



Java Servlets

CSCI 201

Principles of Software Development

Jeffrey Miller, Ph.D.
jeffrey.miller@usc.edu



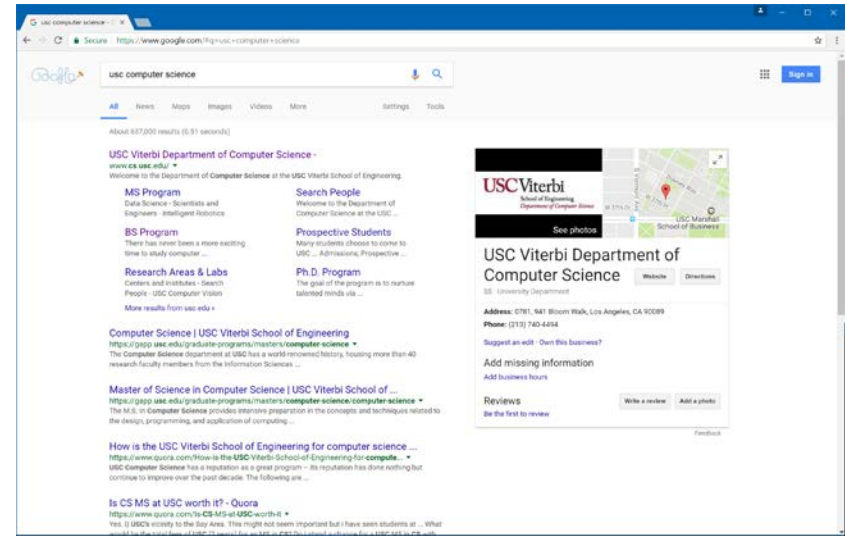
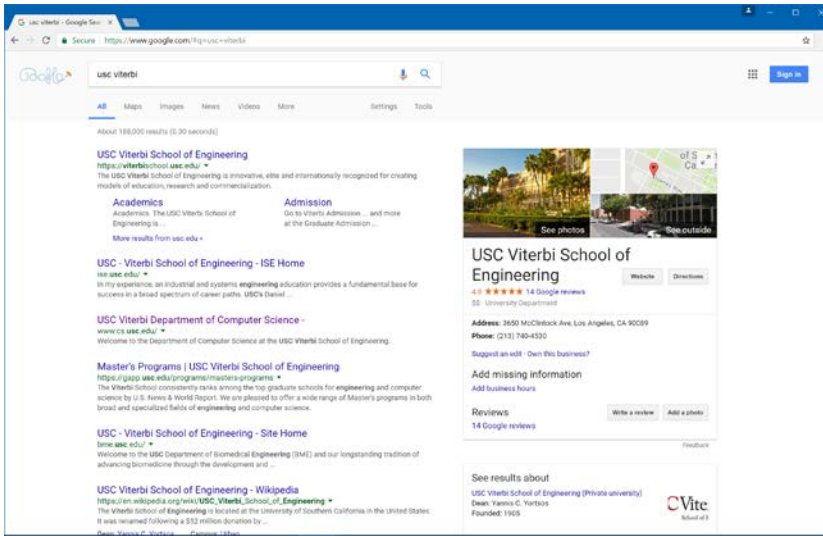
Outline

- Java Servlets
- Program

Dynamic Web Content



- Dynamic web content allows different data to be shown on the same web page
 - For example, the Google search result page is the same page that loads different data based on the search terms



