

MVC: Model View Controller

1. **View** - Frontend
 - 1.1. Html5
 - 1.2. Css3
 - 1.3. Js
 - 1.4. Bootstrap
 - 1.5. JQuery
 - 1.6. Angular
 - 1.7. React js
2. **Controller** - Backend
 - 2.1. Php
 - 2.2. Java
 - 2.2.1. Core java
 - 2.2.2. Advanced java
 - 2.2.3. Spring Boot
 - 2.3. Python
 - 2.4. .net
 - 2.5. c#
 - 2.6. R
3. **Model** - Database Connectivity
 - 3.1. Mysql
 - 3.2. Postgre

SDLC - Software Development Life Cycle

1. Requirement gathering & Requirement Analysis - User Stories
 - 1.1. User Requirement Specification
 - 1.2. Customer Requirement Specification
2. Design - Architects
 - 2.1. High level Design
 - 2.2. Low level Design
3. Development - Java, Python, R - White box testing
4. Testing
5. Deployment
6. Maintenance

Java: Language

Computer - binary Language (1,0)

Int age = 23; - keyword identifier = value;

1. Compilers - File
2. Interpreter - Line to Line

C, C++ - High level Languages - Compilers

Java, Python - Compilers & Interpreters

Js - interpreters

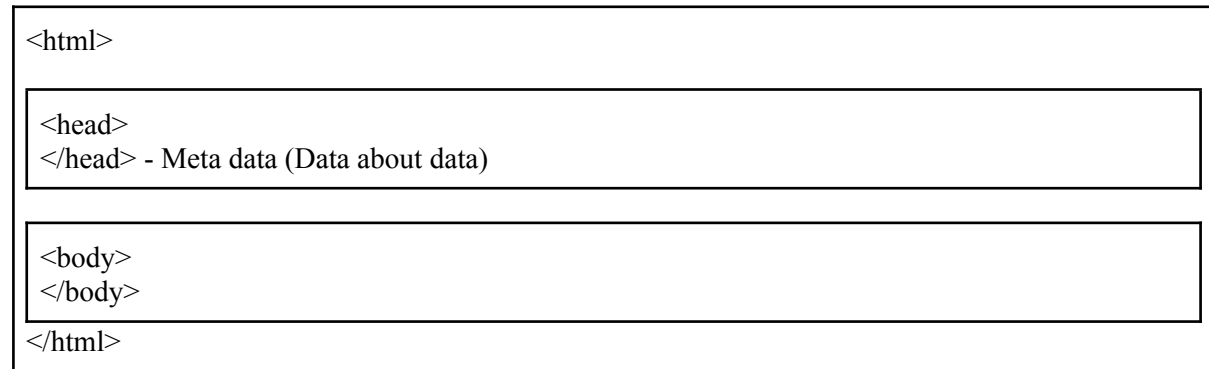
HTML5: Hyper Text Markup Language

Hyper Text - Text Within a Text / Text about a text

Syntax: <h1>HELLO</h1>

- <h1> - Open Tag
- HELLO - Content
- </h1> - Close Tag
- H1 - Element

<!DOCTYPE HTML>



Setup:

- Editor - Notepad, Notepad++, IDE (VSCode, Sublime text)
- Browser

VSCode download → System Installer

Index.html

Shift + I (Enter)

Extensions: Line Server → Ritwick Dey

Go Live

<h1>---</h1>

<h2>---</h2>

<h3>---</h3>

<h6>---</h6>

<h7>---</h7>

Lorem5 (Enter)

Right click → Open with Live server

<p>-----</p>

- Src - Property name

- — - Property Value

Create Folder img

- Absolute path position (root) - From Google
- Shortcut path position (Relative) - img/1.jpg

Assignment-1:

AP Districts → 5

Each page for each district → Description → Paragraph



home.html



about.html



contact.html

Website - Collection of pages

`home`

Href → Hyper refer

target=""_blank" → New tab

download ="sample"

Mail to

List:

1. Order list - `` ``
2. Unordered list - ``
3. Description list - `<dl></dl>` `<dt></dt>` `<dd></dd>`

Order list:

``

``

``

``

`<ol type="A">`

`<ol type="a">`

`<ol type="I">`

`<ol type="I">`

Unorder list:

``

``

`<ul type="circle">`

<ul type="square">

<ul type="none">

Description list:

<dl>

<dt>

<dd>

</dd>

</dt>

</dl>

Table:

<table border = "2">

<tr>

<th colspan = "2" rowspan = "2">

</th>

</tr>

<tr>

<td>

</td>

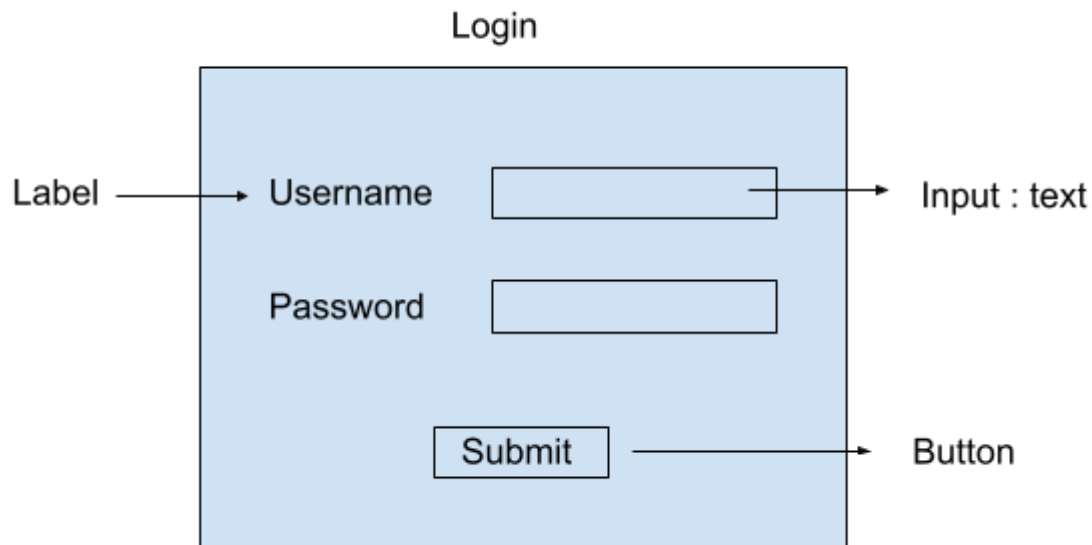
</tr>

</table>

Forms:

Input:

- Text
- Radio
- Checkbox
- Dropdown
- Label



```

For = " " id = " "
<form action = "">
<input type = "text">
<label for = "un">User Name:</label>
<input type = "text" id = "un" placeholder = "Enter Username">
<br>
<label for = "pwe">Password:</label>
<input type = "Password" name = " " placeholder = "Enter Password">
<br>
<input type = "Submit" value = "Submit">
<label for = "gn">Gender:</label>
<input type = "radio" name = " " id = "gn">Male
"
"
<br>
<label for = "email">Email</label>
<input type = "email" name = " " id = "email" placeholder = "Enter Email">
<br>
<label for = "Country">Country</label>
<input type = "Checkbox" name = " " id = "Country">IND
<br>
<label for = "State">State</label>
<select name = " " id = "state">
    <option value = "select">Select</option>
    <option value = "ap">AP</option>
</select>
<label for = "dob">DOB:</label>
<input type = "date" name = " " id = "dob">
<br>
<label for = "Upload">Upload</label>
<input type = "file" name = " " id = "upload">
<br>
<input type = "search" name = " " id = " ">
  
```

CSS:

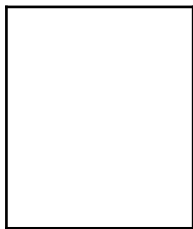
Element_Selector

```
{  
    Property name: property value;  
    "  
}
```

```
<h1>Hello</h1>
```

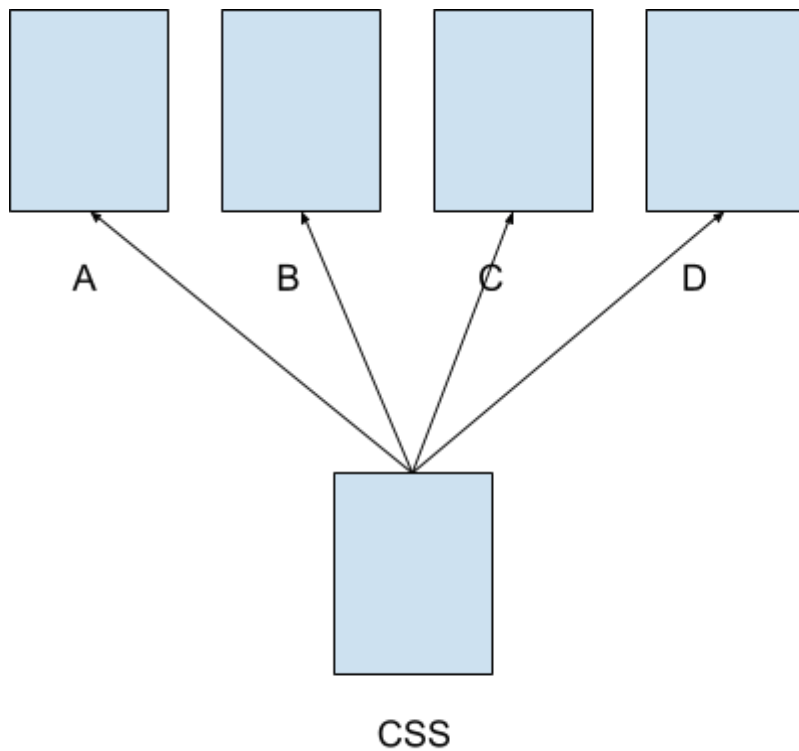
```
H1{  
    Color:  
    Fontsize:  
}
```

- **Inline** - `<h1 style = "color:red">HTML</h1>`
- **Internal**



home.html

- **External**



```
<h1 style = "color: red; text-align:center;">....</h1>  
<style>  
    H1{
```

```

        Color:blue;
        Text-align:center;
    }
</style>

```

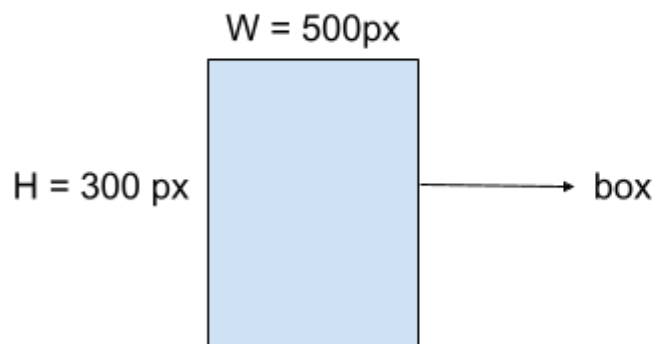
Sty.css:

```

Body{
    Background-color:burlywood;
}
<link rel = "stylesheet" href = "sty.css">

```

Div:



```
<div>
```

1. Element name (div)
2. Id (#)
3. Class (.)
4. *

```
Div{
```

```

    Width: 300px;
    Height: 400px;
    Background-color: red;

```

```
}
```

```
#box1 {
```

```

    Width: 200px;
    Height: 300px;
    Background-color: rebeccapurple;

```

```
}
```

```
#box2.box21 {
```

```

    Width: 400px;
    Height: 200px;
    Background-color: crimson;

```

```
}
```

```
<div></div>
```

```
<div id = "box1"></div>
```

```
<div id = "box2">  
    <div class = "box21">  
    </div>  
</div>
```

Assignment-2:

Table: Online Billing

Sample billing slips

Invoice - 2

Forms:

Personal Details

Salary Details

Employee Details

Container:


```
<form action = “ “>
<fieldset>
  <legend>Personal Details</legend>
  <label for = “fn”>First Name</label>
  <input type = “text” id = “fn”
```

Background:

- Bg-color
 - Color-name
 - RGB(0-255)
 - Hexa(#FFFF)
- Bg-image
 - Position
 - Size
- Linear gradient
- Blend

```
<div class = “Container”>
```

```
<style>
.container{
  Width: 50%;
  Height: 400 vh;
  Background-color: chocolate;
}
```

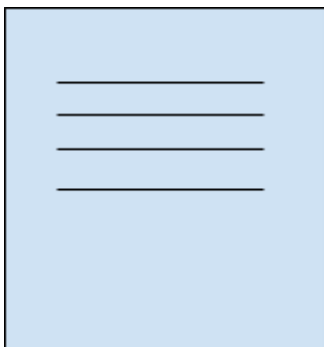
```
background-image:url(img/2.jpeg);
Background-position:center;
Background-size:cover;
```

Background-repeat: no-repeat;
background: linear-gradient(to left/right, rgba(0,0,0,0.6), rgba(0,0,0,0.6), url(img/));
0.6 → 0.0 - 0.9

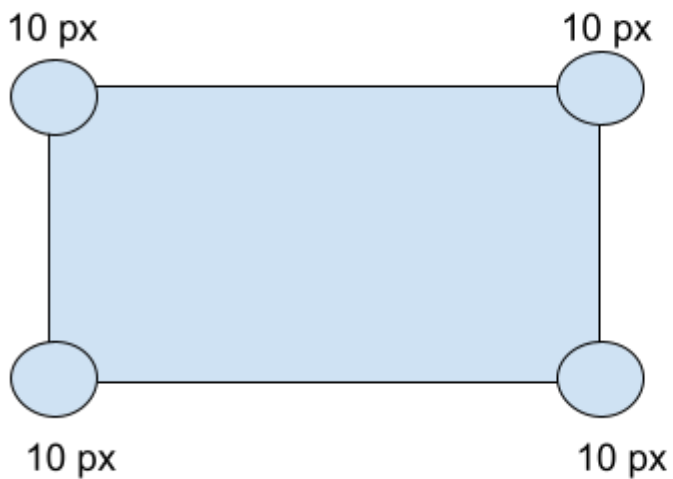
Background-repeat: repeat-x/y;
Background-position: center;
Background-size: cover;

Background-color: blueviolet;
Background-image: url(img/5.jpg);
Background-position: center;
Background-size: cover;
Background-blend-mode: lighten/darken/difference;

Border:



1. Border-style
2. Border-width
3. Border-color
4. radius



<style>

P{

Width: 300px;
Height: 300px;
Background-color: aqua;

```
    Border-style: double;
    Border-width: 5px;
    Border-color: red;
    Border-radius: 10px/20px; → Pick opposite value
    Border-radius: 50%;
}
```

```
Img{
    Width: 400px;
    Height: 400px;
    Border-style: outset;
    Border-width: 10px;
    Border-color: chocolate;
    Border-radius: 30px;
    Border-radius: 50%;
}
}
```

Display:

```
<h1>HELLO</h1>      HELLO
<h1>Hi</h1>          Hi
<h1>WELCOME</h1>     WELCOME
```

