LECTURE 05
CSS & FORMS & LABELS

Recap on some CSS styling
Styling a Form
Labels
Marie Brennan

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

<img src = “picture.jpg” alt= “pic” >

Block level elements

- <div>
- <h1, h2, h3……h6>
- <p>
- <blockquote>

Definition List

- <dd>
- <dl>
- <dt>

- Ordered List: <ol>
- Unordered List: <ul>
- List Item <li>
- <table>
- <tr>
- <td>
- <th>

Inline Elements

- <span>
- <a href = “ ”>
- <strong>
- <code>
- <em>
- <small>
- <pre>
- <sub>
- <sup>

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

Replaceable Inline Elements

- These all have defined dimensions for width and height.

  - <img>

- Form elements

■ <form>

■ <input>

■ <select >

■ <textarea>

■ <input type = “text”>

■ <input type = “submit”>

Block & Inline Elements

- Block Elements
  - – force a line break before and after the element.
  - – may contain inline elements as well as other block-level elements.
- Inline Elements
  - – appear inline with the text and do not cause line breaks.
  - – may contain only data and other inline elements.

<div> and <span>

- They are Generic Container Elements
- <div> and <span>, are used by CSS to define a section of the document as either block-level or inline.
- <div> is a generic block-level element
- <span> is a generic inline-level element
- When used with the class and id attributes,

<div> and <span> effectively allow you to define your own HTML markup tags.

Applying a Style

- Techniques for applying style to an HTML document

– Inline Styles (don’t use !)

– Internal Style Sheet (don’t use !)

– External Style Sheet

External Style Sheet

- Simple text file that contains only style declarations
- Can be linked to any page in a Web site
- To make all h1 headings in a Web site appear in red font & centered, link each page to a common style-sheet containing:

```
h1 {
    color: red;
    text-align:center;}
```

External Style Sheet

External .css File (imported)

```
<head>
.....
    <style>
        @import  “css/test_style.css”;
    </style>
</head>
```

CSS Rule declaration

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*
  - *form, fieldset, legend, input, select …….etc.*
- class selectors
- id selectors

HTML element selectors

- The rule is applied to all instances of a specific HTML tag element

(e.g. all <p> elements)

syntax

```
selector {
    property: value;
    property: value; ...
}
```

Example

```
p {
    text-align: center;
    color: #060;
    font: italic;
    }

<p> Hello world </p>
```

Class Selector

- for ALL HTML element instances belonging to a class, specified via the class attribute
- Can use the class any number of times in your program

syntax

```
  .classname {
    property: value;
    property:value; …
    }
```

Example

```
.ex2 {
     border: 2px solid black;
     padding: 0.5rem;
     text-align: left;
     color:#900;
}
```

- <p class="ex2">
  Hello world </p>

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

```
#id1 {
    font-family:sans-serif;
    border: thin solid red;
    padding-left:0.5rem;
    color:#330;
}
<div id="id1">
This text is enclosed in a block
    defined by div
</div>
```

Setting the Element Display Style

- Most page elements are classified as either inline elements or block-level elements
- CSS can be used to change the display style applied to any element
- – Syntax

display: type;

e.g. display:block;

display:inline;   ....(lots more)

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

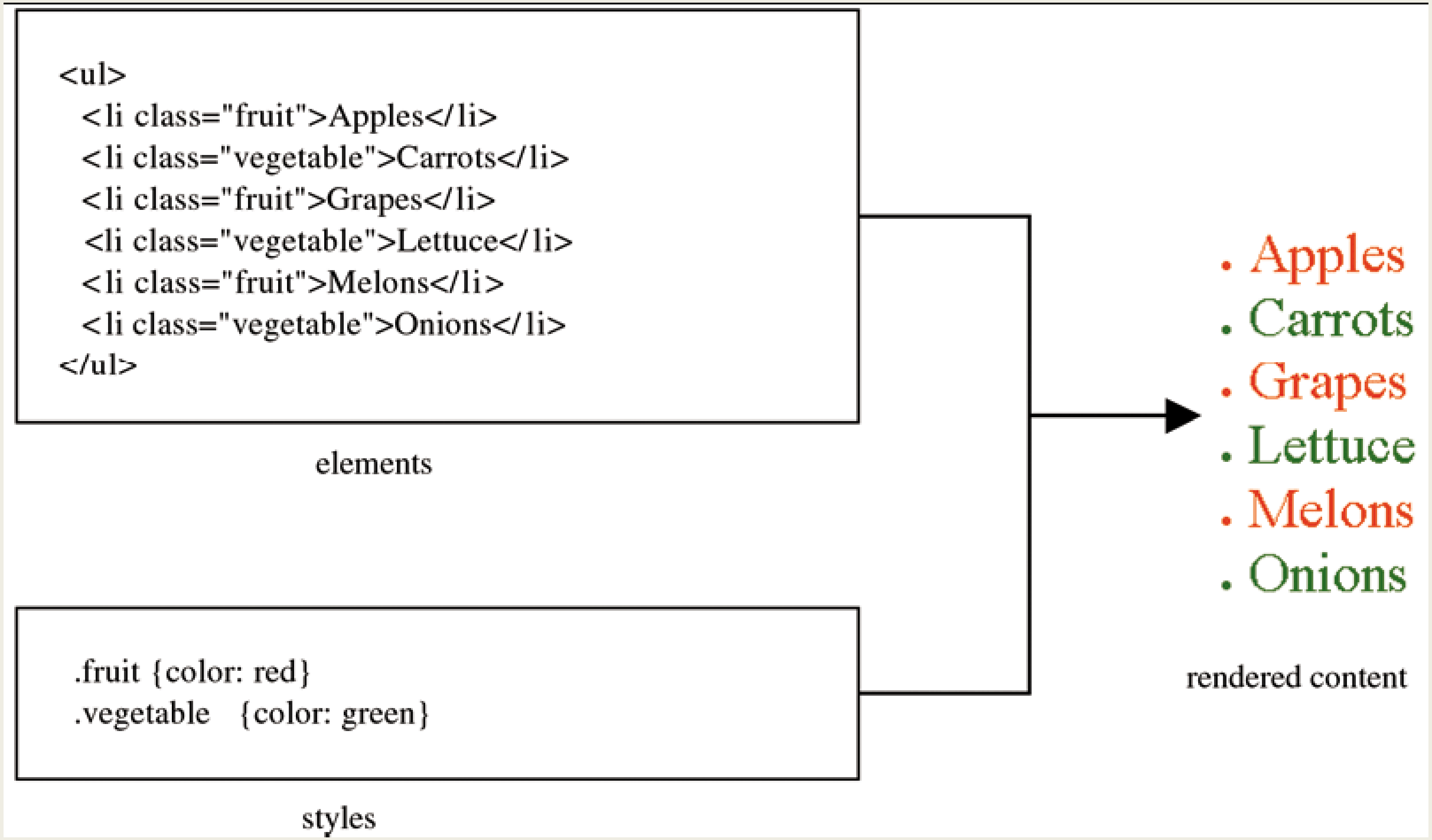
- Used to identify a group of elements

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

.class {styles}

- NOTE: *fullstop* sign before classname

Applying Styles to IDs & Classes



```
<ul>
  <li class="fruit">Apples</li>
  <li class="vegetable">Carrots</li>
  <li class="fruit">Grapes</li>
  <li class="vegetable">Lettuce</li>
  <li class="fruit">Melons</li>
  <li class="vegetable">Onions</li>
</ul>
```

