## COMP 4021 Internet Computing

#### HTML Forms

**David Rossiter** 

#### Browser-Server Interaction

- So far we have only gotten pages from the server
  - We never send any information to the server --- well, except an URL to get the page we want (Oh! We do send cookies underneath a page access)
- Now, we learn how to send information to the server via a form, which would perform actions based on the information received
- In general, we learn how browser and server interact (both ways) to accomplish a task!

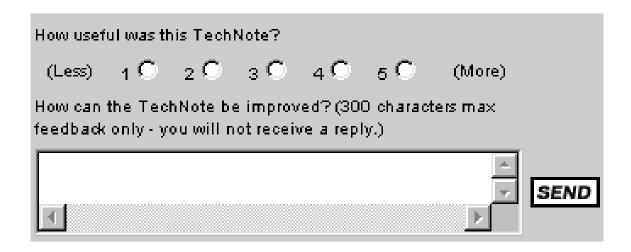
#### What is a Form?

- Forms are the easiest way to send data from a browser to a server program
- The user can enter various types of information, then send them to the server

- Browsers receive data from server most of the time
- Besides form, have you learnt any other situations under which data (of any kind) are sent to the server?

### Common Form Usage

Typical example:



After the Submit button is pressed, the data is (usually) sent to a particular program running on a particular server, somewhere on the internet

#### Constructing a Form

method is the method used to transfer the data to the server

- Either get or post
- Differences discussed later

```
<form [method="get"
action="show_environment.php"</pre>
```

HTML tags for form content go here

- action is the location of the server side script
- By default, it is in the samedirectory as the html page
- Or use absolute path: https://...
- As you know, you should use lower case letters for all HTML, which includes forms

#### Reset & Submit Buttons

- □ Press <input type="reset" value="here"/> to clear the form.
- □ Press <input type="submit" value="here"/> to submit your data.

Press here to clear the form. Press here to submit your data.

#### Text Input and Label Fields

■ Name: <input name="name" size="32"

value="(type your name here)" maxlength="255"/>

Name: (type your name here)

- Name has no semantic relation with the input
- Use <label> and nest <input> inside it:
- □ <label>Name: <input name="name" size="32" value="(type your name here)" maxlength="255"/></label>
- Or use id to associate label with input field
- <label for="name1">Name:</label>
   <input id="name1" name="name" value=... />

#### Text Area Fields

□ Any comments you would like to make: <br/>
<textarea name="comments" rows="3" cols="32"> </textarea>

Any comments you would like to make:



■ You can also use **<label>** to tag the description of the <textarea> and for other types of input such as radio buttons to be described later

#### Radio Button Input

checked />Sales<br/>

```
□ Student: <input type="radio" name="degree"
  value="masters"/>
  Masters <input type="radio" name="degree"
  value="undergraduate" checked/>Undergraduate
           Student: O Masters O Undergraduate
□ Indicate your area(s) of specialty <br/>
  <input type="checkbox" name="speciality" value="1"/>
  Management<br/>
                                  Indicate your area(s) of speciality
  <input type="checkbox"</pre>
                                   Management
  name="speciality" value="2"
```

☑ Sales

#### Select Input

```
Age range: < select name="age_range">
 <option>10-20</option>
<option selected>20-30
<option>30-40
                                      20-30
                             Age range:
 <option>40-50</option>
                                      10-20
<option>50-60</option>
                                      20-30
<option>60-70</option>
                                      30-40
                                      40-50
</select>
                                      50-60
                                      60-70
```

#### Complete Form Code 1/5

```
<html>
<body>
<h1>HTML Form Example</h1>
<form method="get"</pre>
action="show_environment.php">
<label for="name1"> Name: </label>
<input id="name1" name="name" size="32" value="(type your name here)"
   maxlength="255" />
```

### Complete Form Code 2/5

```
 Student:
<input id="masters" type="radio" name="degree" value="masters" />
<label for="masters">Masters</label>
<input id="ug" type="radio" name="degree" value="undergraduate" checked />
<label for="ug">Undergraduate</label>
<label for "age"> Age range:</label>
<select id="age" name="age range">
  <option>10-20
  <option selected>20-30
  <option>30-40
   <option>40-50
  <option>50-60
   <option>60-70</option> </select>
```

