

Web Application Development

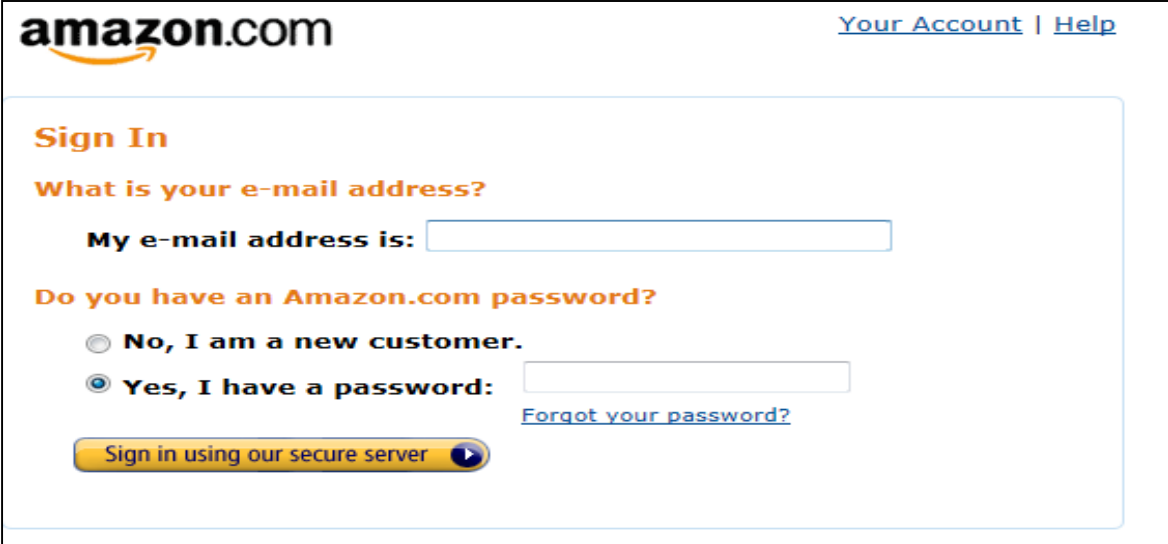
1

LECTURER - MARIE BRENNAN

INTRODUCTION TO HTML FORMS

FORMS

- Suppose we want to allow people viewing our web site to enter information (e.g. their names, their email addresses etc.).
- *Forms* make it possible to collect and process user input, and formulate personal replies (among other things).
- Lots of applications e.g.
 - ☐ Online Shopping
 - ☐ Web based surveys
 - ☐ etc.



The screenshot shows the Amazon.com sign-in interface. At the top left is the Amazon logo and 'amazon.com'. At the top right are links for 'Your Account' and 'Help'. The main section is titled 'Sign In' in orange. Below it is the question 'What is your e-mail address?' in orange. This is followed by the text 'My e-mail address is:' and a text input field. Below that is the question 'Do you have an Amazon.com password?' in orange. There are two radio button options: 'No, I am a new customer.' and 'Yes, I have a password:'. The 'Yes' option is selected. To the right of the 'Yes' option is a password input field. Below the password field is a link 'Forgot your password?'. At the bottom is a yellow button with the text 'Sign in using our secure server' and a play icon.

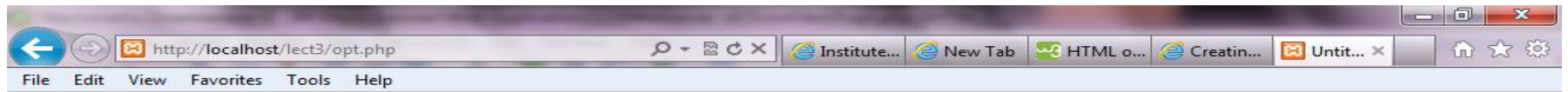
FORMS

- Two aspects of using forms
 - (a) collecting input
 - (b) processing input
- (a) is done with standard HTML but
- (b) is a bit trickier and is usually accomplished by writing programs which run on the server, ie PHP

HTML Forms

- Textfield
- Hidden field
- Password field
- Fieldset
- Label
- Submit / Reset Button
- Radio Button
- Checkboxes

- Dropdown menu
- Scrolling menu
- Listbox selection
- Optgroup
- Textarea
- File Uploading
- Tab Order



- Volvo
- Swedish Cars**
- Volvo
- Saab
- German Cars**
- Mercedes
- Audi



EXAMPLE - INPUT FORM



The screenshot shows a Netscape browser window with the address bar displaying `http://localhost:8080/register.html`. The page title is "USER REGISTRATION". Below the title, a note indicates that fields marked with an asterisk (*) are required. The form contains several input fields: "First Name*", "Last Name*", "E-Mail*", "Zip Code*", "User Name*", "Password*", and "Confirm Password*". There are also checkboxes for music preferences (Rock, Pop, Bluegrass, Blues, Jazz, Country) and radio buttons for email notifications (Yes, No). At the bottom, there are "Submit" and "Reset" buttons. The browser's status bar at the bottom shows "Document Done".

USER REGISTRATION
* Required Fields

First Name*

Last Name*

E-Mail*

Zip Code*

User Name*

Password*

Confirm Password*

What music are you interested in?
☐ Rock ☐ Pop ☐ Bluegrass
☐ Blues ☐ Jazz ☐ Country

Would you like to receive e-mail notifications on our special sales?
☒ Yes ☐ No

Document Done

FORMS

- A form is an area that can contain form elements.
- Form elements allow the user to enter information (e.g. **text fields, textarea fields, drop-down menus, radio buttons, checkboxes**, etc.)
- A form should have at least one control like a **Submit** button that activates the **Action** attribute of the form.
- Action attributes specifies the URL or JSP that will be called when the user clicks the Submit button
- A form is defined with the **<form> tag**.
- **<form action = method = “post”>**
- **<input type =>**
- **<input type =>**
- **.....**
- **</form>**

Form Template

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Form Template</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8"/>
</head>
<body>
<form action = "formscript.php" method = "post">
Put your textboxes, drop down menus, radio buttons etc. in here.
</form>
</body>
</html>
```


INPUT TAG

- `<form action = method = "post">`
- `<input type =>`
- `<input type =>`

- Most commonly used form tag is the `<input>` tag.
- Describes the input type
- Attributes of `<input>` tag
 - *type* - kind of input field it will be.
 - text, hidden, password, and submit or reset.
 - *name* - Assigns a name to the given field so that you can reference it later.
 - *size* - Sets the horizontal width of the field. The unit of measurement is in blank spaces.
 - *maxlength* - Max number of characters that can be entered.
 - *value* - specifies value of data e.g. " ", or any default value e.g. xyz

INPUT TAG - TEXT FIELDS

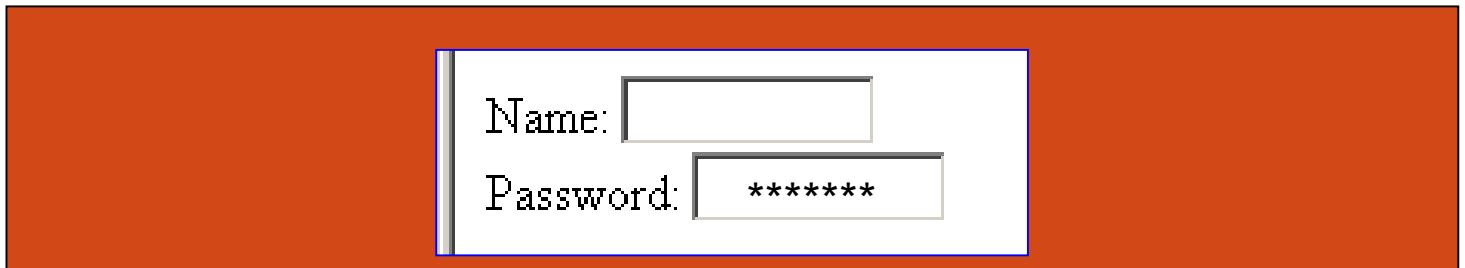
Text fields are used when you want the user to type letters, numbers, etc. in a form.

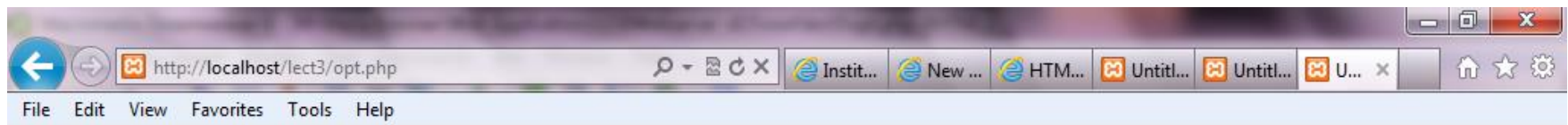
```
<form method="post" action="mailto:youremail@email.com">
```

```
Name: <input type="text" size="10" maxlength="40" name="name"  
      value = " "> <br >
```

