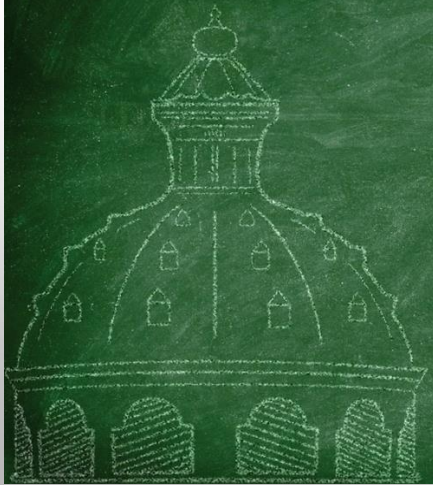
CUSTOMER INTERFACE DESIGN AND DEVELOPMENT

SITA1502



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(Deemed to be University)

Code : SITA1502

Class : CUSTOMER INTERFACE DESIGN AND DEVELOPMENT

Batch : 2019- Btech -IT

Tutor : DR.SENDURU SRINIVASULU

Academic Year 2021-2022

SITA1502	CUSTOMER INTERFACE DESIGN AND DEVELOPMENT	L	T	P	Credits	Total Marks
		3	*	0	3	100

COURSE OBJECTIVES

- To develop static websites and dynamic web applications.
- To learn new emerging web technologies.
- To gain knowledge and skills required for web development careers.
- To develop skills in the use and application of specific methods in user experience design.

UNIT 1 HTML, XML, CSS AND RWD **9 Hrs.**

Introduction To HTML- DHTML , XML – Structuring XML document using DTD – Schemas – XML parsers – DOM – SAX presentation technologies – XSL – XFORMS – XHTML – Transformations – XSLT – XLINK – XPATH – XQuery.
Responsive Web Design-Intro-Fluid Grid-Viewport-Media Queries-Images.
Introduction To CSS-Syntax, Selectors-Types of style sheets.

UNIT 2 CLIENT SIDE SCRIPTING **9 Hrs.**

Java Script – Advantages – Data types – Variables – Operators – Control statements – Functions – Objects and arrays – Windows and frames – Forms. AJAX – XMLHttp Request (XHR) – Create Object – Request – Response – Ready state.

UNIT 3 SERVER SIDE SCRIPTING **9 Hrs.**

Introduction To PHP – Data Types – Control Structures – Arrays - Function – Html Form with PHP –Form Handling and Validation - File Handling – Cookies – Sessions – Filters – Exception Handling - Database Connectivity With MySQL.

UNIT 4 ANGULAR JS AND JQUERY **9 Hrs.**

Angular JS Expression – Modules – Directives – Data Binding – Controllers – Scopes – Filters – Services – Tables – Events – Form – Validation. jQuery Syntax – Selects – Events – jQuery Effects – jQuery – jQuery HTML – jQuery Traversing.

UNIT 5 UX AND UI **9 Hrs.**

UX Introduction -Elements of UX Design- UX Design Process- Research Methods and Tools-Understanding User Needs and Goals. UX Design Process: Visual Design Principles-Information Design and Visualization-Interaction Design-Prototyping Tools-Usability Test. UI Introduction-User Interface Components -Tools and Processes.

Max.45 Hrs.

COURSE OUTCOMES

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Able to work with XML technologies.
CO2 - Design web page to perform form validation using client-side scripting language.
CO3 - Implement new technologies such as Angular JS & jQuery.
CO4 - Develop web applications using server-side scripting language.
CO5 - Understand the differences between usability and user experience.
CO6 - Effectively select and utilize design thinking processes and UX/UI tools.

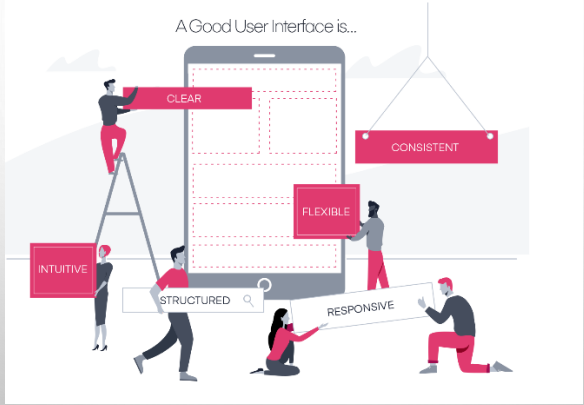
TEXT/ REFERENCE BOOKS

1. Jeffrey C. Jackson, Web Technologies: A Computer Science Perspective, Pearson Education, 2009
2. Kogent Learning Solutions Inc., Web Technologies Black Book, Dreamtech Press, 2009.
3. Ken Williamson, Learning AngularJ S: A Guide to Angular JS Development, O'Reilly, 2015
4. Jon Duckett, JavaScript and JQuery: Interactive Front-End Web Development, John Wiley and Sons Inc., 2014.
5. Callum Macrae, Learning from JQuery, O'Reilly, 2013.
6. Steve Krug, Dont Make Me Think,2nd Edition, New Riders Publishing,USA,2006.

END SEMESTER EXAMINATION QUESTION PAPER PATTERN

Max. Marks : 100	**Exam Duration : 3 Hrs.**
PART A : 10 Questions of 2 marks each-No choice	**20 Marks**
PART B : 2 Questions from each unit with internal choice, each carrying 16 marks	**80 Marks**

INTRODUCTION TO HTML


A Good User Interface is...
CLEAR
CONSISTENT
FLEXIBLE
INTUITIVE
STRUCTURED
RESPONSIVE

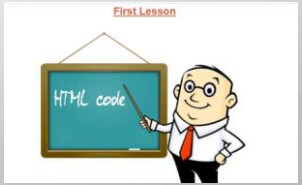
UNIT 1 --HTML, XML, CSS AND RWD

Introduction To HTML
- DHTML , XML
- Structuring XML document using DTD
- Schemas
- XML parsers
- DOM
- SAX presentation technologies
- XSL
- XFORMS
- XHTML
- Transformations
- XSLT
- XLINK
- XPATH
- XQuery.

Responsive Web Design-Intro-Fluid Grid-Viewport-Media Queries Images.
Introduction To CSS-Syntax, Selectors-Types of style sheets.

INTRODUCTION TO HTML

- HTML IS THE STANDARD MARKUP LANGUAGE FOR WEB PAGES.
- WITH HTML YOU CAN CREATE YOUR OWN WEBSITE.
- HTML IS EASY TO LEARN - YOU WILL ENJOY IT!



HTML INTRODUCTION

- **HTML IS THE STANDARD MARKUP LANGUAGE FOR CREATING WEB PAGES.**

WHAT IS HTML?

- HTML STANDS FOR HYPER TEXT MARKUP LANGUAGE
- HTML IS THE STANDARD MARKUP LANGUAGE FOR CREATING WEB PAGES
- HTML DESCRIBES THE STRUCTURE OF A WEB PAGE
- HTML CONSISTS OF A SERIES OF ELEMENTS
- HTML ELEMENTS TELL THE BROWSER HOW TO DISPLAY THE CONTENT
- HTML ELEMENTS LABEL PIECES OF CONTENT SUCH AS "THIS IS A HEADING", "THIS IS A PARAGRAPH", "THIS IS A LINK", ETC.

WHAT IS AN HTML ELEMENT?

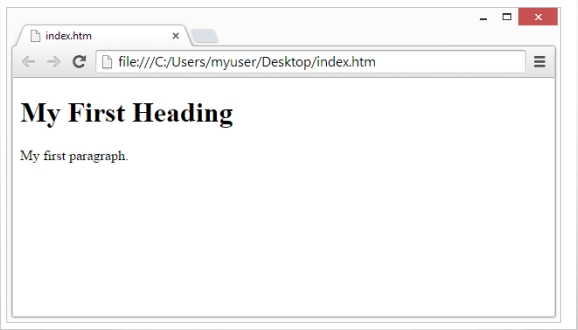
- AN HTML ELEMENT IS DEFINED BY A START TAG, SOME CONTENT, AND AN END TAG:
- <TAGNAME>CONTENT GOES HERE...</TAGNAME>
- THE HTML **ELEMENT** IS EVERYTHING FROM THE START TAG TO THE END TAG:
- <H1>MY FIRST HEADING</H1>
- <P>MY FIRST PARAGRAPH.</P>

Start tag	Element content	End tag
<h1>	My First Heading	</h1>
	My first paragraph.	
 	none	*none*

WEB BROWSERS

- THE PURPOSE OF A WEB BROWSER (CHROME, EDGE, FIREFOX, SAFARI) IS TO READ HTML DOCUMENTS AND DISPLAY THEM CORRECTLY.
- A BROWSER DOES NOT DISPLAY THE HTML TAGS, BUT USES THEM TO DETERMINE HOW TO DISPLAY THE DOCUMENT:

WEB BROWSERS



HTML PAGE STRUCTURE

Below is a visualization of an HTML page structure:

```
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

HTML HISTORY

HTML History

Since the early days of the World Wide Web, there have been many versions of HTML:

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2
1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
2016	W3C Candidate Recommendation: HTML 5.1
2017	W3C Recommendation: HTML5.1 2nd Edition
2017	W3C Recommendation: HTML5.2

DEFINITIONS

- W W W – WORLD WIDE WEB.
- HTML – **HYPERTEXT MARKUP LANGUAGE** – THE LANGUAGE OF WEB PAGES ON THE WORLD WIDE WEB.

  HTML IS A TEXT FORMATTING LANGUAGE.
- URL – UNIFORM RESOURCE LOCATOR.
- BROWSER – A SOFTWARE PROGRAM WHICH IS USED TO SHOW WEB PAGES.

- "NORMAL TEXT" SURROUNDED BY BRACKETED *TAGS* THAT TELL BROWSERS HOW TO DISPLAY WEB PAGES
- PAGES END WITH ".HTM" OR ".HTML"
- HTML EDITOR – A WORD PROCESSOR THAT HAS BEEN SPECIALIZED TO MAKE THE WRITING OF HTML DOCUMENTS MORE EFFORTLESS.

TAGS

- CODES ENCLOSED IN BRACKETS
- USUALLY PAIRED

  <TITLE>MY WEB PAGE</TITLE>
- ***NOT*** CASE SENSITIVE

  <TITLE> = <TITLE> = <TITLE>

CHOOSING TEXT EDITOR

- THERE ARE MANY DIFFERENT PROGRAMS THAT YOU CAN USE TO CREATE WEB DOCUMENTS.
- HTML EDITORS ENABLE USERS TO CREATE DOCUMENTS QUICKLY AND EASILY BY PUSHING A FEW BUTTONS. INSTEAD OF ENTERING ALL OF THE HTML CODES BY HAND.
- THESE PROGRAMS WILL GENERATE THE HTML SOURCE CODE FOR YOU.

CHOOSING TEXT EDITOR

- HTML EDITORS ARE EXCELLENT TOOLS FOR EXPERIENCED WEB DEVELOPERS; HOWEVER; IT IS IMPORTANT THAT YOU LEARN AND UNDERSTAND THE HTML LANGUAGE SO THAT YOU CAN EDIT CODE AND FIX "BUGS" IN YOUR PAGES.
- FOR THIS COURSE, WE WILL FOCUS ON USING THE STANDARD MICROSOFT WINDOWS TEXT EDITORS, NOTEPAD. WE MAY USE ALSO TEXTPAD.

STARTING NOTEPAD

NOTEPAD IS THE STANDARD TEXT EDITOR THAT COMES WITH THE MICROSOFT WINDOWS OPERATING SYSTEM. TO START NOTEPAD IN WINDOWS 9X OR XP FOLLOW THE STEPS BELLOW:

- CLICK ON THE "START" BUTTON LOCATED ON YOUR WINDOWS TASK BAR.
- CLICK ON "PROGRAMS" AND THEN CLICK ON THE DIRECTORY MENU LABELED "ACCESSORIES".
- LOCATE THE SHORTCUT "NOTEPAD" AND CLICK THE SHORTCUT ONCE.

HTML PAGE CREATION & EDITING

IN THIS CHAPTER YOU WILL LEARN TO CREATE HTML PAGES WITH A STANDARD TEXT EDITOR.

OBJECTIVES

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. CHOOSE A TEXT EDITOR.
2. CREATE A BASIC STARTING DOCUMENT.
3. UNDERSTAND AND SET DOCUMENT PROPERTIES.
4. VIEW YOUR RESULTS IN A BROWSER.

CREATING A BASIC STARTING DOCUMENT

```
<HTML>
<HEAD>
  <TITLE>AL AL-BAYT UNIVERSITY</TITLE>
</HEAD>
<BODY>
  THIS IS WHAT IS DISPLAYED.
</BODY>
</HTML>
```

CREATING A BASIC STARTING DOCUMENT

- THE HEAD OF YOUR DOCUMENT POINT TO ABOVE WINDOW PART. THE TITLE OF YOUR DOCUMENT APPEARS IN THE VERY TOP LINE OF THE USER'S BROWSER. IF THE USER CHOOSES TO "BOOKMARK" YOUR PAGE OR SAVE AS A "FAVORITE"; IT IS THE TITLE THAT IS ADDED TO THE LIST.
- THE TEXT IN YOUR TITLE SHOULD BE AS DESCRIPTIVE AS POSSIBLE BECAUSE THIS IS WHAT MANY SEARCH ENGINES, ON THE INTERNET, USE FOR INDEXING YOUR SITE.

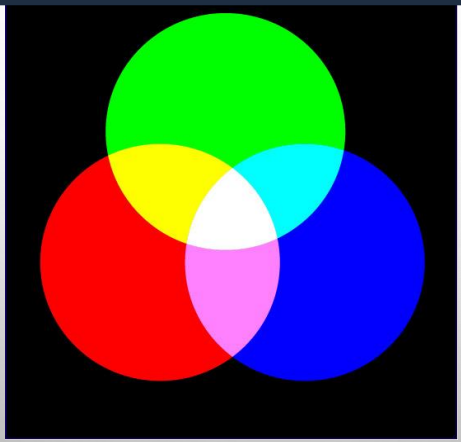
SETTING DOCUMENT PROPERTIES

- DOCUMENT PROPERTIES ARE CONTROLLED BY ATTRIBUTES OF THE BODY ELEMENT. FOR EXAMPLE, THERE ARE COLOR SETTINGS FOR THE BACKGROUND COLOR OF THE PAGE, THE DOCUMENT'S TEXT AND DIFFERENT STATES OF LINKS.

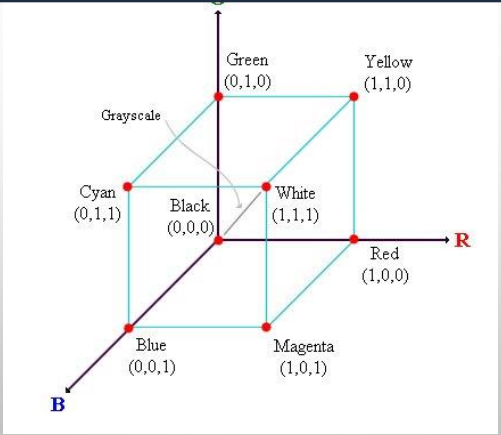
COLOR CODES

- COLORS ARE SET USING "RGB" COLOR CODES, WHICH ARE, REPRESENTED AS HEXADECIMAL VALUES. EACH 2-DIGIT SECTION OF THE CODE REPRESENTS THE AMOUNT, IN SEQUENCE, OF RED, GREEN OR BLUE THAT FORMS THE COLOR. FOR EXAMPLE, A RGB VALUE WITH 00 AS THE FIRST TWO DIGITS HAS NO RED IN THE COLOR.

Main Colours



RGB Colour Model



16 BASIC COLORS

Color Name	RGB Triplet	Hexadecimal	Color Name	RGB Triplet	Hexadecimal
Aqua	(0,255,255)	00FFFF	Navy	(0,0,128)	000080
Black	(0,0,0)	000000	Olive	(128,128,0)	808000
Blue	(0,0,255)	0000FF	Purple	(128,0,128)	800080
Fuchsia	(255,0,255)	FF00FF	Red	(255,0,0)	FF0000
Gray	(128,128,128)	808080	Silver	(192,192,192)	C0C0C0
Green	(0,128,0)	008000	Teal	(0,128,128)	008080
Lime	(0,255,0)	00FF00	White	(255,255,255)	FFFFFF
Maroon	(128,0,0)	800000	Yellow	(255,255,0)	FFFF00

COLOR CODES

1.	WHITE	1.	#FFFFFF
2.	BLACK	2.	#000000
3.	RED	3.	#FF0000
4.	GREEN	4.	#00FF00
5.	BLUE	5.	#0000FF
6.	MAGENTA	6.	#FF00FF
7.	CYAN	7.	#00FFFF
8.	YELLOW	8.	#FFFF00
9.	AQUAMARINE	9.	#70DB93
10.	BAKER'S CHOCOLATE	10.	#5C3317
11.	VIOLET	11.	#9F5F9F
12.	BRASS	12.	#B5A642
13.	COPPER	13.	#B87333
14.	PINK	14.	#FF6EC7
15.	ORANGE	15.	#FF7F00

COLOR CODES

- IF YOU REQUIRE MORE INFORMATION ABOUT COLOR VALUES, THERE IS AN EXCELLENT SITE ENTITLED “VGDESIGN’S INTERACTIVE COLOR CUBE” THAT DISPLAYS THE BACKGROUND COLOR CODE WHEN YOU PUT YOUR CURSOR OVER A SMALL COLOR SAMPLE. THE WEB ADDRESS IS :

  HTTP://WWW.VGDESIGN.COM/COLOR.HTML

THE BODY ELEMENT

- THE BODY ELEMENT OF A WEB PAGE IS AN IMPORTANT ELEMENT IN REGARDS TO THE PAGE'S APPEARANCE. HERE ARE THE ATTRIBUTES OF THE **BODY** TAG TO CONTROL ALL THE LEVELS:

  TEXT="#RRGGBB" TO CHANGE THE COLOR OF **ALL THE TEXT** ON THE PAGE (**FULL PAGE TEXT COLOR**.)

- THIS ELEMENT CONTAINS INFORMATION ABOUT THE PAGE'S BACKGROUND COLOR, THE BACKGROUND IMAGE, AS WELL AS THE TEXT AND LINK COLORS.

BACKGROUND COLOR

- IT IS VERY COMMON TO SEE WEB PAGES WITH THEIR BACKGROUND COLOR SET TO WHITE OR SOME OTHER COLORS.
- TO SET YOUR DOCUMENT'S BACKGROUND COLOR, YOU NEED TO EDIT THE <BODY> ELEMENT BY ADDING THE BGCOLOR ATTRIBUTE. THE FOLLOWING EXAMPLE WILL DISPLAY A DOCUMENT WITH A WHITE BACKGROUND COLOR:

```
<BODY BGCOLOR="#FFFFFF"></BODY>
```

TEXT COLOR

- THE TEXT ATTRIBUTE IS USED TO CONTROL THE COLOR OF ALL THE NORMAL TEXT IN THE DOCUMENT. THE DEFAULT COLOR FOR TEXT IS BLACK. THE TEXT ATTRIBUTE WOULD BE ADDED AS FOLLOWS:

<BODY BGCOLOR="#FFFFFF" TEXT="#FF0000"></BODY>

IN THIS EXAMPLE THE DOCUMENT'S PAGE

COLOR IS WHITE AND THE TEXT WOULD BE RED.

LINK, VLINK, AND ALINK

THESE ATTRIBUTES CONTROL THE COLORS OF THE DIFFERENT LINK STATES:

1. LINK – INITIAL APPEARANCE – DEFAULT = BLUE.

2. VLINK – VISITED LINK – DEFAULT = PURPLE.

3. ALINK –ACTIVE LINK BEING CLICKED–DEFAULT= YELLOW.

THE FORMAT FOR SETTING THESE ATTRIBUTES IS:

```
<BODY BGCOLOR="#FFFFFF" TEXT="#FF0000" LINK="#0000FF"
    VLINK="#FF00FF"
   ALINK="FFFF00"> </BODY>
```

USING IMAGE BACKGROUND

- THE BODY ELEMENT ALSO GIVES YOU ABILITY OF SETTING AN IMAGE AS THE DOCUMENT'S BACKGROUND.
- AN EXAMPLE OF A BACKGROUND IMAGE'S HTML CODE IS AS FOLLOWS:

```
<BODY BACKGROUND="HI.GIF" BGCOLOR="#FFFFFF"></BODY>
```

PREVIEWING YOUR WORK

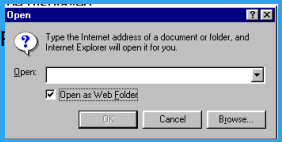
- ONCE YOU HAVE CREATED YOUR BASIC STARTING DOCUMENT AND SET YOUR DOCUMENT PROPERTIES IT IS A GOOD IDEA TO SAVE YOUR FILE.
- TO SAVE A FILE, IN NOTEPAD, FOLLOW THESE STEPS:

1. LOCATE AND CLICK ON THE MENU CALLED "FILE".
2. SELECT THE OPTION UNDER FILE MENU LABELED "SAVE AS".
3. IN THE "FILE NAME" TEXT BOX, TYPE IN THE ENTIRE NAME OF YOUR FILE (INCLUDING THE EXTENSION NAME .HTML).

EDIT, SAVE AND VIEW CYCLE

- TO PREVIEW YOUR WORK, OPEN A WEB BROWSER AND DO THE FOLLOWING:

1. CLICK ON THE MENU LABELED “FILE”.
2. LOCATE THE MENU OP



EDIT, SAVE AND VIEW CYCLE

3. IN THE "OPEN" DIALOG BOX, CLICK ON THE "BROWSE" BUTTON AND LOCATE YOUR WEB DOCUMENT.
4. CLICK "OK" ONCE YOU HAVE SELECTED YOUR FILE.

- THE WEB BROWSER WILL LOAD THE SAME DOCUMENT BUT WITH THE NEW REVISIONS. THIS PROCESS IS THE EDIT, SAVE AND VIEW CYCLE.

HEADINGS, PARAGRAPHS, BREAKS & HORIZONTAL RULES

IN THIS CHAPTER YOU WILL ADD HEADINGS TO YOUR PAGE, INSERT PARAGRAPHS, ADD SOME BREAKS, AND ADD HORIZONTAL RULES.

OBJECTIVES

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. LIST AND DESCRIBE THE DIFFERENT HEADING ELEMENTS.
2. USE PARAGRAPHS TO ADD TEXT TO A DOCUMENT.
3. INSERT BREAKS WHERE NECESSARY.
4. ADD A HORIZONTAL RULE.

HEADINGS, <HX> </HX>

- INSIDE THE **BODY** ELEMENT, HEADING ELEMENTS **H1** THROUGH **H6** ARE GENERALLY USED FOR MAJOR DIVISIONS OF THE DOCUMENT. HEADINGS ARE PERMITTED TO APPEAR IN ANY ORDER, BUT YOU WILL OBTAIN THE BEST RESULTS WHEN YOUR DOCUMENTS ARE DISPLAYED IN A BROWSER IF YOU FOLLOW THESE GUIDELINES:

1. **H1**: SHOULD BE USED AS THE HIGHEST LEVEL OF HEADING, **H2** AS THE NEXT HIGHEST, AND SO FORTH.
2. YOU SHOULD NOT SKIP HEADING LEVELS: E.G., AN **H3** SHOULD NOT APPEAR AFTER AN **H1**, UNLESS THERE IS AN **H2** BETWEEN THEM.

HEADINGS, <HX> </HX>

```
<HTML>
<HEAD>
<TITLE> EXAMPLE PAGE</TITLE>
</HEAD>
<BODY>
<H1> HEADING 1 </H1>
<H2> HEADING 2 </H2>
<H3> HEADING 3 </H3>
<H4> HEADING 4 </H4>
<H5> HEADING 5 </H5>
<H6> HEADING 6 </H6>
</BODY>
</HTML>
```

HEADING 1

HEADING 2

HEADING 3

HEADING 4

HEADING 5

HEADING 6

PARAGRAPHS, <P> </P>

- PARAGRAPHS ALLOW YOU TO ADD TEXT TO A DOCUMENT IN SUCH A WAY THAT IT WILL AUTOMATICALLY ADJUST THE END OF LINE TO SUITE THE WINDOW SIZE OF THE BROWSER IN WHICH IT IS BEING DISPLAYED. EACH LINE OF TEXT WILL STRETCH THE ENTIRE LENGTH OF THE WINDOW.

PARAGRAPHS, <P> </P>

```
<HTML><HEAD>
<TITLE> EXAMPLE PAGE</TITLE>
</HEAD>
<BODY></H1> HEADING 1 </H1>
<P> PARAGRAPH 1, ….</P>
<H2> HEADING 2 </H2>
<P> PARAGRAPH 2, ….</P>
<H3> HEADING 3 </H3>
<P> PARAGRAPH 3, ….</P>
<H4> HEADING 4 </H4>
<P> PARAGRAPH 4, ….</P>
<H5> HEADING 5 </H5>
<P> PARAGRAPH 5, ….</P>
<H6> HEADING 6</H6>
<P> PARAGRAPH 6, ….</P>
</BODY></HTML>
```

HEADING 1

PARAGRAPH 1,….

HEADING 2

PARAGRAPH 2,….

HEADING 3

PARAGRAPH 3,….

HEADING 4

PARAGRAPH 4,….

HEADING 5

PARAGRAPH 5,….

HEADING 6

PARAGRAPH 6,….

BREAK,

- LINE BREAKS ALLOW YOU TO DECIDE WHERE THE TEXT WILL BREAK ON A LINE OR CONTINUE TO THE END OF THE WINDOW.
- A
 IS AN EMPTY ELEMENT, MEANING THAT IT MAY CONTAIN ATTRIBUTES BUT IT DOES NOT CONTAIN CONTENT.
- THE
 ELEMENT DOES NOT HAVE A CLOSING TAG.

BREAK,

```
<HTML>
<HEAD>
<TITLE> EXAMPLE PAGE</TITLE>
</HEAD>
<BODY>
<H1> HEADING 1 </H1>
<P>PARAGRAPH 1, <BR>
LINE 2 <BR> LINE 3 <BR>....
</P>
</BODY>
</HTML>
```

HEADING 1

PARAGRAPH 1,....

LINE 2

LINE 3

....

HORIZONTAL RULE, <HR>

- THE <HR> ELEMENT CAUSES THE BROWSER TO DISPLAY A HORIZONTAL LINE (RULE) IN YOUR DOCUMENT.
- <HR> DOES NOT USE A CLOSING TAG, </HR>.

HORIZONTAL RULE, <HR>

Attribute	Description	Default Value
SIZE	Height of the rule in pixels	2 pixels
WIDTH	Width of the rule in pixels or percentage of screen width	100%
NOSHADE	Draw the rule with a flat look instead of a 3D look	Not set (3D look)
ALIGN	Aligns the line (Left, Center, Right)	Center
COLOR	Sets a color for the rule (IE 3.0 or later)	Not set

HORIZONTAL RULE, <HR>

```
<HTML>
<HEAD>
<TITLE> EXAMPLE PAGE</TITLE>
</HEAD>
<BODY>
<H1> HEADING 1 </H1>
<P>PARAGRAPH 1, <BR>
LINE 2 <BR>
<HR>LINE 3 <BR>
</P>
</BODY>
</HTML>
```

HEADING 1

PARAGRAPH 1,….

LINE 2

LINE 3

CHARACTER FORMATTING

IN THIS CHAPTER YOU WILL LEARN HOW TO ENHANCE YOUR PAGE WITH BOLD, ITALICS, AND OTHER CHARACTER FORMATTING OPTIONS.

OBJECTIVES

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. CHANGE THE COLOR AND SIZE OF YOUR TEXT.
2. USE COMMON CHARACTER FORMATTING ELEMENTS.
3. ALIGN YOUR TEXT.
4. ADD SPECIAL CHARACTERS.
5. USE OTHER CHARACTER FORMATTING ELEMENTS.

BOLD, ITALIC AND OTHER CHARACTER FORMATTING ELEMENTS

- **<FONT SIZE="+2">** TWO SIZES BIGGER**</FONT>**
- THE SIZE ATTRIBUTE CAN BE SET AS AN ABSOLUTE VALUE FROM 1 TO 7 OR AS A RELATIVE VALUE USING THE "+" OR "-" SIGN. NORMAL TEXT SIZE IS 3 (FROM -2 TO +4).
- **<B> BOLD </B>**
- **<I>** ***ITALIC*** **</I>**
- **<U>** **UNDERLINE** **</U>**
- COLOR = "#RRGGBB" THE COLOR ATTRIBUTE OF THE FONT ELEMENT. E.G., **<FONT COLOR="#RRGGBB">THIS TEXT HAS COLOR</FONT>**
- **<PRE> PREFORMATTED </PRE>** TEXT ENCLOSED BY PRE TAGS IS DISPLAYED IN A MONO-SPACED FONT. SPACES AND LINE BREAKS ARE SUPPORTED WITHOUT ADDITIONAL ELEMENTS OR SPECIAL CHARACTERS.

BOLD, ITALIC AND OTHER CHARACTER FORMATTING ELEMENTS

- **<EM> *EMPHASIS* </EM>** BROWSERS USUALLY DISPLAY THIS AS ITALICS.
- **<STRONG> STRONG </STRONG>** BROWSERS DISPLAY THIS AS BOLD.
- **<TT> TELETYPE </TT>** TEXT IS DISPLAYED IN A MONO-SPACED FONT. A TYPEWRITER TEXT, E.G. FIXED-WIDTH FONT.
- **<CITE> CITATION </CITE>** REPRESENTS A DOCUMENT CITATION (**ITALICS**). **FOR TITLES OF BOOKS, FILMS, ETC. TYPICALLY DISPLAYED IN ITALICS. (*A BEGINNER'S GUIDE TO HTML*)**

BOLD, ITALIC AND OTHER CHARACTER FORMATTING ELEMENTS

```
<P> <FONT SIZE="+1"> ONE SIZE
   LARGER </FONT> - NORMAL –

<FONT SIZE="-1"> ONE SIZE
   SMALLER </FONT> <BR>

<B> BOLD</B> - <I> ITALICS</I> -
   <U> UNDERLINED </U> -

<FONT COLOR="#FF0000">
   COLORED </FONT> <BR>

<EM> EMPHASIZED</EM> -
   <STRONG> STRONG </STRONG>
   - <TT> TELE TYPE </TT> <BR>
```

ONE SIZE LARGER - NORMAL – ONE SIZE SMALLER
BOLD - *ITALICS* - UNDERLINED - COLORED
EMPHASIZED - **STRONG** - TELE TYPE

ALIGNMENT

- SOME ELEMENTS HAVE ATTRIBUTES FOR ALIGNMENT (ALIGN) E.G. HEADINGS, PARAGRAPHS AND HORIZONTAL RULES.
- THE THREE ALIGNMENT VALUES ARE : LEFT, RIGHT, CENTER.
- <CENTER></CENTER> WILL CENTER ELEMENTS.

ALIGNMENT

- **<DIV ALIGN="VALUE"></DIV>** REPRESENTS A DIVISION IN THE DOCUMENT AND CAN CONTAIN MOST OTHER ELEMENT TYPE. THE ALIGNMENT ATTRIBUTE OF THE DIV ELEMENT IS WELL SUPPORTED.
- **<TABLE></TABLE>** INSIDE A TABLE, ALIGNMENT CAN BE SET FOR EACH INDIVIDUAL CELL.

SPECIAL CHARACTERS & SYMBOLS

- THESE CHARACTERS ARE RECOGNIZED IN HTML AS THEY BEGIN WITH AN AMPERSAND AND END WITH WITH A SEMI-COLON E.G. **&VALUE;** THE VALUE WILL EITHER BE AN ENTITY NAME OR A STANDARD ASCII CHARACTER NUMBER. THEY ARE CALLED **ESCAPE SEQUENCES**.
- THE NEXT TABLE REPRESENTS SOME OF THE MORE COMMONLY USED SPECIAL CHARACTERS. FOR A COMPREHENSIVE LISTING, VISIT THE W3C'S SECTION ON SPECIAL CHARACTERS AT:
  HTTP://WWW.W3.ORG/MARKUP/HTMLPLUS/HTMLPLUS_13.HTML

SPECIAL CHARACTERS & SYMBOLS

Special Character	Entity Name	Special Character	Entity Name
Ampersand	& &	Greater-than sign	> >
Asterisk	∗ **	Less-than sign	< <
Cent sign	¢ ¢	Non-breaking space	
Copyright	© ©	Quotation mark	" "
Fraction one qtr	¼ ¼	Registration mark	® ®
Fraction one half	½ ½	Trademark sign	™ ™

SPECIAL CHARACTERS & SYMBOLS

- ADDITIONAL ESCAPE SEQUENCES SUPPORT ACCENTED CHARACTERS, SUCH AS:
- **&OUML;**
  - A LOWERCASE O WITH AN UMLAUT: Ö
- **&NTILDE;**
  - A LOWERCASE N WITH A TILDE: Ñ
- **&EGRAVE;**
  - AN UPPERCASE E WITH A GRAVE ACCENT: È

NOTE: UNLIKE THE REST OF HTML, THE ESCAPE SEQUENCES ARE **CASE SENSITIVE**. YOU CANNOT, FOR INSTANCE, USE < INSTEAD OF <.

ADDITIONAL CHARACTER FORMATTING ELEMENTS

- **<STRIKE>** STRIKE-THROUGH TEXT**</STRIKE>**

DEL IS USED FOR STRIKE AT THE LATEST BROWSERS

- **<BIG>** PLACES TEXT IN A BIG FONT**</BIG>**
- **<SMALL>** PLACES TEXT IN A SMALL FONT**</SMALL>**
- **_{** PLACES TEXT IN SUBSCRIPT POSITION **}**
- **^{** PLACES TEXT IN SUPERSCRIPT STYLE POSITION **}**

EXAMPLE

