Samiksha shende
078

LAB EXAM

Web Based Java Programming

Q] Create a table in the database containing the columns to store book details like: book name,authors, & price.

Create a JSP page to read Book details and perform CRUD operations using JDBC

Class BookDao.txt

```
package bookstore;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;

public class BookDao {
    private String url;
    private String username;
    private String password;
    private Connection  connection;

    public BookDao(String url, String username, String password) {
        super();
        this.url = url;
        this.username = username;
        this.password = password;
    }

    public void connect () throws SQLException {
        if (connection==null || connection.isClosed()) {
            try {
                Class.forName("com.mysql.jdbc.Driver");
            } catch(ClassNotFoundException e) {
                throw new SQLException (e);
            }
            connection = DriverManager.getConnection(url, username,
password);
        }
    }

    public void disconnect () throws SQLException {
        if (connection!=null && !connection.isClosed()) {
            connection.close();
        }
    }
```

```
public boolean insertBook(Book book) throws SQLException {
        String sql = "INSERT INTO book (title, author, price) VALUES (?, ?, ?)";
        connect();

        PreparedStatement statement = connection.prepareStatement(sql);
        statement.setString(1, book.getTitle());
        statement.setString(2, book.getAuthor());
        statement.setFloat(3, book.getPrice());

        boolean rowInserted = statement.executeUpdate() > 0;
        statement.close();
        disconnect();
        return rowInserted;
}

public List<Book> listAllBooks() throws SQLException {
        List<Book> listBook = new ArrayList<>();

        String sql = "SELECT * FROM book";

        connect();

        Statement statement = connection.createStatement();
        ResultSet resultSet = statement.executeQuery(sql);

        while (resultSet.next()) {
                int id = resultSet.getInt("book_id");
                String title = resultSet.getString("title");
```

```
            String author = resultSet.getString("author");
            float price = resultSet.getFloat("price");

            Book book = new Book(id, title, author, price);
            listBook.add(book);
        }

        resultSet.close();
        statement.close();

        disconnect();

        return listBook;
    }

    public boolean deleteBook(Book book) throws SQLException {
        String sql = "DELETE FROM book where book_id = ?";

        connect();

        PreparedStatement statement = connection.prepareStatement(sql);
        statement.setInt(1, book.getId());

        boolean rowDeleted = statement.executeUpdate() > 0;
        statement.close();
        disconnect();
        return rowDeleted;
    }
```

```
	public boolean updateBook(Book book) throws SQLException {
		String sql = "UPDATE book SET title = ?, author = ?, price = ?";
		sql += " WHERE book_id = ?";
		connect();

		PreparedStatement statement = connection.prepareStatement(sql);
		statement.setString(1, book.getTitle());
		statement.setString(2, book.getAuthor());
		statement.setFloat(3, book.getPrice());
		statement.setInt(4, book.getId());

		boolean rowUpdated = statement.executeUpdate() > 0;
		statement.close();
		disconnect();
		return rowUpdated;
	}

	public Book getBook(int id) throws SQLException {
		Book book = null;
		String sql = "SELECT * FROM book WHERE book_id = ?";

		connect();

		PreparedStatement statement = connection.prepareStatement(sql);
		statement.setInt(1, id);

		ResultSet resultSet = statement.executeQuery();

		if (resultSet.next()) {
```

```
                        String title = resultSet.getString("title");

                        String author = resultSet.getString("author");

                        float price = resultSet.getFloat("price");


                        book = new Book(id, title, author, price);

                }


                resultSet.close();

                statement.close();

                disconnect();

                return book;

        }


}
```

BookForm.txt:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
   <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
        <head>
                <meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
                <title>Book Store Title</title>
        </head>
        <body>
                <center>
                        <h1>Books Management</h1>
                        <h2>
                        <a href = "ControllerServlet?action=new">Add New
Book</a>
                        <a href = "ControllerServlet?action=list">List All Book</a>
                        </h2>
                </center>
                <div align=center>
                        <table border="1" cellpadding="5">
                                <caption><h2>List of Books</h2></caption>
                                <tr>
                                        <th>Id</th>
                                        <th>Title</th>
                                        <th>Author</th>
                                        <th>Price</th>
                                        <th>Actions</th>
                                </tr>
                                <c:forEach items="${listBook}" var="book">
                                <tr>
                                        <td><c:out value="${book.id}"/></td>
                                        <td><c:out value="${book.title}"/></td>
                                        <td><c:out value="${book.author}"/></td>
                                        <td><c:out value="${book.price}"/></td>
                                        <td>
                                                <a
href="ControllerServlet?action=edit&id=<c:out
value='${book.id}'/>">Edit</a>
                                                <a
href="ControllerServlet?action=delete&id=<c:out
value='${book.id}'/>">Delete</a>
                                        </td>
                                </tr>
                                </c:forEach>
                        </table>
                </div>

