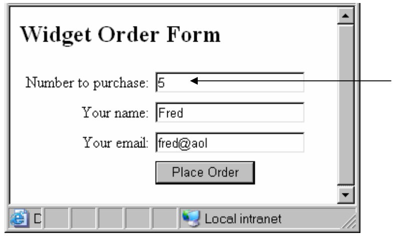
Server-side Web Development and Programming

Lecture 4:
Java Server Pages for Complex Forms

Parsing Numeric Input

- `getParameter` method returns a String



Widget Order Form

Number to purchase: 5

Your name: Fred

Your email: fred@aol

This is “5”, not the number 5!

- Must parse strings into numbers before performing numeric computations

Parsing Numeric Input

Useful built-in methods:

- Parsing a whole number string (such as “57”):
  `int Integer.parseInt(String)`
- Parsing a decimal string (such as “5.7”)
  `double Double.parseDouble(String)`

- Example:

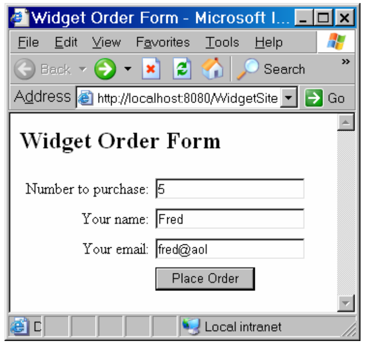
```
String quantity = request.getParameter(“quantity”);
int quantityNumber = Integer.parseInt(quantity);
```

Numeric Input Example

```
<%
   String name = request.getParameter("customerName");
   String email = request.getParameter("customerEmail");
   String quantity = request.getParameter("quantity");
   double pricePerUnit = 9.95;
   int quantityNumber = Integer.parseInt(quantity);
   double totalCost = pricePerUnit * quantityNumber;
 %>

 <h2>Order Confirmation</h2>
 <p>Thank you for your order of <%= quantity %> widgets, <%= name %>.</p>
 <p>At $<%= pricePerUnit %>, your bill will be $<%= totalCost %>.</p>
 <p>You will shortly recieve an email confirmation at <%= email %>.</p>
```

Numeric Input Example


Widget Order Form - Microsoft I...
File Edit View Favorites Tools Help
Back Search
Address http://localhost:8080/WidgetSite Go
Local intranet

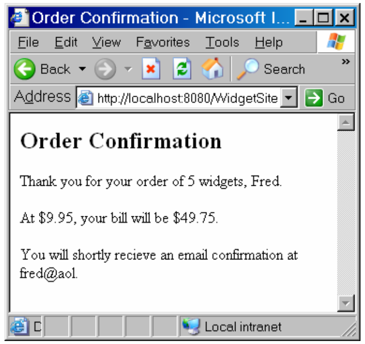
Widget Order Form

Number to purchase: 5

Your name: Fred

Your email: fred@aol

Place Order


Order Confirmation - Microsoft I...
File Edit View Favorites Tools Help
Back Search
Address http://localhost:8080/WidgetSite Go
Local intranet

Order Confirmation

Thank you for your order of 5 widgets, Fred.

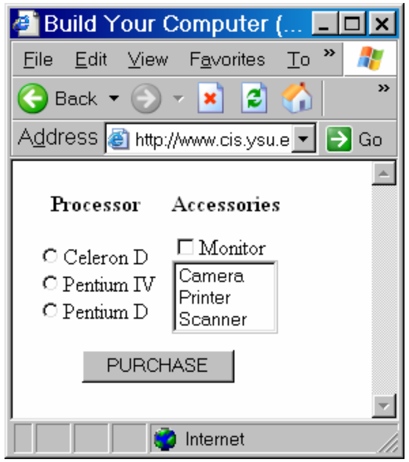
At $9.95, your bill will be $49.75.

You will shortly recieve an email confirmation at fred@aol.

