Introduction

• Frames
• HTML Forms
• CGI Scripts
• Dynamic Documents
• HTML Tools
• Next-Generation HTML
Frames

- Frame Usage
- Frame Tags
- Frequently Asked Questions
- Additional Frame Tag Attributes
- Targeted Hyperlinks
- Frameset Design Algorithm
Advanced HTML
Frame Usage

- Common content
- Form verification
- Table of contents
- Search engines
Frame Tags

- `<FRAMESET>`
- `<FRAME>`
- Attributes
  - `COLS`
  - `ROWS`
  - `NAME`
  - `SRC`
- `<NOFRAMES>`

From Greenlaw/Hepp, *In-line/On-line: Fundamentals of the Internet and the World Wide Web*
Frequently Asked Questions

• Can COLS and ROWS be used in conjunction with one another? If so, what is the result?
• How are the source files loaded into the frames?
• If each document in a frameset has a title, which title is displayed in the lone title bar when the group of frames is being displayed?
• How can the HTML source code be viewed for a document created using the frameset tag?
• How can the source code of individual frames be viewed?
Frequently Asked Questions

• Is it possible to load a document that is itself a document consisting of a frameset declaration into the SRC attribute of a frame tag?

• Can a hyperlink be targeted into a specific frame?

• Can a new browser be launched automatically, if you do not wish to load a large page into a small frame?

• Can frames be nested?

• Is there an easy way to produce any frame layout?
Advanced HTML

Frames

Additional Frame Tag Attributes

- NORESIZE
- SCROLLING
- MARGINHEIGHT
- MARGINWIDTH
- FRAMEBORDER
Targeted Hyperlinks

- TARGET attribute
  - Of <A> tag
  - Of <BASE> tag
- “Special targets”
  - _blank
  - _top
My Frame Example – use of FRAMESET

```html
<HTML>
<HEAD>
<TITLE>Savannah Area Gator Club</TITLE>
</HEAD>
<FRAMESET ROWS="20%,*">
  <FRAME SCROLLING=NO NAME="gcframe1" SRC="gator01.htm">
  <FRAMESET COLS="20%,*">
    <FRAME NAME="gcframe2" SRC="gator02.htm">
    <FRAME NAME="gcframe3" SRC="gator03.htm">
  </FRAMESET>
</FRAMESET>
</FRAMESET>
</HTML>
```
Advanced HTML

My Frame Example – Navigational FRAME2

<HTML>
<TITLE>"gator02"</TITLE>
<BODY BGCOLOR="orange">
<li><A TARGET="gcframe3" HREF="gator03.htm">Home Page</A>
<li><A TARGET="gcframe3" HREF="mission.htm">Mission and Officers</A>
<li><A TARGET="gcframe3" HREF="programs.htm">Programs & Ticket Exchange</A>
<li><img src="http://pic.geocities.com/clipart/s/t/Icnrylbluest.gif"><A TARGET="gcframe3" HREF="pictures.htm">Club Pictures</A>
<li><A TARGET="gcframe3" HREF="otherlinks.htm">Other Gator Sites</A>
<li><A TARGET="gcframe3" HREF="schedule.htm">2002 Football Schedule</A>
<li><A TARGET="gcframe3" HREF="membership.htm">Membership</A>
<li><img src="http://pic.geocities.com/clipart/s/t/Icnrylbluest.gif"><A TARGET="gcframe3" HREF="memberapply.htm">Apply or Renew Your Membership</A>
<li><A TARGET="gcframe3" HREF="scholarship.htm">Gunn Memorial Scholarship</A>
<li><A TARGET="gcframe3" HREF="ufnews.htm">UF Academic News</A>
<li><A TARGET="gcframe3" HREF="ufsports.htm">UF Sports News & Basic NCAA Regs Info</A>
<li><A HREF="mailto:fkgator@yahoo.com">E-mail us!!</A>
</BODY>
</HTML>
Advanced HTML

The Savannah Area Gator Club On-Line

UNIVERSITY OF FLORIDA GATORS®


Welcome to THE SWAMP, where only the GATORS® get out alive

From Greenlaw/Hepp, In-line/On-line: Fundamentals of the Internet and the World Wide Web
<HTML>

<HEAD>
<TITLE>Nesting Frames</TITLE>
</HEAD>

<FRAMESET ROWS = “50%,50%”>
   <FRAMESET COLS = “33%,67%”>
      <FRAME SRC = “left-third.html”
          NAME = “left-third”>
      <FRAME SRC = “right-twothirds.html”
          NAME = “right-twothirds”>
   </FRAMESET>
</FRAMESET>