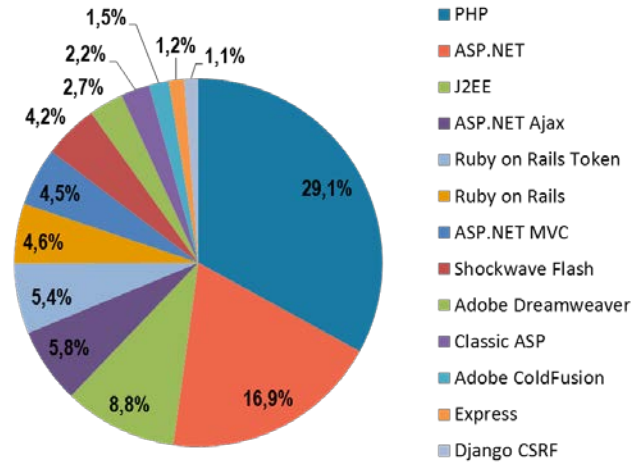
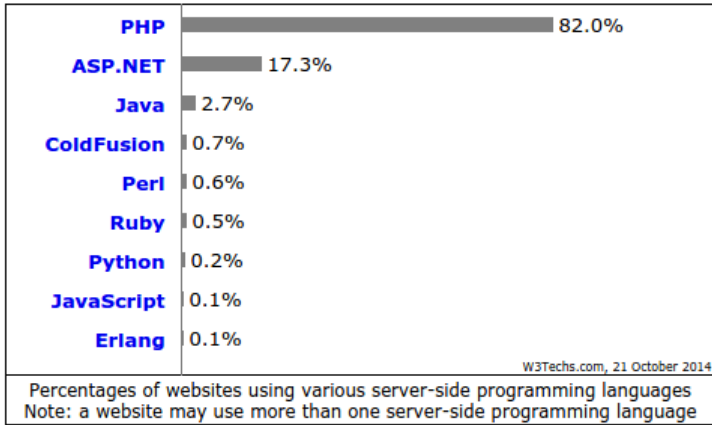
Back-End Web Languages



- Most dynamic content is based on a server program running and generating the front-end code in real-time
- There are many back-end web languages
 - › Java – Servlets, JSPs, JavaBeans, Enterprise Java Beans (EJBs)
 - › .NET – C#, VB
 - › PHP
 - › Ruby
 - › Python
 - › Server-side JavaScript – node.js
 - › CGI – C, Perl

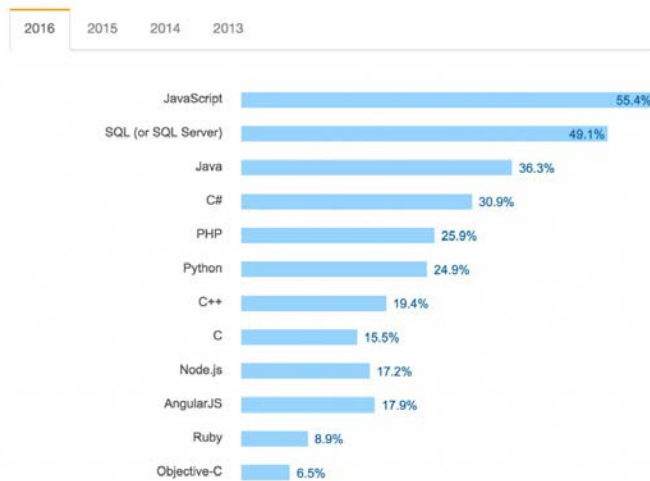


Which Language to Choose?

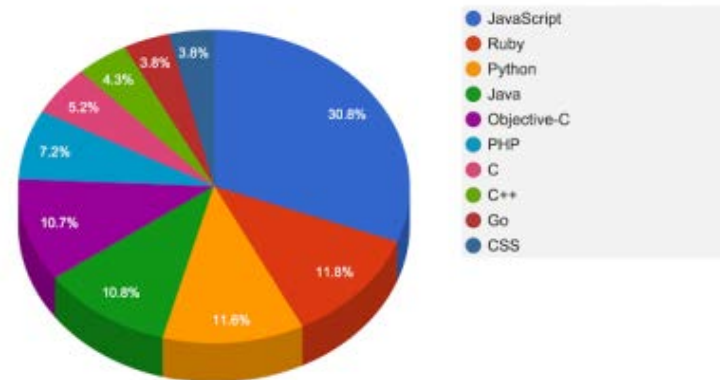


Source: [Builtwith](#)

