

XHTML Forms and JavaScript Processing

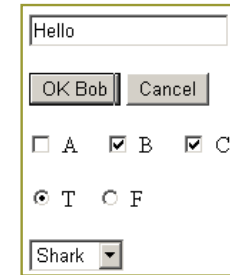
- ❖ Forms provide a standard data entry method for users to send information to a web server
 - ◆ Clicking button calls a CGI script on server
 - ◆ CGI = **Common Gateway Interface**
 - ◆ CGI scripts are usually provided by your ISP
 - ◆ Can be written in PERL, Server side JavaScript, Python, ASP, Java, C, etc.
- ❖ Forms can be sent using email (Usually Disabled)
- ❖ Forms can be a Graphical User Interface (GUI) which call JavaScript functions
 - ◆ Clicking a button or menu item calls a JavaScript function

Copyright © 2005 R.M. Laurie 1

The Form's Shell

- ❖ The form layout may contain invisible elements and visible elements which are used for user information entry modes:

- ◆ Text box
- ◆ Button: Submit, Reset
- ◆ Check boxes
- ◆ Radio buttons
- ◆ Menus



Copyright © 2005 R.M. Laurie 2

Form Element

- ❖ `<form>` element is a block level element
 - ◆ All elements contained within `<form>` are part of the form
- ❖ `<form name="frmBob" method="post" action="/cgi-bin/scriptname.cgi">`
 - ◆ `name` attribute is the identifier (or label) of the form
 - ◆ `method` specifies how data is sent to server (CGI)
 - ◆ `post` = Appends data to browser HTTP request
 - ◆ `action` specifies the CGI script on web server to process the sent data
- ❖ `</form>`
 - ◆ All form elements must be closed

Copyright © 2005 R.M. Laurie 3

Text Box and Text Area Elements

- ❖ **Text box element** is for single line text input
`<input type="text" name="txtIntro" value="Hello" size="20" />`
 - ◆ `type="text"` defines as a text box
 - ◆ `name` is the optional identifier for the text box
 - ◆ `value` is the value initially displayed in the field
 - ◆ `size` is the width of text box in characters
 - ◆ `maxlength` limits characters entered to a maximum
- ❖ **Text area element** is for multi-line text input
`<textarea rows="4" cols="30" name="tarGreet">Hello</textarea>`
 - ◆ `rows` is the height
 - ◆ `cols` is the width
 - ◆ `name` is the optional identifier for the text box

Copyright © 2005 R.M. Laurie 4

Submit, Reset and Image Buttons

- ❖ **Submit button** triggers the form action
 - `<input type="submit" name="btOK" value="OK Bob" />`
 - ◆ type defines as a submit button
 - ◆ value is displayed in the caption of the button
- ❖ **reset button** reinitializes form fields
 - `<input type="reset" value="Cancel" />`
- ❖ **input button** usually used to call function
 - `<input type="button" name="btCalc" value="Calculate" onclick="calculate()" />`
- ❖ **input image** triggers action specified in name
 - `<input type="image" name="btBack" src="btback.gif" />`

Copyright © 2005 R.M. Laurie 5

Form, Text Box, and Text Area Code

```

<head>
  <title>Text Demo</title>
</head>
<body style="background-color: #CCFFCC">
  <form name="frmBob" method="post" action="/cgi-bin/script.cgi">
    <p> <input type="text" name="txtbxBob" value="Hello"
      size="20" maxlength="30" />
    </p>
    <p> <textarea rows="3" cols="25" name="txtarBob">
      Greetings</textarea> </p>
    <p> <input type="submit" name="btnOK" value="OK Bob" />
      <input type="reset" name="btnCancel" value="Cancel" /> </p>
  </form>
</body>

```

Copyright © 2005 R.M. Laurie 6

Check Box and Radio Button Code

```

<head>
  <title>Check Box and Radio Buttons</title>
</head>
<body style="background-color: #CCCCFF">
  <form name="frmAnswer" method="post" action="/cgi-bin/script.cgi">
    <p> <input type="checkbox" name="chk1A" value="AnsA" /> A
    <input type="checkbox" name="chk1B" value="AnsB"
      checked="checked" /> B <input type="checkbox" name="chk1C"
      value="AnsC" checked="checked" /> C</p>
    <p><input type="radio" name="radQues2" value="AnsT"
      checked="checked" />T <input type="radio" name="radQues2"
      value="AnsF" />F</p>
    <p> <input type="submit" name="btSubmit" value="SEND" />
    <input type="reset" name="btReset" value="Clear" /> </p>
  </form>
  <code>chk1B=AnsB&chk1C=AnsC&radQues2=AnsT&btnSubmit=SEND</code>
</body>

