**COURSE CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Unit No** | **Unit Name** | **Teaching**  **Hours** | **Distribution of Marks** |
| **1** | **Introduction to the basics of coding** | **15** | **20** |
| **2** | **Designing and Planning Web pages** | **30** | **40** |
| **3** | **Business process Automation/ERP** | **09** | **10** |
| **4** | **Introduction to Cloud and IOT Concepts** | **12** | **10** |
| **5** | **Cyber Security and Safety** | **12** | **20** |
| **6** | **Sample Experiments** |  |  |
| **Total** | | **78** | **100** |

**EVALUATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Internal Assessment (IA) Details** | | | | |
| **IA NO** | **Type of Assessment** | **Duration**  **(In Minutes)** | **Max**  **Marks** | **Conversion** |
| **IA-1** | **Written Test-1**  **(Theory Part)** | **60** | **20** | **Average of Both Test** |
| **IA-2** | **Written Test-2**  **(Theory Part)** | **60** | **20** |
| **IA-3** | **Skill Test** | **120** | **20** | **Average of 3 Skill Test** |
| **IA-4** | **Skill Test** | **120** | **20** |
| **IA-5** | **Skill Test** | **120** | **20** |
| **IA-6** | **Student Activity** | **-** | **20** | **20** |
| **Total IA Marks** | | | | **60** |

**Semester-End Evaluation (End Practical Exam)**

|  |  |  |
| --- | --- | --- |
| **SL NO** | **Topics** | **Marks Allotted** |
| 01 | Design-Written  Skill Test 1:Algorithm/Flowchart/Visual Design  Skill Test 2: Website visual design  Skill Test 3 : Work Flow or Project Plan or Cyber Security Plan or Cloud Service Concept | **10** |
| 02 | Implementation  Skill Test 1:Algorithm/Flowchart/Visual Design  Skill Test 2: Website visual design  Skill Test 3 : Work Flow or Project Plan or Cyber Security Plan or Cloud Service Concept | **20** |
| 03 | Appeal and Presentation | **10** |
| **Grand Total** | | **40** |

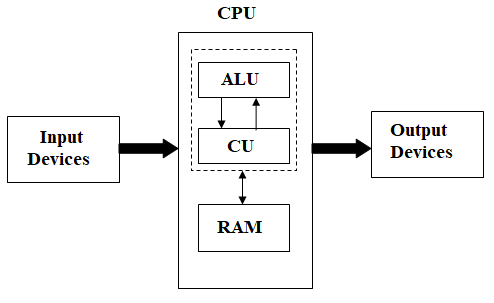
**UNIT-I**

**Introduction to the Basics of Coding**

**1.1 Introduction to Computer Programming**

**1.1.1 Definition Computer**

* A computer is an electronic device that accepts binary input data, processing it according to a program stored in its memory and produces the output or results.
* **Working of Computer**

****

**1. Input Devices/Unit**

* Computer accepts encoded information through input unit.
* Input devices are **keyboard, scanner, MICR (Magnetic Ink character reorganization), mouse** etc.

**2. Output Devices/Unit**

* Computer returns the computed results or error messages through output unit.
* Output devices are **monitor, printer, plotters** etc.

**3. Memory Unit**

* Memory unit is a storage area of computer.
* There are two types of memory

1. Primary Memory 🡪 RAM (Random Access Memory)
2. Secondary Memory 🡪 Hard Disk

**4. CPU – Central Processing Unit**

* The heart of the computer is called as CPU. The CPU has two parts

1. ALU – Arithmetic logic unit
2. CU – Control Unit

**a. ALU🡪**It performs the operations like arithmetic operations and logical operations

**b. CU 🡪**It controls all operations of the all units. It establishing co-operation among all devices connected to it.

**1.1.2 Computer Programming Languages**

**Computer Program :**

* Program means **set of instructions**.
* Computer works on set of instruction called **Computer Program** which specify the way to carry out a specify task.

**Computer Programming Language**

* Computer programming languages are used to develop the program means write set of instructions to perform the specific task
* The following are classification of programming language

1. **Machine Level Language**

* Computer System can understood the data in terms of **0 and 1 (Binary number)**

1. **Assembly Level Language**

* Programs can be written using **Symbolic Codes**.

1. **High Level Language**

* Programs can be written by using natural language like English
* High-Level Languages are C Language, C++, Java, Python, C# etc

**1.3 Algorithm**

* Algorithm is a **step-by-step procedures** for solving the problems.
* Algorithms are language independents
* A user writes algorithms in his/her own language.
* Algorithms cannot be executed on computers.

**Characteristics of Algorithm**

1. **Input**

* Algorithm may receive one or more input data before execution of algorithm

1. **Output**

* Algorithm will produce result after completion of its operation.

1. **Finiteness**

* Algorithm stops after a finite number of instructions are executed.

1. **Effective**

* The result of each steps are uniquely defined and only depends on the input.

1. **Definite**

* Each and every steps of the algorithms are precisely defined.

**Example -1**

**Write a algorithm to add the two numbers.**

Step 1: Start

Step 2: Read two numbers

Read **A** and **B**

Step 3: Add two numbers

**C=A+B**

Step 4: Display the result

Write **C**

Step 5: Stop.