        </body>
</html>
```

3] Booklist.txt:

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

   pageEncoding="ISO-8859-1"%>

   <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

Samiksha shende
078

```
<html>

        <head>

                <meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">

                <title>Book Store Title</title>

        </head>

        <body>

                <center>

                        <h1>Books Management</h1>

                        <h2>

                        <a href = "ControllerServlet?action=new">Add New
Book</a>

                        <a href = "ControllerServlet?action=list">List All Book</a>

                        </h2>

                </center>

                <div align=center>

                        <table border="1" cellpadding="5">

                                <caption><h2>List of Books</h2></caption>

                                <tr>

                                        <th>Id</th>

                                        <th>Title</th>

                                        <th>Author</th>

                                        <th>Price</th>
```

```
                    <th>Actions</th>

                </tr>

                <c:forEach items="${listBook}" var="book">

                <tr>

                    <td><c:out value="${book.id}"/></td>

                    <td><c:out value="${book.title}"/></td>

                    <td><c:out value="${book.author}"/></td>

                    <td><c:out value="${book.price}"/></td>

                    <td>

                        <a
href="ControllerServlet?action=edit&id=<c:out
value='${book.id}'/>">Edit</a>

                        <a
href="ControllerServlet?action=delete&id=<c:out
value='${book.id}'/>">Delete</a>

                    </td>

                </tr>

                </c:forEach>

            </table>

        </div>



    </body>

</html>
```

4] ControllerServlet.txt

```
package bookstore;


import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;


public class BookDao {


	private String url;
	private String username;
	private String password;
	private Connection  connection;


	public BookDao(String url, String username, String password) {
		super();
		this.url = url;
		this.username = username;
		this.password = password;
```

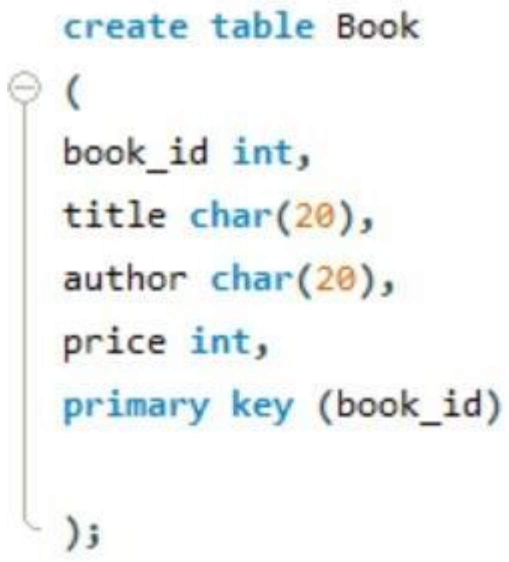
```
    }

    public void connect () throws SQLException {
        if (connection==null || connection.isClosed()) {
            try {
                Class.forName("com.mysql.jdbc.Driver");
            } catch(ClassNotFoundException e) {
                throw new SQLException (e);
            }
            connection = DriverManager.getConnection(url, username, password);
        }
    }

    public void disconnect () throws SQLException {
        if (connection!=null && !connection.isClosed()) {
            connection.close();
        }
    }

    public boolean insertBook(Book book) throws SQLException {
        String sql = "INSERT INTO book (title, author, price) VALUES (?, ?, ?)";
        connect();
        PreparedStatement statement =
connection.prepareStatement(sql);
        statement.setString(1, book.getTitle());
        statement.setString(2, book.getAuthor());
        statement.setFloat(3, book.getPrice());


        boolean rowInserted = statement.executeUpdate() > 0;
        statement.close();
        disconnect();
        return rowInserted;
    }


    public List<Book> listAllBooks() throws SQLException {
        List<Book> listBook = new ArrayList<>();


        String sql = "SELECT * FROM book";


        connect();


