Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

Web Basics – HTML5

Lab Book

Document Revision History

Date	Revision No.	Author	Summary of Changes
2-Jun-05	1.0	Kumar B	Initial Draft
1-Oct-09	1.0	Ummeaiman Diwanji	Quality review. Transferred to the new template.
1-Oct-09	1.0	Anu Mitra	Final review
16/06/10	2	Anu Mitra	Refinements
27/05/11	3	Anu Mitra	Integration Refinements
21-Apr-2015	4	Rathnajothi P	Revamp/Refinement as per revised TOC
May-2016	4.1	Anjulata	Refinement as per revised TOC
July – 2020	4.2	Neelima	Refinement as per revised TOC

Table of Contents

Getting Started

Overview

This lab book is a guided tour for learning HTML. It comprises solved examples and 'To Do' assignments. Follow the steps provided in the solved examples and work out the 'To Do' assignments given.

Setup Checklist for HTML5

Here is what is expected on your machine in order for the lab to work.

Minimum System Requirements

- Intel Pentium 90 or higher (P166 recommended)
- Microsoft Windows 95, 98, or NT 4.0, 2k, XP.
- Memory: 32MB of RAM (64MB or more recommended)
- Internet Explorer 6.0 or higher

Please ensure that the following is done:

- A editor like Notepad, Eclipse, Visual Studio 2008 is installed.

Instructions

- For all coding standards refer Appendix A. All lab assignments should refer coding standards.
- Create a directory by your name in drive <drive>. In this directory, create a subdirectory html_assgn. For each lab exercise create a directory as lab <lab number>.
- You may also look up the on-line help provided in the MSDN library.
- The faculty will introduce you to the editor to be used.

Learning More (Bibliography)

- HTML Source Book by Ian S. Graham
- HTML: Complete Concepts and Techniques by Gary B. Shelly
- HTML: The Definitive Guide by Chuck Musciano
- Dynamic HTML: The Definitive Reference by Danny Goodman
- HTML: The Complete Reference by Thomas A. Powell

Lab 1: HTML Basics

Goals	• Understand the process of creating an HTML page and viewing it in a browser window. • Learn to apply physical or logical character effects. • Learn to manage document spacing
Time	10 minutes

Problem 1: Resume Creation

Problem Statement:

Create your resume page as per the format details given below :

- **Name:** Kalpana Chawla
- **Position:** Software Developer
- **About:** Enthusiastic software engineer with 3+ years experience participating in the complete SDLC of successfully launched applications. Eager to join ABC Inc. to deliver mission-critical technology and business solutions to Fortune 500 companies and some of the most recognized brands in the world. In my previous roles, reduced downtime by 15% and warranty costs by 30%; identified and resolved a process bottleneck that reduced coding efficiency by up to 25%.
- **Education:**

  College / University: NIET, NOIDA
  Graduation: B.E (Computer Science)
  Year: July 2015 - July 2019

  College / University: IIT, Powai
  Graduation: M.Tech (Computer Science)
  Year: July 2019 - July 2021

- **Skills:**
  1. Python
  2. JavaScript
  3. Django / Flask
  4. Angular / React

Note :

There should be **four DIV** tags .

- First **DIV** tag with **ID name-section** should contain:-
  1. **H1** tag with **Name** value
  2. **H2** tag with **Position** value**.**
- Second **DIV** tag with **ID about-section** should contain :-
  1. **P** tag with **About** value**.**
- Third **DIV** tag with **ID education-section** contains:-
  1. **H2** tag with text **"Education"**.

2. **HR** tag.
3. **TABLE** tag with two **rows** each row contains :-
4. **TR** tag with **College/University** value.
5. **TD** tag with **Graduation** value.
6. **TD** tag with **Year** value.