```
Password: <input type="password" size="10" maxlength = "10" name = "password"> <br>
```

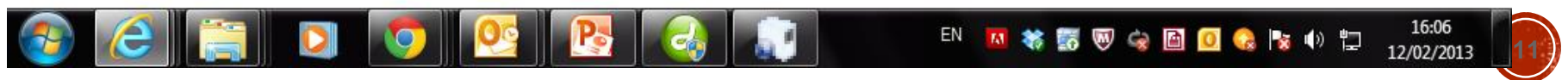
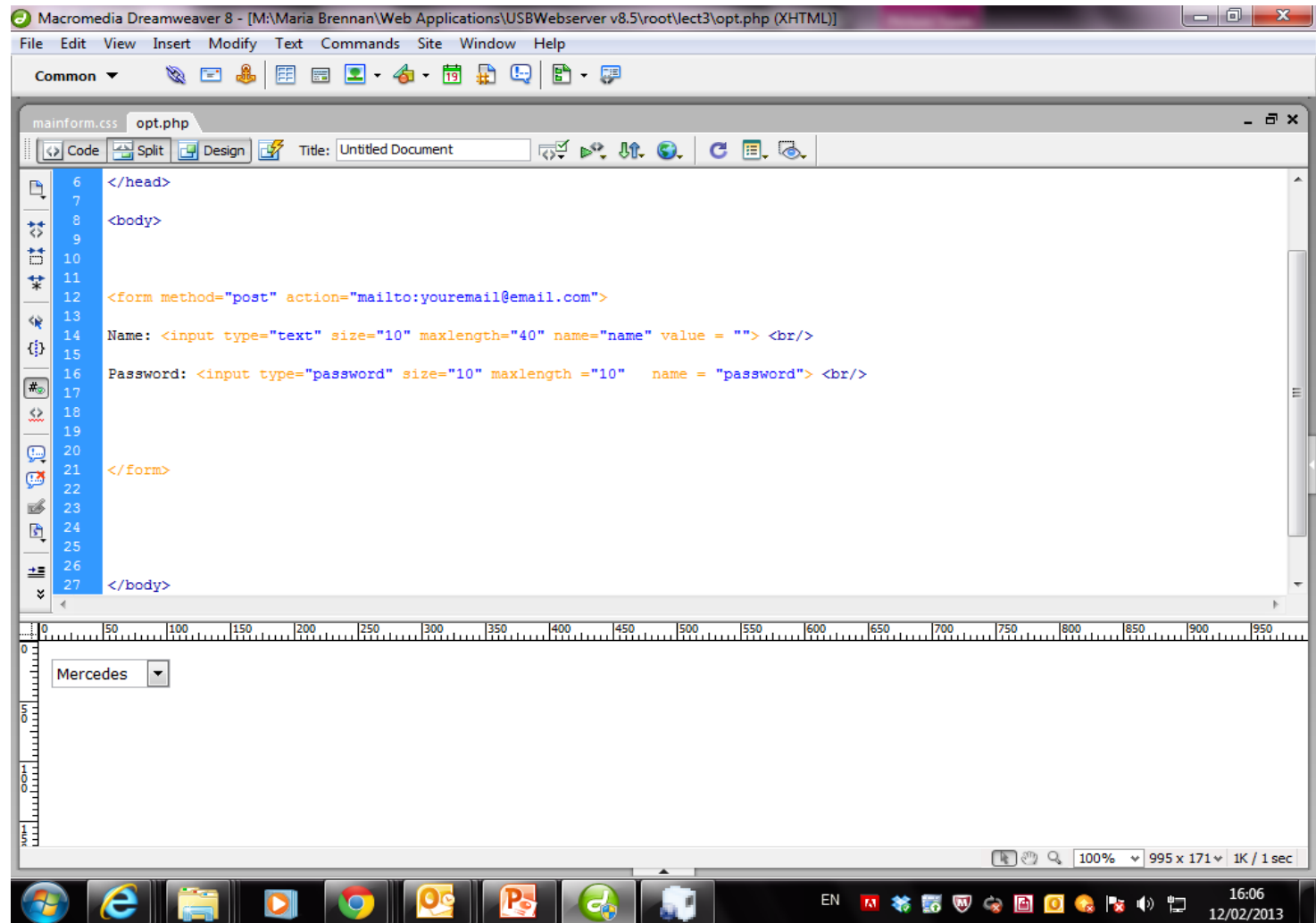
```
</form>
```





Name:

Password:



INPUT TAG - TEXT FIELDS

- Every form should have at least a **SUBMIT** button
- May also have a **RESET** button or user named button also.
- Submit button should be the last item of your form and have its
- *name* attribute set to "**Send**" or "**Submit**".

ATTRIBUTES OF BUTTONS

- **Type** – identifies the type of button
- **Submit** button – activates the Action attribute of the form when clicked
- **Reset** attribute resets all controls on the form to the default values when it's clicked
- Attribute of type **button** creates a Javascript button which can activate a Javascript `OnClick()` method

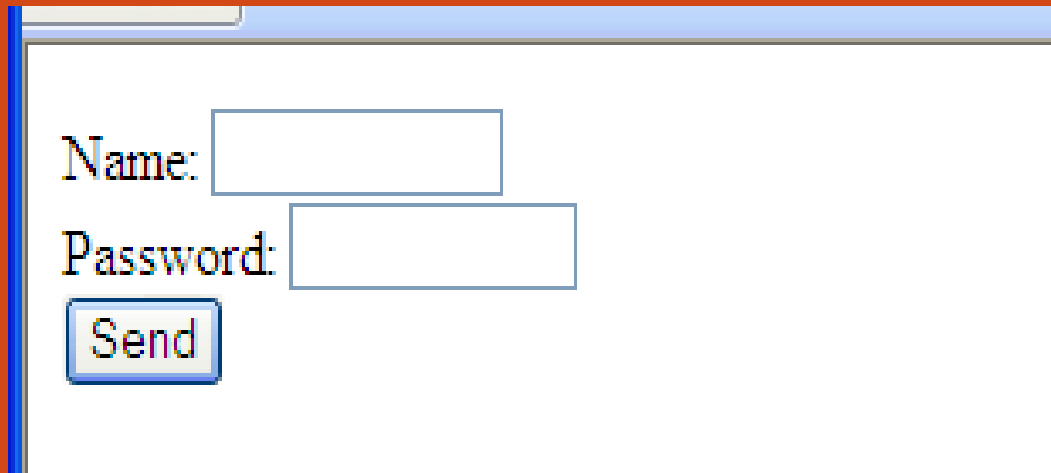
INPUT TAG - BUTTONS

- Examples of buttons
- `<input type = “submit” value = “Submit”>`
- `<input type = “reset” value = “Reset”>`
- `<input type = “button” value = “Enter data”>`

INPUT TAG - SUBMIT BUTTON

- Now add the **submit** functionality to your form.
- Must also **add a destination for information on the form** and specify how you want it to travel to that place.
- Add the following attributes to your <form> does this.
 - **action** - Specifies the URL to send the data to ...a fake email address in this example
 - **method** – use GET or POST methods
- **POST** sends the data without displaying any of the information to the visitor.
- **GET** sends the data and displays the information

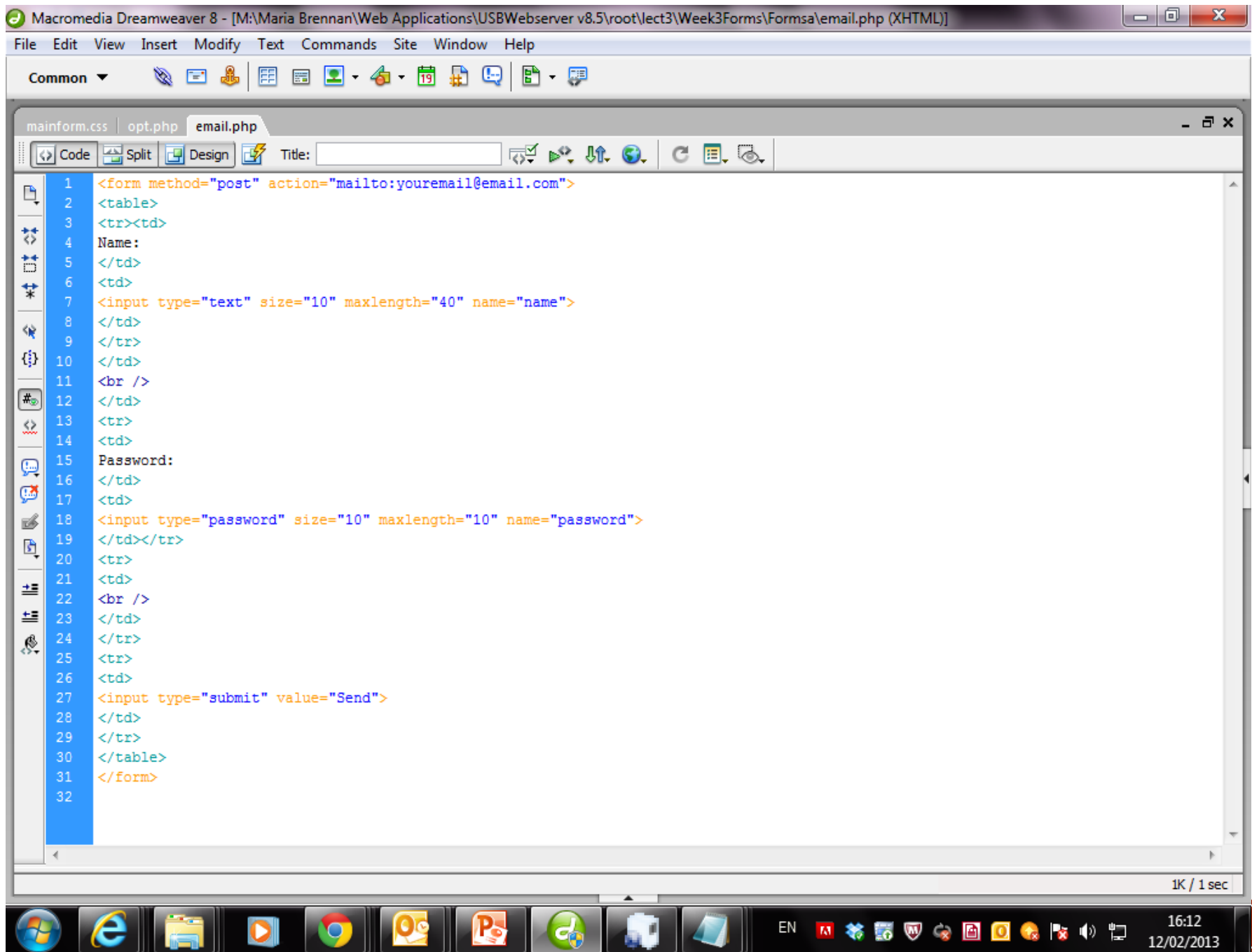
HTML FORM EMAIL CODE



Name:

Password:

Simply change the email address to your own and you will have set up your first functional form!



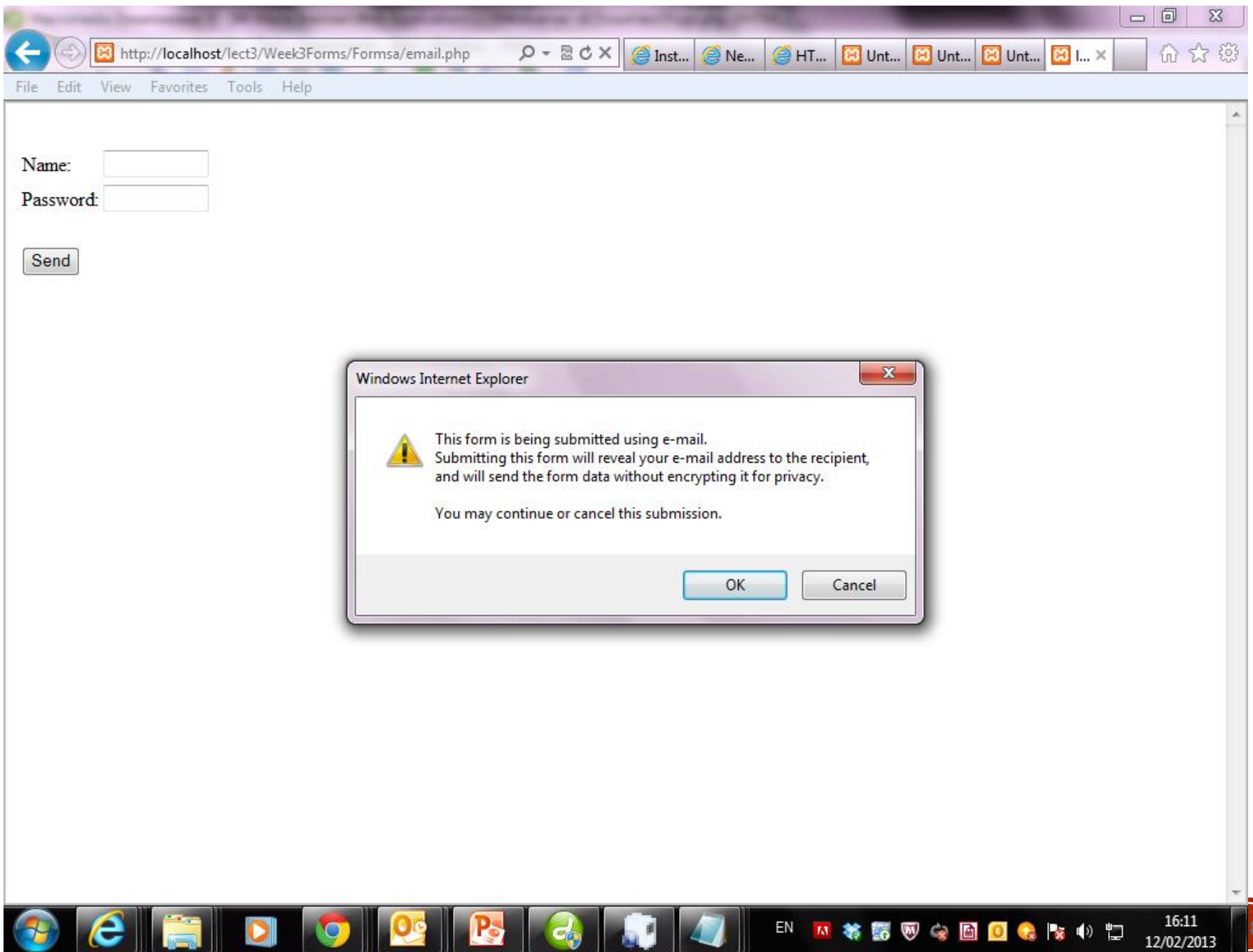
```
1 <form method="post" action="mailto:youremail@email.com">
2 <table>
3 <tr><td>
4 Name:
5 </td>
6 <td>
7 <input type="text" size="10" maxlength="40" name="name">
8 </td>
9 </tr>
10 </td>
11 <br />
12 </td>
13 <tr>
14 <td>
15 Password:
16 </td>
17 <td>
18 <input type="password" size="10" maxlength="10" name="password">
19 </td></tr>
20 <tr>
21 <td>
22 <br />
23 </td>
24 </tr>
25 <tr>
26 <td>
27 <input type="submit" value="Send">
28 </td>
29 </tr>
30 </table>
31 </form>
32
```

1K / 1 sec

EN

16:12
12/02/2013

17



INPUT TAG - RADIO BUTTONS

- Radio Buttons are used when you want the user to select one of a **limited number of choices**
- Radio buttons are a popular form of interaction ...on **quizzes, questionnaires**, and other web sites that give the user a multiple choice.
- Attributes of the radio button.
 - **type** – “**radio**” for radio button
 - **name** - defines which **set** of radio buttons that it is a part of
 - **value** - specifies what will be sent if the user chooses this radio button.
 - Only one value will be sent for a given group of radio buttons (see *name* for more information).
 - **checked** – selects a control
- Example has 2 groups of radio buttons: **shade** and **size**
- Note that only one option can be chosen.

INPUT TAG - RADIO BUTTONS

What kind of shirt are you wearing?

Shade: ☐ Dark ☐ Light

Size: ☐ Small ☐ Medium ☐ Large

Email Myself

Input Tag - Radio Buttons

```
<form method="post" action="mailto:youremail@email.com">
```

What kind of shirt are you wearing?

