AN INDUSTRIAL TRAINING REPORT

on

Basic of Web Development

Submitted in the fulfilment of the requirement
for the award of the degree of

Bachelor of Technology

in

INFORMATION TECHNOLOGY

by

Shubangirani S. Kulkarni
(Roll No. 20170776)

DEPARTMENT OF INFORMATION TECHNOLOGY
DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY
Lonere-402103, Tal.-Mangaon, Dist.-Raigad (M.S.) INDIA.

2019-2020

Certificate

The report submitted by Basic of Web Development Shubangirani S. Kulkarni (Roll No. 20170776) is approved for the partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Information Technology.

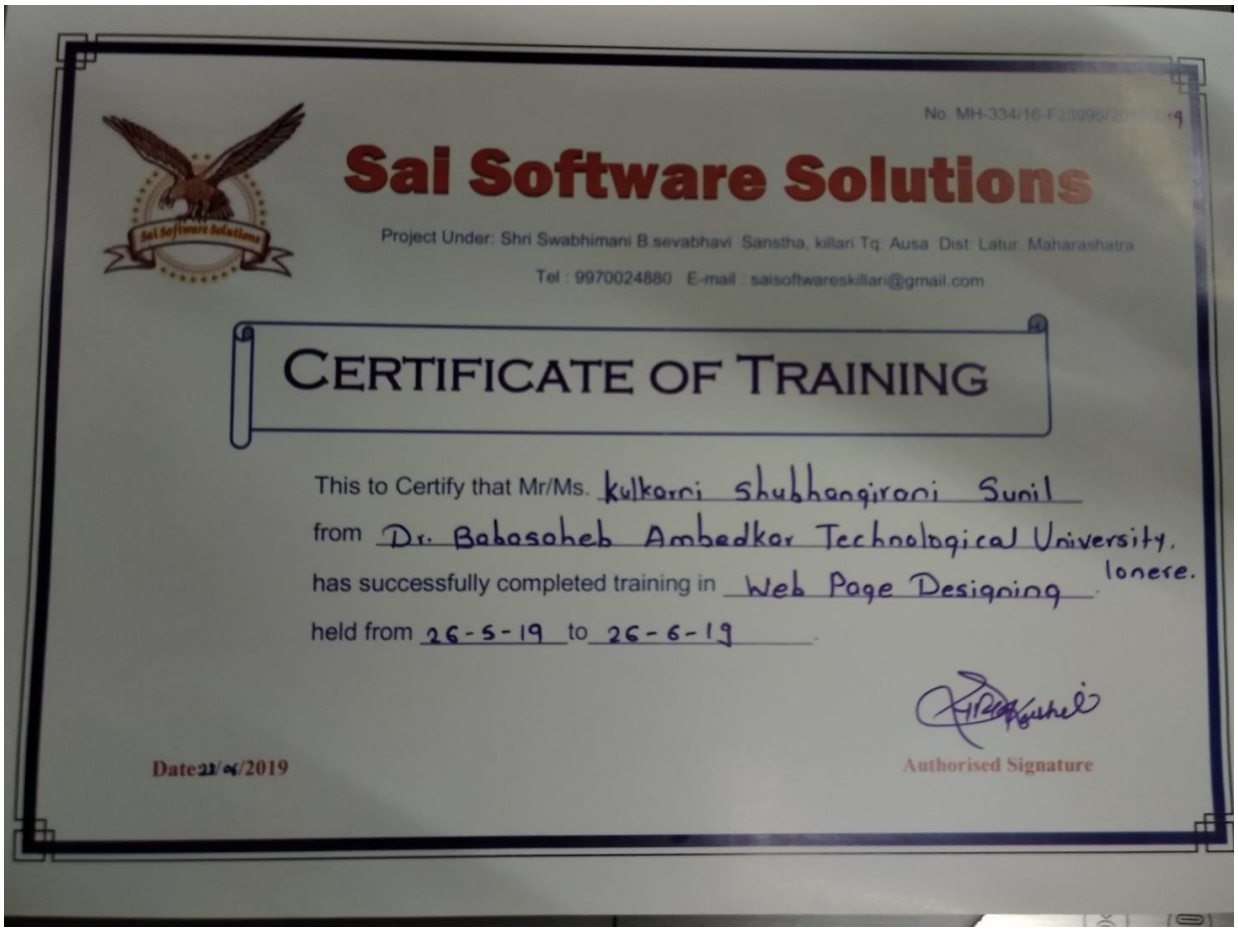
(Dr. S. M. Jadhav)
Head
Department of Information Technology