HELLO Hi WELCOME

1. Inline - Cannot define width & height
2. Inline-block - can define “
3. Block
4. None
5. Flex-box

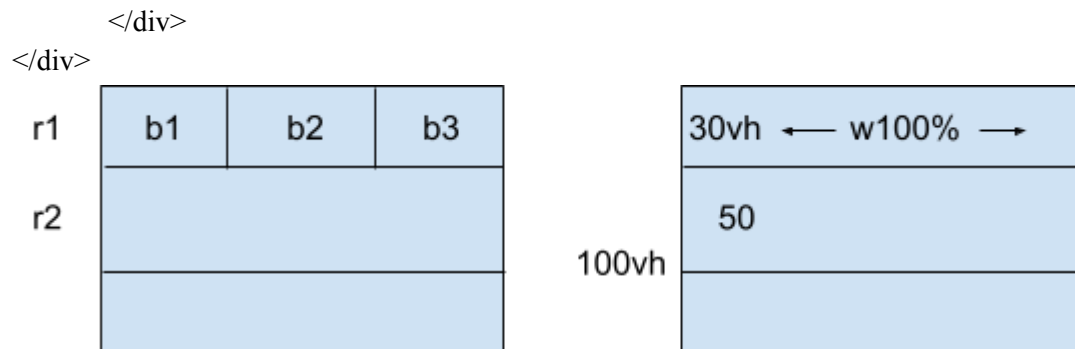
```
<style>
P{
    Display: inline;
    Width: 150px;
    Background-color: aqua;
    Height: 20px;
}
```

Display: inline-block;

Display: block;

Display: none;

```
<div class = “container”>
    <div class = “block1”>
```

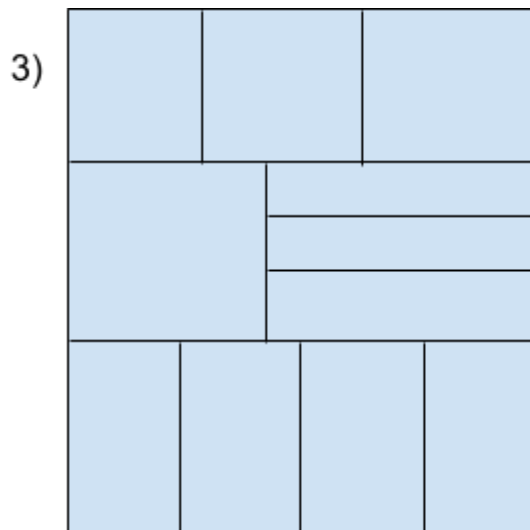
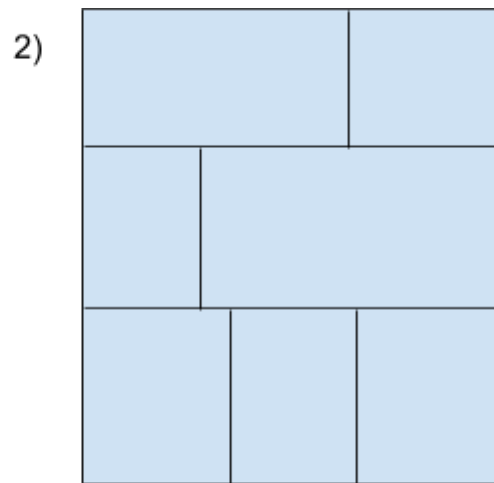
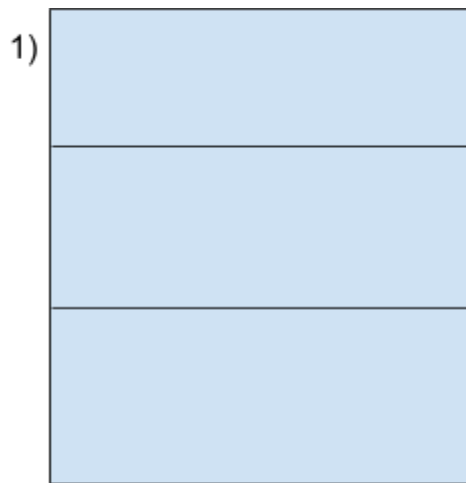


```

<div class = "container">
<div class = "block1">
.box*3
<div class = "box1">box1</div>
<div class = "box2">box2</div>
<div class = "box3">box3</div>

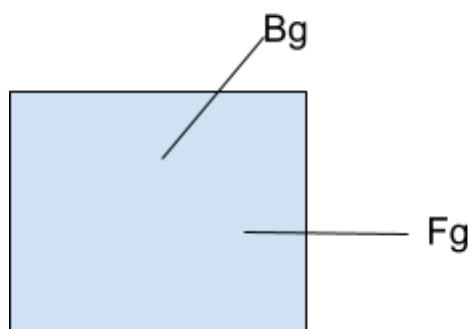
<style>
.container{
    Width: 100%px;
    Background-color: aqua;
}
.block1{
    Width: 100%;
    Height: 300vh;
    Background-color: brown;
    Display: flex;
    Justify-content: space-between/space-evenly;
}
.block1.box1{
    Width: 30%;
    Height: 30vh;
    Background-color: blue;
}
.block1.box2{
    Width: 30%;
    Height
  
```

Assignment-3:



```
<div>
<div class = "box">
```

Background:



Bg-color:

- Color
 - Color-name
 - rgb(0-255)
 - hexacodes(#F8F8)
- bg-linear((),()),img()
- bg-img

```
<div class = "Container">
<img src = "img/1.jpg" alt = "">
</div>
```

```
<style>
.container{
    Width: 50%;
    Height: 50vh;
    Background-color: antiquewhite;
}
Img{
    Width: 300px;
    Height: 300px;
}
```

```
Background-color: #F8F8F8;
background:linear-gradient(to right/left rgba(0,0,0,0),rgba(0,0,0,0));
Alpha - (0-0.9(1))
background-image:url(img/2.jpg);
Background-repeat:no-repeat;
Background-position:center;
Background-size:cover;
Background-color:brown;
Background-blend-mode:lighten;
background:linear-gradient(rgba(0,0,0,0.6),rgba(0,0,0,0.6)),url(img/2.jpg);
```

Borders:

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

```
<div class = "container">
<p>....</p>
<img src = "img/1.jpg" alt = "">
```

```
<style>
.container{
    width:100%;
    Height:100vh;
}
P{
    Width:400px;
    Height:200px;
    Border-style:double;
    Border-color:brown;
    Border-radius:10px 20px;
```

```

}
Img{
    Width:300px;
    Height:300px;
    Border-style:groove;
    Border-width:10px;
    Border-color:greenyellow;
    Border-radius:circle;
}

```

Display:

HTML <p>HTML</p>

CSS <p>CSS</p>

JS <p>JS</p>

HTML CSS JS

1. Inline
2. Inline-block
3. Block
4. None
5. flex

<div class = “container”>

<p>html</p>

<p>css</p>

<p>java</p>

<p>python</p>

<style>

```

.container{
    width:100%;
    Height:100vh;
}

```

}

P{

Background-color:aqua;

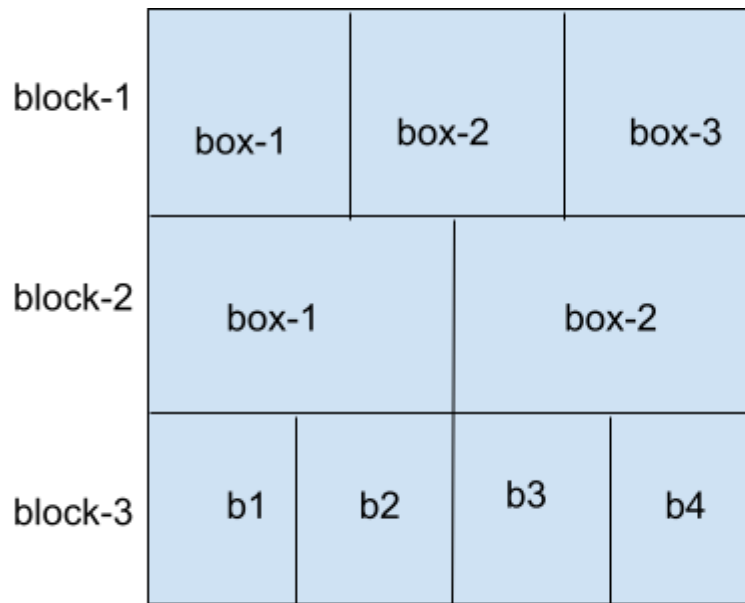
Display:inline-block;