```
<P><STRIKE> STRIKE-THROUGH TEXT </STRIKE></BR>

<BIG>PLACES TEXT IN A BIG FONT </BIG><BR>

<SMALL> PLACES TEXT IN A SMALL FONT</SMALL><BR>

<SUB> PLACES TEXT IN SUBSCRIPT POSITION </SUB>
NORMAL
<SUP> PLACES TEXT IN SUPERSCRIPT STYLE POSITION </SUP><BR>
  </P>
```

LISTS

IN THIS CHAPTER YOU WILL LEARN HOW TO CREATE A VARIETY OF LISTS.

OBJECTIVES

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. CREATE AN UNORDERED LIST.
2. CREATE AN ORDERED LIST.
3. CREATE A DEFINED LIST.
4. NEST LISTS.

LIST ELEMENTS

- HTML SUPPLIES SEVERAL LIST ELEMENTS. MOST LIST ELEMENTS ARE COMPOSED OF ONE OR MORE <LI> (LIST ITEM) ELEMENTS.
- UL : UNORDERED LIST. ITEMS IN THIS LIST START WITH A LIST MARK SUCH AS A BULLET. BROWSERS WILL USUALLY CHANGE THE LIST MARK IN NESTED LISTS.

<UL>

<LI> LIST ITEM …**</LI>**

<LI> LIST ITEM …**</LI>**

</UL>

- LIST ITEM …
- LIST ITEM …

LIST ELEMENTS

- YOU HAVE THE CHOICE OF THREE BULLET TYPES: **DISC(DEFAULT), CIRCLE, SQUARE.**
- THESE ARE CONTROLLED IN NETSCAPE NAVIGATOR BY THE "TYPE" ATTRIBUTE FOR THE <UL> ELEMENT.

<UL TYPE="SQUARE">

<LI> LIST ITEM …</LI>

<LI> LIST ITEM …</LI>

<LI> LIST ITEM …</LI>

</UL>

- LIST ITEM …
- LIST ITEM …
- LIST ITEM …

LIST ELEMENTS

- OL: ORDERED LIST. ITEMS IN THIS LIST ARE NUMBERED AUTOMATICALLY BY THE BROWSER.

<OL>

<LI> LIST ITEM …</LI>

<LI> LIST ITEM …</LI>

<LI> LIST ITEM …</LI>

</OL>

1. **LIST ITEM …**
2. **LIST ITEM …**
3. **LIST ITEM**

- YOU HAVE THE CHOICE OF SETTING THE TYPE ATTRIBUTE TO ONE OF FIVE NUMBERING STYLES.

LIST ELEMENTS

TYPE	Numbering Styles	
1	**Arabic numbers**	**1,2,3, ……**
a	**Lower alpha**	**a, b, c, ……**
A	**Upper alpha**	**A, B, C, ……**
i	**Lower roman**	**i, ii, iii, ……**
I	**Upper roman**	**I, II, III, ……**

LIST ELEMENTS

- YOU CAN SPECIFY A STARTING NUMBER FOR AN ORDERED LIST.

<OL TYPE =“I”>

<LI> LIST ITEM …</LI>

<LI> LIST ITEM …</LI>

</OL>

<P> TEXT ….</P>

<OL TYPE=“I” START=“3”>

<LI> LIST ITEM …</LI>

</OL>

LIST ELEMENTS

i. LIST ITEM …

ii. LIST ITEM …

TEXT ….

iii. LIST ITEM …

LIST ELEMENTS

- **DL: DEFINITION LIST**. THIS KIND OF LIST IS DIFFERENT FROM THE OTHERS. EACH ITEM IN A DL CONSISTS OF ONE OR MORE **DEFINITION TERMS (DT ELEMENTS),** FOLLOWED BY ONE OR MORE **DEFINITION DESCRIPTION (DD ELEMENTS).**

```
<DL>
<DT> HTML </DT>
<DD> HYPER TEXT MARKUP LANGUAGE </DD>
<DT> DOG </DT>
<DD> A HUMAN'S BEST FRIEND!</DD>
</DL>
```

HTML

HYPER TEXT MARKUP LANGUAGE

DOG

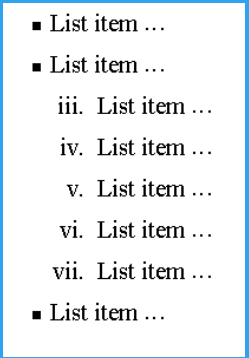
A HUMAN'S BEST FRIEND!

NESTING LISTS

- YOU CAN NEST LISTS BY INSERTING A UL, OL, ETC., INSIDE A LIST ITEM (LI).

EXAMPLE

```
<UL TYPE = "SQUARE">
<LI> LIST ITEM …</LI>
<LI> LIST ITEM …
<OL TYPE="I" START="3">
<LI> LIST ITEM …</LI>
<LI> LIST ITEM …</LI>
<LI> LIST ITEM …</LI>
<LI> LIST ITEM …</LI>
<LI> LIST ITEM …</LI>
</OL>
</LI>
<LI> LIST ITEM …</LI>
</UL>
```



WHAT WILL BE THE OUTPUT?

```
<H1 ALIGN="CENTER">SAFETY TIPS FOR CANOEISTS</H1>
<OL TYPE="a" START="2">
<LI>Be able to swim </LI>
<LI>Wear a life jacket at all times </LI>
<LI>Don't stand up or move around. If canoe tips,
        <UL>
        <LI>Hang on to the canoe </LI>
        <LI>Use the canoe for support and </LI>
         <LI>Swim to shore
        </UL> </LI>
<LI>Don't overexert yourself </LI>
<LI>Use a bow light at night </LI>
</OL>
```

THE OUTPUT….

SAFETY TIPS FOR CANOEISTS

b. Be able to swim
c. Wear a life jacket at all times
d. Don't stand up or move around. If canoe tips,
    - Hang on to the canoe
    - Use the canoe for support and
    - Swim to shore
e. Don't overexert yourself
f. Use a bow light at night

```
<H1 ALIGN="CENTER">SAFETY TIPS FOR
CANOEISTS</H1>
<OL TYPE="a" START="2">
<LI>Be able to swim </LI>
<LI>Wear a life jacket at all times </LI>
<LI>Don't stand up or move around. If canoe tips,
<UL>
<LI>Hang on to the canoe </LI>
<LI>Use the canoe for support
<OL type="I" start="4">
<LI> Be careful </LI>
<LI> Do not look around</LI>
</LI> </OL>
 <LI>Swim to shore
</UL> </LI>
<LI>Don't overexert yourself </LI>
<LI>Use a bow light at night </LI>
</OL>
```

What will be the output?

THE OUTPUT….

SAFETY TIPS FOR CANOEISTS

b. Be able to swim

c. Wear a life jacket at all times

d. Don't stand up or move around. If canoe tips,

- Hang on to the canoe
- Use the canoe for support

IV. Be careful

V. Do not look around

- Swim to shore

e. Don't overexert yourself

f. Use a bow light at night

IMAGES

IN THIS CHAPTER YOU WILL LEARN ABOUT IMAGES AND HOW TO PLACE IMAGES IN YOUR PAGES.

OBJECTIVES

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. ADD IMAGES TO YOUR PAGES.

IMAGES

- **<IMG>**THIS ELEMENT DEFINES A GRAPHIC IMAGE ON THE PAGE.
- **IMAGE FILE (SRC:SOURCE):** THIS VALUE WILL BE A URL (LOCATION OF THE IMAGE) E.G. HTTP://WWW.DOMAIN.COM/DIR/FILE.EXT OR /DIR/FILE.TXT.
- **ALTERNATE TEXT (ALT):** THIS IS A TEXT FIELD THAT DESCRIBES AN IMAGE OR ACTS AS A LABEL. IT IS DISPLAYED WHEN THEY POSITION THE CURSOR OVER A GRAPHIC IMAGE.
- **ALIGNMENT (ALIGN):** THIS ALLOWS YOU TO ALIGN THE IMAGE ON YOUR PAGE.

IMAGES

- **WIDTH (WIDTH):** IS THE WIDTH OF THE IMAGE IN PIXELS.
- **HEIGHT (HEIGHT):** IS THE HEIGHT OF THE IMAGE IN PIXELS.
- **BORDER (BORDER):** IS FOR A BORDER AROUND THE IMAGE, SPECIFIED IN PIXELS.
- **HSPACE:** IS FOR HORIZONTAL SPACE ON BOTH SIDES OF THE IMAGE SPECIFIED IN PIXELS. A SETTING OF 5 WILL PUT 5 PIXELS OF INVISIBLE SPACE ON BOTH SIDES OF THE IMAGE.
- **VSPACE:** IS FOR VERTICAL SPACE ON TOP AND BOTTOM OF THE IMAGE SPECIFIED IN PIXELS. A SETTING OF 5 WILL PUT 5 PIXELS OF INVISIBLE SPACE ABOVE AND BELLOW THE IMAGE.

SOME EXAMPLES ON IMAGES

1) <IMG SRC="JORDAN.GIF" BORDER=4>

2) <IMG SRC=" JORDAN.GIF" WIDTH="60" HEIGHT="60">

3) <IMG SRC="JORDAN.GIF" ALT="THIS IS A TEXT THAT GOES WITH THE IMAGE">

4) <IMG SRC=" JORDAN.GIF " HSPACE="30" VSPACE="10" BORDER=20>

5) < IMG SRC =" JORDAN.GIF" ALIGN="LEFT">

BLAST BLAST BLAST BLAST BLAST

ANCHORS, URLS AND IMAGE MAPS

IN THIS CHAPTER YOU WILL LEARN ABOUT UNIFORM RESOURCE LOCATOR, AND HOW TO ADD THEM AS ANCHOR OR LINKS INSIDE YOUR WEB PAGES.

OBJECTIVES

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. INSERT LINKS INTO DOCUMENTS.
2. DEFINE LINK TYPES.
3. DEFINE URL.
4. LIST SOME COMMONLY USED URLS.
5. PLAN AN IMAGE MAP.

HOW TO MAKE A LINK

1) THE TAGS USED TO PRODUCE LINKS ARE THE <A>
AND </A>. THE <A> TELLS WHERE THE LINK SHOULD START AND
THE </A> INDICATES WHERE THE LINK ENDS. EVERYTHING BETWEEN
THESE TWO WILL WORK AS A LINK.

2) THE EXAMPLE BELOW SHOWS HOW TO MAKE THE WORD HERE WORK AS A LINK TO YAHOO.

```
CLICK <A HREF="HTTP://WWW.YAHOO.COM">HERE</A>
   TO
GO TO YAHOO.
```

MORE ON LINKS

<BODY LINK="#C0C0C0" VLINK="#808080" ALINK="#FF0000">

- LINK - STANDARD LINK - TO A PAGE THE VISITOR HASN'T BEEN TO YET. (STANDARD COLOR IS BLUE - #0000FF).
  VLINK - VISITED LINK - TO A PAGE THE VISITOR HAS BEEN TO BEFORE. (STANDARD COLOR IS PURPLE - #800080).
  ALINK - ACTIVE LINK - THE COLOR OF THE LINK WHEN THE MOUSE IS ON IT. (STANDARD COLOR IS RED - #FF0000).

IF THE PROGRAMMER WHAT TO CHANGE THE COLOR

- CLICK <A HREF="HTTP://WWW.YAHOO.COM"><FONT COLOR="FF00CC">HERE</FONT></A> TO GO TO YAHOO.

INTERNAL LINKS

- INTERNAL LINKS : LINKS CAN ALSO BE CREATED INSIDE LARGE DOCUMENTS TO SIMPLIFY NAVIGATION. TODAY'S WORLD WANTS TO BE ABLE TO GET THE INFORMATION QUICKLY. INTERNAL LINKS CAN HELP YOU MEET THESE GOALS.

1. SELECT SOME TEXT AT A PLACE IN THE DOCUMENT THAT YOU WOULD LIKE TO CREATE A LINK TO, THEN ADD AN ANCHOR TO LINK TO LIKE THIS:

   <A NAME="BOOKMARK_NAME"></A>

   THE NAME ATTRIBUTE OF AN ANCHOR ELEMENT SPECIFIES A LOCATION IN THE DOCUMENT THAT WE LINK TO SHORTLY. ALL NAME ATTRIBUTES IN A DOCUMENT MUST BE UNIQUE.

2. NEXT SELECT THE TEXT THAT YOU WOULD LIKE TO CREATE AS A LINK TO THE LOCATION CREATED ABOVE.

   <A HREF="#BOOKMARK_NAME">GO TO BOOK MARK</A>

E-MAIL (ELECTRONIC MAIL)

E.G. MAILTO:KMF@YAHOO.COM

- THE TYPE OF SERVICE IS IDENTIFIED AS THE MAIL CLIENT PROGRAM. THIS TYPE OF LINK WILL LAUNCH THE USERS MAIL CLIENT.
- THE RECIPIENT OF THE MESSAGE IS KMF@YAHOO.COM

<A HREF="MAILTO:KMF@YAHOO.COM">SEND ME MORE INFORMATION </A>

IMAGE MAPS

- IMAGE MAPS ARE IMAGES, USUALLY IN GIF FORMAT THAT HAVE BEEN DIVIDED INTO REGIONS; CLICKING IN A REGION OF THE IMAGE CAUSE THE WEB SURFER TO BE CONNECTED TO A NEW URL. IMAGE MAPS ARE GRAPHICAL FORM OF CREATING LINKS BETWEEN PAGES.
- THERE ARE TWO TYPE OF IMAGE MAPS:

  CLIENT SIDE AND SERVER SIDE

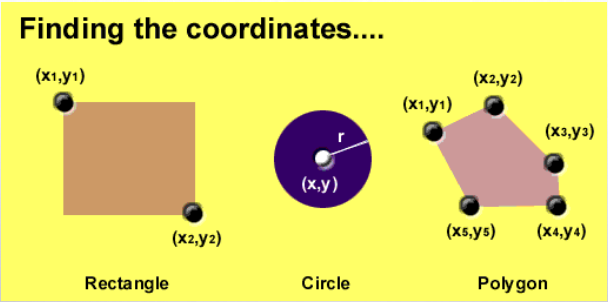
BOTH TYPES OF IMAGE MAPS INVOLVE A LISTING OF CO-ORDINATES

THAT DEFINE THE MAPPING REGIONS AND WHICH URLS THOSE

COORDINATES ARE ASSOCIATED WITH. THIS IS KNOWN AS THE MAP

FILE.

AREA SHAPES USED



CLIENT-SIDE IMAGE MAPS

- CLIENT-SIDE IMAGE MAPS (USEMAP) USE A MAP FILE THAT IS PART OF THE HTML DOCUMENT (IN AN ELEMENT CALLED MAP), AND IS LINKED TO THE IMAGE BY THE WEB BROWSER.