Examiner:

1. ----------(Name: Prof. E. G. Meshram)

Place: Dr. Babasaheb Ambedkar Technological University, Lonere.
Date:-

Certificate



No. MH-334/16-F23096/2019

Sai Software Solutions

Project Under: Shri Swabhimani B.sevabhavi Sanstha, killari Tq: Ausa Dist: Latur. Maharashtra

Tel : 9970024880 E-mail : saisoftwareskillari@gmail.com

CERTIFICATE OF TRAINING

This to Certify that Mr/Ms. Kulkarni Shubhangirani Sunil

from Dr. Babasaheb Ambedkar Technological University, lonere.

has successfully completed training in Web Page Designing.

held from 26-5-19 to 26-6-19.

Date 22/07/2019

Authorised Signature

Acknowledgement

This work is just not an individual contribution till its completion. I take this opportunity to express a deep gratitude towards my teacher, for providing excellent guidance encouragement and inspiration throughout the Training work. Without his invaluable guidance, this work would never have been a successful one.

I wish to express my sincere thanks to the all department faculties and staff members for their support. I would also like to thank my all classmates for their valuable guidance and helpful discussion.

Shubangirani S. Kulkarni

Abstract

The key to having successful and fully functional web applications is in their communication with the user. It is of no surprise that human/machine interaction is a popular topic of research and development.

A solution has been proposed for the internationalisation and customization of web applications in general. Taking into account web standards and the back-end and front-end architecture of web development, a method for providing a multi-language web interface has been planned from scratch and explained in details in this report.

Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 What is Web Technology?

Web Technology Introduction Web servers and web browsers are communicating client-server computer programs for distributing documents and information, generally called web data, over the Internet. Web data are marked up in the HTML language for presentation and interaction with people in web browsers.

Each web server uses an IP address or domain name as well as a port number for its identification. People use web browsers to send data requests to web servers with the HTTP protocol, and the web servers running on server computers either retrieve the requested data from local disks or generate the data on-the- y, mark up the data in HTML, and send the resulting HTML les back to the web browsers to render. Apache, Tomcat and IIS are popular web server programs, and IE and Firefox are popular web browsers. The term Web 2.0 was coined in 1999 to describe web sites that use technology beyond the static pages of earlier web sites.

It is closely associated with Tim O'Reilly because of the O'Reilly Media Web 2.0 conference which was held in late 2004. Although Web 2.0 suggests a new version of the World Wide Web, it does not refer to an update to any technical specification, but rather to cumulative changes in the ways software developers and end users use the Web. The word technology refers to the making, modification, usage, and knowledge of tools, machines, techniques, crafts, systems, and methods of organization, in order to solve a problem, improve a pre-existing solution to a problem, achieve a goal, handle an applied input/output relation or perform a specific function.

It can also refer to the collection of such tools, including machinery, modifications, arrangements and procedures. Technologies significantly enact human as well as other animal species' ability to control and adapt to their natural environments. The term can either be applied generally or to specific areas. Web design is the visual aesthetics and page layout of a website. It goes hand-in-hand with web development in the creation of a static website or dynamic web application.

Even if you dont consider yourself a creative person, its still a good idea to learn web design. No matter whether you want to make static websites or dynamic web applications, design is an important part of the process. The design is the rst thing people notice when they arrive on a website, and if its not good enough, theyll leave. You dont want to create a site that frustrates people, do you? If you do identify yourself as a creative person, you might even consider a career as a web designer. Today, there is a huge need for competent web designers with up-to-date skills. Its not an opportunity to be sneezed at. Learn coding and youll be a valuable asset to many. But learn coding and design and youll be second to none!