```
<br />
```

Shade:

```
<input type="radio" name="shade" value="dark">
```

Dark

```
<input type="radio" name="shade" value="light">
```

Light

Size:

```
<input type="radio" name="size" value="small">
```

Small

```
<input type="radio" name="size" value="medium">
```

Medium

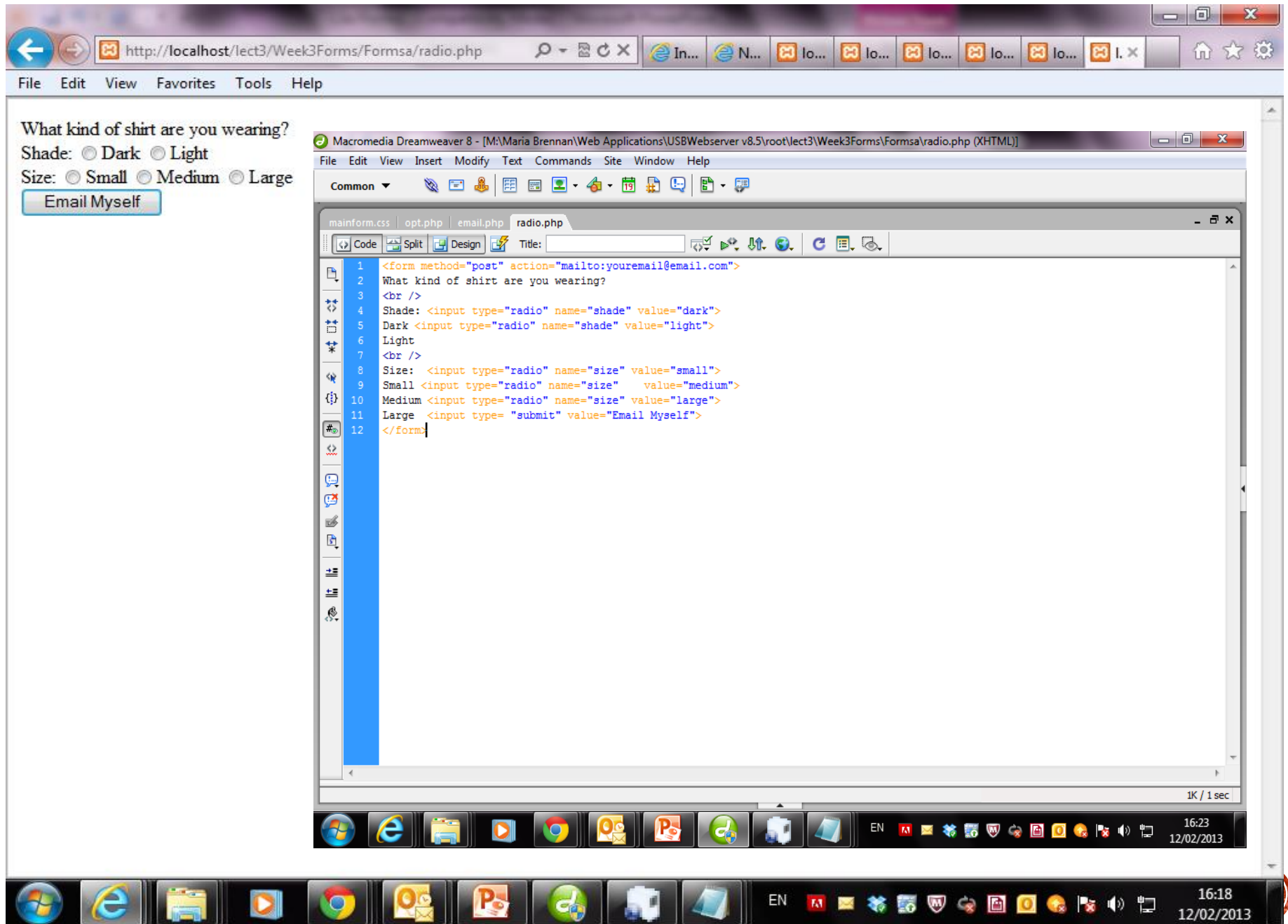
```
<input type="radio" name="size" value="large">
```

Large

```
<br />
```

```
<input type="submit" value="Email Myself">
```

```
</form>
```

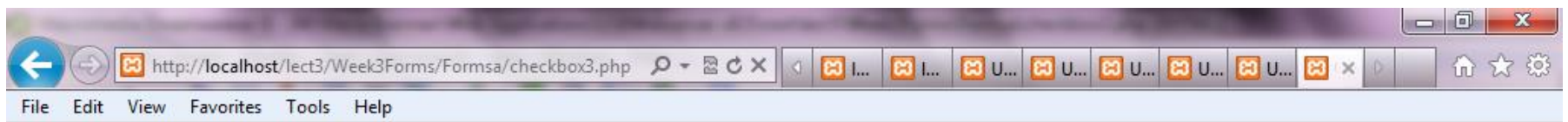


INPUT TAG - CHECKBOXES

- Checkboxes are used when you want the user to select one or more options of a limited number of choices
- The check box's *name* and *value* and *checked* attributes behave the same as a radio button.

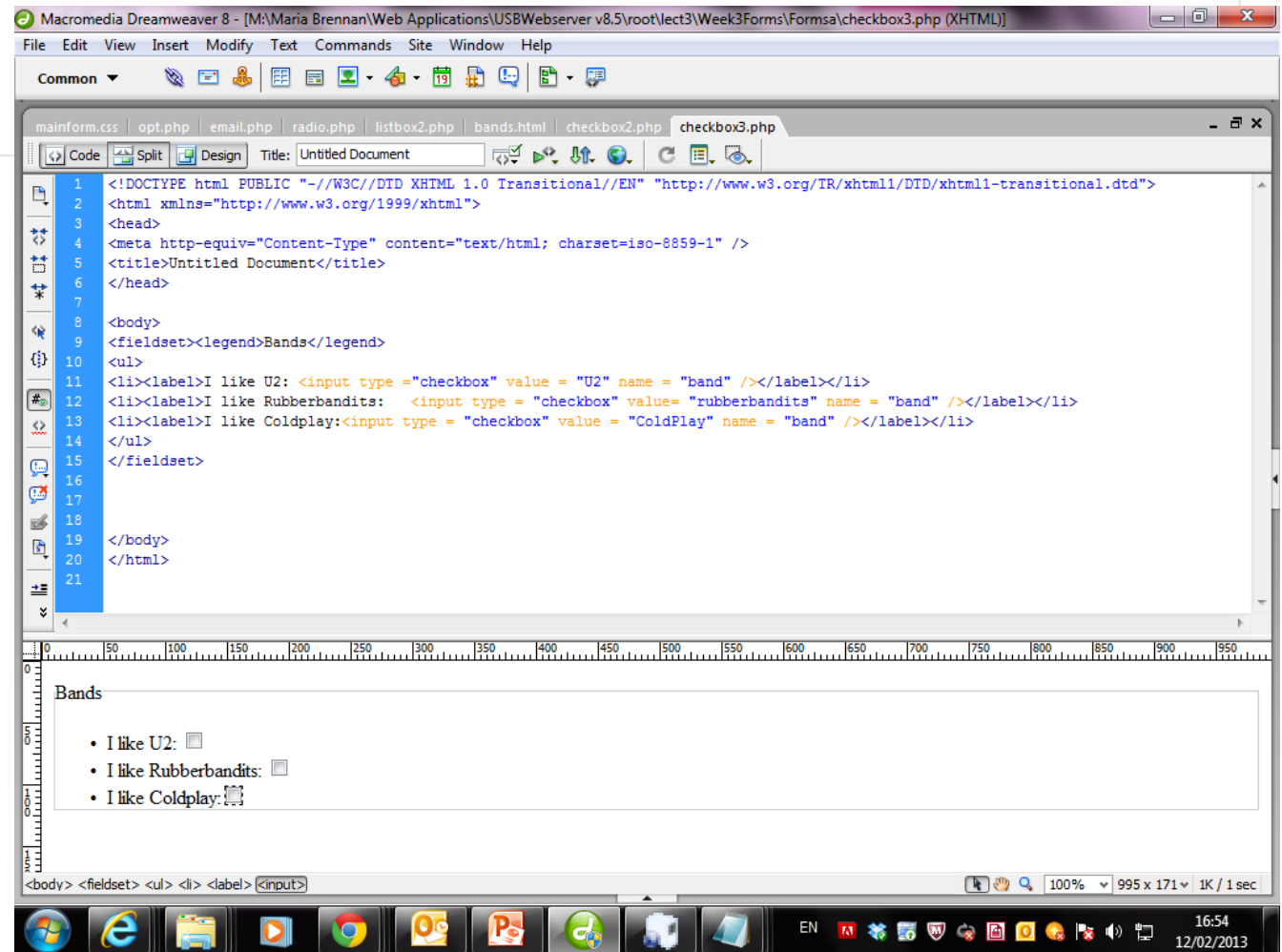
Note: [] with checkboxes

I have a bike: ☐
I have a car: ☐
I have an airplane: ☒



Bands

- I like U2: ☐
- I like Rubberbandits: ☐
- I like Coldplay: ☐



DROPDOWN MENU

- A drop-down box is a selectable list.
- Drop down menus are created with the `<select>` and `<option>` tags.
- `<select>` is the **list itself** and each `<option>` is an available choice for the user.

```
<form action="">
```

```
<select name="cars">
```

```
    <option value="volvo"> Volvo    </option>
```

```
    <option value="saab">  Saab    </option>
```

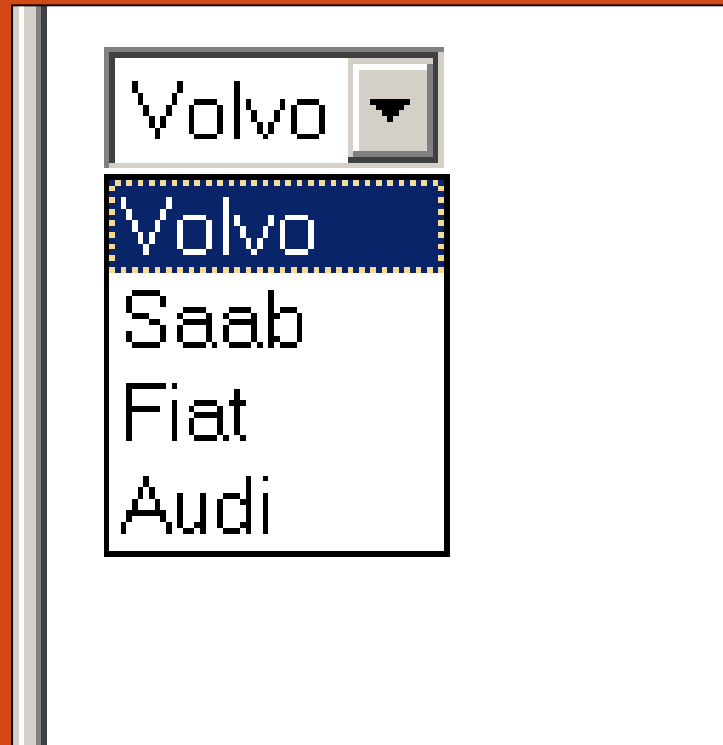
```
    <option value="fiat">   Fiat    </option>
```

```
    <option value="audi">  Audi     </option>
```

```
</select>
```

```
</form>
```

DROPDOWN MENU



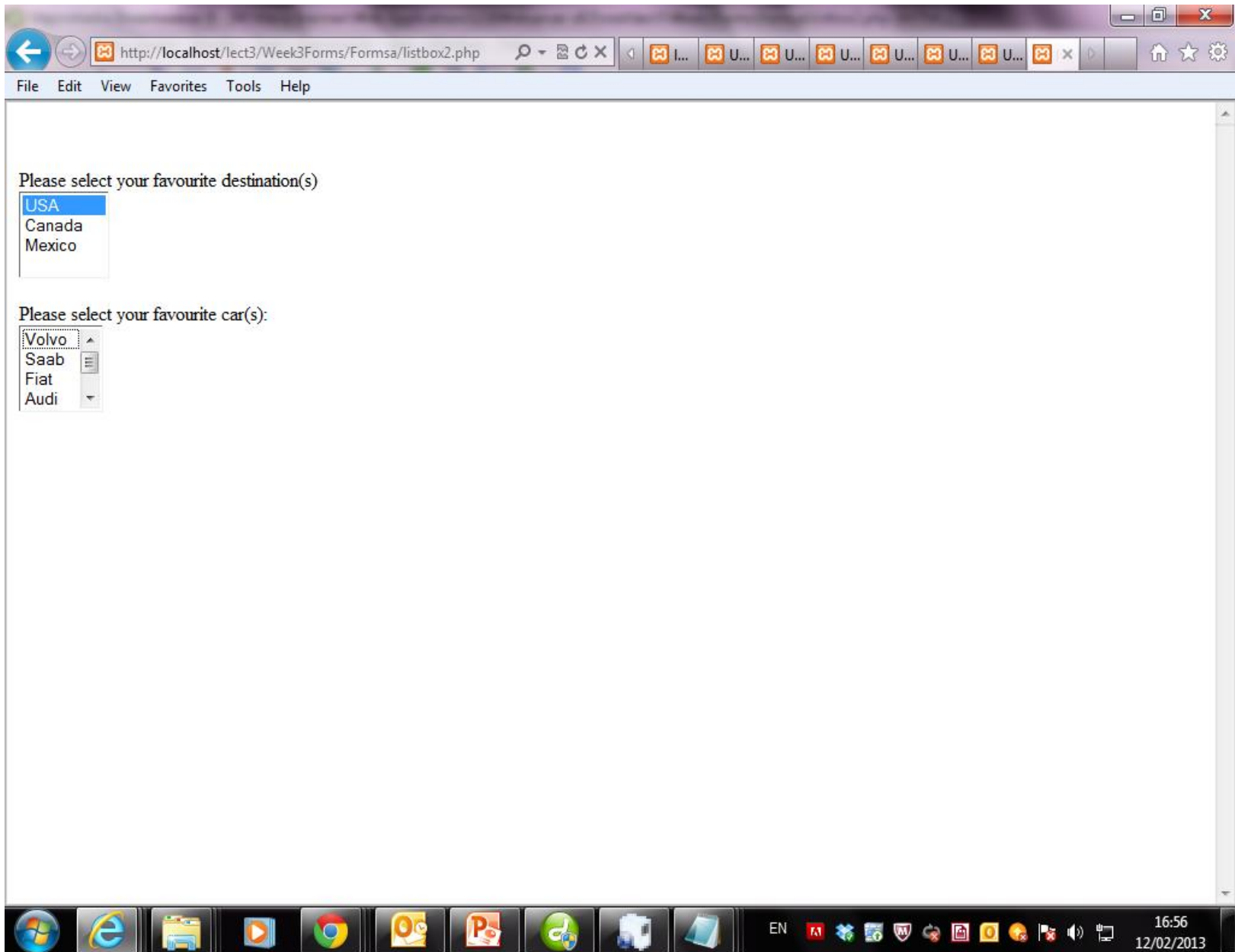
DROPDOWN MENU

- Another type of form, a **highlighted selection list**.
- **size** attribute selects how many options will be shown at once before needing to scroll
- **selected** option tells the browser which choice to select by default.