#### Complete Form Code 3/5

```
Indicate your area(s) of specialty<br/>
<i><small>Click on one or more areas</small></i>
<br/><br/>
<input type="checkbox" name="specialty" value="1"/>Management<br/>
<input type="checkbox" name="specialty" value="2"/>Sales<br/>
<input type="checkbox" name="specialty" value="3"/>Human Resources<br/>
<input type="checkbox" name="specialty" value="4"/>IT/ database<br/>
<input type="checkbox" name="specialty" value="5"/>IT/ networking<br/>>
<input type="checkbox" name="specialty" value="6"/>IT/ internet<br/>
<input type="checkbox" name="specialty" value="7"/>IT/ general<br/>
```

### Complete Form Code 4/5

#### Complete Form Code 5/5

```
Press <input type="submit" value="here"/> to submit your data.
<br/>
<br/>
Press <input type="reset" value="here"/> to clear the form.
<br/>
<br/>
</form>
</body>
</html>
```

# Complete Form Display

- After making the selections shown on the right the form is then submitted by clicking on the submit button
- The response is shown in the following two slides

#### **HTML Form Example** Name: Donald Student: Masters Undergraduate Age range: 20-30 🕶 Indicate your area(s) of specialty Click on one or more areas Management ✓ Sales ✓ Human Resources ✓ IT/ database IT/ networking IT/ internet ☐ IT/ general Any comments you would like to make: Great Send Press here to submit your data.

to clear the form

### Data Seen By Server Side Program 1/2

This PHP program sees the following environment variables....

- Environment var PATH is /usr/local/bin:/usr/bin:/bin
- Environment var DOCUMENT ROOT is /usr/local/apache-1.3.28/htdocs
- Environment var HTTP\_ACCEPT is text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,\*/\*;q=0.5
- Environment var HTTP\_ACCEPT\_CHARSET is ISO-8859-1,utf-8;q=0.7,\*;q=0.7
- Environment var HTTP ACCEPT ENCODING is gzip, deflate
- Environment var HTTP\_ACCEPT\_LANGUAGE is en-us
- Environment var HTTP\_CONNECTION is keep-alive
- Environment var HTTP HOST is ihome.ust.hk
- Environment var HTTP\_KEEP\_ALIVE is 300
- Environment var HTTP\_USER\_AGENT is Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.3) Gecko/20070309 Firefox/2.0.0.3

### Data Seen By Server Side Program 2/2

- Environment var REMOTE ADDR is 218.102.38.10
- Environment var REMOTE PORT is 4111
- Environment var SCRIPT\_FILENAME is /home/rossiter/public\_html/cgi-bin/show\_environment.php
- Environment var SERVER ADDR is 143.89.14.183
- Environment var SERVER ADMIN is cchelp@ust.hk
- Environment var SERVER NAME is ihome.ust.hk
- Environment var SERVER PORT is 80
- Environment var SERVER\_SOFTWARE is Apache/1.3.28 (Unix) mod\_jk/1.2.15 FrontPage/5.0.2.2634 mod\_ssl/2.8.15 OpenSSL/0.9.7c
- Environment var TZ is GMT-8
- Environment var GATEWAY\_INTERFACE is CGI/1.1
- Environment var SERVER PROTOCOL is HTTP/1.1
- Environment var REQUEST\_METHOD is GET
- Environment var QUERY\_STRING is name=Donald&degree=undergraduate&age\_range=20-30 &speciality=2&speciality=3&speciality=4&comments=Great&version\_date=15\_feb\_2007
- Environment var REQUEST\_URI is /~rossiter/cgi-bin/show\_environment.php?name=Donald&degree=undergraduate&age\_range =20-30&speciality=2&speciality=3&speciality=4&comments=Great&version\_date=15\_feb\_2007
- Environment var SCRIPT\_NAME is /~rossiter/cgi-bin/show\_environment.php
- Environment var PHP\_SELF is /~rossiter/cgi-bin/show\_environment.php