        Statement statement = connection.createStatement();
        ResultSet resultSet = statement.executeQuery(sql);


        while (resultSet.next()) {
            int id = resultSet.getInt("book_id");
            String title = resultSet.getString("title");
            String author = resultSet.getString("author");
            float price = resultSet.getFloat("price");

            Book book = new Book(id, title, author, price);
            listBook.add(book);
        }

        resultSet.close();
        statement.close();

        disconnect();

        return listBook;
    }

    public boolean deleteBook(Book book) throws SQLException {
        String sql = "DELETE FROM book where book_id = ?";

        connect();

        PreparedStatement statement =
connection.prepareStatement(sql);
```

```
        statement.setInt(1, book.getId());


        boolean rowDeleted = statement.executeUpdate() > 0;

        statement.close();

        disconnect();

        return rowDeleted;

    }


    public boolean updateBook(Book book) throws SQLException {

        String sql = "UPDATE book SET title = ?, author = ?, price = ?";

        sql += " WHERE book_id = ?";

        connect();


        PreparedStatement statement = connection.prepareStatement(sql);

        statement.setString(1, book.getTitle());

        statement.setString(2, book.getAuthor());

        statement.setFloat(3, book.getPrice());

        statement.setInt(4, book.getId());


        boolean rowUpdated = statement.executeUpdate() > 0;

        statement.close();

        disconnect();

        return rowUpdated;
    }


    public Book getBook(int id) throws SQLException {

        Book book = null;

        String sql = "SELECT * FROM book WHERE book_id = ?";


        connect();


        PreparedStatement statement =
connection.prepareStatement(sql);

        statement.setInt(1, id);


        ResultSet resultSet = statement.executeQuery();


        if (resultSet.next()) {

            String title = resultSet.getString("title");

            String author = resultSet.getString("author");

            float price = resultSet.getFloat("price");


            book = new Book(id, title, author, price);

        }


        resultSet.close();

        statement.close();
```

```
                disconnect();
                return book;
        }


}
```

5] WEB.xml.txt

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns="http://xmlns.jcp.org/xml/ns/javaee"
        xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
                http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd"
        id="WebApp_ID" version="3.1">
        <display-name>Books Management Web Application</display-name>

  <welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
  </welcome-file-list>


        <context-param>
                <param-name>url</param-name>
                <param-value>jdbc:mysql://localhost:3306/adjavlab</param-value>
        </context-param>
    <context-param>

        <param-name>username</param-name>

        <param-value>root</param-value>

    </context-param>


    <context-param>

        <param-name>password</param-name>

        <param-value>pruthvi</param-value>

    </context-param>


    <servlet>

        <servlet-name>ControllerServlet</servlet-name>

        <servlet-class>bookstore.ControllerServlet</servlet-class>

    </servlet>


    <servlet-mapping>

        <servlet-name>ControllerServlet</servlet-name>

        <url-pattern>/ControllerServlet</url-pattern>

    </servlet-mapping>



</web-app>
```

