

L2: C1P2

Friday, 8 March, 2019 3:06 PM

Objectives

1. Create navigation list
2. Link to files within website with hypertext link
3. Absolute & relative folder path
4. Link to email-address & telephone numbers
5. Mark content using lists
6. Define list styles
7. Apply text & font styles

Working with Hypertext Links

1. **Hypertext:** Text enclosed with <a>, anchor tags
 - a. Syntax: content
2. **Inline images:** Turn into links by enclosing **img** inside "a" tags
3. **Link destination:** The destination a hypertext link is pointing
 - a. Can point to:
 - i. Another section in same document
 - ii. Different document
 - iii. Different web pages
 - iv. Web objects
4. **Avoid special characters, keep filenames short**

Specifying folder path

1. Specify file location/path
2. **Absolute path:** Precise location of file in computer's entire folder structure
3. **Relative path:** File location in relative to current document's location
 - a. Handling:
 - i. **Same location:** Only file name
 - ii. **In Subfolder:** Include subfolder name
 - iii. **Above:** Use double period (..)
 - iv. **Sibling folder** (different folder, same level): Move up then down

Link to location in same document

1. Mark location by giving ID (anchor)
 - a. **ID attribute:** Unique, non-case sensitive way to identify elements in HTML document
 - i. Example: <h1 id="A">A</h1>
2. **Create a link**
 - a. **Href** to an anchor (note to "select" ID's, you indicate it with a "#" in front)
 - i. Eg: A
 - b. Can also **href** text

Link to location in different documents

1. **Using anchor tag, indicate destination/reference in front of "#"**
 - a. Example: content

Working with Image Maps

1. **Hotspots**
 - a. Meaning: Defined area, acts as a hyperlink
 - b. Zones within an image that links to different destination

- c. Types:
 - i. **Server-side image maps**
 - ii. **Client-side image maps**

2. Client-side image map

- a. Insert into image map into HTML file
- b. Locally process image map
- c. Pros:
 - i. Can easily test websites, all processing local
 - ii. More responsive than server-side maps
- d. Browser status bar display target of hotspot
- e. Older browser don't support

3. Working with image maps

- a. Create image map that matches a region of inline image to destination
- b. Syntax:
 - i. `<map name="map" id="map">`
 - 1) Hotspots
 - ii. `</map>`
- c. Map = name of map
- d. Hotspots = location of hotspots within map
 - i. Defined using 3 properties:
 - 1) Shape
 - 2) Location
 - 3) Destination (href)
 - ii. Syntax:
 - 1) `<area shape="shape" coords="coordinates" href="url" alt="text" />`
 - 2) Shape="rect"/"circle"/"polygon"
 - iii. Location:
 - 1) **Rectangular: 2 points**
 - a) Upper-left corner
 - b) Lower-right corner
 - c) Ex: coords="168, 110, 225, 145"
 - 2) **Circular: Center + radius**
 - a) Ex: coords="82, 78, 80"
 - 3) **Polygon: Coordinates for every vertex**
 - a) Ex: coords="13,60,13,270,370,..."
 - 4) **Default: 0, 0, x y** (0,0 represent the starting pixel of the image, the image stretches x pixels to the right and y pixels down)
 - a) Ex: coords="0,0,x,y"
 - i) X = width of inline image
 - ii) Y = image height
 - b) Any spot not covered by another hotspot will activate default link

4. Server-side image map

- a. Image map stored on server
 - i. Supported by most browsers
 - ii. Can be slow
 - iii. Browser status bar does not display target

Linking to Internet & Other resources

1. Scheme:location
 - a. Scheme = resource type
 - b. Location = resource
2. **Protocol:** Rule defining how information is passed between devices
3. **Linking to web resource:**

a. <http://server/path/filename#id>

4. **Email:**

a. **mailto:** address

5. **Phone:**

a. **tel:** phone