1. HTML
    a. origins
        i. Hypertext Markup Language
        ii. Developed in 1991 by British physicist Tim Berners-Lee
        iii. Allows the formatting of information to be defined separately from the content
        iv. Based on SGML (Standard Generalized Markup Language) – a metalanguage which defines formatting for documents
        v. Browsers understand HTML
        vi. Web servers hold the content and send out HTML to the client (browser)

    b. governing body
        i. W3C – World Wide Web Consortium, an international standards body
        ii. Formed to resolve HTML incompatibility issues among vendors in the 1990's

    c. white space – free format
        i. Any spaces, tabs, new lines are ignored when the browser interprets the HTML
        ii. Use the HTML tag <pre> .. </pre> to maintain the spacing content
        iii. Use
 to force a new line

    d. comments in HTML
        i. <!-- A comment in HTML and ignored by the browser -->

    e. browser differences
        i. window panel dimensions
            1. Internet Explorer, Mozilla Firefox, Safari
        ii. colour depth
            1. the more bits per pixel, the greater the colour palette (e.g. 16 bits per pixel = 2^{16} colours)
            2. resolution - how many pixels will fit on a video screen
            3. 'websafe' colours
                a. set of 216 colours which will display without dithering on video cards supporting only 256 colours (the '8-bit' cards – now obsolete)
                b. dithering – process of creating the illusion of colour depth by placing alternating pixels of different colours next to each other to make a new colour

    f. tags
        i. always enclosed by angle brackets < and >
        ii. no space after the <
        iii. in HTML tags can be lowercase or uppercase

iv. in HTML tags can drop their end tag

g. structure
    i. file name uses an extension of ".html" or ".htm"
    ii. <!DOCTYPE>
        1. tells the browser which HTML definition type (transitional, strict, frameset) this document uses
    iii. <html>
        1. start of the HTML content … ends with </html>
    iv. <head>
        1. <title>
            a. Show the window title
            b. Text allowed – no HTML allowed in title
        2. <meta>
            a. Information about this HTML document, author
        3. <head> <title> .. </title> </head>
    v. <body>
        1. start of the HTML body .. ends with </body>

h. attributes
    i. HTML attributes have the form attribute=value (for example, <img src= reddot.png>)

i. inline
    i. tags that will not cause automatic line break in the browser window
    ii. <b>, <i>, , , , <img>, <a>, <q>,
j. block
    i. tags that will cause automatic line breaks in the browser window
    ii. <p>, <dl>, <table>, <ul>, <ol>, , <li>, <h1>, <h2>

k. image <img>
    i. formats – some proprietary, some free
        1. png
            a. seen as the replacement of gif – has better compression and not limited to 256 colours
        2. bmp
            a. Windows bitmap file
            b. DIB file (device independent bitmap)
        3. jpeg
            a. 'lossy' – uses compression to store image data
            b. allows 16.7 million colours (24 bit)
            c. photorealistic images
            d. does not support transparency
        4. gif

            a. standard gif image can have at most 256 colours from a palette of 16.7M
            b. gifs can be ‘animated’ by stacking frames of images together
            c. useful for small images having less detail
        5. svg
            a. scalable vector graphics
            b. greater compression of images using formulas not bitmaps
    ii. attributes
        1. alt
            a. show what image represents if the image cannot be displayed
        2. title
            a. pop-up bubble briefly describing the image
        3. border
            a. can define a border surrounding the image
            b. deprecated in favour of CSS
        4. width and height
            a. horizontal and vertical scale factors
            b. can be in measurements of pixels, in, cm, or %
            c. deprecated in favour of CSS

l. table
    i. definition
        1. <table> table rows </table>
        2. table rows : <tr> row data </tr>
        3. row data : <td> data </td>
    ii. a table may be defined within the cell of another table (nested table)
    iii. tables rows usually have the same number of columns; table columns usually have the same number of rows
    iv. width
        1. widest cell in the column determines the width of the entire column
        2. the width attribute fixes the width of a table or column
    v. border
        1. size of the border around the table
    vi. rowspan
        1. when a cell spans two or more rows, specify the rowspan attribute in the <td> tag
    vii. colspan
        1. when a cell spans two or more columns, specify the rowspan attribute in the <td> tag
    viii. cellspacing