1.2 Developement Tools

programmers have a variety of tools available to enhance the software development process. Some common tools include:

1. Editors:- An editor allows the programmer to enter the program source code and save it to les. Most programming editors increase programmer productivity by using colors to highlight language features. The syntax of a language refers to the way pieces of the language are arranged to make well-formed sentences. To illustrate, the sentence

The tall boy runs quickly to the door.

uses proper English syntax. By comparison, the
sentence Boy the tall runs door to quickly the.

is not correct syntactically. It uses the same words as the original sentence, but their arrangement does not follow the rules of English. Similarly, programming languages have strict syntax rules that must be followed to create well- formed programs. Only well-formed programs are acceptable and can be compiled and executed. Some syntax-aware editors can use colors or other special annotations to alert programmers of syntax errors before the program is compiled.

2. Compiler:- A compiler translates the source code to target code. The target code may be the machine language for a particular platform or embedded device. The target code could be another source language; for example, the earliest C++ compiler translated C++ into C, another higher-level language. The resulting C code was then processed by a C compiler to produce an executable program. (C++ compilers today translate C++ directly into machine language.)

3. Interpreters:- An interpreter is like a compiler, in that it translates higher-level source code into machine language. It works di erently, however. While a compiler produces an executable program that may run many times with no additional translation needed, an interpreter translates source code statements into machine language as the program runs.

A compiled program does not need to be re- compiled to run, but an interpreted program must be interpreted each time it is executed. In general, compiled programs execute more quickly than interpreted programs because the translation activity occurs only once. Interpreted programs, on the other hand,

can run as is on any platform with an appropriate interpreter; they do not need to be recompiled to run on a di erent platform. Python, for example, is used mainly as an interpreted language, but compilers for it are available. Interpreted languages are better suited for dynamic, explorative development which many people feel is ideal for beginning programmers.

4. Debuggers:- A debugger allows programmers to simultaneously run a program and see which source code line is currently being executed.

The values of variables and other program elements can be watched to see if their values change as expected. Debuggers are valuable for locating errors (also called bugs) and repairing programs that contain errors. (See Section 3.4 for more information about programming errors.)

5.Pro lers:- A pro ler is used to evaluate a program's performance. It indicates how many times a portion of a program is executed during a particular run, and how long that portion takes to execute. Pro lers also can be used for testing purposes to ensure all the code in a program is actually being used somewhere during testing. This is known as coverage.

It is common for software to fail after its release because users exercise some part of the program that was not executed any time during testing. The main purpose of pro ling is to nd the parts of a program that can be improved to make the program run faster.

Chapter 2

HTML

2.1 What is HTML?

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as img and input directly introduce content into the page. Other tags such as HTML since 1997. surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which a ects the behavior and content of web pages. Inclusion of CSS de nes the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational

2.2 HTML Tags

The following are some facts about HTML and XHTML tags:

Web pages are just plain text. You can view or edit the source code using any text editor. "Tags" provide web browsers with instructions about the web page, such as where to display images, and how the document is structured. Tags are always enclosed in angle brackets:<>.

Tags are comprised of elements and attributes. An element is an object on a page (such as a heading, paragraph, or image), and attributes are qualities that describe that element (such as width and height). Tags usually travel in pairs. An opening tag begins a section of page content, and a closing tag ends it. For example, to markup a section of text as a paragraph, you would open the paragraph with an opening paragraph tag and close it with a closing paragraph tag (closing tags always proceed the element with a /). A few tags are called non-container tags, because they don't contain any content - they stand alone. Examples are images and line breaks.

XHTML requires that all open tags must be closed, even if they're not container tags. Therefore, non-container tags end in />. For example, the tag for a line break is
. Tags in HTML are not case sensitive, but in XHTML all tags must be in lower case. Even when coding in HTML, you should get in the habit of writing tags in lower case.

White space is ignored by web browsers. So, if you hit the space bar multiple times within a document, only one of those spaces will actually be displayed by the browser. Tags can be nested. For example,this text is italicized and bold. Note that the order of nested tags is important: The container tags surrounding any content should be symmetrical.

2.3 HTML Comments

Comment is a piece of code which is ignored by any web browser. It is a good practice to add comments into your HTML code, especially in complex documents, to indicate sections of a document, and any other notes to anyone looking at the code. Comments help you and others understand your code and increases code readability.

HTML comments are placed in between tags. So, any content placed with-in tags will be treated as comment and will be completely ignored by the browser.