I. Most Popular Technologies



Programming Language Popularity By Github Projects



Servlet 3-Tier Architecture

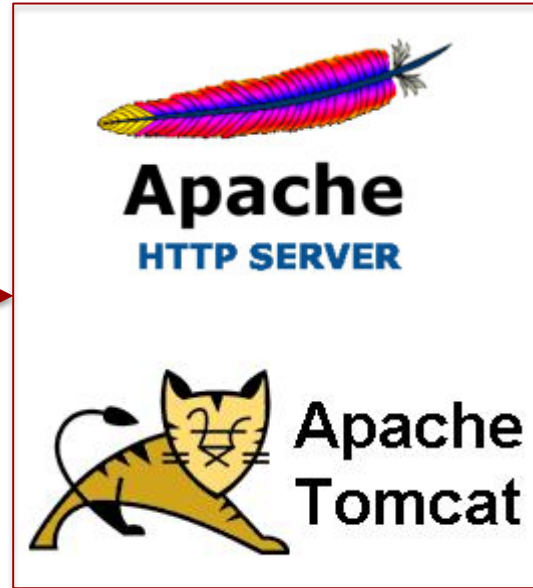


Client



Server

Web/Application Server



Database



Java Servlets



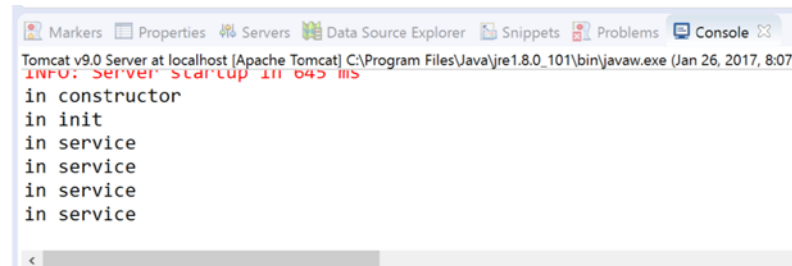
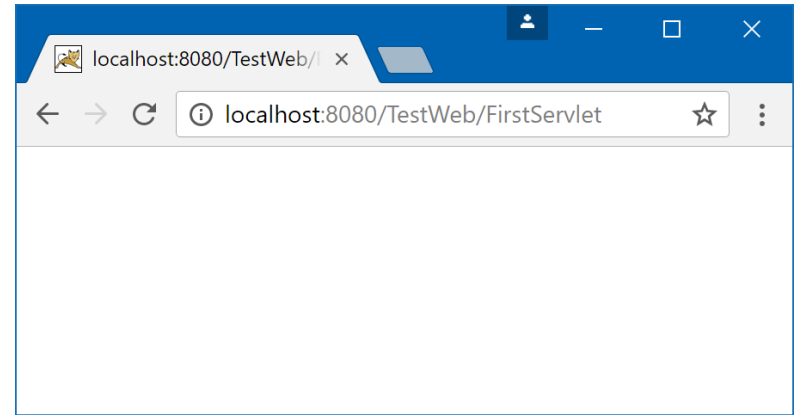
- Servlets are Java classes that can serve dynamic web content
- Servlets get compiled and executed on the server and generate client-side code to send back out to the browser
 - › Client-side code is HTML, CSS, and JavaScript
- To create a servlet, extend the `HttpServlet` class
 - › `void doGet(HttpServletRequest req, HttpServletResponse resp)`
 - › `void doPost(HttpServletRequest req, HttpServletResponse resp)`
 - › `void service(HttpServletRequest req, HttpServletResponse resp)`
 - Dispatches to `doGet` or `doPost` if not overridden
 - › `void init(ServletConfig config)`



My First Servlet



```
1 package csci201;
2 import java.io.IOException;
3 import javax.servlet.ServletConfig;
4 import javax.servlet.ServletException;
5 import javax.servlet.annotation.WebServlet;
6 import javax.servlet.http.*;
7
8 @WebServlet("/FirstServlet")
9 public class TestServlet extends HttpServlet {
10     private static final long serialVersionUID = 1L;
11     public TestServlet() {
12         super();
13         System.out.println("in constructor");
14     }
15     public void init(ServletConfig config) throws ServletException {
16         System.out.println("in init");
17     }
18     protected void service(HttpServletRequest request, HttpServletResponse response)
19         throws ServletException, IOException {
20         System.out.println("in service");
21     }
22 }
```



Annotations



- Annotations allow us to specify configuration parameters that are used by the application server in our .java files
- Before annotations (or even now if you want), you would have to modify the web.xml file of the application server

```
1 <servlet>
2   <servlet-name>mytest</servlet-name>
3   <init-param>
4     <param-name>n1</param-name>
5     <param-value>v1</param-value>
6   </init-param>
7   <init-param>
8     <param-name>n2</param-name>
9     <param-value>v2</param-value>
10  </init-param>
11 </servlet>
12 <servlet-mapping>
13   <servlet-name>mytest</servlet-name>
14   <url-mapping>/myurl</url-mapping>
15 </servlet-mapping>
```

- Instead, we can write the following line immediately above the class declaration of our servlet

