If source servlet comp wants to talk with dest servlet comp , we should write a logic to create object of Dest servlet comp in source servlet comp and invoke the service(-,-) method because servlet container only has to take care creating dest web comp object and implementing dest web comp object life cycle.

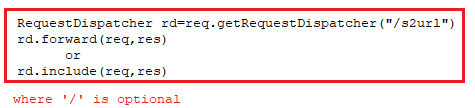
The communication between two web components happens through the “RequestDispatcher” object.

Different ways to create RequestDispatcher obj

=================================

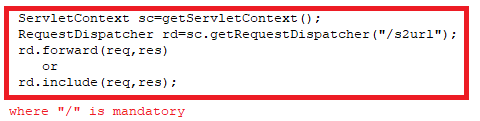
The RequestDispatcher object is created in source servlet comp.

Approach1: Using request object

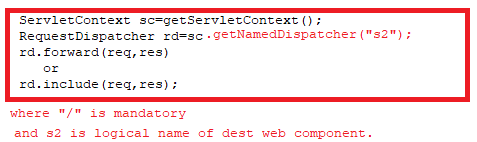


Approach2 : using Servlet Context object.

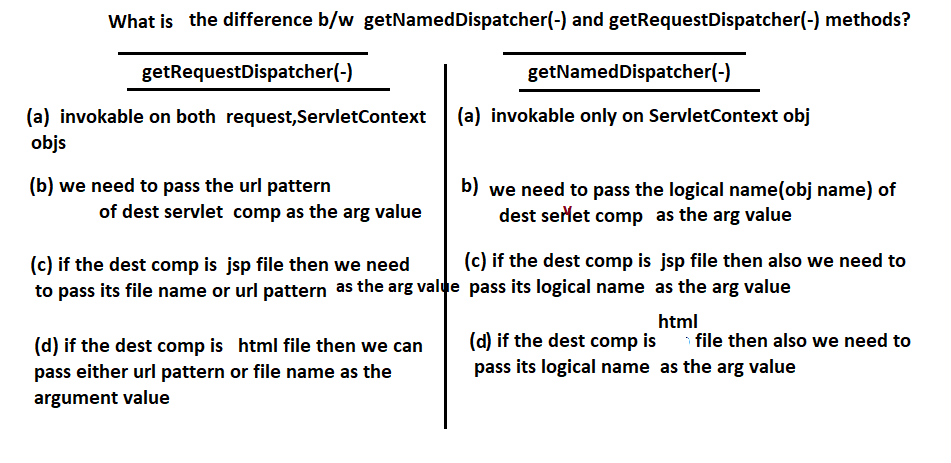
Method1:



Method2:





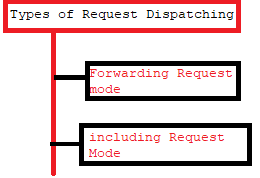


1.Types of Request Dispatching:

=====================

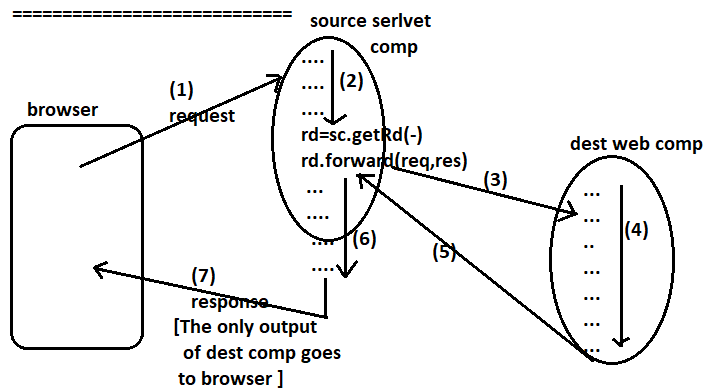
a. Forwarding Request mode Request Dispatching.

b. including Response mode Request Dispatching.