#### The Fields Which Contain Form Data

- In this example, the GET method is being used to send data to the server
- You can see from the last two slides that the QUERY\_STRING field contains the form data
- Because the GET method is being used, the REQUEST\_URI field also has the form data
- The server side program can extract the data from either of those strings

### Images in Forms

If you just want to put an image in a form for visual effect and nothing more, just use the usual

```
<img src="image.gif" /> instruction
```

- □ If you want to use an image to act as a submit button, use input type="image"
- For example:

```
<input type="image" name="image"
src="send_button.gif"/>
```



#### Images in Forms

Possible syntax:

```
<input type="image"
name="image-id" src="image-url" [align=alignment] />
```

- There is another advantage to using images
- The exact (x, y) position that the user clicks on is automatically sent to the server side program along with all the other form data
- □ The server side program may choose to use it or ignore it

### Images in Forms

What is your favourite Hong Kong shopping area?

Mong Kok 旺角

You Ma Tei 油麻地

**Kowloon City** 

**Hung Hom** 

#### Example clickable map:

```
What is your favourite
Hong Kong shopping area?
<br/>>
```

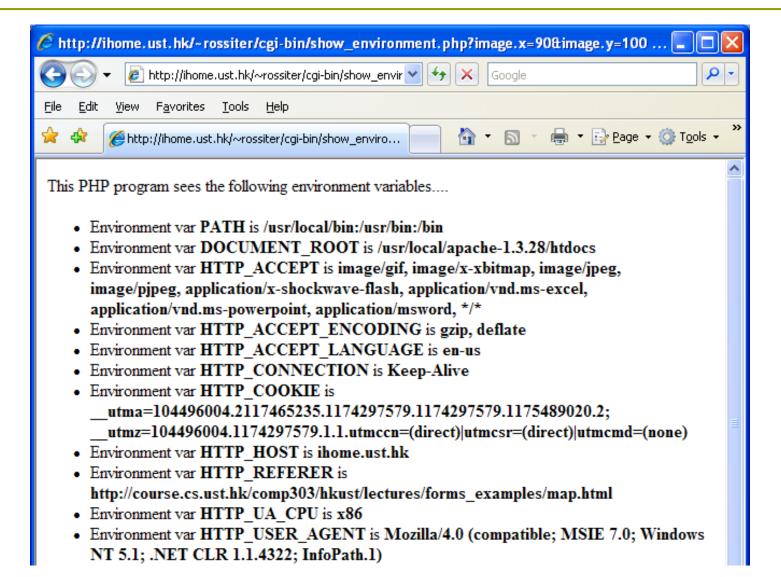
```
Sheuna
                       North
                        Point
     Central
                              Tai Koo
             Chai Causeway Bay
    Admiralty
            Happy Valley
    show environment.php">
```

Tsim Sha Tsui 尖沙咀 Tsim Sha Tsui East

```
<form method="get"</pre>
action="http://ihome.ust.hk/~rossiter/cgi-bin/
<input type="image" name="image" src="harbour.gif"/>
</form>
```

- After clicking on the image data is sent to the server program
- This particular server side program simply lists all the data it knows about – see next two slides

### Data Seen By Server Side Program 1/2



### Data Seen By Server Side Program 2/2

- Environment var REMOTE ADDR is 218.102.38.10
- Environment var REMOTE\_PORT is 3949
- Environment var SCRIPT\_FILENAME is /home/rossiter/public\_html/cgibin/show environment.php
- Environment var SERVER\_ADDR is 143.89.14.183
- Environment var SERVER\_ADMIN is cchelp@ust.hk
- Environment var SERVER\_NAME is ihome.ust.hk
- Environment var SERVER PORT is 80
- Environment var SERVER\_SOFTWARE is Apache/1.3.28 (Unix) mod\_jk/1.2.15
   FrontPage/5.0.2.2634 mod\_ssl/2.8.15 OpenSSL/0.9.7c
- Environment var TZ is GMT-8
- Environment var GATEWAY\_INTERFACE is CGI/1.1
- Environment var SERVER PROTOCOL is HTTP/1.1
- Environment var REQUEST\_METHOD is GET
- Environment var QUERY\_STRING is image.x=90&image.y=100
- Environment var REQUEST\_URI is /~rossiter/cgi-bin/show\_environment.php? image.x=90&image.y=100
- Environment var SCRIPT\_NAME is /~rossiter/cgi-bin/show\_environment.php
- Environment var PHP SELF is /~rossiter/cgi-bin/show environment.php