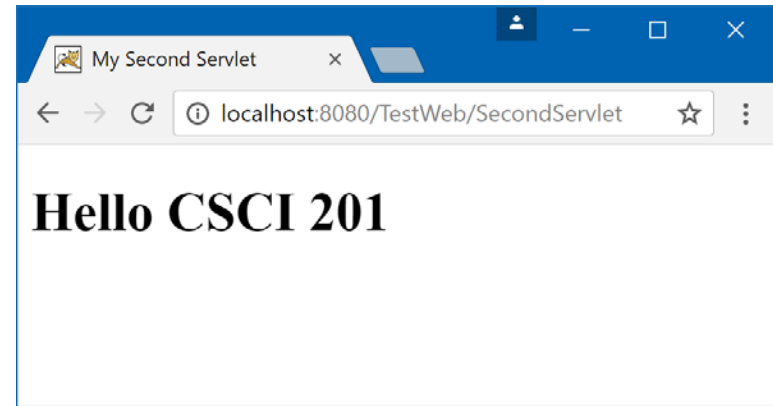
```
1 @WebServlet(name="mytest",
2             urlPatterns={"/myurl"},
3             initParams={
4               @InitParam(name="n1",
5                           value="v1"),
6               @InitParam(name="n2",
7                           value="v2")
8             })
```



My Second Servlet



```
1 package csci201;
2 import java.io.IOException;
3 import java.io.PrintWriter;
4 import javax.servlet.ServletException;
5 import javax.servlet.annotation.WebServlet;
6 import javax.servlet.http.*;
7
8 @WebServlet("/SecondServlet")
9 public class TestServlet extends HttpServlet {
10     private static final long serialVersionUID = 1L;
11     protected void service(HttpServletRequest request, HttpServletResponse response)
12         throws ServletException, IOException {
13         response.setContentType("text/html");
14         PrintWriter out = response.getWriter();
15         out.println("<!DOCTYPE html>");
16         out.println("<html>");
17         out.println("<head>");
18         out.println("<title>My Second Servlet</title>");
19         out.println("</head>");
20         out.println("<body>");
21         out.println("<h1>Hello CSCI 201</h1>");
22         out.println("</body>");
23         out.println("</html>");
24     }
25 }
```

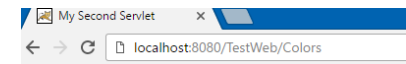


Color Servlet



```
1 // omitted package and imports for space
2 @WebServlet("/Colors")
3 public class TestServlet extends HttpServlet {
4     private static final long serialVersionUID = 1L;
5     private String getColor(int r, int g, int b) {
6         String color = "";
7         color += makeHex(r);
8         color += makeHex(g);
9         color += makeHex(b);
10        return color;
11    }
12    private String makeHex(int color) {
13        String hexString = Integer.toHexString(color);
14        if (hexString.length() == 1) {
15            hexString = "0" + hexString;
16        }
17        return hexString;
18    }
19    protected void service(HttpServletRequest request,
20                          HttpServletResponse response)
21        throws ServletException, IOException {
22        response.setContentType("text/html");
23        PrintWriter out = response.getWriter();
24        out.println("<!DOCTYPE html>");
25        out.println("<html>");
26        out.println("<head>");
27        out.println("<title>My Second Servlet</title>");
28        out.println("</head>");
29        out.println("<body>");
30        out.println("<h1>Color Table</h1>");
```

```
31        out.println("<table>");
32        out.println("<tr><th>Red</th>");
33        out.println("<th>Green</th>");
34        out.println("<th>Blue</th>");
35        out.println("<th>Color</th></tr>");
36        for (int red=0; red < 255; red+=50) {
37            for (int green=0; green < 255; green+=50) {
38                for (int blue=0; blue < 255; blue+=50) {
39                    out.println("<tr>");
40                    out.print("<td>" + red + "</td>");
41                    out.print("<td>" + green + "</td>");
42                    out.print("<td>" + blue + "</td>");
43                    String color = getColor(red, green, blue);
44                    out.print("<td style=\\\"background-color:#\" + color + \";\>");
45                    out.print("</td>");
46                    out.println("</tr>");
47                }
48            }
49        }
50        out.println("</table>");
51        out.println("</body>");
52        out.println("</html>");
53    }
54 }
```



