EDUCATIONAL WEBSITE

PROJECT REPORT

(BTCS 603-18/ BTIT 603-18)

Submitted in partial fulfillment of the

Requirements for the award of the degree

Of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

GGI
Sharpening Minds, Brightening Tomorrow.
Gulzar Group of Institutes

UDER THE GUIDANCE OF :   **STUDENT NAME(Roll. No.)**

Er. Harjit Kaur   **Aanand Yadav (1919420)**

Akanksha raj (1906045)

Rezaul islam(1906134)

Israel Makama(1906090)

I.K. GUJRAL PUNJAB TECHNICAL UNIVERSITY, JALANDHAR

CANDIDATE'S DECLARATION AND CERTIFICATE

_____

We hereby certify that the work, which is being presented in this report entitled, **Educational Website**, in partial fulfillment of the requirements for the degree of BTECH in the Computer Science Engineering, Gulzar Group of Institutes, Khanna, Punjab; by **Aanand Yadav (1919420), Akanksha Raj (1906045), Rezaul Islam(1906134)** and **Israel Makama (1906090)** authentic record of our own work carried out under the supervision of **Er. Harjit Kaur**, Gulzar Group of Institutes, Khanna, Punjab.

We further declare that the matter embodied in this report has not been submitted by us for the award of any other degree.

Candidate(s) Signature:

Aanand Yadav (1919420)

Aakanksha Raj (1906045)

Rezaul Islam (1906135)

Israel Makama (1906090)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Signature of HOD   **Signature of Supervisor**
Dr. Sukhminder Singh Grewal   **Er. Harjit Kaur**

Date:06/06/2022

ACKNOWLEDGMENT

It is our pleasure to acknowledge the contributions of all who have helped us and supported during this Project report.

First, we thank God for helping us in one way or another and providing strength and endurance to us. We wish to express my sincere gratitude and indebtedness to our supervisor, Er. Harjit Kaur, C.S.E & IT department, Gulzar Group of Institutes, Khanna, Punjab; for her intuitive and meticulous guidance and perpetual inspiration in completion of this report. In spite of his/her busy schedule, she rendered help whenever needed, giving useful suggestions and holding informal discussions. Her invaluable guidance and support throughout this work cannot be written down in few words. We also thank her and the department for providing facilities for our work. We would like to extend our sincerest gratitude to our H.O.D and Dean Dr. Sukminder Singh Grewal for his continued support to our endeavor throughout the semester as well as Er. Jaskiran Kaur and Dr. Sita Rani for their unrelenting effort to coordinate the project fulfillment.

We are also humbly obliged by the support of our group members and friends for their love and caring attitude. The sentimental support they rendered to us is invaluable and everlasting. They have helped us through thick and thin and enabled us to complete the work with joy and vigor. We thank the group members for entrusting in each other and following directions, without them this report would never have been possible.

We are also thankful to our parents, elders and all family members for their blessing, motivation and inspiration throughout our work and bearing with us even during stress and bad temper. They have always provided us a high moral support and contributed in all ways in completion of this Capstone report.

ABSTRACT

_____

Web development is one of field of computer science engineering that helps human to learn and share their knowledge universally. This was done to get better understanding of the skills that are needed in the area of website design and development. There were many things that were taken to submit a successful project such as Market Research, identifying audiences.it is used in different fields education to business purposes. The aim of this project to explore different technology. it is designed to help human to learn faster, free and right way. Initially, we discuss about the feasibility study and requirements analysis then install and configuration of important tools, created structure of site as well as design and finally we hosted the web site.

This educational website is project based on web development and related technology. This project has been design using visual studio code as IDE and HTML, CSS, JavaScript as language. The main moto of our project is to guide students to study grow with right decision. Website is easy to access and free for everyone

Finally, project was successful. We were creating structure of educational webpage. We were able to create home page and log in page. After completing this project, we conclude that it is possible to create an website more effectively. The entire process is very simple and user friendly with minimum coding. For the future, speed of website can be improved, or smooth and Server side(backend) also can be added to make dynamic webpage.

TABLE OF CONTENTS

Chapter No.	Title	Page No.

LIST OF FIGURES

LIST OF TABLES

TABLE No.   **DESCRIPTION**   **PAGE No.**

Chapter 1

1.Introduction:

Before creating any website, it is common to visualize the layout, the design and all the features you need intend. In software engineering established a list requirement for program is refers as developing the functional non-functional requirement [4]. Web is a subset of the internet. Web is made up of two components the front-end (client side) and back-end (server side). Client request to the server to get its data. To create web, different technology needed like HTML, CSS JavaScript (for frontend), database, API, PHP (for backend).

"*a set of detailed instructions, usually written on a manuscript to be typeset, concerning style of type, makeup of pages, and the like.[3]*"

A Educational Web Site is very useful Website for Study and growth of advanced carrier. In today's world online study is very wide trend. With the help of this website, you can access some free online courses with Note as yours required. here is different road map available for different field of computer science engineering. Users need to register on first visit of website. After all, need to log in using email address and password.

Software requirements:

- Visual studio code
- My SQL
- Windows 10

Hardware requirements:

- Processor i3 [7]
- Hard disk-5gb [7]
- Memory 1G [7]

1.1. AIM

2. This website describing a specific course (subject, syllabus, notes, project)
3. Study websites that include most or all the information needed but not designed around the specific course.
4. Education website about one branch computer science engineering
5. This website includes related current affairs
6. It includes interactive diagram.

7. It is also included quiz related to the topic.
8. It also gives monthly yearly progress report.
9. This website completely free for user.
10. users interact with the website.
11. loading speed is slow. website take a long time to load

Chapter 2

2. Literature Survey:

Survey research challenging and time-consuming process, but current technology offer education. here is how educational website is can benefit for user. user these day uses online mode of education to gain knowledge. They prefer to read online surf the internet and access website rather than prefer textbook. This website engaged people and help them in exciting way. Educational website with enticing interference helps them conveniently gain knowledge. Thus, student find it interesting and exciting to access educational website. There is various example of education website [6].

Khan academy:

it is a non-profit education website helps to learn school level math, art, computer, physics, chemistry, biology etc. for free.it play major role of flipping the classroom in many schools in the United States of America [5].

Udemy:

Udemy is online learning and teaching marketplace with a lot of courses to help student learn relevant skill in life such as marketing, programming, data science etc. the range of life skills it covers make it so special.[5]

Coursera:

It is online educational platform which provides free MOOCs in subject such as physics, computer science, mathematics, business etc. backed by top university in the respective subject. Here also courses are free student have to pay for the certificate.[5]

Big Think:

Big Think has over 2,000 fellows who have received great fame in their forte. It records tutorial and write content for student. Students can make great use of this website by creating their own distinct ideology, as it provides various opinions on one subject. Moreover, students can get views from experts as well.[6]

Chapter 3

3. SYSTEM DEVELOPMENT LIFE CYCLE:

An effective System Development Life Cycle (SDLC) should result in a high-quality system that meets customer expectations, reaches completion within time and cost evaluations, and works effectively and efficiently in the current and planned Information Technology infrastructure.

System Development Life Cycle (SDLC) is a conceptual model which includes policies and procedures for developing or altering systems throughout their life cycles.

SDLC is used by analysts to develop an information system. SDLC includes the following activities −

- requirements
- design
- implementation
- testing
- deployment
- operations
- maintenance

Systems Development Life Cycle is a systematic approach which explicitly breaks down the work into phases that are required to implement either new or modified Information System.
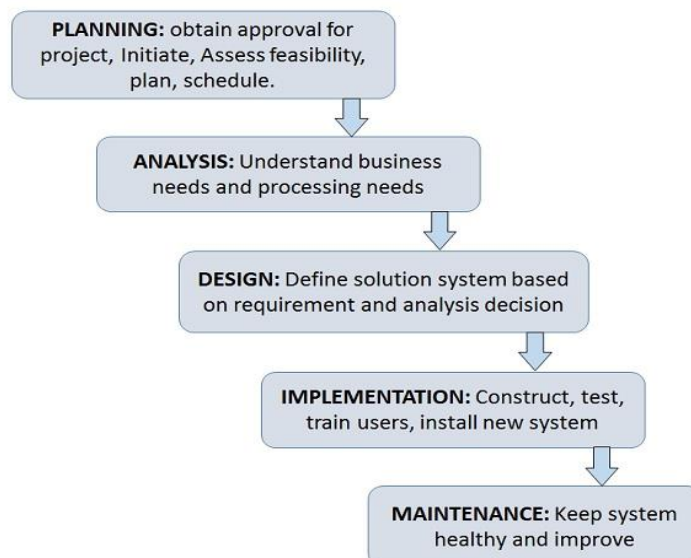
Fig 3.1 System Development Life Cycle

3.1. Planning and requirement analysis

After doing this project and analysing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible given unlimited resources and infinite time. Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future requirements.