<FRAME SRC = “bottom-row.html” NAME = “bottom-row”>
</FRAMESET>

</HTML>
You have selected a square.
Frameset Design Algorithm

- Storyboard
- Framing algorithm:
  - Choose COLS or ROWS for outermost frameset
  - Repeat for each inner frameset until no subdivisions
- Recursive
Advanced HTML

HTML Forms

• Form Tags
• Form Methods
• Mailto URLs
• Form Input Tags
• Frequently Asked Questions
Form Tags

- `<FORM></FORM>`
- `ACTION`
- `METHOD`
- `Cgibin (Common Gateway Interface) directory`
Methods

- Introduction
  - get
  - post
- Comparison: get and post
- Considerations:
  - Programmer’s expertise
  - Number of input parameters
  - Security
Mailto URLs

- As ACTION attribute
- Other uses of mailto ACTION
- Testing forms
- Learn about parameter encodings
- To make use of forms
- Guest book
Form Input Tags

Basic structure:

```
<FORM
   ACTION = http://www.server.com/cgibin/program
   METHOD = "get"
...
</FORM>
```
Form Input Tags

- `<INPUT>`
- **Text Boxes**
  - Rectangular-shaped
  - User enters text
  - `TYPE = “text”`
  - `NAME`
  - `SIZE`
  - `MAXLENGTH`
  - `VALUE`
Form Input Tags

- Checkboxes
  - Icon user selects/deselects
  - CHECKED

- Radio Buttons
  - Only one can be selected
  - “Yes” and “No”
  - Buttons with same NAME
Form Input Tags

Action Buttons

• TYPE
  • Submit
  • Reset

• VALUE
  • “SUBMIT” or “SEND”
  • “RESET” or “CLEAR”
Form Input Tags

Selects

- `<SELECT></SELECT>`
- `MULTIPLE`
- `NAME`
- `SIZE`
- `<OPTION>`
  - `SELECTED`
  - `VALUE`
Form Input Tags

Text Areas

- Multirow text field
- `<TEXTAREA></TEXTAREA>`
- ROWS
- COLS
- WRAP
Form Input Tags

Password

<INPUT TYPE = “password”
NAME = “nn”
SIZE = “40”
MAXLENGTH = “20”>

- Masked field
Code Used to Create Form - 1

```html
<html>
<head>
<title>Savannah Gator Club Membership Application</title>
</head>
<body bgcolor="#FF8C00" text="#0000FF">
<form method="post" action="http://geocities.yahoo.com/forms">
<font face="arial, helvetica">
<h3>Savannah Gator Club® Membership Application</h3>
<img src="irongator.jpg" alt="Almost the real thing">
<TABLE CELLPADDING="5">
<tr><td><b>First Name:</b> <td><input type="text" name="First" size="27">
<tr><td><b>Last Name:</b> <td><input type="text" name="Last" size="20">
<tr><td><b>Spouse Name:</b> <td><input type="text" name="Spouse" size="20">
<tr><td><b>Address:</b> <td><input type="text" name="Address" size="30">
<tr><td><b>City:</b> <td><input type="text" name="City" size="20" value="Savannah">
<tr><td><b>State:</b> <td><input type="text" name="State" size="4" value="GA">
<tr><td><b>Zip:</b> <td><input type="text" name="Zip" size="5">
<tr><td><b>Home Phone:</b> <td><input type="text" name="Hphone" size="12">
</form>
</body>
</html>
```
Advanced HTML

Code Used to Create Form - 2

<tr>
<td><b>Work Phone:</b></td>
<td><input type="text" name="Wphone" size="12"></td>
</tr>
<tr>
<td><b>Cell Phone:</b></td>
<td><input type="text" name="Cphone" size="12"></td>
</tr>
<tr>
<td><b>E-mail Address:</b></td>
<td><input type="text" name="Email" size="35"></td>
</tr>
<tr>
<td><b>Florida Grad:</b></td>
<td><input type="checkbox" name="Grad" value="UFGrad" checked></td>
</tr>
<tr>
<td><b>Year, Degree, College:</b></td>
<td><input type="text" name="YrDegColl" size="40"></td>
</tr>
</TABLE>
<p><br>
<input type="submit" value="Send">
<input type="reset" value="Clear the form">
<input type="hidden" name="login" value="fkgator">
<!-- Option 1 -->
<input type="hidden" name="subject" value="Survey Results">
<!-- Option 2 -->
<input type="hidden" name="required_fields" value="First,Last,Address,City,State,Zip,Hphone,Cphone,Email">
<!-- Option 3
<input type="hidden" name="next_url" value="http://www.geocities.com/fkgator/index.html">
-->
</form>
Result of Form Submission

First = Frank
Last = Katz
Spouse = Jan
Address = 229 Groveland Circle
City = Savannah
State = GA
Zip = 31405
Hphone = 912-555-1212
Wphone = 912-921-5608
Cphone = 912-441-0304
Email = katzfr@drake.armstrong.edu
Grad = UFG Grad
YrDegColl = 1977, BA, CISE
login = fkgator
subject = Survey Results
REMOTE_HOST: 130.254.204.161
Frequently Asked Questions

• Can a user load a new Web page or an image by pressing a “submit” button on a form?

