Assignment No. : 5

TITLE

Title: Implement a sample program demonstrating the use of Servlet

OBJECTIVES

1. Understand about basic concepts of Server side technology
2. To create a database using Sevlet and JDBC

PROBLEM STATEMENT

Create a database table ebookshop (book_id,book_title,book_author,book_price,quantity) using databse like Oracle/MySQL etc. And display (use SQL select query) the table content using servlet

OUTCOME

Students will be able to,

1. Develop a dynamic webpage using HTML and Servlet.
2. Write a server side java application called Servlet to catch the data sent from client,process it and store it on database (MySQL).

SOFTWARE & HARDWARE REQUIREMENTS

Software: Eclipse/Netbeans, Browser

THEORY-CONCEPT

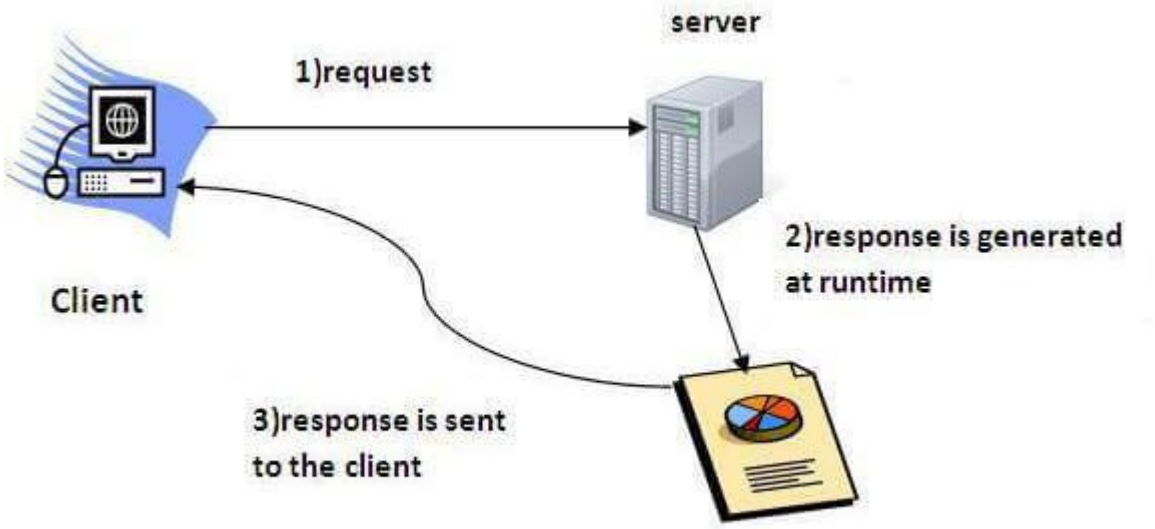
Servlet technology is used to create a web application (resides at server side and generates a dynamic web page).

Servlet technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was common as a server-side programming language. However, there were many disadvantages to this technology.
There are many interfaces and classes in the Servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, etc.

What is a Servlet?

Servlet can be described in many ways, depending on the context.

- Servlet is a technology which is used to create a web application.
- Servlet is an API that provides many interfaces and classes including documentation.
- Servlet is an interface that must be implemented for creating any Servlet.
- Servlet is a class that extends the capabilities of the servers and responds to the incoming requests. It can respond to any requests.
- Servlet is a web component that is deployed on the server to create a dynamic web page



CGI (Common Gateway Interface)

CGI technology enables the web server to call an external program and pass HTTP request information to the external program to process the request. For each request, it starts a new process.

Disadvantages of CGI

There are many problems in CGI technology:
If the number of clients increases, it takes more time for sending the response.
For each request, it starts a process, and the web server is limited to start processes.
It uses platform dependent language e.g. C, C++, perl.

Advantages of Servlet

There are many advantages of Servlet over CGI. The web container creates threads for handling the multiple requests to the Servlet. Threads have many benefits over the Processes such as they share a common memory area, lightweight, cost of communication between the threads are low. The advantages of Servlet are as follows:

- Better performance: because it creates a thread for each request, not process.
- Portability: because it uses Java language.
- Robust: JVM manages Servlets, so we don't need to worry about the memory leak, garbage collection, etc.
- Secure: because it uses java language.

JDBC

JDBC API is a Java API that can access any kind of tabular data, especially data stored in a Relational Database. JDBC works with Java on a variety of platforms, such as Windows, Mac OS, and the various versions of UNIX.

