Announcements

- Class Web Site:
  - http://www.cs.umd.edu/projects/passport/Classes/Spring2008/
  - You can find this link at the end of the main passport site
    - http://www.cs.umd.edu/projects/passport/webPage/
- Envelops!
- Certificates
- Project #4

Computer Science

- What is Computer Science?
- What areas can we identify in computer science?
- Is a programmer a computer scientist?
- How can you become a computer science major?

Favicon

- Icon that appears next to the websites' URL in the address bar.
- Example: http://www.umd.edu/
- You also see them when you bookmark a page.
- The icon is represented by a file named favicon.ico
- In the resources section of the class web page you can find a link to a favicon generator.
- To place the favicon in your page insert the following html tag in the head section of your html file

```
<link rel="shortcut icon" href="favicon.ico" />
```

- **Example:** FaviconExample.html
- **Link:**

http://www.cs.umd.edu/~nelson/WebExamples/FaviconExample/FaviconExample.html

Round Corners

- You can achieve round corners by using <img> and setting the background of a div area.
- You can generate the corners using a image editing software (e.g., Photoshop) or by using some online generators (see Resources section for some links)
- **Example:** RoundedCorners/roundCorners.html
- Currently Mozilla-based browsers support a property for generating rounded corners. This will be part of CSS3.
- **Example:** MozillaRoundedCorners/MozillaRoundedCorners.html

Multimedia

- Several video file formats. For example:
  - Macromedia Flash
  - MPEG (MPEG-4)
  - QuickTime
  - Windows Media
  - Others
- Each has its advantages and disadvantages
- Two main approaches to access video, audio, animation:
  - **File Downloading** (HTTP streaming/HTTP delivery)
  - **Streaming**

Multimedia

- **File Downloading**
  - Entire file must be downloaded to your computer before starting to viewing the media (video, audio, animation, etc.). You have to wait before viewing the media.
  - This approach is relatively easy and cheap to implement via either:
    - hyperlink to the file. When the user selects the link the appropriate player will start processing the media file
    - Embedding – media appears in a portion of the web page rather than in a separate application.
- **Streaming**
  - File is sent in a stream and user can start viewing the media almost immediately.
  - Streaming allows for broadcasting of live events (webcasting).
  - Streaming services can be expensive.
  - **Example:**
  - http://webclust1.liquidcompass.cc/sos4cust/WTOP/audio_player.php?id=WTOP
- **Progressing Downloading**
  - File is downloaded but the user can start viewing the media once a portion of the file has been downloaded

Multimedia

- Due to its cost and simplicity file downloading (HTTP Streaming Video) is the option you may frequently use.
- To deploy HTTP Streaming video:
  - Create a media file.
  - Upload the file to the web server.
  - Make a hyperlink or embed the media in your page.
- In order to embed media we use the html object tag. Each media file (e.g., quicktime, mpeg, etc.) follows a particular configuration illustrated in the following examples.
- Important: In firefox you must make sure you have the appropriate plugin (https://addons.mozilla.org/en-US/firefox/browse/type:7)
- For our examples, we needed to install the windows media plugin
- **Example:** EmbeddingAVI.html
- **Example:** EmbeddingWMV.html
- **Example:** EmbeddingQuicktimeV1(V2).html
- **Example:** EmbeddingFlash.html

Placing Your Videos on the Web

- Transfer the video your computer.
  - Video software mentioned below can help you in this process (“Capturing Video”).
- Edit video if necessary. Example of video editing software:
  - PowerDirector
  - ShowBiz DVD
  - Ulead VideoStudio
  - Nero Ultra Edition
- Export video in a web-compatible format like Quicktime, Windows Media, etc.
- Place the video in your web site.

Forms

- **Forms** - means by which information passes from the user to a server.
- For now we will use forms to read values to be processed by our JavaScript programs.
- **<form> tag**
  - Defines the form.
  - It has two attributes: action and method
  - **action** – indicates where the form contents will be sent when the form is submitted.
  - **method** – defines how the contents will be sent (post/get).
- **<input>tag**
  - Appears inside of the <form> tag.
  - Defines several input data alternatives.
  - The general format is: **<input type="ALTERNATIVE" />**
  - **ALTERNATIVE** can be text, password, checkbox, radio, file, submit, image, button, reset, hidden
- We have can several forms in our document.
- **Example:** Forms.html

Timer

- setInverval
  - Allow us to schedule an activity at a particular interval
- **Example:** TestudoAnimation.html

Generation of Random Values

- Example: RandomValues.html

Events

- **Event –** Notification that something has occurred.
- Example of situations that make the web browser generate an event.
  - Browser finishes loading a document.
  - When the user clicks on a button.
  - When the user moves the mouse.
  - Others.
- **Event handler** (also known as event listener)
  - JavaScript function or code fragment that is executed when a particular event occurs.
- **Event handler registration**
  - Associating an event handler with a particular event.
- **Example:** EventEx.html

Event-driven Programming

- **Normal (control flow-based) programming**
  - Approach
    - Start at main().
    - Continue until end of program or exit().
- **Event-driven programming**
  - Start at main().
  - Register event handlers.
  - Await events & perform associated computation.
- **GUIs (Graphical User Interfaces)**
  - Example of event-driven software.

Event Handler Attributes for most HTML

- ***Mouse Related***
  - **onclick** – mouse button is pressed and released.
  - **ondblclick** – mouse button is double-click over element.
  - **onmouseover** – mouse moves over element.
  - **onmouseout** – mouse moves off element.
  - **onmousemove** – mouse pointer is moved.
  - **onmousedown** – mouse is pressed down while cursor is over the element.
  - **onmouseup** – mouse is released while the cursor is over the element.
- ***Keyboard Related***
  - **onkeypress** – key pressed and release.
  - **onkeydown** – key is pressed.
  - **onkeyup** – key is released.
- ***Other***
  - Keep in mind that there additional handlers that are specific to certain tags. We will address those later on.