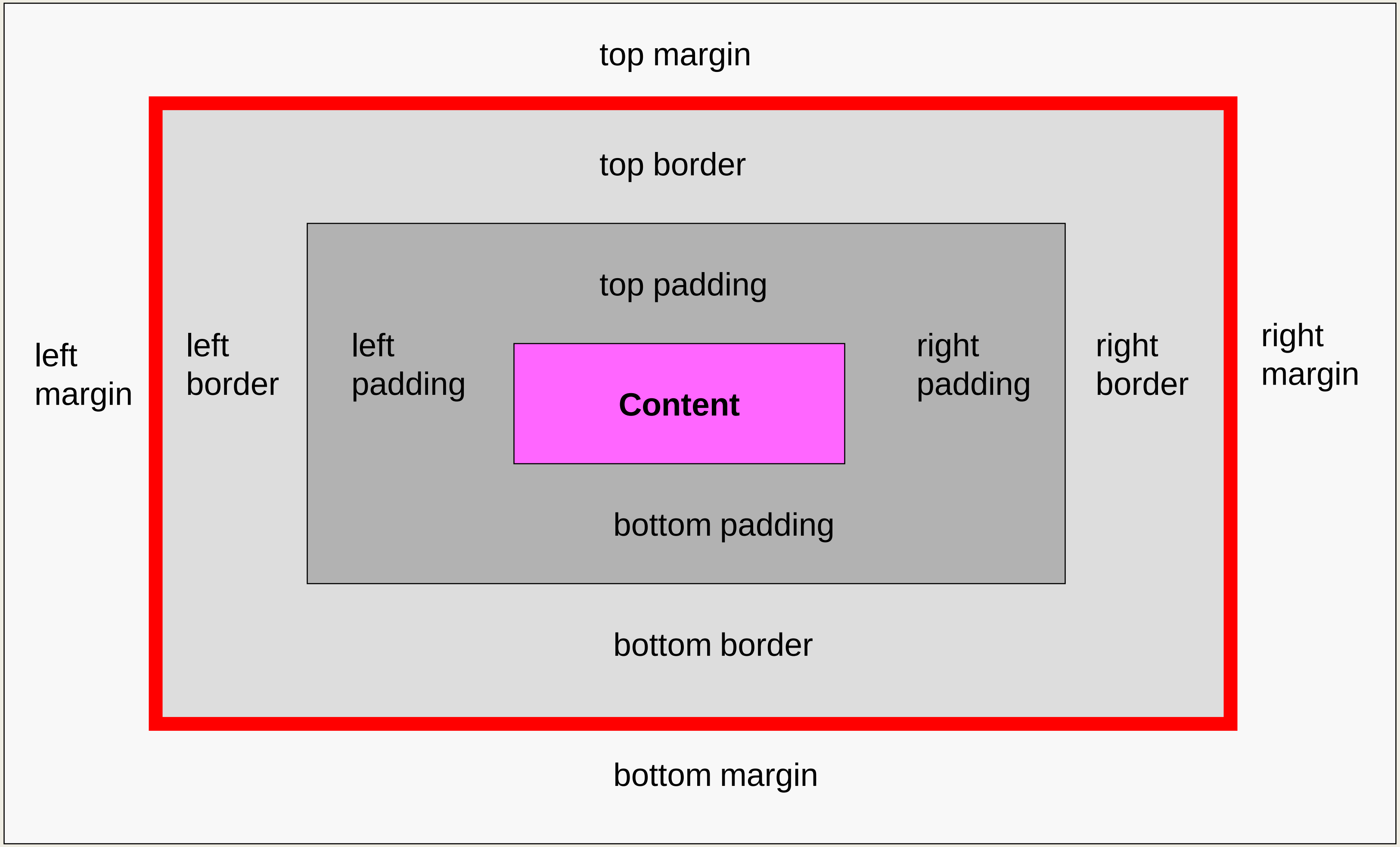
```
.fruit {color: red}
.vegetable   {color: green}
```

Borders, Margins, and Padding

Box model

- Identifies the different parts of the element
- Elements described by box model
- – Margin between the box and other
- elements
- – Border of the box
- – Padding between element’s content and
- border
- – Element’s content

The CSS Box Model



Borders, Margins, and Padding
Quick re-cap

border: 5px solid red;

margin-top: 100px;
margin-bottom: 100px;
margin-right:50px;
margin-left: 50px;

margin: 100px 50px;

padding-top: 25px;
padding-bottom: 25px;
padding-right: 50px;
padding-left: 50px;

padding: 25px;

Border Styles

- dotted:
- dashed:
- solid:
- double:
- groove:
- ridge:
- inset:
- outset

CSS AND FORMS

Label Tag s

- Label Tag is one of the easiest ways to make your web forms more usable.

- Labels make a form more accessible, & easier to use because of how most browsers implement them.

- When you assign a label to a form field **(using label for =),** it makes that entire label clickable. ☺

- When it's clicked on, the focus is moved to the form field that is labeled. ☺

Label Tags ..using For & Id attribute

Use the Label tag with the "for" attribute & id attribute

```
<label for="lastname"> Last Name </label>
        <input type="text" name="lastname" id="lastname">
```

NOTE – for attribute name must be the same as the id attribute name. It associates the label with the field. NNB…very important

label.html

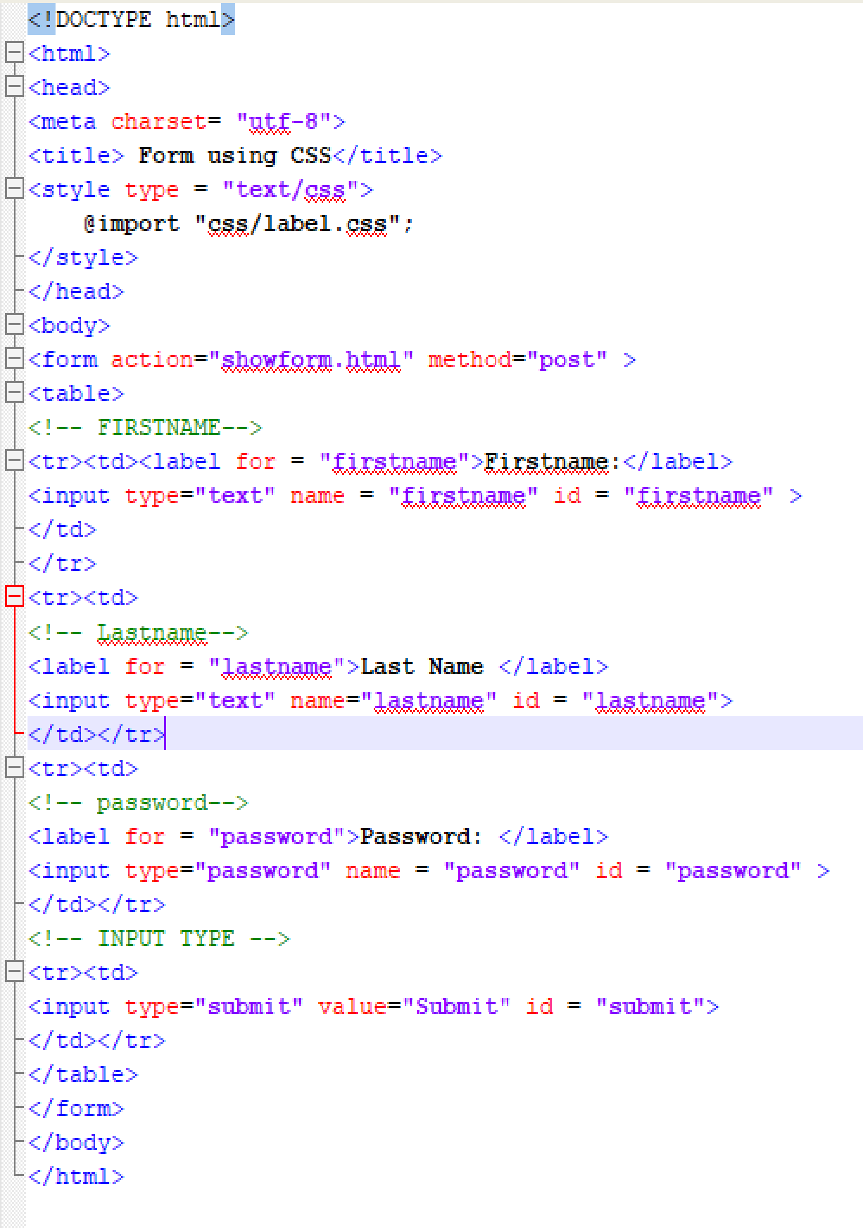
Label Tags ..without For

- The label tag can be used simply to place a label on a form.
- If the for attribute is not used, it is not associated with a form element.
- Label(s) may still be used, with appropriate CSS, for positioning a label on a form
- E.g
  - *<label> Email Address</label>*
  - *<label> Make of car </label>*

