SERVLET - DAY 2 ASSIGNMENT

Harshit Kushmakar| 16896

1. Create a form that accepts a name, password and pass it on to the servlet

a. The receiving servlet should check the name and password against a hard coded

list of name-password pairs stored in a HashMap.

b. HashMap should be initialized with some dummy usernames and passwords in

the constructor/init method of the servlet.

c. Servlet should return either 'Name/Password Match' or 'Name/Password Does

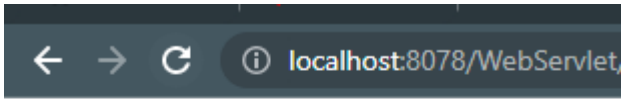
Not Match' back to client.

```
package servletassignment2;

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

@WebServlet("/LoginServlet")

public class LoginServlet extends HttpServlet {

    private Map<String, String> userPasswords;

    public void init() throws ServletException {
        userPasswords = new HashMap<String, String>();
        userPasswords.put("kushmakar", "kushmakar12");
        userPasswords.put("suman", "suman12");
        userPasswords.put("Charlie", "charlie12");
    }

    public void doGet(HttpServletRequest request, HttpServletResponse
response) throws  IOException {

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

```
        if (userPasswords.containsKey(name) &&
userPasswords.get(name).equals(password)) {
            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("Name/Password Match");
        } else {
            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("Name/Password Does Not Match");
        }
    }
}
```

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Login Form</title>
</head>
<body>
<form action="LoginServlet" method="get">
  <label for="name">Name:</label>
  <input type="text" id="name" name="name" required><br>
  <label for="password">Password:</label>
  <input type="password" id="password" name="password" required><br>
  <input type="submit" value="Login">

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

Name/Password Match



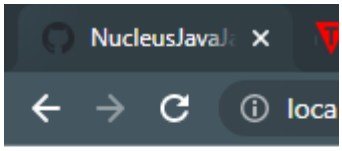
Name:

Password:

Login

2. In the above Question, use forward and redirect. If the username and password is correct, set a request attribute named “message” and the value for the same is “'Name/Password Match”. Forward the request using RequestDispatcher forward() method to another servlet and print all three values there – Username and password entered by user and the ‘message’ attribute. If the password is incorrect, redirect the request to an error page using ServletResponse.sendRedirect().

```
package servletassignment2;

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

@WebServlet("/DisplayServlet")
public class DisplayServlet extends HttpServlet {

    public void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {

request
        String name = request.getParameter("name");
        String password = request.getParameter("password");
        String message = (String) request.getAttribute("message");


        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><head><title>Login
Result</title></head><body>");
        out.println("<p>Username: " + name + "</p>");
        out.println("<p>Password: " + password + "</p>");
        out.println("<p>Message: " + message + "</p>");
        out.println("</body></html>");
    }
}
```



Username: null

Password: null

Message: null

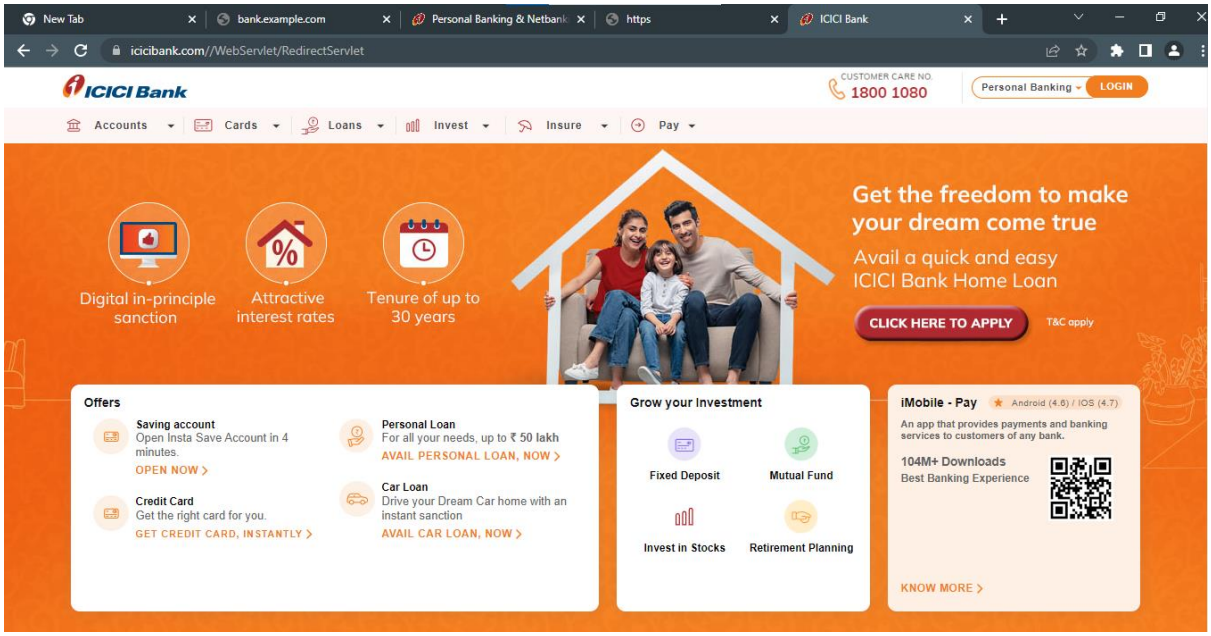
3. In the above question, replace sendRedirect() method with sendError() method.