```
<IMG SRC="NOTE.GIF"  WIDTH=200 HEIGHT=200
BORDER="5" USEMAP="#MAP1">
<MAP NAME="MAP1">
<AREA SHAPE="RECT" COORDS="0,0,90,90"
HREF="HI.HTML"  ALT="SEE ME…">
<AREA SHAPE="RECT" COORDS="100,100,160,160"
HREF="DIVPARA.HTML"  ALT="SEE HIM…" >
<AREA SHAPE="CIRCLE" COORDS="150,50,20"
HREF="HOUSE.HTML"  ALT="SEE IT…" >
</MAP>
```

WE CAN USE POLY AS WELL AS RECT……

SHAPES, COORDS

- TYPES OF SHAPES
  - RECT → USED FOR SQUARES AND ORDERED SHAPES.
  - CIRCLE → USED FOR CIRCLES.
  - POLY → USED FOR UNORDERED SHAPES.
- NUMBER OF COORDENATIONS FOR EACH SHAPE:
  - RECT →4 NUMBERS FOR TWO CORNERS
  - CIRCLE →3 NUMBERS FOR THE CENTER & R
  - POLY → DEPENDS ON THE NUMBER OF CORNERS OF THE SHAPE(2 NUMBERS FOR EACH CORNER)

TABLES

IN THIS CHAPTER YOU WILL LEARN THAT TABLES HAVE MANY USES IN HTML.

OBJECTIVES:

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO:

1. INSERT A TABLE.
2. EXPLAIN A TABLE'S ATTRIBUTES.
3. EDIT A TABLE.
4. ADD A TABLE HEADER.

TABLES

- THE <TABLE></TABLE> ELEMENT HAS FOUR SUB-ELEMENTS:

1. TABLE ROW<TR></TR>.
2. TABLE HEADER <TH></TH>.
3. TABLE DATA <TD></TD>.
4. CAPTION <CAPTION></CAPTION>.

- THE TABLE ROW ELEMENTS USUALLY CONTAIN TABLE HEADER ELEMENTS OR TABLE DATA ELEMENTS.

TABLES

```
<TABLE BORDER="1">
<TR>
<TH> COLUMN 1 HEADER </TH>
<TH> COLUMN 2 HEADER </TH>
</TR>
<TR>
<TD> ROW1, COL1 </TD>
<TD> ROW1, COL2 </TD>
</TR>
<TR>
<TD> ROW2, COL1 </TD>
<TD> ROW2, COL2 </TD>
</TR>
</TABLE>
```

TABLES

Column 1 Header	Column 2 Header
Row1, Col1	Row1, Col2
Row2, Col1	Row2, Col2

TABLES ATTRIBUTES

- **BGCOLOR**: SOME BROWSERS SUPPORT BACKGROUND COLORS IN A TABLE.
- **WIDTH**: YOU CAN SPECIFY THE TABLE WIDTH AS AN ABSOLUTE NUMBER OF PIXELS OR A PERCENTAGE OF THE DOCUMENT WIDTH. YOU CAN SET THE WIDTH FOR THE TABLE CELLS AS WELL.
- **BORDER**: YOU CAN CHOOSE A NUMERICAL VALUE FOR THE BORDER WIDTH, WHICH SPECIFIES THE BORDER IN PIXELS.
- **CELLSPACING**: CELL SPACING REPRESENTS THE SPACE BETWEEN CELLS AND IS SPECIFIED IN PIXELS.

TABLE ATTRIBUTES

- **CELLPADDING**: CELL PADDING IS THE SPACE BETWEEN THE CELL BORDER AND THE CELL CONTENTS AND IS SPECIFIED IN PIXELS.
- **ALIGN**: TABLES CAN HAVE LEFT, RIGHT, OR CENTER ALIGNMENT.
- **BACKGROUND**: BACKGROUND IMAGE, WILL BE TITLED IN IE3.0 AND ABOVE.
- BORDERCOLOR, BORDERCOLORDARK.

TABLE CAPTION

- A TABLE CAPTION ALLOWS YOU TO SPECIFY A LINE OF TEXT THAT WILL APPEAR CENTERED ABOVE OR BELLOW THE TABLE.

<TABLE BORDER=1 CELLPADDING=2>

<CAPTION ALIGN="BOTTOM"> LABEL FOR MY TABLE </CAPTION>

- THE CAPTION ELEMENT HAS ONE ATTRIBUTE ALIGN THAT CAN BE EITHER TOP (ABOVE THE TABLE) OR BOTTOM (BELOW THE TABLE).

TABLE HEADER

- TABLE DATA CELLS ARE REPRESENTED BY THE TD ELEMENT. CELLS CAN ALSO BE TH (TABLE HEADER) ELEMENTS WHICH RESULTS IN THE CONTENTS OF THE TABLE HEADER CELLS APPEARING CENTERED AND IN BOLD TEXT.

TABLE DATA AND TABLE HEADER ATTRIBUTES

- **COLSPAN:** SPECIFIES HOW MANY CELL COLUMNS OF THE TABLE THIS CELL SHOULD SPAN.
- **ROWSPAN:** SPECIFIES HOW MANY CELL ROWS OF THE TABLE THIS CELL SHOULD SPAN.
- **ALIGN:** CELL DATA CAN HAVE LEFT, RIGHT, OR CENTER ALIGNMENT.
- **VALIGN:** CELL DATA CAN HAVE TOP, MIDDLE, OR BOTTOM ALIGNMENT.
- **WIDTH:** YOU CAN SPECIFY THE WIDTH AS AN ABSOLUTE NUMBER OF PIXELS OR A PERCENTAGE OF THE DOCUMENT WIDTH.
- **HEIGHT:** YOU CAN SPECIFY THE HEIGHT AS AN ABSOLUTE NUMBER OF PIXELS OR A PERCENTAGE OF THE DOCUMENT HEIGHT.

BASIC TABLE CODE

```
<TABLE BORDER=1 WIDTH=50%>
<CAPTION>  <H1>SPARE PARTS <H1> </CAPTION>
<TR><TH>STOCK NUMBER</TH><TH>DESCRIPTION</TH><TH>LIST
   PRICE</TH></TR>
<TR><TD BGCOLOR=RED>3476-AB</TD><TD>76MM
   SOCKET</TD><TD>45.00</TD></TR>
<TR><TD >3478-AB</TD><TD><FONT COLOR=BLUE>78MM SOCKET</FONT>
   </TD><TD>47.50</TD></TR>
<TR><TD>3480-AB</TD><TD>80MM SOCKET</TD><TD>50.00</TD></TR>
</TABLE>
```

Spare Parts

Stock Number	Description	List Price
3476-AB	76mm Socket	45.00
3478-AB	78mm Socket	47.50
3480-AB	80mm Socket	50.00

TABLE DATA AND TABLE HEADER ATTRIBUTES

```
<TABLE BORDER=1 CELLPADDING =2>
<TR> <TH> COLUMN 1 HEADER</TH> <TH> COLUMN 2 HEADER</TH>
  </TR>
<TR> <TD COLSPAN=2> ROW 1 COL 1</TD> </TR>
<TR> <TD ROWSPAN=2>ROW 2 COL 1</TD>
<TD> ROW 2 COL2</TD> </TR>
<TR> <TD> ROW 3 COL2</TD> </TR>
</TABLE>
```

TABLE DATA AND TABLE HEADER ATTRIBUTES

Column 1 Header	Column 2 Header
Row 1 Col 1	
Row 2 Col 1	Row 2 Col 2
	Row 3 Col 2

SPECIAL THINGS TO NOTE

- **TH, TD AND TR SHOULD ALWAYS HAVE END TAGS.**
  ALTHOUGH THE END TAGS ARE FORMALLY OPTIONAL, MANY BROWSERS WILL MESS UP THE FORMATTING OF THE TABLE IF YOU OMIT THE END TAGS. IN PARTICULAR, YOU SHOULD ***ALWAYS*** USE END TAGS IF YOU HAVE A TABLE WITHIN A TABLE -- IN THIS SITUATION, THE TABLE PARSER GETS HOPELESSLY CONFUSED IF YOU DON'T CLOSE YOUR TH, TD AND TR ELEMENTS.
- **A DEFAULT TABLE HAS NO BORDERS**
  BY DEFAULT, TABLES ARE DRAWN WITHOUT BORDER LINES. YOU NEED THE BORDER ATTRIBUTE TO DRAW THE LINES.
- **BY DEFAULT, A TABLE IS FLUSH WITH THE LEFT MARGIN**
  TABLES ARE PLOPPED OVER ON THE LEFT MARGIN. IF YOU WANT CENTERED TABLES, YOU CAN EITHER: PLACE THE TABLE INSIDE A DIV ELEMENT WITH ATTRIBUTE ALIGN="CENTER".
  MOST CURRENT BROWSERS ALSO SUPPORTS TABLE ALIGNMENT, USING THE ALIGN ATTRIBUTE. ALLOWED VALUES ARE "LEFT", "RIGHT", OR "CENTER", FOR EXAMPLE: <TABLE ALIGN="LEFT">. THE VALUES "LEFT" AND "RIGHT" FLOAT THE TABLE TO THE LEFT OR RIGHT OF THE PAGE, WITH TEXT FLOW ALLOWED AROUND THE TABLE. THIS IS ENTIRELY EQUIVALENT TO IMG ALIGNMENT

WHAT WILL BE THE OUTPUT?

```
<TABLE BORDER WIDTH="750">
<TR> <TD COLSPAN="4" ALIGN="CENTER">PAGE BANNER</TD></TR>

<TR> <TD ROWSPAN="2" WIDTH="25%">NAV LINKS</TD><TD
   COLSPAN="2">FEATURE ARTICLE</TD> <TD ROWSPAN="2"
   WIDTH="25%">LINKED ADS</TD></TR>

   <TR><TD WIDTH="25%">NEWS COLUMN 1 </TD> <TD
   WIDTH="25%"><NEWS COLUMN 2 </TD></TR>
</TABLE>
```

THE OUTPUT

Figure 5-26 Column widths set to 25%

FRAMES

- FRAMES ARE A RELATIVELY NEW ADDITION TO THE HTML STANDARD. FIRST INTRODUCED IN NETSCAPE NAVIGATOR 2.0.

OBJECTIVES:

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO:

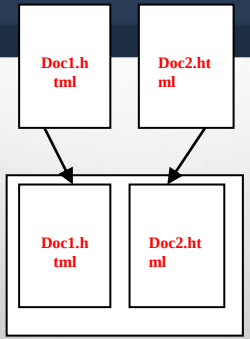
- CREATE A FRAME BASED PAGE.
- WORK WITH THE FRAMESET, FRAME, AND NOFRAMES ELEMENTS.
- USE THE ATTRIBUTES OF THE FRAMES ELEMENTS TO CONTROL THE DISPLAY.
- SET TARGETS APPROPRIATELY.

FRAMES

- A FRAMED PAGE IS ACTUALLY MADE UP OF MULTIPLE HTML PAGES. THERE IS ONE HTML DOCUMENT THAT DESCRIBES HOW TO BREAK UP THE SINGLE BROWSER WINDOW INTO MULTIPLE WINDOWPANES. EACH WINDOWPANE IS FILLED WITH AN HTML DOCUMENT.

- FOR EXAMPLE TO MAKE A FRAMED PAGE WITH A WINDOWPANE ON THE LEFT AND ONE ON THE RIGHT REQUIRES THREE HTML PAGES. ***DOC1.HTML*** AND ***DOC2.HTML*** ARE THE PAGES THAT CONTAIN CONTENT. ***FRAMES.HTML*** IS THE PAGE THAT DESCRIBES THE DIVISION OF THE SINGLE BROWSER WINDOW INTO TWO WINDOWPANES.

FRAMES



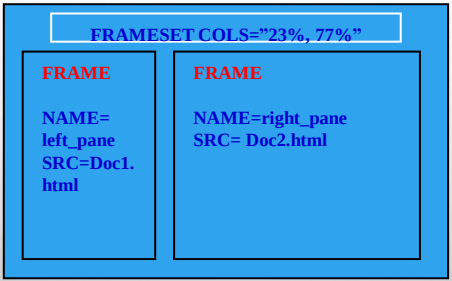
FRAME PAGE ARCHITECTURE

- A **<FRAMESET>** ELEMENT IS PLACED IN THE HTML DOCUMENT BEFORE THE **<BODY>** ELEMENT. THE **<FRAMESET>** DESCRIBES THE AMOUNT OF SCREEN REAL ESTATE GIVEN TO EACH WINDOWPANE BY DIVIDING THE SCREEN INTO **ROWS** OR **COLS**.
- THE **<FRAMESET>** WILL THEN CONTAIN **<FRAME>** ELEMENTS, **ONE PER DIVISION** OF THE BROWSER WINDOW.
- NOTE: BECAUSE THERE IS NO **BODY** CONTAINER, FRAMESET PAGES CAN'T HAVE BACKGROUND IMAGES AND BACKGROUND COLORS ASSOCIATED WITH THEM.

FRAME PAGE ARCHITECTURE

```
<HTML>
<HEAD>
<TITLE> FRAMED PAGE </TITLE>
<FRAMESET COLS="23%,77%">
<FRAME SRC="DOC1.HTML">
<FRAME SRC="DOC2.HTML">
</FRAMESET >
</HEAD>