```

chk1B=AnsB&chk1C=AnsC&radQues2=AnsT&btnSubmit=SEND

Select Menu

- ❖ Select menus use **select** and **option** elements
- ❖ Select menus work well with setting parameters
- ❖ Can be used to provide a Graphical User Interface (GUI) for JavaScript Programs
- ❖ This example utilizes a select menu to choose one of three functions:
 - ◆ Square
 - ◆ Square Root
 - ◆ Factorial
- ❖ Calculate button click calls **Calculate()** function
 - ◆ **onclick** is an event (Stay Tuned)

Copyright © 2005 R.M. Laurie 8

❖ Calculate button click calls **Calculate()** function

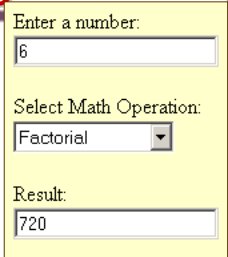
```
<head> <title>Forms Example</title>
<script type="text/javascript">
<!-- Calculate() function will go here --> </script> </head>
<body style="background-color: #FFFFCC">
<form name="frmCalc" action="">
<p>Enter a number:<br />
<input type="text" name="txtEntry" size="20" /></p>
<p>Select Math Operation:<br />
<select name="mnuMathOp">
<option selected="selected">- Choose One -</option>
<option>Square</option>
<option>Square Root</option>
<option>Factorial</option>
</select> </p>
<p>Result:<br />
<input type="text" name="txtResult" size="20" /></p>
<p> <input type="button" name="btCalc" value="Calculate"
onclick="Calculate()" />
<input type="reset" name="btClear" value="Clear" /></p>
</form> </body>
```

Calculate() function must be placed in head and can be called anywhere in body or head portion of document

```
function Calculate()
{
    var Entry, Result="", I, Selection;
    Entry = parseFloat(document.frmCalc.txtEntry.value);
    Selection = document.frmCalc.mnuMathOp.selectedIndex;
    if(Selection == 1) Result = Entry * Entry;
    else if(Selection == 2)
        Result = Math.sqrt(Entry);
    else if(Selection == 3)
    {
        Result = 1;
        for(I = 1; I <= Entry; I++)
            Result = Result * I;
    }
    else window.alert("Select an Operation!");
    document.frmCalc.txtResult.value = Result;
}
```

Select Menu *onchange* Event

```
<head><title>Select Example</title>
<script type="text/javascript" src="Calculate.js" >
</script> </head>
<body style="background-color: #FFFFCC">
<form name="frmCalc" action="">
<p>Enter a number:<br />
<input type="text" name="txtEntry" size="20" />
</p>
<p>Select Math Operation:<br />
<select name="mnuMathOp" onchange="Calculate()" >
<option selected="selected">- Choose One -</option>
<option>Square</option>
<option>Square Root</option>
<option>Factorial</option>
</select> </p>
<p>Result:<br />
<input type="text" name="txtResult" size="20" /></p>
</form> </body>
```



Linking to External JavaScript Files

❖ JavaScript functions can be stored in a separate files and linked to from multiple XHTML documents

❖ Advantages:

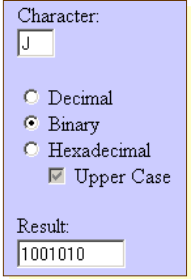
- ◆ Code Reuse and HTML Coding Simplification
- ◆ Caching of a shared script improves performance
- ◆ Can be shared across multiple servers using URL
- ◆ Can link to common functions

❖ Simply use a SRC attribute in <SCRIPT> tag

```
<head>
<title>Forms Example</title>
<script type="text/javascript" src=" ../mathop.js"></script>
</head>
```

GUI Using Radio Buttons and Check Box

- ❖ Radio buttons and check boxes can enhance a GUI Form
- ❖ In this example you can type in a single ASCII character and convert it to the specified number system
- ❖ Note that the checkbox is enabled only when hexadecimal is selected
- ❖ The display changes when the character is changed, any radio button is clicked, or the check box is clicked (when enabled)



