

HTML TABLES AND FORMS

CLASS LOGISTICS

- Select your **own** project
 - Submission on: **13-December-2019**
-
- Project **must** have:
 - Good User Interface / Presentation
 - Client-side Validation
 - Server-side Validation

Ask for Help

- Harshani Nagahamulla –
`harshani.nagahamulla@ucdconnect.ie`
- Chen Jingrui – `jingrui.chen@ucdconnect.ie`
- Liu Dairui - `dairui.liu@ucdconnect.ie`
- Wang Xihao - `WangxhWo@163.com`

HTML Tables

A grid of cells

- A **table** in HTML is created using the **<table>** element
- Tables can be used to display:
- Many types of content
 - Calendars, financial data, lists, etc...
- Any type of data
 - Images
 - Text
 - Links
 - Other tables




HTML Tables

Chapter 4

	Free	Basic	Premium
Upload Space	50MB	200MB	Unlimited
Daily Uploads	1	10	Unlimited
Total Uploads	20	100	Unlimited
Social Sharing		✓	✓
Analytics			✓
Price per year	Free	\$ 9.99	\$ 19.99

Chapter 4

Artist Inventory











Artist	Work Details			
	Title	Year	Home	
 Jacques-Louis David		The Death of Marat	1793	Royal Museums of Fine Arts of Belgium
		The Intervention of the Sabine Women	1793	Royal Museums of Fine Arts of Belgium

Chapter 4

October 2014

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	
« Sep						Nov »

Paintings

	Title	Artist	Year	Genre	
	Death of Marat	David, Jacques-Louis	1793	Romanticism	
	Lictors Bearing to Brutus the Bodies of his Sons	David, Jacques-Louis	1789	Romanticism	
	Liberty Leading the People	Delacroix, Eugene	1830	Romanticism	
	Arrangement in Grey and Black	Whistler, James Abbott	1871	Realism	
	Mademoiselle Caroline Riviere	Ingres, Jean-Auguste	1806	Neo-Classicism	

Tables Basics

Rows and cells

- An HTML **<table>** contains any number of rows (**<tr>**)
- A table header is defined with the **<th>** tag
- each row contains any number of table data cells (**<td>**)
- Content goes inside of **<td></td>** tags
- **<table>**

<tr>

<th> Name of Book </th>

</tr>

<tr>

<td> Alice in Wonderland </td>

</tr>

</table>



content

A Simple Example

```
<table border="1" width="100%">
  <tr>
    <td>Row 1, cell 1</td>
    <td>Row 1, cell 2</td>
    <td>Row 1, cell 3</td>
  </tr>
  <tr>
    <td>Row 2, cell 1</td>
    <td>Row 2, cell 2</td>
    <td>Row 2, cell 3</td>
  </tr>
  <tr>
    <td>Row 3, cell 1</td>
    <td>Row 3, cell 2</td>
    <td>Row 3, cell 3</td>
  </tr>
</table>
```

With Table Headings

th

```
<table border="1" width="100%">
  <tr>
    <th>Column 1</th>
    <th>Column 2</th>
    <th>Column 3</th>
  </tr>
  <tr>
    <td>Row 1, cell 1</td>
    <td>Row 1, cell 2</td>
    <td>Row 1, cell 3</td>
  </tr>
  <tr>
    <td>Row 2, cell 1</td>
    <td>Row 2, cell 2</td>
    <td>Row 2, cell 3</td>
  </tr>
  <tr>
    <td>Row 3, cell 1</td>
    <td>Row 3, cell 2</td>
    <td>Row 3, cell 3</td>
  </tr>
</table>
```


Cells that span many columns

```
<html>
<body>
  <table>
    <tr>
      <th>Name</th>
      <th colspan="2">Phone Number</th>
    </tr>
    <tr>
      <td>Bill Gates</td>
      <td>55575854</td>
      <td>55575855</td>
    </tr>
  </table>
</body>
</html>
```

Name	Telephone	
Bill Gates	55577854	55577855

Cells that span many rows

```
<table>
  <tr>
    <th>Name:</th>
    <td>Bill Gates</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td>55577854</td>
  </tr>
  <tr>
    <td>55577855</td>
  </tr>
</table>
```

Name:	Bill Gates
Telephone:	55577854
	55577855

Semantic structuring

```
<table border="1" width="100%">
  <thead>
    <tr>
      <th>Fruits</th>
      <th>Vitamin A</th>
      <th>Vitamin C</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>Apples</th>
      <td>98 ui</td>
      <td>8.4 mg</td>
    </tr>
    <tr>
      <th>Oranges</th>
      <td>295 ui</td>
      <td>69.7 mg</td>
    </tr>
  </tbody>
</table>
```

Changing Column Width

```
<table border="1" width="100%">
  <col style="width:40%">
  <col style="width:30%">
  <thead>
  <tr>
    <th>Fruits</th>
    <th>Vitamin A</th>
    <th>Vitamin C</th>
  </tr>
  </thead>
  <tbody>
  <tr>
    <th>Apples</th>
    <td>98 ui</td>
    <td>8.4 mg</td>
  </tr>
  <tr>
    <th>Oranges</th>
    <td>295 ui</td>
    <td>69.7 mg</td>
  </tr>
  </tbody>
</table>
```

What happens when col style width is more than 100%?

Summary of Tables

- Every table starts and end with the `<table>` element
- You use the attributes `colspan` and `rowspan` to merge cells
- A table is divided into a head (`<thead>` element) and a body (the `<tbody>` element)
- Inside the head and body the table is divided into rows using the `<tr>` element
- No columns in HTML tables; Use `<td>` to define individual data cells
- Whenever you need to define a heading cell, use the `<th>` element instead of the `<td>` element

INTRODUCING FORMS

HTML Forms

- Makes an HTML page interactive
- Consists of two parts:
 - Front-end → HTML Forms
 - Back-end → PHP/Perl/Python/Ruby/Java/some-other-language
- How forms work:
 - Browser displays form to user
 - User fills out form and clicks `submit`
 - Browser sends data to the server
 - Server processes and sends data back

HTML Forms

Richer way to interact with server

- **Forms** provide the user with an alternative way to interact with a web server.
 - Forms provide rich mechanisms like:
 - Text input
 - Password input
 - Options Lists
 - Radio and check boxes
 - Date fields

Simple HTML Form

Richer way to interact with server

```
<!DOCTYPE HTML>
<html lang="en">

<body>
<form action="/action_page.py">
  First Name: <br/>
  <input type="text" name="firstname" value="Mickey"> <br/>
  Last Name: <br/>
  <input type="text" name="lastname" value="Mouse"> <br/>
  <br> <br>
  <input type="submit" value="Press Submit">
</form>

</body>
</html>
```

First Name:

Last Name:

FORMS CONTROL ELEMENTS

Input Type Examples

Type	Description
<code><input type="text"></code>	Defines a one-line text input field
<code><input type="radio"></code>	Defines a radio button (for selecting one of many choices)
<code><input type="submit"></code>	Defines a submit button (for submitting the form)

Input Type - Fieldset

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personal information:</legend>
    First name:<br>
    <input type="text" name="firstname" value="Mickey">
    <br>
    Last name:<br>
    <input type="text" name="lastname" value="Mouse">
    <br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

Personal information:

First name:

Last name:

Input Type - Text

- Default type for any input, if you haven't specified anything
- Size of input allowed:
 - Default – 20 characters
 - Change using size and maxlength

Favourite song:

```
<input size="42" maxlength="100"><br/>
```

Input Type - Textarea

- Sometimes you need to enter multiple lines of input (e.g., comment on an article or blogpost)
- Size can be changed by:
 - `rows`
 - `cols`

Favourite quote:

```
<textarea cols="27" rows="4">One man's variable is another man's constant</textarea>
```

Input Type - Radio

```
<form>  
  <input type="radio" name="gender" value="male" checked> Male<br>  
  <input type="radio" name="gender" value="female"> Female<br>  
  <input type="radio" name="gender" value="other"> Other  
</form>
```

- ☒ Male
- ☐ Female
- ☐ Other

Input Type - Option

```
<form action="/action_page.php">
  <select name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <br><br>
  <input type="submit">
</form>
```

Volvo ▼

Volvo

Saab

Fiat

Audi

Input Type - checkbox

- Used when you want to give your users the ability to choose zero/one/several options
- The `value` is sent when the user submits the form
- Set the `checked` property, if you want to make a default selection

```
<fieldset>
  <legend>What are your favourite colours?</legend>
  <input type="checkbox" name="favorite_colour" value="Yellow">Yellow<br>
  <input type="checkbox" name="favorite_colour" value="Purple" checked>Purple<br>
  <input type="checkbox" name="favorite_colour" value="Blue">Blue<br>
  <br>
</fieldset>
```