- Fourth **DIV** tag with **ID skills-section** should contain:-
  1. **H2** tag with text **"Skills".**
  2. **HR** tag**.**
  3. **UL** tag contains **Skills** values as **list items.**

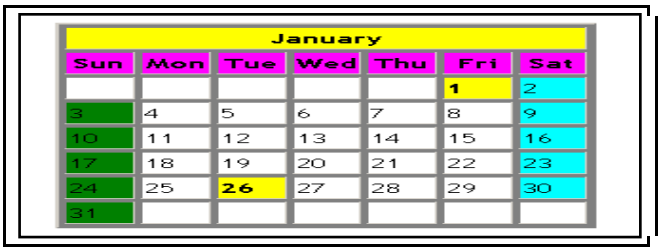
Lab 2: Creating Tables

Goals	At the end of this lab session you will understand: • Attributes of a Table • Table Headers • Table Data • Table Formatting • Control Table Borders • Grouping of Columns
Time	15 minutes

Problem 1: Calendar

Problem Statement:

Design a web page to display a calendar for a month using html table.



January						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure : Calendar

Note: Background colors to be used: For all the Sundays: green, for all the Saturdays: aqua, for 1, 26 Jan: yellow

Lab 3: Working with Lists

Goals	At the end of this lab session you will be able to use following types of lists: • Numbered List • Bulleted List • Directory List • Glossary List
Time	5 minutes

Problem 1: Subjects

Problem Statement:

Create a web page to display a list as shown in the figure that follows.

Display the list of Subjects using List

1. Language
    i. English
        A. Prose
        B. Poetry
    ii. Hindi
        A. Prose
        B. Poetry
    iii. Marathi
        A. Prose
        B. Poetry
2. Social Study
    i. History
    ii. Geography
3. Science
    i. Physics
        A. Part1
        B. Part2
    ii. Chemistry
        A. Organic
        B. Inorganic
    iii. Biology
        A. Botany
        B. Zoology
4. Maths
    i. Algebra
    ii. Geometry

Figure: Subject list

Lab 4: Working with Links

Goals	At the end of this lab session you will be able to: • Create links to web documents. • Create links to email. • Create hyperlinks for lists and table data. • Provide target for hyperlink.
Time	20 minutes

Problem 1: Welcome to Big Company

Problem Statement:

Design a simple home page for a company with a heading and 3 links – About, Products, Contact as given in the figure below.

Figure : Big Company home page

When you click the "About" hyperlink, following page should be displayed.

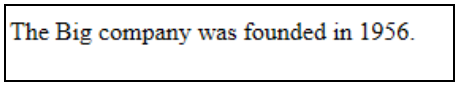


Figure: About

When you click the **Back** button on the browser toolbar, they should be redirected to the page *prob1.html*. Click the "Products" hyperlink to reach the following page:

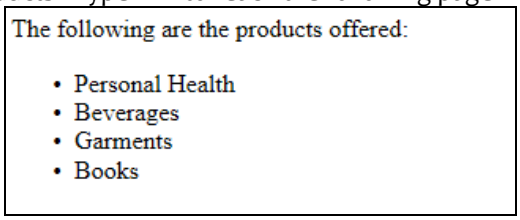


Figure: Products

When you click the **Back** button on the browser toolbar, they are redirected to page *prob1.html*. Click the "Contact" hyperlink. It opens Outlook Express and the e-mail address given in the *To* field, which is lnd.in@capgemini.com in the following illustration, is displayed in the New message window. This email address is specified in the *mailto* attribute.