**Example – 2**

**Write a algorithm to find the largest of three numbers**

Step 1: Start

Step 2: Read three numbers

**A,B,C**

Step 3: Find the largest number

**if A>B and A>C**

Write **A** is largest Number

**if B>A and B>C**

Write **B** is largest Number

**else**

Write **C** is largest Number

Step 5: Stop.

**Example – 3**

**Write a algorithm to calculate and print sum of N numbers**

Step 1: Start

Step 2: Read and Initialize

Read **N** Size

Assign **SUM**=0

Step 3: Read N numbers

Step 4: Repeat upto N Numbers

**i=1,2,……N**

**SUM=SUM+i**

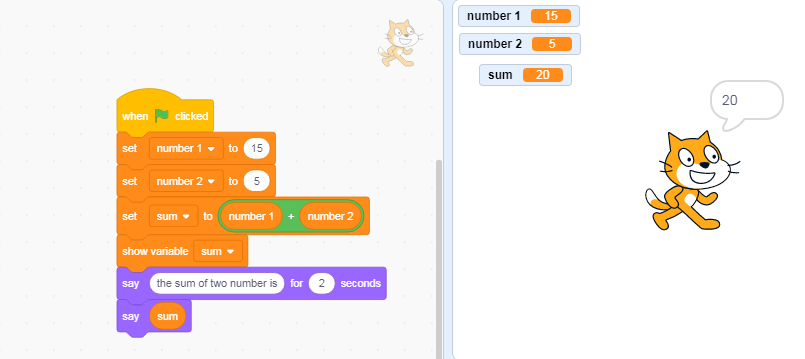
Step 5: Display the SUM

Write SUM

Step 6: Stop.

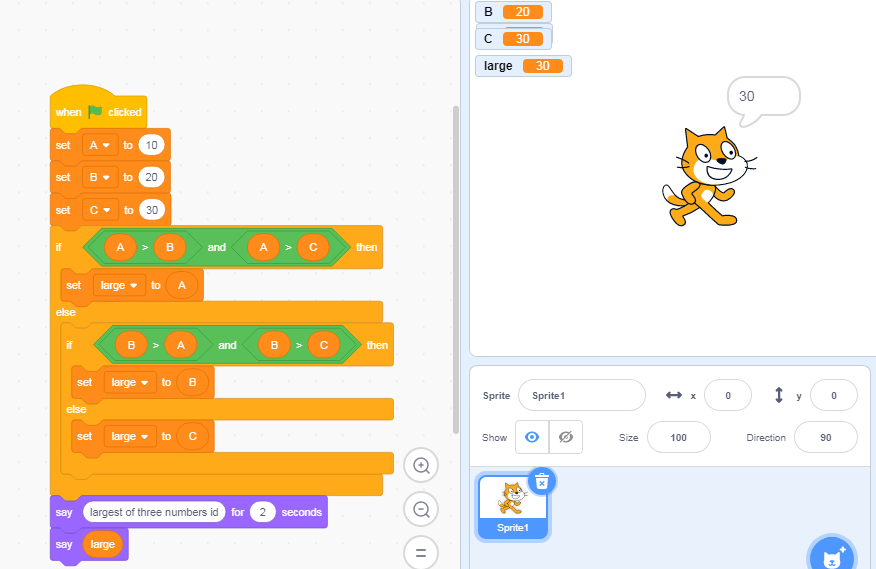
Write a code for above programmable problems to use scratch /code.org

1. **Steps to write a code for addition of two numbers**
2. Open scratch software.
3. Click on New option
4. Choose Events and the block options. For example consider this Drag and drop this block in the algorithm part.
5. Repeat the step-3 to write the script for addition of two numbers as mentioned in the figure 1
6. Save the project and RUN & EXIT from the software.



**Figure 1**

1. **Write a scratch code to find the largest of three numbers**
2. Open scratch software.
3. Click on New
4. Choose Events and the block. For example consider this Drag and drop this block in the algorithm part.
5. Repeat the step-3 to write the script for finding the largest of three numbers as mentioned in the figure 2.
6. Save the projects 🡪Run &exit from software



**Figure 2**

1. **Write a scratch code for calculate and print sum of N numbers**
2. Open scratch software.
3. Click on New
4. Choose Events and the block. For example consider this Drag and drop this block in the algorithm part.
5. Repeat step-3 to write the script to find sum of n numbers as mentioned in the figure 3.
6. Save the projects
7. Run &exit from software

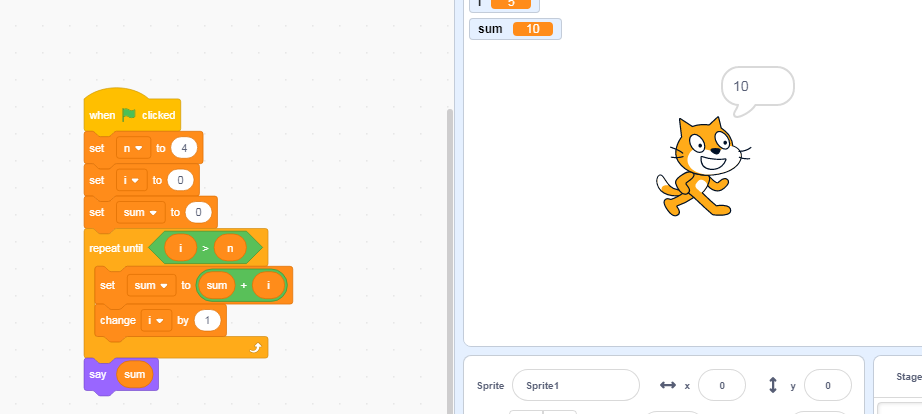


Figure 3

* 1. **Flowchart**
* Flowchart is a **graphical representation of an algorithm or program.**
* Flowchart illustrates the structure and sequence of operations of a program.
* Flowchart describes the end-to-end flow of a particular activity.
* There are two types of flowcharts

1. **Program Flowchart**

* Program flowchart is used to illustrate the structure and sequence operations of the program.
* Different type’s flowchart symbols are used to design the program flowchart.

