



**DEPARTMENT OF**

**COMPUTER SCIENCE & ENGINEERING**

## **Experiment 3.1**

**Student Name : Jitesh kumar**

**Branch: BE-CSE**

**Semester: 5TH**

**Subject Code: 20CSP-321**

**UID : 20BCS2334**

**Section/Group: 20BCS903 A**

**Subject Name : PBLJ Lab**

**1.Aim:** Create a palindrome creator application for making a longest possible palindrome out of given input string.

**2.Software/Hardware Requirements:** VS Code or Eclipse

**3. Algorithm/ PsuedoCode:**

**STEP 1-** Create a index.jsp file in a webapp directory.

**STEP 2** - Create a package named as fun and create a java file named as functions.java .

**STEP 3** - functions.java file contains the logic for checking the palindromic substring .

**STEP 4-** At Last start the server and display the output on the web browser.

**STEP 5-** EXIT

**CODE:**

**Index.jsp**

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



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

```
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<style>
body{
    background: linear-gradient(45deg, red, blue);
background-size: cover;          color: white;
align-items: center;
}
h1{
    text-align:center;
}
.fall{
    border: 2px solid orange;background: blue;
padding: 5px;          max-width: 500px;
height: 100px;        margin: auto;
font-size: 19px;
}
input{
    width: 250px;
}
button{
    position: relative;
left: 170px;
    margin: 10px; width: 60px;height:30px;
cursor:pointer;border-radius:5px;
}    button:hover{
color:white;
background: black;
}
</style>
</head>
<body>
    <h1>find the Longest Palindromic Substring</h1>
    <form class="fall" name="funcitons"
action="<%=request.getContextPath()%>/functions" method="post">
        Enter the Palindromic String: <input class="check" type="text" name="pal"
size="50"><br>
        <button type="submit">Submit</button>
        <button type="reset">Reset</button>
    </form>
    <h1> longest Palindromic SubString <br/> <%=request.getAttribute("ans")
%></h1>
</body>
</html>
```



**DEPARTMENT OF**

**COMPUTER SCIENCE & ENGINEERING**

## Functions.java

```
package fun;

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;

/**
 * Servlet implementation class functions
 */
@WebServlet(name="functions",urlPatterns={"/functions"}) public
class functions extends HttpServlet {

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        String a=request.getParameter("pal");

//        String fun=request.getParameter("fun");

        try {

//            System.out.println(a+fun);

            int n=a.length();

            String ans;
            if(n<=1) {

                ans=a;

                request.setAttribute("ans",ans);

            request.getRequestDispatcher("index.jsp").forward(request,response);

        }
    }
}
```



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

Use Case Learning Project

```
else {  
    int len=1,s=0;  
    int low,high;  
    for(int i=1;i<n;i++) {  
        low=i-1;  
        high=i+1;  
        while(high<n&& a.charAt(high)==a.charAt(i)) {  
            high++;  
        }  
        while(high<n&& a.charAt(low)==a.charAt(i)) {  
            low++;  
        }  
        while(low>=0 && high<n &&  
a.charAt(low)==a.charAt(high)) {  
            low--;  
            high++;  
        }  
        int length=high-low-1;  
        if(len<length) {  
            len=length;  
            s=low+1;  
        }  
    }  
    ans=a.substring(s,s+len);  
}
```



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

Distance Learning Program

```
        request.setAttribute("ans",ans);

request.getRequestDispatcher("index.jsp").forward(request,response);

    }

    }catch(Exception e) {

        System.out.println(e);

    }

}

}
```

OUTPUT:

**find the Longest Palindromic Substring**

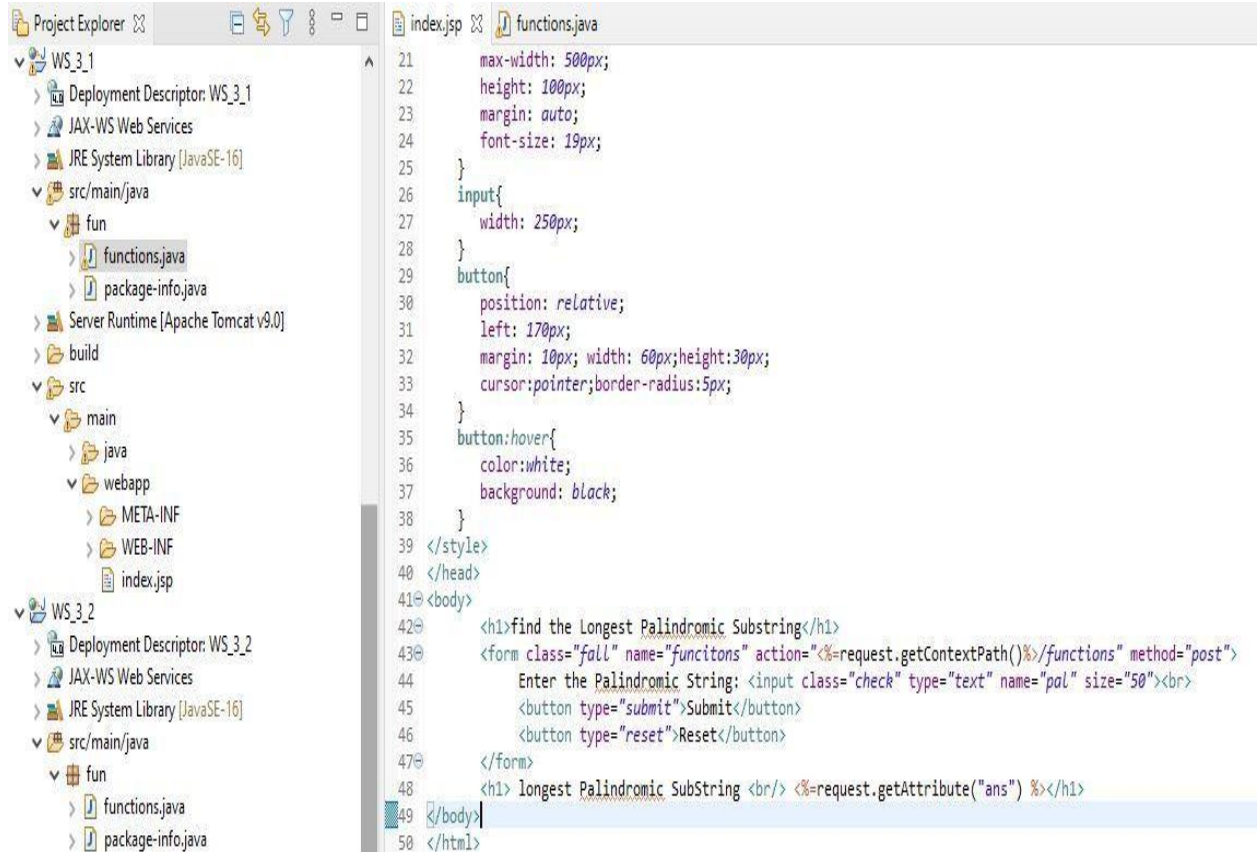
Enter the Palindromic String:

**longest Palindromic SubString**  
**hiihih**

**find the Longest Palindromic Substring**

Enter the Palindromic String:

**longest Palindromic SubString**  
**null**



The screenshot shows an IDE with a project explorer on the left and a code editor on the right. The project explorer displays a web application structure with two workspaces, WS\_3\_1 and WS\_3\_2. WS\_3\_1 contains a deployment descriptor, JAX-WS Web Services, JRE System Library, and source files for a 'fun' package. WS\_3\_2 contains a deployment descriptor, JAX-WS Web Services, JRE System Library, and source files for a 'fun' package. The code editor shows the 'functions.java' file, which contains CSS styles and HTML code for a web form. The form is titled 'find the Longest Palindromic Substring' and includes an input field for a palindromic string, a submit button, and a reset button. The code also includes a JavaScript function to find the longest palindromic substring and display the result.

```
21 max-width: 500px;
22 height: 100px;
23 margin: auto;
24 font-size: 19px;
25 }
26 input{
27 width: 250px;
28 }
29 button{
30 position: relative;
31 left: 170px;
32 margin: 10px; width: 60px; height: 30px;
33 cursor: pointer; border-radius: 5px;
34 }
35 button: hover{
36 color: white;
37 background: black;
38 }
39 </style>
40 </head>
41 <body>
42 <h1>find the Longest Palindromic Substring</h1>
43 <form class="fall" name="funcitons" action="<%=request.getContextPath()%>/functions" method="post">
44 Enter the Palindromic String: <input class="check" type="text" name="pal" size="50"><br>
45 <button type="submit">Submit</button>
46 <button type="reset">Reset</button>
47 </form>
48 <h1> longest Palindromic SubString <br/> <%=request.getAttribute("ans") %></h1>
49 </body>
50 </html>
```

```
index.jsp  functions.java
1 package fun;
2
3 import java.io.IOException;
4
5
6
7
8
9
10 /**
11  * Servlet implementation class functions
12  */
13 @WebServlet(name="functions",urlPatterns={"/functions"})
14 public class functions extends HttpServlet {
15     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
16         String a=request.getParameter("pal");
17         String fun=request.getParameter("fun");
18         try {
19             // System.out.println(a+fun);
20             int n=a.length();
21             String ans;
22             if(n<=1) {
23                 ans=a;
24                 request.setAttribute("ans",ans);
25                 request.getRequestDispatcher("index.jsp").forward(request,response);
26             }
27             else {
28                 int len=1,s=0;
29                 int low,high;
30                 for(int i=1;i<n;i++) {
31                     low=i-1;
32                     high=i+1;
33                     while(high<n&&a.charAt(high)==a.charAt(i)) {
34                         high++;
35                     }
36                 }
37             }
38         }
39     }
40 }
```

### Learning outcomes (What I have learnt):

1. Learn About the servlet
2. Learn about jsp and dynamic web project
3. Learn about the tomcat server and its integrations with the java.