Complex Input Elements

- Checkboxes
- Radio Buttons
- Lists

- Require more complex handling
  – Will do truly complex handling in multipage servlet structures



Processor Accessories

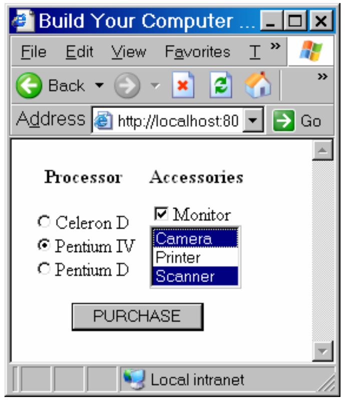
○ Celeron D
○ Pentium IV
○ Pentium D

□ Monitor

Camera
Printer
Scanner

PURCHASE

Example Result



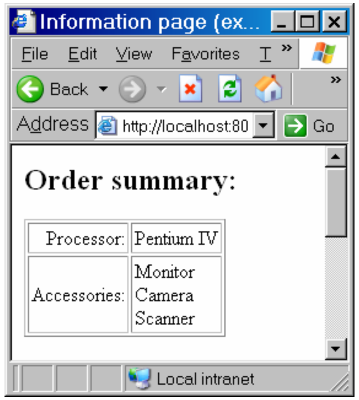
Processor Accessories

○ Celeron D
● Pentium IV
○ Pentium D

☑ Monitor

Camera
Printer
Scanner

PURCHASE



Order summary:

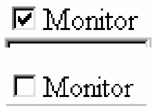
Processor:	Pentium IV
Accessories:	Monitor Camera Scanner

Checkbox HTML

- Basic form:

```
<INPUT TYPE="checkbox“ NAME="monitor">
Monitor
```

- String passed to server:
  – “ ...`monitor=on`... ” if monitor is checked
  – No mention of `monitor` if not checked



☑ Monitor

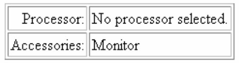
□ Monitor

Checkbox JSP

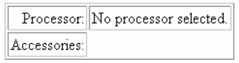
- If execute JSP code:
  ```
  String monitor =
            request.getParameter("monitor");
  ```
  `monitor` will have the value
  - "on" if the checkbox was checked
  - null if the checkbox not checked
    - null is always returned if ask for value of parameter which was not passed in the request string

Conditions in Java

- JSP may need to do different things depending on checkbox
  - Display "Monitor" if checked
  - Display nothing if not checked



Processor:	No processor selected.
Accessories:	Monitor



Processor:	No processor selected.
Accessories:	

- This requires a Java condition
  - Basic syntax like C++/JavaScript
    ```
    if(condition) {
      statements to execute if true
      }
    else {
      statements to execute if false
      }
    ```

Conditional HTML Display

- Key: Display different html based on condition
- Put html in conditional statement
  - Must use <% and %> to differentiate Java, html

```
 <%
   if (condition) {
%>
     html to display if condition true
<%
     }
   else {
%>
     html to display if condition false
<%
  }
%>
```

Checkbox Example

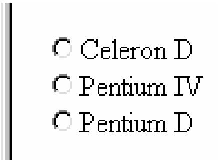
```
<%
if (monitor != null) {
%>
Monitor<br>
<%
  }
%>
```

If this Java condition is true (the monitor is not null)

Display this html

Radio Button HTML

- Convention: Only one in group checked
- Must give all in group same name
- Must give each different VALUE

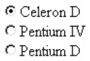


○ Celeron D
○ Pentium IV
○ Pentium D

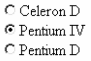
```
<INPUT TYPE="radio"
       NAME="processor" VALUE="Celeron D">
  Celeron D<BR>
<INPUT TYPE="radio"
       NAME="processor" VALUE="Pentium IV">
  Pentium IV<BR>
<INPUT TYPE="radio"
       NAME="processor" VALUE="Pentium D">
  Pentium D
```

Radio Button JSP

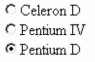
- Sent in form …name= value… to server
  - processor=Celeron+D
  - processor=Pentium+IV
  - processor=Pentium+D



● Celeron D
○ Pentium IV
○ Pentium D



○ Celeron D
● Pentium IV
○ Pentium D



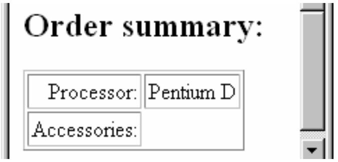
○ Celeron D
○ Pentium IV
● Pentium D

- Can access value using:

```
String processor =
          request.getParameter("processor");
```