1. **System Flowchart**

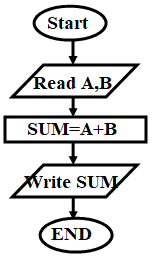
* System flowchart is used to represents a graphic model of the physical system.
  + 1. **Flowchart Notations or symbols**
* The following are the different types of the symbols used to constructs a flowcharts.

|  |  |  |
| --- | --- | --- |
| **Symbols** | **Name** | **Meaning** |
|  | Oval  (Start/End) | It is used to represent start and end of the flowchart. |
|  | Flow Lines  (Direction of flow) | It is used to indicate the flow of logic by connecting symbols. |
|  | Parallelogram  (Input and Output) | It is used for input and output operations |
|  | Rectangle  (Processing) | It is used for arithmetic operations and data manipulations |
|  | Diamond  (Decision) | It is used to represent the operations in which there are two alternatives either true or false. |
|  | Circle  (Connector) | It is used to join different flow lines |
|  | Predefined Process | It is used to represent a group of statements performing one processing tasks. |
|  | Preparation or Initialization | It is used for loop |

**1.4.2 Example for Flowchart**

**Example :1**

**Write a flowchart to addition/subtract of two numbers**



**Example :2**

**Write a flowchart to find the largest/smallest of 3 numbers**

**Read any three integer numbers (A, B, C)**

**If A>B && A>C**

**Print A**

**If B>A && B>C**

**Print B**

**Print C**

**False**

**False**

**True**

**True**

**Example: 3**

**Write a flowchart to find the sum of the given N number.**



**UNIT-II**

**Designing and Planning Web pages**

* 1. **Browser and its Types**
* The browser is application software. It is used to access the web pages in internet.
* The browser helps the users to get the information from internet.
* The following are the types of browsers

1. Internet Explorer - Developed by Microsoft
2. Google Chrome - Developed by Google
3. Safari - Developed by Apple
4. Mozilla Firebox - Developed by Mozilla Cooperation
5. Opera - Opera Cooperation
6. Microsoft Edge – Developed by Microsoft for windows 10 Operating System
7. Netscape - Developed by Sun-Microsystems for java programs
   * 1. **Web Page**

* Web page is a web document commonly written by using Hyper Text Mark-up Language (HTML). This document is accessible through the internet by using browser.
* Web pages are identified by using unique URL (Uniform Resource Locator).
* For Example - google.com is the name of the web page and it’s a URL
* There are two types web page

1. **Static Web page** - Static web pages is designed by using HTML. This page shows the same content each time when it’s loaded. These pages are client-side pages
2. **Dynamic Web page** - Dynamic web pages are designed by using scripting language like PHP, ASP or JSP. These pages contain server-side code, which allows the server to generate unique content each time when it is loaded.

* **Website 🡪** Website is a collection of related web pages

**2.1.2 WWW-**

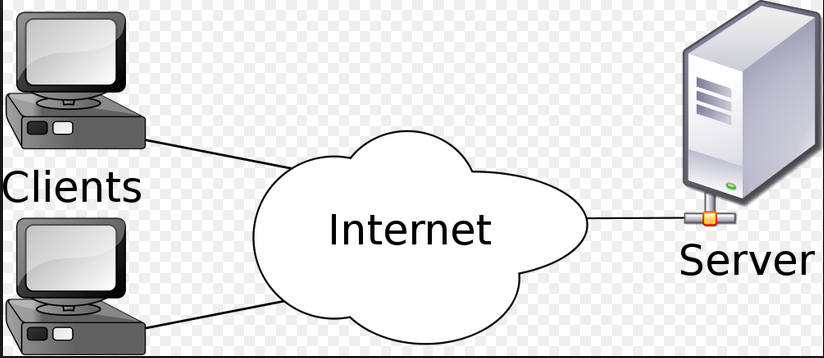
* WWW means “Word Wide Web”. It is system of internet server to identify the formatted documents by using URL.

**2.1.3 Client-Server Model (Web-Server)**

* Client – Server module consists of two systems

1. Client Systems - Means Requested System
2. Server System - Response System

* Both Systems are communicated with each other through network or internet.
* First, Client System initiates the connection to the server through request. The server processes the request and sends the response to the client.



**2.1.4 Web Server**

* Web Server is a program that processes the request sent by the client through HTTP protocol.
* The web client sends a request to web server for requesting a web page or file.