```
<form action="">
<select name="cars" size = "2">
    <option value="volvo">Volvo</option>
    <option value="saab" selected> Saab</option>
    <option value="fiat"> Fiat</option>
    <option value="audi">Audi</option>

</select>
</form>
```





LIST BOX SELECTION

- Convert a drop down menu to a list box by adding the **multiple** attribute

```
<form action="">
<select name="cars" multiple>
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
</select>
</form>
```

LIST BOX SELECTION

- If the user is allowed to select more than one item for the list then an array should be used:

```
<form action="">  
<select name="cars[]" multiple>  
<option value="volvo">Volvo</option>  
<option value="saab">Saab</option>  
<option value="fiat">Fiat</option>  
<option value="audi">Audi</option>  
</select>  
</form>
```

TEXTAREA

A user can enter text in a text area.

Attributes;

Rows – number of visible lines in text are. A scroll bar appears if the number of lines in the text box exceeds this setting.

Cols – specifies width of text area

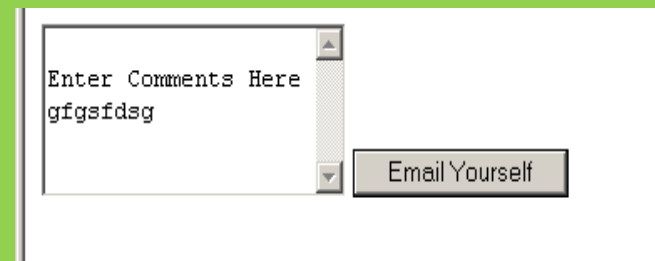
```
<form method="post" action="mailto:youremail@email.com">  
<textarea rows="5" cols="20" wrap="physical" name="comments">
```

Enter Comments Here

```
</textarea>
```

```
<input type="submit" value="Email Yourself">
```

```
</form>
```



• Comments

Enter Comments Here

Email Yourself

Macromedia Dreamweaver 8 - [M:\Maria Brennan\Web Applications\USBWebserver v8.5\root\lect3\Week3Forms\Formsa\textArea.php (XHTML)]

File Edit View Insert Modify Text Commands Site Window Help

Common

mainform.css | opt.php | email.php | radio.php | listBox2.php | bands.html | checkbox2.php | checkbox3.php | textArea.php

Code Split Design Title:

```
1 <form method="post" action="process.php">
2
3 <li>
4 <label for="form-textarea">Comments</label><br>
5 <textarea rows="5" cols="20" wrap="physical" name="comments" id="form-textarea">
6 Enter Comments Here
7 </textarea></li>
8 <br><br>
9 <input type="submit" value="Email Yourself">
10
11 </form>
12
```

WRAP describes how the text in the text area should wrap at the end of lines. Until this attribute came along, browsers generally did not do word wrapping. If you typed a line that was longer than the display area, the line just kept going, hopefully with the display area scrolling along. This was not the way people are used to entering text, so Netscape added the WRAP attribute.

1K / 1 sec

FIELDSET AROUND DATA

This example demonstrates how to draw a border with a caption around your data.

```
<fieldset>
```

```
<legend>
```

Health information:

```
</legend>
```

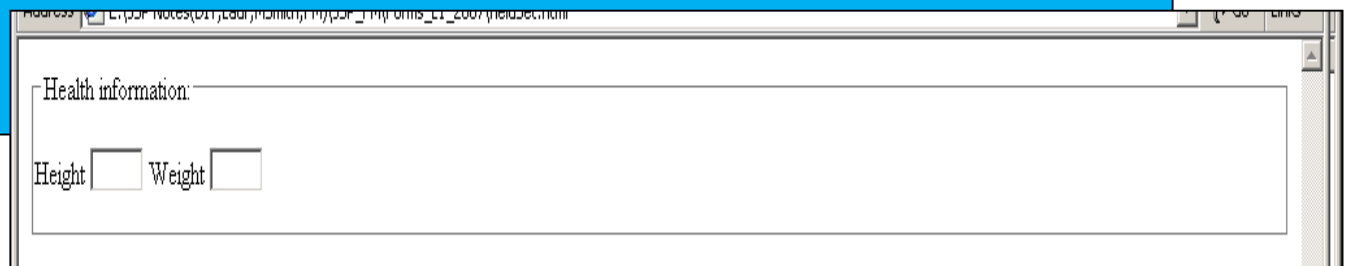
```
<form action="">
```

