

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("category.jsp");

rd.include(request, response);

out.println("<b style='height: 6vh; width: 120px; padding: 1.5rem; background-color: white; border: 1px solid red; font-size: 18px; color: red; margin-bottom: 1rem;'>Add category failed!</b>");;

}

else {

categoryList = qs.fetchCategories();

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("category.jsp");

rd.forward(request, response);

}

}

else {

categoryList = qs.fetchCategories();

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("category.jsp");

rd.forward(request, response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**AddQuestionServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

import com.to.Question;

/\*\*

\* Servlet implementation class AddQuestionServlet

\*/

@WebServlet("/AddQuestionServlet")

public class AddQuestionServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public AddQuestionServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

RequestDispatcher rd = null;

String title = request.getParameter("title");

String option1 = request.getParameter("option1");

String option2 = request.getParameter("option2");

String option3 = request.getParameter("option3");

String option4 = request.getParameter("option4");

String answer = request.getParameter("answer");

int categoryId = Integer.parseInt(request.getParameter("category"));

Category category = qs.getCategory(categoryId);

if(

title.replaceAll("\\s", "").length() != 0 &&

option1.replaceAll("\\s", "").length() != 0 &&

option2.replaceAll("\\s", "").length() != 0 &&

option3.replaceAll("\\s", "").length() != 0 &&

option4.replaceAll("\\s", "").length() != 0 &&

(answer.replaceAll("\\s", "").length() != 0 &&

(answer.equals(option1) || answer.equals(option2) ||

answer.equals(option3) || answer.equals(option4))) &&

category != null

) {

Question question = new Question();

question.setTitle(title);

question.setOption1(option1);

question.setOption2(option2);

question.setOption3(option3);

question.setOption4(option4);

question.setAnswer(answer);

question.setCategory(category);

List<Question> questions = qs.addQuestion(question);

request.setAttribute("questions", questions);

}

rd = request.getRequestDispatcher("QuestionServlet");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**AddQuizServlet.java:**

import java.io.IOException;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Quiz;

/\*\*

\* Servlet implementation class AddQuizServlet

\*/

@WebServlet("/AddQuizServlet")

public class AddQuizServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public AddQuizServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

RequestDispatcher rd = null;

String title = request.getParameter("title");

int category = Integer.parseInt(request.getParameter("category"));

QuizServices qs = new QuizServices();

Quiz quiz = qs.addQuiz(title,category);

if(quiz == null) {

rd = request.getRequestDispatcher("QuizServlet");

rd.forward(request, response);

}

else {

request.setAttribute("quiz", quiz);

rd = request.getRequestDispatcher("QuizDetailServlet");

rd.forward(request, response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**AdminDashboardServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Quiz;

/\*\*

\* Servlet implementation class AdminDashboardServlet

\*/

@WebServlet("/AdminDashboardServlet")

public class AdminDashboardServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public AdminDashboardServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

List<Quiz> quizList = qs.fetchQuizes();

request.setAttribute("quizes", quizList);

RequestDispatcher rd = request.getRequestDispatcher("admin/dashboard.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**AttemptQuizServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Question;

import com.to.Quiz;

/\*\*

\* Servlet implementation class AttemptQuizServlet

\*/

@WebServlet("/AttemptQuizServlet")

public class AttemptQuizServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public AttemptQuizServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

RequestDispatcher rd = null;

int id = Integer.parseInt(request.getParameter("id"));

Quiz quiz = qs.fetchQuizById(id);

request.setAttribute("quiz", quiz);

int categoryId = quiz.getCategory().getCategory\_id();

List<Question> questions = qs.fetchQuestionsByCategory(categoryId);

request.setAttribute("questions", questions);

rd = request.getRequestDispatcher("attemptQuiz.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**CategoryServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

import com.to.Quiz;

/\*\*

\* Servlet implementation class CategoryServlet

\*/

@WebServlet("/CategoryServlet")

public class CategoryServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public CategoryServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

List<Category> categoryList = qs.fetchCategories();

request.setAttribute("categoryList", categoryList);

RequestDispatcher rd = request.getRequestDispatcher("category.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**DeleteCategoryServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import java.util.ArrayList;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

/\*\*

\* Servlet implementation class DeleteCategoryServlet

\*/

@WebServlet("/DeleteCategoryServlet")

public class DeleteCategoryServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public DeleteCategoryServlet() {

super();

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

QuizServices qs = new QuizServices();

RequestDispatcher rd =null;