1. amount of space between cells

ix. cellpadding
    1. amount of space between the cell border its contents

x. thead tfoot tbody
    1. define the table header, table footer, and table body

m. form

i. structure
    1. <form> …. </form>
    2. a form may not be defined within another form

ii. form attributes
    1. name
        a. unique name for the form (for Javascript)
    2. id
        a. unique identifier for form (for CSS)
    3. action
        a. define the script that will run when submit button is pressed
        b. the URL to which the user entered form data will be submitted
    4. method
        a. method= "GET" is default if not defined – encodes the form information in the URL (less secure) – the form values you entered will appear in the URL
        b. method= "POST' is more secure

iii. form elements
    1. attributes
        a. disabled= "true" – form element never receives focus
        b. readonly= "true" – form element information cannot be overridden
        c. accesskey – keyboard shortcut to form element
        d. tabindex – define the form element tab order

iv. textarea
    1. multiline text
    2. rows attribute – number of rows
    3. cols attribute – number of columns

v. select
    1. shows a "dropdown menu list"
    2. option attribute
    3. to specify a default: selected= "selected"
    4. multiple= "multiple"
    5. similar to a listbox
    6. optgroup attribute
        a. <optgroup label= "Country">
        <option>Canada</option>

<option>USA</option>
</optgroup>

vi. input
    1. text
        a. simple one-line text input
        b. value attribute defines a default, pre-entered string
        c. maxlength attribute – maximum characters users may enter
    2. password
        a. inputted text is echoed as asterisks
        b. low level of security
    3. radio
        a. selection of mutually exclusive option
        b. name attribute among options must be defined the same
        c. value attribute may be any text or number
        d. checked attribute
    4. checkbox
        a. selection of multiple options
        b. name attribute among options
        c. value attribute may be any text or number
        d. checked attribute
    5. hidden
        a. a form element not displayed but its value is used by the action script
    6. HTML 5 attributes for input element
        a. pattern
            i. use a regular expression to validate input data (e.g. \d or [0-9] for a single digit, \d{2} for exactly two digits, \d+ for one or more digits)
        b. required – field may not be left blank upon submit
        c. type= “range” min= “0” max= “100”
    7. submit button
        a. value attribute defines the text within the button
        b. name attribute defines the element
        c. click action calls the action script
    8. reset button
        a. value attribute defines the text within the button
        b. click action resets all the form information
    9. label
        a. text, radio, checkbox form elements need to have a descriptive name on the display form
        b. <label for= “fname”> First name</label> <input type= “text”>

vii. progress – shows a progress bar

viii. meter – shows a bar with a colour range fill
ix. fieldset
    1. define a visual grouping of form elements
    2. <fieldset> <legend>Name of group</legend> … </fieldset>

n. deprecated tags and attributes
    i. HTML tags and attributes no longer meant to be used, e.g.
    ii. Replaced with CSS definitions

 replaced by CSS text-align:center
<u> replaced by CSS text-decoration:underline
<font color= "red"> replaced by color:red
<img src=a.bmp width=100px> replaced by style= "margin:100px"

o. anchor
    i. x
    ii. URL
        1. uniform resource locator, a unique name for a web resource
    iii. image
        1. use image as an anchor
            <img src= "a.png">
    iv. absolute URL
        1. defines a fully qualified URL
        2. scheme (http, https, ftp, sftp)
        3. webserver (www)
        4. subdomain (server)
        5. domain (com)
        6. webserver, subdomain, domain can be replaced with the server's IP address (204.174.65.19)
           http://www.server.com/data/index.html

    v. relative
        1. defines the location of the resource relative to the current document
        2. allows for easier relocation of the HTML files and folder to another server or folder
        3. Example:
           href= "work/dd.html" means a link to the subfolder work and its file dd.html
           href= "../dd.html" means a link to the parent folder and its file dd.html
           href= "/work/dd.html" means a link to the web server root folder work and its file dd.html
           href= "./dd.html" means a link in the current folder to file dd.html
    vi. name

1. a name attribute means a named anchor within the document, use the hash character #