```
Height <input type="text" size="3">
```

```
Weight <input type="text" size="3">
```

```
</form>
```

```
</fieldset>
```



The screenshot shows a web browser window with the address bar displaying a local file path. The main content area shows the rendered HTML code. A legend box with the title "Health information:" is displayed. Below the legend, there are two text input fields, one labeled "Height" and one labeled "Weight".

QUIZ

- Create a form that contains the following fieldset details:

Customer Information

1. Full name:

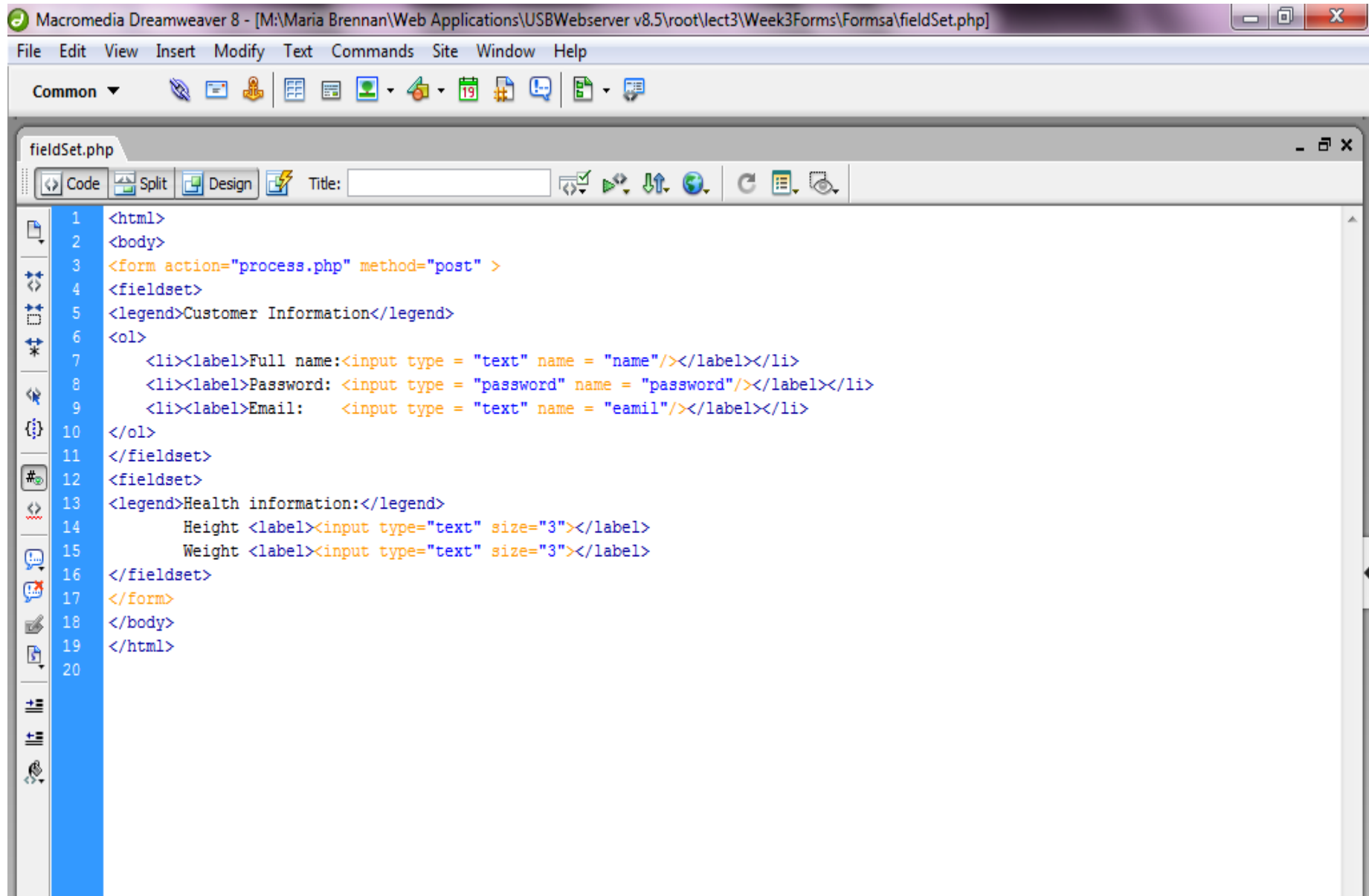
2. Password:

3. Email:

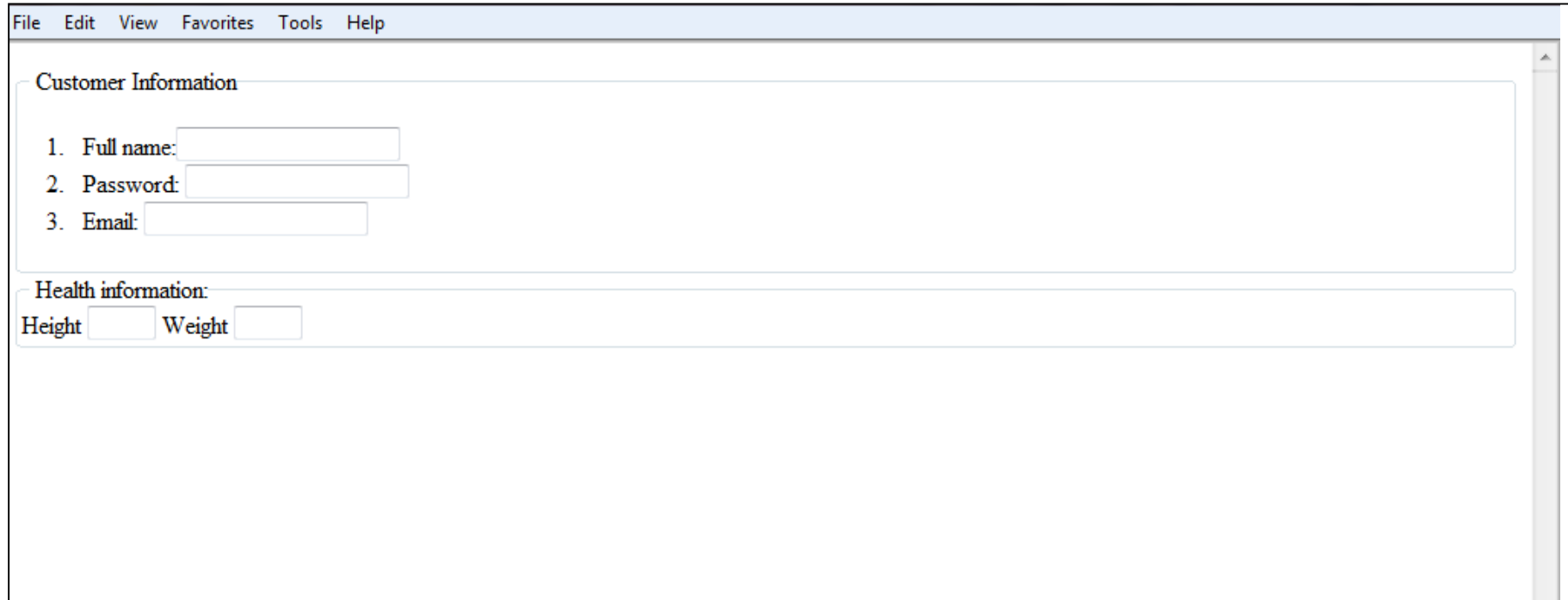
Health information:

Height

Weight



DISPLAY IN BROWSER



A screenshot of a web browser window. The browser's menu bar at the top includes 'File', 'Edit', 'View', 'Favorites', 'Tools', and 'Help'. The main content area displays a form with two sections. The first section, titled 'Customer Information', contains three numbered items: '1. Full name:' followed by a text input field, '2. Password:' followed by a text input field, and '3. Email:' followed by a text input field. The second section, titled 'Health information:', contains two labels, 'Height' and 'Weight', each followed by a text input field. A vertical scrollbar is visible on the right side of the form area.

File Edit View Favorites Tools Help

Customer Information

1. Full name:

2. Password:

3. Email:

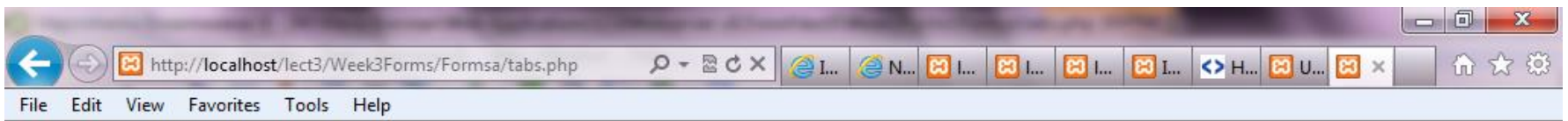
Health information:

Height Weight

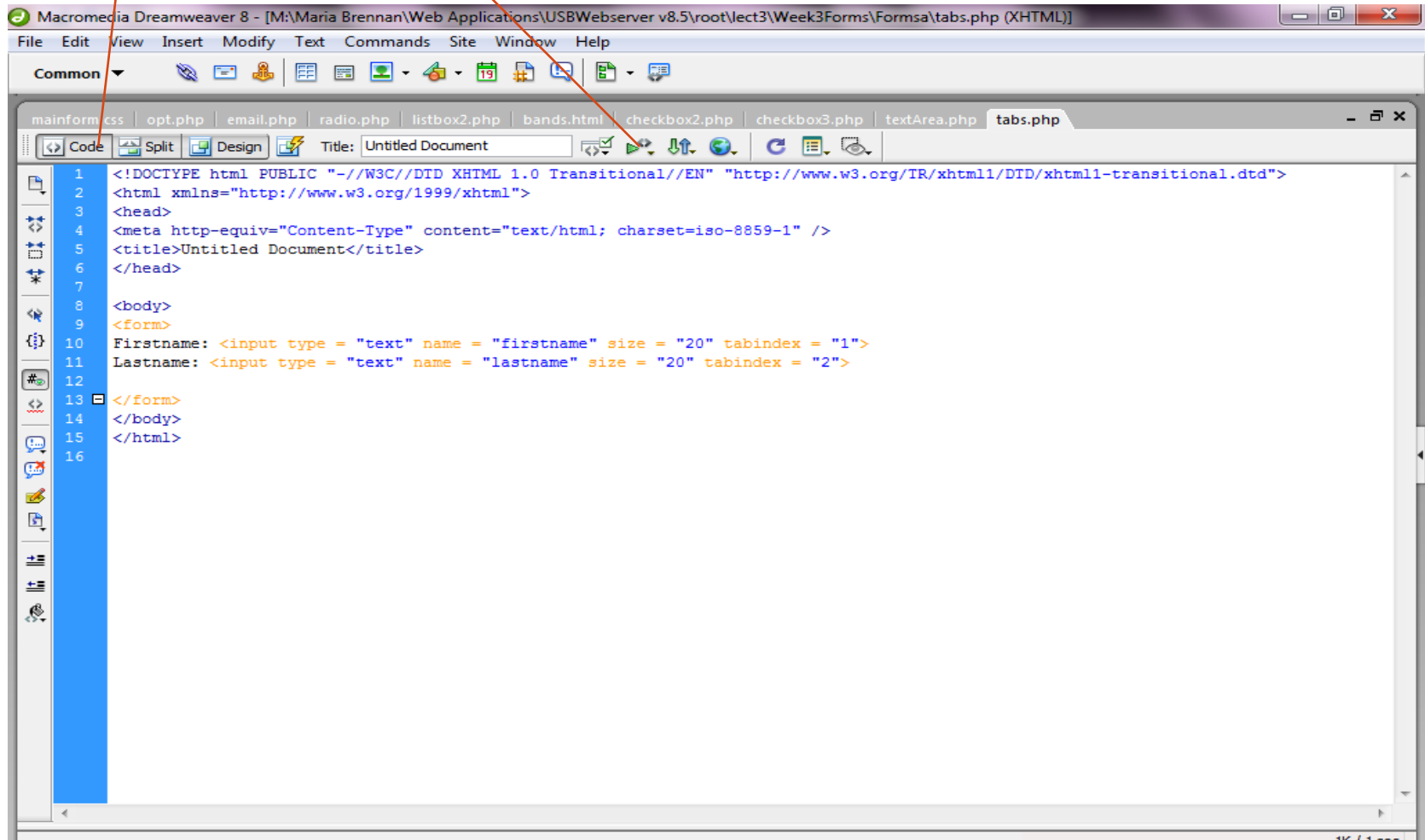
SET TAB ORDER OF CONTROLS

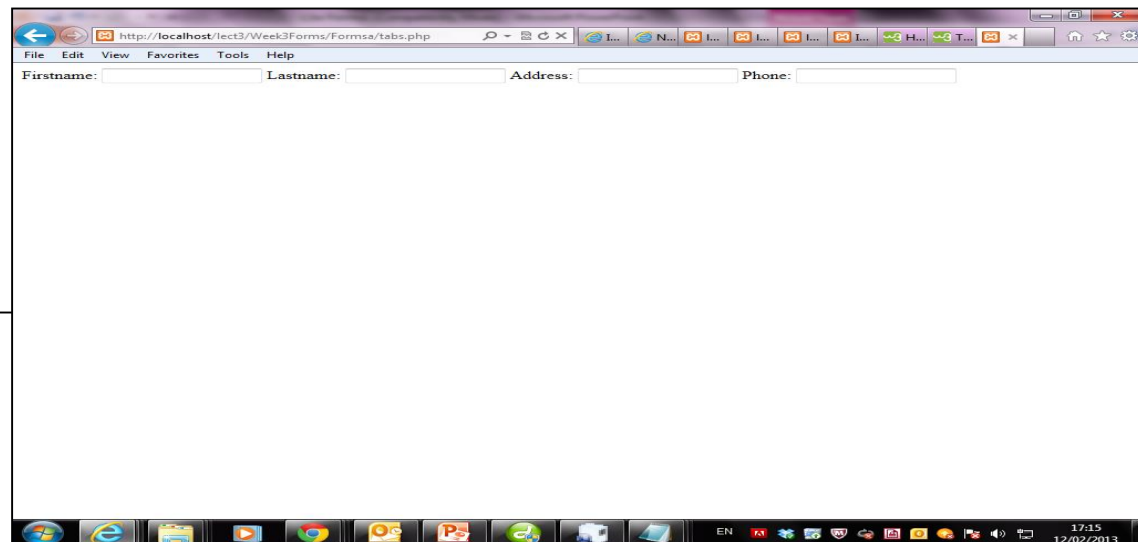
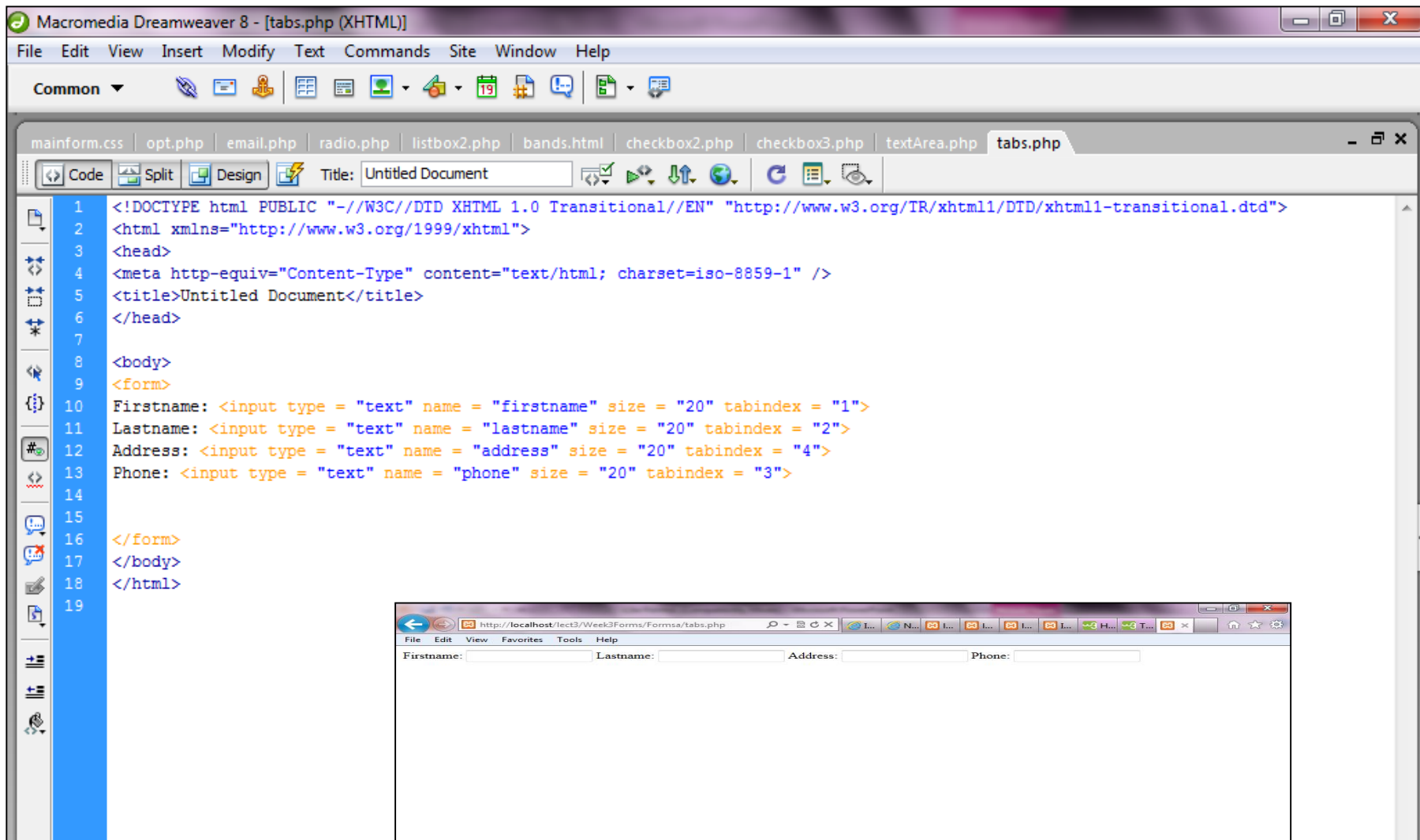
- **Tab order** – determines the **sequence** in which the controls on a form receive the focus from the tab key when the Tab key is pressed
- By default, tab order is same as the sequence in which the controls are coded in HTML document
- To modify tab order use the **tabindex** attribute

```
<input type = "text" name = "firstname" size = "20" tabindex = "1">  
<input type = "text" name = "lastname" size = "20" tabindex = "2">
```



Firstname: Lastname:





File Uploading

- You can also provide a facility to upload files using the following code:
- `enctype="multipart/form-data"` (must be included)
- `<input type = "file" ...>`

```
<form enctype="multipart/form-data" action="upload.php" method="post" >
<h2>What files are you sending?</h2>
<p><input type= "file" name="uploadfile" size="30" />
</form>
```

What files are you sending?

Macromedia Dreamweaver 8 - [M:\Maria Brennan\Web Applications\USBWebserver v8.5\root\lect3\Week3Forms\Forms\uploading.php (XHTML)]

File Edit View Insert Modify Text Commands Site Window Help

Common

mainform.css | opt.php | email.php | radio.php | listBox2.php | bands.html | checkBox2.php | checkBox3.php | textArea.php | tabs.php | uploading.php

Code Split Design Title: Uploading Files

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5   <title>Uploading Files</title>
6 </head>
7 <body>
8
9
10 <form method="post" enctype="multipart/form-data" action="upload.php">
11
12
13 <h2>What files are you sending?</h2>
14
15 <p><input type="file" name="uploadfile" size="30" />
16
17 </form>
18 </body>
19 </html>
20
21 </body>
22 </html>
23
```

The enctype attribute specifies how the form-data should be encoded when submitting it to the server.