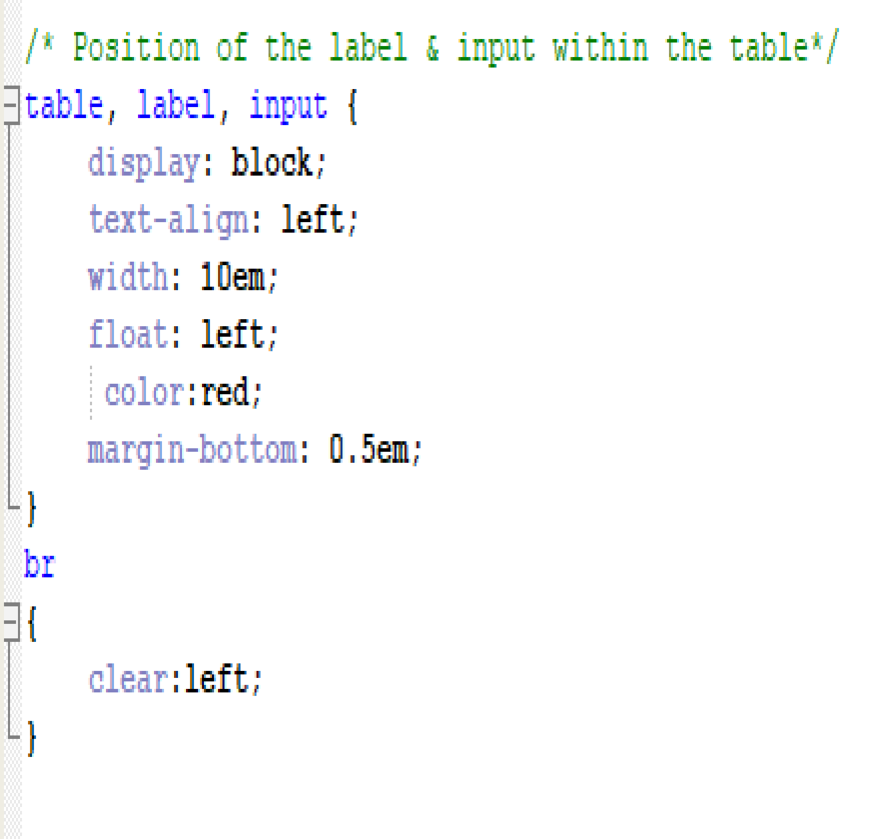
label.html

```
<form action="showform.html" method=“post" >
<label for = "firstname"></label>

      Firstname: <input type="text" name = "firstname" id = "firstname" >
<br><br>
<label for = "lastname"></label>

      Last Name <input type="text" name="lastname" id = "lastname” >
<br><br>
<label for = "password"></label>

      Password: <input type="password" name = "password" id = "password" >
<br><br>
      <input type="submit"  id = “submit” value="Submit" >
      <input type=“reset" >
</form>
```

label.html

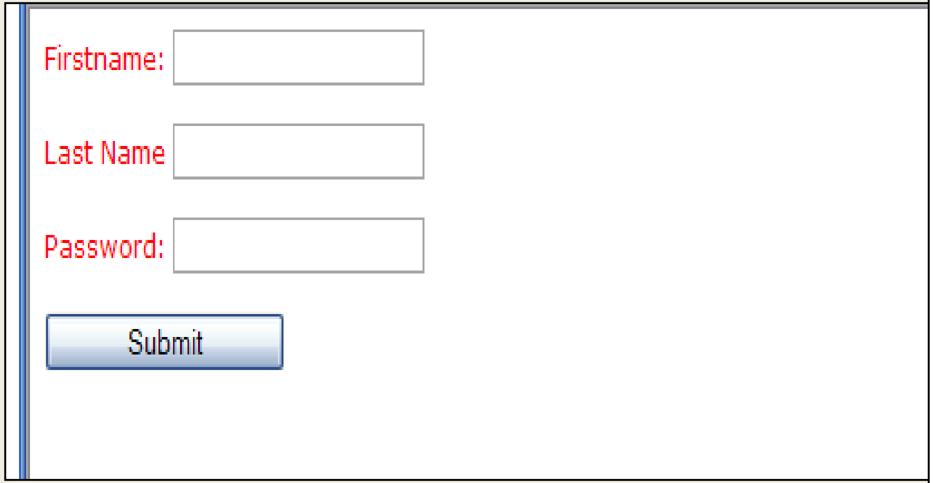


```
<!DOCTYPE html>
<html>
<head>
<meta charset= "utf-8">
<title> Form using CSS</title>
<style type = "text/css">
    @import "css/label.css";
</style>
</head>
<body>
<form action="showform.html" method="post" >
<table>
<!-- FIRSTNAME-->
<tr><td><label for = "firstname">Firstname:</label>
<input type="text" name = "firstname" id = "firstname" >
</td>
</tr>
<tr><td>
<!-- Lastname-->
<label for = "lastname">Last Name </label>
<input type="text" name="lastname" id = "lastname">
</td></tr>
<tr><td>
<!-- password-->
<label for = "password">Password: </label>
<input type="password" name = "password" id = "password" >
</td></tr>
<!-- INPUT TYPE -->
<tr><td>
<input type="submit" value="Submit" id = "submit">
</td></tr>
</table>
</form>
</body>
</html>
```



```
/* Position of the label & input within the table*/
table, label, input {
    display: block;
    text-align: left;
    width: 10em;
    float: left;
     color:red;
    margin-bottom: 0.5em;
}
br
{
    clear:left;
}
```

CSS styling for Label.html



- Label & text input field

display:block;

float: left;

width : x

margin: x

- label

Colour, text-align etc.

Laying out a Form using CSS

- Use the label element for layout
- Change the label element & input tags of your form from inline elements to **block elements**
- 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.
- Note the final <br> following the input fields.
- This is there, so that the rest of the page does not line up to the right of the final form element.

Styling for label.html

```
/* Style label */


label {
        text-align:         right;
        width:              5em;
        padding-right:      1em;
        font-size:          0.9em;
        color:              red;
      font-family:          Verdana,
       Arial, Helvetica, sans-serif;
       }
```

```
/* Position of Label & input */


label, input {
        display:    block;
        width:      10em;
        float:      left;
        margin-bottom: 0.5em;
}


br{
 clear:  left;
}
```

label.css

Sample styling for label.html

- /* Style the form */

```
form{
  width:     10em ;
  border:   1px solid red;
  background-color:   gray ;
  padding:    1em;
  etc.
   }
```

- /* Style the submit & button */

```
#submit{
  width:   1em ;
  border:  3px outset  red ;
  background-color:   blue ;
  padding:   1em ;
  etc.
   }
```

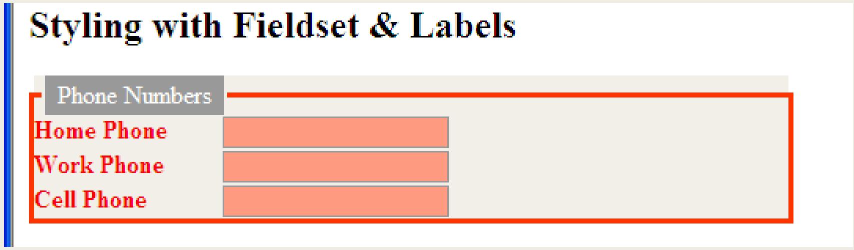
Fieldset & Legend…

- 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

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>
```

Styling Fieldset & Legends



Styling with Fieldset & Labels

Phone Numbers

Home Phone

Work Phone

Cell Phone

Fieldset

- Border, width, colour

Legend

- Colour, padding

Input & labels

- Float left

Display as block

Width --- xxx

Input

- color, width

Label

- color, width

br

- clear floats

Styling the Fieldset, Legend & input

```
    fieldset {
       border: 0.2em solid #fd3400;
       width: 30em;
       background-color: #f2efe8;
       clear:both;
     }


     legend {
       border: 0;
       background-color: #999999;
       color: #ffffff;
       padding: 0.2em 0.5em;
     }
```

fieldsetLegend.css

```
input {
  color: #000;
  background-color: #fe9a80;
  border: 0.1em solid #999999;
}

label{
                width: 120px;
                font-weight: bold;
                color: red;
}

label, input {
                float:left;
                display:block;
                }

br{
        clear:left;
     }
```

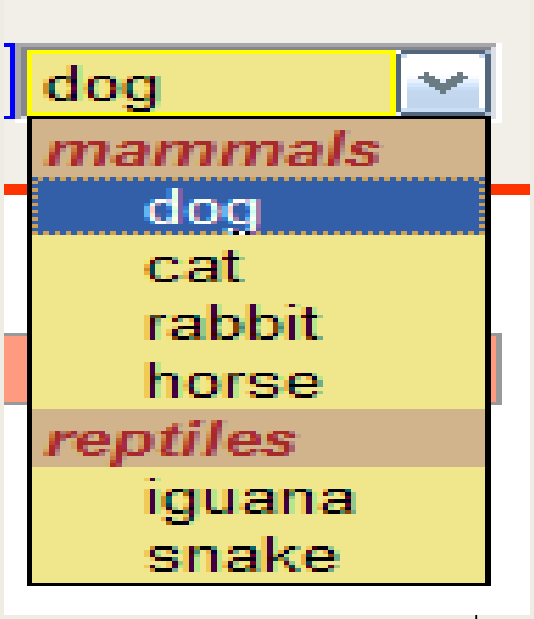
Optgroup Tag....

- ***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 color ....test in multiple browsers

Styled Optgroup….

Name your favorite pet