1.1Forwarding Request Mode:

===================



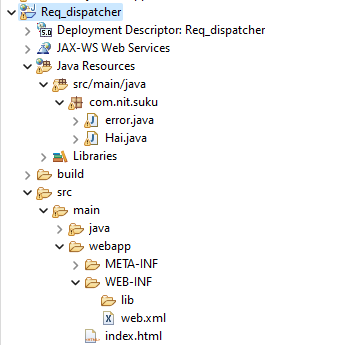
* Rd.forward(-,-) performs forwarding request mode of servlet Communication.
* The Source servlet comp and dest web comp uses the same req,res objs So the request data given to source servlet comp is visible and accessible in dest comp.
* To pass additional data from source servlet comp to dest comp use request attributes.
* The source servlet comp and dest web comp can be there either in same web application or in two different web application of same server.
* Source servlet comp directly interacts with dest comp.
* The dest comp can be a servlet comp or jsp comp or html file.
* The statements placed before and after rd.forward(-,-) in source servlet executes but their response will be discareded.
* Always place rd.forward(-,-) in source servlet comp as conditional statement to execute.

UseCase::

The servlet comp that executes only for the exception raised in other main servlet comp is called Error Servlet comp.

The ErrorServlet comp is useful to display the exception related messages for end user as non-technical guiding messages by discarding partial output generated by MainServlet comp before raising the exception.

Example:-



Index.html

========

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h2 align="center">Request Dispatcher Example</h2>

<form action="./Hai" method="get">

Enter a value:<input type="text" name="v1"/>

<input type="submit"/>

</form>

</body>

</html>

Hai.java

======

**package** com.nit.suku;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** jakarta.servlet.RequestDispatcher;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**public** **class** Hai **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest req, HttpServletResponse res) **throws** ServletException, IOException {

String v1=req.getParameter("v1");

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

**if**(v1.equals("suku"))

{

RequestDispatcher rd=req.getRequestDispatcher("/error");

rd.forward(req, res);

pw.write("Control came back to main servlet");

}

**else** {

pw.write("<h2>"+v1+"</h2>");

}

}

}

Error.java

========

**package** com.nit.suku;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.annotation.WebServlet;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**public** **class** error **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse res) **throws** ServletException, IOException {

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

pw.write("<h2>"+"EnteredWrongValue"+"</h2>");

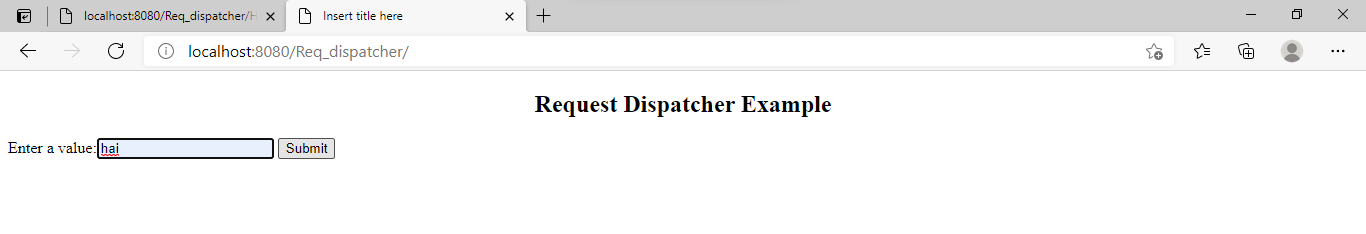
}

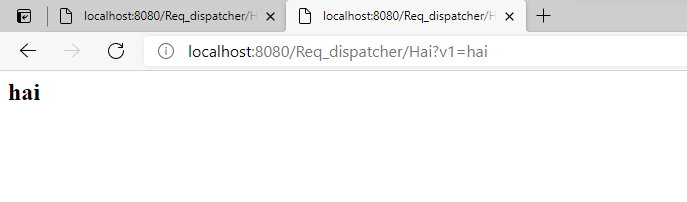
}

Output:

Run1:

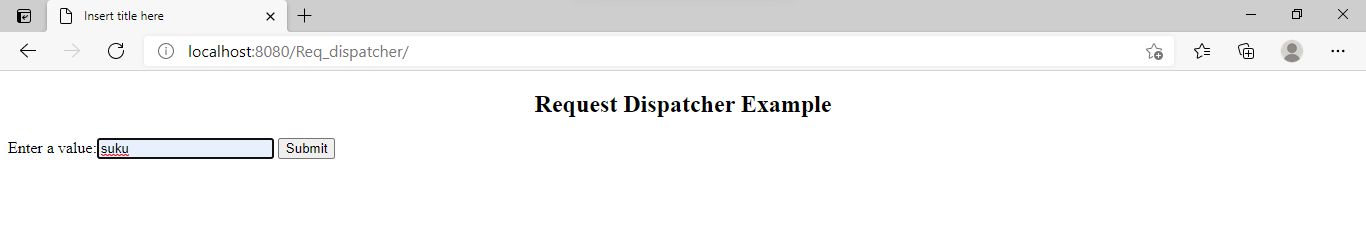
====

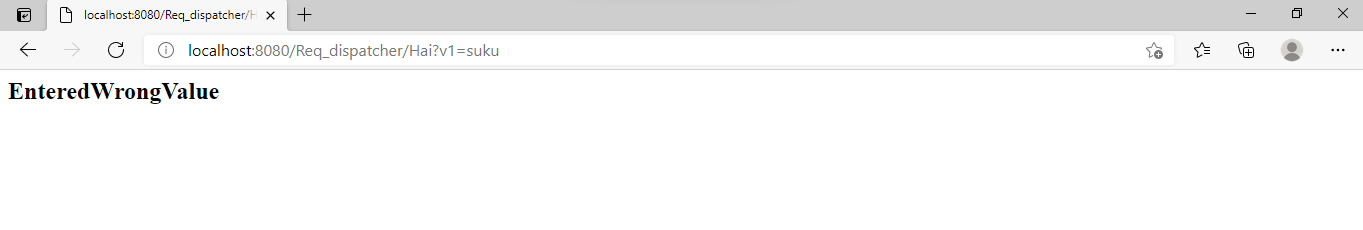


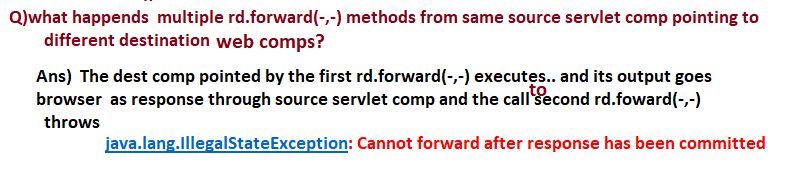


Run2:

====

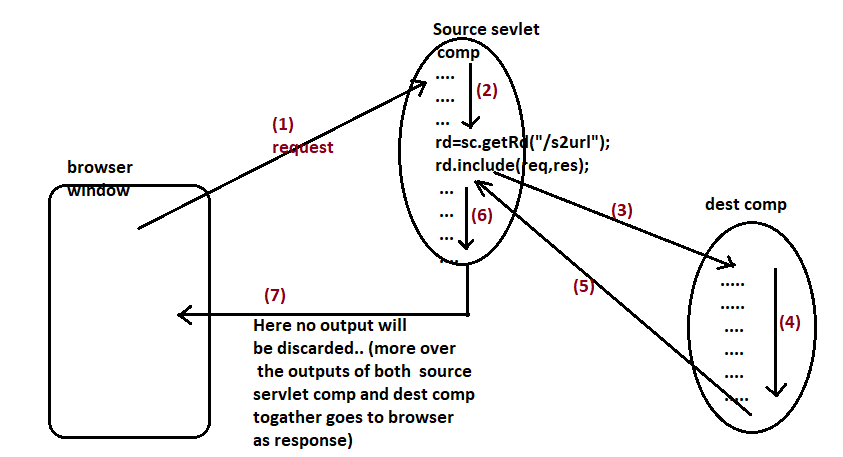






1.2.Include Request mode:

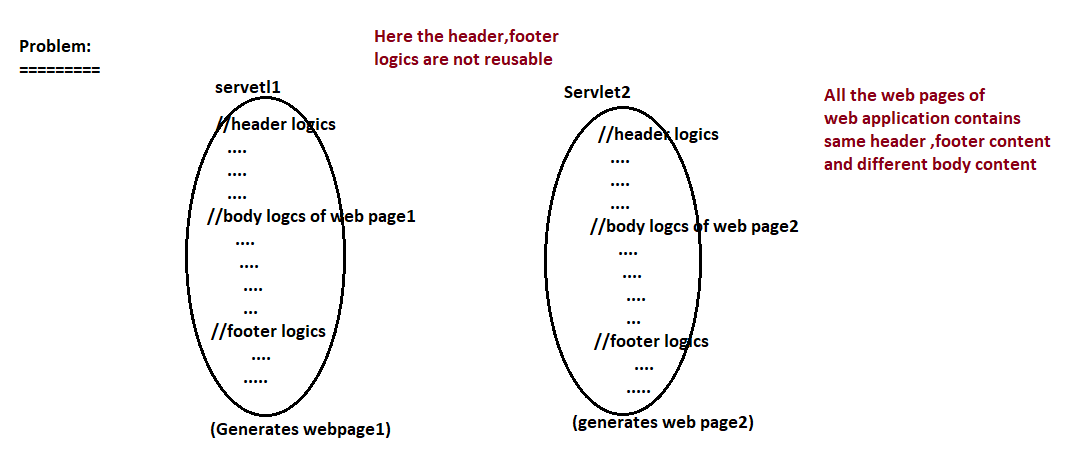
===================

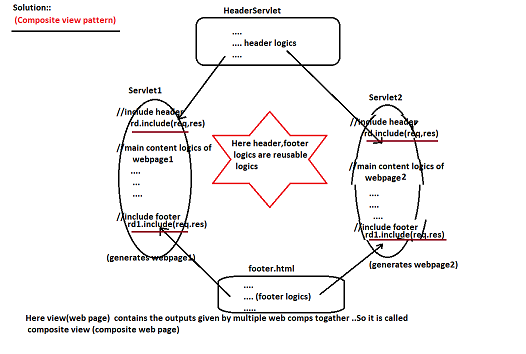


* The source servlet comp and the dest comp uses same req,res objs, so the request data coming to source servlet comp is visible and accessible in dest web comp.
* If source servlet comp wants to send additional data to dest comp then use request attributes support.
* The response contains the outputs of both source servlet comp and dest comp.
* The dest comp output will be included to the source servlet comp in the place where rd.include(-,-) is called.

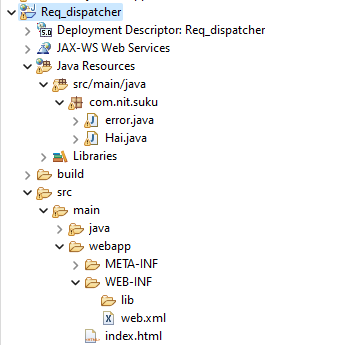
If rd.include(-,-) is called in the middle of source servlet comp then the dest comp output will be included in the middle of source servlet comp.

UseCase:: keeping common logics like header,footer logics in separate web comp and including their outputs in main web comps.





Example:-



Index.html

========

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h2 align="center">Request Dispatcher Example</h2>

<form action="./Hai" method="get">

Enter a value:<input type="text" name="v1"/>

<input type="submit"/>

</form>

</body>

</html>

Hai.java

======

**package** com.nit.suku;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** jakarta.servlet.RequestDispatcher;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**public** **class** Hai **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest req, HttpServletResponse res) **throws** ServletException, IOException {

String v1=req.getParameter("v1");

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

**if**(v1.equals("suku"))

{

RequestDispatcher rd=req.getRequestDispatcher("/error");

pw.write("<h2>Before include method in Source servlet</h2>");

rd.include(req, res);

pw.write("<h2>After Include Method in source servlet</h2>");