Width:150px;

Height:30px;

}

Flex:



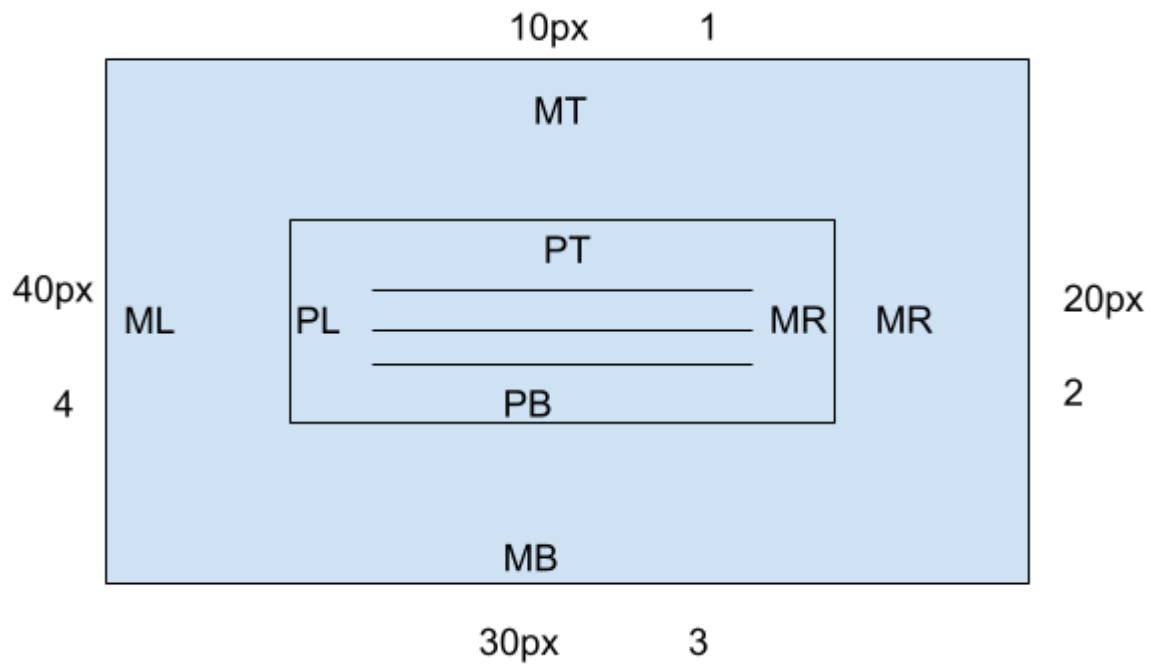
```

<div class = "container">
<div class = "block-1">block-1</div>
    <div class = "box-1">box-1</div>
    <div class = "box-2">box-2</div>
    <div class = "box-3">box-3</div>
<div class = "block-2">block-2</div>
<div class = "block-3">block-3</div>

<style>
.container{
    width:100%;
    Height:100vh;
    Background-color:aqua;
}
.block-1{
    justify-content:space-around;/between/evenly
    width:100%;
    Height:
    Display:flex;
}
.block-1.box-1{
    width:30%;
    Height:30vh;
    Background-color:darkolivegroove;
}

```

Margin & padding:



Margin: 10px 20px 30px 40px

```
<div class = "container">
```

```
<p>...</p>
```

```
<style>
```

```
.container{
```

```
}
```

```
P{
```

```
Border:solid 5px red;
```

```
Width:400px;
```

```
Height:200px;
```

```
Margin-left:10px;
```

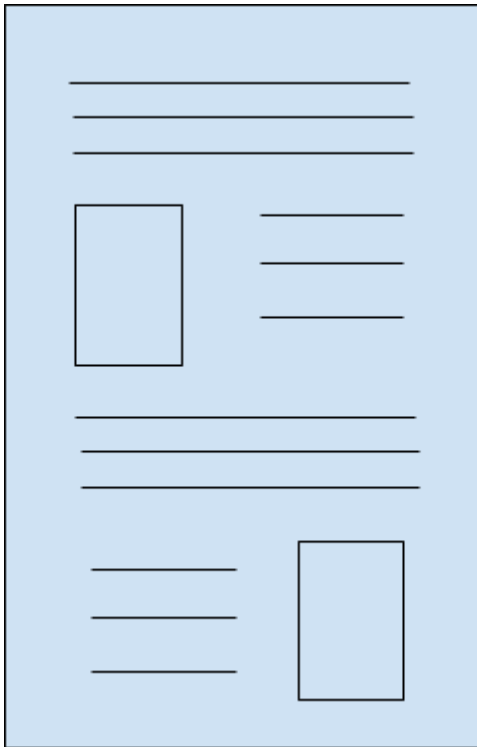
```
Margin-top:
```

```
Margin:20px 50px 60px 100px;
```

```
Padding:20px;
```

```
}
```

Float:



<p>...</p>

<img src = "img/3.jpg" alt = "" width = "150px" height

<style>

#img1 {

Width:

Height:

Float:left;

}

#img2{

Margin:20px;

}

Font:

<h3>...</h3>

<h4>...</h4>

<style>

H3{

Font-family: sans-serif;

Font-size: 2.1rem;

Font-style: italic;

Color: brown;

}

H4{

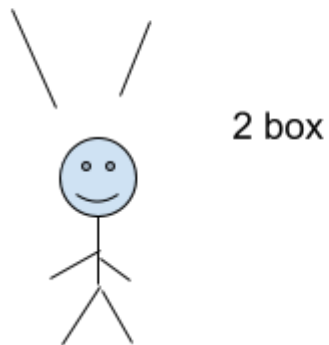
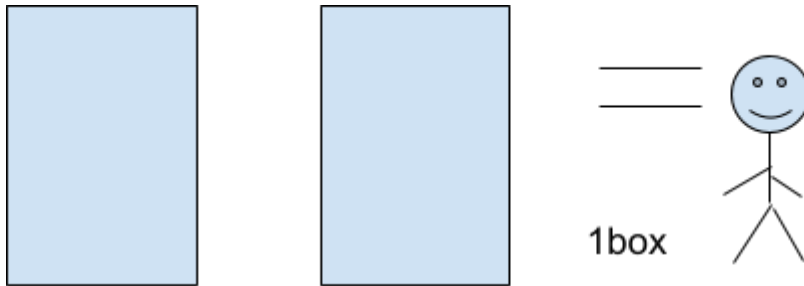
Font-family:serif;

Font-weight:900;

font-variant:small-caps;

}

Positions:



```
<div class = "box1">box1</div>
```

```
<div class = "box2">box2</div>
```

```
<style>
```

```
.box1 {
```

```
    Width:150px;
```

```
    Height:150px;
```

```
    Background-color:aqua;
```

```
    Left:200px;
```

```
    position:absolute;/relative
```

```
}
```

```
.box2 {
```

```
    Width:150px;
```

```
    Height:150px;
```