```
<label id = “pets” >Name your favorite pet </label>
<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>
```



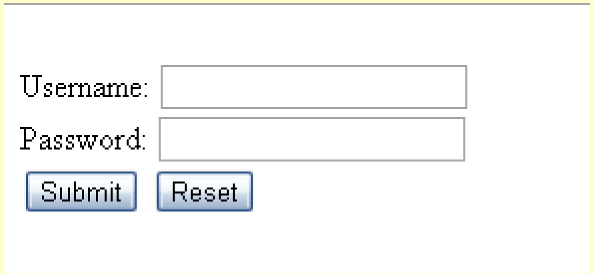
optGroup.html

Styled Optgroup

```
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

```
<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>
<div id = “button”>
            <input type="submit" id="submit" value="Submit" >
             <input type=“reset" id=“reset" >
</div>
<br>
</form>
```

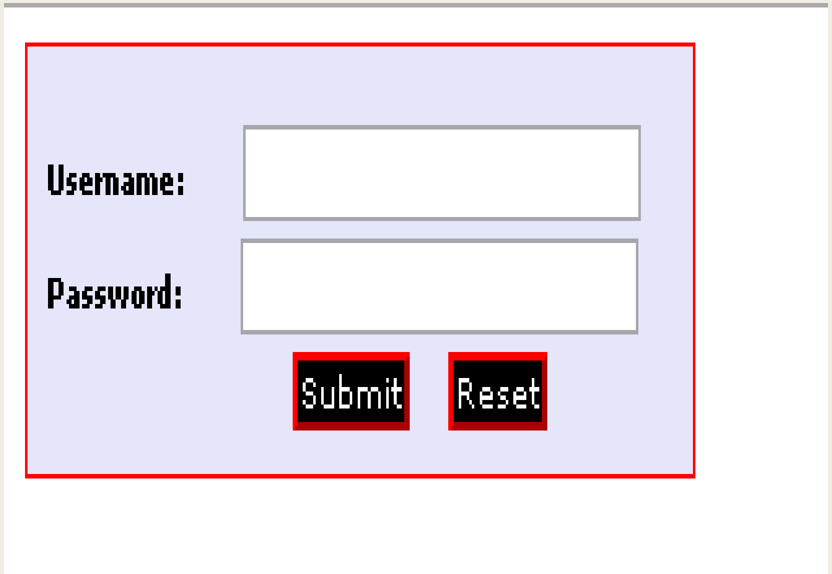


Login Form with Styles applied

```
form{
  width:            20em;
  font-family: Arial, Helvetica, sans-serif;
  font-size:        0.8em;
  border:                      1px solid red;
  background-color: lavender;
}


label{
  font-family: Verdana, Arial, Helvetica, sans-
      serif;
  font-size:        0.7em;
  width:            15em;
  margin:   0em 1em 0em 0em;
  font-weight:      bold;
  padding:          1em;
}
```

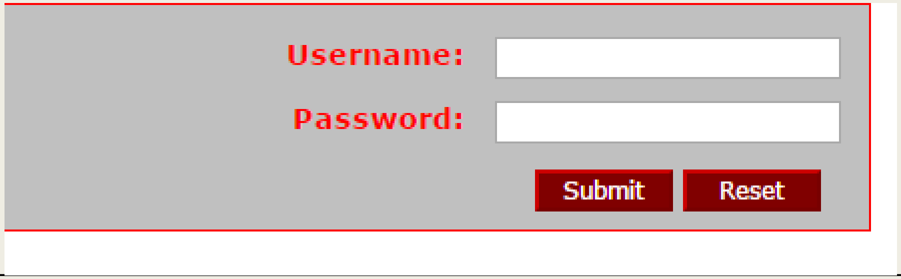
```
#submit,#reset {
  font-family:         Verdana, Arial,
Helvetica, sans-serif;
  font-size:           0.8em;
  margin:     0.2em 1em 0em 0em;
  padding:             0.1em;
  border:              2px outset red;
  background-color: black;
  color:               white;
  margin-bottom:   1em;
}


#buttons{
 margin-left:          8em;
              }
```

Login Form with Styles applied



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*



Positioning of Form Labels etc.

```
label, input {
        display:        block;
        float:          left;
}


br{
        clear:          left;
}
```

Style the form etc.

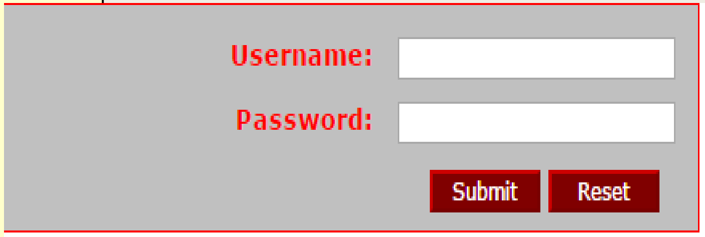
```
form{
  width:                       30em;
  padding:          0em;
  margin:           20em;
  border:           0.1em solid red;
  text-align:       right;
  font-family: Arial, Helvetica,
      sans-serif;
  font-size:        0.8em;
  background-color: silver;
}
```

```
.label {
  font-family: Verdana, Arial,
      Helvetica, sans-serif;
  font-size:    0.7em;
  width:        15em;
  margin:       0em 1em 0em
      0em;
  padding:      0em;
  color:        red;
  font-weight:          bold; }
```

Style Submit Button

Position the button using margin-left, add a width, and a border

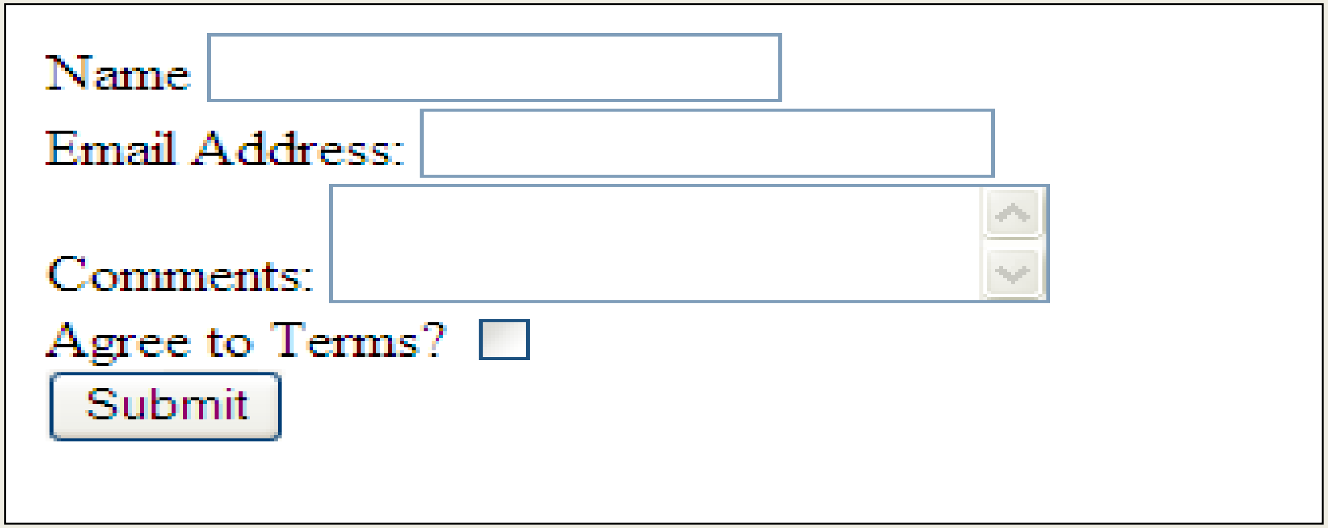
```
input[type = “submit”], input[type = "reset"]
 {
  font-family:        Verdana;
  font-size:          0.8em;
  padding:            0.1em;
  background-color:         maroon;
  color:              white;
  border:             2px  outset  #c00;
  width:              8em;
  margin-left:        23em;
}
```



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>
```

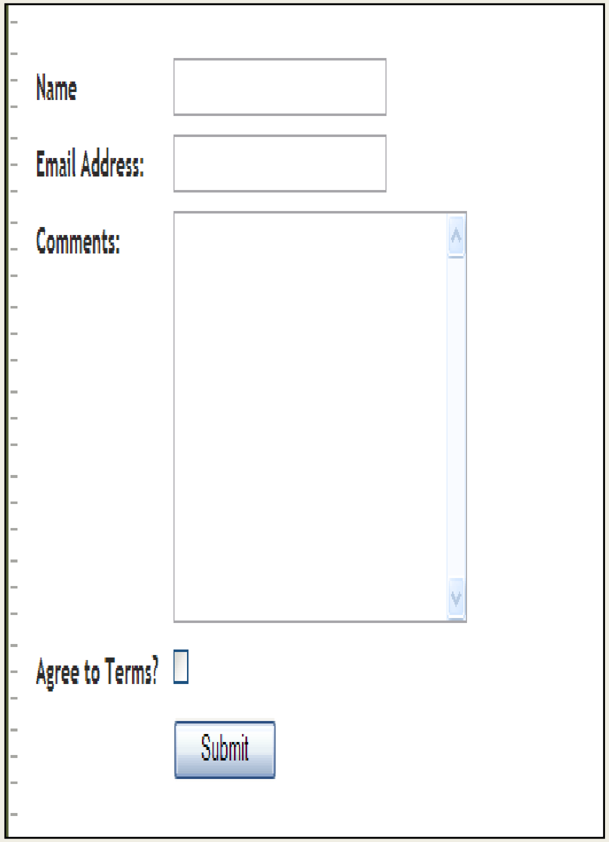
1. Form without any CSS



2. Styling the Form with CSS