Tag	Name	Code Example	Browser View
<!--	**comment**	**<!**--This can be viewed in the **HTML** part of a document--**>**	Nothing will show (Tip)
<a -	**anchor**	**<a href**="http://www.domain.com/"> Visit Our Site****	Visit Our Site (Tip)
	bold	****Example****	**Example**
<big>	**big (text)**	**<big>**Example**</big>**	Example (Tip)
<body>	**body of HTML document**	**<body>**The content of your **HTML** page**</body>**	Contents of your web page (Tip)
 	line break	The contents of your page** **The contents of your page	The contents of your web page The contents of your web page
	center	** **This will center your contents** **	This will center your contents
<dd>	**definition description**	<dl> <dt>Definition Term</dt> **<dd>Definition of the term</dd>** <dt>Definition Term</dt> **<dd>Definition of the term</dd>** </dl>	Definition Term **Definition of the term** Definition Term **Definition of the term**
<dl>	**definition list**	**<dl>** <dt>Definition Term</dt> <dd>Definition of the term</dd> <dt>Definition Term</dt> <dd>Definition of the term</dd> **</dl>**	Definition Term Definition of the term Definition Term Definition of the term
<dt>	**definition term**	<dl> **<dt>Definition Term</dt>** <dd>Definition of the term</dd> **<dt>Definition Term</dt>** <dd>Definition of the term</dd> </dl>	**Definition Term** Definition of the term **Definition Term** Definition of the term
	emphasis	This is an ****Example**** of using the emphasis tag	This is an *Example* of using the emphasis tag
<embed>	**embed object**	**<embed** src="yourfile.mid" width="100%" height="60" align="center">	(Tip)

Figure 2.1: Document Structure 1

	font	**<font** face="Times New Roman" size="4">Example****	Example (Tip)
	font	**<font** face="Times New Roman" size="+3" color="#ff0000">Example****	Example (Tip)
<form>	**form**	**<form** action="mailto:you@yourdomain.com "> Name: <input name="Name" value="" size="10"> Email: <input name="Email" value="" size="10"> <input type="submit"> **</form>**	Name: Email: Submit (Tip)
<h1> <h2> <h3> <h4> <h5> <h6>	**heading 1** **heading 2** **heading 3** **heading 4** **heading 5** **heading 6**	** **Heading 1 Example** ** ** **Heading 2 Example** ** ** **Heading 3 Example** ** ** **Heading 4 Example** ** ** **Heading 5 Example** ** ** **Heading 6 Example** **	Heading 1 Heading 2 Heading 3 Heading 4 Heading 5 Heading 6
<head>	**heading of HTML document**	**<head>**Contains elements describing the document**</head>**	Nothing will show
<hr>	**horizontal rule**	**<hr />**	Contents of your web page (Tip) Contents of your web page
<hr>	**horizontal rule**	**<hr** width="50%" size="3" />	Contents of your web page Contents of your web page
<hr>	**horizontal rule**	**<hr** width="50%" size="3" noshade />	Contents of your web page Contents of your web page

Figure 2.2: Document structure 2

<hr> (Internet Explorer)	**horizontal rule**	**<hr** width="75%" color="#ff0000" size="4" />	Contents of your web page Contents of your web page
<hr> (Internet Explorer)	**horizontal rule**	**<hr** width="25%" color="#6699ff" size="6" />	Contents of your web page Contents of your web page
<html>	**hypertext markup language**	**<html>** <head> <meta> <title>Title of your web page</title> </head> <body>**HTML web page** contents </body> **</html>**	Contents of your web page
	italic	****Example****	*Example*
<img>	**image**	**<img** src="Earth.gif" width="41" height="41" border="0" alt="text describing the image" />	 (Tip)