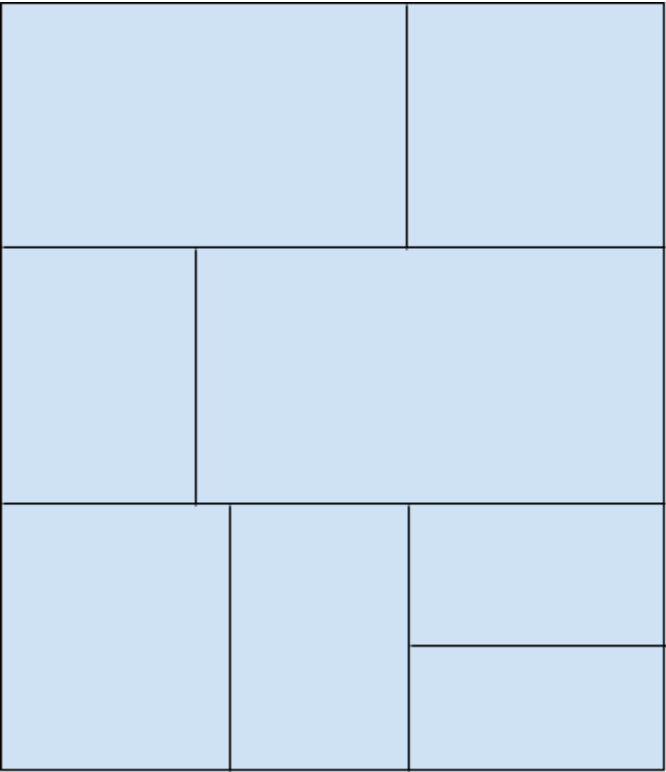
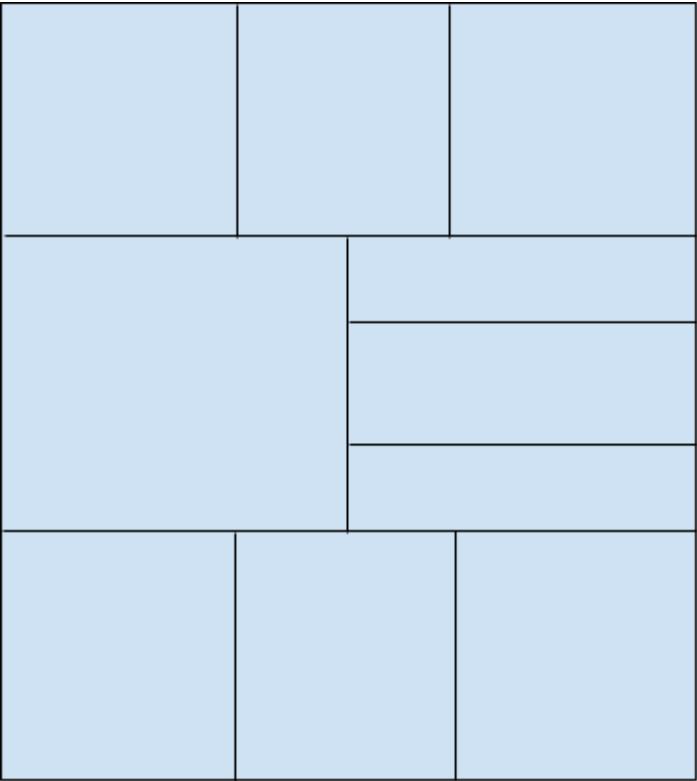
```
    Background-color:brown;
```

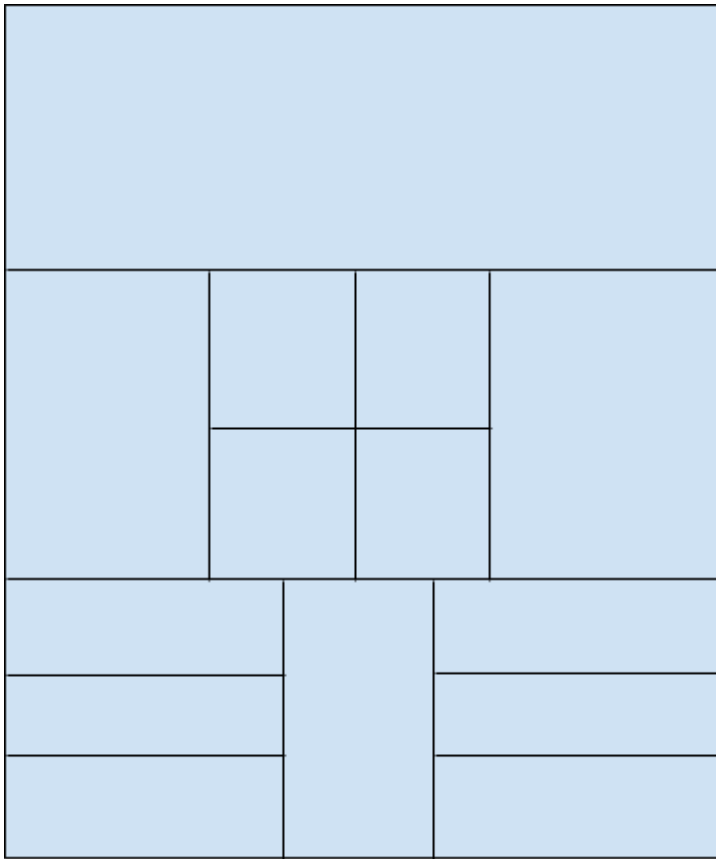
```
    Top:-50px;
```

```
    Position:absolute;
```

```
}
```

Assignment 4:





Animations:

@keyFrame name
0% - 100%

Transactions:

<h1>...</h1>
<div

<style>

H1 {

Margin-top: 100px;

Transform: rotate(-40deg);

Text-align: center;

Animation-name: mytext;

Animation-duration: 5s;

}

@keyframes mytext{

0%{

Transform: translate(-500px);

Color: red;

Transform: rotate(30deg);

}

25%{

Transform: translate(-200px);

```

Color: blue;
}
50%{
Transform: translateX(500px);
Color: chartreuse;
}
75%{
Transform: translateX(900px);
Color: chocolate;
}
100%{
Transform: translate(0);
Color: cornflowerblue;
}
}
}
.box1 {
Width: 120px;
Height: 120px;
Background-color: red;
Animation-name: mytext;
Animation-duration: 5s;
}

```

Hover:

<h1>...</h1>

```

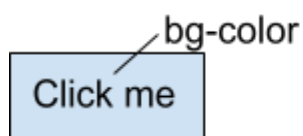
<style>
H1 {
Text-align: center;
}
H1:hover{
Color:red;
}

```

Fontawesome - icons

Icons

Phone



HTML

```
<button class="btn">Click me</button>
```

CSS

```
.btn{
```

W:

H:
Bg:
Color:
Border:
}

Bootstrap:

```
<button class="btn">Click me</button>
```

```
<style>
.btn{
Width: 200px;
Height: 150px;
Background-color: brown;
Color: white;
Font-size: 40px;
Border-radius: 20px;
Border: none;
}
```

Bootstrap

Responsive - Any device compatibility

Include via CDN

Copy & place in header - bootstrap.min.css & bootstrap.bundle.min.js

```
<button class="btn btn-primary m-4">Click me</button>
<p class="h1"/>display-1</p>
```

```
<style>
.btn{
Width: 200px;
}
```