Note: The enctype attribute can be used only if method="post".

<body> <form>

1K / 1 sec

EN

17:17
12/02/2013



What files are you sending?

Forms using Tables

— Personal Information —

First Name:	<input type="text"/>
Last Name:	<input type="text"/>
Address:	<input type="text"/> <input type="text"/>
City:	<input type="text"/>
State:	<input type="text" value="Select a State/Province"/> ▼
Zip/Postal Code:	<input type="text"/>
Country:	<input type="text" value="US - United States"/> ▼

Personal Information

Please enter your name and address as they are listed for your debit card, credit card, or bank account.

— Contact Information —

	How to Contact You? —
	<input type="radio"/> Phone <input checked="" type="radio"/> Email
Email:	<input type="text"/>
	We will never sell or disclose your email address to anyone. Once your account is setup, you may add additional email addresses.
Re-enter Email:	<input type="text"/>
	Must match the email address you just entered above.

Contact Information

Please enter your full email address, for example, **name@domain.com**

It is important that you provide a valid, working email address that you have access to as it must be verified before you can use your account.

Please enter a land line number, not a mobile phone number.

FORMS AND TABLES

- Tables are a great way to align form details in a neat orderly fashion.
- The following examples show with and without table formats:

Name:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="Send"/>	

**Without
Tables**

Name:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="Send"/>	

With Tables

FORMS AND TABLES

- Example without using a Table

```
<form method="post" action="mailto:youremail@email.com">  
Name:  
<input type="text" size="10" maxlength="40" name="name">  
</br>  
Password:  
<input type="password" size="10" maxlength="10" name="password">  
</br>  
<input type="submit" value="Send">  
</form>
```

Name:

Password:

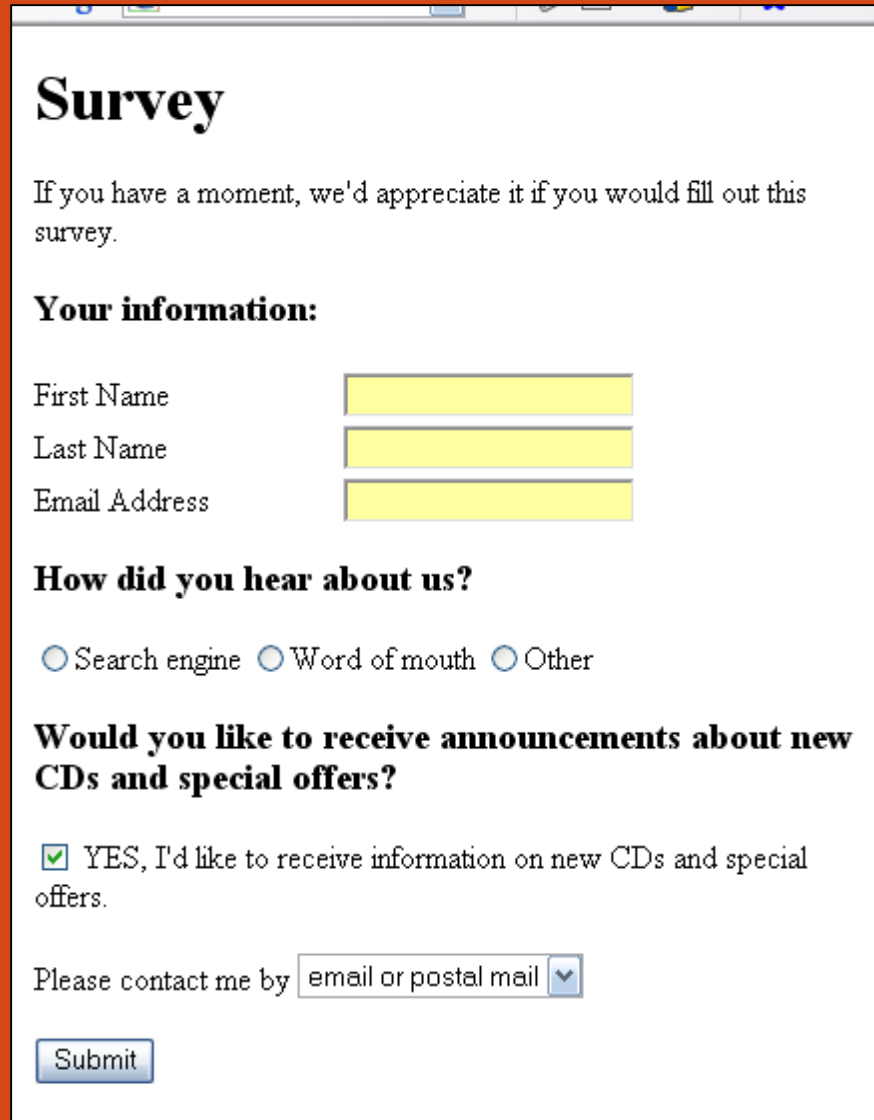
FORMS AND TABLES

- Example using a Table

```
<form method="post" action="mailto:youremail@email.com">
<table>
<tr>
<td>Name:</td>
<td><input type="text" size="10" maxlength="40" name="name"></td>
<td></br></td></tr>
<tr><td>Password:</td>
<td><input type="password" size="10" maxlength="10" name="password"></td>
<td></br></td></tr>
<tr><td><input type="submit" value="Send">
</td></tr>
</table>
</form>
```

Name:	<input type="text"/>
Password:	<input type="password"/>
<input type="submit" value="Send"/>	

What form elements are on this form?



Survey

If you have a moment, we'd appreciate it if you would fill out this survey.

Your information:

First Name

Last Name

Email Address

How did you hear about us?

☐ Search engine ☐ Word of mouth ☐ Other

Would you like to receive announcements about new CDs and special offers?

☒ YES, I'd like to receive information on new CDs and special offers.

Please contact me by

FORMATTING FORMS WITH CSS

48

RECAP – CSS

- **Block Level Elements**

- force a line break before and after the element.

- **Inline Level Elements**

- appear inline with the text and do not cause line breaks.

- **Replaceable Inline Elements**

- These elements insert content into the page. They have dimensions for width and height. eg. Images

**

BLOCK LEVEL ELEMENTS

● `<div>`

● `<h1, h2, h3.....h6>`

● `<p>`

● `<blockquote>`

Definition List

● `<dd>`

● `<dl>`

● `<dt>`

● Ordered List: ``

● Unordered List: ``

● List Item ``

● `<table>`

● `<tr>`

● `<td>`

● `<th>`

INLINE ELEMENTS

- ****

- <a>

-

- <code>

-

- <pre>(fixed width)

- <sub>(subscript)

- <sup>(superscript)

There were more inline elements, in previous versions of HTML, but they have now become deprecated features.

INLINE ELEMENTS

- These all have defined dimensions for **width** and **height**.

- ``

- Form elements

- `<input>`

- `<select>`

- `<textarea>`

New HTML5 Elements – for defining Structure of a Page

`<header>`

`<nav>`

`<article>`

`<section>`

`<section>`

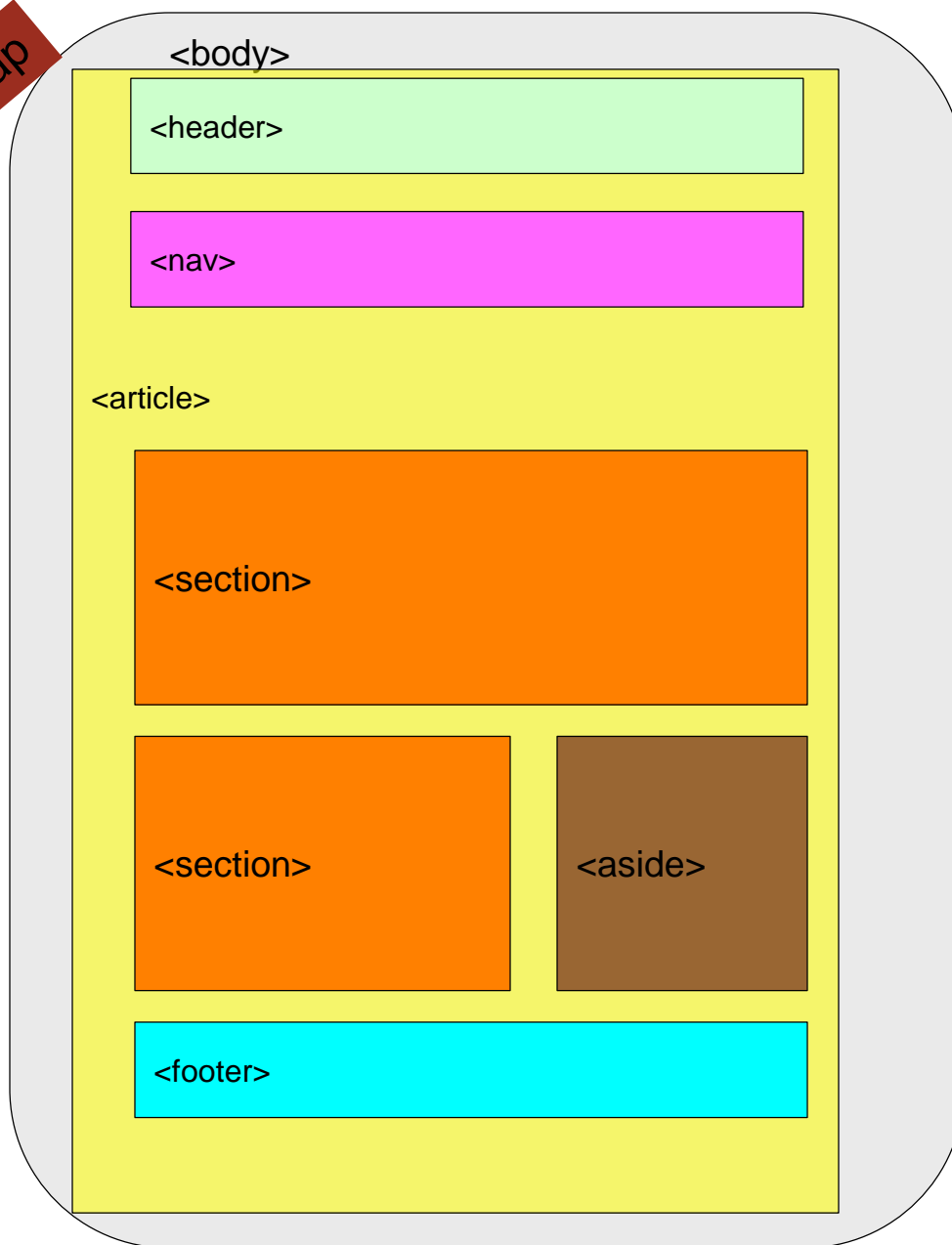
`<section>`

`<aside>`

`<footer>`

recap

recap



An HTML5 page illustrating the use of header, nav, section, aside, and footer elements.

APPLYING A STYLE TO ALL ELEMENTS ON THE PAGE

CSS– External Style Sheet

CSS rules may be declared using:

- any HTML element (**tag**) selector e.g.
 - ▣ body, br, h1...h6, p
 - ▣ table, tr,td,th,caption
 - ▣ ul, ol, li, dd,dt
 - ▣ and now..... form, fieldset, legend, input, selectetc.
- **class** selectors
- **id** selectors

ID SELECTOR

- Used with single instances of an HTML element
- id attribute is set to a unique name
- Can only be used once

syntax

```
#idname {  
  property: value;  
  property: value;  
  ... }
```

example

```
#idEg1 {  
  
  font-family: sans-serif;  
  border: thin solid red;  
  padding-left: 1px;  
  color: #330;  
}
```

```
<div id="idEg1">
```

This text is enclosed in a
block defined by div

```
</div>
```


APPLYING STYLES TO IDS & CLASSES

To apply a style to an element based

- on the value of its **id** attribute

#id {styles}

NOTE: # sign
before id

Class attribute

- Used to identify a group of elements

To apply a style to a group of elements based on value of **class** attribute

.class {styles}

▪ NOTE: dot
before classname

FORM ELEMENTS

CSS AND FORMS

```
<input type="text" name="lastname" value="lastname">
```

- Input type
- **Name** : The name attribute is used to reference elements in a JavaScript, or to reference form data after a form is submitted.

**<INPUT TYPE="TEXT" NAME="LASTNAME"
VALUE="LASTNAME">**

- **Value :**The value attribute specifies the value of an <input> element.
- **The value attribute is used differently for different input types:**
 - ❖ For "button", "reset", and "submit" - it defines the text on the button
 - ❖ For "text", "password", and "hidden" - it defines the initial (default) value of the input field
 - ❖ For "checkbox", "radio", "image" - **it defines the value associated with the input (this is also the value that is sent on submit)**

Laying out a Form using CSS

- Then **you can** float them **within the form** and lay out the form just like you would any other block element.
- If your labels have large amounts of text, you'll need to play with the widths.