The server-side program can extract the form data (i.e., X and Y values) from either the QUERY\_STRING or REQUEST\_URI fields

#### Get vs. Post

■ A reminder - there's two ways to send information from client to server; 'get' or 'post'

```
<form method="get"
action="http://ihome.ust.hk/~rossiter/cgi-bin/show_environment.php">
....
</form>
```

#### Example Get

■ When the form data is sent, all the parameter names and values are shown on the browser location line



#### The Get Method

#### Good points

- All the parameter names and values are shown helpful for debugging a system you are developing (https will encrypt them for transmission)
- Easy for a program to build up the single text sequence and send it
- A bookmark URL contains all the query parameters

#### Bad points

- Maximum length of a URL is about 2000 characters, sometimes is not enough
- Users can see all the parameter names and values on the browser location line, even the 'hidden' ones
- Users can easily change the values and send them to your system,
   could cause trouble for you

#### The Post Method

- When the form data is sent using the post method, all the parameter names and values, etc, are added after the HTTP header
- Good points
  - User cannot see any parameter names/ values
  - There is no limit on how many bytes you can send from client to server i.e. can send many large MP3 files, etc, using the Post method
- Bad points
  - Cannot bookmark the page to go to the same page (or cause the same actions) on the server
    - Bookmark the link does not save the parameters

Form + JavaScript

#### Useful JavaScript Events

- You already know lots of events for handling the mouse/keyboard, etc - here are some events which can be used for handling forms:
- onsubmit fires when the user submits the form data (fires just before the data is actually sent)
- onreset fires when the user resets all the fields in a form (fires just before the data is actually reset)
- onchange fires just after the contents of something is changed
- onselect fires when user selects text in a text field

### JavaScript Processing

- Commonly, forms are 'dead'
  - User just types/ selects, then submits
- Can do client-side processing of form data
  - Could use JavaScript to do some kind of action, such as assess checking data when an event occurs
  - Don't actually have to send data to server, could use a form just for text input to JavaScript, etc

### Example JavaScript Processing

- Example JavaScript checking when user triggers *onsubmit*:
  - Checking student ID field has correct number of characters
  - Checking telephone number field only has digits
- Example JavaScript change of image when user triggers onchange:
  - Nothing yet entered, dull image:
  - As soon as text entered, bright image:



#### Hidden Fields

#### Hidden Fields

- Hidden fields are useful for implementing 'system' variables shared between the server and the client
- A hidden field is also useful for the client to send information to the server (without using the standard input fields such as text fields and selection boxes)

#### Hidden Fields

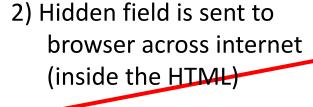
- □ Use *input type="hidden"* for hidden fields; e.g.
  - <input type="hidden" name="creation\_time" value="15\_2\_2010\_14:32"/>
- □ The field does not appear on screen; hence, users would not see it or know about it unless using "view source"
- The user can't change the value of the field
- This method can be used for storing 'state' information (i.e., which both browser and server code can read and change)
  - This can be very useful, particularly if the user/ browser does not permit cookies on their machine
  - However, cookies and sessions are more commonly used, to be discussed later

### Example Hidden Field Usage

1) HTML page is generated by server <input type="hidden" name="username" value="rossiter" /> . . .