Input Type – Radio Button

- Used whenever you need to give the user a choice
- Like a checkbox, but forces a single choice

```
<fieldset>
```

```
    <legend>What is Your Favorite Pet?</legend>
```

```
    <input type="radio" name="favorite_pet" value="Cats">Cats<br>
```

```
    <input type="radio" name="favorite_pet" value="Dogs">Dogs<br>
```

```
    <input type="radio" name="favorite_pet" value="Birds">Birds<br>
```

```
    <br>
```

```
</fieldset>
```

Radio Button - Checked

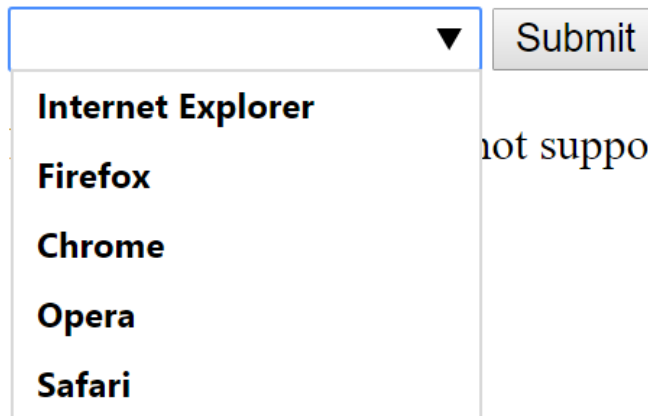
- Sometimes you want to ensure that at least one option is selected
- Use the checked property to make sure that there is a default selection

```
<fieldset>
  <legend>What is Your Favorite Pet?</legend>
  <input type="radio" name="favorite_pet" value="Cats">Cats<br>
  <input type="radio" name="favorite_pet" value="Dogs" checked>Dogs<br>
  <input type="radio" name="favorite_pet" value="Birds">Birds<br>
  <br>
</fieldset>
```

Input Type - Datalist

```
<form action="/action_page.php">
  <input list="browsers" name="browser">
    <datalist id="browsers">
      <option value="Internet Explorer">
      <option value="Firefox">
      <option value="Chrome">
      <option value="Opera">
      <option value="Safari">
    </datalist>
  <input type="submit">
</form>
```

Note: The datalist tag is not supported in Safari or IE9 (and earlier).



A screenshot of a web form. On the left, there is a dropdown menu with a downward arrow icon. The menu is open, showing a list of browser names: Internet Explorer, Firefox, Chrome, Opera, and Safari. To the right of the dropdown menu is a button labeled "Submit".

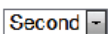
Select Lists

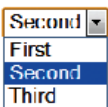
Chose an option, any option.

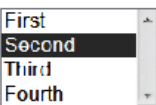
- **<select>** element is used to create a multiline box for selecting one or more items
 - The options are defined using the **<option>** element
 - can be hidden in a dropdown or multiple rows of the list can be visible
 - Option items can be grouped together via the **<optgroup>** element.

Select Lists

- Select List Examples

Select: 

Select: 

Select: 

Cities: 

```
<select name="choices">
  <option>First</option>
  <option selected>Second</option>
  <option>Third</option>
</select>
```


```
<select size="3" ... >
```

```
<select ... >
  <optgroup label="North America">
    <option>Calgary</option>
    <option>Los Angeles</option>
  </optgroup>
  <optgroup label="Europe">
    <option>London</option>
    <option>Paris</option>
    <option>Prague</option>
  </optgroup>
</select>
```

Which Value to send

Select Lists Cont.

- The **value** attribute of the `<option>` element is used to specify what value will be sent back to the server.
- The value attribute is optional; if it is not specified, then the text within the container is sent instead

Select: 
First
Second
Third

```
<select name="choices">  
  <option>First</option>  
  <option>Second</option>  
  <option>Third</option>  
</select>
```

?choices=Second

```
<select name="choices">  
  <option value="1">First</option>  
  <option value="2">Second</option>  
  <option value="3">Third</option>  
</select>
```

?choices=2

Checkboxes

I accept the software license ☒

```
<label>I accept the software license</label>  
<input type="checkbox" name="accept" >
```

Where would you like to visit?

- ☒ Canada
- ☐ France
- ☒ Germany

```
<label>Where would you like to visit? </label><br/>  
<input type="checkbox" name="visit" value="canada">Canada<br/>  
<input type="checkbox" name="visit" value="france">France<br/>  
<input type="checkbox" name="visit" value="germany">Germany
```

?accept=on&visit=canada&visit=germany



Specialized Controls

- I'm so special
 - **<input type=hidden>**
 - **<input type=file>**

Upload a travel photo
 No file chosen



Upload a travel photo
 IMG_0020.JPG

```
<form method="post" enctype="multipart/form-data" ... >
  ...
  <label>Upload a travel photo</label>
  <input type="file" name="photo" />
  ...
</form>
```

File Example

- Note the `method`
- Note the `enctype`
- These are essential, if you want to upload a file

```
<html>
<body>
<form action="process.php" enctype="multipart/form-data" method="POST">
    <label>Upload your photo:</label>
    <input type="file" name="userphoto" />
</form>
</body>
</html>
```

Number and Range

Rate this photo:

```
<label>Rate this photo: <br/>  
<input type="number" min="1" max="5" name="rate" />
```

Grumpy  Ecstatic

```
Grumpy  
<input type="range" min="0" max="10" step="1" name="happiness" />  
Ecstatic
```

Rate this photo:

Grumpy Ecstatic

Controls as they appear in browser
that doesn't support these input types

Example of Number and Range

```
<fieldset>
  <legend>Your opinion about the photo</legend>
  <label for="rating">Rate this photo</label>
  <input type="number" min="1" max="10" name="rating"><br/>
</fieldset>
```

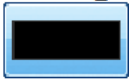
Grumpy

```
<input type="range" min="0" max="5" step="1" name="expression" />
```

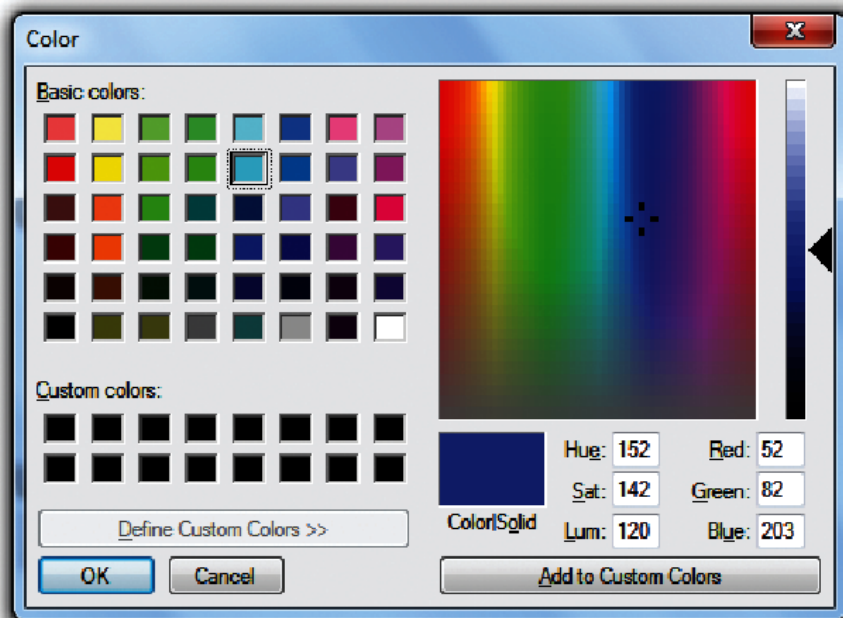
Happy

Color

Background Color:



```
<label>Background Color: <br/>
<input type="color" name="back" />
```



Background Color:

Control as it appears in browser that
doesn't support this input type

Date and Time Controls

Dates and times often need validation when gathering this information from a regular text input control.

From a user's perspective, entering dates can be tricky as well: you probably have wondered at some point in time when entering a date into a web form, what format to enter it in, whether the day comes before the month, whether the month should be entered as an abbreviation or a number, and so on.