Figure 2.3: Empty Tag 3

<input>	**input field**	Example 1: <form method=post action="/cgi-bin/example.cgi"> **<input** type="text" size="10" maxlength="30"> **<input** type="Submit" value="Submit"> </form>	Example 1: (Tip) Submit
<input> (Internet Explorer)	**input field**	Example 2: <form method=post action="/cgi-bin/example.cgi"> **<input** type="text" style="color: #ffffff; font-family: Verdana; font-weight: bold; font-size: 12px; background-color: #72a4d2;" size="10" maxlength="30"> **<input** type="Submit" value="Submit"> </form>	Example 2: (Tip) Submit
<input>	**input field**	Example 3: <form method=post action="/cgi-bin/example.cgi"> <table border="0" cellspacing="0" cellpadding="2"><tr><td bgcolor="#8463ff"><input type="text" size="10" maxlength="30"></td><td bgcolor="#8463ff" valign="Middle"> <input type="image" name="submit" src="yourimage.gif"></td></tr> </table> </form>	Example 3: (Tip) GO
<input>	**input field**	Example 4: <form method=post action="/cgi-bin/example.cgi"> Enter Your Comments: <textarea wrap="virtual" name="Comments" rows=3 cols=20 maxlength=100></textarea> **<input** type="Submit" value="Submit"> **<input** type="Reset" value="Clear"> </form>	Example 4: (Tip) Submit Clear
<input>	**input field**	Example 5: <form method=post action="/cgi-bin/example.cgi"> Select an option: <select> <option >option 1</option> <option selected>option 2</option> <option>option 3</option> <option>option 4</option> <option>option 5</option> <option>option 6</option> </select> **<input** type="Submit" value="Submit"> </form>	Example 5: Tip) Select an option: option 2 Submit
<input>	**input field**	Example 6:	Example 6: (Tip)

Figure 2.4: Tables 4

		<form method=post action="/cgi-bin/example.cgi"> Select an option: Select an option: 	Select an option: Option 1 Option 2 Option 3 Select an option: Selection 1 Selection 2 Selection 3 Submit
	list item	Example 1: <menu> <li type="disc">List item 1 <li type="circle">List item 2 <li type="square">List item 3 </MENU> Example 2: <ol type="i"> - List item 1 - List item 2 - List item 3 - List item 4 	Example 1: (Tip) **List item 1** **List item 2** **List item 3** Example 2: i. **List item 1** ii. **List item 2** iii. **List item 3** iv. **List item 4**
<link>	**link**	<head> <link rel="stylesheet" type="text/css" href="style.css" /> </head>	
<marquee> (Internet Explorer)	**scrolling text**	**<marquee** bgcolor="#cccccc" loop="-1" scrollamount="2" width="100%">Example Marquee**</marquee>**	(Tip)

Figure 2.5: Tables 5

<table>
<tr><td><strike></td><td>deleted text</td><td><strike>Example</strike></td><td><s>Example</s></td></tr>
<tr><td></td><td>strong emphasis</td><td>Example</td><td>Example</td></tr>
<tr><td><table></td><td>table</td><td>Example 1:

<table border="4" cellpadding="2" cellspacing="2" width="100%">
<tr>
<td>Column 1</td>
<td>Column 2</td>
</tr>
</table>

Example 2: (Internet Explorer)

<table border="2" bordercolor="#336699" cellpadding="2" cellspacing="2" width="100%">
<tr>
<td>Column 1</td>
<td>Column 2</td>
</tr>
</table>

Example 3:</td><td>Example 1: (Tip)<table><tr><td>Column 1</td><td>Column 2</td></tr></table>
Example 2: (Tip)<table><tr><td>Column 1</td><td>Column 2</td></tr></table>
Example 3: (Tip)<table><tr><td>Column 1</td><td>Column 2</td></tr></table></td></tr>
</table>

Figure 2.6: Tables 7

		<table cellpadding="2" cellspacing="2" width="100%"> <tr> <td bgcolor="#cccccc">Column 1</td> <td bgcolor="#cccccc">Column 2</td> </tr> <tr> <td>Row 2</td> <td>Row 2</td> </tr> **</table>**	Row 2 Row 2
<td>	**table data**	<table border="2" cellpadding="2" cellspacing="2" width="100%"> <tr> **<td>Column 1</td> <td>Column 2</td>** </tr> </table>	**Column 1** **Column 2**
<th>	**table header**	 <table> <tr> **<th>Column 1</th>** **<th>Column 2</th> <th>Column 3</th>** </tr> <tr> <td>Row 2</td> <td>Row 2</td> <td>Row 2</td> </tr> <tr> <td>Row 3</td> <td>Row 3</td> <td>Row 3</td>	**Colum n 1 Colum n 2 Colum n 3** Row 2 Row 2 Row 2 Row 3 Row 3 Row 3 Row 4 Row 4 Row 4

Figure 2.7: Tables 8

Chapter 3

CSS

CSS is a language that describes the style of an HTML document.

CSS describes how HTML elements should be displayed.

This tutorial will teach you CSS from basic to advanced.

3.1 What is CSS?

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page.

Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs,variations in display for di erent devices and screen sizes as well as a variety of other e ects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML. CSS is a language that describes the style of an HTML document.

CSS describes how HTML elements should be displayed. This tutorial will teach you CSS from basic to advanced. The main styling is stored in an external spreadsheet, although the HTML DOM style object has also been used to change some settings while the HTML is being loaded.

For example the Save button for the edit module has set visibility 8 to hidden in the external stylesheet, but the property changes to visible when the edit button is clicked to avoid potential confusion. Another method I have used to change properties inside JS is the jQuery .css() method. I have found it to be e ective in changing background and border properties of objects.

3.2 Advantages Of CSS

1.CSS saves time You can write CSS once and then reuse same sheet in multiple HTML pages. You can de ne a style for each HTML element and apply it to as many Web pages as you want.

2.Pages load faster If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

3.Easy maintenance To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

4.Superior styles to HTML CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

5.Multiple Device Compatibility Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, di erent versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

6.Global web standards Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

3.3 External CSS

With an external style sheet, you can change the look of an entire website by changing just one le! Each page must include a reference to the external style sheet le inside the <link> element. The<link> element goes inside the<head>section:

3.4 Internal CSS

An internal style sheet may be used if one single page has a unique style. Internal styles are de ned within the<style> element, inside the <head> section of an HTML page:

3.5 Inline CSS

An inline style may be used to apply a unique style for a single element. To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

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

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

</body>
</html>
```

Figure 3.1: external1 css

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color: powderblue;}
h1   {color: blue;}
p    {color: red;}
</style>
</head>
<body>

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

</body>
</html>
```