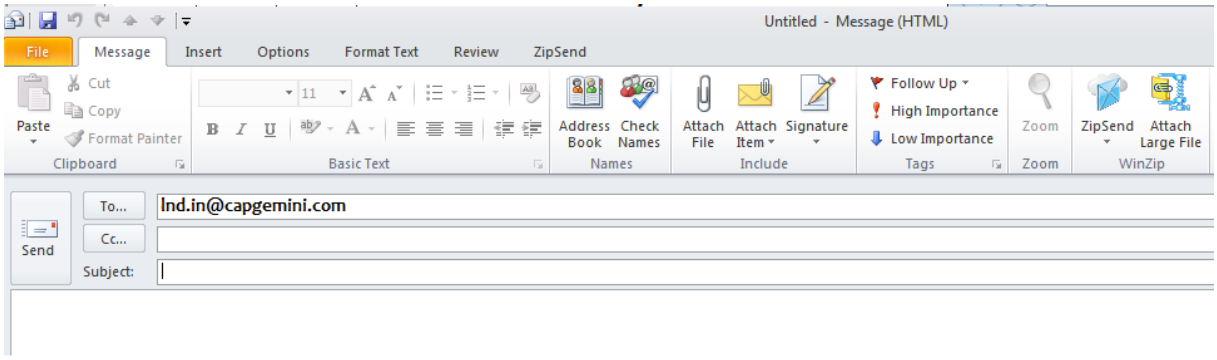


Figure: Contact

Problem 2: Employee Details

Problem Statement:

Design a simple home page for a company to display employee details as given below.

Empcode	Emp Name	Dept Code	Experience
1001	Kiran Rao	10	8 Yrs.
1002	Aamir Khan	20	5 Yrs.
1003	Ishita Shah	30	10 Yrs.

Figure : EmployeeDetails

When you click the department code "10" hyperlink, page with following content should be displayed.

This is Sales department located at Mumbai...

Figure: Sales Department

When you click the department code "20" hyperlink, page with following content should be displayed.

This is training department located at Pune...

Figure: Training Department

When you click the department code "30" hyperlink, page with following content should be displayed.

This is accounts department located at Chennai...

Figure: Accounts Department

Lab 5: Image Handling

Goals	At the end of this lab session you will be able to: • Understand the use of inline images. • Attributes of an inline image. • Text and image aligning. • Use of an image as a hyperlink.
Time	15 minutes

Problem 1: Images with Clickable Areas

Problem Statement:

Create a web page with some images as shown in following figure:

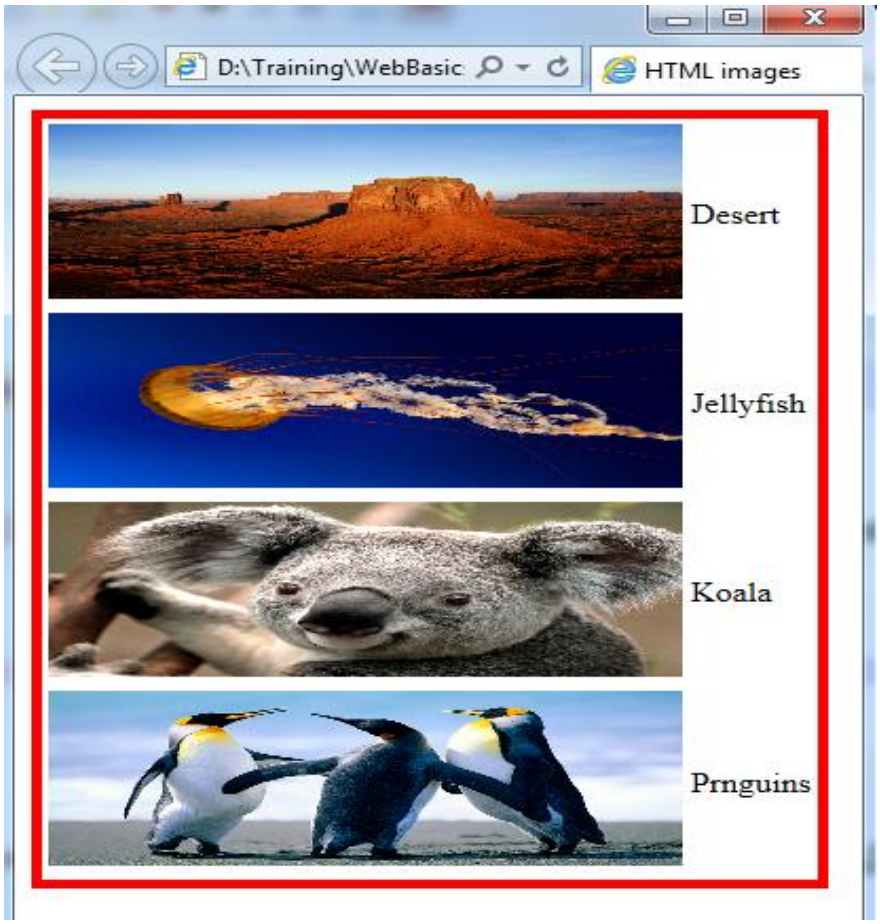


Figure 1 : Images

Lab 6: HTML Forms for User Input

Goals	At the end of this lab session you will be able to: • Understand the role of forms in web pages. • Understand various HTML elements used in forms. • Develop HTML forms in web pages.
Time	15 minutes

Problem 1: Employee Details

Problem Statement:

Design a web page *prob2.html* to accept the following employee details:

- Employee Name (Max 20 characters).
- Employee Code (Max 4 characters).
- Department (Use radio buttons).
- Date of Join (Use the format dd/mm/yyyy).
- Address.
- Training programs attended (Use check boxes).
- Training programs need to attend (Use select box).
- Send the information at empinfo@capgemini.com.

Name	
Employee Code	
Department	○ Admin ○ HR ○ Technical ○ Accounts
date of Joining	1 01 2014 dd mm YYYY
Address	
Training Program attended	☐ HTML/DHTML ☐ Java ☐ Client/Server ☐ .NET
Training program need to attend	JavaScript
Send the information at	empinfo@capgemini.com

Figure : Employee Details

Lab 7: New Form Elements

Goals	At the end of this lab session, you will be able to: • Develop web pages using HTML5 enhanced form elements
Time	20 minutes

Problem 1: Candidate Details

Problem Statement:

Design a web page StudentInfoForm.html to accept the following student details:

1. Name (Accept only characters , Max 15 characters)
2. Password (Max 15 characters)
3. Phone number(Accept 10 digits)
4. Gender (Make use of radio button)
5. Date of Birth (Make use of date field and date of birth should not be greater than current date)
6. Email (Accept valid Email)
7. Highest Qualification (Make use of datalist to populate data like B.Tech, M.Tech, MBA, MCA, MSc, MA, BSC..)
8. Courses interested in (Make use of check box)
9. Comments to mention regarding Degree / External Certificates (Make use of textarea)
10. Uploading Degree / External certificates (Make use of file input type)
11. Use Placeholders to describe the type of input.
12. All fields marked (*) are mandatory

Candidate Information

Name: *	
Password: *	
Phone number: *	
Gender: *	Male Female
Date of Birth: *	mm/dd/yyyy
Email: *	
Highest Qualification: *	Select Highest Qualification
Courses Interested in: *	Java HTML 5 CSS 3 Angular JS JQuery
Comments: (Mention External Certifications if any)	
Upload Degree / External Certificates: *	Choose Files No file chosen
	Submit Information Clear

All fields marked (*) are mandatory

Figure : Candidate Details

Appendix A: HTML Standards

Key Things To Keep In Mind:

- HTML standards help you reach the widest possible audience.
- There are many technologies that are *associated* with HTML because they are used on a Web page or in conjunction with HTML. But these technologies are *not* HTML:
    - CGI (Common Gateway Interface)
    - Java
    - JavaScript(JavaScript is also *not* Java)
    - Dynamic HTML (DHTML)
    - XML (Extensible Markup Language)
    - A variety of other emerging technologies

      For each of it, please follow the coding conventions, specified by that technology.
- Sometimes you need to break the rules and use non-standard syntax for good reasons. Try to keep this to a minimum.

How to Follow HTML Standards

Identify which version of HTML you are using in your document through the DOCTYPE line at the top of your file.

See the W3C site for more information on document types and DOCTYPE statements.

The important thing to remember is that a DOCTYPE statement is essential to assist validation software in checking your document.