List<Category> categoryList = new ArrayList<Category>();

int id = Integer.parseInt(request.getParameter("id"));

Category category = qs.getCategory(id);

if(category == null) {

categoryList = qs.fetchCategories();

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("category.jsp");

rd.forward(request, response);

}

else {

categoryList = qs.deleteCategory(id);

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("category.jsp");

rd.forward(request, response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**DisplayCategoryServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

/\*\*

\* Servlet implementation class DisplayCategoryServlet

\*/

@WebServlet("/DisplayCategoryServlet")

public class DisplayCategoryServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public DisplayCategoryServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

QuizServices qs = new QuizServices();

RequestDispatcher rd =null;

int id = Integer.parseInt(request.getParameter("id"));

Category category = qs.getCategory(id);

if(category == null) {

out.println("<p>Invalid Category!</p>");

}

else {

request.setAttribute("category", category);

rd = request.getRequestDispatcher("displayCategory.jsp");

rd.forward(request, response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**LeaderBoard.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Quiz;

import com.to.Score;

/\*\*

\* Servlet implementation class LeaderBoard

\*/

@WebServlet("/LeaderBoard")

public class LeaderBoard extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public LeaderBoard() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

RequestDispatcher rd =null;

int qid = Integer.parseInt(request.getParameter("id"));

Quiz quiz = qs.fetchQuizById(qid);

List<Score> scores = qs.getScores(qid);

request.setAttribute("scores", scores);

request.setAttribute("quiz", quiz);

rd = request.getRequestDispatcher("leaderboard.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**LogoutServlet.java:**

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 javax.servlet.http.HttpSession;

/\*\*

\* Servlet implementation class LogoutServlet

\*/

@WebServlet("/LogoutServlet")

public class LogoutServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public LogoutServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

HttpSession session = request.getSession(false);

if(session != null) {

session.invalidate();

response.sendRedirect("index.jsp");

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**QuestionServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

import com.to.Question;

/\*\*

\* Servlet implementation class QuestionServlet

\*/

@WebServlet("/QuestionServlet")

public class QuestionServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public QuestionServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

RequestDispatcher rd = null;

List<Question> questions = qs.fetchAllQuestions();

request.setAttribute("questions", questions);

List<Category> categoryList = qs.fetchCategories();

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("questions.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**QuizDetailServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

import com.to.Question;

import com.to.Quiz;

/\*\*

\* Servlet implementation class QuizDetailServlet

\*/

@WebServlet("/QuizDetailServlet")

public class QuizDetailServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public QuizDetailServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

RequestDispatcher rd =null;

QuizServices qs = new QuizServices();

Quiz quiz = (Quiz)request.getAttribute("quiz");

Category category = quiz.getCategory();

List<Question> questions = qs.fetchQuestionsByCategory(category.getCategory\_id());

request.setAttribute("questions", questions);

rd = request.getRequestDispatcher("quizDetail.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**QuizServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

import com.to.Quiz;

/\*\*

\* Servlet implementation class QuizServlet

\*/

@WebServlet("/QuizServlet")

public class QuizServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public QuizServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

RequestDispatcher rd = null;

QuizServices qs = new QuizServices();

List<Quiz> quizes = qs.fetchQuizes();

List<Category> categoryList = qs.fetchCategories();

request.setAttribute("quizes", quizes);

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("quiz.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**StartQuizServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Question;

import com.to.Quiz;

/\*\*

\* Servlet implementation class StartQuizServlet

\*/

@WebServlet("/StartQuizServlet")

public class StartQuizServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public StartQuizServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

QuizServices qs = new QuizServices();

RequestDispatcher rd = null;

int id = Integer.parseInt(request.getParameter("id"));

int offset = Integer.parseInt(request.getParameter("offset"));

Quiz quiz = qs.fetchQuizById(id);

System.out.println("quiz: "+quiz);

if(quiz != null) {

List<Question> questions = qs.fetchQuestionsByCategory(quiz.getCategory().getCategory\_id());

Question question = qs.getQuestionByCategory(quiz.getCategory().getCategory\_id(), offset);

request.setAttribute("questions", questions);

request.setAttribute("question", question);

request.setAttribute("quiz", quiz);

rd = request.getRequestDispatcher("playQuiz.jsp");

rd.forward(request, response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**UpdateCategoryServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Category;

/\*\*

\* Servlet implementation class UpdateCategoryServlet

\*/

@WebServlet("/UpdateCategoryServlet")

public class UpdateCategoryServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public UpdateCategoryServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