- We gathered, analysed, and validated the information
- Requirements and prototypes for the new system were defined.
- Evaluating the alternatives and prioritizing the requirements.
- Examine the information needs of end-user and enhances the system goal

3.2 Defining Requirements

Once the requirement analysis is done, the next stage is to certainly represent and document the software requirements and get them accepted from the project stakeholders.

This is accomplished through "SRS"- Software Requirement Specification document which contains all the product requirements to be constructed and developed during the project life cycle.

3.3 Designing the Software

- Includes the design of home page course ,log in page Code structure etc.
- Transform the analysis and specifications into logical structure, which contains detailed and complete set of specifications that can be implemented in a programming language.
- Create a contingency, training, maintenance, and operation plan
- Review the proposed design. Ensure that the final design must meet the requirements.

3.4 implementation:
- Implement the design into source code through coding.
- Combine all the modules together into coherent website without error and defect
- Using HTML structure of website is made and using CSS design of page and JavaScript .

Chapter 4

4. INTRODUCTION TO CONCEPT AND TOOLS:

4.1 INTRODUCTION OF HTML

HTML stands for **Hyper Text Markup Language**, which is the most widely used language on Web to develop web pages. **HTML** was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML, and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version, which is an extension to HTML 4.01, and this version was published in 2012.[5]

4.1.1   NEED OF HTML?

Originally, **HTML** was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

HTML is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning HTML:

- **Create Web site** - You can create a website or customize an existing web template if you know HTML well.
- **Become a web designer** - If you want to start a carrer as a professional web designer, HTML and CSS designing is a must skill.
- **Understand web** - If you want to optimize your website, to boost its speed and performance, it is good to know HTML to yield best results.
- **Learn other languages** - Once you understands the basic of HTML then other related technologies like JavaScript, php, or angular are become easier to understand.

4.1.2 Applications of HTML

As mentioned before, HTML is one of the most widely used language over the web. I'm going to list few of them here:

- **Web pages development** - HTML is used to create pages which are rendered over the web. Almost every page of web is having html tags in it to render its details in browser.
- **Internet Navigation** - HTML provides tags which are used to navigate from one page to another and is heavily used in internet navigation.
- **Responsive UI** - HTML pages now-a-days works well on all platform, mobile, tabs, desktop or laptops owing to responsive design strategy.
-
- **Offline support** HTML pages once loaded can be made available offline on the machine without any need of internet.
- **Game development**- HTML5 has native support for rich experience and is now useful in gaming development arena as well.

Prerequisites

Before proceeding you should have a basic working knowledge with Windows or Linux operating system, additionally you must be familiar with −

- Experience with any text editor like notepad, notepad++, or edit plus etc.
- How to create directories and files on your computer.
- How to navigate through different directories.
- How to type content in a file and save them on a computer.
- Understanding about images in different formats like JPEG, PNG format.

```
<!DOCTYPE html>
<html>

   <head>
      <title>This is document title</title>
   </head>   <body>
      <h1>This is a heading</h1>
```

This is a heading

Document content goes here....

4.1.3 HTML Tags

As told earlier, HTML is a markup language and makes use of various tags to format the content. These tags are enclosed within angle braces **<Tag Name>**. Except few tags, most of the tags have their corresponding closing tags. For example, **<html>** has its closing tag **</html>** and **<body>** tag has its closing tag **</body>** tag etc.

Table4.1: HTML tags

Sr.No	Tag & Description
1	**<!DOCTYPE...>** This tag defines the document type and HTML version.
2	**<html>** This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags.
3	**<head>** This tag represents the document's header which can keep other HTML tags like <title>, <link> etc.
4	**<title>** The <title> tag is used inside the <head> tag to mention the document title.
5	**<body>** This tag represents the document's body which keeps other HTML tags like <h1>, <div>, <p> etc.
6	**<h1>** This tag represents the heading.
7	**<p>** This tag represents a paragraph.

To learn HTML, you will need to study various tags and understand how they behave, while formatting a textual document. Learning HTML is simple as users have to learn the usage of different tags in order to format the text or images to make a beautiful webpage.

World Wide Web Consortium (W3C) recommends using lowercase tags starting from HTML 4.

HTML Document Structure

A typical HTML document will have the following structure −

```
<html>
  <head>
    Document header related tags
  </head>
  <body>
    Document body related tags
  </body>
 </html>
```

4.1.3.1 Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements **<h1>, <h2>, <h3>, <h4>, <h5>,** and **<h6>**. While displaying any heading, browser adds one line before and one line after that heading.

E x a m p l e :

```
<!DOCTYPE html>
<html>
<head>
    <title>Heading Example</title>
  </head>
        <body>
    <h1>This is heading 1</h1>
    <h2>This is heading 2</h2>
    <h3>This is heading 3</h3>
    <h4>This is heading 4</h4>
    <h5>This is heading 5</h5>
    <h6>This is heading 6</h6>
  </body>
```

```
</html>
```

This will produce the following result –

Paragraph Tag

The **<p>** tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening <p> and a closing </p> tag as shown below in the example −

Example:

```
<!DOCTYPE html>
<html>

  <head>
    <title>Paragraph Example</title>
  </head>
<body>
    <p>Here is a first paragraph of text.</p>
    <p>Here is a second paragraph of text.</p>
    <p>Here is a third paragraph of text.</p>
  </body>

</html>
```

This will produce the following result –

Here is a first paragraph of text

Here is a second paragraph of text.

Here is a third paragraph of text.

Line Break Tag

Whenever you use the **
** element, anything following it starts from the next line. This tag is an example of an **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The <br /> tag has a space between the characters **br** and the forward slash. If you omit this space, older browsers will have trouble rendering the line break, while if you miss the forward slash character and just use <br> it is not valid in XHTML

Suppose you want to use the phrase "12 Angry Men." Here, you would not want a browser to split the "12, Angry" and "Men" across two lines −

An example of this technique appears in the movie "12 Angry Men."

In cases, where you do not want the client browser to break text, you should use a nonbreaking space entity ** ** instead of a normal space. For example, when coding the "12 Angry Men" in a paragraph, you should use something similar to the following code –

4.1.3.2 HTML Table Tags

Table4.2: HTML Table Tags

Tag	Description
<table>	It defines a table.
<tr>	It defines a row in a table.
<th>	It defines a header cell in a table.
<td>	It defines a cell in a table.
<caption>	It defines the table caption.
<colgroup>	It specifies a group of one or more columns in a table for formatting.
<col>	It is used with <colgroup> element to specify column properties for each column.
<tbody>	It is used to group the body content in a table.

<thead>	It is used to group the header content in a table.
<tfooter>	It is used to group the footer content in a table.

4.1.3.3 HTML Phrase tag

The HTML phrase tags are special purpose tags, which defines the structural meaning of a block of text or semantics of text. Following is the list of phrase tags, some of which we have already discussed in HTML formatting.

- o Abbreviation tag : <abbr>
- o Acronym tag: <acronym> (not supported in HTML5)
- o Marked tag: <mark>
- o Strong tag: <strong>
- o Emphasized tag : <em>
- o Definition tag: <dfn>
- o Quoting tag: <blockquote>
- o Short quote tag : <q>
- o Code tag: <code>
- o Address tag: <address>

4.1.4 HTML - Elements

An **HTML element** is defined by a starting tag. If the element contains other content, it ends with a closing tag, where the element name is preceded by a forward slash as shown below with few tags –

Table4.3:HTML elements

Start Tag	Content	End Tag
<p>	This is paragraph content.	</p>
<h1>	This is heading content.	</h1>

<div>	This is division content.	</div>

So here <p>.... </p> is an HTML element, <h1>...</h1> is another HTML element. There are some HTML elements which don't need to be closed, such as <img../>, <hr /> and <br /> elements. These are known as void elements.

HTML documents consists of a tree of these elements, and they specify how HTML documents should be built, and what kind of content should be placed in what part of an HTML document.

4.1.5 HTML - Attributes

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts − a **name** and a **value**

- The **name** is the property you want to set. For example, the paragraph **<p>** element in the example carries an attribute whose name is **align**, which you can use to indicate the alignment of paragraph on the page.
- The **value** is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: **left, centre** and **right**.

4.1.5.1 Core Attributes

The four core attributes that can be used on the majority of HTML elements (although not all) are −

- Id
- Title
- Class
- Style

.4.1.5.2 Internationalization Attributes

There are three internationalization attributes, which are available for most (although not all) XHTML elements.

- dir
- lang
- xml:lang

The dir. Attribute

The **dir.** attribute allows you to indicate to the browser about the direction in which the text should flow. The dir. attribute can take one of two values, as you can see in the table that follows –

Table4.4:attributes

Value	Meaning	
ltr	Left to right (the default value)	
rtl	Right to left (for languages such as Hebrew or Arabic that are read right to left)	
title	User Defined	"Pop-up" title of the elements.

4.1.6 HTML - Formatting

Bold Text

Anything that appears within ... element, is displayed in bold .