• There is a button that looks like a submit button, but it says something else. Is this accomplished with forms? How?

• Is there a way to use a form to create a navigational button such that when a user submits the form, I can have another page loaded into a specific frame?

• Can you name a form?
CGI Scripts

- Scripts and Forms
- Security
Scripts and Forms

- After user submits form
- Server receives request
- Server runs script
- Inputs to script are values from form
Dynamic Documents

- Client Pull
- Server Push
- New-Address Notification
Client Pull

Browser (client) initiates document’s change

• Demand on Internet
• HTTP-EQUIV
• CONTENT
• \(<\text{META HTTP-EQUIV} = \text{“refresh” CONTENT = “n”}>\)
• Reloading different document
• Example:
  \(<\text{META HTTP-EQUIV} = \text{“refresh”}
  \text{CONTENT = “10;}
  \text{URL=\text{http://www.mycomputer.edu/\~you/data.html”}>\)
Server Push

• Server initiates the document’s change
• Demand on server
• Java or Perl

New-Address Notification

Important use of client pull
HTML Tools

• Editors
• Syntax Checkers
• Converters
Editors

• Software tool

• Two types:
  • WYSIWYG (“what you see is what you get”)
  • Tag-oriented
Editors

Editor Components

- Color selector
- Documentation
- HTML converter
- Hyperlink inserter
- Spell checker
- Tag generator
- Template creator
- Tutorial
- Viewer
Editors

Editor Usage

• Generate draft
• Develop pages expected to be around for while
• Develop consistent set of pages
Editors

Editor Drawbacks

- Featuritis
- Incomplete support
- Poor error reporting
- Programming style
- User interface
Editors

Editor Availability

- Frontpage, Microsoft
- HotDog Pro, Sausage Software
- HoTMetaL, SoftQuad
- Navigator Gold, Netscape
- SiteMill, Adobe
Syntax Checkers

- Code Analysis
- Error report
- Feature subsetting
- Image analysis
- Link verification
- Structure report
- Making suggestions
- Load timing
Syntax Checkers

- Syntax Checker Availability
  - Doctor HTML
  - Dr. Watson
  - Weblint

Converters

- Converts a document into HTML format
- On-line resume
Next-Generation HTML

• Extensible markup language (XML)
• Cascading style sheets (CSS)
CSS – In-Line Styles

- Allows you to change behavior of an HTML tag to achieve a desired appearance.
- Change affected is a “local” one, that is, applies only to the *single* tag where the in-line style is included.
- For example:
  - `<H5 STYLE = “font-style:italic”>In-Line/On-Line: Cascading Style Sheets</H5>`
  - STYLE *attribute* is used
  - Render this heading in Italics
- Can combine several style properties into one in-line tag:
  - `<H5 STYLE = “background:gray;text-transform:uppercase;text-align:center;font-style:italic”>In-Line/On-Line: Cascading Style Sheets</H5>`
  - Produces a heading with a gray background, in all uppercase, has text centered, and appears in italics.
- Netscape and IE render the STYLEs slightly different
- Note that a full colon separates the element from it's characteristics, and a semi-colon separates modifications to multiple style elements
CSS – Internal Styles

• Apply the style “globally” rather than “locally,” in this case, throughout the document
• Must use the <STYLE> </STYLE> tags, and they MUST be included in the HEAD portion of an HTML document (so now you know one thing the HEAD is good for)
• Syntax is to include the modifications within braces { }
• Example of <STYLE> H3 {background:gray;text-align:center} P {font-style:italic;text-align:justify} </STYLE> yields:
• Every H3 header in the document will have a gray background and be center aligned; every paragraph will be rendered in italics and fully justified
• Note that the syntax – use of colons and semi-colons – is the same as in In-Line CSS
CSS – External Styles

- Apply the style “globally” rather than “locally,” in this case, throughout a collection of documents.
- Must be applied from a FILE, with a .css extension (example on page 426), related to the document in a <LINK> tag – no ending tag, which must also be placed in the HEAD section of an HTML document.
- Syntax is identical to that of Internal Styles, such as use of braces, colons, and semi-colons.
CSS – Advantages and Disadvantages

• Advantages:
  • Collect all style definitions in one place
  • Reduce time for maintaining a set of web pages
  • Test new layouts for a collection of web pages with only a few global edits
  • Create a master layout for a web presentation.

• Disadvantages:
  • They are not supported in a consistent manner by all browsers
  • If you make a syntax error, it can be hard to debug your code