}

**else** {

pw.write("<h2>"+v1+"</h2>");

}

}

}

Error.java

=======

**package** com.nit.suku;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.annotation.WebServlet;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**public** **class** error **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse res) **throws** ServletException, IOException {

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

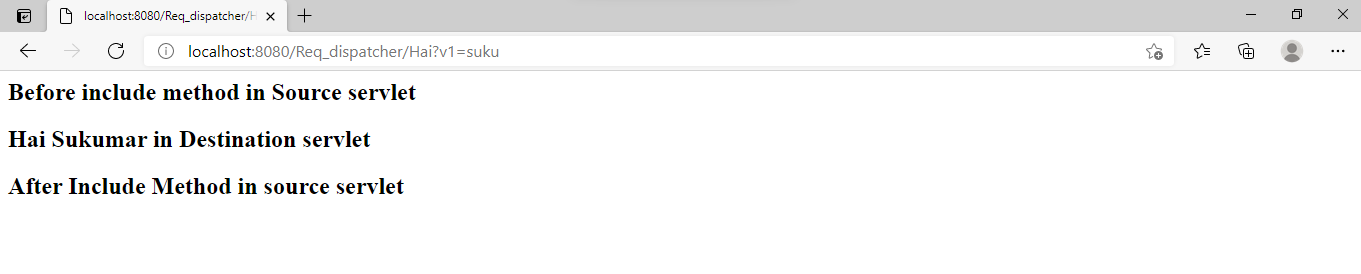
pw.write("<h2>"+"Hai Sukumar in Destination servlet"+"</h2>");

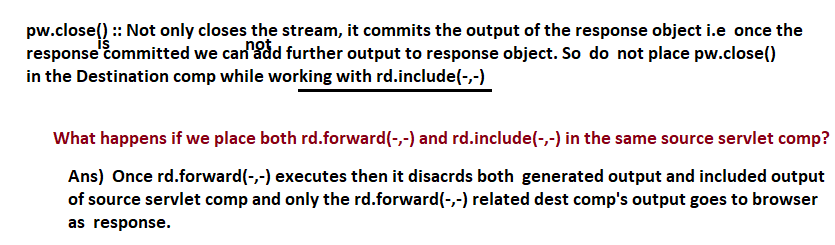
}

}

Output:-

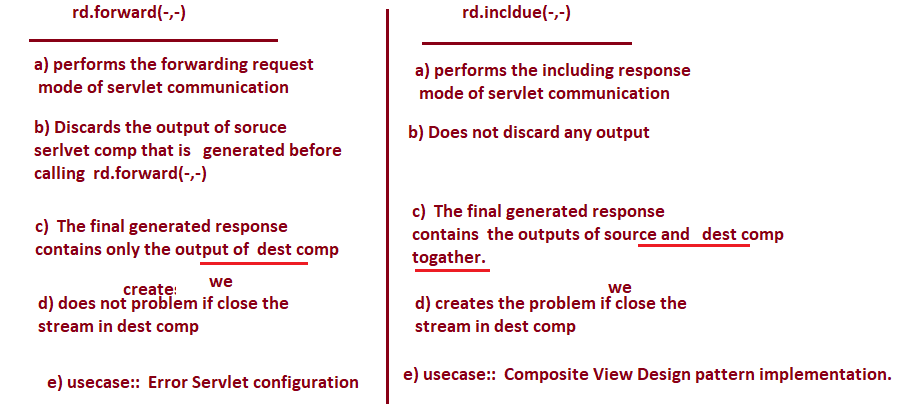






Diff between forward(-,-) and include(-,-) method?

===================================



Limitations of Request Dispatching based servlet communication:

=============================================

1. The source servlet comp and dest comp can be there either in same web application or in two different web applications of same server. But can not be there in two different web applications of two different servers belonging to same or different machines.
2. The dest web com must be html file or jsp file or servlet comp. It can not be in other than java technologies web comp like php comp, asp.net comp, express js comp and etc…

To overcome these problems take the support of send Redirection.