Italic Text

Anything that appears within <i>...</i> element is displayed in italicized as shown below −

Underlined Text

Anything that appears within <u>...</u> element, is displayed with underline

Strike Text

Anything that appears within <strike>...</strike> element is displayed with strikethrough, which is a thin line through the text

Superscript Text

The content of a ^{...} element is written in superscript; the font size used is the same size as the characters surrounding it but is displayed half a character's height above the other characters.

Subscript Text

The content of a _{...} element is written in subscript; the font size used is the same as the characters surrounding it but is displayed half a character's height beneath the other characters.

Inserted Text

Anything that appears within <ins>...</ins> element is displayed as inserted text.

Deleted Text

Anything that appears within ~~...~~ element, is displayed as deleted text.

Larger Text

The content of the <big>...</big> element is displayed one font size larger than the rest of the text surrounding i

Smaller Text

The content of the <small>...</small> element is displayed one font size smaller than the rest of the text surrounding

4.1.8 HTML Image

HTML image tag **is used to display image on the web page. HTML img tag is an empty tag that contains attributes only, closing tags are not used in HTML image element.**

Let's see an example of HTML image.

1.

1) src

It is a necessary attribute that describes the source or path of the image. It instructs the browser where to look for the image on the server.

The location of image may be on the same directory or another server.

2) alt

The alt attribute defines an alternate text for the image, if it can't be displayed. The value of the alt attribute describes the image in words. The alt attribute is considered good for SEO prospective.

3) width

It is an optional attribute which is used to specify the width to display the image. It is not recommended now. You should apply CSS in place of width attribute.

4) height

It h3 the height of the image. The HTML height attribute also supports iframe, image and object elements. It is not recommended now. You should apply CSS in place of height attribute.

HTML Lists

HTML Lists are used to specify lists of information. All lists may contain one or more list elements. There are three different types of HTML lists:

1. Ordered List or Numbered List (ol)
2. Unordered List or Bulleted List (ul)
3. Description List or Definition List (dl)

<button>	It defines a clickable button.

4.1.9 HTML I frame

In this article, we will know **HTML I frames**, their implementation through the examples. The frame in HTML stands for **Inline Frame**. The " I frame " tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders. An inline frame is used to embed another document within the current HTML document. The HTML iframe name attribute is used to specify a reference for an <Iframe> element. The name attribute is also used as a reference to the elements in JavaScript. The iframe is basically used to show a webpage inside the current web page. The ' **src** ' attribute is used to specify the URL of the document that occupies the iframe.

Syntax:
<iframe src="URL" title="description"></iframe>

Accepted Attribute: The following attributes can be used with the <iframe> tag in HTML.
- HTML <iframe> allow Attribute
- HTML <iframe> allowfullscreen attribute

- HTML <iframe> allowpaymentrequest attribute
- <u>HTML <iframe> height attribute</u>
- <u>HTML <iframe> width attribute</u>
- HTML <iframe> loading attribute
- <u>HTML <iframe> scrolling attribute</u>
- <u>HTML <iframe> name attribute</u>
- <u>HTML <iframe> referrerpolicy attribute</u>
- <u>HTML <iframe> sandbox attribute</u>
- <u>HTML <iframe> src attribute</u>
- **<u>HTML <iframe> srcdoc attribute</u>**

<u>4.10.</u> HTML - Backgrounds

By default, your webpage background is white in colour. You may not like it, but no worries. HTML provides you following two good ways to decorate your webpage background.

- HTML Background with Colours
- HTML Background with Images

4.1.11.1 Html Background with Colors

The **bgcolor** attribute is used to control the background of an HTML element, specifically page body and table backgrounds.

Note − The *bgcolor* attribute deprecated in HTML5. Do not use this attribute.

Following is the syntax to use bgcolor attribute with any HTML tag.

<tagname bgcolor = "color_value"...>

4.1.11.2 Html Background with Images

The **background** attribute can also be used to control the background of an HTML element, specifically page body and table backgrounds. You can specify an image to set background of your HTML page or table.

Note − The *background* attribute deprecated in HTML5. Do not use this attribute.

Following is the syntax to use background attribute with any HTML tag.

Note − The *background* attribute is deprecated and it is recommended to use Style Sheet for background setting.

4.1.12 HTML Color Coding Methods

There are following three different methods to set colours in your web page −

- **Colour names** − You can specify colour names directly like green, blue or red.
- **Hex codes** − A six-digit code representing the amount of red, green, and blue that makes up the colour.

External JavaScript

If you are going to define a functionality which will be used in various HTML documents then it's better to keep that functionality in a separate JavaScript file and then include that file in your HTML documents. A JavaScript file will have extension as **.js** and it will be included in HTML files using <script> tag.

Internal Script

You can write your script code directly into your HTML document. Usually we keep script code in header of the document using <script> tag, otherwise there is no restriction and you can put your source code anywhere in the document but inside <script> tag.

Event Handlers

Event handlers are nothing but simply defined functions which can be called against any mouse or keyboard event. You can define your business logic inside your event handler which can vary from a single to 1000s of line code.

Following example explains how to write an event handler. Let's write one simple function *EventHandler()* in the header of the document. We will call this function when any user brings mouse over a paragraph.

4.2.1 INTRODUCTION OF CSS:

CSS tutorial or CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

- o CSS stands for Cascading Style Sheet.
- o CSS is used to design HTML tags.
- o CSS is a widely used language on the web.
- o HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

CSS Syntax

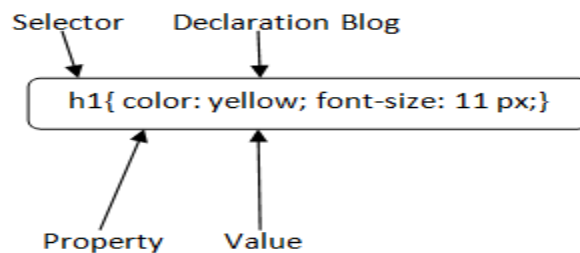A CSS rule set contains a selector and a declaration block.

Fig4.1. CSS syntax

Selector: Selector indicates the HTML element you want to style. It could be any tag like <h1>, <title> etc.

Declaration Block: The declaration block can contain one or more declarations separated by a semicolon. For the above example, there are two declarations:

1. color: yellow.
2. font-size: 11px;

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

Property: A Property is a type of attribute of HTML element. It could be color, border etc.

Value: Values are assigned to CSS properties. In the above example, value "yellow" is assigned to colour property.

CSS Selector

CSS selectors are used *to select the content you want to style*. Selectors are the part of CSS rule set. CSS selectors select HTML elements according to its id, class, type, attribute etc.

There are several different types of selectors in CSS.

1. CSS Element Selector
2. CSS Id Selector
3. CSS Class Selector
4. CSS Universal Selector

5. CSS Group Selector

4.2.2 How to add CSS

CSS is added to HTML pages to format the document according to information in the style sheet. There are three ways to insert CSS in HTML documents.

1. Inline CSS
2. Internal CSS
3. External CSS

---

1) Inline CSS

Inline CSS is used to apply CSS on a single line or element.

2) Internal CSS

Internal CSS is used to apply CSS on a single document or page. It can affect all the elements of the page. It is written inside the style tag within head section of html.

For example:

<style>

1. p{color: blue}
2. **</style>**

For more visit here: Internal CSS

3) External CSS

External CSS is used to apply CSS on multiple pages or all pages. Here, we write all the CSS code in a css file. Its extension must be .css for example style.css.For example:

p{color: blue}

You need to link this style.css file to your html pages like this:

<link rel="stylesheet" type="text/css" href="style.css"**>**

The link tag must be used inside head section of html.

4.2.3 CSS Background

CSS background property is used to define the background effects on element. There are 5 CSS background properties that affects the HTML elements:

1. background-color
2. background-image
3. background-repeat
4. background-attachment
5. background-position

4.2.3.1 Background-color

This property is used to set the background color of an element. The background of an element covers the total size, including the padding and border and excluding margin. It can be applied to all HTML elements.

Syntax

1. element {
2.   background-color: color_name|transparent|initial|inherit;
3. }

Let's discuss the possible values of this property.

- o **color name:** It is used for defining the background color value or the color codes. It can be given by using the color name, hexadecimal value, or rgb() value.
- o **transparent:** It is the default value of this property, which is used to specify the transparent background-color.
- o **initial:** It is not used to set the background color. It sets the default value.

4.2.4 CSS Border

The CSS border is a shorthand property used to set the border on an element. Border properties are used to specify the style, color and size of the border of an element. The CSS border properties are given below

- border-style
- border-color
- border-width
- border-radius

Table4.5: list of border style:

Value	Description
none	It doesn't define any border.
dotted	It is used to define a dotted border.
dashed	It is used to define a dashed border.
solid	It is used to define a solid border.
double	It defines two borders wIth the same border-width value.
groove	It defines a 3d grooved border. effect is generated according to border-colour value.
ridge	It defines a 3d ridged border. effect is generated according to border-colour value.
inset	It defines a 3d inset border. effect is generated according to border-colour value.
outset	It defines a 3d outset border. effect is generated according to border-colour value.

