**Experiment 1: Develop static webpage using HTML tags,list tags,image tags.**

<html>

<head>

<U><h1 align="center">SHRI VISHNU ENGENEERING COLLEGE FOR WOMEN</h1></U>

<h2 align="center">(svecw)</h2>

</head>

<body>

<hr></hr>

</br>

<i><p>BRANCHES</P></i>

<ol>

<b><li>CSE</li></b>

<UL>

<li>section A</li>

<li>section B</li>

<li>section C</li>

</ul>

<b><li>IT</li></b>

<UL>

<li>section A</li>

<li>section B</li>

<li>section C</li>

</ul>

<b><li>ECE</li></b>

<UL>

<li>section A</li>

<li>section B</li>

</ul>

<b><li>EEE</li></b>

<UL>

<li>section A</li>

</ul>

<b><li>CIVIL</li></b>

<UL>

<li>section A</li>

</ul>

<b><li>MECHANICAL</li></b>

<UL>

<li>section A</li>

</ul>

</ol>

<hr></hr>

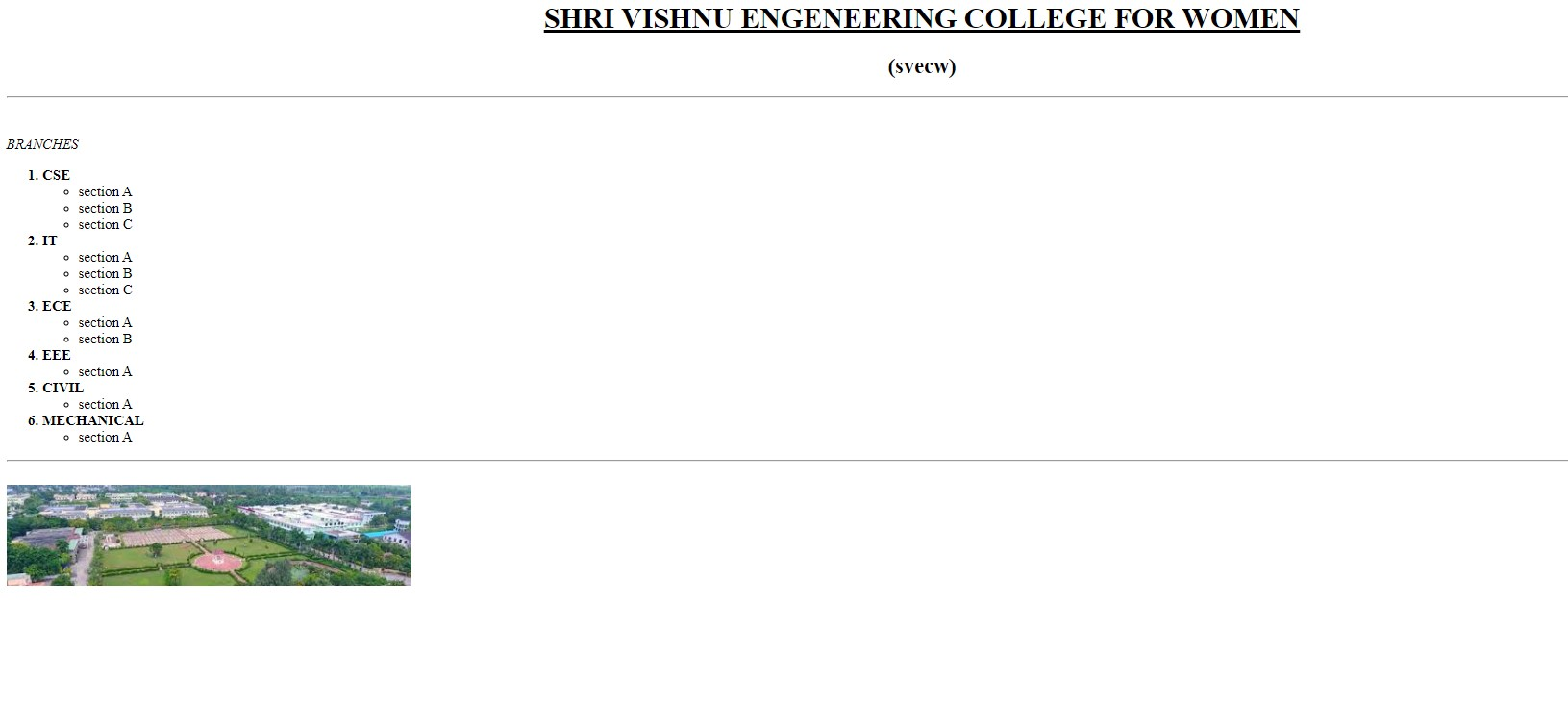
<br/>

<img src="/home/user/Pictures/svecw.jpeg"/>

</body>

</html>

Output:



**Experiment 2: Demonstrate table tag to create different orientation of table in static web page.**

<html>

<head>

<u><h1>TABLE</h1></u>

</head>

<body>

<table bgcolor="white" cellspacing="5" cellpadding="10" border="2">

<tr>

<th>SNO.</th>

<th>NAME</th>

<th>BRANCH</th>

<th>SECTION</th>

<th>REG.NO.</th>

<th>YEAR</th>

</tr>

<tr>

<td>1</td>

<td>Rina</td>

<td>IT</td>

<td>c</td>

<td>22b01a12g2</td>

<td align="center" rowspan="5">II</td>

</Tr>

<tr>

<td>2</td>

<td>Hasmitha</td>

<td>ECE</td>

<td>B</td>

<td>22b01a12e6</td>

</Tr>

<tr>

<td>3</td>

<td>Tulasi</td>

<td>Mechanical</td>

<td>A</td>

<td>22b01a12e7</td>

</Tr>

<tr>

<td>4</td>

<td>Pranavi</td>

<td>Civil</td>

<td>A</td>

<td>22b01a12e7</td>

</Tr>

<tr>

<td>5</td>

<td>Bhavana</td>

<td>EEE</td>

<td>A</td>

<td>22b01a12eh4</td>

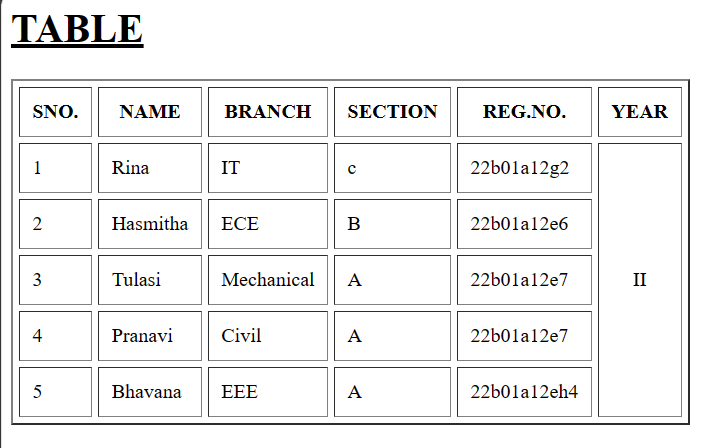
</Tr>

</table>

</body>

</html>

Output:



**Experiment 3 : Develop static web page having different partitions using iframes.**

<html >

<head>

<u><h1 align="center">iframe</h1></u>

</head>

<body>

<div>

<iframe src="exp1.html" height="500" width="500"></iframe>

&nbsp;

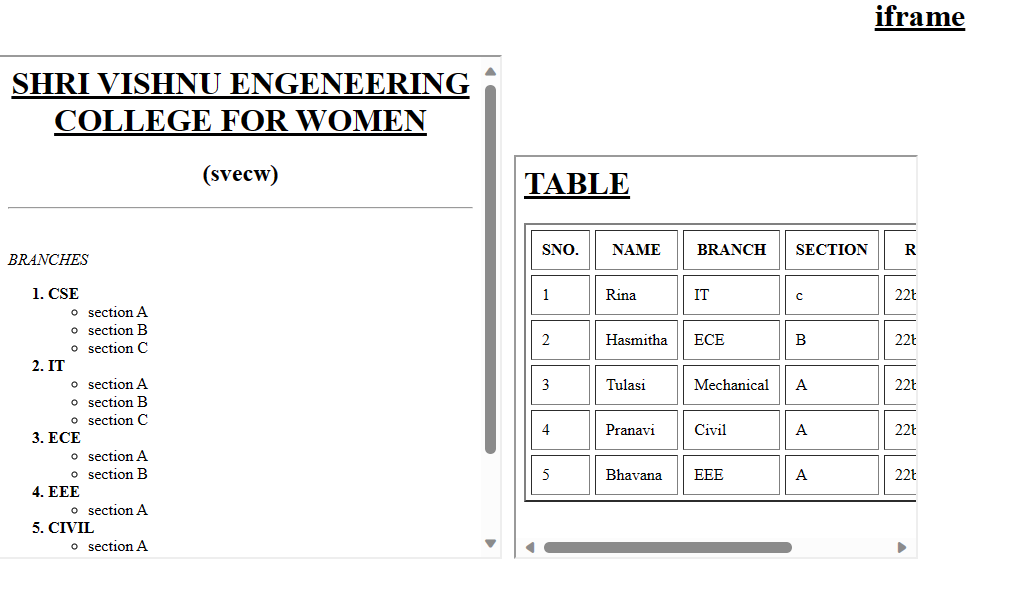
<iframe src="exp2.html" height=400 width="400"></iframe>