And display in html:

```
<%= processor %>
```

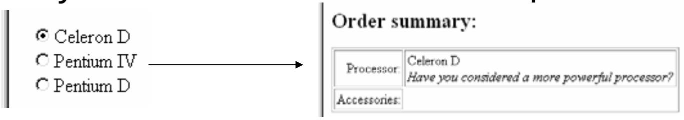


Order summary:

Processor:	Pentium D
Accessories:	

String Comparison

- May need to base html on value passed:



● Celeron D
○ Pentium IV
○ Pentium D

Order summary:

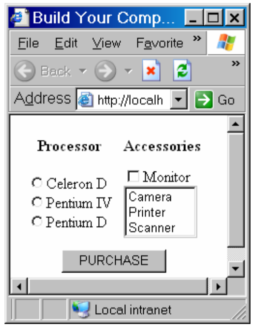
Processor	Celeron D *Have you considered a more powerful processor?*
Accessories:	

- Must use **.equals** method to compare strings
  Basic form:
  `if (string1.equals(string2) { …`
- Example:

```
<% if (processor.equals(“Celeron IV”) { %>
<br/><i>Have you considered a more powerful
processor?</i>
<% } %>
```

Null Input

- User may not choose any radio button!
  - processor will have value null
  - Executing .equals on null will give run time exception
    - User should never see this!



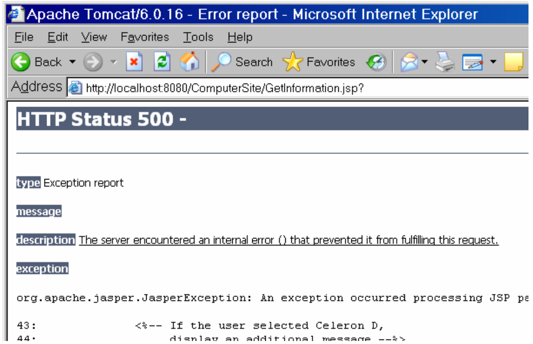
Processor Accessories

○ Celeron D
○ Pentium IV
○ Pentium D

□ Monitor

Camera
Printer
Scanner

PURCHASE



Apache Tomcat/6.0.16 - Error report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/ComputerSite/GetInformation.jsp?

HTTP Status 500 -

type Exception report

message

description The server encountered an internal error () that prevented it from fulfilling this request.

exception

```
org.apache.jasper.JasperException: An exception occurred processing JSP pa

43:              <%-- If the user selected Celeron D,
44:                   display an additional message --%>
```

Detecting Null Input

Basic form of code:

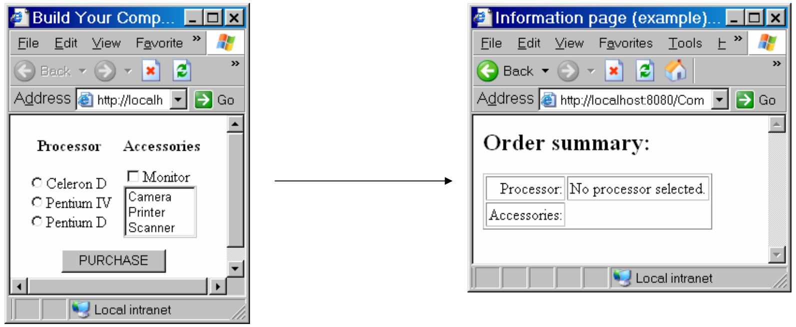
```
if (variable != null) {
   if (variable.equals(value1)) {…}
   if (variable.equals(value2)) {…}
   }
else {
   code for case where no button selected
   }
```

Detecting Null Input

Example:

```
<% if (processor != null) { %>
          <%= processor %>
          <% if (processor.equals("Celeron D")) { %>
              <br/><i>Have you considered a more
               powerful processor?</i>
          <% } %>
      <%   }
        else {
      %>
          No processor selected.
      <%
          }
      %>
```