4.2.5 CSS Display

CSS display is the most important property of CSS which is used to control the layout of the element. It specifies how the element is displayed.

Every element has a default display value according to its nature. Every element on the webpage is a rectangular box and the CSS property defines the behaviour of that rectangular box.

4.2.5.1CSS Display default properties
Table4.6: CSS Display default properties

default value	inline
Inherited	no
animation supporting	no
Version	css1
JavaScript syntax	object.style.display="none"

syntax

1. display: value;

4.2.5.2 CSS display values

There are following CSS display values which are commonly used.

1. display: inline.
2. display: inline-block.
3. display: block;
4. display: run-in;
5. display: none;

4.2.5 CSS Cursor

It is used to define the type of mouse cursor when the mouse pointer is on the element. It allows us to specify the cursor type, which will be displayed to the user. When a user hovers on the link, then by default, the cursor transforms into the hand from a pointer.

Let's understand the property values of the cursor.

Table 4.7 CSS cursor

Help	It is in the form of a question mark or ballon, which represents that help is available.

None	It is used to indicate that no cursor is rendered for the element.
No-drop	It is used to represent that the dragged item cannot be dropped here.
s-resize	It indicates an edge box is to be moved down. It indicates the south direction.
Row-resize	It is used to indicate that the row can be vertically resized.
Se-resize	It represents the south/east direction, which indicates that an edge box is to be moved down and right.
Wait	It represents an hourglass.
<url>	It indicates the source of the cursor image file.
w-resize	It indicates the west direction and represents that the edge of the box is to be shifted left.
Zoom-in	It is used to indicate that something can be zoomed in.
Zoom-out	It is used to indicate that something can be zoomed out.

4.2.6 CSS Buttons

In HTML, we use the button tag to create a button, but by using CSS properties, we can style the buttons. Buttons help us to create user interaction and event processing. They are one of the widely used elements of web pages.

During the form submission, to view or to get some information, we generally use buttons.

4.2.7 CSS Float

The **CSS float property** is *a positioning property*. It is used *to push an element to the left or right*, allowing other element to wrap around it. It is generally used with images and layouts.To understand its purpose and

origin, let's take a look to its print display. In the print display, image is set into the page such that text wraps around it as needed.
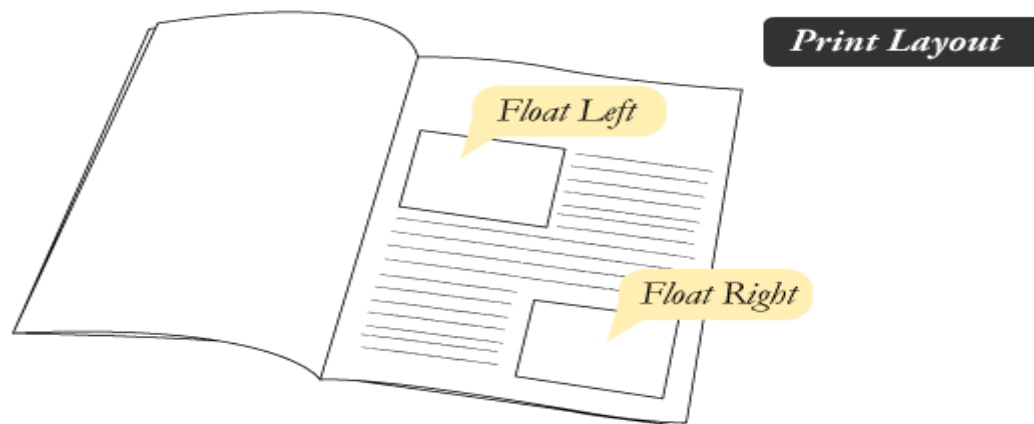

Fig4.2. Display float property

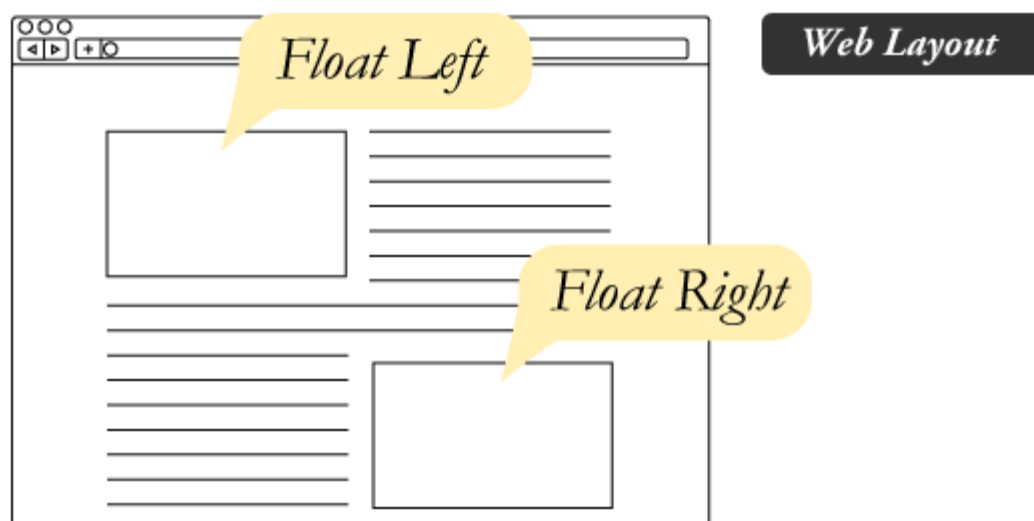

Fig4.3. Display float property

CSS Margin property is used to define the space around elements. It is completely transparent and doesn't have any background colour. It clears an area around the element.

4.2.8 CSS MARGIN

Top, bottom, left and right margin can be changed independently using separate properties. You can also change all properties at once by using shorthand margin property.

4.2.8.1 CSS Margin Properties

Table4.8 CSS margin properties

Property	Description
Margin	This property is used to set all the properties in one declaration.
margin-left	it is used to set left margin of an element.
margin-right	It is used to set right margin of an element.
margin-top	It is used to set top margin of an element.
margin-bottom	It is used to set bottom margin of an element.

4.2.8.2 CSS Margin Values

Table4.9 CSS margin value

Value	Descripton
Auto	This is used to let the browser calculate a margin.
Length	It is used to specify a margin pt, px, cm, etc. its default value is 0px.
%	It is used to define a margin in percent of the width of containing element.
Inherit	It is used to inherit margin from parent element.

4.2.9 CSS Comments

CSS comments are generally written to explain your code. It is very helpful for the users who reads your code so that they can easily understand the code.

Comments are ignored by browsers.

Comments are single or multiple lines statement and written within /*............*/ .

4.2.10 CSS Text Effects

We can apply different effects on the text used within an HTML document. Some properties can be used for adding the effects on text.

Using CSS, we can style the web documents and affects the text. The properties of the text effect help us to make the text attractive and clear. There are some text effect properties in CSS that are listed below:

- o wordbreak
- o text-overflow
- o word-wrap
- o writing-mode

4.2.11 CSS Variables

The CSS variables are used to add the values of custom property to our web page. The custom properties are sometimes referred to as cascading variables or CSS variables. The authors define these entities that contain specific values and can be reused throughout the document. These entities are set using the custom property notation and can be accessed by using the var () function.

The variables store the values and have a scope in which they can be used.CSS variables are advantageous because they allow us to reuse the same value at multiple places. The name of the variable is easy to understand and use, as compared to the color values.A variable in CSS is defined by using the two dashes (--) at the beginning, followed by the name, which is case-sensitive.

4.2.12 CSS Pseudo-elements

A pseudo-class can be defined as a keyword which is combined to a selector that defines the special state of the selected elements. Unlike the pseudo-classes, the pseudo-elements are used to style the specific part of an element, whereas the pseudo-classes are used to style the element.

As an example, a pseudo-element can be used to style the first letter or the first line of an element. The pseudo-elements can also be used to insert the content after or before an element.

Syntax

Pseudo-element has a simple syntax which is given as follows:

1. selector::pseudo-element {
2.   property: value;
3. }

Possible values

number: It means that the element's stack level is set to the given value. It allows negative values.

auto: It means that the order of the stack is equivalent to the parent, i.e., default.

4.2.13 CSS Transition

The CSS transitions are effects that are added to change the element gradually from one style to another, without using flash or

You should specify two things to create CSS

transition.

- o The CSS property on which you want to add an effect.
- o The time duration of the effect.

Let's take an example which defines transition effect on width property and duration of 3 seconds.

4.2.14 CSS Grid

A grid can be defined as an intersecting set of horizontal lines and vertical lines. CSS Grid layout divides a page into major regions. It defines the relationship between the parts of a control built from <u>HTML</u> primitives in terms of layer, position, and size. Grid property offers a grid-based layout system having rows and columns. It makes the designing of web pages easy without positioning and floating.

Similar to the table, it enables a user to align the elements into rows and columns. But compared to tables, it is easy to design layout with the <u>CSS</u> grid. We can define columns and rows on the grid by using **grid-template-rows** and **grid-template-columns** properties.