3) User enters form data as usual and presses

Submit





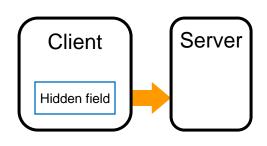
4) Data is sent to server across internet, including the hidden field

- 5) Server program looks at hidden field, responds appropriately:
- . . . Thanks rossiter, here's some other web information for you

#### Example – Color Selector

- An HTML form does not provide a simple input field which handles color selection
- We can build a color selector using several DIVs and then send the color using a hidden field





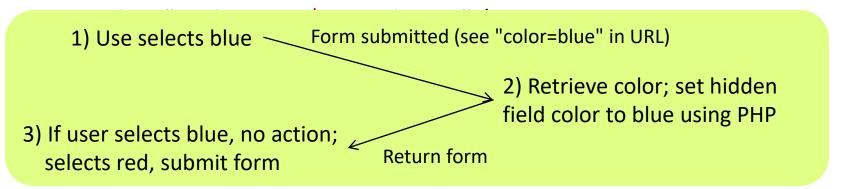
Run Demo

### Example 1: Form + JavaScript

(2) Function sets hidden field function select(color) { value and submits the form document.getElementById("color").value=color; document.getElementById("color").form.submit(); } (3) onclick passes the selected <style> color to the "select" function (1) Format DIVs as div { a color selector position: absolute; <h2>Please select a color:</h2> width: 50px; <div style="position:relative"> height: 50px; } <div style="background-color:red;left:0px"</pre> </style> onclick="select('red')"></div> <div style="background-color:yellow;left:50px"</pre> onclick="select('yellow')"></div> (4) "select" sets the "value" <!-- other four colors are not shown --> </div> attribute to the selected color <form action="..." method="GET"> and "submit " 4 <input type="hidden" name="color" id="color" value="" /> </form>

#### Setting Hidden Field in the Server

- □ In Example 1, the selected color is always sent to the server even when the same color is clicked several times
- It is better to send the selected color only when it is different from the color previously selected color
- □ In Example 2, the server sends back the user's last selected color to the browser by setting a hidden field's value with PHP: <input type="hidden" name="color" id="color"</p>



# Example 2: Adding PHP code in Server Program

```
<?php
  $my_color = $_GET['color']; // get "color" submitted from browser
?>
<html> <head>
                    New color
                                    Retrieve hidden form field
    <script>
    function select(newColor) {
                                     Previous color (set by php via hidden field)
      var color = document,getElementById("color");
      if (newColor != color.value) {
                                                                Form is submitted only
         if (!confirm("Do you want to change " +
                                                                if the value is changed
                       color.value + " to " + newColor + "?"))
                                                                to a new value
           return;
         color.value=newColor;
         color.form.submit();
                                                                        Continued
    </script> </head>
                                                                        in next slide
```

### Example 2 Cont.

```
<body> <h2>Please select a color:</h2>
 <div style="position:relative">
   <div style="background-color:red;left:0px"</pre>
                                                                 The color hidden
      onclick="select('red')"></div>
   <div style="background-color:yellow;left:50px"</pre>
                                                                 field with a value set
      onclick="select('yellow')"></div>
                                                                 by the server
   <div style="background-color:green;left:100px"</pre>
      onclick="select('green')"></div>
                                                                 $my color is set in
   <div style="background-color:cyan;left:150px"</pre>
                                                                 PHP code (top of
      onclick="select('cyan')"></div>
                                                                 previous slide) to
   <div style="background-color:blue;left:200px"</pre>
                                                                 return user's
      onclick="select('blue')"></div>
   <div style="background-color:magenta;left:250px"</pre>
                                                                 previously selected
      onclick="select('magenta')"></div> </div>
                                                                 color
  <form action="..." method="GET">
     <input type="hidden" name="color" id="color"
         value="<?php print $my_color; ?>" /> </form></body></html>
```

### Take Home Message

- Form is a very natural and popular way to present information to users and solicit input from them
- Hidden fields can be used to embed data in the form without users seeing the data (e.g., encode user name in a hidden field)
- Hidden fields in forms are very useful in passing states of the clients to the server
  - Those states can be passed to the server when the form is submitted
  - No additional action is required other than submitting the form (which the user will do anyway); user indeed does not know about this