 … - defines a named anchor

… … - links to the above anchor

2. CSS
    a. Definitions
        i. cascading style sheet
        ii. define the document styles in a separate, reusable document
        iii. syntax – selector { attribute : value; }
        iv. comments in CSS use /* … */ structure
    b. properties
        i. inheritance
            1. CSS styles defined in a parent element apply to its child elements. E.g. a colour text style defined for a <h1> element will apply to the contained tags.
    c. selectors
        i. group together
            1. If the selectors share the same style, they can be listed together separated by commas

            h1, h2, p {color: green; font-family: “Times New Roman”; }
        ii. class
            1. If a style is to be shared among some selectors, a class can be defined
            2. class name must be a single word not starting with a number
            3. has a lower precedence than ID named selectors

            .red {color:red;}
            h1.blue {color:blue;}

             appears in red
             appears in black
            <h1 class= “red”> header in red
            <h1 class= “blue”> header in blue </h1>
        iii. ID
            1. If a style is to be defined for a single instance of a tag, an ID can be defined
            2. ID name must be single word not starting with a number
            3. has a higher precedence than class named selectors

            #green {color:green;}
            div#large {font-size:20pt;}

 this text is green
 large text

iv. Context dependent
    1. descendant
        a. a style that is used only when an element is contained within another – a descendant
        b. separate the selectors by a space

        p b { color:blue; }

         abc   show as blue show as black

    2. child
        a. a style that is used when an element is a child of another

        div > p {color: yellow; }

          show as yellow show as black

    3. sibling
        a. a style that is used when two elements are adjacent

        h1 + h2 { margin-bottom: 5mm; }

        <h1> abc </h1> add a margin bottom of 5mm
        <h2> xyz </h2>

d. Rules
    i. inline
        1. span tag
        2. define a CSS style within a line of text to be used just once

        blue text

    ii. embedded
        1. style tag
        2. define one or more CSS styles used only within this HTML document – usually defined in the <head> section

        <style> CSS definitions </style>

    iii. external
        1. link tag
        2. define one or more CSS styles which are to be shared among other HTML documents
        3. more than one css file can be linked into a document

<link href= “default.css” type= “text/css” rel= “stylesheet”>
(attributes may be defined in any order)

iv. precedence
    1. users may define their own custom stylesheet for the browser – these styles have lower precedence than the styles defined by the style author

    2. inheritance
        a. many CSS properties are inherited from the parent element to the child element (e.g. font, color are inherited but not border or margin properties)

    3. cascade
        a. styles defined in different locations (embedded, external) for the same selectors will cascade – but external styles have lower precedence than embedded styles which have lower precedence than inline styles
    4. specificity
        a. those selectors without class name or ID name have lowest precedence; selectors with class name have next higher precedence; selectors with ID name have next higher; styles having !important have highest precedence
        b. a style defined for a descendant element has **precedence**

```
p  {color:blue;}
div p { color:red;}

<div>header
<p> this is red</p>
</div>
```

e. link
    i. pseudo-class uses the colon after the selector

```
a:link    { color: red }
a:visited { color: green; font-size: 85% }
a:active  { color: blue; font-size: 125% }
a:hover   { color: black; font-size: 85% }
```

f. colours
    i. color
        1. for changing the text colour
        2. defined using one of:

a. name, there are 140 names approved by W3C including aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow

ii. background-color

iii. background-image

g. font

i. font-family types – serif, sans-serif, monospace, cursive, fantasy

ii. font-style – default is normal; can be italic or oblique

iii. font-weight – how thick or thin the characters appear

1. default is 400, bold is 700

iv. font-size – how large the font is, default is 100% em

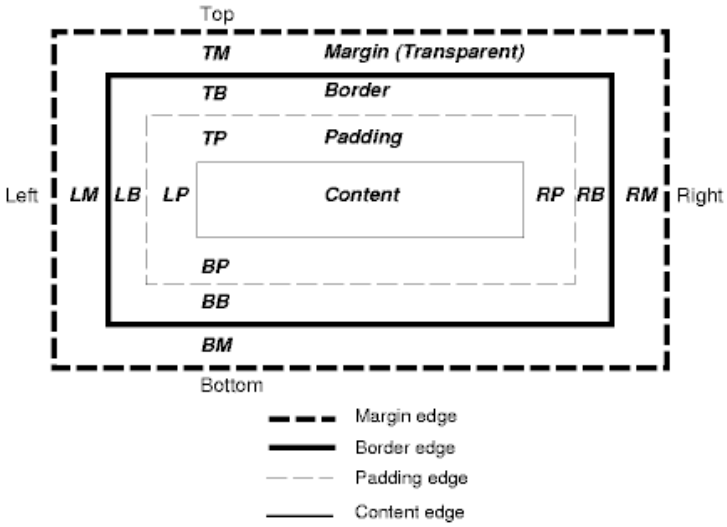
1. relative (em, ex, px)
2. absolute (in, cm, mm, pt, pc)