```
label{
display:block;
float: left;
width: 120px;
font-weight: bold;
}

input, textarea{
width: 180px;
margin-bottom: 5px;
}

textarea{
width: 250px;
height: 150px;
}
```

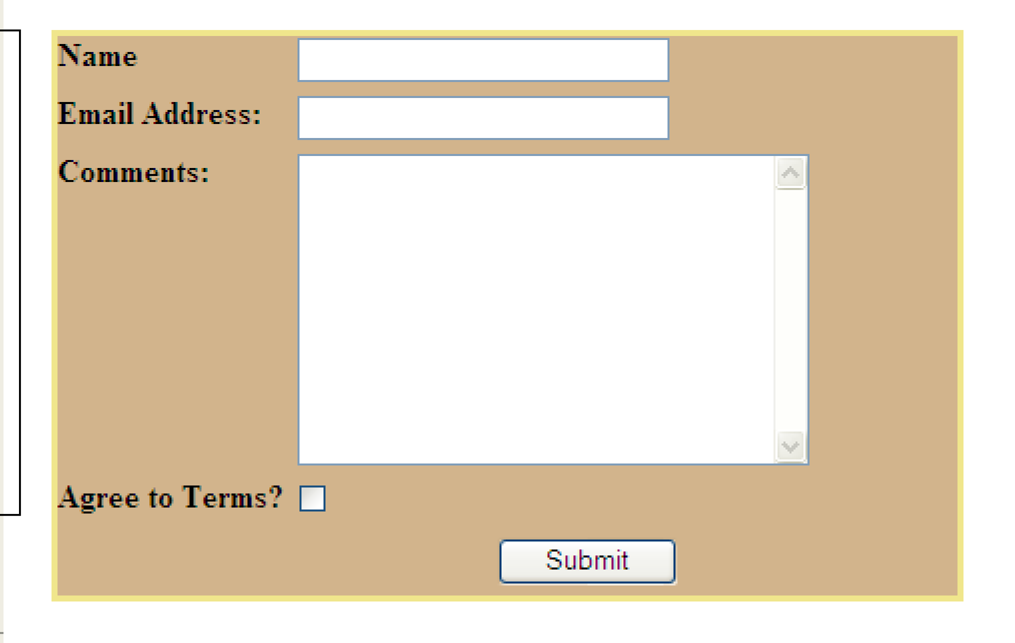
```
#terms{
width: 1em;
}

#submitbutton{
margin-left: 120px;
margin-top: 5px;
width: 90px;
}

br{
clear: left;
}
```



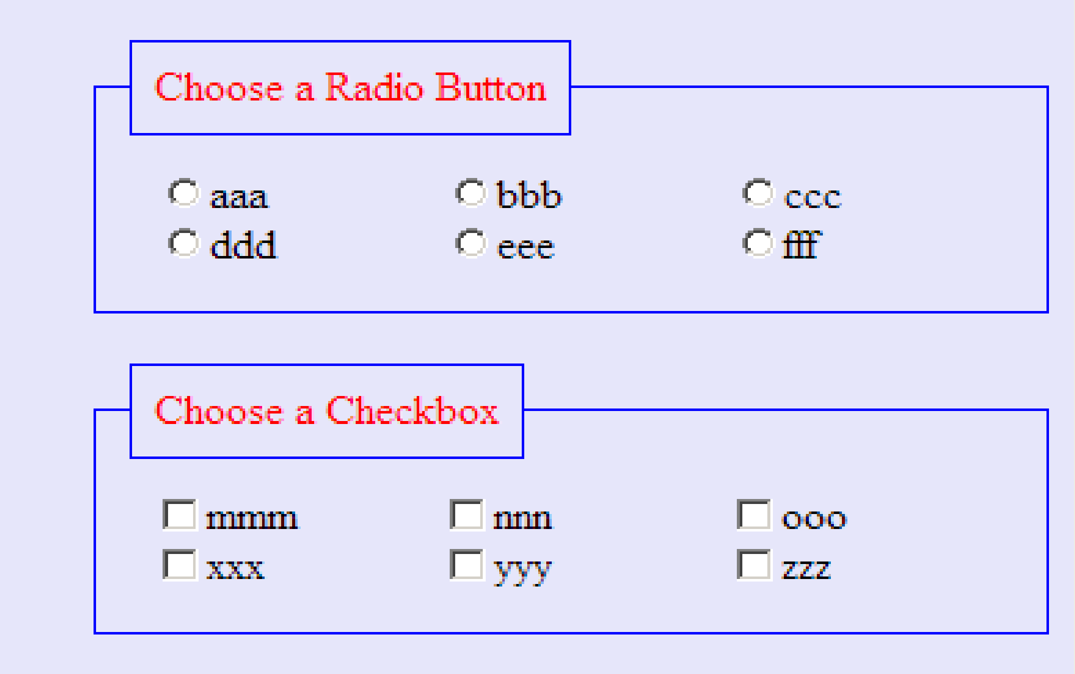
3. Form styled with CSS

- View Sample Code accompanying lecture

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

….
```



Positioning Radio & Checkboxes



Choose a Radio Button

○ aaa ○ bbb ○ ccc
○ ddd ○ eee ○ fff

Choose a Checkbox

□ mmm □ nnn □ ooo
□ xxx □ yyy □ zzz

Positioning Radio & Checkboxes

- Must put a **<label> … </label>** around each

radio button and each checkbox in order to position the radio button or checkbox.

- In the following example, each label belongs to (or is assigned to) a particular **class**.
- The class name is used in the CSS to position the radio button or checkbox.
- The properties: **margin** and **width** are used in the CSS

Positioning Radio Buttons

```
<legend> Choose a Radio Button</legend>
   <label class = "radio">
          <input type = "radio" name = "rb" value = "aaa"> aaa</label>
  <label class = "radio">
          <input type = "radio" name = "rb" value = "bbb"> bbb</label>
  <label class= "radio">
          <input type = "radio" name = "rb" value = "ccc">ccc   </label>
<br>
<label class = "radio">
          <input type = "radio" name = "rb" value = "ddd"> ddd          </label>
<label class = "radio" >
          <input type = "radio" name = "rb" value = "eee"> eee  </label>
<label class= "radio">
          <input type = "radio" name = "rb" value = "fff">fff     </label>
<br>
```

Positioning Checkboxes

```
<label class = "checkbox">Checkbox</label>
        <input type = "checkbox" name = "cb" value = “xxx">
<label class = "checkbox label>
        <input type = "checkbox" name = "cb" value = “yyy">>
     <label class = "checkbox label>
        <input type = "checkbox" name = "cb" value = zzz">
<label class = "checkbox label>
        <input type = "checkbox" name = "cb" value = “aaa“
<label class = "checkbox ></label>
        <input type = "checkbox" name = "cb" value = “bbb">
<label class = "checkbox> </label>
        <input type = "checkbox" name = "cb" value = “ccc">
```

Positioning Radio Buttons & Checkboxes

```
radio{
          margin-left: 0.5em;
          width:       5.5em;
          }
```

```
.checkboxes{
               margin-left:    0.5em;
               width:          5.5em;
               }
```

Alternatively, you could use:

input[type = “radio”]

input[type = “checkbox”]

to target the radio buttons & checkboxes

Style the Form

- Style the text input / textarea background when it gets focus

```
input:focus, textarea:focus {
    background-color: yellow;
}
```

Styling a Form

- Can use CSS with any of the following:
  - *Form*
  - *Label*
  - *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

Typical Steps for Styling a Form..

- 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 fieldsets and write legends
- style labels to ***float left, text to right, display block, and define width***
  - *may need small right margin, to separate labels from inputs*

Typical Steps for Styling a Form..

- By floating the "label" tag to the left, the text description of each form field appears to the left, resulting in a "two column" look for the form.
- Width of the "label" is controlled, so it will comfortably contain the longest text description in the form.
- Use <*br*> tag with "*clear: left*" after each form field. This prevents the floating <label> tag from potentially spilling over to content beneath it.
- Use margins, such as "margin-top" and "margin-bottom" to add some nice margins between form fields.
- Put a <*div*> around the buttons & set a *margin-left.*
- Style the buttons like the form – width, border, background-colour & font color. Add padding & margins, as appropriate.