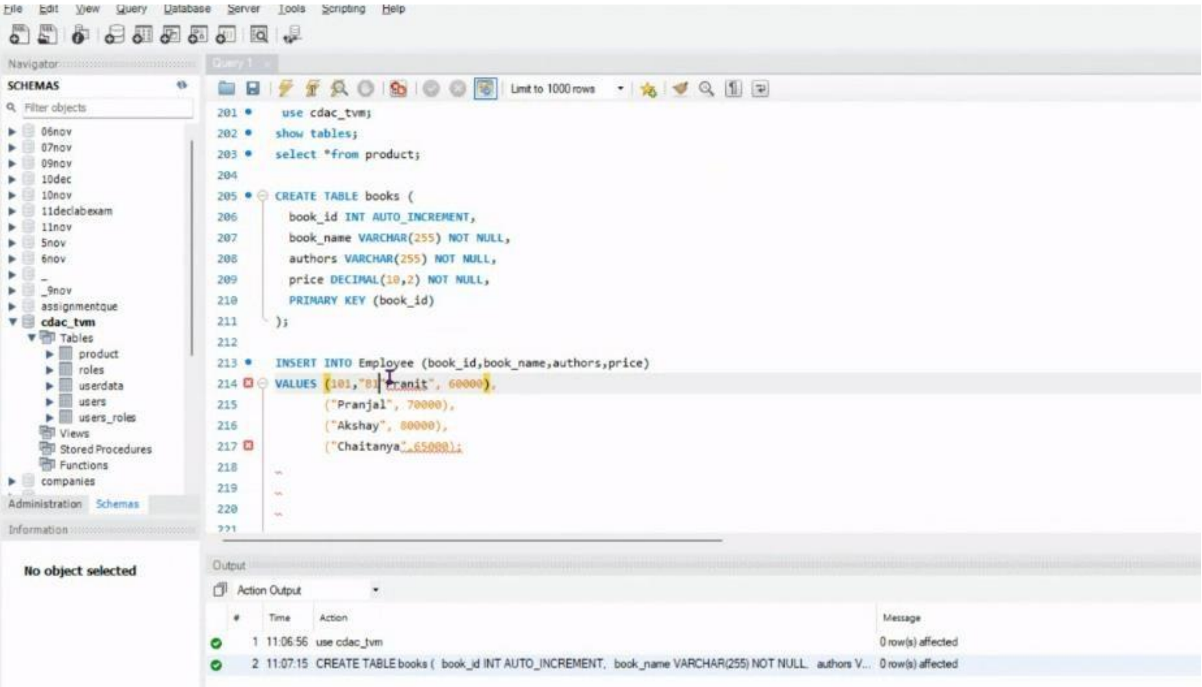


```
create table Book
(
book_id int,
title char(20),
author char(20),
price int,
primary key (book_id)

);
```

6] book.txt

```
package bookstore;

public class Book {
	private int id;
	private String title;
	private String author;
	private float price;

	public Book() {
	}

	public Book(int id) {
		this.id = id;
```

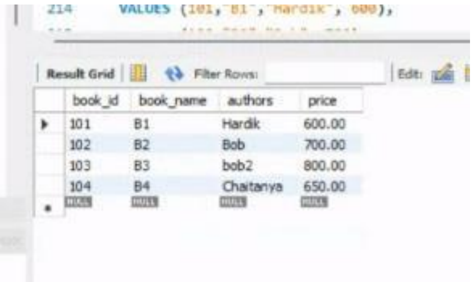
```
    }


    public Book(String title, String author, float price) {
        this.title = title;
        this.author = author;
        this.price = price;
    }
    public Book(int id, String title, String author, float price) {
        this.id = id;
        this.title = title;
        this.author = author;
        this.price = price;
    }


    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
        }

        public String getAuthor() {

                return author;

        }

        public void setAuthor(String author) {

                this.author = author;

        }

        public float getPrice() {

                return price;

        }

        public void setPrice(float price) {

                this.price = price;

        }

}
```



```
 use cdac_tvm;
show tables;
select *from product;

CREATE TABLE books (
  book_id INT AUTO_INCREMENT,
  book_name VARCHAR(255) NOT NULL,
  authors VARCHAR(255) NOT NULL,
  price DECIMAL(10,2) NOT NULL,
  PRIMARY KEY (book_id)
);

INSERT INTO Employee (book_id,book_name,authors,price)
VALUES (101,"81"Pranit", 60000),
       ("Pranjal", 70000),
       ("Akshay", 80000),
       ("Chaitanya",65000);
```

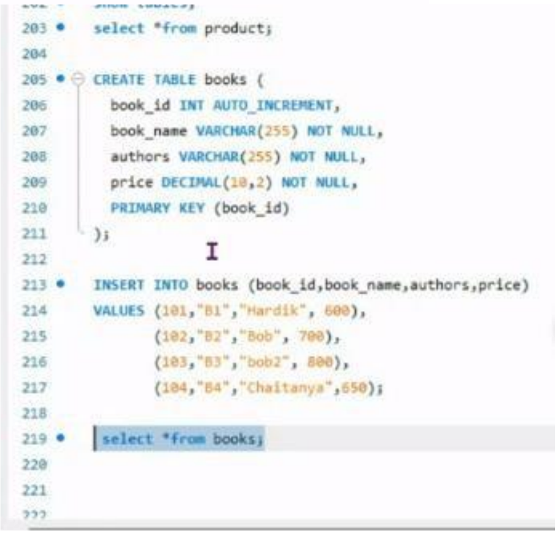
#	Time	Action	Message
1	11:06:56	use cdac_tvm	0 row(s) affected
2	11:07:15	CREATE TABLE books (book_id INT AUTO_INCREMENT, book_name VARCHAR(255) NOT NULL, authors V...	0 row(s) affected