**2.2 URL (Uniform Resource Locator)**

* The general format of URL is

**Scheme:Address i.e.**

**Protocol://username@hostname/path/filename**

* The scheme specifies the communication protocol a such as **HTTP,FTP, file, telnet** etc
* The most commonly used protocol for client and server is **HTTP (Hypertext Transfer protocol).**
* The HTTP protocol works on bases of **request-response mechanism**.
* The **default port number for** **HTTP protocol is 80**
* For HTTP protocol, the address is in the following form

**http://domain\_name/path\_to\_document**

* For file, the address is in the following form

**file://path\_to\_document**

* The **host name is the server name that stores all web documents**

**2.2.1 URL Paths**

* The path to the document is similar to the path of the file or directory in the file system.
* All directory names and files are separated by the separate character (/).
* The URL path that includes all the directories along the path to the file is called as **complete path.**
* Example

**http://www.dte.kar.nic.in/file/f99/store.html**

* Sometimes, the URL path is specified in the configuration file of the server. In such case no need to specify the complete path for accessing a particular file that path is called as **partial path**.
* Example

**http://www.dte.kar.nic.in/store.html**

**2.3 Domain names and Domain Name System (DNS)**

* Domain name is used for finding and identifying computers on the Internet.
* Computers use IP addresses, which are a series of number. However, it is difficult to remember IP addresses, domain names were developed and used to identify entities on the internet rather than using IP addresses.
* There can be two or three domain names.

1. The first domain name is **host name**
2. The second domain name gives the domain of which the first domain is part.
3. The last domain name identifies the type of organization in which the host resides, which is the largest domain in the sites name

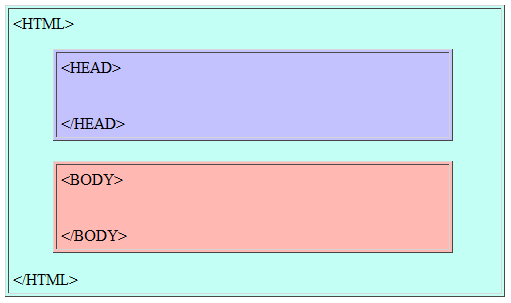
* Example

**www.dte.kar.nic.in**

* Here **dte** is the host name
* The other parts **kar** is the local domain
* The **nic** is the domain which is part of **in**
* The hostname and the entire domain names are together called a **fully qualified domain name.**

**2.4 Introduction to HTML**

* The HTML language is used to create the static web pages.
* HTML means **H**yper **T**ext **M**arkup **L**anguage.
* HTML is a markup language and it consists of **set of tags**
* All tags are built-in tag surrounded by angle brackets like **<tagname>**
* All tags must have starting tag and ending tag except <br> <b>, <hr> tag
* Ending tag must have slash before the tag name **</tagname>**
* All tags are used for designing the content
* The “**Hyper Text**” is a text which is linked to another text.
* The syntax of HTML language is

****

* **The general syntax of tag is**

**<tagname attribute\_name1=”value1” ...... attribute\_nameN=”valueN” >**

**Information**

**</tagname>**

**Note – Attributes are optional. The attributes are used for design the information.**

* It consists of two sections

1. **Heading Section**

* The heading section is define by using
* <head> tag
* The <head>tag give the general information about the html documents such as title of page using <title> tag.
* This is optional

1. **Body Section**

* The body section defined by using <body> tag
* The <body>contains the information that information may appear in browser
* It also consists of set of tags which are used for design the information.
* This tag has one attribute called **bgcolor.** It will give background color for webpage.
* **Heading Tags (h1 to h6)**
* These tags are used to give the **size of the information**.
* Here <h1> is the largest size and <h6> is the smallest size
* The syntax is

**<h1>Information </h1>**

**<h2>Information </h2>**

**<h3>Information </h3>**

**<h4>Information </h4>**

**<h5>Information </h5>**

**<h6>Information </h6>**

* **The attributes are**

1. **align =”value”.** The values are “left, right, center, justify”

* **Horizontal Tag <hr>**
* The <hr> tag is used to create the **horizontal line**.
* The syntax is <hr>
* **The attributes are**

1. **align= “value”**

* The values are “left, right, center”

1. **size=”value”**

- The value is numeric number in pixels or %. The size defines height of the line

1. **width=”value”**

* The value is numeric number in pixels or %. The width defines the width of the horizontal line
* **Break Tag <br>**
* The <br> tag is used to break the lines.
* **Formatting Tags (<b>, <i>, <u>, <big>, <small>, <sub>, <sup>, <strike>**

1. **<b> tag – Bold tag**

* This tag specifies the bold to give text

1. **<i> tag – Italic tag**

* This tag specifies the Italic style to given text

1. **<u> tag – Underline tag**

* This tag specifies the underline to the given text

1. **<big> tag – Bigger tag**

* This tag defines the bigger text then normal text

1. **<small> tag – Smaller tag**

* This tag defines the smaller text then normal text

1. **<sub> tag – Subscript tag**

* This tag defines the subscript text

1. **<sup> tag – Superscript tag**

* This tag defines the superscript text

1. **<stricke> tag – Strick tag**

* This tag defines strike through text
* **Paragraph <p> tag**
* The <p> defines the paragraph
* **Font <font> tag**
* This tag is used to specify the **font size, font face and color of the text**.
* The syntax is

**<font attribute\_name=”value”> text </font>**

* The attributes are

1. **size=”values”** 🡪 The values from 0 to 7
2. **color=”color name”** 🡪 Name of the color like red,pink,green etc
3. **face=”face name”** 🡪 The face names like“Times New Roman,Cambria, Calibri”etc

* **List Tags <ol> and <ul>**

1. **<ol> Ordered List Tag**

* This tag is used to list the items in particular ordered like numbers, italic numbers, alphabetic etc. All items must be placed between **<li>** tag.
* The **<li> (List) tag** must be present within the <ol> tag
* The syntax is

<ol attribute\_name=”value”>

<li> Text </li>

<li> Text </li>

..........