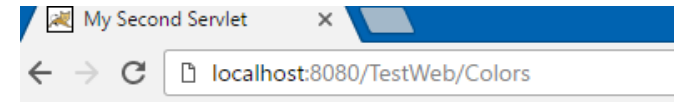
Color Table

Red	Green	Blue	Color
0	0	0	Black
0	0	50	Dark Blue
0	0	100	Blue
0	0	150	Light Blue
0	0	200	Very Light Blue
0	0	250	White
0	50	0	Dark Green
0	50	50	Green
0	50	100	Light Green
0	50	150	Very Light Green
0	50	200	White
0	50	250	White
0	100	0	Dark Red
0	100	50	Red
0	100	100	Light Red
0	100	150	Very Light Red
0	100	200	White
0	100	250	White
0	150	0	Dark Yellow
0	150	50	Yellow

Color Servlet Generated HTML



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>My Second Servlet</title>
5 </head>
6 <body>
7 <h1>Color Table</h1>
8 <table>
9 <tr><th>Red</th><th>Green</th><th>Blue</th><th>Color</th></tr>
10 <tr>
11 <td>0</td>
12 <td>0</td>
13 <td>0</td>
14 <td style="background-color:#000000;"> </td>
15 </tr>
16 <tr>
17 <td>0</td>
18 <td>0</td>
19 <td>50</td>
20 <td style="background-color:#000032;"> </td>
21 </tr>
22 <tr>
23 <td>0</td>
24 <td>0</td>
25 <td>100</td>
26 <td style="background-color:#000064;"> </td>
27 </tr>
...
1306 </table>
1307 </body>
1308 </html>
```



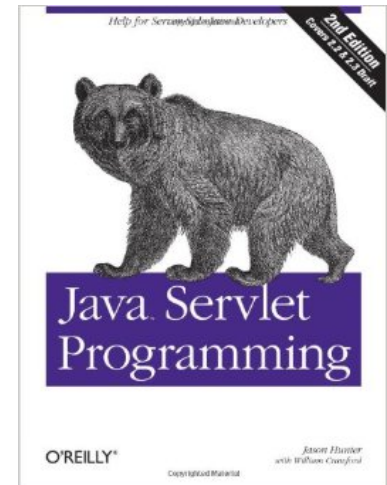
Color Table

Red	Green	Blue	Color
0	0	0	
0	0	50	
0	0	100	
0	0	150	
0	0	200	
0	0	250	
0	50	0	
0	50	50	
0	50	100	
0	50	150	
0	50	200	
0	50	250	
0	100	0	
0	100	50	
0	100	100	
0	100	150	
0	100	200	
0	100	250	
0	150	0	
0	150	50	

Servlets and HTML Forms



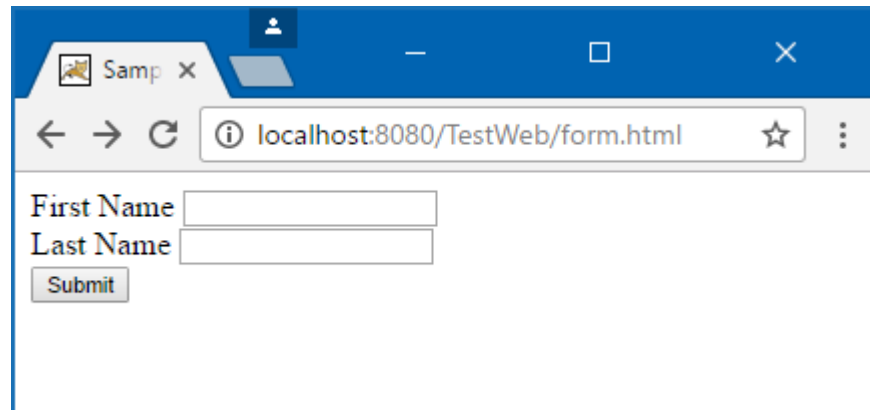
- Servlets can be used to process the data submitted from an HTML form through the `HttpServletRequest` variable in the `doGet`, `doPost`, or `service` method
- Here are a few of the more commonly used methods
 - › `Cookie[] getCookies()`
 - › `String getQueryString()`
 - › `HttpSession getSession()`
 - › `String getParameter(String)`
 - › `Enumeration<String> getParameterNames()`
 - › `String[] getParameterValues(String)`
 - Returns all of the values associated with a specific parameter name



Servlet Form Example