The <u>CSS</u> **grid property** is supported in browsers such as **Google Chrome, Internet Explorer, Firefox, Safari,** and **Opera**.

Grid Container

We can define the grid container by setting the **display** property to **grid** or **inline grid** on an element

4.2.15 CSS Layout

CSS layout is easy to design. We can use CSS layout to design our web page such as home page, contact us, about us etc.

There are 3 ways to design layout of a web page:

1. **HTML Div. with CSS**: fast and widely used now.
2. **HTML Table**: slow and less preferred.
3. **HTML Frameset**: deprecated now.

A CSS layout can have header, footer, left pane, right pane and body part. Let's see a simple example of <u>CSS</u> layout.

4.2.16 CSS Box Model

The components that can be depicted on the web page consist of one or more than one rectangular box.A CSS box model is a compartment that includes numerous assets, such as edge, border, padding and material. It is used to develop the design and structure of a web page. It can be used as a set of tools to personalize the layout

of different components. According to the CSS box model, the web browser supplies each element as a square prism.The following diagram illustrates how the CSS properties of width, height, padding, border and margin dictate that how much space an attribute will occupy on a web page.
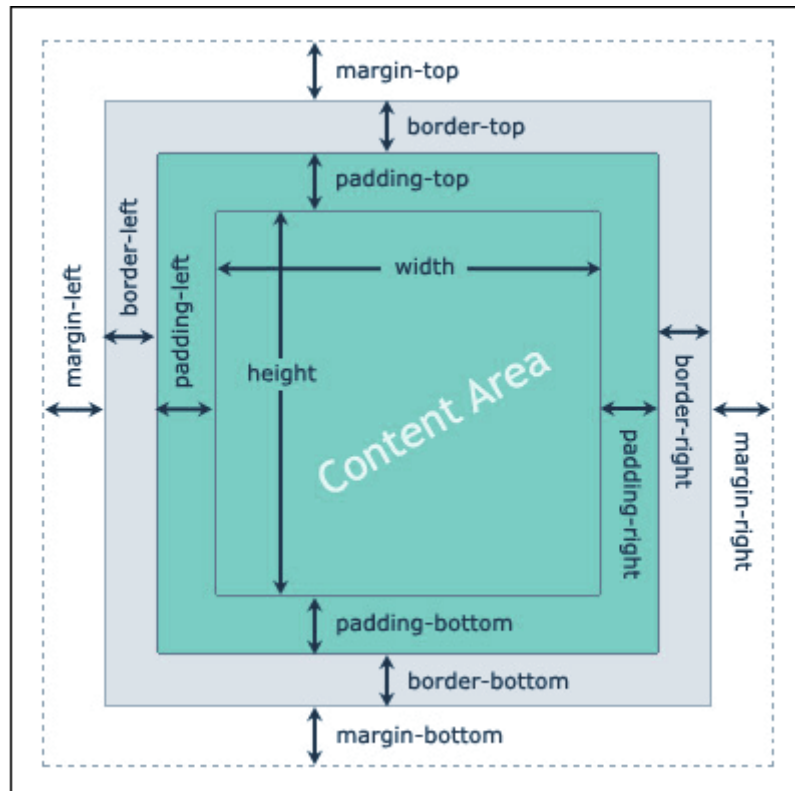

Fig4.4.box model

The CSS box model contains the different properties in CSS. These are listed below.

- **Border**
- **Margin**
- **Padding**
- **Content**

Now, we are going to determine the properties one by one in detail.[3]

Border Field

4.3. Introduction of JavaScript

JavaScript is a lightweight, interpreted **programming** language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. **JavaScript** is very easy to implement because it is integrated with HTML. It is open and cross-platform. **Why to Learn JavaScript**

JavaScript is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning JavaScript:[6]

- JavaScript is the most popular **programming language** in the world and that makes it a programmer's great choice. Once you learnt JavaScript, it helps you developing great front-end as well as back-end software's using different JavaScript based frameworks like jQuery, Node.JS etc.
- JavaScript is everywhere, it comes installed on every modern web browser and so to learn JavaScript you really do not need any special environment setup. For example Chrome, Mozilla Firefox , Safari and every browser you know as of today, supports JavaScript.
- JavaScript helps you create really beautiful and crazy fast websites. You can develop your website with a console like look and feel and give your users the best Graphical User Experience.
- JavaScript usage has now extended to mobile app development, desktop app development, and game development. This opens many opportunities for you as JavaScript Programmer.
- Due to high demand, there is tons of job growth and high pay for those who know JavaScript. You can navigate over to different job sites to see what having JavaScript skills looks like in the job market.
- Great thing about JavaScript is that you will find tons of frameworks and Libraries already developed which can be used directly in your software development to reduce your time to market.

There could be 1000s of good reasons to learn JavaScript Programming. But one thing for sure, to learn any **programming language**, not only JavaScript, you just need to code, and code and finally code until you become expert.

There are many useful **JavaScript frameworks** and libraries available:

- Angular
- React
- jQuery
- Vue.js
- Ext.js
- Ember.js
- Meteor

- Mithril
- Node.js
- Polymer
- Aurelia
- Backbone.js

It is impossible to give a complete list of all the available JavaScript frameworks and libraries. The JavaScript world is just too large and too much new is happening.

4.3.1 Applications of JavaScript Programming

As mentioned before, **JavaScript** is one of the most widely used **programming languages** (Front-end as well as Back-end). It has its presence in almost every area of software development. I'm going to list few of them here:

- **Client-side validation** - This is really important to verify any user input before submitting it to the server and JavaScript plays an important role in validating those inputs at front-end itself.
- **Manipulating HTML Pages** - JavaScript helps in manipulating HTML page on the fly. This helps in adding and deleting any HTML tag very easily using JavaScript and modify your HTML to change its look and feel based on different devices and requirements.
- **User Notifications** - You can use JavaScript to raise dynamic pop-ups on the webpages to give different types of notifications to your website visitors.
- **Back-end Data Loading** - JavaScript provides Ajax library which helps in loading back-end data while you are doing some other processing. This really gives an amazing experience to your website visitors.
- **Presentations** - JavaScript also provides the facility of creating presentations which gives website look and feel. JavaScript provides Reveals and BespokeJS libraries to build a web-based slide presentation.
- **Server Applications** - Node JS is built on Chrome's JavaScript runtime for building fast and scalable network applications. This is an event-based library which helps in developing very sophisticated server applications including Web Servers.

This list goes on, there are various areas where millions of software developers are happily using JavaScript to develop great websites and others software.

4.3.2. JavaScript - Syntax

JavaScript can be implemented using JavaScript statements that are placed within the **<script>... </script>** HTML tags in a web page.

You can place the **<script>** tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the **<head>** tags.The <script> tag alerts the browser program to start interpreting all the text between these tags as a script. A simple syntax of your JavaScript will appear as follows.

4.3.3. JavaScript - Variables

One of the most fundamental characteristics of a programming language is the set of data types it supports. These are the type of values that can be represented and manipulated in a programming language.

JavaScript allows you to work with three primitive data types −

- **Numbers,** e.g., 123, 120.50 etc.
- **Strings** of text e.g. "This text string" etc.
- **Boolean** e.g., true or false.

4.3.4 JavaScript Variables

Like many other programming languages, JavaScript has variables. Variables can be thought of as named containers. You can place data into these containers and then refer to the data simply by naming the container.

Before you use a variable in a JavaScript program, you must declare it. Variables are declared with the **var** keyword as follows.

Storing a value in a variable is called **variable initialization**. You can do variable initialization at the time of variable creation or at a later point in time when you need that variable.

For instance, you might create a variable named **money** and assign the value 2000.50 to it later. For another variable, you can assign a value at the time of initialization as follows.

4.3.4 JavaScript - Operators

Let us take a simple expression **4 + 5 is equal to 9**. Here 4 and 5 are called **operands** and '+' is called the **operator**. JavaScript supports the following types of operators.

- Arithmetic Operators
- Comparison Operators
- Logical (or Relational) Operators
- Assignment Operators
- Conditional (or ternary) Operators

4.3.5. JavaScript - if...else Statement

While writing a program, there may be a situation when you need to adopt one out of a given set of paths. In such cases, you need to use conditional statements that allow your program to make correct decisions and perform right actions.

JavaScript supports conditional statements which are used to perform different actions based on different conditions. Here we will explain the **if. Else** statement.