</HTML>
```

THE DIAGRAM BELOW IS A GRAPHICAL VIEW OF THE DOCUMENT DESCRIBED ABOVE



<FRAMESET> **CONTAINER**

<FRAMESET> : THE FRAMESET ELEMENT CREATES DIVISIONS IN THE BROWSER WINDOW IN A SINGLE DIRECTION. THIS ALLOWS YOU TO DEFINE DIVISIONS AS EITHER ROWS OR COLUMNS.

- **ROWS :** DETERMINES THE SIZE AND NUMBER OF RECTANGULAR ROWS WITHIN A <FRAMESET>. THEY ARE SET FROM TOP OF THE DISPLAY AREA TO THE BOTTOM.

POSSIBLE VALUES ARE:

- ABSOLUTE PIXEL UNITS, I.E. “360,120”.
- A PERCENTAGE OF SCREEN HEIGHT, E.G. “75%,25%”.
- PROPORTIONAL VALUES USING THE ASTERISK (*). THIS IS OFTEN COMBINED WITH A VALUE IN PIXELS , E.G. “360,*”.
- <FRAMESET COLS=“200,20%,*,2*”>

CREATING A FRAMES PAGE

- **COLS:** DETERMINES THE SIZE AND NUMBER OF RECTANGULAR COLUMNS WITHIN A <FRAMESET>. THEY ARE SET FROM **LEFT** TO **RIGHT** OF THE DISPLAY AREA.

POSSIBLE VALUES ARE:

- ABSOLUTE PIXEL UNITS, I.E. “480,160”.
- A PERCENTAGE OF SCREEN WIDTH, E.G. “75%,25%”.
- PROPORTIONAL VALUES USING THE ASTERISK (*). THIS IS OFTEN COMBINED WITH A VALUE IN PIXELS , E.G. “480,*”.

CREATING A FRAMES PAGE

- **FRAMEBORDER :** POSSIBLE VALUES **0**, **1**, **YES**, **NO**. A SETTING OF ZERO WILL CREATE A BORDERLESS FRAME.
- **FRAMESPACING:** THIS ATTRIBUTE IS SPECIFIED IN **PIXELS**. IF YOU GO TO BORDERLESS FRAMES YOU WILL NEED TO SET THIS VALUE TO ZERO AS WELL, OR YOU WILL HAVE A GAP BETWEEN YOUR FRAMES WHERE THE BORDER USED TO BE.
- **BORDER(THICKNESS OF THE FRAME):** THIS ATTRIBUTE SPECIFIED IN PIXELS. A SETTING OF ZERO WILL CREATE A BORDERLESS FRAME. DEFAULT VALUE IS 5.
- **BORDERCOLOR:** THIS ATTRIBUTE IS ALLOWS YOU CHOOSE A COLOR FOR YOUR BORDER. THIS ATTRIBUTE IS RARELY USED.

Internet Explorer and Netscape do disagree, however, as to how you may control the thickness of the borders. Internet Explorer supports the `framespacing` attribute, whose value is the number of pixels you want between frames (see Figure 12.2). The attribute affects all frames and framesets nested within the current frameset as displayed by Internet Explorer. In practice, you should set it once on the outermost `<frameset>` to create a consistent border appearance for all of the frames in a single page.

Netscape accepts only the `border` attribute to define the border width, with an integer value in pixels. Like Internet Explorer, Netscape lets you include the `frameborder` attribute with any `<frameset>` tag, affecting all nested frames and framesets. Unlike Internet Explorer, Netscape lets you include the `border` attribute only in the outermost `<frameset>`, ensuring that all frame borders are the same width within that `<frameset>`.

Since browsers ignore unsupported attributes, it is possible to define frame borders so that both browsers do the right thing. Just make sure to use the same `framespacing` and `border` values.

Finally, with Netscape you can control the color of the frame borders using the `bordercolor` attribute (Figure 12.3). It accepts a color name or hexadecimal triple as its value. A complete list of color names and values can be found in Appendix F, Color Names and Values.

Figure 12.3: Netscape accepts border and bordercolor attributes to control the color and spacing between frames

<FRAME>

<FRAME>: THIS ELEMENT DEFINES A SINGLE FRAME WITHIN A FRAMESET. THERE WILL BE A FRAME ELEMENT FOR EACH DIVISION CREATED BY THE FRAMESET ELEMENT. THIS TAG HAS THE FOLLOWING ATTRIBUTES:

- **SRC:** REQUIRED, AS IT PROVIDES THE URL FOR THE PAGE THAT WILL BE DISPLAYED IN THE FRAME.
- **NAME:** REQUIRED FOR FRAMES THAT WILL ALLOW TARGETING BY OTHER HTML DOCUMENTS. WORKS IN CONJUNCTION WITH THE TARGET ATTRIBUTE OF THE <A>, <AREA>, <BASE>, AND <FORM> TAGS.

<FRAME>

- **MARGINWIDTH**: OPTIONAL ATTRIBUTE STATED IN PIXELS. DETERMINES HORIZONTAL SPACE BETWEEN THE <FRAME> CONTENTS AND THE FRAME'S BORDERS.
- **MARGINHEIGHT**: OPTIONAL ATTRIBUTE STATED IN PIXELS. DETERMINES VERTICAL SPACE BETWEEN THE <FRAME> CONTENTS AND THE FRAME'S BORDERS.
- **SCROLLING**: DISPLAYS A SCROLL BAR(S) IN THE FRAME. POSSIBLE VALUES ARE:

1. **YES** – ALWAYS DISPLAY SCROLL BAR(S).
2. **NO** – NEVER DISPLAY SCROLL BAR(S).
3. **AUTO** – BROWSER WILL DECIDE BASED ON FRAME CONTENTS.

BY DEFAULT: SCROLLING IS AUTO.

<FRAME>

- **NORESIZE:** OPTIONAL – PREVENTS VIEWERS FROM RESIZING THE FRAME. BY DEFAULT THE USER CAN STRETCH OR SHRINK THE FRAME'S DISPLAY BY SELECTING THE FRAME'S BORDER AND MOVING IT UP, DOWN, LEFT, OR RIGHT.

<NOFRAMES>

- **<NOFRAMES>:** FRAME – CAPABLE BROWSERS IGNORE ALL HTML WITHIN THIS TAG INCLUDING THE CONTENTS OF THE BODY ELEMENT. THIS ELEMENT DOES NOT HAVE ANY ATTRIBUTES.

<HTML>

<HEAD>

<TITLE> FRAMED PAGE </TITLE>

</HEAD>

<NOFRAMES>

```
<FRAMESET COLS="23%,77%">
<FRAME  SRC=""   NAME="LEFT_PANE“>
<FRAME  SRC=""   NAME="RIGHT_PANE">
<NOFRAMES>
<P> THIS IS A FRAMED PAGE. UPGRADE YOUR
  BROWSER TO SUPPORT FRAMES.</P>
</NOFRAMES></FRAMESET>
```

COMPOUND FRAMESET DIVISIONS

- IN THIS CASE A SECOND **FRAMESET** ELEMENT WILL BE INSERTED IN THE PLACE OF THE **FRAME** ELEMENT THAT WOULD DESCRIBE THE SECOND ROW.
- THE SECOND **FRAMESET** ELEMENT WILL DIVIDE THE REMAINING SCREEN REAL ESTATE INTO **2** COLUMNS.
- THIS NESTED **FRAMESET** WILL THEN BE FOLLOWED BY **2 FRAME** ELEMENTS TO DESCRIBE EACH OF THE SUBSEQUENT FRAME DIVISIONS CREATED.

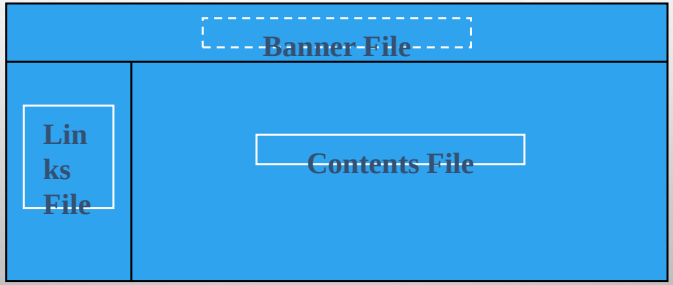
COMPOUND FRAMESET DIVISIONS

```
<HTML>
<HEAD>
<TITLE> COMPOUND FRAMES
    PAGE</TITLE>
</HEAD>
<FRAMESET ROWS="120,*">
<FRAME SRC="BANNER_FILE.HTML"
    NAME"BANNER">
<FRAMESET COLS="120,*">
<FRAME SRC="LINKS_FILE.HTML"
    NAME="LINKS">
<FRAME SRC="CONTENT_FILE.HTML"
    NAME="CONTENT">
```

```
<NOFRAMES>
<P>
DEFAULT
  MESSAGE
</P>
</NOFRAMES
  >
</FRAMESET>
</FRAMESET>
</HEAD>
```

COMPOUND FRAMESET DIVISIONS

YOU MAY WANT TO CREATE A FRAMES DESIGN WITH A COMBINATION OF ROWS AND COLUMNS.



COMPOUND FRAMESET DIVISIONS EXAMPLE

```
<HEAD>
<FRAMESET ROWS="25%,50%,25%"
        <FRAME SRC="">
<FRAMESET COLS="25%,*">
                <FRAME SRC="">
                <FRAME SRC="">
                        </FRAMESET>
        <FRAME SRC="">
</FRAMESET>
</HEAD>
```

OUTPUT

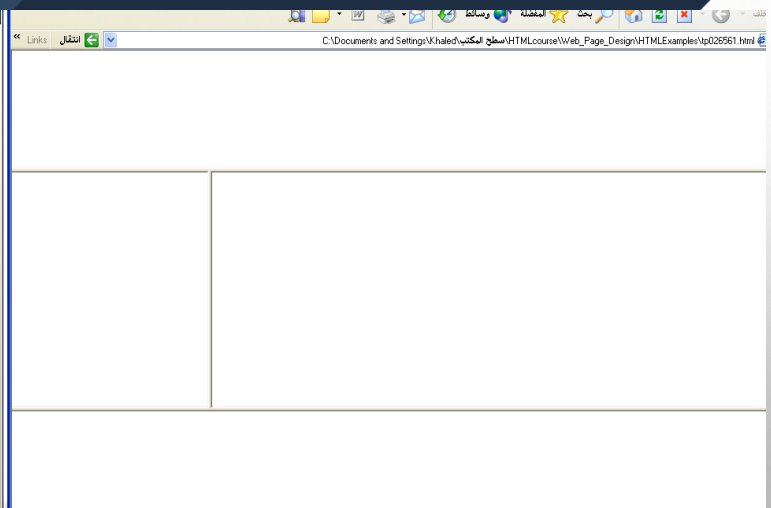

Links
انتقال
C:\Documents and Settings\Khaled\سطح المكتب\HTMLcourse\Web_Page_Design\HTMLExamples\tp026561.html

Figure 5-14: Frames created with <FRAMESET ROWS="50%, 50%">

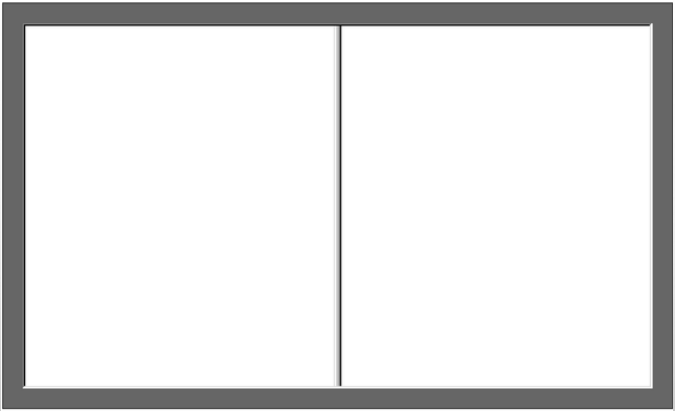


Figure 5-15: Frames created with <FRAMESET COLS="50%, 50%">

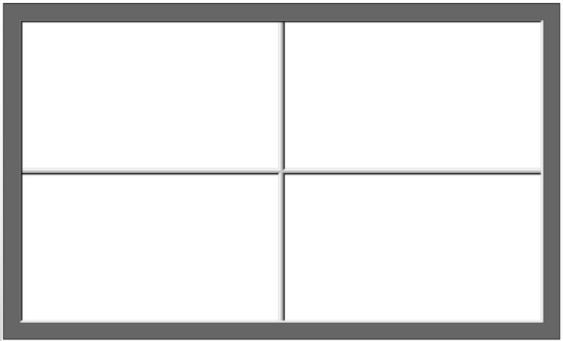


Figure 5-13: Frames created with <FRAMESET ROWS="50%, 50%" COLS="50%, 50%">

FRAME FORMATTING

- **EXAMPLE:**

```
<FRAMESET ROWS="20%, *, 20%">
    <FRAME SRC="HEADER.HTML"
  NORESIZE SCROLLING=NO>
    <FRAME SRC="BODY.HTML">
    <FRAME SRC="NAVIGATIONBAR.HTML"
  NORESIZE   SCROLLING=NO>
</FRAMESET>
```

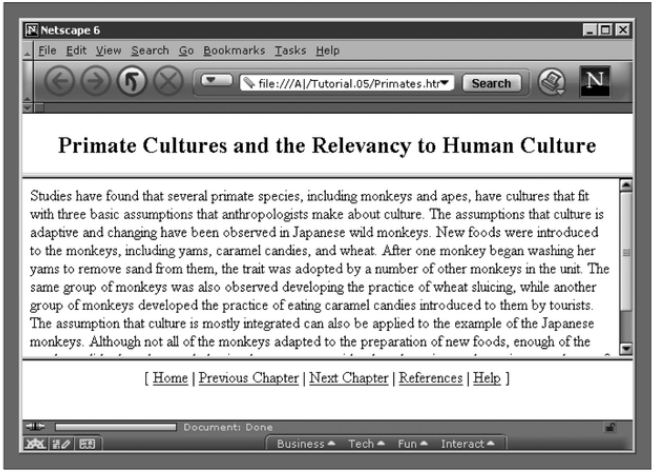


Figure 5-24: Output of program that includes NORESIZE and SCROLLING attributes

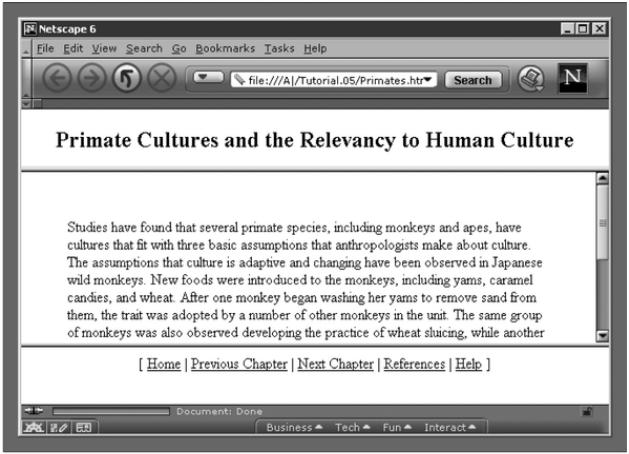


Figure 5-25: Middle frame changed to <FRAME SRC="body.html" MARGINHEIGHT=50 MARGINWIDTH=50>

WHAT DO THE FOLLOWING MEAN?

1) <FRAMESET COLS="2*, 3*, 5*">

2) <FRAMESET COLS="150, 20%, *, 3*">

SO WHAT ARE THE SPACE-ALLOCATION PRIORITIES? ABSOLUTE PIXEL VALUES ARE ALWAYS ASSIGNED SPACE FIRST, IN ORDER FROM LEFT TO RIGHT. THESE ARE FOLLOWED BY PERCENTAGE VALUES OF THE TOTAL SPACE. FINALLY, PROPORTIONAL VALUES ARE DIVIDED BASED UPON WHAT SPACE IS LEFT.

GENERIC FRAME FORMULA

- THE <FRAME> TAG HAS SIX ASSOCIATED ATTRIBUTES: SRC, NAME, MARGINWIDTH, MARGINHEIGHT, SCROLLING, AND NORESIZE. HERE'S A COMPLETE GENERIC FRAME:
- <FRAME SRC="URL"
  NAME="WINDOW_NAME"
  SCROLLING=YES|NO|AUTO
  MARGINWIDTH="VALUE"
  MARGINHEIGHT="VALUE"
  NORESIZE>

WHAT WILL BE THE OUTPUT?

```
<FRAMESET ROWS="*, 2*, *"   COLS="2*, *">
<FRAME SRC=“”>
<FRAME SRC=“”>
<FRAME SRC=“”>
<FRAME SRC=“”>
<FRAME SRC=“”>
<FRAME SRC=“”>
</FRAMESET>
```

TARGETS

- WHEN YOU USE LINKS FOR USE IN A FRAMES ENVIRONMENT YOU WILL NEED TO SPECIFY AN ADDITIONAL ATTRIBUTE CALLED **TARGET**.
- THE **TARGET** ATTRIBUTE USES THE NAME ATTRIBUTE OF THE **FRAME** ELEMENT.
- IF WE WERE TO PLACE A LINK IN DOC1.HTML THAT LINKED TO DOC3.HTML AND WE WANTED DOC3.HTML TO BE DISPLAYED IN THE RIGHT WINDOWPANE; THE HTML CODE WOULD APPEAR IN DOC1.HTML AS FOLLOWS:

```
<A HREF="DOC3.HTML" TARGET="RIGHT_PANE">LINK TO DOCUMENT 3
</A>
```

SPECIAL TARGETS

- THERE ARE **4** SPECIAL TARGET NAMES THAT CANNOT BE ASSIGNED BY THE NAME ATTRIBUTE OF THE FRAME TAG.

1. **TARGET="_TOP"** : THIS LOADS THE LINKED DOCUMENT INTO THE FULL BROWSER WINDOW WITH THE URL SPECIFIED BY THE HREF ATTRIBUTE. ALL FRAMES DISAPPEAR, LEAVING THE NEW LINKED PAGE TO OCCUPY THE ENTIRE WINDOW. THE BACK IS TURNED ON.
2. **TARGET="_BLANK"** : OPENS AN UNNAMED NEW BROWSER WINDOW AND LOADS THE DOCUMENT SPECIFIED IN THE URL ATTRIBUTE INTO THE NEW WINDOW (AND YOUR OLD WINDOW STAYS OPEN). THE BACK IS TURNED OFF. OTHER WINDOWS REMAINS ON.
3. **TARGET="_SELF"** : LOADS THE DOCUMENT IN THE SAME WINDOW WHERE THE ANCHOR WAS {*CLICKED*}. THIS IS THE **DEFAULT** SETTING FOR LINKING ELEMENTS.
4. **TARGET="_PARENT"** : THE _PARENT FRAME IS A PRIOR FRAMESET THAT THE CURRENT FRAMESET WAS "SPAWNED" FROM. IF THERE ISN'T ONE IT IS THE BROWSER WINDOW. THE DOCUMENT IS LOADED INTO THE AREA OCCUPIED BY THE COLUMNS OR ROWS FRAMESET CONTAINING THE FRAME THAT CONTAINS THE LINK. THE BACK IS TURNED ON. ALL WINDOWS DISAPPEAR.

If a frame contains the following link, then clicking the link launches a new, unnamed browser display window that contains the content defined in stuff.HTM. This can be a simple HTML document, or an entirely new FRAMESET definition.

1. `<A HREF="stuff.html" TARGET="_blank">`

If a frame contains the following link, then clicking the link will simply cause the frame which contains the link to clear, and its content will be replaced with whatever is in stuff.htm.

2. `<A HREF="stuff.html" TARGET="_self">`

If a frame contains the following link, the frameset that contains the frame that contains this link will be replaced by stuff.HTM.

3. `<A HREF="stuff.html" TARGET="_parent">`

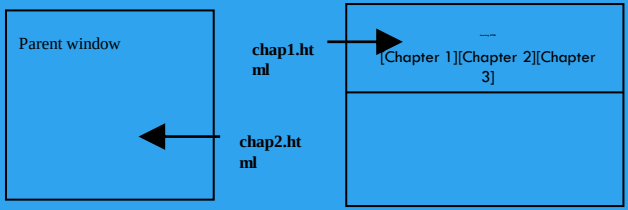
Finally, if a frame contains the following link, clicking the link replaces the entire browser window with the contents of stuff.HTM.

4. `<A HREF="stuff.html" TARGET="_top">`

Targeting links to frames

The TARGET attribute allows you to specify the frame into which a page is to be loaded into in a frames setting.

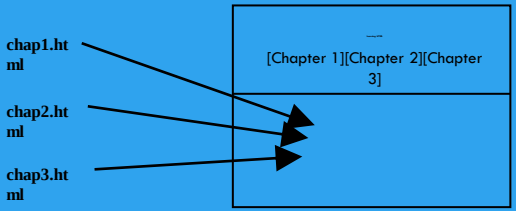
<A HREF="chap1.html" TARGET="_self"> [Chapter 1]</A>
<A HREF="chap1.html" TARGET="_parent"> [Chapter 2]</A>



Targeting links to frames

The TARGET attribute allows you to specify the frame into which a page is to be loaded into in a frames setting.

<A HREF="chap1.html" TARGET="bottom"> [Chapter 1]</A>
<A HREF="chap2.html" TARGET="bottom"> [Chapter 2]</A>
<A HREF="chap3.html" TARGET="bottom"> [Chapter 3]</A>



FORMS

- FORMS ADD THE ABILITY TO WEB PAGES TO NOT ONLY PROVIDE THE PERSON VIEWING THE DOCUMENT WITH DYNAMIC INFORMATION BUT ALSO TO OBTAIN INFORMATION FROM THE PERSON VIEWING IT, AND PROCESS THE INFORMATION.

OBJECTIVES:

UPON COMPLETING THIS SECTION, YOU SHOULD BE ABLE TO

1. CREATE A FORM.
2. ADD ELEMENTS TO A FORM.
3. DEFINE CGI (COMMON GATEWAY INTERFACE).
4. DESCRIBE THE PURPOSE OF A CGI APPLICATION.
5. SPECIFY AN ACTION FOR THE FORM.

- FORMS WORK IN ALL BROWSERS.
- FORMS ARE PLATFORM INDEPENDENT.

FORMS

- TO INSERT A FORM WE USE THE <FORM></FORM> TAGS. THE REST OF THE FORM ELEMENTS MUST BE INSERTED IN BETWEEN THE FORM TAGS.

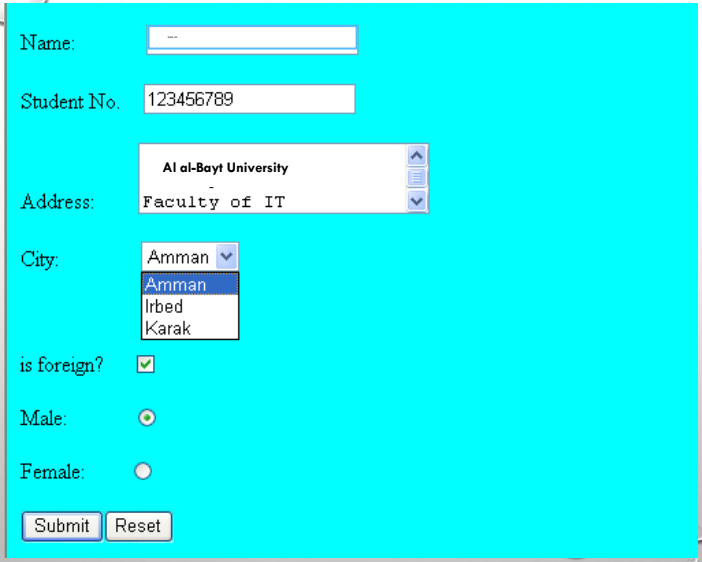
```
<HTML> <HEAD>
<TITLE> SAMPLE FORM</TITLE>
</HEAD>
<BODY BGCOLOR="FFFFFF">
<FORM ACTION = HTTP://WWW.XNU.COM/FORMTEST.ASP>
<P> FIRST NAME: <INPUT TYPE="TEXT" NAME="FNAME" MAXLENGTH="50"> </P>
<P> <INPUT TYPE="SUBMIT" NAME="FSUBMIT1" VALUE="SEND INFO"> </P>
</FORM>
</BODY> </HTML>
```

<FORM> ELEMENT ATTRIBUTES

- **ACTION:** IS THE **URL** OF THE **CGI** (COMMON GATEWAY INTERFACE) PROGRAM THAT IS GOING TO ACCEPT THE DATA FROM THE FORM, PROCESS IT, AND SEND A RESPONSE BACK TO THE BROWSER.
- **METHOD: GET** (DEFAULT) OR **POST** SPECIFIES WHICH **HTTP** METHOD WILL BE USED TO SEND THE FORM'S CONTENTS TO THE WEB SERVER. THE CGI APPLICATION SHOULD BE WRITTEN TO ACCEPT THE DATA FROM EITHER METHOD.
- **NAME:** IS A FORM NAME USED BY VBSCRIPT OR JAVASCRIPTS.
- **TARGET:** IS THE TARGET FRAME WHERE THE RESPONSE PAGE WILL SHOW UP.

FORM ELEMENTS

- FORM ELEMENTS HAVE PROPERTIES: TEXT BOXES, PASSWORD BOXES, CHECKBOXES, OPTION(RADIO) BUTTONS, SUBMIT, RESET, FILE, HIDDEN AND IMAGE.
- THE PROPERTIES ARE SPECIFIED IN THE TYPE ATTRIBUTE OF THE HTML ELEMENT <INPUT></INPUT>.


Name:
Student No.
123456789
Address:
Al al-Bayt University
Faculty of IT
City:
Amman
Amman
Irbed
Karak
is foreign?
Male:
Female:
Submit
Reset

FORM ELEMENTS

<INPUT> Element's Properties
TYPE= Type of INPUT entry field.
NAME = Variable name passed to CGI application
VALUE= The data associated with the variable name to be passed to the CGI application
CHECKED= Button/box checked
SIZE= Number of visible characters in text field
MAXLENGHT= Maximum number of characters accepted.

TEXT BOX

- **TEXT BOXES**: USED TO PROVIDE INPUT FIELDS FOR TEXT, PHONE NUMBERS, DATES, ETC.

<INPUT TYPE= " TEXT " >

BROWSER WILL DISPLAY

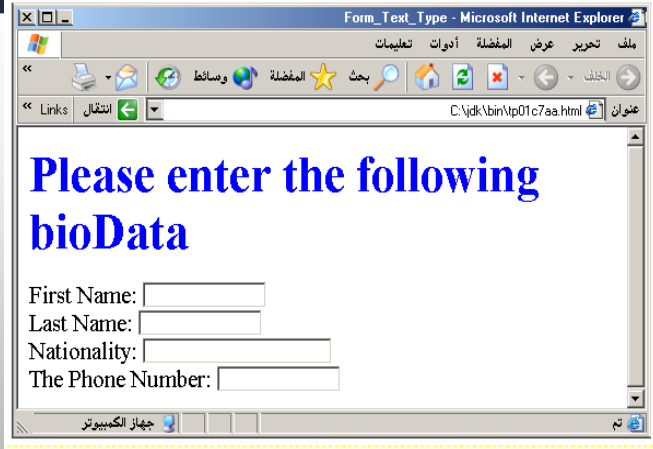
TEXTBOXES USE THE FOLLOWING ATTRIBUTES:

- **TYPE:** TEXT.
- **SIZE:** DETERMINES THE SIZE OF THE TEXTBOX IN CHARACTERS. **DEFAULT=20** CHARACTERS.
- **MAXLENGHT :** DETERMINES THE MAXIMUM NUMBER OF CHARACTERS THAT THE FIELD WILL ACCEPT.
- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **VALUE:** WILL DISPLAY ITS CONTENTS AS THE DEFAULT VALUE.

EXAMPLE ON TEXT BOX

```
<TITLE>FORM_TEXT_TYPE</TITLE>
</HEAD> <BODY>
<H1> <FONT COLOR=BLUE>PLEASE ENTER THE FOLLOWING
  BIODATA</FONT></H1>
<FORM NAME="FOME1"  METHOD= " GET " ACTION= " URL " >
FIRST NAME: <INPUT TYPE="TEXT" NAME="FNAME"
SIZE="15" MAXLENGTH="25"><BR>
LAST NAME: <INPUT TYPE="TEXT" NAME="LNAME"
SIZE="15" MAXLENGTH="25"><BR>
NATIONALITY: <INPUT TYPE="TEXT" NAME="COUNTRY"
SIZE="25" MAXLENGTH="25"><BR>
THE PHONE NUMBER: <INPUT TYPE="TEXT" NAME="PHONE"
SIZE="15" MAXLENGTH="12"><BR>
</FORM> </BODY> </HTML>
```

OUTPUT


Form_Text_Type - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
عنوان C:\jdk\bin\tp01c7aa.html
Links انتقال
Please enter the following bioData
First Name:
Last Name:
Nationality:
The Phone Number:
جهاز الكمبيوتر
تم

PASSWORD

- **PASSWORD:** USED TO ALLOW ENTRY OF PASSWORDS.

<INPUT TYPE= " PASSWORD " >

BROWSER WILL DISPLAY

TEXT TYPED IN A PASSWORD BOX IS STARRED OUT IN THE BROWSER DISPLAY.

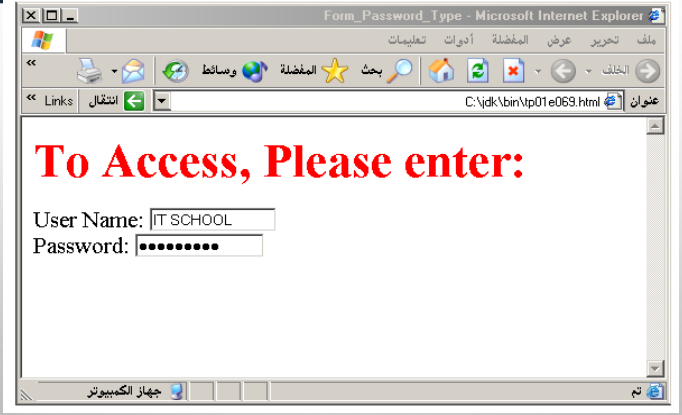
PASSWORD BOXES USE THE FOLLOWING ATTRIBUTES:

- **TYPE:** PASSWORD.
- **SIZE:** DETERMINES THE SIZE OF THE TEXTBOX IN CHARACTERS.
- **MAXLENGHT:** DETERMINES THE MAXIMUM SIZE OF THE PASSWORD IN CHARACTERS.
- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **VALUE:** IS USUALLY BLANK.

EXAMPLE ON PASSWORD BOX

```
<HTML><HEAD>
<TITLE>FORM_PASSWORD_TYPE</TITLE></HEAD>
<BODY>
<H1> <FONT COLOR=RED>TO ACCESS, PLEASE
ENTER:</FONT></H1>
<FORM NAME="FOME2"  ACTION="URL"  METHOD="GET">
USER NAME: <INPUT TYPE="TEXT" NAME="FNAME"
SIZE="15" MAXLENGTH="25"><BR>
PASSWORD: <INPUT TYPE="PASSWORD"
NAME="PWORD"   VALUE="" SIZE="15"
MAXLENGTH="25"><BR>
</FORM></BODY> </HTML>
```

OUTPUT


Form_Password_Type - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
عنوان C:\jdk\bin\tp01e069.html انتقال Links
To Access, Please enter:
User Name: IT SCHOOL
Password:
تم
جهاز الكمبيوتر

HIDDEN

- **HIDDEN:** USED TO SEND DATA TO THE CGI APPLICATION THAT YOU DON'T WANT THE WEB SURFER TO SEE, CHANGE OR HAVE TO ENTER BUT IS NECESSARY FOR THE APPLICATION TO PROCESS THE FORM CORRECTLY.

<INPUT TYPE="HIDDEN">

NOTHING IS DISPLAYED IN THE BROWSER.

HIDDEN INPUTS HAVE THE FOLLOWING ATTRIBUTES:

- **TYPE:** HIDDEN.
- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **VALUE:** IS USUALLY SET A VALUE EXPECTED BY THE CGI APPLICATION.

CHECK BOX

- **CHECK BOX:** CHECK BOXES ALLOW THE USERS TO SELECT MORE THAN ONE OPTION.

<INPUT TYPE="CHECKBOX">

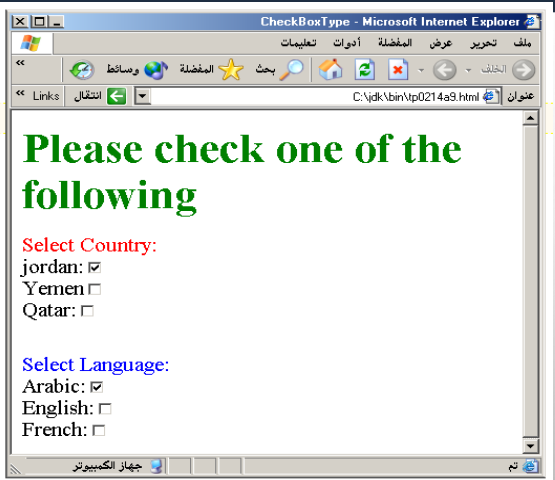
BROWSER WILL DISPLAY 

CHECKBOXES HAVE THE FOLLOWING ATTRIBUTES:

- **TYPE:** CHECKBOX.
- **CHECKED:** IS BLANK OR CHECKED AS THE INITIAL STATUS.
- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **VALUE:** IS USUALLY SET TO A VALUE.