Firstname:

Last Name

Password:

Submit

You are being specific here

```
input[type=text]{  
  width: 100%;  
  padding: 12px 20px;  
  margin: 8px 0;  
  display: inline-block;  
  border: 1px solid #ccc;  
  border-radius: 4px;  
  box-sizing: border-box;  
}
```

The CSS3 **box-sizing** property allows us to include the padding and border in an element's total width and height.

If you set **box-sizing: border-box;** on an element padding and border are included in the width and height:

An alternative to using a table for nice layout!


Label.html formatted with label.css

Firstname:

Last Name

Password:

Submit



```
input[type=submit]
{
    width: 100%;
    background-color: #4CAF50;
    color: white;
    padding: 14px 20px;
    margin: 8px 0;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}


input[type=submit]:hover
{
    background-color: red;
}
```

Firstname:

Last Name

Password:

Submit



```
input[type=submit]
{
  width: 100%;
  background-color: #4CAF50;
  color: white;
  padding: 14px 20px;
  margin: 8px 0;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

input[type=submit]:hover
{
  background-color: #45a049;
}
```


Firstname:

LastName:

Password:

```
form{  
  width: 50px ;  
  border: 3 px blue ;  
  background-color: olive;  
  padding: 4em ;  
  etc.  
}
```

```
input[type=submit] {  
  background-color:red;  
  width: 14em;  
  border: 3px red ;  
  background-color: pink;  
  padding: 4em;}
```

FIELDSET & LEGEND....

- **Fieldsets and Legends** make your forms more usable
- Help group similar portions of your form together under a heading
- Most people **can't process more than 5 to 7 items at a time**
- If your form is longer than that, you'll want to group it into sections.
- Can create a fieldset without a legend, but the legend adds even more **usability**
- Can style fieldset and legend
- Looks nice 😊

FIELDSET & LEGEND.....

```
<fieldset>
<legend>Phone Numbers</legend>
  <label for="home">Home Phone</label>
  <input type="text" name="home"
id="home">
<br>
  <label for="work">Work Phone</label>
  <input type="text" name="work"
id="work">
<br>
  <label for="cell">Cell Phone</label>
  <input type="text" name="cell" id="cell">
</fieldset>
```

fieldsetLegend.html

Styling with Fieldset & Labels

Phone Numbers

Home Phone

Work Phone

Cell Phone

Personal Details

Firstname:

Last Name

Password:

Submit

STYLING FIELDSET & LEGEND.....

Fieldset

- Border, width, colour

Legend

- Colour, padding

Input & labels

- Float left

Display as block

Width --- x

As before

Input

- color, width

Label

- color, width

STYLING THE FIELDSET, LEGEND & INPUT

Styling with Fieldset & Labels

Phone Numbers	
Home Phone	<input type="text"/>
Work Phone	<input type="text"/>
Cell Phone	<input type="text"/>

Personal Details

Firstname:

Last Name

Address

Submit

STYLING THE FIELDSET, LEGEND & INPUT

Personal Details

Firstname:

Last Name

Address

Submit

```
fieldset
{
    border: 0.9em;
    width: 30em;
    background-color: grey;
}

legend {
    border: 10;
    background-color: grey;
    color: blue;
    padding: 0.2em 0.5em;
}
```

OPTGROUP TAG....1



Optgroup Tag works with **drop-down** or **select lists** the same way that **fieldset** works with entire forms.

- Surround your option elements with the **optgroup** element.
- Adds **headlines** and **groupings** to your options list
- Makes them easier to understand and use - especially for long lists
- Can style your *optgroup* to provide color coding or other **visual cues** to the different sections.
- Some browsers display the *optgroup* labels in bold text, others with a dark background colortest in multiple browsers

STYLED OPTGROUP...2

Name your favorite pet

<label id = "pets" >Name your favorite pet

<select name="pets" id = "pets">

<optgroup label="mammals">

<option>dog</option>

<option>cat</option>

<option>rabbit</option>

<option>horse</option>

</optgroup>

<optgroup label="reptiles">

<option>iguana</option>

<option>snake</option>

</optgroup>

</select>

</label>



optGroup.html

STYLED OPTGROUP.....

optGroup.css

```
select {  
  background-color: yellow;  
}  
  
optgroup {  
  background-color: tan;  
  color: brown;  
  font-weight: bold;  
}  
  
optgroup option {  
  background-color: khaki;  
  color: #000;
```

Login Form - no Styles Applied

login.html

```
<form action = "showform.html" method = "post" >
<br>
<label for = "name" >Username: </label>
    <input type="text" name = "name" id = "name" >
<br>
<label for = "password" > Password: </label>
    <input type="password" name = "password" id = "password" >
<br>
    <input type="submit" id="submit" value="Submit" >
    <input type="reset" id="reset" >
<br>
</form>
```

Firstname:

Lastname:

Password:

Login Form – Style Applied

login.html

Firstname:

Lastname:

Password:

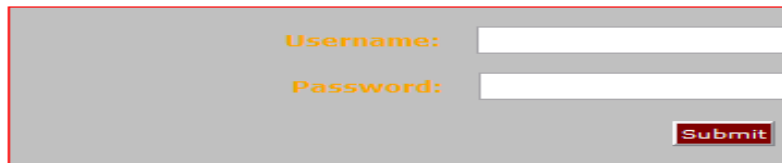
```
input[type=text]{  
  width: 30%;  
  padding: 12px 20px;  
  margin: 8px 0;  
  display: inline-block;  
  border: 1px solid #ccc;  
  border-radius: 4px;  
  box-sizing: border-box;  
}
```

```
input[type=submit] {  
  width: 10%;  
  background-color: #4CAF50;  
  color: white;  
  margin: 8px 0;  
  border: none;  
  border-radius: 4px;  
  cursor: pointer;  
  padding-left: 20px;  
}
```

```
input[type=reset] {  
  width: 10%;  
  background-color: #4CAF50;  
  color: white;  
  margin: 8px 0;  
  border: none;  
  border-radius: 4px;  
  cursor: pointer;  
  padding-left: 20px;  
}
```

COLOUR ON FORMS ...

- **Colors** on form elements make them a lot more usable, easier to understand and fill out.
- Like with almost all HTML elements, you can change the **background colors**, **foreground or text colors**, and **border colors**.
- Modifying the form colors is a great way to improve their *usability*

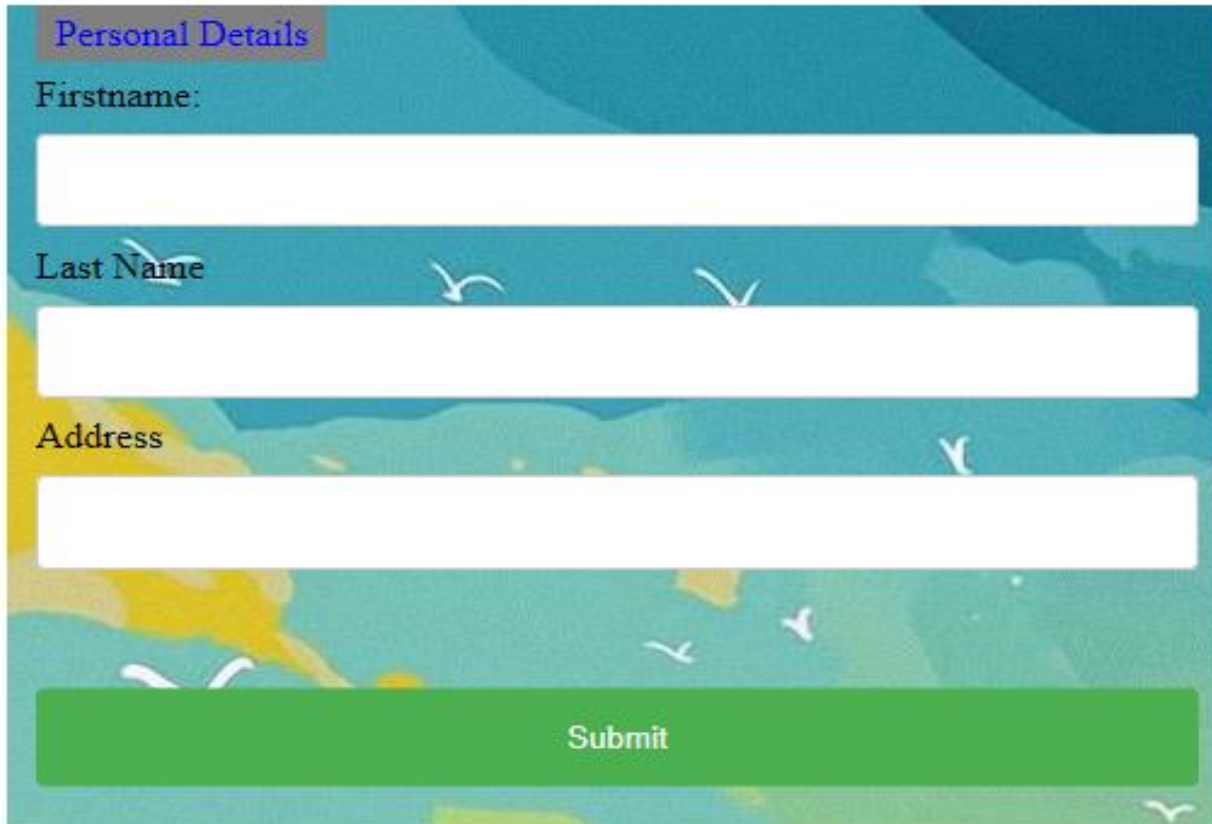


Username:

Password:

[View login2.html](#)

Login Form – Style Applied



Personal Details

Firstname:

Last Name

Address

Submit

```
fieldset
{
    border: 0.9em;
    width: 30em;
    background-image: url("back.jpg");
}
```

ANOTHER FORM STYLING EXAMPLE

```
<form action = "#" method = "post">
<label for="user">Name</label>
    <input type="text" name="user" id = "user" >

    <br>
<label for="emailaddress">Email Address:</label>
    <input type="text" name="emailaddress" id = "emailaddress" >

    <br>
<label for="comments">Comments:</label>
    <textarea name="comments" id = "comments" ></textarea>

    <br>
<label for="terms">Agree to Terms?</label>
    <input type="checkbox" name="terms" id = "terms" >

    <br>
    <input type="submit" name="submitbutton" id="submitbutton"
    value="Submit" >

    <br>
</form>
```

1. FORM WITHOUT ANY CSS

Name

Email Address:

Comments:

Agree to Terms? ☐

2. STYLING THE FORM WITH CSS

```
label{  
  display:block;  
width: 120px;  
font-weight: bold;  
}  
  
input, textarea{  
width: 180px;  
margin-bottom: 5px;  
}  
  
textarea{  
width: 250px;  
height: 150px;  
}
```

```
#terms{  
width: 1em;  
}  
  
#submitbutton{  
margin-left: 120px;  
margin-top: 5px;  
width: 90px;  
}
```

3. FORM STYLED WITH CSS

```
form{  
margin-left: 100px;  
width:      450px;  
background-color: tan;  
border: 3px solid khaki;  
}
```

....

Name

Email Address:

Comments:

Agree to Terms? ☐

STYLING A FORM

- Can use CSS with any of the following:
 - Form
 - Input
 - Fieldset
 - Legend
 - Select
 - Text, textarea, optgroup, optgroup option, button etc.

STYLING A FORM

- No hard and fast rules for styling
- Tips overleaf
- Plan the visual layout of your form in advance

- set **body** *background color* (if appropriate)
- set **form** *padding / margin* to zero (then override if appropriate)
- set form *width, background-colour, border.*
- set form **list items** to **list-style none** [if using lists]
- create **fieldset** and write **legends**
 - may need small right margin, to separate labels from inputs

RADIO BUTTONS - RECAP

- You can only select one radio button from a group
- They all have the same **name** attribute
- They all have different **values**

Size:

Size: ☐ Small ☐ Medium ☐ Large

```
<input type="radio" name="size" value="small">Small
```

```
<input type="radio" name="size" value="medium"> Medium
```

```
<input type="radio" name="size" value="large">Large
```

```
<br>
```

CHECKBOXES – RECAP

- `<input type = "checkbox">`
- They all have a different **name** attribute
- **name, value, id** = "different for each"

```
<input type= "checkbox" name="bike" id = "bike"  
value = "bike" >Bike <br>
```

```
<input type= "checkbox" name="car" id = "car"  
value= "car" >Car <br>
```

```
<input type= "checkbox" name="bus" id = "bus"  
value= "bus" >Bus<br>
```

Form with Radio Buttons & Checkboxes

The form contains the following elements:

- Name:** A text input field.
- Email Address:** A text input field.
- Feedback:** A large text area.
- Gender:** Three radio buttons labeled "Male", "Female", and "other radio".
- Hobbies:** Three checkboxes labeled "Tennis", "Reading", and "Basketball".
- Agree to Terms?:** A single checkbox.
- Submit** and **Reset** buttons at the bottom.