Flow Chart of if-else

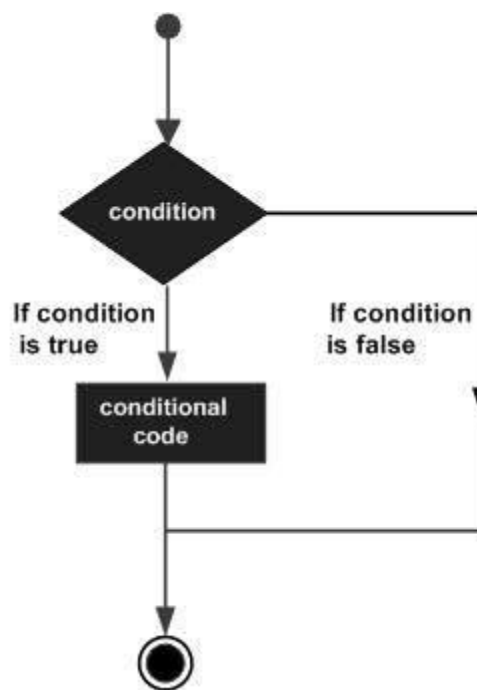The following flow chart shows how the if-else statement works.

Fig4.5.flow chart of else if statement

JavaScript supports the following forms of **if. Else** statement −

- if statement
- if...else statement
- if...else if... statement

4.3.6. JavaScript - Switch Case

You can use multiple **if...else…if** statements, as in the previous chapter, to perform a multiway branch. However, this is not always the best solution, especially when all of the branches depend on the value of a single variable.

Starting with JavaScript 1.2, you can use a **switch** statement which handles exactly this situation, and it does so more efficiently than repeated **if...else if** statements.

Flow Chart

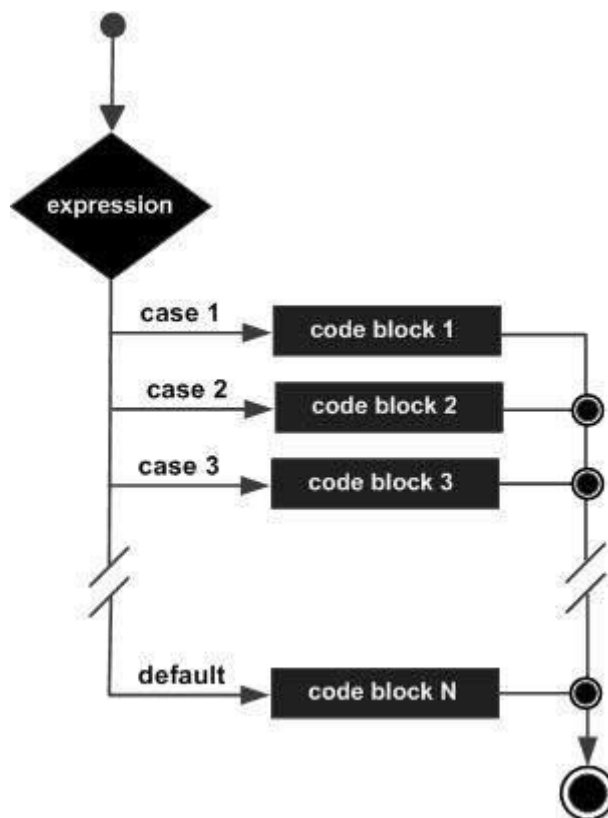The following flow chart explains a switch-case statement works.

Fig4.6. Flowchart of switch case working

Syntax

The objective of a **switch** statement is to give an expression to evaluate and several different statements to execute based on the value of the expression. The interpreter checks each **case** against the value of the expression until a match is found. If nothing matches, a **default** condition will be used.

switch (expression) {
case condition 1: statement(s)
break;

case condition 2: statement(s)
break;
...

case condition n: statement(s)

break;

default: statement(s)

}

4.3.7 JavaScript - Loop Control

JavaScript provides full control to handle loops and switch statements. There may be a situation when you need to come out of a loop without reaching its bottom. There may also be a situation when you want to skip a part of your code block and start the next iteration of the loop.

To handle all such situations, JavaScript provides **break** and **continue** statements. These statements are used to immediately come out of any loop or to start the next iteration of any loop respectively.

The break Statement

The **break** statement, which was briefly introduced with the *switch* statement, is used to exit a loop early, breaking out of the enclosing curly braces.

Flow Chart

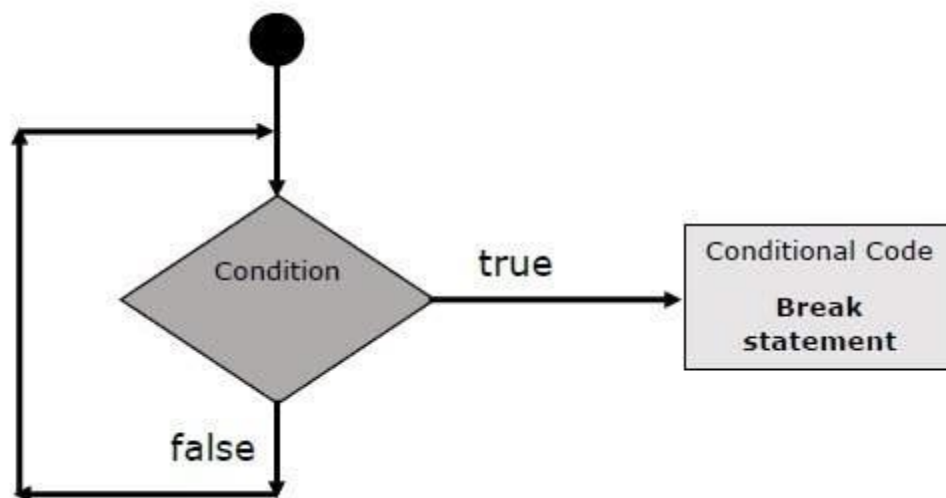The flow chart of a break statement would look as follows −

Fig4.7.flowchart of break statement

4.3.8 JavaScript - Functions

A function is a group of reusable code which can be called anywhere in your program. This eliminates the need of writing the same code again and again. It helps programmers in writing modular codes. Functions

allow a programmer to divide a big program into a number of small and manageable functions. Like any other advanced programming language, JavaScript also supports all the features necessary to write modular code using functions. You must have seen functions like **alert ()** and **write()** in the earlier chapters. We were using these functions again and again, but they had been written in core JavaScript only once. JavaScript allows us to write our own functions as well. This section explains how to write your own functions in JavaScript.

Function Definition

Before we use a function, we need to define it. The most common way to define a function in JavaScript is by using the **function** keyword, followed by a unique function name, a list of parameters (that might be empty), and a statement block surrounded by curly braces.

Syntax

The basic syntax is shown here.

```
<script type = "text/JavaScript">
<!--
function function name(parameter-list) {
statements
}
//-->
</script>
```

4.3.9 JavaScript – Events

What is an Event?

JavaScript's interaction with HTML is handled through events that occur when the user or the browser manipulates a page.When the page loads, it is called an event. When the user clicks a button, that click too is an event. Other examples include events like pressing any key, closing a window, resizing a window, etc. Developers can use these events to execute JavaScript coded responses, which cause buttons to close windows, messages to be displayed to users, data to be validated, and virtually any other type of response imaginable. Events are a part of the Document Object Model (DOM) Level 3 and every HTML element contains a set of events which can trigger JavaScript Code

4.3.10 JavaScript - Page Printing

Many times, you would like to place a button on your webpage to print the content of that web page via an actual printer. JavaScript helps you to implement this functionality using the print function of window object.

The JavaScript print function window. Print () prints the current web page when executed

JavaScript - The Number Object

The **Number** object represents numerical date, either integers or floating-point numbers. In general, you do not need to worry about **Number** objects because the browser automatically converts number literals to instances of the number class.

Syntax

The syntax for creating a **number** object is as follows −

var Val = new Number(number);

In the place of number, if you provide any non-number argument, then the argument cannot be converted into a number, it returns **NaN** (Not-a-Number).

Number Properties

Table4.10 Number Properties

Sr.No.	Property & Description
1	MAX_VALUE The largest possible value a number in JavaScript can have 1.7976931348623157E+308
2	MIN_VALUE The smallest possible value a number in JavaScript can have 5E-324
3	NaN Equal to a value that is not a number.
4	NEGATIVE_INFINITY A value that is less than MIN_VALUE.
5	POSITIVE_INFINITY

	A value that is greater than MAX_VALUE
6	prototype A static property of the Number object. Use the prototype property to assign new properties and methods to the Number object in the current document
7	constructor Returns the function that created this object's instance. By default this is the Number object.

4.3.11 JavaScript - The Strings Object

The **String** object lets you work with a series of characters; it wraps JavaScript's string primitive data type with a number of helper methods.

As JavaScript automatically converts between string primitives and String objects, you can call any of the helper methods of the String object on a string primitive.

Syntax

Use the following syntax to create a String object −

var val = new String(string);

The **String** parameter is a series of characters that has been properly encoded.

String Properties

Tabe4.11 String Properties

Sr.No.	Property & Description
1	constructor Returns a reference to the String function that created the object.
2	length Returns the length of the string.

3	prototype
	The prototype property allows you to add properties and methods to an object.

3.3.12 JavaScript - The Arrays Object

The Array object lets you store multiple values in a single variable. It stores a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

Syntax

Use the following syntax to create an Array object −

var fruits = new Array("apple", "orange", "mango");

The Array parameter is a list of strings or integers. When you specify a single numeric parameter with the Array constructor, you specify the initial length of the array. The maximum length allowed for an array is 4,294,967,295.