Samiksha shende
078



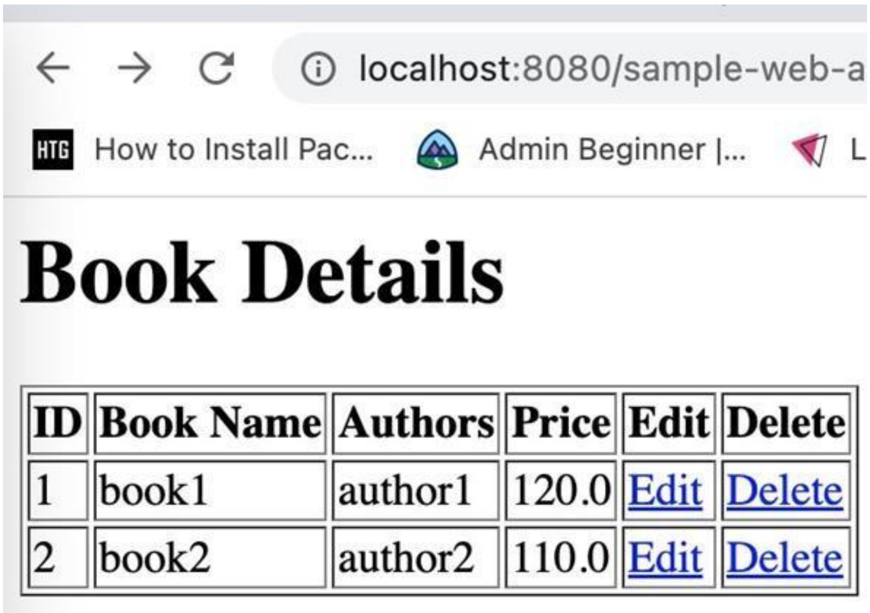
```
VALUES (101,"B1","Hardik", 600),
```

book_id	book_name	authors	price
101	B1	Hardik	600.00
102	B2	Bob	700.00
103	B3	bob2	800.00
104	B4	Chaitanya	650.00
NULL	NULL	NULL	NULL



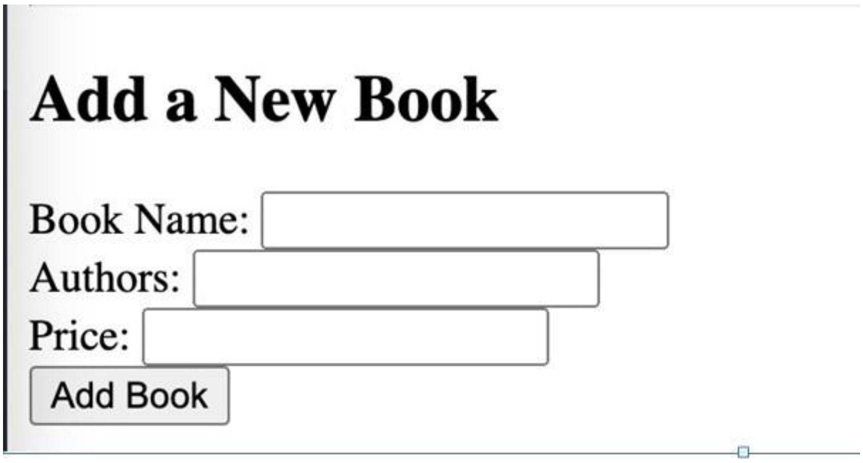
```
show tables;
select *from product;

CREATE TABLE books (
  book_id INT AUTO_INCREMENT,
  book_name VARCHAR(255) NOT NULL,
  authors VARCHAR(255) NOT NULL,
  price DECIMAL(10,2) NOT NULL,
  PRIMARY KEY (book_id)
);

INSERT INTO books (book_id,book_name,authors,price)
VALUES (101,"B1","Hardik", 600),
       (102,"B2","Bob", 700),
       (103,"B3","bob2", 800),
       (104,"B4","Chaitanya",650);

select *from books;
```

Samiksha shende
078



Book Details

ID	Book Name	Authors	Price	Edit	Delete
1	book1	author1	120.0	Edit	Delete
2	book2	author2	110.0	Edit	Delete



Add a New Book

Book Name:
Authors:
Price: