Maharishi International University 1971-1995

MAHARISHI UNIVERSITY OF MANAGEMENT

Engaging the Managing Intelligence of Nature

CS472: Web Application Programming

Teaching Faculty: Dr. Keith Levi

2019

Maharishi's Eleventh Year of Global Ram Raj

Maharishi University of Management -Fairfield, Iowa © 2016

Wholeness Statement

In this lecture we introduce the basic technologies that make up the Internet, the World Wide Web and the Hyper Text Markup Language (HTML). We will see that many technologies are built on top of other technologies.

Life is found in layers and the TM Technique gives us access to the full range of our awareness and thoughts.

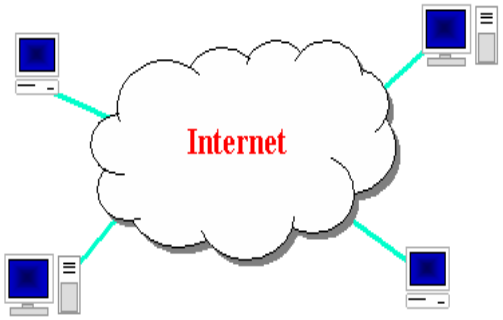
Main Point Preview

The internet is a global computer network that uses IP addresses to uniquely identify computers on the network. Through the TCP protocol each IP address can work with multiple services at the same time. One of these services is the HTTP protocol which is used by the World Wide Web to transport HTML pages.

These are many layers of the internet. *Familiarity with the deep layers is necessary for optimal understanding and use of the surface layers. TM settles the mind so that it is more connected to the deeper layers that are the basis of the more expressed layers.*

The Internet

- A connection of computer networks using the Internet Protocol (IP)
- layers of communication protocols: IP → TCP/UDP → HTTP/FTP/POP/“MTP/““H...



- What's the difference between the Internet and the World Wide Web (WWW)?
- The Web is the collection of web sites and pages around the world; the Internet is larger and also includes other services such as email, chat, online games, etc.

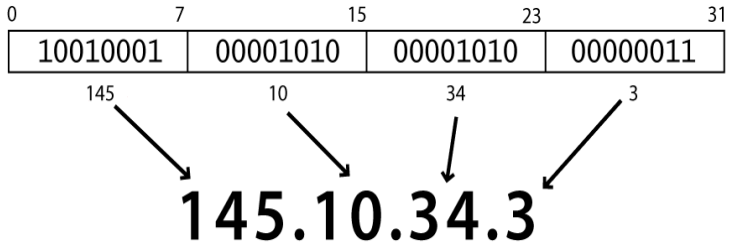
Brief history

- Began as a US Department of Defense network
  - called ARPANET (1960s-70s)
- Initial services: Electronic mail, File transfer
- Opened to commercial interests in late 80s
- WWW created in 1989-91 by Tim Berners-Lee
- Popular Web browsers released: Netscape 1994, IE 1995
- Amazon.com opens in 1995; Google January 1996; Facebook 2004; Uber 2009

- https://en.wikipedia.org/wiki/History_of_the_Internet

Internet Protocol (IP) IPv4

- The underlying system of communication for all data sent across the Internet.
- Each device has a 32-bit IP address written as four 8-bit numbers (0-255)
- There are two types of IP addresses, servers used to have static IP address while users usually get a dynamic IP address from their ISP.
- Find out your local IP address: in a terminal, type: ipconfig (Windows) or ifconfig (Mac/Linux)



- IPv6 addresses are 128-bit IP address written in hexadecimal and separated by colons. An example IPv6
- address could be written like this: 3ffe:1900:4545:3:200:f8ff:fe21:67cf

Transmission Control Protocol (TCP)

- Adds multiplexing, guaranteed message delivery on top of IP
- **Multiplexing**: multiple programs using the same IP address
  - **port**: a number given to each program or service
  - port 80: web browser (port 443 for secure browsing)
  - port 25: email
  - port 22: ssh
  - Port 21: File transfer (FTP)
  - port 5190: AOL Instant Messenger
  - more common ports
- Some programs (games, streaming media programs) use simpler UDP protocol instead of TCP

Web servers and Browsers

- **Web server**: software that listens for web page requests
  - Apache
  - Microsoft Internet Information Server (IIS)
- **Web browser**: fetches/displays documents from web servers
  - Mozilla Firefox
  - Microsoft Internet Explorer (IE)
  - Apple Safari
  - Google Chrome
  - Opera
  - Browser usage statistics

Domain Name System (DNS)

- A set of servers that map written names to IP addresses
  - Example: www.cs.mum.edu → 69.18.50.54

Uniform Resource Locator (URL)

An identifier for the location of a document on a web site

a basic

URL:http://www.abc.com/info/index.html

~~~ ~~~~~~~~~~ ~~~~~~~~~~

protocol host path

- Upon entering this URL into the browser, it would:
- Ask the DNS server for the IP address of www.abc.com
- Connect to that IP address at port 80
- Ask the server to GET /info/index.html
- Display the result page on the screen
~~~

More advanced URLs

- **Anchor**: jumps to a given section of a web page
  - http://www.textpad.com/download/index.html#downloads
  - fetches **`index.html`** then jumps down to part of the page labeled downloads
- **Port**: for web servers on ports other than the default 80
  - http://www.cs.mum.edu:8080/secret/money.txt
- **Query string**: a set of parameters passed to a web program
  - http://www.google.com/search?q=miserable+failure&start=10
  - parameter q is set to "miserable+failure"
  - parameter start is set to 10

Hypertext Transport Protocol (HTTP)

- a set of commands that a computer can send to a server to request files
- Some HTTP commands (your browser sends these internally):
  - **GET**: Requests a specific file or resource from the server
  - **POST**: Submits form information to the server
  - **PUT**: Uploads a file to the server
  - **HEAD**: Requests information about a file from the server, but not the file's entire contents.

HTTP request and response messages

- Request

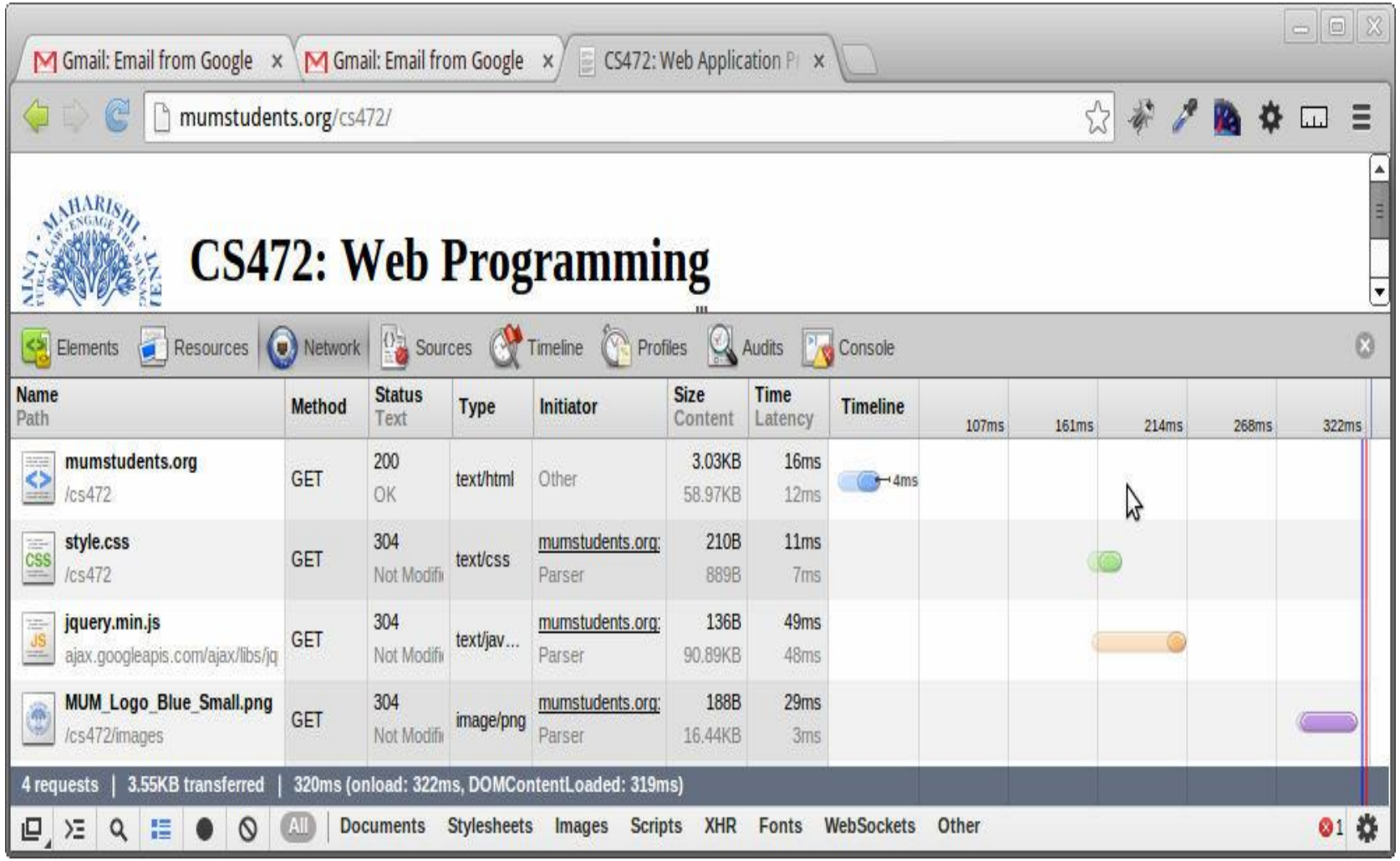
GET /index.html HTTP/1.1
HOST: mumstudents.org

- Response

HTTP/1.1 200 OK
Date: Sun, 30 Aug 2015 16:08:38 GMT
Server: Apache/2.4.7 (Ubuntu)
Last-Modified: Fri, 12 Dec 2014 20:23:01 GMT
Accept-Ranges: bytes
Content-Length: 942
Content-Type: text/html

<!DOCTYPE html>
<html>
...

Request and Response



HTTP status code

- When you request a document from a web server, it sends this document back to you, along with a number called HTTP status code.

Number	Meaning
200	OK
301-303	page has moved (permanently or temporarily)
403	you are forbidden to access this page
404	page not found
500	internal server error
complete list	

MIME Types (Multi-Purpose Internet Mail Extension)

- A two-part identifier which many web protocols use to categorize each type of data. broad type/subtype

MIME Type	File extension(s)	Description
application/octet-stream	.exe	Executable programs
audio/mpeg	.mp3, .mpg	MPEG or MP3 music
image/gif, image/jpeg, image/png	.gif, .jpg, .png	GIF, JPEG, PNG images
text/css	.css	Style sheets
text/html	.html, .htm, .php	Web pages
MIME Type list		

Web languages / technologies

- Hypertext Markup Language (HTML): used for writing web pages
- Cascading Style Sheets (CSS): stylistic info for web pages
- PHP Hypertext Processor (PHP): dynamically create pages on a web server
- JavaScript: interactive and programmable web pages
- Asynchronous JavaScript and XML (Ajax): accessing data for web applications
- Extensible Markup Language (XML): metalanguage for organizing data
- JavaScript Object Notation (JSON): lightweight data-interchange format that is largely replacing XML in modern web apps

How browsers display a webpage

1. User machines have IP address on the Internet
2. Server machines have IP address and Domain Name
3. Domain names and IP addresses are registered at global DNS Server
4. When the user opens a browser window and asks for www.test.com
5. First, the browser will check the local DNS (host file) for the IP address of that domain
6. If not found, it will connect to ISP and ask it for the DNS
7. Once retrieved, the browser will send another request to that server
8. Requests are delivered by the IP protocol, collected by the TCP protocol, and processed by HTTP or HTTPS protocol
9. The server will send the browser a response with HTML code.
10. The browser will interpret the HTML code line by line and start building the web page.
11. For every resource not found in the browser cache, the browser will send a new request to the server again asking for that resource and so on.

Main Point

The internet is a global computer network that uses IP addresses to uniquely identify computers on the network. Through the TCP protocol each IP address can work with multiple services at the same time. One of these services is the HTTP protocol which is used by the World Wide Web to transport HTML pages.

These are many layers of the internet. *Familiarity with the deep layers is necessary for optimal understanding and use of the surface layers. TM settles our mind so that it is more connected to the deep layers that are the basis of the more expressed layers.*

Main Point Preview

The Hyper Text Markup Language uses tags to demarcate different sections of a text. An HTML page always starts with a <html> tag, inside of which it has a <head> tag to describe the page, and a <body> tag of the contents that will actually be displayed. These are the tags you will use for every HTML page. This is a foundational concept

Well begun is half done. Start with a good foundation and build upon that.

Hypertext Markup Language (HTML)

- Describes the content and structure of information on a web page
- Surrounds text content with opening and closing tags
- Each tag and its content is called an element
  - Syntax: **<element>**content**</element>**
  - Example: ** **This is a paragraph** **
- Most whitespace is insignificant in HTML (ignored or collapsed to a single space)
- The newest version is HTML5

Structure of an HTML5 page

- The **header** describes the page and the **body** contains the page's contents
  - An HTML page is saved into a file ending with extension **.html**
- **DOCTYPE** tag tells browser the HTML version.
  - In this case, HTML5.

```
<!DOCTYPE html>
<html>
  <head>
     <meta charset="utf-8"/>
     <title></title>
  </head>
  <body>
     Page contents
  </body>
</html>
```

Web Page Metadata

- Data about data - Information describes the page itself

```
<meta charset="utf-8"/>
<meta name="description" content="Learn HTML"/>
<meta name="keywords" content="HTML,CSS"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
```

- Placed in the **head** section of your HTML page
- **meta** tags often have both the **name** and **content** attributes
- HTML5 introduced a method to give web designers control over the viewport (the user's visible area of a web page), through the <meta> tag

Metadata

Attribute	Value	Description
charset	character_set	Specifies the character encoding for the HTML document
name	application-name author description generator keywords viewport	Specifies a name for the metadata
content	text	Gives the value associated with the http-equiv or name attribute
http-equiv	content-type default-style refresh	Provides an HTTP header for the information/value of the content attribute

Favorites icon ("favicon")

```
<link href="filename" type="MIME type" rel="relationship"/>
<link href="yahoo.gif" type="image/gif" rel="shortcut icon"/>
```

- The link tag, placed in the head section, attaches another file to the page
- In this case, an icon to be placed in the browser title bar and bookmarks
- E.g. https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_link_sizes

Relative vs Absolute URL

- **Relative URL**
  - index.html (path relative)
  - graphics/image.png
  - ../about.html (directory relative)
  - ../../stories/stories.html
  - /image.png (root relative)
- **Absolute URL**
  - http://www.mysite.com
  - C:\website\images\image.png
- Why won't the last example work when we move our code to production?

Common Error on Relative URL

- Placing a leading “/” on a relative URL

http://www.myexample.com/mysite/files/index.html
http://www.myexample.com/mysite/files/images/smiley.gif

Looking for URL:
http://www.myexample.com/images/smiley.gif

Block-level Elements

- A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).

- Examples of block-level elements:
  -
  - <h1> - <h6>
  - <p>
  - <form>

Paragraph: <p>

- Paragraphs of text (block)

```
<p>
Lorem quis lorem.  Pellentesque ultrices nunc id mauris
posuere pulvinar.
</p>
```

Placed within the body of the page

- More paragraph examples

Headings: **<h1>, <h2>, ..., <h6>**

- Headings to separate major areas of the page (block)

```
<h1>Maharishi University</h1>
<h2>Department of Computer  Science</h2>
<h3>WAP Course</h3>
```

Maharishi University

Department of Computer Science

WAP Course

Horizontal Rule: <hr />

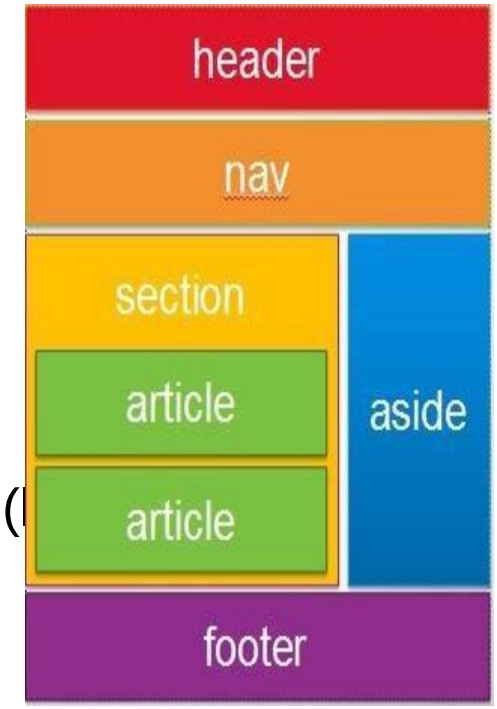
- A line to separate sections (block)

```
<h1>CS472 Web Programming</h1>
<p>Very good, fun course - HTML, CSS, JavaScript, JSP, Servlets</p>
<hr />
<h1>CS421 Modern Programming Practices</h1>
<p>Profound course - OOAD, Java8 features</p>
```

New Semantic Elements

They generally have no default outward appearance on the page, instead they give insight into the structure of the page.

- **section**: defines a section in a document.
- **header**: specifies a header for a document or section.
- **footer**: specifies a footer for a document or section.
- **nav**: defines a set of navigation links.
- **aside**: defines some content aside from the content it is placed in (
- **article**: specifies independent, self-contained content.



More Semantic Tags

Main point

The Hyper Text Markup Language uses tags to demarcate different sections of a text. An HTML page always starts with a <html> tag, inside of which it has a <head> tag to describe the page, and a <body> tag of the contents that will actually be displayed. These are the tags you will use for every HTML page. This is a foundational concept

Well begun is half done. Start with a good foundation and build upon that.

Main Point Preview

More common tags include inline elements, lists, tables. The most important concept is to use tags based on their semantics (meaning), not based on their visual effect (which can easily be changed). By using tags for their meaning, clients (including non visual) will better understand the meaning of your page and use it more effectively.

Clear and settled awareness allows us to better grasp the meaning of information and take the right actions for success.

Inline Elements

- An inline element does not start on a new line and only takes up as much width as necessary.

- Examples of inline elements:
  -
  - <a>
  - <img>

Images: <img>

- Inserts a graphical image into the page (inline)
  - Another get request
- The src attribute specifies the image URL
- HTML5 also requires an alt attribute describing the image
- title attribute is an optional tooltip (on ANY element)

```
<img src="../images/WhiteLily.jpg" alt="White Lily"/>
```

- See example: lecture01-examples/image.html

Links: <a>

- Links, or "anchors", to other pages (inline)
- href can be absolute or relative URL
- Anchors are inline elements

```
<p>
 <a href="story1.html"><img src="bruce.jpg" alt="full story. " />Bruce
 Wayne, the richest man in Gotham City, is the alter ego of Batman.
 Bam! </a>
</p>
```

- NOTE: In HTML5, you can wrap links around “block-level” elements
- See example: lecture01-examples/link-old.html, lecture01-examples/link-html5.html

Line break:

- Forces a line break in the middle of a block element (inline)

```
<p>Teddy said it was a hat, <br /> So I put it on.</p>
<p>Now Daddy's saying, <br /> Where the heck's the toilet plunger gone?</p>
```

- Warning: Don't over-use `br` (guideline: >= 2 in a row is bad)
- `<br />` should not be used to separate paragraphs or used multiple times in a row to create spacing

Phrase elements : ,

- em: emphasized text (usually rendered in italic)
- strong: strongly emphasized text (usually rendered in bold)

```
<p>HTML is
   <em>really</em>,<strong>REALLY</strong>
    fun!
</p>
```

HTML is *really*,**REALLY** fun!

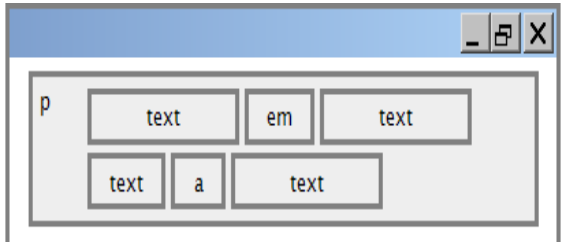
- As usual, the tags must be properly nested for a valid page.