HTML5 Date and Time Controls

Date:

March 2013

Mon	Tue	Wed	Thu	Fri	Sat	Sun
25	26	27	28	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

Today

Time:

02:02 AM

DateTime:

2013-03-08 05:46 UTC

DateTime Local:

2013-03-13 12:02

```
<label>Date: <br/>  
<input type= "date" value="" />
```

```
<input type= "time" value="" />
```

```
<input type= "datetime" value="" />
```

```
<input type= "datetime-local" value="" />
```

HTML5 Date and Time Controls

Month:

March, 2013



March, 2013



Sun	Mon	Tue	Wed	Thu	Fri	Sat
24	25	26	27	28	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

This month

Clear

`<input type="month" value="2013-03"/>`

Week:

2013-W10

March



2013



Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
9	25	26	27	28	1	2	3
10	4	5	6	7	8	9	10
11	11	12	13	14	15	16	17
12	18	19	20	21	22	23	24
13	25	26	27	28	29	30	31
14	1	2	3	4	5	6	7

Today

`<input type="week" value="2013-W10"/>`

HTML Controls

Type	Description
date	Creates a general date input control. The format for the date is "yyyy-mm-dd".
time	Creates a time input control. The format for the time is "HH:MM:SS", for hours:minutes:seconds.
datetime	Creates a control in which the user can enter a date and time.
datetime-local	Creates a control in which the user can enter a date and time without specifying a time zone.
month	Creates a control in which the user can enter a month in a year. The format is "yyyy-mm".
week	Creates a control in which the user can specify a week in a year. The format is "yyyy-W##".

Finally

- Submit
- Reset

```
<input type="reset" value="Clear form" />
```

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

<form> element

- Two essential features of any form, namely the **action** and the **method** attributes.
 - The **action** attribute specifies the URL of the server-side resource that will process the form data
 - The **method** attribute specifies how the query string data will be transmitted from the browser to the server.
 - GET
 - POST

GET vs POST

A screenshot of a web browser window titled 'Sample Form'. The address bar shows 'form.html'. The form contains a 'Details' section with a 'Title' text input field containing 'Central Park' and a 'Country' dropdown menu set to 'United States'. Below these fields is a 'Submit' button.

`<form method="get" action="process.php">`

`GET /process.php?title=Central+Park&where=United+States http/1.1`

querystring

`<form method="post" action="process.php">`

```
POST /process.php http/1.1
Date: Sun, 20 May 2012 23:59:59 GMT
Host: www.mysite.com
User-Agent: Mozilla/4.0
Content-Length: 47
```

HTTP Header

`title=Central+Park&where=United+States`

querystring

GET vs POST

- **Advantages and Disadvantages**

- Data can be clearly seen in the address bar.
- Data remains in browser history and cache.
- Data can be bookmarked
- Limit on the number of characters in the form data returned.

POST

- Data is binary data.
- Data is hidden from user.
- Submitted data is not stored in cache, history, or bookmarks.

GET vs POST

- GET must NOT be used when sending sensitive information! GET is best suited for short, non-sensitive, amounts of data, because it has size limitations too.
- Always use POST if the form data contains sensitive or personal information. The POST method does not display the submitted form data in the page address field.
- POST has no size limitations, and can be used to send large amounts of data.

AUDIO AND VIDEO

Audio and Video

- Until HTML5, adding audio or video to a web page typically required making use of additional, often proprietary, plug-ins to the browser
- Perhaps the most common way of adding audio and video support until recently was through Adobe Flash
- it is possible now with HTML5 to add these media features in HTML without the involvement of a plug-in

Using the <audio> element

```
<html>
<audio
  id="harrybelafonte"
  controls
  src="http://localhost/media/malaika.mp3">
  Your browser does not support the <code>audio</code> element.
</audio>
</html>
```

Properties of Audio

```
<html>
<audio controls="" loop="yes" autoplay="no" >
  <source src="foo.opus" type="audio/ogg; codecs=opus"/>
  <source src="foo.ogg" type="audio/ogg; codecs=vorbis"/>
  <source src="foo.mp3" type="audio/mpeg"/>
</audio>
</html>
```

Using the <video> element

```
<html>
<video controls
    muted
    src="/media/stream_of_water.webm"
    width="300"
    height="200">
```

Sorry, your browser doesn't support embedded videos.

```
</video>
</html>
```

Another video example

```
<html>
<video controls>
  <source src="foo.mp4" type="video/mp4">
  <source src="foo.webm" type="video/webm">
  <p>Your browser doesn't support HTML5 video. Here is
    a <a href="foo.mp4">link to the video</a> instead.</p>
</video>
</html>
```

Authoritative Reference

- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element>
- Use this website for any doubts on which element does what

That's all, folks!

- Questions?