Typical Steps for Styling a Form..

- Style input and textarea focus, background colour, text colour (as appropriate)

```
input:focus, textarea:focus {
    background-color: yellow;
}
```

(In Chrome, textboxes may hop around the place – bug!)

Recap – Forms

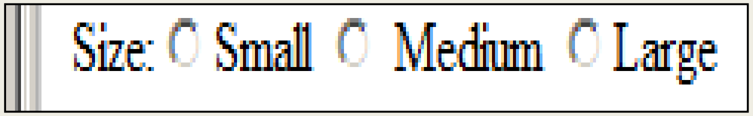
- Have styled forms based on a 2-column style layout in previous lecture
- Labels were placed on the left-hand side of the forms
- Textboxes e.g. text, password, textarea elements were placed on the right-hand side of the labels and all were aligned
- Buttons were centred in the form, as appropriate.

Forms – this lecture

- Looking at radio buttons & checkboxes
  - *Aligning them horizontally*
  - *Aligning them vertically*
  - *Having 2 rows of checkboxes*
- Styling fieldset & legends
- No hard and fast rules for layout forms!!
- Many approaches – this is one that works (in Chrome)
- Using px as a measure, in examples. Ideally should be using rems – Why?

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: ○ Small ○ Medium ○ Large

Size:

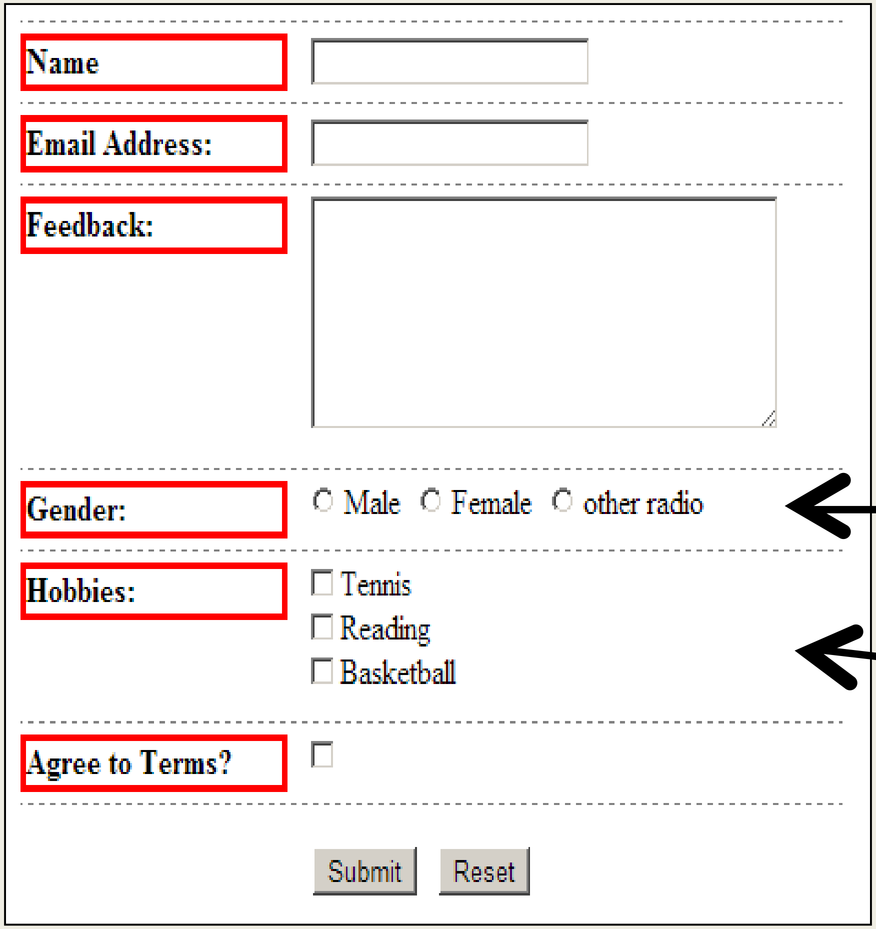
```
<input type="radio" name="size" value="small">Small
 <input type="radio" name="size" value="medium"> Medium
 <input type="radio" name="size"  value="large">Large
 <br>
```

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



Name	
Email Address:	
Feedback:	
Gender:	○ Male ○ Female ○ other radio
Hobbies:	□ Tennis □ Reading □ Basketball
Agree to Terms?	□

Radio Buttons - horizontal

Checkboxes - vertical

choices.html

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

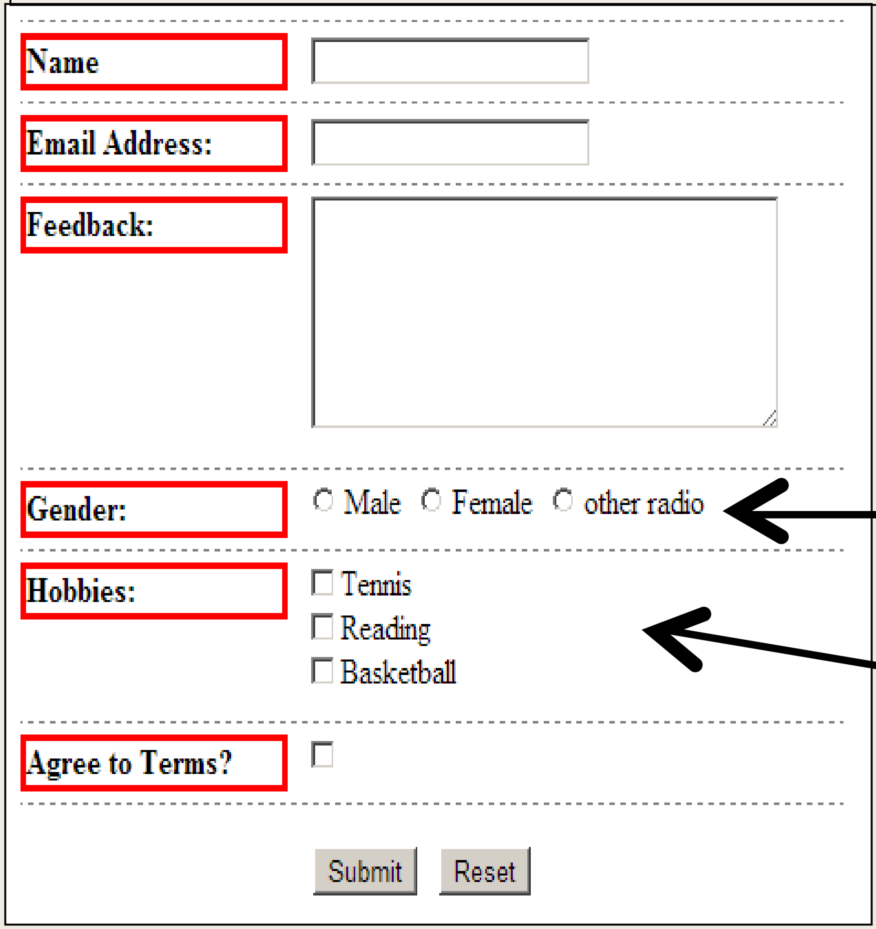
choices.html

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

choices.html

Form with Radio Buttons & Checkboxes



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

Radio Buttons - horizontal

Checkboxes - vertical

choices.html

choices.html

Labels and input tags done, as in previous lecture

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

Radio Buttons

- **<span>** class used around all the radio buttons in order to display the text beside the radio button

Checkboxes

- **Blank Label <label class = "checkboxfix"></label>** put before 2nd and 3rd checkbox to align them with 1st checkbox

1. choices.html

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

choices.html

```
<p>
<label for="email">Email Address:</label>
        <input type=“email"  name ="email" id = "email" >
</p><p>
<label for="comments">Feedback:</label>
        <textarea  name = "comments" id = "comments"></textarea>
</p><p>
<label >Gender:</label>
        <input type="radio" name=“gender" value = "male" >
                <span class = "radio"> Male </span>
        ……………………</p>
```

choices.html

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

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

<label class = “checkboxfix”></label>

  <input type="checkbox" name="reading"  id = "reading" value =
"reading“ > Reading<br>

<label class = “checkboxfix”></label>

 <input type="checkbox" name="basketball"  id = "basketball" value =
"basketball“  > Basketball<br>.......................