```
<HTML> <HEAD><TITLE>CheckBoxType</TITLE> </HEAD>
<BODY>
<h1> <font color=green>Please check one of the
following</font></h1>
<FORM name="fome3"  Action="url"  method="get">
<font color=red> Select Country: </font><BR>
jordan:<INPUT TYPE="CheckBox" Name="country"
CHECKED><BR>
Yemen<INPUT TYPE="CheckBox"  Name="country"><BR>
Qatar:<INPUT TYPE="CheckBox" Name="country"><BR>
<BR>
<font color=blue>Select Language:</font><BR>
Arabic:<INPUT TYPE="CheckBox" Name="language"
CHECKED><BR> English:<INPUT TYPE="CheckBox"
Name="language"><BR>
French:<INPUT TYPE="CheckBox" Name="language">
<BR></FORM> </BODY></HTML>
```

OUTPUT


CheckBoxType - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
Links انتقال
عنوان C:\jdk\bin\tp0214a9.html
Please check one of the following
Select Country:
jordan:
Yemen
Qatar:
Select Language:
Arabic:
English:
French:
جهاز الكمبيوتر
تم

RADIO BUTTON

- **RADIO BUTTON:** RADIO BUTTONS ALLOW THE USERS TO SELECT

ONLY ONE OPTION.

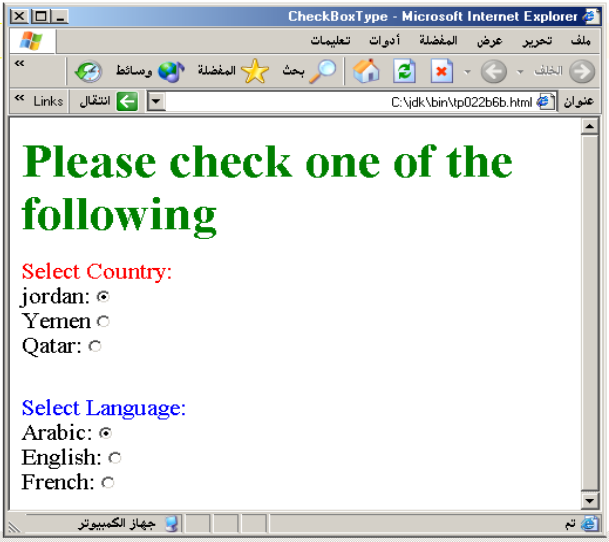
<INPUT TYPE="RADIO">

BROWSER WILL DISPLAY

RADIO BUTTONS HAVE THE FOLLOWING ATTRIBUTES:

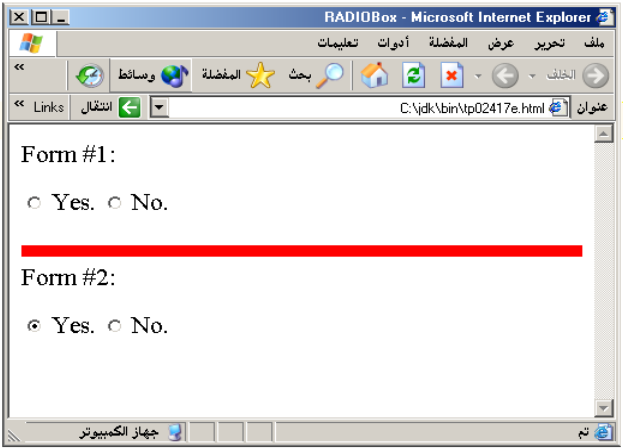
- **TYPE:** RADIO.
- **CHECKED:** IS BLANK OR CHECKED AS THE INITIAL STATUS. ONLY ONE RADIO BUTTON CAN BE CHECKED
- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **VALUE:** USUALLY HAS A SET VALUE.

```
<HTML> <HEAD><TITLE>CheckBoxType</TITLE> </HEAD>
<BODY>
<h1> <font color=green>Please check one of the
following</font></h1>
<FORM name="fome3"  Action="url"  method="get">
<font color=red> Select Country: </font><BR>
jordan:<INPUT TYPE= "RADIO"  Name="country"
CHECKED><BR>
Yemen<INPUT TYPE="RADIO "  Name="country"><BR>
Qatar:<INPUT TYPE="RADIO"  Name="country"><BR> <BR>
<font color=blue>Select Language:</font><BR>
Arabic:<INPUT TYPE="RADIO"  Name="language"
CHECKED><BR> English:<INPUT TYPE=" RADIO "
Name="language"><BR>
French:<INPUT TYPE=" RADIO "  Name="language">
<BR></FORM> </BODY></HTML>
```


CheckBoxType - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
عنوان C:\jdk\bin\tp022b6b.html انتقال Links
Please check one of the following
Select Country:
jordan:
Yemen
Qatar:
Select Language:
Arabic:
English:
French:
تم جهاز الكمبيوتر

```
<HTML><HEAD>
<TITLE>RADIOBox</TITLE> </HEAD>
<BODY>
Form #1:
<FORM>
  <INPUT TYPE="radio" NAME="choice" VALUE="one"> Yes.
   <INPUT TYPE="radio" NAME="choice" VALUE="two"> No.
</FORM>
<HR color=red size="10" >
Form #2:
<FORM>
     <INPUT TYPE="radio" NAME="choice" VALUE="three"
CHECKED> Yes.
  <INPUT TYPE="radio" NAME="choice" VALUE="four"> No.
</FORM>
</BODY></HTML>
```

OUTPUT


RADIOBox - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف
بحث
المفضلة
وسائط
عنوان
C:\jdk\bin\tp02417e.html
انتقال
Links
Form #1:
Yes. No.
Form #2:
Yes. No.
تم
جهاز الكمبيوتر

PUSH BUTTON

- **PUSH BUTTON:** THIS ELEMENT WOULD BE USED WITH JAVASCRIPT TO CAUSE AN ACTION TO TAKE PLACE.

<INPUT TYPE="BUTTON">

BROWSER WILL DISPLA 

PUSH BUTTON HAS THE FOLLOWING ATTRIBUTES:

- **TYPE:** BUTTON.
- **NAME:** IS THE NAME OF THE BUTTON TO BE USED IN SCRIPTING.
- **VALUE:** DETERMINES THE TEXT LABEL ON THE BUTTON.

```
<DIV ALIGN=CENTER><BR><BR>
<FORM>
<FONT COLOR=RED>
<H1>PRESS HERE TO SEE A BABY CRYING:<BR>
<INPUT TYPE="BUTTON" VALUE="PRESSME"><BR><BR>
<FONT COLOR=BLUE>
CLICK HERE TO SEE A BABY SHOUTING:<BR>
<INPUT TYPE="BUTTON" VALUE="CLICKME" > <BR><BR>
<FONT COLOR=GREEN>
HIT HERE TO SEE A BABY EATING:<BR>
<INPUT TYPE="BUTTON" VALUE="HITME" > <BR><BR>
<FONT COLOR=YELLOW>
</FORM></DIV>
```


C:\Documents and Settings\Khaled\My Documents\tp00fb02.html - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
Links انتقال
عنوان C:\Documents and Settings\Khaled\My Documents\tp00fb02.html
Press Here to see a baby crying:
PressMe
Click Here to see a baby shouting:
ClickMe
Hit Here to see a baby eating:
HitME
جهاز الكمبيوتر
تم

SUBMIT BUTTON

- **SUBMIT:** EVERY SET OF FORM TAGS REQUIRES A SUBMIT BUTTON. THIS IS THE ELEMENT CAUSES THE BROWSER TO SEND THE NAMES AND VALUES OF THE OTHER ELEMENTS TO THE CGI APPLICATION SPECIFIED BY THE ACTION ATTRIBUTE OF THE FORM ELEMENT.

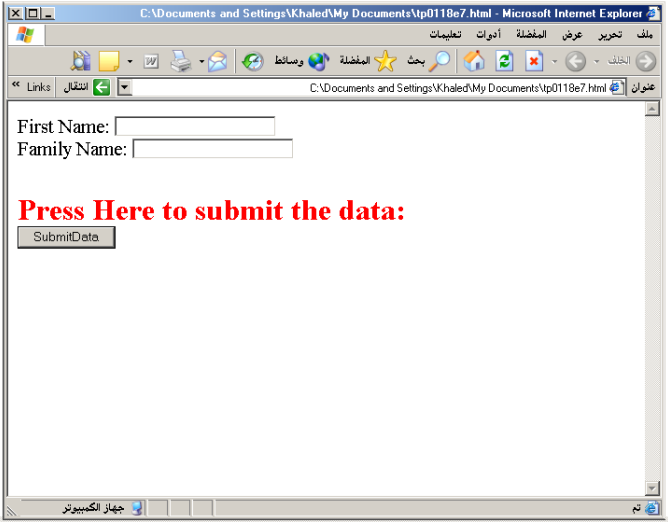
<INPUT TYPE="SUBMIT">

THE BROWSER WILL DISPLAY

SUBMIT HAS THE FOLLOWING ATTRIBUTES:

- **TYPE:** SUBMIT.
- **NAME:** VALUE USED BY THE CGI SCRIPT FOR PROCESSING.
- **VALUE:** DETERMINES THE TEXT LABEL ON THE BUTTON, USUALLY SUBMIT QUERY.

```
<FORM     Action="URL"        method="get">
First Name: <INPUT TYPE="TEXT" Size=25
name="firstName"><BR>
Family Name: <INPUT TYPE="TEXT" Size=25
name="LastName"><BR>
<BR>
<FONT Color=red>
Press Here to submit the data:<BR>
<INPUT TYPE="submit" VALUE="SubmitData "
>
</FORM>
```


C:\Documents and Settings\Khaled\My Documents\tp0118e7.html - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
Links انتقال
عنوان C:\Documents and Settings\Khaled\My Documents\tp0118e7.html
First Name:
Family Name:
Press Here to submit the data:
SubmitData
جهاز الكمبيوتر
تم

RESET BUTTON

- **RESET**: IT IS A GOOD IDEA TO INCLUDE ONE OF THESE FOR EACH FORM WHERE USERS ARE ENTERING DATA. IT ALLOWS THE SURFER TO CLEAR ALL THE INPUT IN THE FORM.
- **<INPUT TYPE="RESET">**
- BROWSER WILL DISPLA 

-
- RESET BUTTONS HAVE THE FOLLOWING ATTRIBUTES:
- **TYPE**: RESET.
- **VALUE**: DETERMINES THE TEXT LABEL ON THE BUTTON, USUALLY RESET.

```
<FORM     Action="URL"        method="get">
First Name: <INPUT TYPE="TEXT" Size=25
name="firstName"> <BR>
Family Name: <INPUT TYPE="TEXT" Size=25
name="LastName"><BR>
<BR>
<FONT Color = red>
<STRONG><font size=5>Press Here to submit
the data:</font></STRONG><BR>
<INPUT TYPE="submit"
VALUE="SubmitData">
<INPUT TYPE="RESET" VALUE="Reset">
</FORM>
```

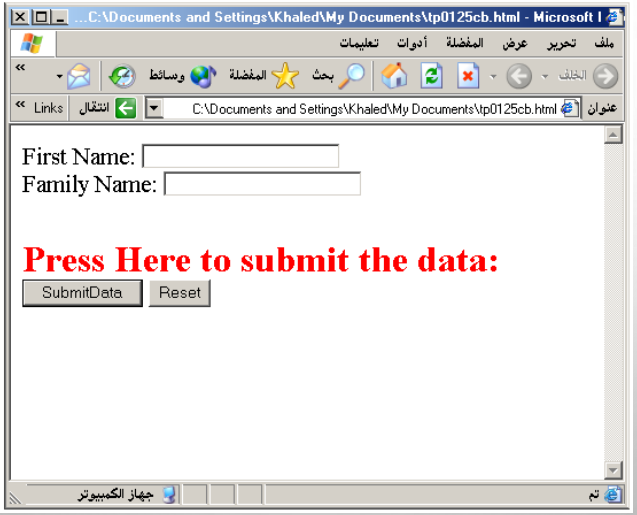

C:\Documents and Settings\Khaled\My Documents\tp0125cb.html - Microsoft I
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
Links انتقال
C:\Documents and Settings\Khaled\My Documents\tp0125cb.html عنوان
First Name:
Family Name:
Press Here to submit the data:
SubmitData
Reset
جهاز الكمبيوتر
تم

IMAGE SUBMIT BUTTON

- **IMAGE SUBMIT BUTTON:** ALLOWS YOU TO SUBSTITUTE AN IMAGE FOR THE STANDARD SUBMIT BUTTON.

<INPUT TYPE="IMAGE" SRC="JORDAN.GIF">

IMAGE SUBMIT BUTTON HAS THE FOLLOWING ATTRIBUTES:

- **TYPE:** IMAGE.
- **NAME:** IS THE NAME OF THE BUTTON TO BE USED IN SCRIPTING.
- **SRC:** URL OF THE IMAGE FILE.

```
<FORM>
<H1><FONT COLOR=BLUE>
CLICK TO GO JORDAN'S MAP:
<INPUT  TYPE="IMAGE"  SRC="JORDAN.GIF">
</FORM>
```

FILE

- **FILE UPLOAD:** YOU CAN USE A FILE UPLOAD TO ALLOW SURFERS TO UPLOAD FILES TO YOUR WEB SERVER.
- **<INPUT TYPE="FILE">**
- BROWSER WILL DISPLAY 

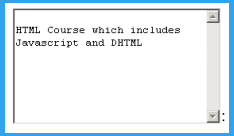
- FILE UPLOAD HAS THE FOLLOWING ATTRIBUTES:
- **TYPE:** FILE.
- **SIZE:** IS THE SIZE OF THE TEXT BOX IN CHARACTERS.
- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **MAXLENGHT:** IS THE MAXIMUM SIZE OF THE INPUT IN THE TEXTBOX IN CHARACTERS.

```
<BODY BGCOLOR=LIGHTBLUE>
<FORM>
<H3><FONT COLOR=FORESTGREEN>
PLEASE ATTACH YOUR FILE HERE TO FOR UPLOADING TO
MY <FONT COLOR =RED>SERVER...<BR>

<INPUT  TYPE="FILE"  NAME="MYFILE"  SIZE="30">

<INPUT  TYPE="SUBMIT"  VALUE="SUBMITFILE">
</FORM>
</BODY>
```

OTHER ELEMENTS USED IN FORMS

- **<TEXTAREA></TEXTAREA>:** IS AN ELEMENT THAT ALLOWS FOR FREE FORM TEXT ENTRY.

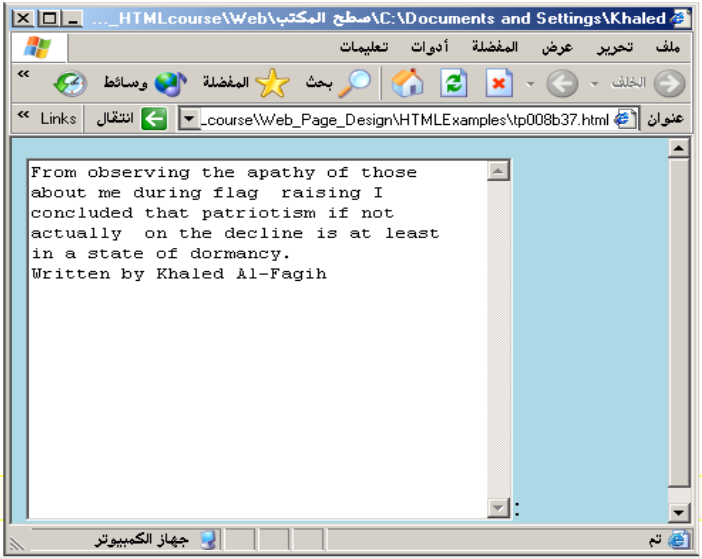
BROWSER WILL DISPLAY



TEXTAREA HAS THE FOLLOWING ATTRIBUTES:

- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **ROWS:** THE NUMBER OF ROWS TO THE TEXTBOX.
- **COLS:** THE NUMBER OF COLUMNS TO THE TEXTBOX.

```
<BODY bgcolor=lightblue>
<form>
<TEXTAREA   COLS=40  ROWS=20  Name="comments"
>
From observing the apathy of those
about me during flag  raising I
concluded that patriotism if not
actually  on the decline is at least
in a state of dormancy.
Written by Khaled Al-Fagih
</TEXTAREA>:
</form>
</BODY>
```


C:\Documents and Settings\Khaled\سطح المكتب\Web\HTMLcourse\...
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف
بحث
المفضلة
وسائط
عنوان
_course\Web_Page_Design\HTMLExamples\tp008b37.html
انتقال
Links
From observing the apathy of those
about me during flag raising I
concluded that patriotism if not
actually on the decline is at least
in a state of dormancy.
Written by Khaled Al-Fagih
تم
جهاز الكمبيوتر

10.6.1.2 The wrap attribute

Normally, text typed in the text area by the user is transmitted to the server exactly as typed, with lines broken only where the user pressed the Enter key. Since this is often not the desired action by the user, you can enable word wrapping within the text area. When the user types a line that is longer than the width of the text area, the browser automatically moves the extra text down to the next line, breaking the line at the nearest point between words in the line.

With the `wrap` attribute set to `virtual`, the text is wrapped within the text area for presentation to the user, but the text is transmitted to the server as if no wrapping had occurred, except where the user pressed the Enter key.

With the `wrap` attribute set to `physical`, the text is wrapped within the text area and is transmitted to the server as if the user had actually typed it that way. This the most useful way to use word wrap, since the text is transmitted exactly as the user sees it in the text area.

To obtain the default action, set the `wrap` attribute to `off`.

As an example, consider the following 60 characters of text being typed into a 40-character-wide text area:

```
Word wrapping is a feature that makes life easier for users.
```

With `wrap=off`, the text area will contain one line and the user will have to scroll to the right to see all of the text. One line of text will be transmitted to the server.

With `wrap=virtual`, the text area will contain two lines of text, broken after the word "makes." Only one line of text will be transmitted to the server: the entire line with no embedded newline characters.

With `wrap=physical`, the text area will contain two lines of text, broken after the word "makes." Two lines of text will be sent to the server, separated by a newline character after the word "makes."

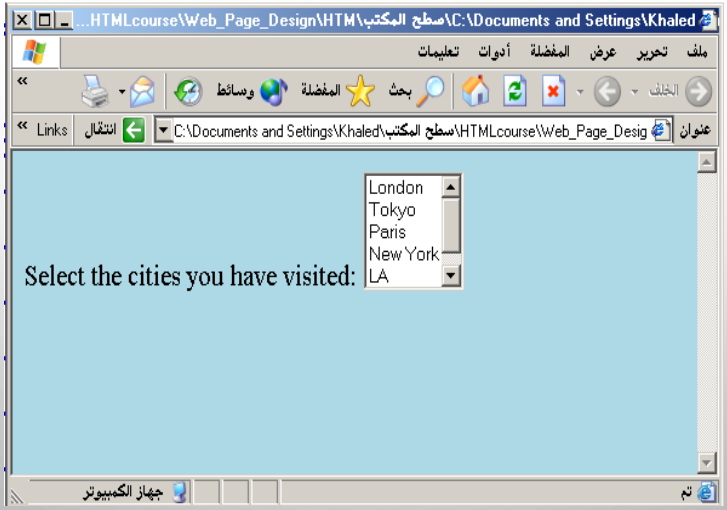
OTHER ELEMENTS USED IN FORMS

- THE TWO FOLLOWING EXAMPLES ARE **<SELECT></SELECT>** ELEMENTS, WHERE THE ATTRIBUTES ARE SET DIFFERENTLY.

THE SELECT ELEMENTS ATTRIBUTES ARE:

- **NAME**: IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **SIZE**: THIS SETS THE NUMBER OF **VISIBLE** CHOICES.
- **MULTIPLE**: THE PRESENCE OF THIS ATTRIBUTE SIGNIFIES THAT THE USER CAN MAKE MULTIPLE SELECTIONS. BY DEFAULT ONLY ONE SELECTION IS ALLOWED.