</ol>

* **The attributes are**

**1. type=”value” 🡪** The value may be **1,A,a,I, i. Any single number**

1. **<ul> Unordered List Tag**

* This tag is used to list the items using symbols like square, disc and circle. All items must be placed between **<li>** tag
* The **<li> (List) tag** must be present within the <ol> tag
* The syntax is

<ul attribute\_name=”value”>

<li> Text </li>

<li> Text </li>

..........

</ul>

* **The attributes are**

**1. type=”value” 🡪** The value may be **circle, square, disc**

* **Table <table> Tags**
* The <table> tag is used to create the table. This tag contains sub tags (th, tr, td) to define the structure of the table.

1. **<th> Tag – Table Header** 🡪 This tag defines the header or column name for the table. It is optional
2. **<tr> Tag – Table Row** 🡪 This tag defines the row for the table. This tag may have number of cells defined by the <td> tag
3. **<td> Tag – Table Data** 🡪 This tag defines the cells in row.

* **The syntax is**

**<table attribute\_name=”value” >**

**<th> Column name </th>**

**<th> Column name </th>**

**<tr>**

**<td> Data </td>**

**<td> Data </td>**

**</tr>**

**...........**

**</table>**

* **The attributes are**

1. **align=”values” 🡪** The values are **left, right** and **center**
2. **bgcolor=”colorname” 🡪** The color names like red, blue, etc
3. **width=”values” 🡪** The values in terms of numbers with pixels or %
4. **border=”values” 🡪** The values in terms of numbers with pixels or %

* The **<td> tag** also supports the attributes

1. **colspan=”number”** 🡪 Combining the number of cells. The number of cells are to be mentioned.

* **Hyper Link Tag - <a> (Anchor) Tag**
* This tag is used to establish the link between the webpage’s
* The general syntax is

**<a href=”Path of the another web pag”> Text </a>**

* **The other attributes are**

1. target=”values” 🡪 The values
2. \_blank 🡪 Open the linked webpage in new window or tab
3. \_self 🡪Open the linked webpage in the same frame
4. \_top 🡪 Open the linked webpage in the full body of the window
5. framename 🡪 Open the linked webpage in the given frame name

* **Image <img> Tag**
* This tag is used to add the images to the web pages
* The general syntax

**<img src=”Path of the image”> Text </a>**

* This tag must have one attribute called “**src” i.e. source**
* **The other attributes are**

1. height=”value” 🡪 Specifies the height of an image
2. width=”value” 🡪Specifies the width of an image.

* **Moving (marquee Tag) Tag**
* This tag is a scrolling tag and used to move the text in browser.
* The general syntax is

<marquee attributename=”value” ......> TEXT </marquee>

* The attributes are

1. **width=”Value”** 🡪 Specifies the width of the marquee. Values may be numbers.
2. **height=”Value”** 🡪 Specifies the height of the marquee. Values may be numbers.
3. **direction=”Value” 🡪** Specifies the direction of the text. Values may be **up, down, right and left.**
4. **scrollamount=”Value” 🡪** Specifies the speed of the marquee text. Values may be number.
5. **bgcolor=”Color name” 🡪** Specifies the background color.
6. **behavior=”Value” 🡪** Specifies the type of scrolling of the marquee. Value may be **scroll, slide and alternate.**

* **frame and frameset tag**
* The <frameset> tag is used to divide the browser window into frames. Each frames are indicated by using <frame> tag. The <frame> tag will open the given webpage in a particular frame.
* Don’t use the body tag in program because frameset is itself body tag.
* The general syntax is

**<frameset attributenme=”value” ....... >**

**<frame src=”Path of the webpage”>**

**.............**

**</frameset>**

* **The attributes of the frameset tag are**

1. **cols=”Values” 🡪** It divides the browser into column wise or it creates the vertical frames. For example – cols=”50,60,70”. This will create 3 frames with size 50, 60 and 70. Another example – cols=”40,\*”. This will create 2 frames with one frame size is 40 and other frame size is \*.
2. **rows=”Values” 🡪** It divides the browser into rows wise or it creates the horizontal frames. For example – cols=”50,60,70”. This will create 3 frames with size 50, 60 and 70. Another example – cols=”40,\*”. This will create 2 frames with one frame size is 40 and other frame size is \*.
3. **border=”Value” 🡪** Specifies the width of the border of each frames.

* **The attributes of the <frame> tag are**

1. name=”Name of the frame” 🡪 Gives the frame name
2. src=”Path of the webpage” 🡪 Specify which webpage we need to open in frame.