<div id = "buttons">

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

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

</div>
```

choices.css

- Using paragraphs <p> tags to separate the various labels. (Previous programs used <br> tag)
- <form> tag: width
- <p> tag:
  - *Border-top (1px dashed gray)*
  - *clear any items (labels) that have floated to the left*
- Labels

  – float left **(remember to clear floats – done with <p> in example)**

  - displayed as a block

  - styled (red text etc.)

choices.html, .css

- input tags, textarea move in, using margin-left
- textarea is given a width and height in CSS, rather than rows & cols in HTML document
- <span> class for radio button text is floated to left
- "checkboxfix" creates a dummy (hidden) label before the 2nd & 3rd checkboxes
- #buttons puts a left-margin around the buttons
- Add any margins, padding, styling as appropriate

choices.css

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

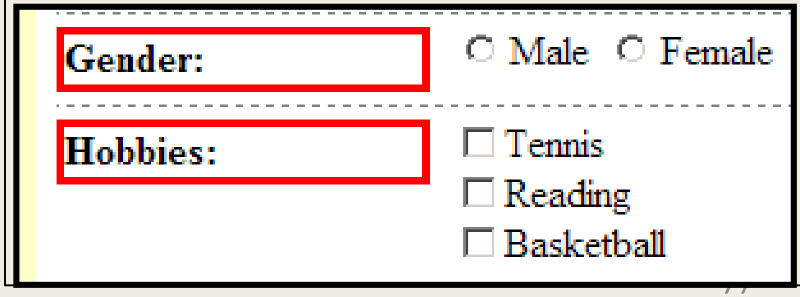
- Align all the input elements i.e using input tag in css
- i.e. <input type = "   "> text, password, email, radio, checkbox

```
/* move ALL input fields in  */
input{
        margin-left: 10px;
        }
```

choices.css

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

.radio {

        float:left;

}

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

.checkboxfix{

        width:140px;

        display:block;

        float:left:

        border:1px solid white;

}
```

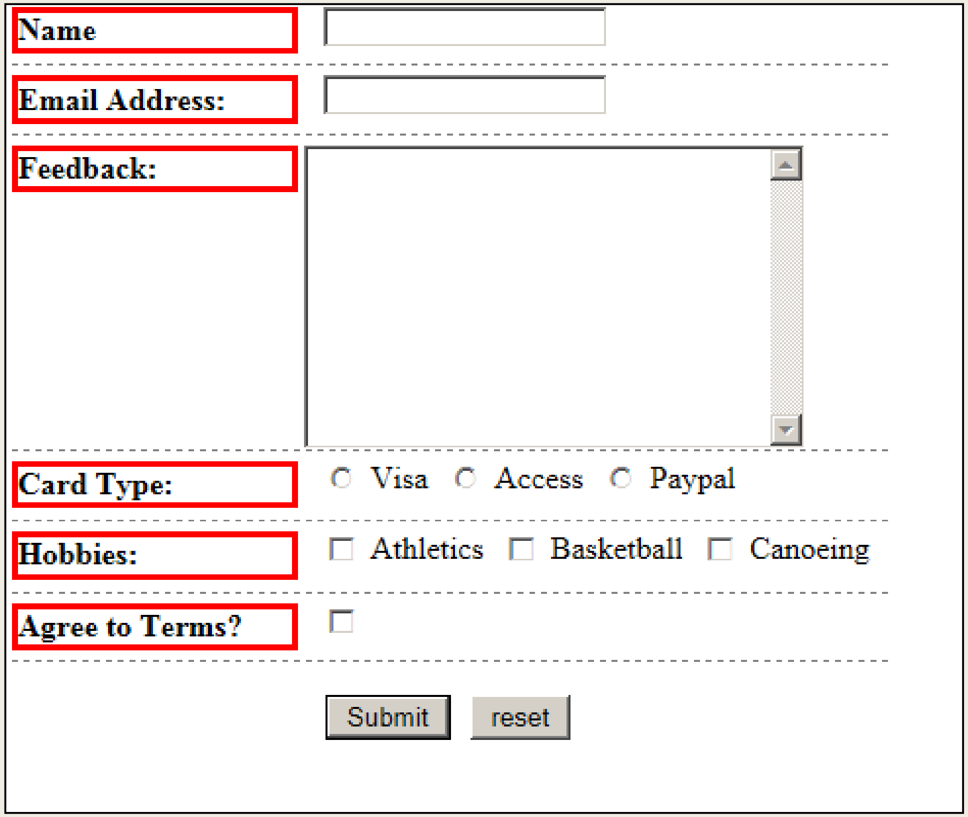
```
/* align buttons */

#buttons{

        margin-left: 150px;

}

/* clear floats*/

br{

        clear:left;

}
```



Gender: ○ Male ○ Female

Hobbies: □ Tennis
□ Reading
□ Basketball

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?	□

Aligning more radio buttons horizontally - html

- Going to put in three radio buttons and align them horizontally
- <span> 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>
```

Aligning more radio buttons horizontally - css

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

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


.radio{
        float:left;
}
```

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>
```

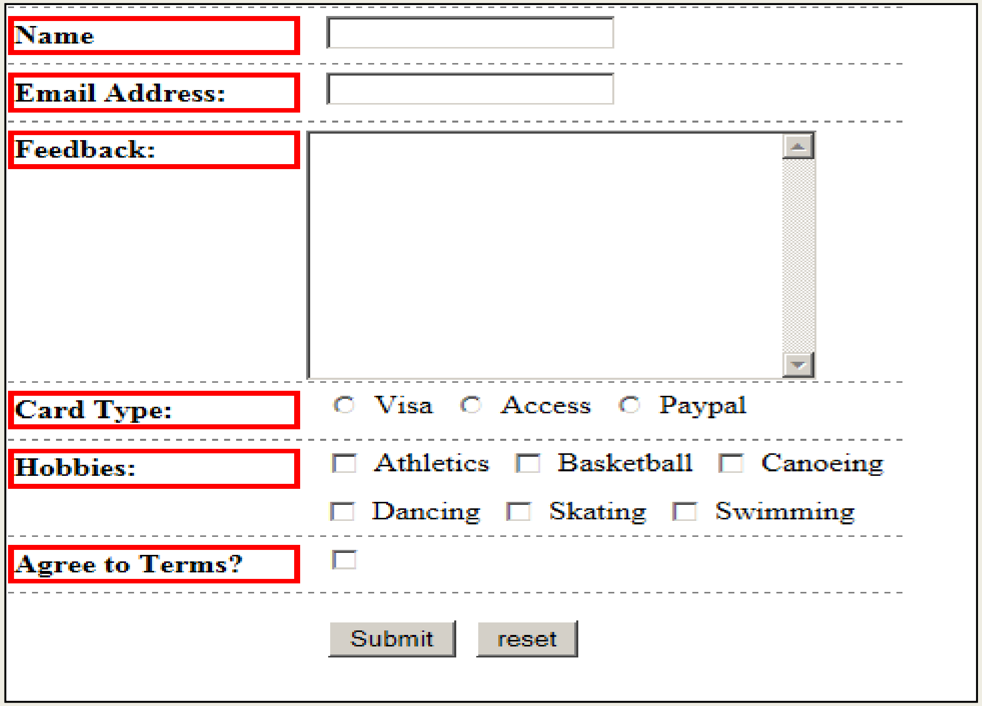
Aligning Checkboxes Horizontally

- Used **span class** called box to float checkboxes

  to left

```
/* used to line up  checkboxes  */
 .box{
    float: left;
}
```

Form with two rows of horizontal checkboxes



Name	
Email Address:	
Feedback:	
Card Type:	○ Visa ○ Access ○ Paypal
Hobbies:	□ Athletics □ Basketball □ Canoeing □ Dancing □ Skating □ Swimming
Agree to Terms?	□

formboxes_Horizontal.html

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>

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>
              ut 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>
```