Figure 3.2: internal css

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

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

</body>
</html>
```

Figure 3.3: inline css

Chapter 4

BOOTSTRAP

Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter. It was released as an open source product in August 2011 on GitHub.

4.1 What is Bootstrap?

Bootstrap is a front-end framework that is developed to support creating dynamic websites and web applications. It is one of the most preferred front-end frameworks as it aids an easy and fast processing to develop a website. It supports all major browsers and fast loading responsive web pages.

Bootstrap Bootstrap consists of HTML and CSS-based design templates for various interface com-ponents and is aimed to ease web development. By updating the CSS, you can adapt to modern trends quickly. The developers should concentrate more on interaction components as the bootstrap itself will take care of standard views of data, which can be altered later if you wish to.

Bootstrap is compatible with almost all the latest version browsers such as Internet Explorer, Google Chrome, Opera, Firefox, and Safari. It supports the responsive web design and dynamically adjusts the layout of web pages by considering the characteristics of the device used.

4.2 Why use Bootstrap

Mobile rst approach Bootstrap 3, framework consists of Mobile rst styles throughout the entire library instead them of in separate les. Browser Support It is supported by all popular browsers. Like crome, refox, uc browser

Easy to get started With just the knowledge of HTML and CSS anyone can get started with Bootstrap. Also the Bootstrap o cial site has a good documentation. Responsive design Bootstrap's responsive CSS adjusts to Desktops, Tablets and Mobiles. More about the responsive design is in the

chapter Bootstrap Responsive Design.

Provides a clean and uniform solution for building an interface for developers. It contains beautiful and functional built-in components which are easy to customize. It also provides web based customiza-tion. And best of all it is an open source

4.3 What Bootstrap Package Includes?

1.Sca olding Bootstrap provides a basic structure with Grid System, link styles, and background.
This is is covered in detail in the section Bootstrap Basic Structure.

2.CSS Bootstrap comes with the feature of global CSS settings, fundamental HTML elements styled and enhanced with extensible classes, and an advanced grid system. This is covered in detail in the section Bootstrap with CSS.

3.Components Bootstrap contains over a dozen reusable components built to provide iconography, dropdowns, navigation, alerts, pop-overs, and much more. This is covered in detail in the section Layout Components.

4.JavaScript Plugins Bootstrap contains over a dozen custom jQuery plugins. You can easily include them all, or one by one. This is covered in details in the section Bootstrap Plugins.

5.Customize You can customize Bootstrap's components, LESS variables, and jQuery plugins to get your very own version.

4.4 Layout

Containers are the most basic layout element in Bootstrap and are required when using our default grid system. Choose from a responsive, xed-width container (meaning its max-width changes at each breakpoint) or uid-width (meaning its 100While containers can be nested, most layouts do not require a nested container.

.

4.5 Advantages of Bootstrap

1.Speed of Development

The speed of development is one of its major advantages. If you want to develop an application or a website promptly, it is imperative to consider using Bootstrap. It helps to save your coding e ort by o ering less CSS functionality and pre-built blocks of code rather than structuring code from the scratch. Ready-made themes of Bootstrap will help achieve your needs through a faster route.

2.Responsiveness

According to CISCOs predictions, global mobile data tra c will increase approximately 11-fold between 2013 and 2018. These statistics points to the need for a responsive website in varied kinds of mobile devices.

Bootstrap is equipped with responsive layout and 12-column grid system that help dynamically adjust the website to a suitable screen resolution. The responsive utility classes feature of Bootstrap enables you to hide / show a certain section of content for a particular screen size.

3.Consistency

Consistency was the fundamental principle behind the introduction of Bootstrap. It ensures the ultimate consistency regardless of designer/developer, who is working on it. Moreover, the results work uniformly across various browsers and the output remains same.

4.Customizable

Bootstrap facilitates abundant customization and helps developers in designing tailor made websites, according to their speci cations. It has the facility to select any feature that is actually needed to create a customized website. With this feature, one can get rid of what they do not require.

5.Support

Bootstrap helps to x issues promptly with an immense support community. Bootstrap also releases continual updates to x any new issues. Currently, it is being developed, hosted and maintained by GitHub with over 9000 commits and 500 contributors.

Bootstrap is an awesome framework with rich features. It is the latest in innovation for responsive development and supports designing of websites and apps faster, easier and better.

FacebookTwitterGoogle+Share0

Chapter 5

JAVASCRIPT

What is JavaScript?

JavaScript is a very powerful **client-side scripting language**. JavaScript is used mainly for enhancing the interaction of a user with the webpage. In other words, you can make your webpage more lively and interactive, with the help of JavaScript. JavaScript is also being used widely in game development and Mobile application development.

How to Run JavaScript?

Being a scripting language, JavaScript cannot run on its own. In fact, the browser is responsible for running JavaScript code. When a user requests an HTML page with JavaScript in it, the script is sent to the browser and it is up to the browser to execute it. The main advantage of JavaScript is that all modern web browsers support JavaScript. So, you do not have to worry about whether your site visitor uses Internet Explorer, Google Chrome, Firefox or any other browser. JavaScript will be supported. Also, JavaScript runs on any operating system including Windows, Linux or Mac. Thus, JavaScript overcomes the main disadvantages of VBscript (Now deprecated) which is limited to just IE and Windows.

Tools You Need

To start with, you need a text editor to write your code and a browser to display the web pages you develop. You can use a text editor of your choice including Notepad++, Visual Studio Code, Sublime Text, Atom or any other text editor you are comfortable with. You can use any web browser including Google Chrome, Firefox, Microsoft Edge, Internet Explorer etc.

A Simple JavaScript Program

You should place all your JavaScript code within **<script> tags** (<script> and </script>) if you are keeping your JavaScript code within the HTML document itself. This helps your browser distinguish your JavaScript code from the rest of the code. As there are other client-side scripting languages (Example: VBScript), it is highly recommended that you specify the scripting language you use. You have to use the type attribute within the <script> tag and set its value to text/javascript like this:

```
<html>
<head>
    <title>My First JavaScript code!!!</title>
    <script type="text/javascript">
        alert("Hello World!");
    </script>
</head>
<body>
</body>
</html>
```

Chapter 6

CONCLUSION

This course has been an excellent and rewarding experience. I can conclude that there have been a lot I've learnt from my work at there search centre. Needless to say, the technical aspects of the work I've done are not awless and could be improved provided enough time. As someone with no prior experience in html, css, bootstrap.

Whatsoever I believe my time spent in research and discovering new languages was well worth it and contributed to nding an acceptable solution to an important aspect of web design and development. Two main things that Ive learned the importance of are time-management skills and self-motivation. Although I have often stumbled upon these problems at University, they had to be approached di erently in a working environment.

I have yet to complete my studies, in order to achieve a bachelor degree in Computing (with Honours). Working with web development languages has increased my interest in them, hence prompting me to transfer to the Web Design and Development course at my university.