Assignment 4

1 Problem Statement

Add dynamic web application features to previously selected application using Servlet, JSP and backend (MySQL / MongoDB).

2 Learning Objectives

In this assignment, students will :

1.Study the concept of servlet

2.Execute servlets and understand server-side programming.

3 Learning Outcomes

After completion of this assignment, students will be able to :

1. Understand server-side programming

2. Define and execute servlets

4 Requirements

Hardware : 64-bit 2.8 GHz processor, 4 GB RAM

Software : 64-bit OS, Web Browser

5 Theory

Servlets are small programs that execute on the server side. Servlets are pieces of Java source code that add functionality to a web serverServlet provides full support for sessions, a way to keep track of a particular user over time as a website's pages are being viewed. They also can communicate directly with a web server using a standard interface.

Servlet Life Cycle

- A servlet's life cycle methods function similarly to the life cycle methods of applets. The init(ServletConfig) method is called automatically when a web server first begins a servlet to handle the user's request. The init() method is called only once. ServletConfig is an interface in the javax.servlet package, containing the methods to find out more about the environment in which a servlet is running.
- The servlet action is in the service() method. The service() method checks the HTTP request type (GET, POST, PUT, DELETE etc.) and calls doGet(), doPost(), doPut(), doDelete() etc. methods. A GET request results from normal request for a URL or from an HTML form that has no METHOD specified. The POST request results from an HTML form that specifically lists POST as the METHOD.
- The destroy() method is called when a web server takes a servlet offline.

Using Servlets
One of the main tasks of a servlet is to collect information from a web user and present something back in response. Collection of information is achieved using form, which is a group of text boxes, radio buttons, text areas, and other input fields on the web page. Each field on a form stores information that can be transmitted to a web server and then sent to a Java servlet. web browsers communicate with servers by using Hypertext Transfer Protocol (HTTP).

- Form data can be sent to a server using two kinds of HTTP requests: get and post. When web page calls a server using get or post, the name of the program that handles the request must be specified as a web address, also called uniform resource locator (URL). A get request affixes all data on a form to the end of a URL. A post request includes form data as a header and sent separately from the URL. This is generally preferred, and it's required when confidential information is being collected on the form.
- Java servlets handle both of these requests through methods inherited from the HTTPServlet class: doGet(HttpServletRequest, HttpServletResponse) and doPost(HttpServletRequest, HttpServletResponse). These methods throw two kinds of exceptions: ServletException, part of javax.servlet package, and IOException, an exception in the java.io package.

- The getparameter(String) method is used to retrieve the fields in a servlet with the name of the field as an argument. Using an HTML document a servlet communicates with the user.
- While preparing the response you have to define the kind of content the servlet is sending to a browser. The setContentType(String) method is used to decide the type of response servlet is communicating. Most common form of response is written using an HTML as: setContentType("text/html").
- To send data to the browser, you create a servlet output stream associated with the browser and then call the println(String) method on that stream. The getWriter() method of HttpServletResponse object returns a stream. which can be used to send a response back to the client

6 Output

Screenshots of result

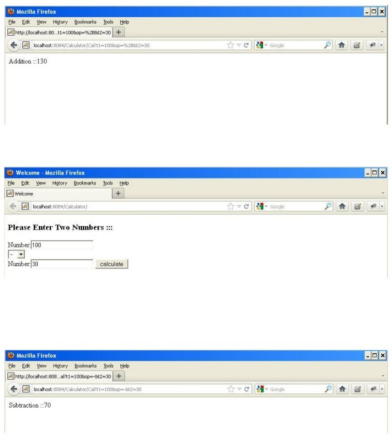

Figure 1: Output

7 Conclusion

Hence through this assignment, we learnt the concept of servlets and successfully implemented the assignment using servlets.