Docs - opensource

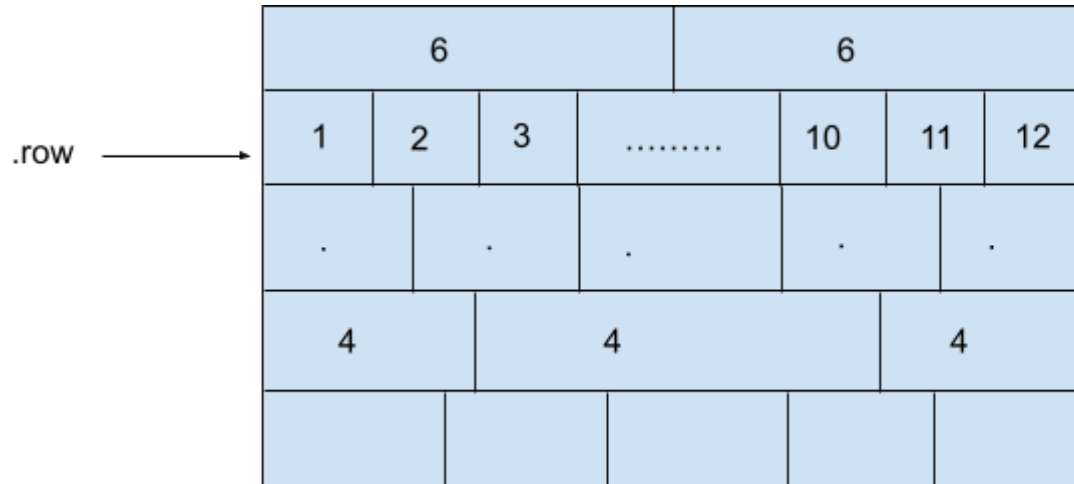
Container:

1. Container → Max-width
.container
2. Container-fluid → Full width
.container-fluid

```
<h1>...</h1>
<div class="container">
<h1>...</h1>
</div>
<div class="container-fluid">
<h1>...</h1>
```

<div>

Grid: Combination of rows & columns



```
<div class="container">
<div class="row">
<div class="col blue">1</div>
.
.
.
<div class="col red">12</div>
```

```
<style>
.red{
Background-color: red;
}
.blue{
Background-color:blue;
}
```

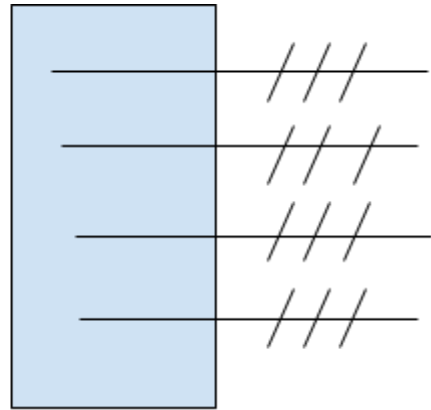
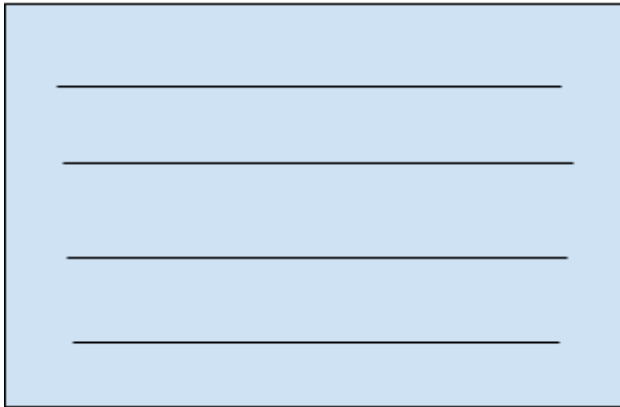
Inspect → responsive

```
<p>...</p>
<div class="col-8 red">1</div>
<div class="col-4 blue">2</div>
```

Assignment 5:

Bootstrap grid with previous assignment

Breakpoints:



Col-sm → 576px
 Col-md → 768px
 Col-lg → 992px
 Col-xl → 1200px
 Col-xxl → 1400px

```
<div class="container">
<div class="row">
<div class="col-md-6"></div>
<div class="col-sm-6"></div>
</div>
</div>
```

```
<head>
<link...
<script... → bootstrap
</head>
<body>
<div class="container">
<h1>...</h1>
<p>...</p>
<p class="h1">...</p> → 1 - 6
<p class="display-1">...</p> → 1- 6
<p class="small">...</p>
<p class="mark">...</p>
<p class="display-1">Lorem<div class="mark">ipsum</div>
```

Bootstrap document → Content → Typography

Text & Background colors:

class="txt"

1. Primary
2. Success
3. Info
4. Warning
5. Danger

6. Secondary
7. White
8. Dark
9. light

```

<p class="text-primary">....
<p class="text-info">...
<p class="text-warning">....
<p class="text-dark">....
<p class="text-success">...
<p class="text-secondary">...
<p class="text-light">....

```

Text-primary-30

```

<p class="text-primary bg-danger">
"Text-info bg-primary"
"Text-warning bg-info"
"Text-dark bg-light"
"Text-success bg-

```

Table:

```

<div class="container-fluid">
<div class="row">
<div class="col-md">
<table class="table">
<tr>
<th>ID</th>
<th>Name</th>
<th>Location</th>
</tr>
<tr>
<td>101</td>
<td>Vidya</td>
<td>rjy</td>
</tr>

```

```

"Table table - striped"
-bordered
-hover
-dark table-striped table-hover
-borderless
-primary table-striped table-hover
-sm
"Table-responsive"

```

Content → Tables

Images:

```
<div class="container-fluid">
<div class="row">
<div class="col-md">

</div>
</div>
</div>
```

Img-thumbnail

Rounded-circle

Content → Images

Jumbotrons:

```
<div class="container-fluid">
<div class="row">
<div class="col-md">
<div class="bg-primary text-danger mt-4 p-5 rounded">
<h1>...</h1>
<p>...</p>
</div>
</div>
</div>
```

Alerts:

```
<div class="container-fluid">
<div class="row">
<div class="col-md">
<div class="alert alert-info alert-dismissible">
<p>....</p>
<strong>Click me!</strong> —> <a href="#" class="alert-link">Read more!</a>
<button type="button" class="btn-close" data-bs-dismiss="alert"></button>
</div>
```

Button:

```
<div class="container-fluid">
.
.
.
<div class="d-grid">
<button class="btn btn-outline-primary btn-lg btn-block">Click</button>
```

Components → button

Cards:

```
<div class="row">
```

```

<div class="col-md-12">Navbar</div>
<div class="row">
<div class="col-md-12">Slide</div>
<div class="row mt-5">
<div class="col-md-4">Card
.
.
.
</div>

```

Forms:

```

<div class="container-fluid">
<div class="row">
<div class="col-sm-12">
<div class="mt-4 mb-4">
<label for="fname" class="form-label">FirstName</label>
<input type="text" id="fname" class="form-control">
</div>
</div>
<div class="mt-3 mb-3">
<label for="lname">Lastname</label>
<input type="text" id="lname" placeholder="Enter lastname" class="form-control">

```

Forms → Form control

Assignment 6:

Tables & Forms in html billing using bootstrap

Flipkart sign up form

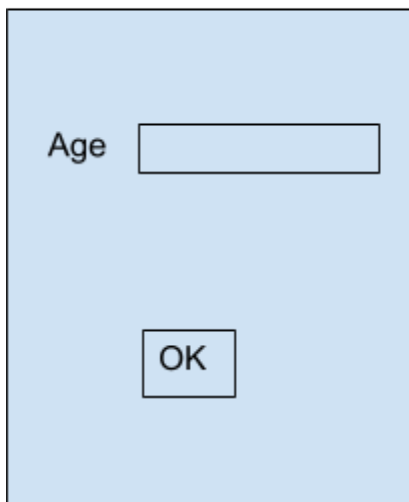
Pixabay → Cards

Royal Enfield → Clone

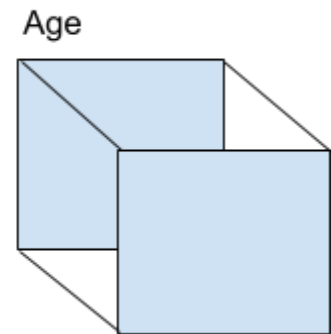
gvkinfradevelopers.com → Clone

JavaScript:

- Low level
 - Assembly Language (1,0)
- High level
 - General purpose → (C, C++, Java, Python, .net, C#, R)
 - Scripted → JS
 - Special → Android
 - OOPS → Java, Python



Keyword/database
 / identifier
 Int age = 32;
 Age = 32;
 Int = 32



Compiler - File to file translate
 Interpreter - Line to line translate

- Editor
- Browser

```
<h1 id = "test">...</h1>
```

```
<script>
```

```
document.getElementById('test').innerHTML = "Welcome to js";
```

```
</script>
```

- Head
 - <head>
 - <script>
 - Document.get
 - </script>
 - </head>
- Body
- External
 - myscript.js

```
Function msg()
```

```
{
document.write("hello");
}
```

```
<script src = "myscript.js"></script>
```

```
<input type = "submit" value = "click" onclick = "msg()">
```

```
<script>
```

```
Function msg()
```

```
{
```

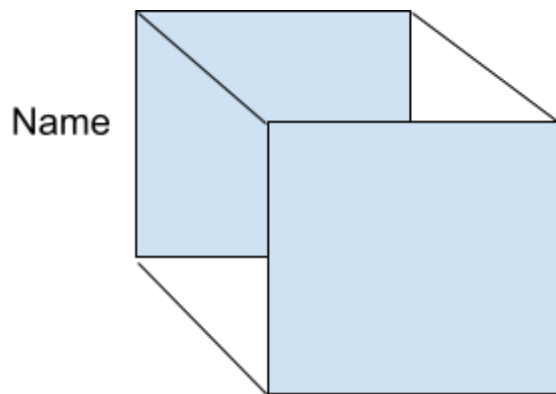
```
document.write("Welcome to js....!");  
}  
</script>
```

Memory/Storage:

- Primary
 - RAM
 - ROM
- Secondary
 - HDD
 - CD
 - Floppy Disk

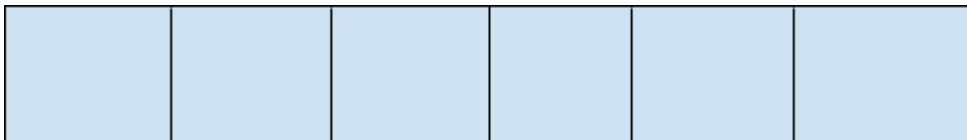
Containers:

- Variable

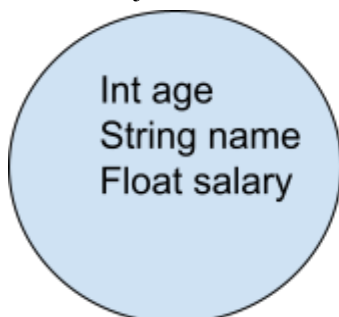


```
String name = "vidya";  
Int x = 100;
```

- Arrays



- Objects



1. Var - rewrite
2. Let - rewrite
3. Const

```
Var x = 100;  
document.write(x); 100  
X = 1000;
```

```
Let x = 1000;  
X = 2000;
```

```
Const dob = 1993;  
Let year = 2024;  
Let age = year - dob;
```

```
<script>  
var x = 100;  
Var y = 200;  
Var z = x+y;  
document.write(z);  
</script>
```

```
var/let/const var x = 100;  
document.write(x+"<br>");  
X = 1000;  
document.write(x);
```

Inspect → Console

Operator: Symbol

1. Arithmetic:

+	10+10=20	
-	10-10=0	
*	10*10=100	
/	10/10=1	Quotient
%	10%10=0	Remainder

```
<script>  
Let x = 100;  
Let y = 200;  
document.write("Sum of two numbers"+(x+y)+"<br>");  
document.write("Sub of two numbers"+(x-y)+"<br>");  
document.write("Mul of two numbers"+(x*y)+"<br>");  
document.write("Div of two numbers"+(x/y)+"<br>");  
document.write("Mod of two numbers"+(x%y)+"<br>");
```

Assignment 7:

- Power bill system
Units = 100 → 2rs
Price = 2rs
Total =

- Super market billing system
- Marks

S1 s2 s3 s4 s5 s6
 Total marks
 Average

2. Comparison / Relational operators - Boolean values T/F

<script>

Let x = 10;

Let y = 20;

document.write(x==y);

document.wirte(x<=y);

document.wirte(x>=y);

document.wirte(x!=y);

document.wirte(x<y);

document.wirte(x>y);

</script>

3. Bitwise Operators

AND - &

OR - |

XOR - ^

NOT - ~

A = 20	2	20 - 0	16	8	4	2	1	0	
B = 10	2	10 - 0	2	2	2	2	2	2	
	2	5 - 0	1	0	1	0	0	0	
	2	2 - 1							
		1 - 0							
(10 == 30 ^ 20 == 33)					1	0	1	0	0
F	F	=	F		0	1	0	1	0
					0	0	0	0	0
					1	1	1	1	0
									- &(0)
									- 1(36)

4. Logical operators

AND - &&

OR - ||

Age	G1	G2
	1-15	50-100
	5000/-	5000/-

Find Age

23

OK

(age >= 1 && age <= 15) || (age >= 50 && age <= 100)

<script>


```

Let age = 10;
if((age >= 1 && age <= 15)|| (age >= 50 && age <= 100))
{
    document.write("got 5000");
}
Else
{
    document.write("not get");
}
</script>

```

$\frac{20 > 45}{F}$ $\frac{60 \leq 75}{T}$

AND

OR

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |

5. Assignment Operator

X = 100 20 =

X = 100 + 20 +=

X = x + 20 -=

X = 120 *=

X += 20 /=

%=

6. Conditional Operator

Application

Age

OK

Driving Licence

18

18-60

Age ≥ 18 && age ≤ 60

<script>

Let age = 18;

if(age ≥ 18 && age ≤ 60)

{

document.write("Eligible for DL");

}

Else

{

document.write("Not eligible for DL");

}

| | 7 | | | 8 | | | 32 | |
|---|---|---|---|---|---|---|----|----|
| 2 | 7 | 3 | 2 | 8 | 4 | 2 | 32 | 16 |
| | 6 | | | 8 | | | 32 | |
| | 1 | | | 0 | | | 0 | |

Let n = 5;

if(n%2==0)

{

document.write("Even");

}

Else

{

document.write("Odd");

}

Marks = 35-100

if(m ≥ 35 && m ≤ 100)

{

}

If else Ladder:

<script>

Let m = 23;

if(m ≥ 35 && m ≤ 55)

{

document.write("C Grade");

}

Else if(m ≥ 55 && m < 75)

{

Document.wirte("B Grade");

}

Else if(m ≥ 75 && m ≤ 100)

{

document.write("A Grade");

}

```

Else
{
    document.write("Not Valid");
}
</script>

```

Switch(Expression) – int / String

```

{
    Case 1: _____
        Break;
    Case 2: _____
    Case 3: _____
    .
    .
    .
    Default: _____
}
<script>
Var ch = "a";
Var r;
switch(ch)
{
    Case "a":
        R = "a";
        Break;
    Case "b":
        R = "b";
        Break;
    Default:
        R = "not valid";
}
document.write(r);

```

Assignment 8:

- Speed Test

| | | |
|------------|------------|-------------|
| 20 - 45 km | 46 - 74 km | 75 - 120 km |
| Normal | Moderate | Danger |

- Billing System

Product-id:

Product-name:

Price:

Quantity:

OP

Billing System

P_id:

P_name:

Price:

Quantity:

Total:

GST:

Discount:

Grand total:

3. Condition

a. T = 1000 - 20000

GST 12%

Offer 5%

b. T = 21000 - 39000

GST 13%

Offer 8%

c. T = 40000 - 100000

GST 18%

Offer 10%

I/P:

Price

Quantity

$T = p * q$

O/P:

Price

Quantity

Total

GST

Offer

Grand total

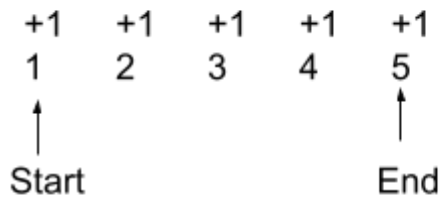
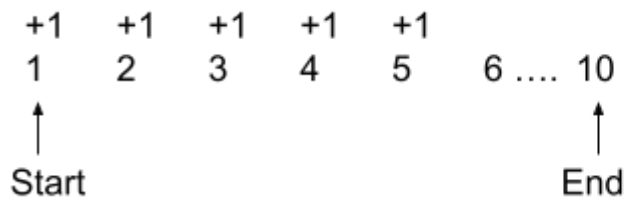
Iterations/Loops:

1. For loop
2. While loop
3. Do-while loop

Hello

```
document.write("Hello");
```

```
document.write("Hello");
```



```

    1    2    4
for(int i=1;i <= 5;i++)
{
    3
}

```

```

for(Expression 1;Expression 2;Expression 3)
{
}

```

Exp-1

Initialization

Int i = 1

Exp-2

Condition

I <= 5

Exp-3

increment/decrement

i++/i--

For loop:

<script>

for(let i=1;i <= 5;i++)

{

document.write(i+"hello");

}

</script>

-1 -1

10 9 8 1

I = 10;i >= 1;i--

While	Do while
Pre Condition	Post Condition

While(Exp) { Statements }	Do{ Statements } while(Exp);
------------------------------------	------------------------------------

While:

```
<script>
Let i = 1;
while(i <= 10)
{
    document.write("hi");
    I++;
}
</script>
```

Do while:

```
<script>
Let i = 1;
Do{
    document.write(i);
    I++;
}while(i <= 10);
```

```
1 * 5 = 5
2 * 5 = 10
3 * 5 = 15
4 * 5 = 20
```

```
.
.
.
```

```
10 * 5 = 50
```

```
for(let i = 1;i <= 10;i++)
{
    d.w(i+"*"+5+"="+i*5);
}
```

Sum of N numbers:

```
N = 5
1+2+3+4+5=15
Let sum = 0;
for(let i = 1;i <= n;i++)
{
    Sum += i;
}
d.w(sum)
```

```
<script>
```

```

Let n = 10;
Let i = 1;
Let sum = 0;
while(i <= n)
{
    Sum += i;
    I++;
}
document.write(sum);
</script>

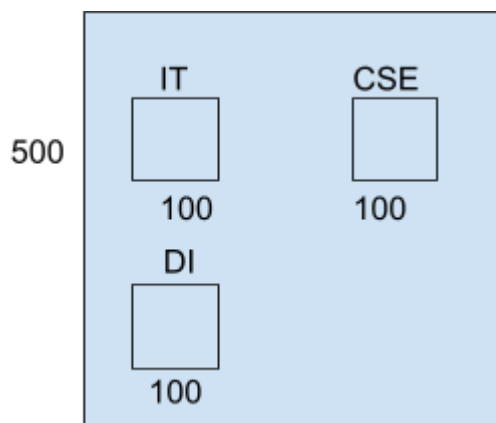
```

```

Do{
    document.write(i);
    I++;
}while(i >= n);

```

Functions: Reusability & portability



```

d.w(a+b)
d.w(a-b)
d.w(a*b)
d.w(a/b)
d.w(a%b)

```

20 Req → 1 → Registration & Signup

Signup()

```

{
    _____
    _____
    _____
}

```

Signin()

```

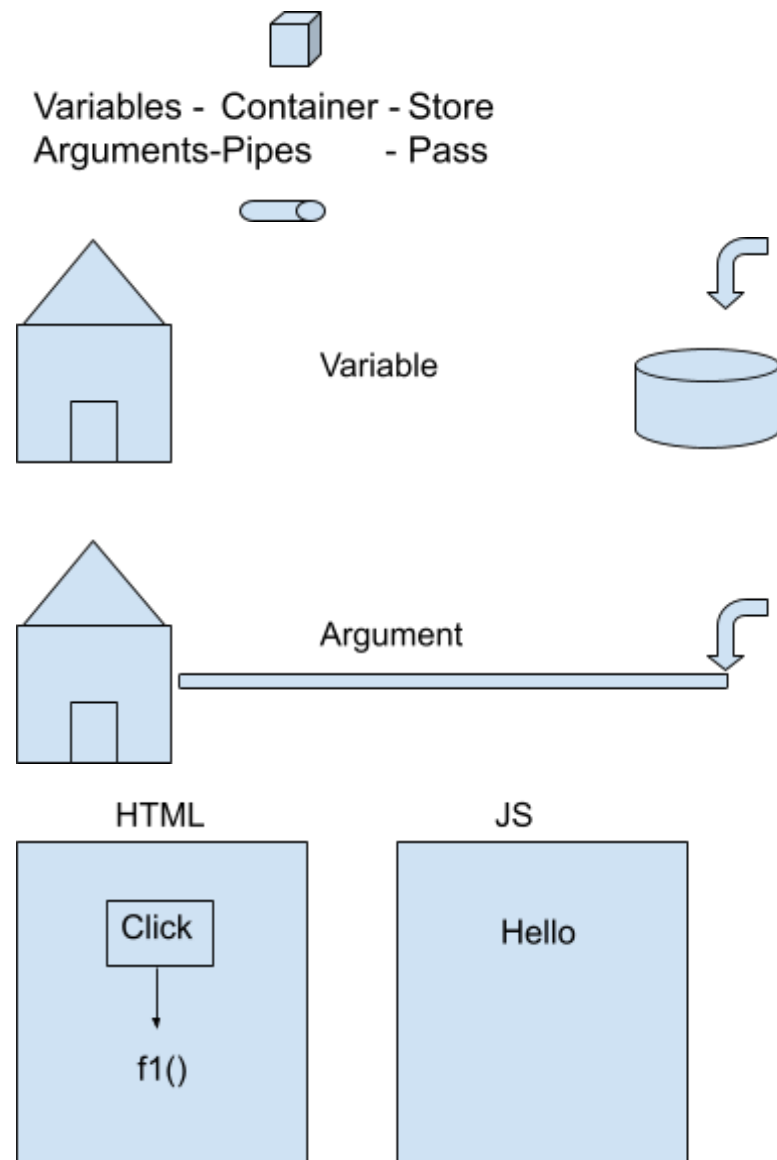
{
    _____
    _____
    _____
}

```

```

}
Function add(arg1, arg2....)
{
    Group of statements
}

```



Onclick

```

Function f1()
{
    alert("Hello");
}

```

```

<body>
<input type = "button" value = "click" onclick = "f1()">
<script>
Function f1()
{

```



```
        alert("Hello");  
    }  
</script>  
</body>
```

Assignment 9:

1. Factors
 $N = 5$
2. Factorials
 $5!$
3. 1-100 Sum of odd numbers