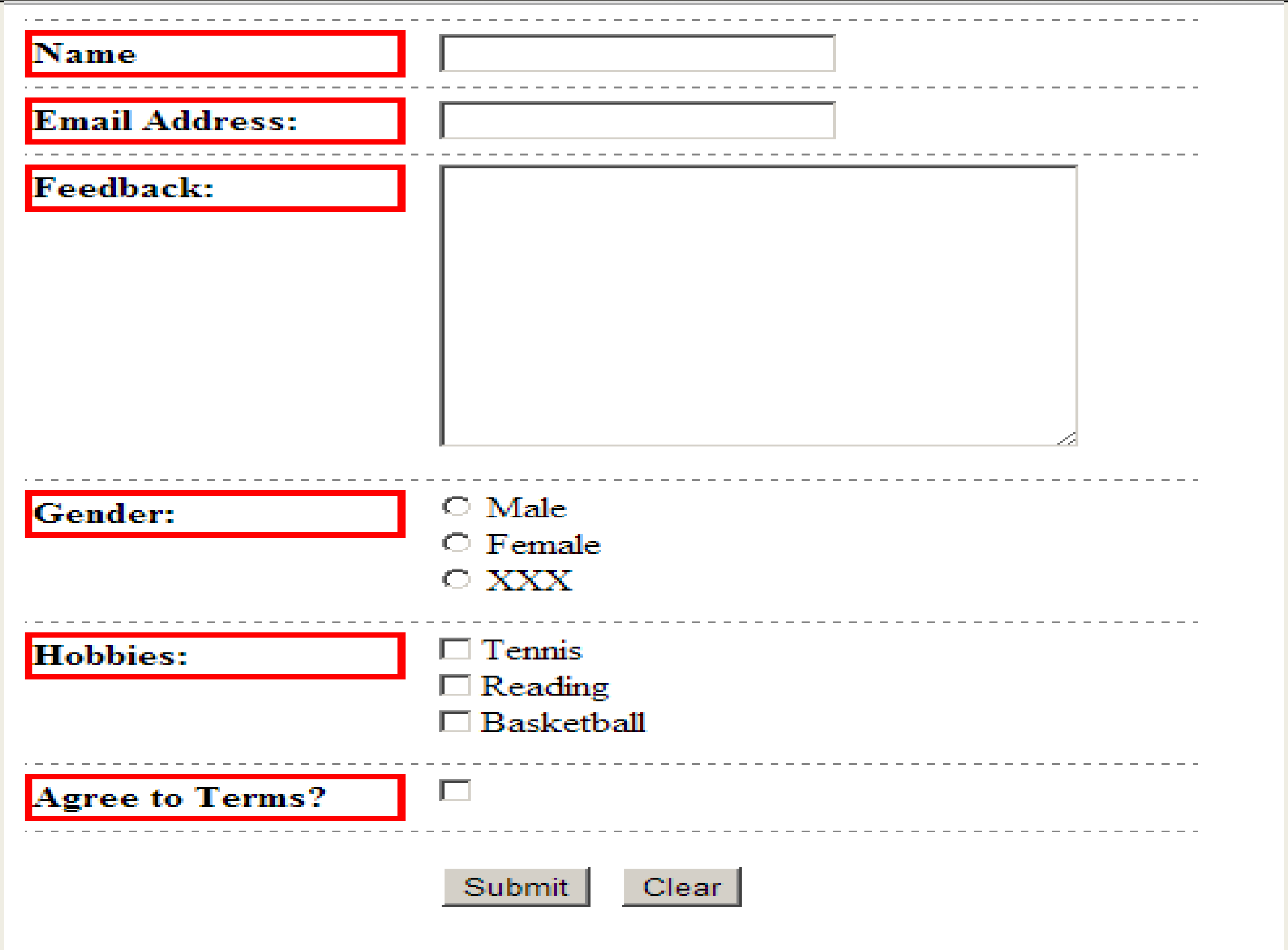
Blank label

Aligning 2 rows of checkboxes horizontally

- styling for id = “row2”
- Note – display, float, margin, clear

```
#row2{
        clear: left;
        float: left;
        display: block;
        margin-bottom: 5px;
            margin-right: 3px;
        border:1px solid white;
        }
```

Aligning radio buttons & checkboxes vertically



Name	
Email Address:	
Feedback:	
Gender:	○ Male ○ Female ○ XXX
Hobbies:	□ Tennis □ Reading □ Basketball
Agree to Terms?	□

Aligning Labels and Input boxes

```
label, input{
        float:left;
        display:block;
        }
/* style label */
label{
        font-weight: bold;
        border: 1px solid red;
        width: 140px;
        margin-right:3px;
}
```

Aligning radio buttons vertically

- Code added for a 3rd radio button below
- <br> added after each radio button
- Blank label in class “radiofix” created to align the 2nd, 3rd radio button with first radio button

```
<label>Gender:</label>
  <input type="radio" name=“gender" value = "male" > Male
   <br>
<label class = “radiofix”></label>
  <input type="radio" name=" gender " value = "female” > Female        <br>
<label class = “radiofix”></label>
<input type="radio" name=" gender " value = "xxx" > XXX
</p>
```

Aligning radio buttons vertically

```
/* 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;
}
/* .radiofix will take all the properties associated with a label except it will
have a white border */  (see checkbox fix later)
```

Align Checkboxes vertically

- Use a dummy label class called **“checkboxfix”** to align the 2nd & 3rd etc. checkboxes under the first checkbox.
- Doing the same CSS with the class called radiofix
- Clear the floats after the <br> tags in the CSS

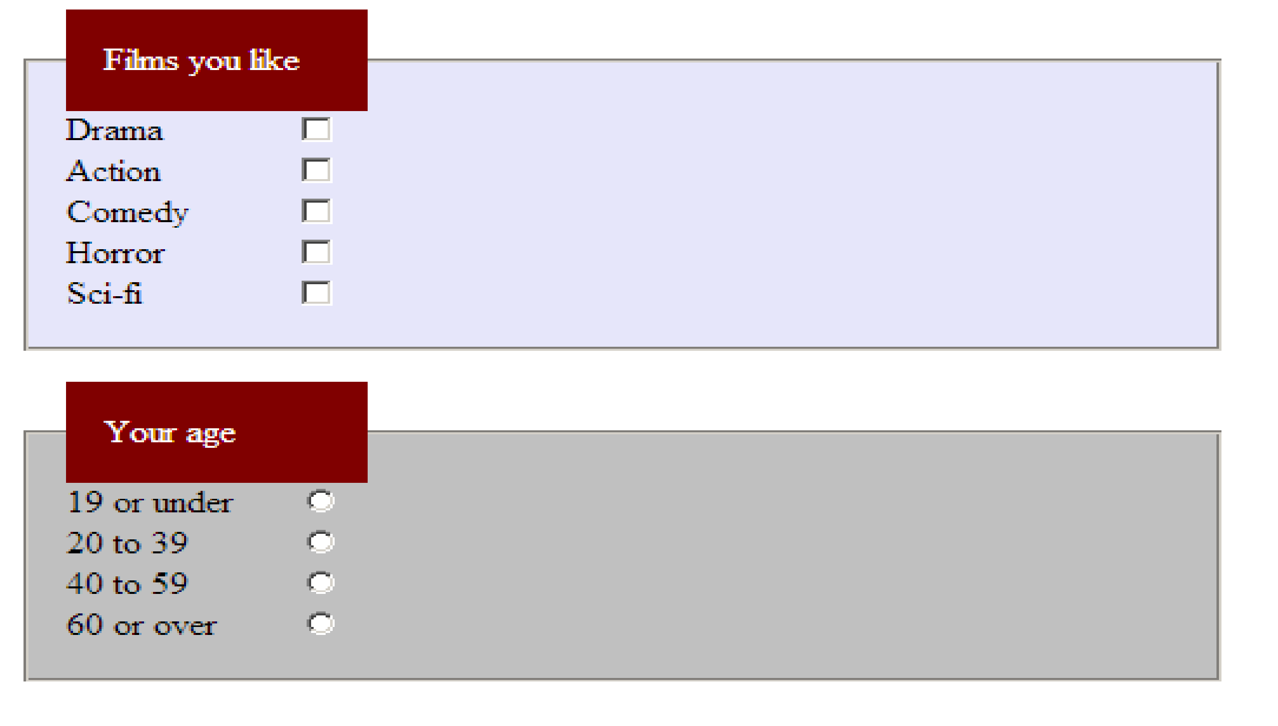
```
<input type="checkbox" name="tennis" value  = "tennis"
        id = "tennis> Tennis<br>

<label class = “checkboxfix”></label>

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

<label class = “checkboxfix”></label>

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

4. Align Checkboxes vertically

```
/* Position the Radio buttons and Checkboxes
vertically  - using all the properties for a blank label
*/

.radiofix, .checkboxfix{

        display:block;

        width:140px;

        float:left:

        border:1px solid white;

        margin-left: 3px;

}
```

```
Br
{
        clear:both;
}
```

Styling Fieldsets



Films you like

Drama □
Action □
Comedy □
Horror □
Sci-fi □

Your age

19 or under ○
20 to 39 ○
40 to 59 ○
60 or over ○

Styling Fieldsets

- Example – have 2 fieldsets and want to give each a different background color
- Assign a unique id to each fieldset and use this a different background colour.

```
 #fieldset1{
     background-color: lavender;
     }
#fieldset2{
     background-color: silver;
     }
```

Styling fieldsets

```
<fieldset id = "fieldset1">
<legend>Films you like</legend>
        <label>Drama</label>
                <input type="checkbox" .........
............
</fieldset>
<p></p>
<fieldset id = "fieldset2">
        <legend>Your age</legend>................
</fieldset>
```

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"]*

CSS & Forms

- References:
- http://webdesign.about.com/od/forms/a/aa050707.htm
- http://www.cssdrive.com/index.php/examples/exampleitem/tableless_forms/

THIS WEEKS LAB