h. box model

i. width

ii. height

iii. "TRouBLe" – order of properties (top, bottom, left, right)



i. positioning

i. float

1. sets where an image or text will appear in another element
2. float:left means the element moves to the left in the parent element
3. float:right means the element moves to the right in the parent element

ii. absolute

1. unit types are fixed regardless of monitor size – used for printing
2. in, cm, mm, pt (point, 72 points=1 inch), pc (pica, 1 pc=12pts)

iii. relative

1. unit types adapt to different monitor display types and sizes
2. em, ex (rarely used), px

iv. clear
1. sets the sides of an element where other floating elements are not allowed
2. clear:both means no elements to appear on either side of me

v. fixed
1. indicates the element is to stay on the screen and not scrolled

3. Javascript
   a. Origin
      i. Originally called "LiveScript" as a scripting language by Netscape
      ii. A "worm" malware can be written in Javascript – security issues
      iii. Is not the same as Java but shares object-oriented features

   b. Purpose
      i. Javascript provides dynamic response within a web page
      ii. Javascript allows the web page to respond to user actions

   c. Best used when
      i. Defining form elements; define an onclick attribute for radio buttons, checkbox buttons, submit and reset input types; define an onselect attribute for select elements
      ii. Button actions
      iii. Require date and time functionality

   d. Key features
      i. Source files typically end in .js
      ii. The <script> tag defines Javascript content
      iii. variables are declared using keyword var
      iv. comments use // (single line) or /* … */ structure
      v. statements may end with semicolon
      vi. DOM – Document Object Model
         1. Defines the HTML as a tree with <html> tag as the root and other elements as branches and subbranches
         2. Can use Javascript to inspect and change DOM elements dynamically
         3. Notation example: document.getElementById("element_name").property, document.getElementById("element_name").method or document.getElementsByName("name_attribute_of_elements")

4. jQuery

a. Purpose
    i. A multipurpose JavaScript library used to simplify DOM manipulation
b. Key features
    i. Notation is $(“ id_of_element”).objects
    ii. How do you create a DOM object as opposed to the standard jQuery object?

5. HCI
    a. Definition
    b. Six goals of HCI – safety, utility, effectiveness, efficiency, usability, appeal
    c. UI difficulties
        i. Some UI are designed poorly.. why? .. what are consequences?
        ii. Challenges facing UI designers
    d. Cognition: ESP, Discrimination, Forgetting, Gestalt Psychology
    e. Fitts’ Law
    f. Hicks’ Law
    g. Schneiderman’s eight golden rules
        i. Strive for consistency
        ii. Cater to universal usability
        iii. Offer informative feedback
        iv. Design dialogs to yield closure
        v. Offer error prevention and simple error handling
        vi. Permit easy reversal of actions
        vii. Support internal locus of control
        viii. Reduce short-term memory load

    h. Aspects of good UI design
        i. Affordance
        ii. Forcing function
        iii. Mapping
        iv. Conceptual model

6. XML
    a. Used as a data transport mechanism – especially for e-business usage
    b. Open source – non-proprietary
    c. Multiplatform support – multi-application support
    d. Human-readable
    e. Format:
```
<?xml version="1.0"?>
<root element>
  <child1>
  </child1>
    …
  <childN>
```

</childN>
</root element>

f. DTD – Document Type Definition
    i. Defines the rules which will the XML file will use to define its content (what elements are needed, the order of elements)
    ii. PCDATA means parsed character data – text found between the start and end tags – tags inside the text will be treated as markup and entities will be expanded.
    iii. Required/Implied/Default
g. Properties
    i. Well-formed
        1. XML is well-formed if its elements are syntactically correct (no missing tags)
    ii. Valid
        1. XML is valid if its elements do not break any of the DTD rules

h. XSLT
    i. Used to define formatting for the XML and define any selection of XML data elements in the browser
    ii. Uses XPath functions