- Use tools (supported by W3C) that support standards. In particular, install and use the *Tidy* program or *Tidy GUI* on your computer.
- Use W3C validation markup service to check the syntax of documents you create.
- Refer to W3C for technical and syntax information.

Some Simple HTML standards:

- The names of HTML files should always end with the ".html" extension.

  Example:
  Good: foo.html
  Bad: foo.bar

- Always include a <HTML> tag at the very beginning and a </HTML> tag at the very end of your HTML documents.

- Always use the <HEAD> and </HEAD> tags to define a header section in your HTML documents.

- Always give your documents a title by using the <TITLE> and </TITLE> tags in the header section of your HTML documents.

- Always use the <BODY> and </BODY> tags to define the body in your HTML documents, which is everything in your document between the <HTML> and </HTML> that is not contained in your header section.

- Use the horizontal line tag <HR> to place a horizontal line beneath any prominent headers in your documents to help them stand out from the surrounding information.

  Example:
  <H1>My Document's Title</H1>
  <HR>

- Always include a LINK with REV="MADE" in the header section of your HTML documents identifying you as the author.

  Example:
  <LINK REV="MADE" HREF="mailto:your_logonid@cs.niu.edu">

- Reasonable line lengths (no greater than 80 characters).

- Attributes associated with tags must be enclosed in quotes.

  Example:
  <img src="/images/gelogo.gif" width="200" height="100" alt="My Logo">

- Code is written in a consistent case. All command tags should be completely capitalized, in order for the tags to stand out better from the surrounding text.

  Example:
  Good: This text is <EM>emphasized</EM>.
  Bad: This text is <Em>emphasized.

- All code should include comment tags for readability, particularly when nested tables are used.

- Images have *alt, height*, and *width* attributes. They must be placed in the same directory as the HTML files. These images must be referenced in the code as:

  Example:
  Good: <img src="filename.gif">
  bad: <img src="images/filename.gif">.

- Links are coded correctly. All "HREF=" fields in anchor tags should always be enclosed in quotes.

Example:
Good:<A HREF="http://www.cs.niu.edu/homepage.html">
Bad: <A HREF=http://www.cs.niu.edu/homepage.html>

- Confirm that ©, ®, ™, and SM marks are coded correctly. These special characters should always be coded using their respective ASCII codes. It should also be confirmed that the superscription of these characters is done in a consistent manner.

  Example :
  Please code these special characters as follows:
  and Ampersand: andamp;
  © Copyright: and#169;
  ® Registration: and#174;
  ™ Trademark: and#153;

- Check links. There is nothing more frustrating to users than a broken link (except possibly the blink tag). If the review is of an entire site or a complete section of a site, it is helpful to use an automated link checker. Because there may be hundreds, or even thousands of links, the chance of missing one when checking them by hand is unacceptably high. Since Quality Assurance is not involved in the actual construction of a site, the producer/webmaster needs to verify that links are pointing to the correct pages that those pages still exist, etc.

- If you code a URL which does not specify a file name, always end the URL with a front slash (some browsers choke if you do not do this).

  Example:
  Good:<A HREF="http://www.ibm.com/">
  Good:<A HREF="http://www.cs.niu.edu/~www/">
  Bad:<A HREF="http://www.ibm.com">
  Bad:<A HREF="http://www.cs.niu.edu/~www">

- Whenever possible, use logical formatting tags instead of physical —one. Let the client's browser figure out the best way to display the information.

  Preferred: You should read the book <CITE>Neuromancer</CITE>
  Preferred: This text should <EM>stand out</EM>
  Discouraged: You should read the book <I>Neuromancer</I>
  Preferred: This text should <BOLD>stand out</BOLD>

- Always "sign" any HTML documents that you create. Include a horizontal line and a link to your homepage (using the ADDRESS style) at the very bottom.

  Example:
  ...and this is the end of my document's text.<P>
  <HR>
  <A HREF="http://www.cs.niu.edu/~www/"> <ADDRESS> WWW</ADDRESS></A>
  </BODY>
  </HTML>