Annotations on the right side of the form:

- An arrow points from the "Radio Buttons - horizontal" box to the "Gender" radio buttons.
- An arrow points from the "Checkboxes - vertical" box to the "Hobbies" checkboxes.

1. RADIO BUTTONS — ALIGN HORIZONTALLY

- This form (*choice.html*) is styled as if it were composed of 2 columns: **labels on the left** and **form input fields on the right**
- Going to line up the **radio buttons horizontally** within a styled form

The form is styled with a two-column layout. Labels are on the left, and input fields are on the right. Dashed horizontal lines separate the sections. Labels are highlighted with red boxes.

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<input type="text"/>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> other radio
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

1. CHECKBOXES — ALIGN VERTICALLY

- Going to align the **checkboxes vertically**, relative to the labels
- Label on left-handside
- Checkboxes – one under the next on right-handside

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<input type="text"/>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> other radio
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

FORM WITH RADIO BUTTONS & CHECKBOXES

choices.html

Name

Email Address:

Feedback:

Gender: ☐ Male ☐ Female ☐ other radio

Hobbies: ☐ Tennis ☐ Reading ☐ Basketball

Agree to Terms? ☐

Radio
Buttons -
horizontal

Checkboxes - vertical

CHOICES.HTML

Labels and input tags done, as in previous lecture

Using paragraphs `<p>` tags to separate the various labels this time.

Radio Buttons

- `` class used around all the radio buttons in order to display the text beside the radio button

Checkboxes

- `<class = "boxfix1">` put around 2nd and 3rd checkbox to align them with 1st checkbox

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<input type="text"/>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> other radio
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

CHOICES.HTML

- `<div id = "buttons">` put around the submit & reset buttons to align them in the centre of the form

The form contains the following elements:

- Name:** A text input field.
- Email Address:** A text input field.
- Feedback:** A large text area.
- Gender:** Radio buttons for Male, Female, and other radio.
- Hobbies:** Checkboxes for Tennis, Reading, and Basketball.
- Agree to Terms?:** A checkbox.
- Buttons:** Submit and Reset buttons at the bottom.

A red arrow points from the text in the list above to the Submit and Reset buttons.

CHOICES.HTML WITH LABELS (WILL COVER THESE AGAIN)

```
<label for="email">Email Address:</label>
```

```
<input type="email" name="email" id="email" >
```

```
<label for="comments">Feedback:</label>
```

```
<textarea name="comments" id="comments"></textarea>label >Gender:</label>
```

```
<input type="radio" name="gender" value="male" >
```

```
<span class="radio"> Male </span>
```

CHOICES.HTML

```
<label>Hobbies:</label>
```

```
  <input type="checkbox" name="tennis" id = "tennis" value = "tennis">  
Tennis<br>
```

```
  <input type="checkbox" name="reading" id = "reading" value = "reading"  
class="boxfix1" > Reading<br>
```

```
  <input type="checkbox" name="basketball" id = "basketball" value =  
"basketball" class = "boxfix1" > Basketball<br>
```

.....

```
<div id = "buttons">
```

```
  <input type="submit" value="Submit" >
```

```
  <input type="reset" value="Reset" >
```

```
</div>
```

Name

Email Address:

Feedback:

Gender: ☐ Male ☐ Female ☐ other radio

Hobbies: ☐ Tennis ☐ Reading ☐ Basketball

Agree to Terms? ☐

Submit Reset

CHOICES.CSS

```
form {  
    width:450px;  
}  
p{  
    margin: 0;  
    padding: 5px 0 12px 0;  
    border-top: 1px dashed gray;  
}  
  
label, input{  
    display:block;  
    margin-bottom: 5px;  
}
```

```
/* style label */  
label{  
    font-weight:  
    bold;  
    border: 1px solid red;  
    width: 140px;  
    margin-right:  
    3px;  
}  
  
/* textarea - give it dimensions */  
textarea{  
    width: 250px;  
    height: 100px;  
    margin-left: 10px;  
}
```


CHOICES.CSS

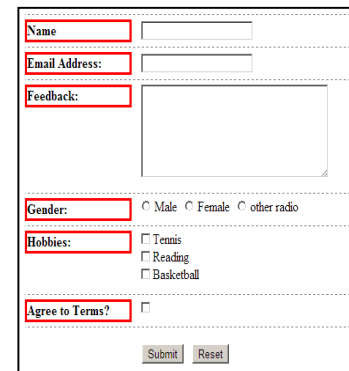
- Align all the input elements i.e using input tag in css
- i.e. `<input type = " " >` text, password, email, radio, checkbox

```
/* move ALL input fields in */
```

```
input{
```

```
    margin-left: 10px;
```

```
}
```



A web form with the following elements:

- Name:
- Email Address:
- Feedback:
- Gender: ☐ Male ☐ Female ☐ other radio
- Hobbies: ☐ Tennis ☐ Reading ☐ Basketball
- Agree to Terms? ☐
- Submit Reset

CHOICES.CSS

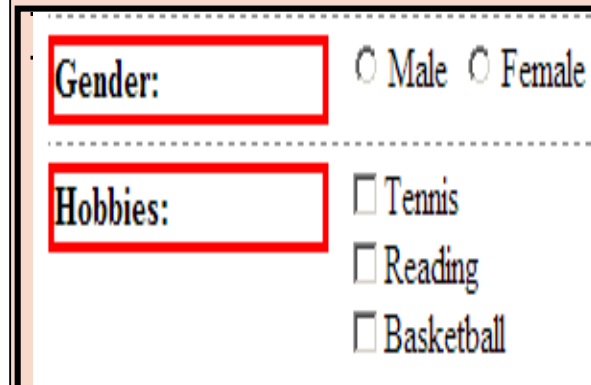
```
/* radio span class -  
radio buttons line up  
horizontally*/
```

```
/* to align 2nd , 3rd  
checkbox vertically */
```

```
.boxfix1{  
    margin-left: -  
    16px;  
}
```

```
/* align buttons */
```

```
#buttons{  
    margin-left:  
    150px;
```



The image shows a preview of a form with two sections. The first section, labeled 'Gender:', has two radio button options: 'Male' and 'Female'. The second section, labeled 'Hobbies:', has three checkbox options: 'Tennis', 'Reading', and 'Basketball'. The form is enclosed in a black border, and the labels 'Gender:' and 'Hobbies:' are highlighted with red boxes. The entire form is set against a light orange background.

ALIGNING MORE RADIO BUTTONS HORIZONTALLY - HTML

Going to put in three radio buttons and align them horizontally

**** class is put around each radio buttons

```
<label>Card Type:</label>
```

```
  <input type="radio" name="card" value =  
  "visa" >
```

```
    <span class = "radio"> Visa
```

```
</span>
```

```
  <input type="radio" name="card" value =  
  "access" >
```

```
    <span class =  
  "radio">Access</span>
```

```
  <input type="radio" name="card" value =  
  "paypal" >
```

```
    <span class =  
  "radio">Paypal</span>
```

- No changes need to be made to css defined already (from previous program)

```
/* used to line up  
radio buttons */
```

```
.radio{  
  
float:left;  
  
}
```

Aligning radio buttons horizontally & one row of checkboxes horizontally

Name

Email Address:

Feedback:

Card Type: ☐ Visa ☐ Access ☐ Paypal

Hobbies: ☐ Athletics ☐ Basketball ☐ Canoeing

Agree to Terms? ☐

formboxes_Horizontal.html

ALIGNING CHECKBOXES HORIZONTALLY

Using a **span** class called **box** to line one row of checkboxes up horizontally

```
<label>Hobbies:</label>
  <input type="checkbox" name="athletics"
value = "athletics" >
<span class = "box">Athletics </span>
  <input type="checkbox" name="basketball"
value = "basketball" >
<span class = "box">Basketball</span>
  <input type="checkbox" name="canoeing"
value = "canoeing" >
    <span class = "box">Canoeing</span>
<br>
```

- Used the [attribute selector](#) for checkbox to line up checkboxes.
Used **span class** called **box** to float checkboxes

to left

```
/* used to line up checkboxes
*/

.box{
    float: left;
}
```

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<input type="text"/>
Card Type:	<input type="radio"/> Visa <input type="radio"/> Access <input type="radio"/> Paypal
Hobbies:	<input type="checkbox"/> Athletics <input type="checkbox"/> Basketball <input type="checkbox"/> Canoeing <input type="checkbox"/> Dancing <input type="checkbox"/> Skating <input type="checkbox"/> Swimming
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="reset"/>	

ALIGNING 2 ROWS OF CHECKBOXES HORIZONTALLY

- One way to align, the second row of checkboxes with the first row, is to **create a blank label** in order to set up the spacing correctly for the extra checkboxes.
- This blank label is called **id = “row2”**
- In our case, the blank label – takes most of the CSS properties associated with the <p> tag and the <label>

3. ALIGNING 2 ROWS OF CHECKBOXES HORIZONTALLY

```
<label>Hobbies:</label>
    <input type="checkbox" name="athletics" value =
    "athletics" >
<span class = "box">Athletics </span>
    <input type="checkbox" name="basketball" value =
    "basketball" >
<span class = "box">Basketball</span>
    <input type="checkbox" name="canoeing" value =
    "canoeing" >
<span class = "box">Canoeing</span> <br>
<label id = "row2"></label>
    <input type="checkbox" name="dancing" value =
    "dancing" >
<span class = "box">Dancing</span>
    <input type="checkbox" name="skating" value =
    "skating" >
<span class = "box">Skating</span>
    <input type="checkbox" name="swimming" value =
    "swimming" >
<span class = "box">Swimming</span>
```

#row2

```
{
clear: left;
float: left;
display: block;
margin-bottom: 5px;
margin-right: 3px;
}
```

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<div></div>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> XXX
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Clear"/>	

ALIGNING RADIO BUTTONS & CHECKBOXES VERTICALLY

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<div></div>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> XXX
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Clear"/>	

4 ALIGNING RADIO BUTTONS VERTICALLY

Code added for a 3rd radio button below

 added after each radio button

Class “boxfix” created to align the 2nd, 3rd radio button with first radio button

```
<label>Gender:</label>
```

```
<input type="radio" name="gender" value="male" > Male
```

```
<br>
```

```
<input type="radio" name="gender" value="female" class="boxfix"> Female  
<br>
```

```
<input type="radio" name="gender" value="xxx" class="boxfix"> Other
```

```
</p>
```

```
/* using attribute selector for radio buttons -- input[type = "radio"] */  
input[type = "radio"]{  
    margin-left: 10px;  
    margin-right: 5px;  
}  
/* used so that labels & radio buttons line up as expected */  
.radio {  
    float: left;  
}  
/* Position the second radio button etc - after a line break */  
br + .boxfix{  
    float:left;  
    margin-left: -18px;  
}
```

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<div></div>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> XXX
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Clear"/>	

ALIGNING RADIO BUTTONS VERTICALLY

br + .boxfix

- Means: Select all elements in the class **boxfix** which are placed immediately after a **
** tag

CSS format:

Selector + selector (br + .boxfix)

- Selects all elements for 2nd selector which are placed immediately after the 1st selector

ALIGN CHECKBOXES VERTICALLY

```
<input type="checkbox" name="tennis" value = "tennis"  
      id = "tennis"> Tennis<br>
```

```
<input type="checkbox" name="reading" value = "reading"  
      id = "reading" class="boxfix1" > Reading<br>
```

```
<input type="checkbox" name="basketball" value =  
"basketball"      id = "basketball" class = "boxfix1" >  
Basketball<br>
```

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<input type="text"/>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> XXX
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Clear"/>	

ALIGN CHECKBOXES VERTICALLY

```
/* Position the Checkboxes  
vertically */
```

```
.boxfix1{  
    margin-left: -16px;  
}
```

Name	<input type="text"/>
Email Address:	<input type="text"/>
Feedback:	<div></div>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> XXX
Hobbies:	<input type="checkbox"/> Tennis <input type="checkbox"/> Reading <input type="checkbox"/> Basketball
Agree to Terms?	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Clear"/>	

RECAP ON SOME NEW CSS

- Can select a particular type of attribute using the **attribute selector**
 - `input[type="text"]`
 - `input[type="password"]`
 - `input[type="radio"]`
 - `input[type="checkbox"]`
 - `input[type="submit"]`

RECAP ON SOME NEW CSS

- Selector: *element + element*
- Example: div + p
- Means: Selects all <p> elements that are placed immediately after <div> elements
- Example in notes:
br + .boxfix
- Selects all elements in the class **boxfix** which are placed immediately after a
 tag

CSS & FORMS

- References:
- <http://webdesign.about.com/od/forms/a/aa050707.htm>
- http://www.cssdrive.com/index.php/examples/exampleitem/talesless_forms/
- http://webdesign.about.com/od/htmltags/p/bltags_optgroup.htm
- <http://usability.com.au/2013/04/accessible-forms-1-labels-and-identification/>