Character:
J

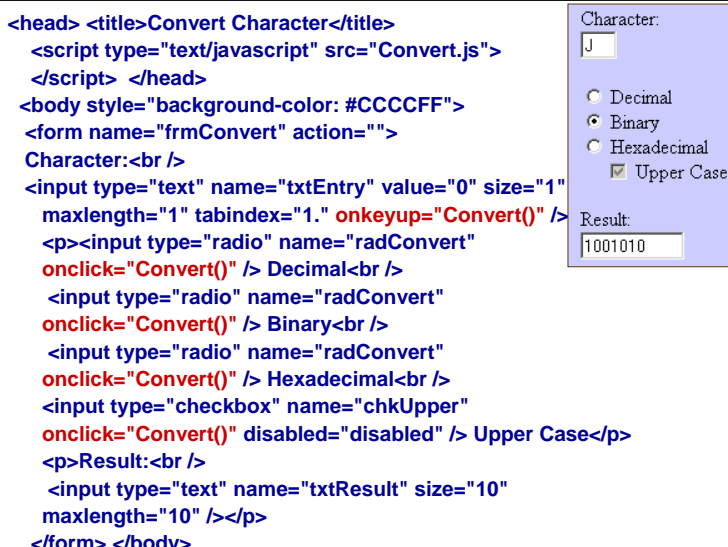
Decimal
 Binary
 Hexadecimal
 Upper Case

Result:
1001010

Copyright © 2005 R.M. Laurie 13

```

<head> <title>Convert Character</title>
<script type="text/javascript" src="Convert.js">
</script> </head>
<body style="background-color: #CCCCCCFF">
<form name="frmConvert" action="">
Character:<br />
<input type="text" name="txtEntry" value="0" size="1"
maxlength="1" tabindex="1." onkeyup="Convert()" />
<p><input type="radio" name="radConvert"
onclick="Convert()" /> Decimal<br />
<input type="radio" name="radConvert"
onclick="Convert()" /> Binary<br />
<input type="radio" name="radConvert"
onclick="Convert()" /> Hexadecimal<br />
<input type="checkbox" name="chkUpper"
onclick="Convert()" disabled="disabled" /> Upper Case</p>
<p>Result:<br />
<input type="text" name="txtResult" size="10"
maxlength="10" /></p>
</form> </body>
    
```



Character:
J

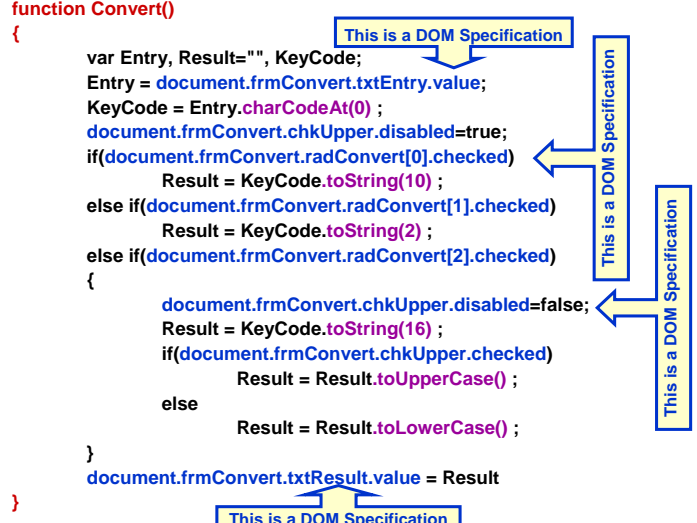
Decimal
 Binary
 Hexadecimal
 Upper Case

Result:
1001010

Copyright © 2005 R.M. Laurie 16

```

function Convert()
{
    var Entry, Result="", KeyCode;
    Entry = document.frmConvert.txtEntry.value;
    KeyCode = Entry.charCodeAt(0);
    document.frmConvert.chkUpper.disabled=true;
    if(document.frmConvert.radConvert[0].checked)
        Result = KeyCode.toString(10);
    else if(document.frmConvert.radConvert[1].checked)
        Result = KeyCode.toString(2);
    else if(document.frmConvert.radConvert[2].checked)
    {
        document.frmConvert.chkUpper.disabled=false;
        Result = KeyCode.toString(16);
        if(document.frmConvert.chkUpper.checked)
            Result = Result.toUpperCase();
        else
            Result = Result.toLowerCase();
    }
    document.frmConvert.txtResult.value = Result
}
    
```



This is a DOM Specification

This is a DOM Specification

This is a DOM Specification

This is a DOM Specification

This is a DOM Specification

Copyright © 2005 R.M. Laurie 16

Event Driven Program Exercise

- ❖ Create program that converts temperatures between Fahrenheit and Celsius Systems
 - ◆ Create a text box for one temperature
 - ◆ Create two command buttons one for each conversion
 - ◆ Create a text box to display the result

Copyright © 2005 R.M. Laurie 16