



# Introduction to Resquest Dispatcher | Servlet Tutorial

## Introduction to Request Dispatcher

**RequestDispatcher** is an interface, implementation of which defines an object which can dispatch request to any resources(such as HTML, Image, JSP, Servlet) on the server.

### Methods of RequestDispatcher

**RequestDispatcher** interface provides two important methods

| Methods                                                                     | Description                                                                                           |
|-----------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| void <code>forward(ServletRequest request, ServletResponse response)</code> | forwards a request from a servlet to another resource (servlet, JSP file, or HTML file) on the server |
| void <code>include(ServletRequest request, ServletResponse response)</code> | includes the content of a resource (servlet, JSP page, HTML file) in the response                     |

### How to get an Object of RequestDispatcher

`getRequestDispatcher()` method of **ServletRequest** returns the object of **RequestDispatcher**.

```
RequestDispatcher rs = request.getRequestDispatcher("hello.html");
rs.forward(request,response);
```



OR

```
RequestDispatcher rs = request.getRequestDispatcher("hello.html");
rs.include(request,response);
```



### Example demonstrating usage of RequestDispatcher

In this example, we will show you how RequestDispatcher is used to **forward** or **include** response of a resource in a Servlet. Here we are using **index.html** to get username and password from the user, **Validate** Servlet will validate the password entered by the user, if the user has entered "studytonight" as password, then he will be forwarded to **Welcome** Servlet else the user will stay on the index.html page and an error message will be displayed.

#### Files to be created :

- **index.html** will have form fields to get user information.
- **Validate.java** will validate the data entered by the user.
- **Welcome.java** will be the welcome page.

- **web.xml** , the deployment descriptor.

index.html

```
<form method="post" action="Validate">
Name:<input type="text" name="user" /><br/>
Password:<input type="password" name="pass" ><br/>
<input type="submit" value="submit">
</form>
```

Validate.java

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

public class Validate extends HttpServlet {

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            String name = request.getParameter("user");
            String password = request.getParameter("pass");

            if(password.equals("studytonight"))
            {
                RequestDispatcher rd = request.getRequestDispatcher("Welcome");
                rd.forward(request, response);
            }
            else
            {
                out.println("<font color='red'><b>You have entered incorrect password</b></font>");
                RequestDispatcher rd = request.getRequestDispatcher("index.html");
                rd.include(request, response);
            }
        }finally {
            out.close();
        }

    }
}
```

Welcome.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Welcome extends HttpServlet {

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {

            out.println("<h2>Welcome user</h2>");
        } finally {
            out.close();
        }

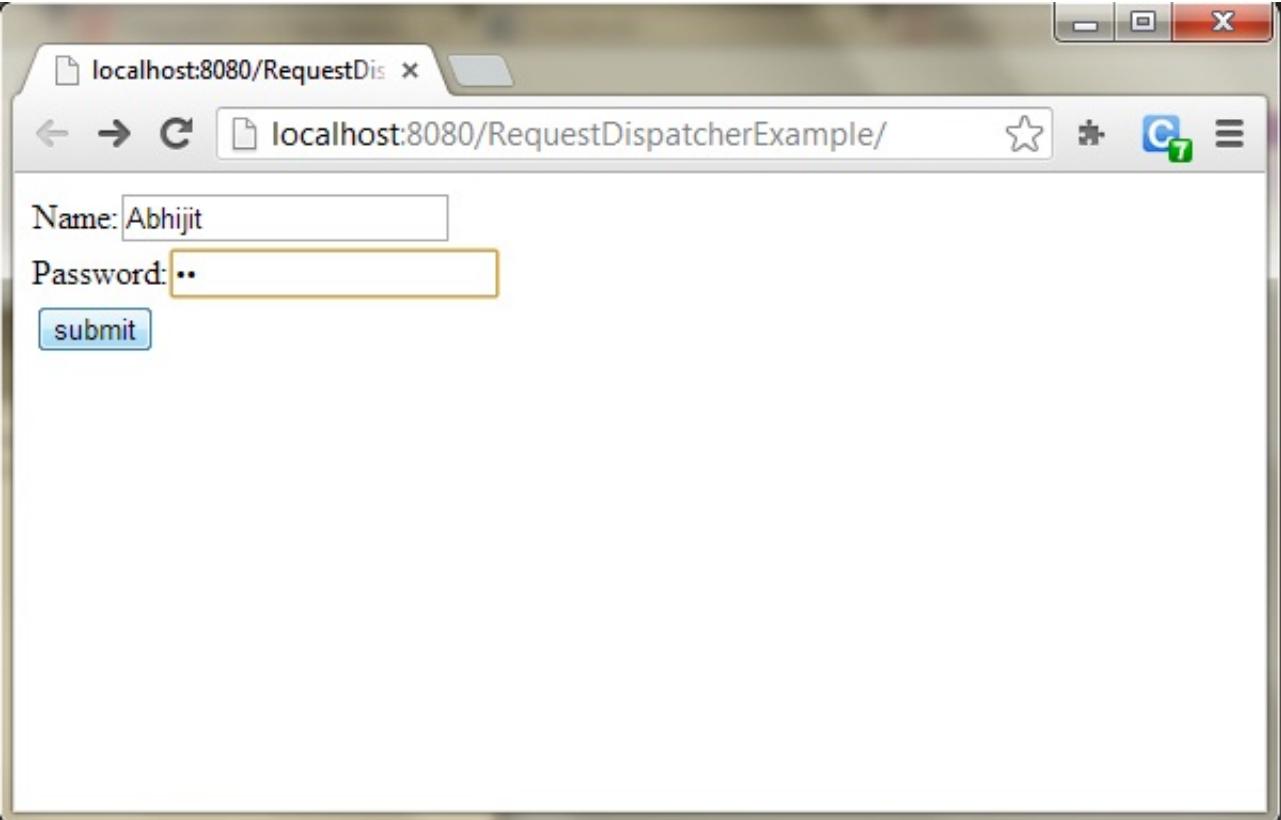
    }
}
```