JDBC stands for Java Database Connectivity, which is a standard Java API for database-independent connectivity between the Java programming language and a wide range of databases.
The JDBC library includes APIs for each of the tasks mentioned below that are commonly associated with database usage.

- Making a connection to a database.
- Creating SQL or MySQL statements.
- Executing SQL or MySQL queries in the database.
- Viewing & Modifying the resulting records.

TECHNOLOGY / TOOL

- IDE - Eclipse Neon.3
- JDK - 1.8 or later
- Apache Tomcat - 8.5
- Servlet - 2.5+
- MySQL - mysql-connector-java-8.0.13.jar

DESIGN / EXECUTION STEPS

To start with interfacing Java Servlet Program with JDBC Connection:
1. Proper JDBC Environment should set-up along with database creation.
2. To do so, download the mysql-connector.jar file from the internet,
3. As it is downloaded, move the jar file to the apache-tomcat server folder,
4. Place the file in lib folder present in the apache-tomcat directory.

To start with the basic concept of interfacing:
Step 1: Creation of Database and Table in MySQL
As soon as jar file is placed in the folder, create a database and table in MySQL,

```
mysql> create database demoprj;
Query OK, 1 row affected (4.10 sec)

mysql> use demoprj
Database changed
```

```
mysql> create table demo(id int(10), string varchar(20));
Query OK, 0 rows affected (1.93 sec)

mysql> desc demo;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| id     | int(10)     | YES  |     | NULL    |       |
| string | varchar(20) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
2 rows in set (0.40 sec)
```

Step 2: Implementation of required Web-pages
Create a form in HTML file, where take all the inputs required to insert data into the database. Specify the servlet name in it, with the POST method as security is important aspects in database connectivity.

Step 3: Creation of Java Servlet program with JDBC Connection
To create a JDBC Connection steps are

1. Import all the packages
2. Register the JDBC Driver
3. Open a connection
4. Execute the query, and retrieve the result
5. Clean up the JDBC Environment

Create a separate class to create a connection of database, as it is a lame process to writing the same code snippet in all the program. Create a .java file which returns a Connection object.

Step 4: To use this class method, create an object in Java Servlet program
Below program shows Servlet Class which create a connection and insert the data in the demo table,

Step 5: Get the data from the HTML file
To get the data from the HTML file, the request object is used which calls getParameter() Method to fetch the data from the channel. After successful insertion, the writer object is created to display a success message.
After insertion operation from Servlet, data will be reflected in MySQL Database

TEST CASES

Manual testing is used to validate the fields like book_id, book_title, book_author, book_price, quantity of the student entered by user with the database.