```
package servletassignment2;

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

public class LoginServlets extends HttpServlet {

    private Map<String, String> userPasswords;

    public void init() throws ServletException {
        // Initialize the user-password map with some dummy data
        userPasswords = new HashMap<String, String>();
        userPasswords.put("Alice", "pa$$word1");
        userPasswords.put("Bob", "pa$$word2");
        userPasswords.put("Charlie", "pa$$word3");
    }

    public void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        String name = request.getParameter("name");
        String password = request.getParameter("password");

        // Check if the name and password match one of the pairs in the map
        if (userPasswords.containsKey(name) &&
userPasswords.get(name).equals(password)) {
            // Set a request attribute for the message
            request.setAttribute("message", "Name/Password Match");

            // Forward the request to the DisplayServlet
            RequestDispatcher rd =
request.getRequestDispatcher("DisplayServlet");
            rd.forward(request, response);
        } else {
            // Send an error response with a 401 Unauthorized status code
            response.sendError(HttpServletResponse.SC_UNAUTHORIZED,
"Invalid username or password");
        }
```

```
    }
}
```

4. Write a servlet that redirects requests form one host to another host, giving an explanation to the client before the redirection. As we generally see during payment and request will be redirected to bank website.

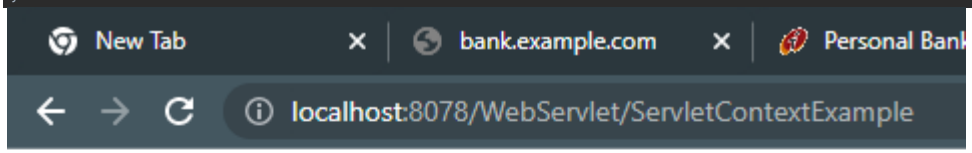
```
package servletassignment2;

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

@WebServlet("/RedirectServlet")
public class RedirectServlet extends HttpServlet {

    public void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {


        String currentHost = request.getServerName();
        String targetHost = "www.icicibank.com/"; // Replace with your
target host name


        response.setContentType("text/html");

        PrintWriter out = response.getWriter();


        out.println("<html>");
        out.println("<head><title>Redirecting to " + targetHost +
"</title></head>");
        out.println("<body>");
        out.println("<h1>Redirecting to " + targetHost + "...</h1>");
        out.println("<p>Your request is being redirected to " + targetHost
+ " for payment processing.</p>");
        out.println("<p>If you are not automatically redirected, please
click <a href=\"https://" + targetHost + "\">here</a>.</p>");
        out.println("</body></html>");


        response.setStatus(HttpServletResponse.SC_MOVED_TEMPORARILY);


        response.setHeader("Location", "https://" + targetHost +
request.getRequestURI());
    }
}
```



5. Create a servlet that displays the servlet config and servlet context parameters. For this purpose, you can create more than one servlet.

```
package servletassignment2;

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


@WebServlet("/ServletContextExample")
public class ServletContextExample extends HttpServlet {

    public void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {

        // Get the servlet context
        ServletContext context = getServletContext();

        // Get the servlet context parameters
        String companyName = context.getInitParameter("companyName");
        String address = context.getInitParameter("address");
        String phone = context.getInitParameter("phone");

        // Set the response content type to HTML
        response.setContentType("text/html");


        PrintWriter out = response.getWriter();
```

```
        out.println("<html>");
        out.println("<head><title>Servlet Context Example</title></head>");
        out.println("<body>");
        out.println("<h1>Servlet Context Example</h1>");
        out.println("<p>Company Name: " + companyName + "</p>");
        out.println("<p>Address: " + address + "</p>");
        out.println("<p>Phone: " + phone + "</p>");
        out.println("</body></html>");
    }
}
```



Servlet Context Example

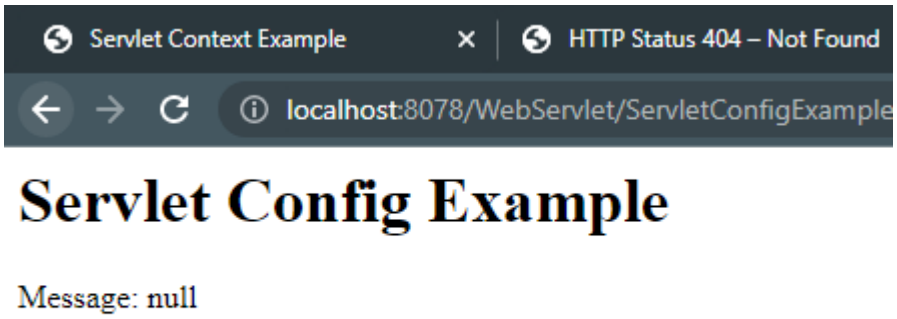
Company Name: null

Address: null

Phone: null

```
package servletassignment2;

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

@WebServlet("/ServletConfigExample")
public class ServletConfigExample extends HttpServlet {

    private String message;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        message = config.getInitParameter("message");
    }

    public void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {

        // Set the response content type to HTML
        response.setContentType("text/html");


        PrintWriter out = response.getWriter();
        out.println("<html>");
```

```
        out.println("<head><title>Servlet Config Example</title></head>");
        out.println("<body>");
        out.println("<h1>Servlet Config Example</h1>");
        out.println("<p>Message: " + message + "</p>");
        out.println("</body></html>");
    }
}
```



6. Map error-code '404' with an underconstruction.html page. Whenever 404 error occurs, instead of showing the default error page, it shows 'underconstruction.html' page.

7. Map java.lang.Exception with an ErrorServlet.