web.xml

```
<web-app>
    <servlet>
```

```
<servlet-name>Validate</servlet-name>
<servlet-class>Validate</servlet-class>
</servlet>
<servlet>
  <servlet-name>Welcome</servlet-name>
  <servlet-class>Welcome</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>Validate</servlet-name>
  <url-pattern>/Validate</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>Welcome</servlet-name>
  <url-pattern>/Welcome</url-pattern>
</servlet-mapping>
<welcome-file-list>
  <welcome-file>index.html</welcome-file>
</welcome-file-list>
</web-app>
```

This will be the first screen. You can enter your Username and Password here.



localhost:8080/RequestDis x

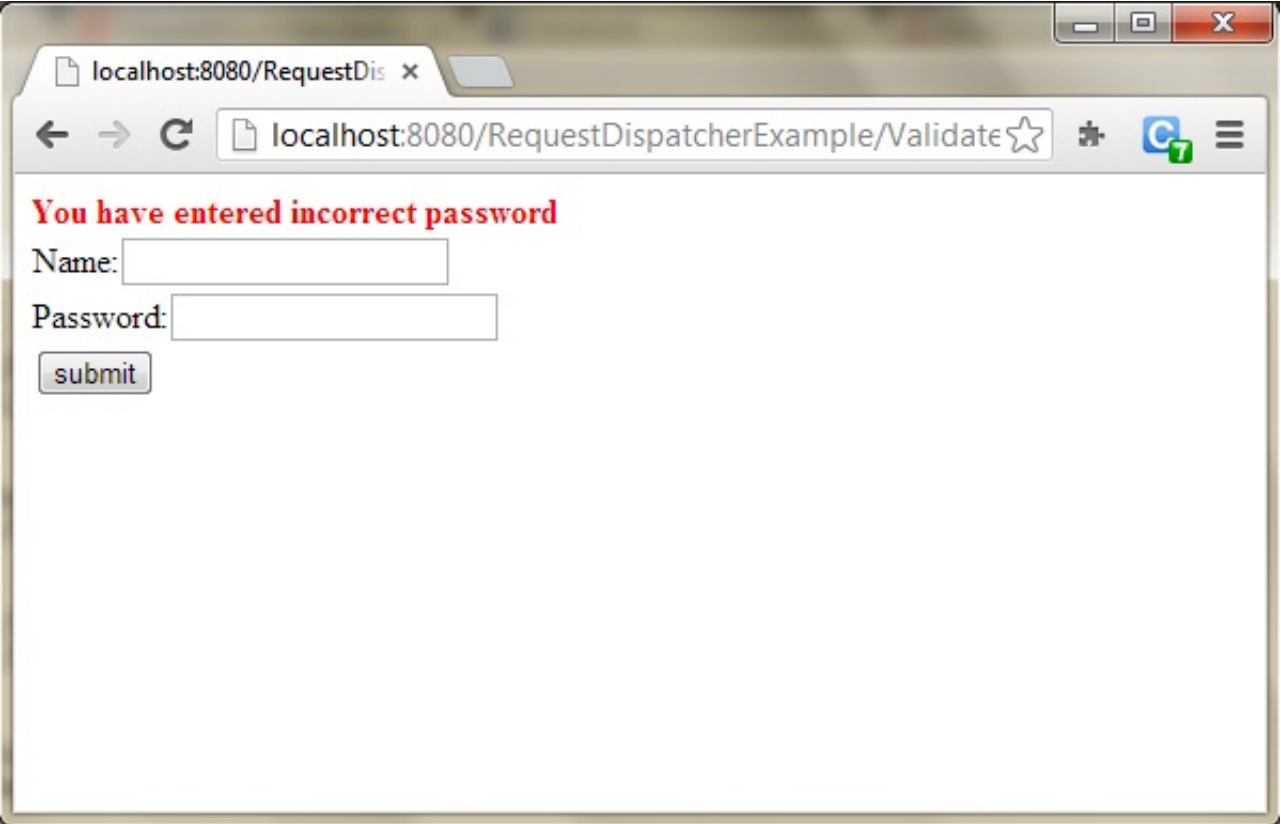
localhost:8080/RequestDispatcherExample/