RequestDispatcher rd = null;

QuizServices qs = new QuizServices();

int id = Integer.parseInt(request.getParameter("id"));

Category category = qs.getCategory(id);

if(category == null) {

out.println("<p>Invalid Category!</p>");

}

else {

String name = request.getParameter("name");

if(!name.replaceAll("\\s", "").equals("")) {

List<Category> categoryList = qs.editCategory(id,name);

request.setAttribute("categoryList", categoryList);

rd = request.getRequestDispatcher("CategoryServlet");

rd.forward(request, response);

}

else {

request.setAttribute("category", category);

rd = request.getRequestDispatcher("displayCategory.jsp");

rd.forward(request, response);

}

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**UserDashboardServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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.services.QuizServices;

import com.to.Quiz;

/\*\*

\* Servlet implementation class UserDashboardServlet

\*/

@WebServlet("/UserDashboardServlet")

public class UserDashboardServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public UserDashboardServlet() {

super();

}

/\*\*

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

\*/

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

try {

QuizServices qs = new QuizServices();

List<Quiz> quizList = qs.fetchQuizes();

request.setAttribute("quizes", quizList);

RequestDispatcher rd = request.getRequestDispatcher("user/dashboard.jsp");

rd.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**UserLoginServlet.java:**

//package com.controllers;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;

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;

import com.services.UserServices;

import com.to.User;

/\*\*

\* Servlet implementation class UserRegisterServlet

\*/

@WebServlet("/UserLoginServlet")

public class UserLoginServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public UserLoginServlet() {

super();

}

/\*\*

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

\*/

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

try {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

UserServices us = new UserServices();

HttpSession session = request.getSession();

String email = request.getParameter("email");

String password = request.getParameter("password");

RequestDispatcher rd = null;

User user = us.checkUser(email, password);

if(user != null) {

session.setAttribute("user", user);

if(user.getEmail().equals("admin@gmail.com")) {

rd = request.getRequestDispatcher("AdminDashboardServlet");

rd.forward(request, response);

}

else {

rd = request.getRequestDispatcher("UserDashboardServlet");

rd.forward(request, response);

}

}

else {

rd = request.getRequestDispatcher("index.jsp");

rd.include(request, response);

out.println("<b style='font-size: 18px; color: red;'>Incorrect Credentials!</b>");

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**ValidateAnswerServlet.java:**

import java.io.IOException;

import java.util.List;

import javax.servlet.RequestDispatcher;

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;

import com.services.QuizServices;

import com.to.Question;

import com.to.Quiz;

import com.to.User;

/\*\*

\* Servlet implementation class ValidateAnswerServlet

\*/

@WebServlet("/ValidateAnswerServlet")

public class ValidateAnswerServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public ValidateAnswerServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

try {

HttpSession session = request.getSession(false);

User user = new User();

if(session != null) {

user = (User)session.getAttribute("user");

}

String answer = null;

int qid = Integer.parseInt(request.getParameter("quiz\_id"));

int questionid = Integer.parseInt(request.getParameter("questionid"));

QuizServices qs = new QuizServices();

Quiz quiz = qs.fetchQuizById(qid);

int index = -1;

Question question = qs.getQuestion(questionid);

List<Question> questions = qs.fetchQuestionsByCategory(quiz.getCategory().getCategory\_id());

for(Question q:questions) {

if(q.getQuestion\_id() == questionid) {

index = questions.indexOf(q);

}

}

RequestDispatcher rd = null;

if(request.getParameter("option1") != null) {

answer = request.getParameter("option1");

}

else if(request.getParameter("option2") != null) {

answer = request.getParameter("option2");

}

else if(request.getParameter("option3") != null) {

answer = request.getParameter("option3");

System.out.println(answer);

}

else if(request.getParameter("option4") != null) {

answer = request.getParameter("option4");

}

else {

rd = request.getRequestDispatcher("StartQuizServlet?id="+quiz.getQuiz\_id()+"&offset="+questions.indexOf(question));

rd.forward(request, response);

}

boolean isCorrect = qs.submitAnswer(quiz.getQuiz\_id(),question.getQuestion\_id(), answer, user.getEmail());

int offset = index+1;

if(offset < questions.size()) {

rd = request.getRequestDispatcher("StartQuizServlet?id="+quiz.getQuiz\_id()+"&offset="+offset);

rd.forward(request,response);

}

else {

rd = request.getRequestDispatcher("LeaderBoard?id="+quiz.getQuiz\_id());

rd.forward(request,response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}