* **Form <form> Tags**
* The <form> tag is used to creating a form for user input.
* This tag contains the following tags

1. <input> tag 🡪 This tag is used to create textbox’s, button’s , passwords, radio, checklist etc.
2. The general syntax is

**<input type=”value” id=””value” name=”value” >**

**Where**

* **Type values are**

**text🡪 This will create text box**

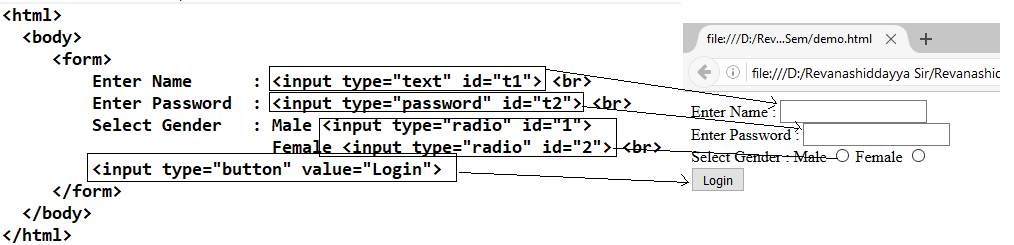
**password 🡪 This will create text box for password**

**button 🡪 This will create button icon**

**radio 🡪 This will create radio button**

**checkbox 🡪 This will create checkbox**

**Example:**

****

**2.4 Cascading Style Sheet (CSS)**

* **CSS** is a style sheet language that describe the presentation of an HTML.
* It describes how the information is displayed on the screen
* CSS can be define in 3 ways

1. **Inline** 🡪 By using style attribute inside the tag of the HTML
2. **Internal 🡪** By using <style > tag in the <head> section
3. **External 🡪** By using **<link>** tag to link an external CSS file

* **Examples:**

1. **Using Style attributes 2.Using <style> tag**

<html>

<head>

<style>

body {background-color: lightgrey;}

h1 {color: blue;}

p {color: green;}

</style>

</head>

<body>

<h1>This is a heading</h1>

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

</body>

</html>

<html>

<body>

<h1 style=”color:red; font-family: verdana;”>

This is a heading</h1>

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

</body>

</html>

|  |
| --- |
| -----link.html-----  <html>  <head>  <link rel="stylesheet" href="l1.css">  </head>  <body>  <h1>This is a heading</h1>  <p>This is a paragraph.</p>  </body>  </html> |

|  |
| --- |
| ------l1.css-----  body {  background-color: blue;  }  h1 {  color: blue;  }  p {  color: green;  } |

**3.Using <link> tag**

* The following are the some of the attributes of CSS

1. background-color:colorname
2. background-image:url(path of image)
3. border-color:colorname
4. border-width:number
5. width:number
6. height:number
7. font-family:values (Verdana, Times New Roman Arial etc)
8. font-size:number
9. margin-right:number
10. margin-left:number
11. margin-bottom:numer
12. margin:number
13. color:colorname
14. text-align:value(left,right, center or justify)

* **<div >tag**
* The <div> tag defines a division or a section in an HTML document.
* The <div> tag is used as a container for HTML elements - which is then styled with CSS or manipulated with JavaScript.
* The <div> tag is easily styled by using the class or id attribute.

|  |
| --- |
| <html>  <head>  <style>  .myDiv {  background-color: lightblue;  text-align: center;  }  </style>  </head>  <body>  <h1>The div element</h1>  <div class="myDiv">  <h2>This is a heading in a div element</h2>  <p>This is some text in a div element.</p>  </div>  <p>This is some text outside the div element.</p>  </body>  </html> |

**Example:-**

**2.4.1 Positioning the Elements**

* The web designer has a full control over the positioning of elements in dynamic HTML.
* There are 5 style properties

1. **Position**
2. **Left**
3. **Right**
4. **Top**
5. **bottom**

* There are 3 possible values/attributes of position property

1. **Absolute Positioning**
2. **Relative Positioning**
3. **Static Positioning**

**2.14.1 Absolute Positioning**

* Absolute positioning means the web designer has display the elements exactly where he wants it.
* Web designer can position attributes like top, left right to set the location
* **Example –**

<html>

<head>

**<style type="text/css">**

**.dd{**

**position:absolute; left : 50px; top:70px; color:red;**

**}**

**</style>**

</head>

<body>

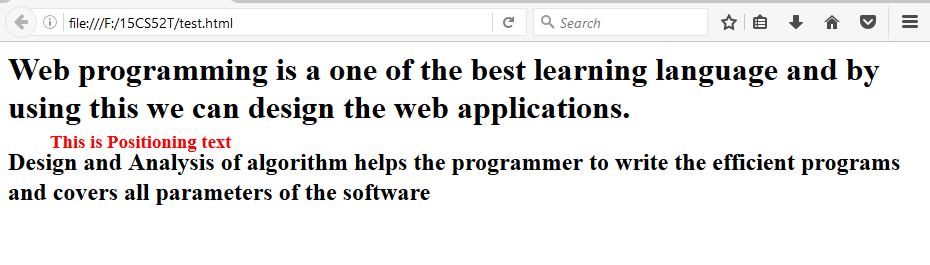
<h1> Web programming is a one of the best learning language and by using this we can design the web applications.</h1>