Name: Abhijit

Password: ..

submit

When you click on Submit, Password will be validated, if it is not 'studytonight' , error message will be displayed.



localhost:8080/RequestDis x

localhost:8080/RequestDispatcherExample/Validate

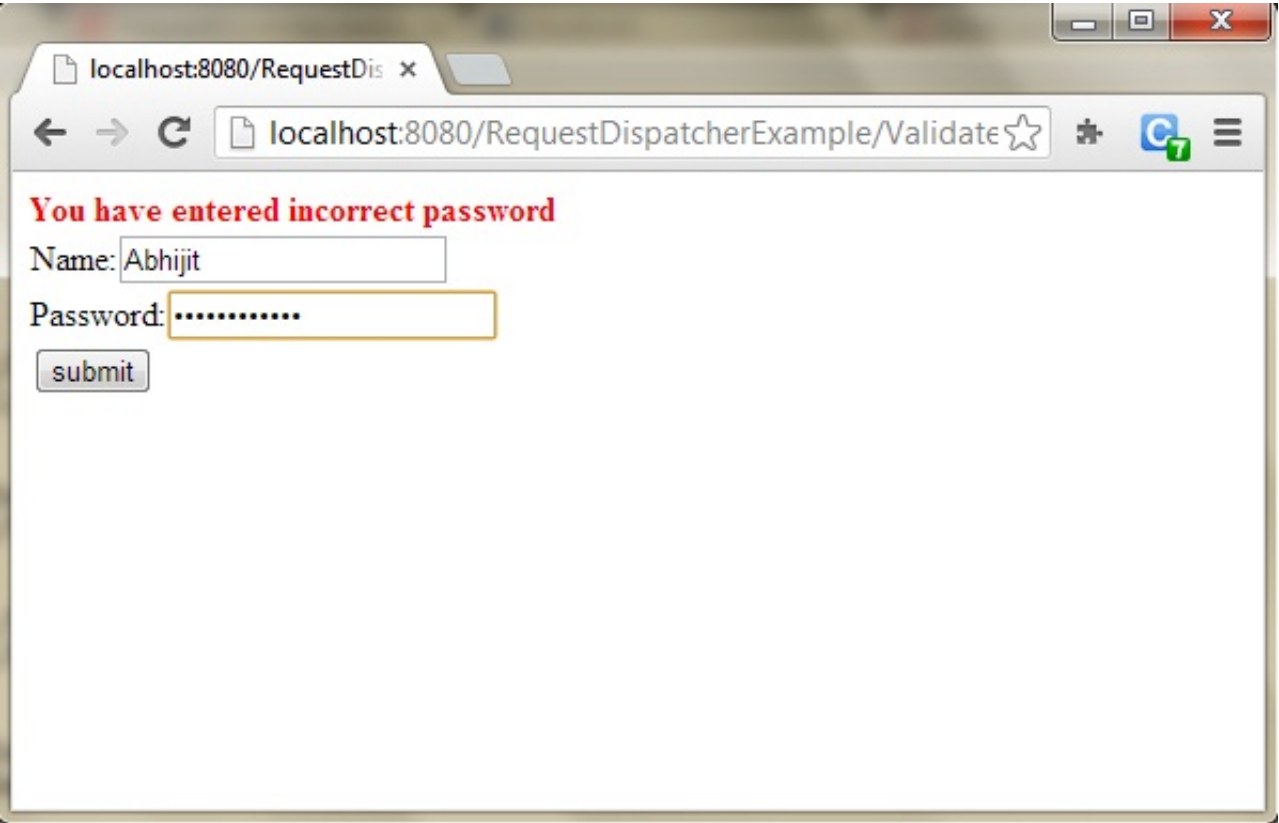
**You have entered incorrect password**

Name:

Password:

submit

Enter any Username, but enter 'studytonight' as password.



Password will be successfully validated and you will be directed to the Welcome Servlet.