You can create array by simply assigning values as follows −

var fruits = ["apple", "orange", "mango"];

You will use ordinal numbers to access and to set values inside an array as follows.

fruits[0] is the first element
fruits[1] is the second element
fruits[2] is the third element

4.3.13 JavaScript - Errors & Exceptions Handling

There are three types of errors in programming: (a) Syntax Errors, (b) Runtime Errors, and (c) Logical Errors.

Syntax Errors

Syntax errors, also called **parsing errors,** occur at compile time in traditional programming languages and at interpret time in JavaScript.

For example, the following line causes a syntax error because it is missing a closing parenthesis.

```
<script type = "text/javascript">
<!--
```

window. Print(;

//-->

</script>

When a syntax error occurs in JavaScript, only the code contained within the same thread as the syntax error is affected and the rest of the code in other threads gets executed assuming nothing in them depends on the code containing the error.

Runtime Errors

Runtime errors, also called **exceptions,** occur during execution (after compilation/interpretation).

For example, the following line causes a runtime error because here the syntax is correct, but at runtime, it is trying to call a method that does not exist.

```
<script type = "text/JavaScript">
<!--
window.printme();
//-->
</script>
```

Exceptions also affect the thread in which they occur, allowing other JavaScript threads to continue normal execution.

Logical Errors

Logic errors can be the most difficult type of errors to track down. These errors are not the result of a syntax or runtime error. Instead, they occur when you make a mistake in the logic that drives your script and you do not get the result you expected.

You cannot catch those errors, because it depends on your business requirement what type of logic you want to put in your program.

The try...catch...finally Statement

The latest versions of JavaScript added exception handling capabilities. JavaScript implements the **try...catch...finally** construct as well as the **throw** operator to handle exceptions.

You can **catch** programmer-generated and **runtime** exceptions, but you cannot **catch** JavaScript syntax errors.

4.3.14 JavaScript - Animation

You can use JavaScript to create a complex animation having, but not limited to, the following elements −

- Fireworks
- Fade Effect
- Roll-in or Roll-out
- Page-in or Page-out
- Object movements

You might be interested in existing JavaScript based animation library: Script.Aculo .us. JavaScript can be used to move a number of DOM elements (<img />, <div> or any other HTML element) around the page according to some sort of pattern determined by a logical equation or function.

JavaScript provides the following two functions to be frequently used in animation programs.

- **setTimeout(function, duration)** − This function calls **function** after **duration** milliseconds from now.
- **setInterval(function, duration)** − This function calls **function** after every **duration** milliseconds.
- **clearTimeout(setTimeout_variable)** − This function calls clears any timer set by the setTimeout() functions.

JavaScript can also set a number of attributes of a DOM object including its position on the screen. You can set *top* and left attribute of an object to position it anywhere on the screen

4.3.15 JavaScript – Multimedia

he JavaScript navigator object includes a child object called plugins. This object is an array, with one entry for each plug-in installed on the browser. The navigator. Plugins object is supported only by Netscape, Firefox, and Mozilla only.

Checking for Plug-Ins

Each plug-in has an entry in the array. Each entry has the following properties −

- **name** − is the name of the plug-in.
- **filename** − is the executable file that was loaded to install the plug-in.
- **description** − is a description of the plug-in, supplied by the developer.
- **mimeTypes** − is an array with one entry for each MIME type supported by the plug-in.
-

4.3.16 JavaScript - Debugging

Every now and then, developers commit mistakes while coding. A mistake in a program or a script is referred to as a **bug**.

The process of finding and fixing bugs is called **debugging** and is a normal part of the development process. This section covers tools and techniques that can help you with debugging tasks..

4.3.17 Error Messages in Firefox or Mozilla

The most basic way to track down errors is by turning on error information in your browser. By default, Internet Explorer shows an error icon in the status bar when an error occurs on the page.

Double-clicking this icon takes you to a dialog box showing information about the specific error that occurred.

Since this icon is easy to overlook, Internet Explorer gives you the option to automatically show the Error dialog box whenever an error occurs.

To enable this option, select **Tools → Internet Options → Advanced tab.** and then finally check the **"Display a Notification About Every Script Error"** box option as shown below −
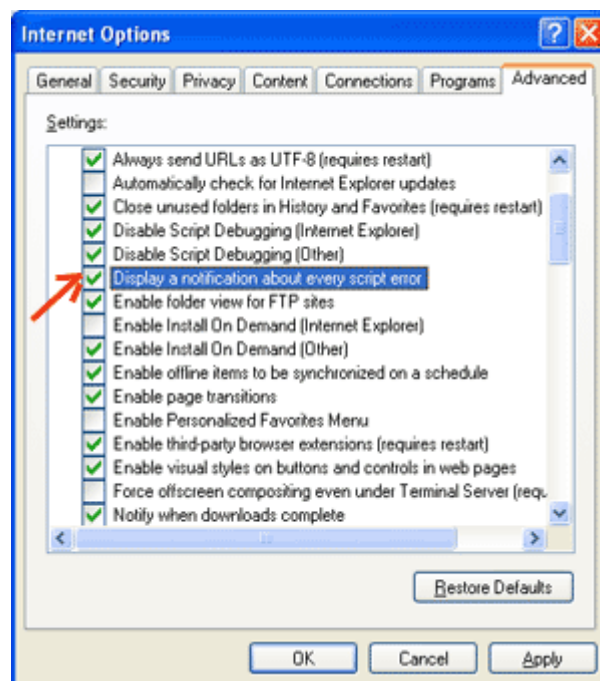

Fig4.8.error message

Error Messages in Firefox or Mozilla

Other browsers like Firefox, Netscape, and Mozilla send error messages to a special window called the **JavaScript Console** or **Error Consol**. To view the console, select **Tools → Error Consol or Web Development**.

Unfortunately, since these browsers give no visual indication when an error occurs, you must keep the Console open and watch for errors as your script executes.
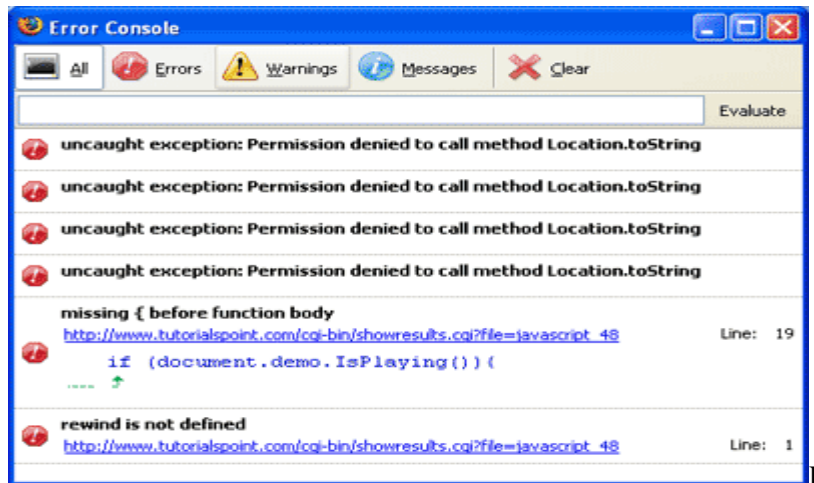

fig4.9.error message notification

Error Notifications

Error notifications that show up on Console or through Internet Explorer dialog boxes are the result of both syntax and runtime errors. These error notification include the line number at which the error occurred.

If you are using Firefox, then you can click on the error available in the error console to go to the exact line in the script having error.

4.3.18 JavaScript - Image Map

You can use JavaScript to create client-side image map. Client-side image maps are enabled by the usemap attribute for the <img /> tag and defined by special <map> and <area> extension tags.The image that is going to form the map is inserted into the page using the <img /> element as normal, except that it carries an extra attribute called use map. The value of the use map attribute is the value of the name attribute on the <map> element, which you are about to meet, preceded by a pound or hash sign. The <map> element actually creates the map for the image and usually follows directly after the <img /> element. It acts as a container for the <area /> elements that actually define the clickable hotspots. The <map> element carries only one attribute, the name attribute, which is the name that identifies the map. This is how the <img /> element knows which <map> element to use. The <area> element specifies the shape and the coordinates that define the boundaries of each clickable hotspot.

It is important to understand the differences between different browsers in order to handle each in the way it is expected. So it is important to know which browser your web page is running in.To get information about the browser your webpage is currently running in, use the built-in **navigator** object.

Navigator Properties

There are several Navigator related properties that you can use in your Web page. The following is a list of the names and descriptions of each.