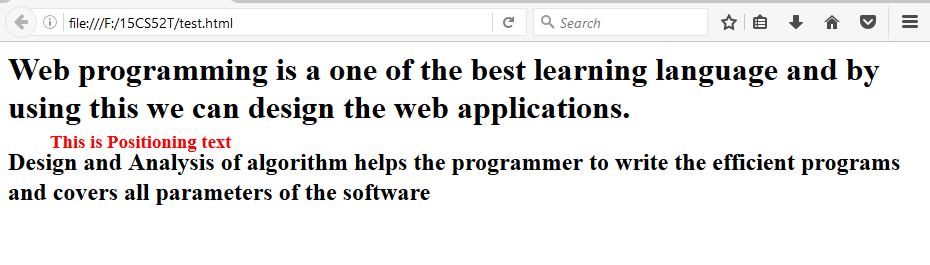
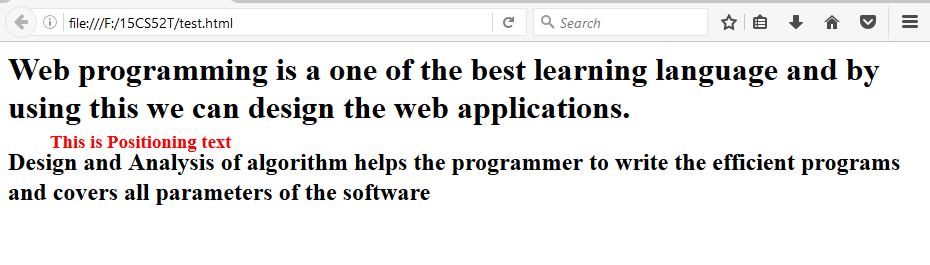
<h2> Design and Analysis of algorithm helps the programmer to write the efficient programs and covers all parameters of the software</h2>

<h3 class="dd" > This is Positioning text</h3>

</body>

</html>

****

****

**2.14.2 Relative Positioning**

* If web designer assign the position attribute as a **relative** then corresponding element gets displayed to the relative position of the previous content.
* The **relative positions get decided with respect to the previous contents**.
* **Example :**

<html>

<head>

**<style type="text/css">**

**.dd{**

**position:relative;**

**left : 50px;**

**top:70px;**

**color:red;**

**}**

**</style>**

</head>

<body>

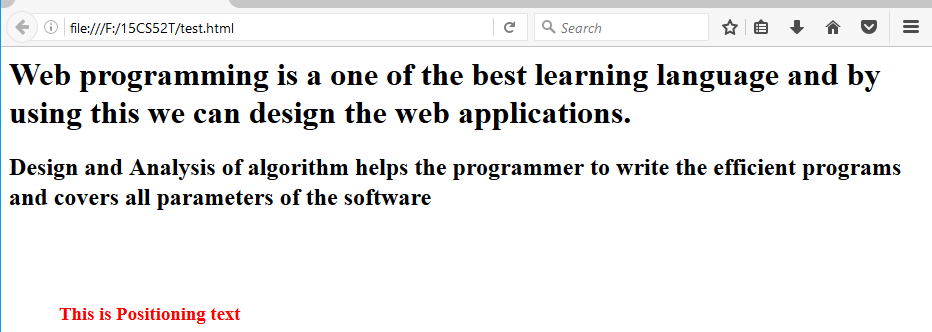
<h1> Web programming is a one of the best learning language and by using this we can design the web applications.</h1>

<h2> Design and Analysis of algorithm helps the programmer to write the efficient programs and covers all parameters of the software</h2>

<h3 class="dd" > This is Positioning text</h3>

</body>

</html>

****

**2.5 Introduction to JavaScript**

* JavaScript was developed by Netscape in 1995. The original name of **JavaScript is LiveScript**. Sun Microsystems joined the Netscape and change name to the JavaScript.
* JavaScript can be effectively used for interaction with the users.
* JavaScript is a most popular **client side scripting language supported by all browsers**
* JavaScript can be implemented by using **<script> tag.**
* The **document.getElementById()** function is used to receive the values of the elements
* “**Number**” is used to convert the text into the number format

**Exercise – 9**

**Design and create web page with JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient**

<html>

<script>

function add()

{

var n1=Number(document.getElementById('t1').value);

var n2=Number(document.getElementById('t2').value);

var sum;

sum= n1 + n2;

alert("Sum :"+sum);

}

function sub()

{

var n1=Number(document.getElementById('t1').value);

var n2=Number(document.getElementById('t2').value);

var s;

s= n1 - n2;

alert("Substract:"+s);

}

function mul()

{

var n1=Number(document.getElementById('t1').value);

var n2=Number(document.getElementById('t2').value);

var m;

m= n1 \* n2;

alert("Multiplication:"+m);

}

function div()

{

var n1=Number(document.getElementById('t1').value);

var n2=Number(document.getElementById('t2').value);

var d;

d= n1 / n2;

alert("Division:"+d);

}

</script>

<body>

<form>

Enter First Number : <input type="text" id="t1"> </br>

Enter Second Number : <input type="text" id="t2"> </br>

<input type="button" id="b1" value="SUM" onclick="add();">

<input type="button" id="b1" value="SUB" onclick="sub();">

<input type="button" id="b1" value="MUL" onclick="mul();">

<input type="button" id="b1" value="DIV" onclick="div()">

</form>

</body>

</html>

**Unit 3:business process automation/ERP**

* **Introduction to BPA:-**
* Business process automation (BPA) is the use of technology to automate repeatable, day-to-day tasks.
* It Achieve [digital transformation](https://en.wikipedia.org/wiki/Digital_transformation), increase [service quality](https://en.wikipedia.org/wiki/Service_quality), improve service delivery or contain costs.
* An **organizational structure**
* An organizational structure is a system that outlines how certain activities are directed in order to achieve the goals of an organization. These activities can include rules, roles, and responsibilities.

a

## Types of organizational structures are:

1. [**Hierarchical org structure**](https://www.lucidchart.com/blog/types-of-organizational-structures#hierarchical)**:**the chain of command goes from the top (e.g., the CEO or manager) down (e.g., entry-level and low-level employees) and each employee has a supervisor.
2. [**Functional org structure**](https://www.lucidchart.com/blog/types-of-organizational-structures#functional)**:** employees are organized according to their **specific skills** and their corresponding function in the company. Each separate department is managed independently.
3. [**Horizontal or flat org structure**](https://www.lucidchart.com/blog/types-of-organizational-structures#horizontal-flat)**:** A horizontal or flat organizational structure fits companies with few levels between upper management and staff-level employees.
4. [**Network org structure**](https://www.lucidchart.com/blog/types-of-organizational-structures#network)**:** A network organizational structure makes sense of the spread of resources. It can also describe an internal structure that focuses more on open communication and relationships rather than hierarchy.