```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Sample Form</title>
5    </head>
6    <body>
7      <form name="myform" method="GET" action="FormServlet">
8        First Name <input type="text" name="fname" /><br />
9        Last Name <input type="text" name="lname" /><br />
10       <input type="submit" name="submit" value="Submit" />
11     </form>
12   </body>
13 </html>
```

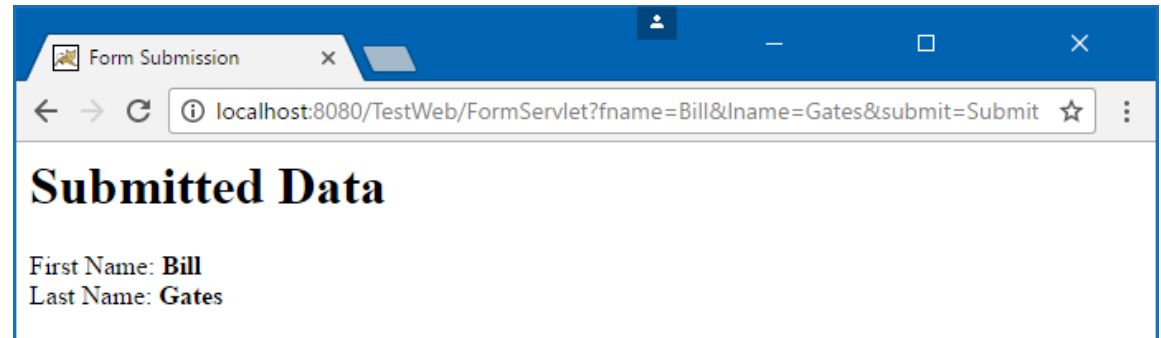


Servlet Form Example



```
1 // package and imports omitted for space
2 @WebServlet("/FormServlet")
3 public class FormServlet extends HttpServlet {
4     private static final long serialVersionUID = 1L;
5     protected void service(HttpServletRequest request, HttpServletResponse response)
6         throws ServletException, IOException {
7         PrintWriter out = response.getWriter();
8         String fname = request.getParameter("fname");
9         String lname = request.getParameter("lname");
10        System.out.println("fname = " + fname);
11        System.out.println("lname = " + lname);
12
13        response.setContentType("text/html");
14        out.println("<html>");
15        out.println("<head><title>Form Submission</title></head>");
16        out.println("<body>");
17        out.println("<h1>Submitted Data</h1>");
18        out.println("First Name:<strong> " + fname + "</strong><br />");
19        out.println("Last Name:<strong> " + lname + "</strong>");
20        out.println("</body>");
21        out.println("</html>");
22    }
23 }
```

Tomcat v9.0 Server at localhost [Apache Tomcat] C:\Program Files\Java\jre1.8.0_101\bin\javaw.exe (Jan 27, 2017, 1:36:43 PM)
Jan 27, 2017 1:37:05 PM org.apache.catalina.core.StandardContext reload
INFO: Reloading Context with name [/TestWeb] has started
Jan 27, 2017 1:37:05 PM org.apache.catalina.core.StandardContext reload
INFO: Reloading Context with name [/TestWeb] is completed
fname = Bill
lname = Gates



Forwarding from a Servlet



- Because servlets have a lot of overhead when generating client-side code, forwarding to a different page is often used
 - › The request and response objects can be forwarded to the page too
 - › The servlet can do some processing of the data, possibly even modify or amend it, then forward to another page
 - › This is separating the display and business logic (view and controller in MVC)

```
1 // omitted package and import statements for space
2 @WebServlet("/FormServlet")
3 public class FormServlet extends HttpServlet {
4     private static final long serialVersionUID = 1L;
5     protected void service(HttpServletRequest request, HttpServletResponse response)
6         throws ServletException, IOException {
7         String username = request.getParameter("username");
8         String next = "/invalidUsername.jsp";
9         if (username != null && username.equals("csci201")) {
10             next = "/validUsername.jsp";
11         }
12         RequestDispatcher dispatch = getServletContext().getRequestDispatcher(next);
13         dispatch.forward(request, response);
14     }
15 }
```


More Servlets



- For more information on Servlets
 - › Go to <http://docs.oracle.com/javaee/6/tutorial/doc/bnafd.html>
 - › Go through one of the many servlet tutorials online





Outline

- Java Servlets
- Program

Program



- Create the following form and process the submitted data with a servlet to display the page on the right with a servlet.

School Form

Email: jeffrey.miller@usc.edu
Password:
Birthday: 01/01/2017
New Student?: Yes No
College: USC
Major: Computer Science
Favorite Color:
Terms and Conditions: I agree.

Submitted Data

Field	Value
Email	jeffrey.miller@usc.edu
Password	csci201
Birthday	2017-01-01
New Student	yes
College	USC
Major	CS
Color	#ffff00
Terms and Conditions	yes