Nesting tags

- Tags must be correctly nested
  - a closing tag must match the most recently opened tag
- The browser may render it correctly anyway, but it is invalid HTML
- i.e., bad nesting
  - What a lovely day
- Good nesting
  - What a lovely day

Review: Block elements VS Inline elements

- block elements contain an entire large region of content
  - Examples: paragraphs, lists, table cells
  - The browser places a margin of whitespace between block elements for separation, normally a newline
- inline elements affect a small amount of content
  - Examples: bold text, code fragments, images
  - The browser allows many inline elements to appear on the same line
  - Must be nested inside a block element



Comments: <!-- -->

- Comments to document your HTML file or "comment out" text
- Many web pages are not thoroughly commented (or at all)
- Useful at the top of page and for disabling code

```
<!– This is comment  -->
<p>
     WAP courses  are<!-- NOT-->   a   lot of fun!
</p>
```

Unordered list: `<ul>`, `<li>`

- `ul` represents a bulleted list of items (block)
- `li` represents a single item within the list (block)

```
<ul>
 <li>No shoes</li>
 <li>No shirt</li>
 <li>No problem!</li>
</ul>
```

- No shoes
- No shirt
- No problem!

Ordered list: `<ol>`

- `ol` represents a numbered list of items (block)
- We can make lists with letters or Roman numerals using CSS (later)

```
<p>Concise business model:</p>
<ol>
 <li>Find customers</li>
 <li>Make profits</li>
</ol>
```

Concise business model:

1. Find customers
2. Make profit!

Nested Lists

- A list can contain other lists

```
<ul>
   <li>Simpsons:
      <ul>
            <li>Homer</li>
            <li>Marge</li>
            </ul>
   </li>
   <li>Family  Guy:
         <ul>
          <li>Peter</li>
          <li>Lois</li>
         </ul>
   </li>
</ul>
```

- Simpsons:
  - Homer
  - Marge
- Family Guy:
  - Peter
  - Lois

Definition list: <dl>, <dt>, <dd>

- `dl` represents a list of definitions of terms (block)
- `dt` represents each term, and `dd` its definition

```
<dl>
 <dt>newbie</dt>
 <dd>one who does not have mad skills</dd>
 <dt>own</dt>
 <dd>to soundly defeat</dd>
 <dt>frag</dt>
 <dd>a kill in a shooting game</dd>
</dl>
```

newbie
: one who does not have mad skills

own
: to soundly defeat

frag
: a kill in a shooting game

Inline quotations: <q>

- A short quotation (inline)

```
<p>
      Quoth the Raven,<q>Nevermore.</q>
</p>
```

Quoth the Raven,"Nevermore."

Why not just write the following?

```
<p>Quoth the Raven,“Nevermore.”</p>
```

- Using <q> allows us to apply CSS styles to quotations

Quotations: <blockquote>

A lengthy quotation (block)

```
<p>As Lincoln said in his famous Gettysburg Address:
</p>
<blockquote>
  <p>Fourscore and seven years ago, our
   fathers brought forth onthis continent a
   new nation, conceived in liberty, and
  dedicated to the proposition that all
   men are created equal.</p>
</blockquote>
```

As Lincoln said in his famous Gettysburg Address:

> Fourscore and seven years ago, our fathers brought forth on this continent a new nation, conceived in liberty, and dedicated to the proposition that all men are created equal.

HTML Character Entities

- A way of representing any Unicode character within a web page
- Complete list of HTML entities
- How would you display the text & on a web page?

character(s)	entity
< >	< >
é è ñ	é è ñ
™ ©	™ ©
π δ Δ	π δ Δ
И	&#1048;
" &	" &

HTML-encoding text

- To display the link text in a web page, its special characters must be encoded as shown below

```
&lt;p&gt;
    &lt;a href=&quot;http://google.com/&quot;&gt; Google&lt;/a&gt;
&lt;/p&gt;
```

```
<p> <a href="http://google.com/"> Google</a> </p>
```

Computer code: <code>

- A short section of computer code (usually shown in a fixed-width font) (inline)

```
<p>
    The <code>ul</code> and <code>ol</code>tags make
    lists.
</p>
```

The ul and ol tags make lists.

Preformatted text: <pre>

- A large section of pre-formatted text (block)
- a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

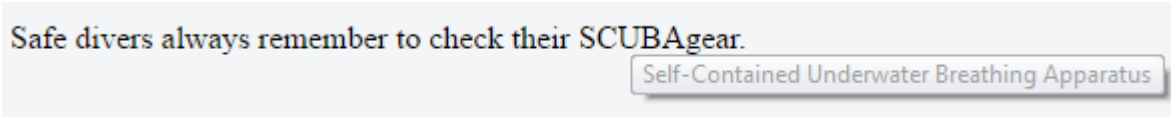
```
<pre>
        Steve Jobs speaks loudly reality
                distortion Apple fans bow down
</pre>
```

```
Steve Jobs speaks loudly reality
        distortion Apple fans bow down
```