Detecting Null Input


Build Your Comp...
File Edit View Favorite
Back
Address http://localh
Go
Information page (example)...
File Edit View Favorites Tools H
Back
Address http://localhost:8080/Com
Go
Local intranet

Processor Accessories

○ Celeron D
○ Pentium IV
○ Pentium D

□ Monitor
Camera
Printer
Scanner

PURCHASE

Order summary:

Processor:	No processor selected.
Accessories:	

- Note: At this level of complexity, may handle with separate redirection servlet

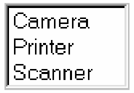
List HTML

- Basic form:

```
<SELECT NAME=“listname” SIZE=“numvisible”>
   <OPTION VALUE=“value1”/> label1
   <OPTION VALUE=“value2”/> label2
   …
</SELECT>
```

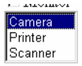
- Example:

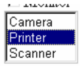
```
<SELECT NAME="peripherals" SIZE="3">
      <OPTION VALUE="Camera“/>Camera
      <OPTION VALUE="Printer“/>Printer
      <OPTION VALUE="Scanner“/>Scanner
    </SELECT>
```


Camera
Printer
Scanner

List JSP

- Sent in form …`name= value`… to server
  - `peripherals=Camera`
  - `peripherals=Printer`
  - `peripherals=Scanner`





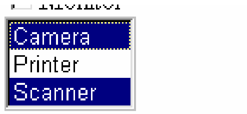
- Can access value using:

```
String processor =
        request.getParameter("peripherals");
```

Multiple Selection Lists

- Can allow user to select multiple options from list with MULTIPLE attribute

```
<SELECT NAME="peripherals" SIZE="3“
                          MULTIPLE>
```



- Sends `name=value` string for each option selected

`…peripherals=camera&peripherals=scanner…`

- `getParameter` method will not work!

JSP for Multiple Selection Lists

- **`String[] request.getParameterValues(String name)`**

  Returns array of values passed for this name

- **Example:**

```
String [] peripherals =
            request.getParameterValues("peripherals");
```

creates the array:

`peripherals`

0	“Camera”
1	“Scanner”

Looping Through Arrays

- Often use loop to process all values selected
- Basic form in Java:

```
for (int i = 0; i < arrayname.length; i++) {
  code to process ith element of arrayname
  }
```

Note: Java has built-in length property for array which evaluates to its size

- Can use to display html multiple times

```
<% for (int i = 0; i < arrayname.length; index++) { %>
  html created from ith element of arrayname
<% } %>
```

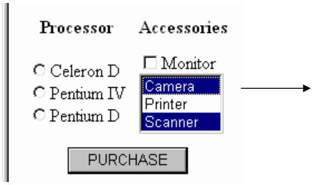
Looping Through Arrays in JSP

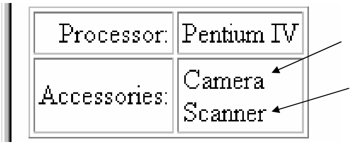
- Example:

```
<% for (int i = 0; i < peripherals.length; i++) { %>
       <%= peripherals[i] %><br\>
<% } %>
```

For each value in the `peripherals` array

Display that value in html





Processor:	Pentium IV
Accessories:	Camera Scanner

`peripherals[0]`

`peripherals[1]`

Checking for NULL Lists

- User may not choose any value in a list
- **`request.parameterValues`** will return null instead of an array
- Must check for this before processing array

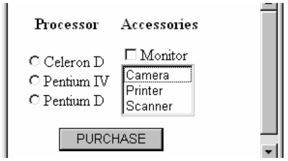
```
if (arrayname != null) {
     for (int i = 0; i < arrayname.length; i++) {…}
     }
else {
     code to handle case where no option selected
     }
```

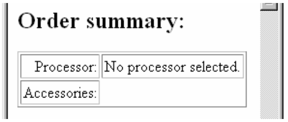
Checking for NULL Lists

- Example:

```
<% if (peripherals != null) { %>
   <% for (int i = 0; i < peripherals.length; i++) { %>
       <%= peripherals[i] %><br>
   <% } %>
<% } %>
```

Only executed if peripherals exist





Order summary:

Processor:	No processor selected.
Accessories:	