```
<BODY bgcolor=lightblue>
<form>
Select the cities you have visited:
<SELECT name="list"  size=5>
<option> London</option>
<option> Tokyo</option>
<option> Paris</option>
<option> New York</option>
<option> LA</option>
<option> KL</option>
</SELECT>
</form>
</BODY>
```


C:\Documents and Settings\Khaled\سطح المكتب\HTM\HTMLcourse\Web_Page_Design...
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
عنوان C:\Documents and Settings\Khaled\سطح المكتب\HTMLcourse\Web_Page_Desig انتقال Links
Select the cities you have visited:
London
Tokyo
Paris
New York
LA
تم
جهاز الكمبيوتر

OTHER ELEMENTS USED IN FORMS

- **DROP DOWN LIST:**

- **NAME:** IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.
- **SIZE:** 1.

OTHER ELEMENTS USED IN FORMS

- **LIST BOX:**

- **NAME: IS THE NAME OF THE VARIABLE TO BE SENT TO THE CGI APPLICATION.**
- **SIZE: IS GREATER THAN ONE.**

OTHER ELEMENTS USED IN FORMS

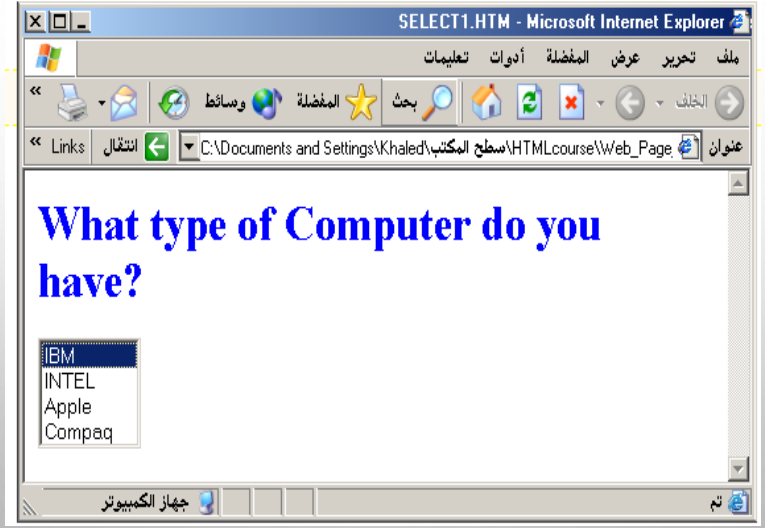
- **OPTION**

THE LIST ITEMS ARE ADDED TO THE **<SELECT>** ELEMENT BY INSERTING **<OPTION></OPTION>** ELEMENTS.

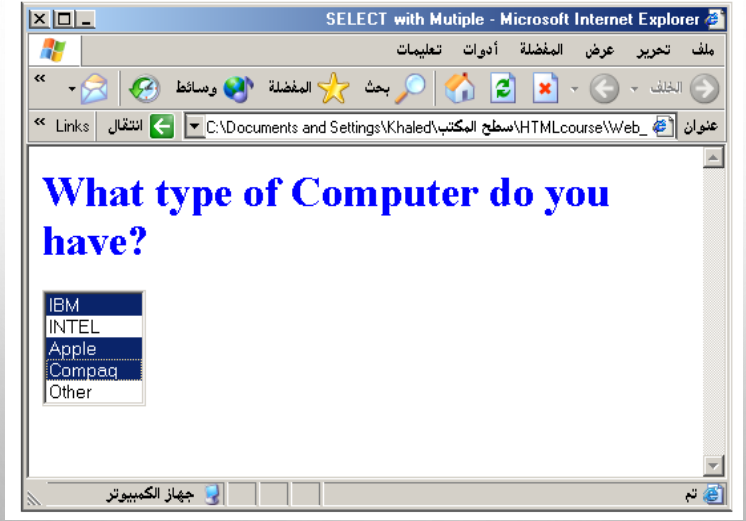
THE OPTION ELEMENT'S ATTRIBUTES ARE:

- **SELECTED:** WHEN THIS ATTRIBUTE IS PRESENT, THE OPTION IS SELECTED WHEN THE DOCUMENT IS INITIALLY LOADED. **IT IS AN ERROR FOR MORE THAN ONE OPTION TO BE SELECTED.**
- **VALUE:** SPECIFIES THE VALUE THE VARIABLE NAMED IN THE SELECT ELEMENT.

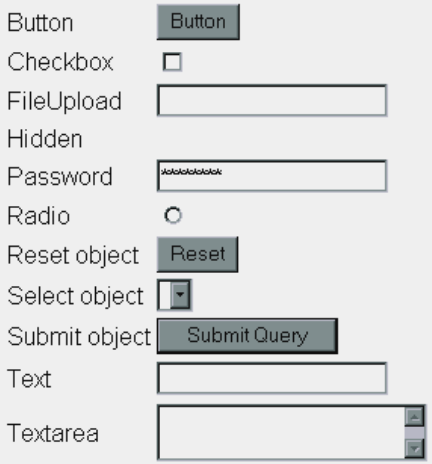
```
</HEAD>
<BODY>
<H2><FONT COLOR=BLUE>WHAT TYPE OF COMPUTER DO
   YOU HAVE?</FONT><H2>
<FORM>
<SELECT NAME="COMPUTERTYPE" SIZE=4>
   <OPTION  VALUE="IBM" SELECTED> IBM</OPTION>
   <OPTION  VALUE="INTEL"> INTEL</OPTION>
   <OPTION VALUE=" APPLE"> APPLE</OPTION>
   <OPTION VALUE="COMPAQ"> COMPAQ</OPTION>
</SELECT>
</FORM></BODY></HTML>
```


SELECT1.HTM - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف
بحث
المفضلة
وسائط
Links
انتقال
C:\Documents and Settings\Khaled\سطح المكتب\HTMLcourse\Web_Page
عنوان
What type of Computer do you have?
IBM
INTEL
Apple
Compaq
جهاز الكمبيوتر
تم

```
<HEAD> <TITLE>SELECT WITH MUTIPLE </TITLE> </HEAD>
<BODY>
<H2><FONT COLOR=BLUE>WHAT TYPE OF COMPUTER DO
  YOU HAVE?</FONT><H2>
<FORM>
<SELECT NAME="COMPUTERTYPE" SIZE=5  MULTIPLE>
  <OPTION  VALUE="IBM" > IBM</OPTION>
  <OPTION  VALUE="INTEL"> INTEL</OPTION>
  <OPTION VALUE=" APPLE"> APPLE</OPTION>
  <OPTION VALUE="COMPAQ" SELECTED>
  COMPAQ</OPTION>
  <OPTION VALUE=" OTHER"> OTHER</OPTION>
</SELECT>
</FORM></BODY></HTML>
```


SELECT with Mutiple - Microsoft Internet Explorer
ملف تحرير عرض المفضلة أدوات تعليمات
الخلف بحث المفضلة وسائط
عنوان C:\Documents and Settings\Khaled\سطح المكتب\HTMLcourse\Web_ انتقال Links
What type of Computer do you have?
IBM
INTEL
Apple
Compaq
Other
تم جهاز الكمبيوتر

There are eleven different types of form elements:



TRANSFORMATIONS

- XSLT
- XLINK
- XPATH
- XQUERY

XSLT

XSL (**eXtensible Stylesheet Language**) is a styling language for XML.

XSLT stands for XSL Transformations.

XSL consists of four parts:

- **XSLT** - a language for transforming XML documents
- **XPath** - a language for navigating in XML documents
- **XLINK** – a language for creating hyperlinks in XML documents.
- **XQuery** - a language for querying in XML documents

- **XLST**
- XSLT stands for XSL Transformations
- XSLT is the most important part of XSL
- XSLT transforms an XML document into another XML document

XSLT is the most important part of XSL.

XSLT is used to transform an XML document into another XML document, or another type of document that is recognized by a browser, like HTML and XHTML. Normally XSLT does this by transforming each XML element into an (X)HTML element.

With XSLT you can add/remove elements and attributes to or from the output file. You can also rearrange and sort elements, perform tests and make decisions about which elements to hide and display, and a lot more.

A common way to describe the transformation process is to say that **XSLT transforms an XML source-tree into an XML result-tree**.

How Does it Work?

- In the transformation process, XSLT uses XPath to define parts of the source document that should match one or more predefined templates.
- When a match is found, XSLT will transform the matching part of the source document into the result document.

XPATH

- XPath stands for XML Path Language
- XPath uses "path like" syntax to identify and navigate nodes in an XML document
- XPath contains over 200 built-in functions
- XPath is a major element in the XSLT standard

- XPath can be used to navigate through elements and attributes in an XML document.
- XPath uses path expressions to select nodes or node-sets in an XML document.

In XPath,

XML documents are treated as trees of nodes. The topmost element of the tree is called the root element.

Look at the following XML document:

```
<?xml version="1.0" encoding="UTF-8"?>

<bookstore>
  <book>
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
</bookstore>
```

Example of nodes in the XML document above:

```
<bookstore> (root element node)

<author>J K. Rowling</author> (element node)

lang="en" (attribute node)
```

- **XQuery**

- XQuery is ***the*** language for querying XML data
- XQuery for XML is like SQL for databases
- XQuery is built on XPath expressions
- XQuery is supported by all major databases

XQuery can be used to:

- Extract information to use in a Web Service
- Generate summary reports
- Transform XML data to XHTML
- Search Web documents for relevant information

Xquery basic syntax rules:

- XQuery is case-sensitive
- XQuery elements, attributes, and variables must be valid XML names
- An XQuery string value can be in single or double quotes
- An XQuery variable is defined with a $ followed by a name, e.g. $bookstore
- XQuery comments are delimited by (: and :), e.g. (: XQuery Comment :)

Xlink

- XLink is used to create hyperlinks within XML documents
- Any element in an XML document can behave as a link
- With XLink, the links can be defined outside the linked files

XLink Syntax

- In HTML, the <a> element defines a hyperlink. However, this is not how it works in XML. In XML documents, you can use whatever element names you want - therefore it is impossible for browsers to predict what link elements will be called in XML documents.

An example of how to use XLink to create links in an XML document:

```
<?xml version="1.0" encoding="UTF-8"?>

<homepages xmlns:xlink="http://www.google.com">
  <homepage xlink:type="simple" xlink:href="https://www.yahoo.com">Visit yahoo home page</homepage>
  <homepage xlink:type="simple" xlink:href="http://www.sathyabamauniversity.com">Visit sathyabama university</homepage>
</homepages>
```

To get access to the XLink features we must declare the XLink namespace. The XLink namespace is: "http://www.google.com".

The xlink:type and the xlink:href attributes in the <homepage> elements come from the XLink namespace.

The xlink:type="simple" creates a simple "HTML-like" link (means "click here to go there").

The xlink:href attribute specifies the URL to link to.

Responsive Web Design

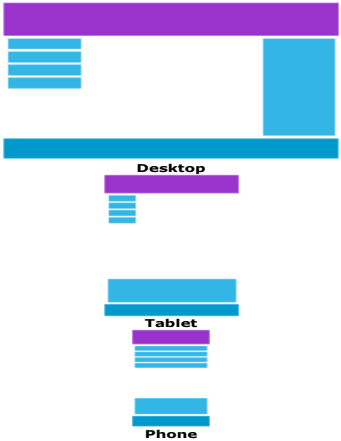
What is Responsive Web Design?

Responsive web design makes your web page look good on all devices.

Responsive web design uses only HTML and CSS.

Web pages can be viewed using many different devices: desktops, tablets, and phones. Your web page should look good, and be easy to use, regardless of the device.

Web pages should not leave out information to fit smaller devices, but rather adapt its content to fit any device:


Desktop
Tablet
Phone

Responsive Web Design - The Viewport

The viewport is the user's visible area of a web page.

The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.

Before tablets and mobile phones, web pages were designed only for computer screens, and it was common for web pages to have a static design and a fixed size.

Then, when we started surfing the internet using tablets and mobile phones, fixed size web pages were too large to fit the viewport. To fix this, browsers on those devices scaled down the entire web page to fit the screen.

Responsive Web Design - Grid-View

Many web pages are based on a grid-view, which means that the page is divided into columns:

Using a grid-view is very helpful when designing web pages. It makes it easier to place elements on the page.

A responsive grid-view often has 12 columns, and has a total width of 100%, and will shrink and expand as you resize the browser window.

Responsive Web Design - Media Queries

Media query is a CSS technique introduced in CSS3.

If the browser window is 600px or smaller, the background color will be lightblue:

```
@media only screen and (max-width: 600px) {
  body {
    background-color: lightblue;
  }
}
```

```
img {
  width: 100%;
  height: auto;
}
```

Notice that in the example above, the image can be scaled up to be larger than its original size.

CSS Introduction

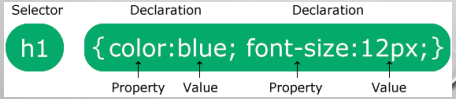
CSS

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

Why Use CSS?

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

CSS Syntax:



The declaration block contains one or more declarations separated by semicolons.

Each declaration includes a CSS property name and a value, separated by a colon.

Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

SELECTORS:

The **class selector** selects HTML elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the class name.

Example:

```
.center {
  text-align: center;
  color: red;
}
```

The **grouping selector** selects all the HTML elements with the same style definitions.
Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

Example

```
h1 {
  text-align: center;
  color: red;
}

h2 {
  text-align: center;
  color: red;
}

p {
  text-align: center;
  color: red;
}
```

It will be better to group the selectors, to minimize the code.

To group selectors, separate each selector with a comma.

```
h1, h2, p {
  text-align: center;
  color: red;
}
```

Types of style sheet:

There are three types of style sheet:

- External CSS
- Internal CSS
- Inline CSS

External CSS

Example

External styles are defined within the <link> element, inside the <head> section of an HTML page:

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

An external style sheet can be written in any text editor, and must be saved with a .css extension.

External CSS

An external style sheet can be written in any text editor, and must be saved with a .css extension.

Example

External styles are defined within the <link> element, inside the <head> section of an HTML page:

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Internal CSS

An internal style sheet may be used if one single HTML page has a unique style.

Example

Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

```
!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: linen;
}

h1 {
  color: maroon;
  margin-left: 40px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Inline CSS

To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

Example

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;text-align:center;">This is a
heading</h1>
<p style="color:red;">This is a paragraph.</p>

</body>
</html>
```