Table4.12 Navigator Properties

Sr.No.	Property & Description
1	**appCodeName** This property is a string that contains the code name of the browser, Netscape for Netscape and Microsoft Internet Explorer for Internet Explorer.
2	**appVersion** This property is a string that contains the version of the browser as well as other useful information such as its language and compatibility.
3	**language** This property contains the two-letter abbreviation for the language that is used by the browser. Netscape only.
4	**mimTypes[]** This property is an array that contains all MIME types supported by the client. Netscape only.
5	**platform[]** This property is a string that contains the platform for which the browser was compiled."Win32" for 32-bit Windows operating systems

Sr.No.	
6	**plugins[]** This property is an array containing all the plug-ins that have been installed on the client. Netscape only.
7	**user Agent[]** This property is a string that contains the code name and version of the browser. This value is sent to the originating server to identify the client.

Navigator Methods

Table4.13 Navigator Methods

Sr.No.	Description
1	**javaEnabled()** This method determines if JavaScript is enabled in the client. If JavaScript is enabled, this method returns true; otherwise, it returns false.
2	**plugings.refresh** This method makes newly installed plug-ins available and populates the plugins array with all new plug-in names. Netscape only.
3	**preference(name,value)** This method allows a signed script to get and set some Netscape preferences. If the second parameter is omitted, this method will return the value of the specified preference; otherwise, it sets the value. Netscape only.
4	**taintEnabled()** This method returns true if data tainting is enabled; false otherwise.

Chapter 5

5. <u>**Results and Discussion:**</u>

- Finally, project was successful

- We were creating structure of educational webpage
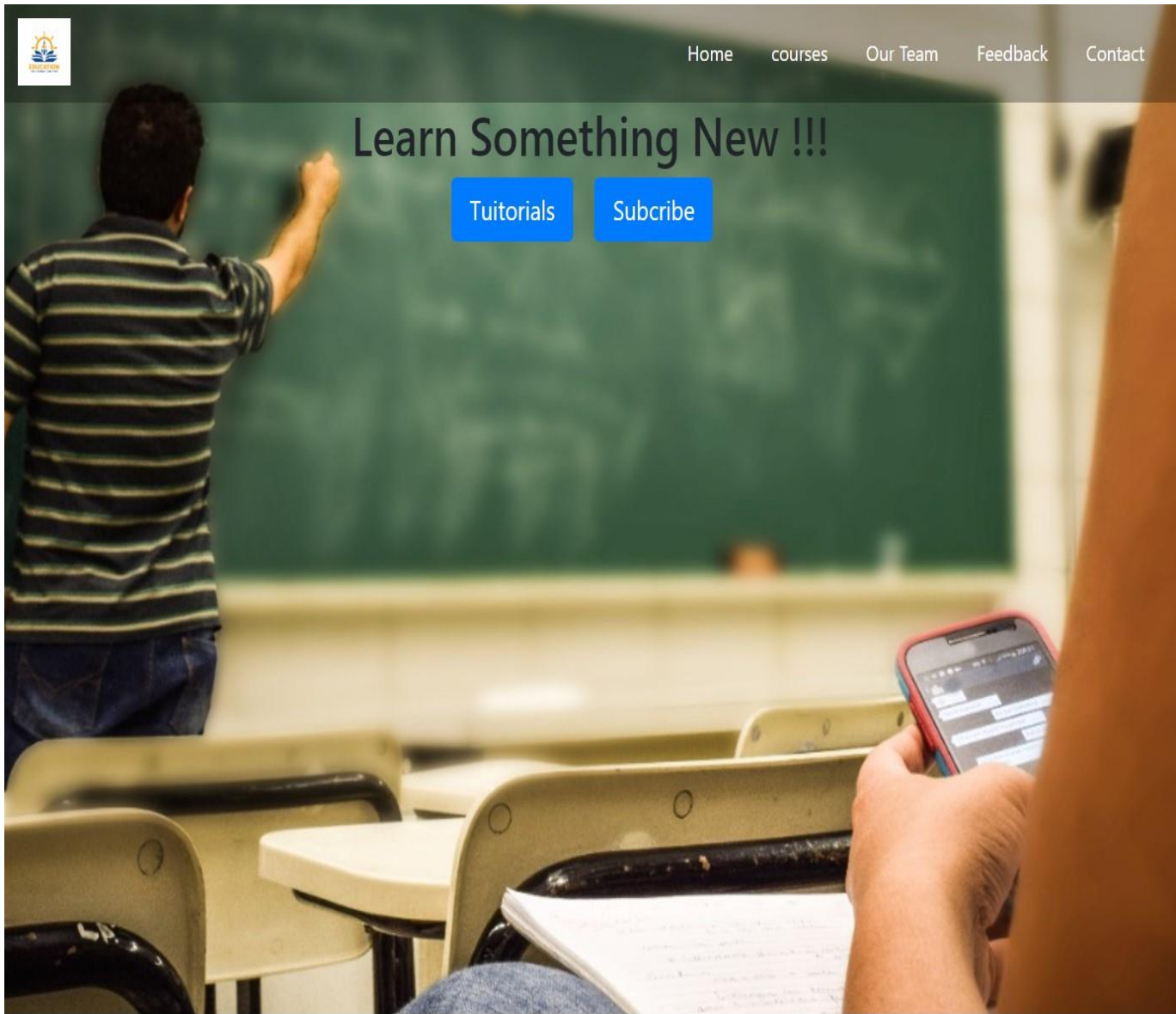
- We were able to create home page and log in page.

Fig 5.1: Home Page

Fig5.2. popular courses on website

Chapter 6

CONCLUSION AND FUTURE SCENE:

- After completing this project, we conclude that it is possible to create an website more effectively
- The entire process is very simple and user friendly with minimum coding.
- Website is easy to access and free for everyone
- For the future, speed of website can be improved or smooth
- Server side(backend) also can be added to make dynamic webpage.
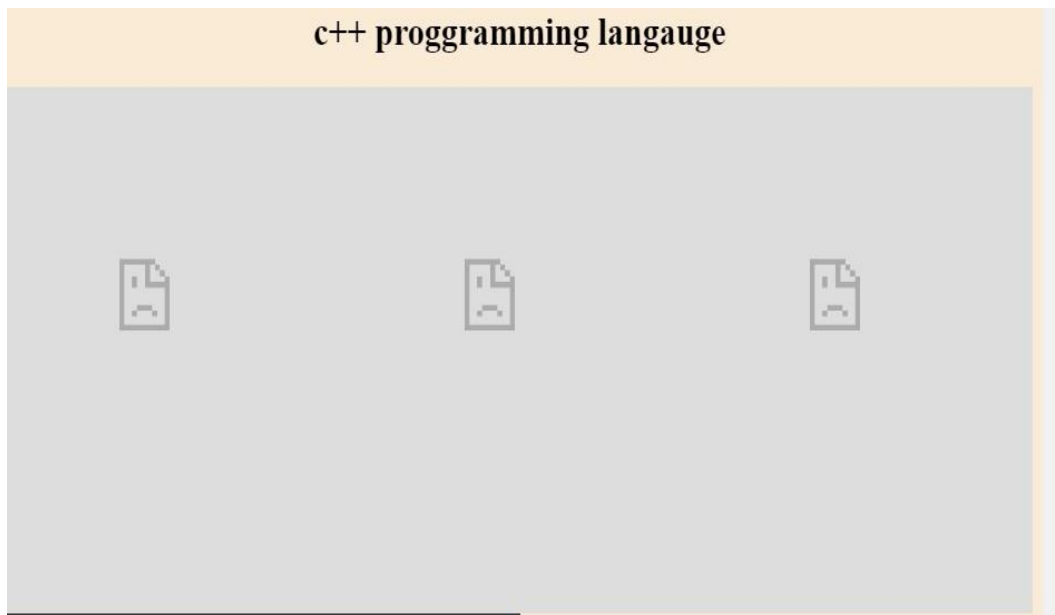- There is still problem that supposed to be solve shown below .

Fig6.1. Video not loaded

7.REFERENCE

[1] W. (n.d.-c). *HTML Tutorial*. W3school. Retrieved June 28, 2022, from https://www.w3schools.com/html/

[2] T.U.T.O.R.I.A.L.P.O.I.N.T.S. (n.d.-b). *HTML Tutorial*. Tutorial Point. Retrieved June 6, 2022, fromhttps://www.tutorialspoint.com/html/index.htm

[3] D.I.C.T.I.O.N.A.R.Y.C.O.M. (2010, May 11). *Definition of markup*. Www.Dictionary.Com. Retrieved March 17, 2022, from https://www.dictionary.com/browse/markup

[4] Kumar, K. (2022, January 1). *educational website report*. Retrieved March 17, 2022, from https://www.slideshare.net/KedarKumar3/educational-website-report

[5] J.A.V.A.T.P.O.I.N.T. (n.d.-a). *Learn HTML Tutorial - java point*. Www.Javatpoint.Com. https://www.javatpoint.com/html-tutorial

[6] GeeksforGeeks. (n.d.). *HTML*. Retrieved June 6, 2022, from https://www.geeksforgeeks.org/html/

[7] M.H.C.C. (2020, February 2). *Hardware and Software Requirements*. Mhcc.Edu. Retrieved March 17, 2022, from https://www.mhcc.edu/OLHardwareSoftwareRequirements/