

# COMP10001 Foundations of Computing

## HTML; Python and the Web

Semester 2, 2018  
Chris Leckie & Nic Geard



THE UNIVERSITY OF  
MELBOURNE

# Lecture Agenda

- Last week:
  - URLs and the Internet
- This lecture:
  - HTML
  - Web programming

# HTML: Document Structure

- All HTML documents should start with a declaration of “document type” on the first line:

```
<!DOCTYPE html>
```

be enclosed within `<html></html>` tags, and contain a “head” (`<head></head>`) and “body” (`<body></body>`) respectively, i.e.:

```
<!DOCTYPE html>
<html>
<head>
...
</head>
<body>
...
</body>
</html>
```

# HTML: The Head

- The head of an HTML document standardly contains a title:

```
<title></title>
```

and will also often contain “meta-data” as attributes to empty `<meta/>` elements, including keywords, character encoding information, a description of the site, ...

```
<head>
<title>HTML Introduction</title>
<meta name="description" content="An intro to HTML"/>
<meta name="keywords" content="HTML, computing, coolness"/>
<meta name="author" content="Tim Baldwin"/>
<meta charset="UTF-8"/>
</head>
```

# HTML: The Body

- Common elements of the body of an HTML document to structure the text are:
  - headers (`<h1></h1>`, `<h2></h2>`, ...)
  - paragraphs (`<p></p>`)
  - line breaks (`<br/>`)
  - horizontal lines (`<hr/>`)
- “Hyperlinks” can be inserted with `<a href=""></a>` over “anchor text”

```
<body>  
<a href="./index.html">Recursive link!</a>  
</body>
```

# HTML: White Space

- White space can be inserted for readability, but is largely ignored by the browser: the browser turns any sequence of white space characters into a single space before processing
- Exception: preformatted information between `<pre>``</pre>` tags is displayed as it appears

# HTML: More on Hyperlinking

- URLs in hyperlinks can be:
  - “absolute URLs”, i.e. complete URLs including hostname, such as  
`http://server/directory_path/filename`
  - “relative URLs”, i.e. relative to the current location on the same server, such as  
`./a_file_in_the_same_directory.html`
- In relative URLs, we often use two special characters:
  - `.` = Current directory
  - `..` = Parent directory (one level up)
- Relative URLs are more flexible, as it is possible to move web page sets around as a group without having to update URLs

# HTML: Multimedia Content

- Images (of varying formats) can be included with `<img src="" alt=""/>`, where `src` specifies the image file location, and `alt` is alternate text (if the image doesn't load)
- Audio files can be included with `<audio><source src="" type=""/>alt</audio>`
- Video files can be included with `<video><source src="" type=""/>alt</video>`



# HTML: Lists

- Enclose unnumbered (bulleted) lists with `<ul></ul>`
  - declare list items with `<li></li>`
- Enclose ordered (numbered) lists with `<ol></ol>`
  - declare list items with `<li></li>`

```
<ul>
  <li>Paul</li>
  <li>John</li>
  <li>George</li>
  <li>Ringo</li>
</ul>
<ol>
  <li>Computing</li>
  <li>Everything else</li>
</ol>
```

# HTML: Tables

- Enclose tables with `<table></table>` (with optional border attribute)
- Render the table a row at a time, enclosing each row with `<tr></tr>`, and each cell with `<td></td>` or `<th></th>` (for column headers)

```
<table border="1">
  <tr>
    <th>First name</th>
    <th>Last name</th>
  </tr>
  <tr>
    <td>Tim</td>
    <td>Baldwin</td>
  </tr>
</table>
```

# HTML: Tables

- Enclose tables with `<table></table>` (with optional border attribute)
- Render the table a row at a time, enclosing each row with `<tr></tr>`, and each cell with `<td></td>` or `<th></th>` (for column headers)

```
<table border="1">  
<tr>  <th>First name</th>  <th>Last name</th>  </tr>  
<tr>  <td>Tim</td>          <td>Baldwin</td>    </tr>  
</table>
```

| First name | Last name |
|------------|-----------|
| Tim        | Baldwin   |

# HTML: Entities

- HTML “entities” are special characters, and take the form `&entity;`
- The most commonly used entities are:

|                         |   |                         |       |
|-------------------------|---|-------------------------|-------|
| <code>&amp;quot;</code> | " | <code>&amp;nbsp;</code> | space |
| <code>&amp;lt;</code>   | < | <code>&amp;apos;</code> | '     |
| <code>&amp;gt;</code>   | > | <code>&amp;amp;</code>  | &     |

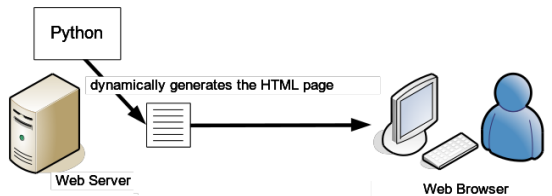
- There are also entities for characters with diacritics, such as `ü` = `&uuml;`, `é` = `&eacute;`, `ì` = `&igrave;`

# Serving HTML Pages

- Static HTML



- Dynamic HTML using Python



# So How does it Work?

- Dynamically generating a web page simply involves “serving” the printed HTML output of a Python script

# Exercise 1

- Write a function to print the sorted contents of list shopping as an HTML list

```
<ul>  
  <li>biscuits</li>  
  <li>bread</li>  
  <li>cheese</li>  
  <li>milk</li>  
</ul>
```

## Exercise 2

- Write a function to print the number of letters shared between each pair of strings in a list of strings `strlist` as an HTML table

```
<table border="1">
  <tr>
    <th>&nbsp;</th><th>apple</th><th>banana</th>
  </tr>
  <tr>
    <th>apple</th><td>5</td><td>1</td></tr>
  </tr>
  <tr>
    <th>banana</th><td>1</td><td>6</td></tr>
  </tr>
</table>
```



# Lecture Summary

- What are HTML elements, tags, attributes and entities?
- What are the essential elements of an HTML document?
- How do you include hyperlinks/multimedia files in HTML documents?
- How do you typeset lists and tables?
- How can you generate dynamic HTML pages from Python?