* **Workflows:** A **Workflow** is a sequence of tasks that processes a set of data.

**Components :**

* **Input**: The materials and resources that are required to complete a step.
* **Transformation**: A specific set of rules that dictate how the input is received and what is done to it.
* **Output**: The materials and resources that are produced by the step and that act as input to the next step(s).
* **Use case:** describesa sequence of actions initiated by the actors**.**

**Steps to create workflow diagram:**

* 1. Open smart draw software using your gmail id and password.
  2. Search workflow templates,and click on workflow.
  3. Using workflow symbols draw a following diagram.

**Workflow diagram for education loan:**

Enquiry with bank officer

Take application form

Submit application form along with all proof

Processing the education loan

Take a approve from the higher authority

If education loan sanction

Sanctioning the loan as per the receipt

Loan amount will sanctioned until the education is completed

no

if agree to the eduction loan

no

if all documents are availble

no

**UNIT 4: Introduction to Cloud**

**4.1 Cloud services:**

**4.1.1 Google drive:**

* Google Drive is a [file storage](https://en.wikipedia.org/wiki/File_hosting_service)  and synchronization service developed by [Google](https://en.wikipedia.org/wiki/Google). Launched on April 24, 2012, Google Drive allows users to store files on their servers, and [share files](https://en.wikipedia.org/wiki/File_sharing).
* Google Drive offers users 15 [GB](https://en.wikipedia.org/wiki/Gigabyte) of free storage through [Google One](https://en.wikipedia.org/wiki/Google_One).
* Google One also offers 100 GB, 200 GB, 2 [TB](https://en.wikipedia.org/wiki/Terabyte), 10 TB, 20 TB, and 30 TB, offered through optional paid plans.
* Files uploaded can be up to 5 terabytes in size.

**4.1.2 Google docs:**

* Google docs is word processor included as part of the free web based Google docs editors offered by Google.
* It also included Google sheets Google slides Google forms Google sites etc.

**4.1.3 Google Co-lab:**

* Colaboratory, or “**Colab**” for short,is a free Jupyter notebook product from Google research.
* It allows anybody to write and execute arbitrary python code through the browser.

and is especially well suited to machine learning, data analysis and education.

* Jupyter notebook is open source web based phython editor with run on your browser.
* For example: []print(“hello world”)

Hello world

**UNIT 5 - CYBERSECURITY AND SAFETY**

* **Browser settings** make it possible for each of us to customize our browsing experience on the Internet.
* **privacy policies**: A Privacy Policy is a legal statement that specifies what the business owner does with the personal data collected from users, along with how the data is processed and for what purposes.

## A password policy is a set of rules designed to enhance computer security by encouraging users to employ strong [passwords](https://en.wikipedia.org/wiki/Password) and use them properly.

## 

## Password length and formation

* the use of both upper-case and lower-case letters ([case sensitivity](https://en.wikipedia.org/wiki/Case_sensitivity))
* inclusion of one or more numerical digits
* inclusion of special characters, such as @, #, $
* prohibition of words found in a password [blocklist](https://en.wikipedia.org/w/index.php?title=Blocklist_(computing)&action=edit&redlink=1" \o "Blocklist (computing) (page does not exist))
* prohibition of words found in the user's personal information
* prohibition of use of company name or an abbreviation
* **Cyber security**: is the practice of defending computers, servers, mobile devices, electronic systems, networks, and data from malicious attacks. It's also known as information technology security or electronic information security.

**Types of cyber threats:** The threats countered by cyber-security are three-fold:

* [**Cybercrime**](https://www.kaspersky.com/resource-center/threats/cybercrime)**:** includes single actors or groups targeting systems for financial gain or to cause disruption.
* **Cyber-attack:** often involves politically motivated information gathering.
* **Cyber terrorism:** is intended to undermine electronic systems to cause panic or fear.
* **Common security threats are:**

### Phishing

[Phishing](https://www.kaspersky.com/resource-center/threats/spam-phishing) is when cybercriminals target victims with emails that appear to be from a legitimate company asking for sensitive information. Phishing attacks are often used to dupe people into handing over credit card data and other personal information.

### Man-in-the-middle attack

### A man-in-the-middle attack is a type of cyber threat where a cybercriminal intercepts communication between two individuals in order to steal data. For example, on an unsecure WiFi network, an attacker could intercept data being passed from the victim’s device and the network.

### Denial-of-service attack

### A denial-of-service attack is where cybercriminals prevent a computer system from fulfilling legitimate requests by overwhelming the networks and servers with traffic. This renders the system unusable, preventing an organization from carrying out vital functions.

1. **Virus:**

A self-replicating program that attaches itself to clean file and spreads throughout a computer system, infecting files with malicious code.