SAMPLE EXPECTED OUTPUT

Users of the System

Admin

Functional Requirements

1. Admin

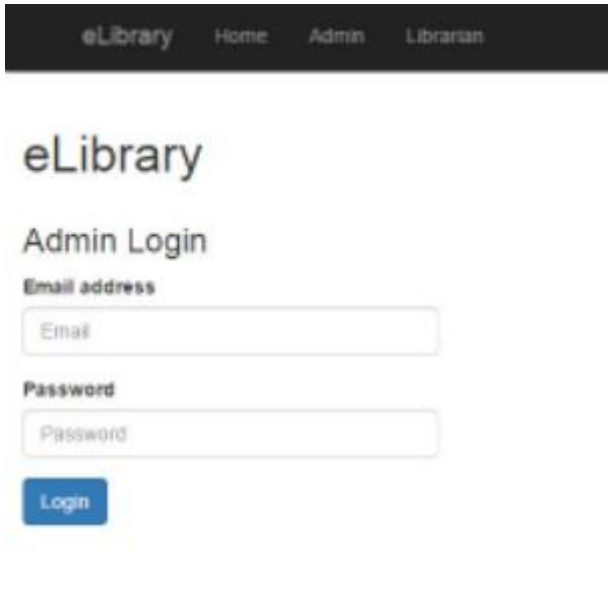
   Can add/edit/view/delete book

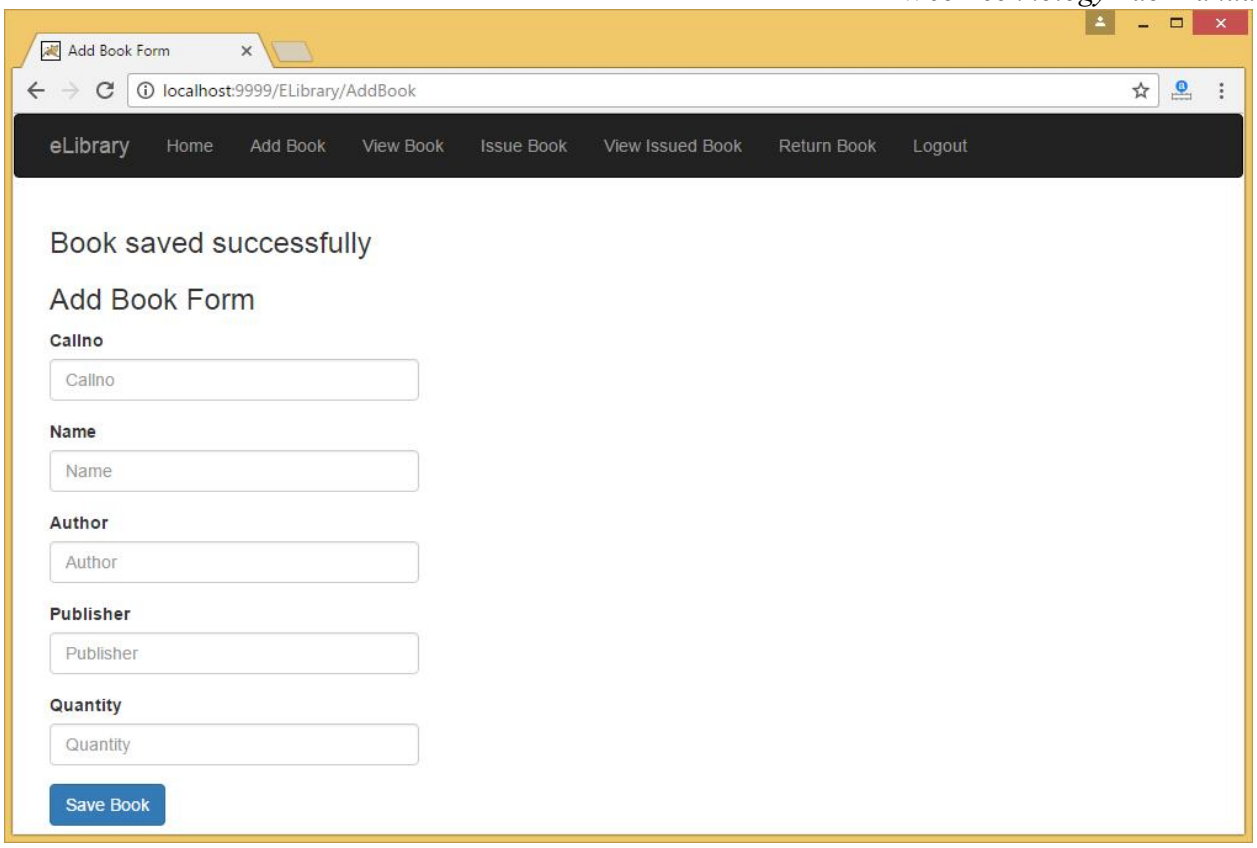
   Can logout

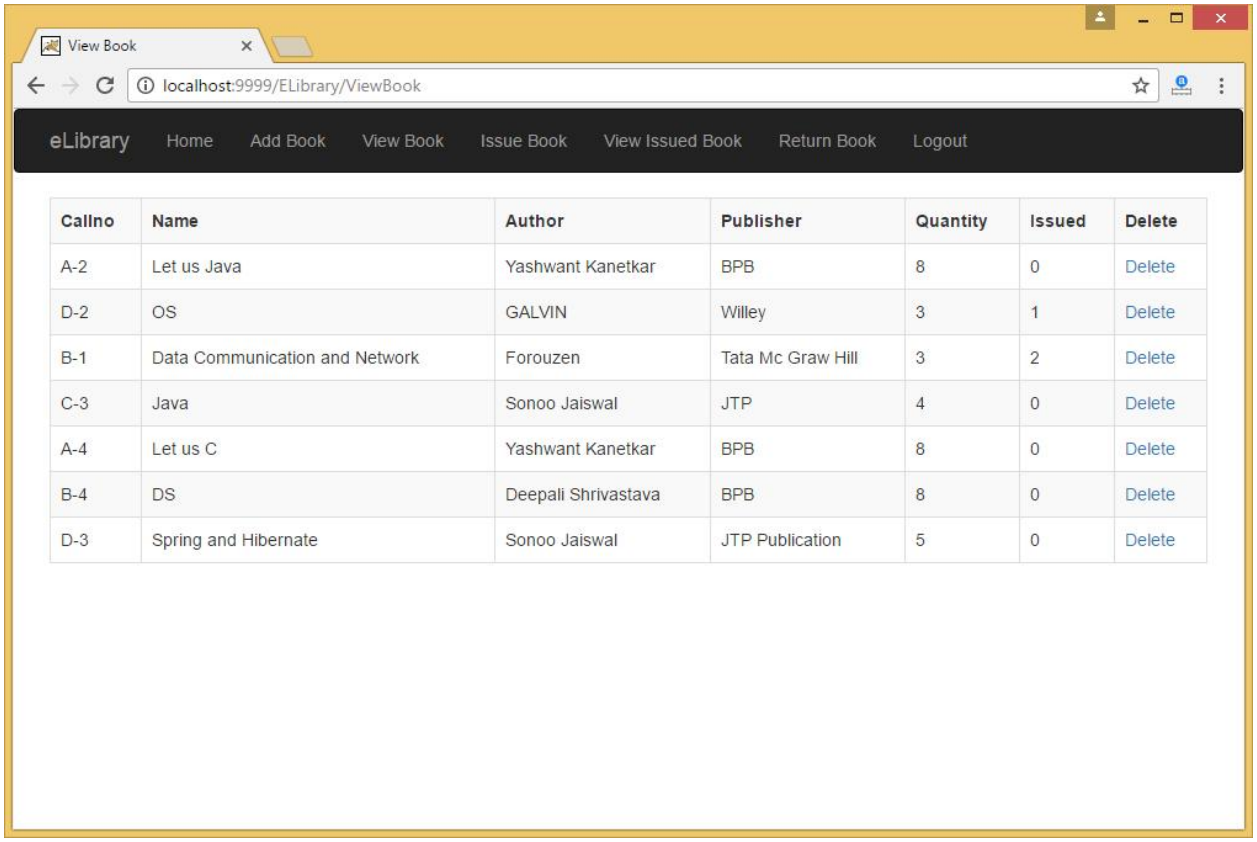
Front End and Back End

Front End: Servlet, HTML, CSS, Bootstrap

Back End: MySQL







CalIno	Name	Author	Publisher	Quantity	Issued	Delete
A-2	Let us Java	Yashwant Kanetkar	BPB	8	0	Delete
D-2	OS	GALVIN	Willey	3	1	Delete
B-1	Data Communication and Network	Forouzen	Tata Mc Graw Hill	3	2	Delete
C-3	Java	Sonoo Jaiswal	JTP	4	0	Delete
A-4	Let us C	Yashwant Kanetkar	BPB	8	0	Delete
B-4	DS	Deepali Shrivastava	BPB	8	0	Delete
D-3	Spring and Hibernate	Sonoo Jaiswal	JTP Publication	5	0	Delete

CONCLUSION/ANALYSIS

Hence, we have performed the dynamic web application using Servlet and MySQL.

SAMPLE ORAL QUESTIONS

1. What is different between web server and application server?
2. Which HTTP method is non-idempotent?
3. What is the difference between GET and POST method?
4. What is MIME Type?
5. What is a web application and what is its directory structure?
6. What is a servlet?
7. What are the advantages of Servlet over CGI?
8. What are common tasks performed by Servlet Container?
9. What is ServletConfig object?
10. What is ServletContext object?
11. What is difference between ServletConfig and ServletContext?
12. What is Request Dispatcher?
13. What is difference between PrintWriter and ServletOutputStream?
14. Can we get PrintWriter and ServletOutputStream both in a servlet?
15. How can we create deadlock situation in servlet?
16. What is the use of servlet wrapper classes?
17. What is SingleThreadModel interface?
18. Do we need to override service() method?
19. Is it good idea to create servlet constructor?
20. What is difference between GenericServlet and HttpServlet?
21. What is the inter-servlet communication?
22. Are Servlets Thread Safe? How to achieve thread safety in servlets?
23. What is servlet attributes and their scope?
24. How do we call one servlet from another servlet?
25. How can we invoke another servlet in a different application?
26. What is difference between ServletResponse sendRedirect() and RequestDispatcher forward() method?
27. Why HttpServlet class is declared abstract?