Abbreviations: <abbr>

- An abbreviation, acronym, or slang term (inline)

```
<p> Safe divers always remember to check their
  <abbr title="Self-Contained Underwater Breathing Apparatus">SCUBA</abbr>
gear.
</p>
```



HTML tables: <table>, <tr>, <td>

- A 2D table of rows and columns of data (block element)

```
<table>
  <tr>
    <td>1,1</td>
    <td>1,2 okay</td>
  </tr>
  <tr>
    <td>2,1 real wide</td>
    <td>2,2</td>
  </tr>
</table>
```

1,1	1,2 okay
2,1 real wide	2,2

- **`table`** defines the overall table, **`tr`** each row, and **`td`** each cell's data
- tables are intended for displaying large row/column data sets

Table headers, captions: <th>, <caption>

```
<table>
  <caption>My important data</caption>
  <tr>
    <th>Column 1</th>
    <th>Column 2</th>
  </tr>
  <tr>
    <td>1,1</td>
    <td>1,2 okay</td>
  </tr>
  <tr>
    <td>2,1 real wide</td>
    <td>2,2</td>
  </tr>
</table>
```

My important data

Column 1	Column 2
1,1	1,2 okay
2,1 real wide	2,2

- **th** cells in a row are considered headers; by default, they appear bold
- a **caption** at the start of the table labels its meaning

The **rowspan**and **colspan**attributes

- **colspan** makes a cell occupy multiple columns; rowspan multiple rows

```
<table>
  <tr>
    <th>Column 1</th>
    <th>Column 2</th>
    <th>Column 3</th>
  </tr>
  <tr>
    <td colspan="2">1,1-1,2</td>
    <td rowspan="3">1,3-3,3</td>
  </tr>
  <tr>
    <td>2,1</td>
    <td>2,2</td>
  </tr>
  <tr>
    <td>3,1</td>
    <td>3,2</td>
  </tr>
</table>
```

Column 1	Column 2	Column 3
1,1-1,2		
2,1	2,2	1,3-3,3
3,1	3,2	

<table>
<tr><th>Column 1</th><th>Column 2</th><th>Column 3</th></tr>
<tr><td colspan="2">1,1-1,2</td><td rowspan="3">1,3-3,3</td></tr>
<tr><td>2,1</td><td>2,2</td></tr>
<tr><td>3,1</td><td>3,2</td></tr>
</table>

Don't use tables for layout!

- tables *appear* to be an easy way to achieve grid-like page layouts
- Many "newbie" web pages do this
- But, a table has semantics; it should be used only to represent an actual table of data
- Instead of tables, use divs, widths/margins, floats, flexbox, grid, etc to perform layout

W3C HTML Validator

- Checks your HTML code to make sure it follows the official HTML syntax
- More picky than the browser, which may render bad HTML correctly

```
<p>
 <a href="http://validator.w3.org/check/referer">
    <img src=http://mumstudents.org/cs472/2013-09/images/w3c-html.png alt="Validate" />
 </a>
</p>
Note that the validation website expects requests to come from a page that was
served by a web server.  Will not work with an html page opened from a file.
E.g., mumstudents.org web server.
```

Main Point

More common tags include inline elements, lists, tables. The most important concept is to use tags based on their semantics (meaning), not based on their visual effect (which can easily be changed). By using tags for their meaning, clients (including non visual) will better understand the meaning of your page and use it more effectively.

Clear and settled awareness allows us to better grasp the meaning of information and take the right actions for success.

CONNECTING THE PARTS OF KNOWLEDGE WITH THE WHOLENESS OF KNOWLEDGE

Layers of Abstraction

1. HTML is the basis of web programming, every web page is composed of HTML.
2. To be an effective web programmer you also must understand the deeper underlying realities of HTTP, TCP, and DNS.

---

3. **Transcendental consciousness** is when our mind is in contact with the deepest underlying reality, the unified field.
4. **Impulses within the Transcendental field:** the infinite dynamism of the unified field constantly expresses itself as all of the layers of the universe
5. **Wholeness moving within itself:** In Unity Consciousness, one experiences that all these layers are ultimately composed of pure consciousness, our own pure awareness.