</div>

</body>

</html>

Output:



**Experiment 4: Develop a web page to demonstrate CSS properties.**

<html>

<head >

<u><h1>

<marquee direction="right" scrollamount="10" bgcolor="yellow">CSS PROPERTY</marquee>

</h1></u>

<style>

p {

color:purple;

font-size:20;

}

h1 {

color:red;

}

</style>

</head>

<body bgcolor="lightblue">

<p>A book is a medium for recording information in the form of writing or images, typically composed of many pages (made of papyrus, parchment, vellum, or paper) bound together and protected by a cover.[1] It can also be

handwritten or printed work of fiction or nonfiction, usually on sheets

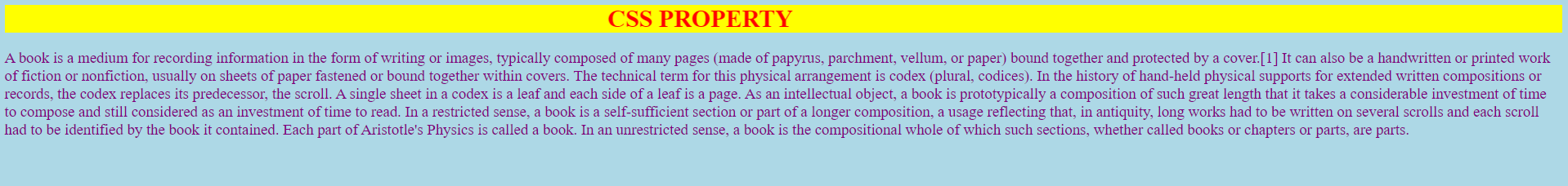
of paper fastened or bound together within covers. The technical term for this physical arrangement is codex (plural, codices). In the history of hand-held physical supports for extended written compositions or records, the codex replaces its predecessor, the scroll. A single sheet in a codex is a leaf and each side of a leaf is a page.

As an intellectual object, a book is prototypically a composition of such great length that it takes a considerable investment of time to compose and still considered as an investment of time to read. In a restricted sense, a book is a self-sufficient section or part of a longer composition, a usage reflecting that, in antiquity, long works had to be written on several scrolls and each scroll had to be identified by the book it contained. Each part of Aristotle's Physics is called a book. In an unrestricted sense, a book is the compositional whole of which such sections, whether called books or chapters or parts, are parts.<p>

</body>

</html>

Output:



**Experiment 5:Demonstrate how pseudo classes can be used to create an effective web page.**

<!DOCTYPE html>

<html>

<head>

<h1>Pseudo Classes</h1>

<style>

a:link {

color: blue;

}

a:visited {

color: pink;

}

a:hover {

color: purple;

}

a:active {

color: orange;

}

</style>

</head>

<body>

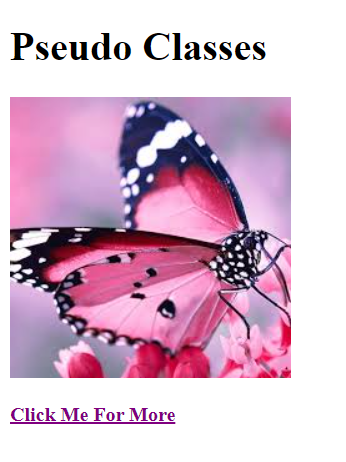
<img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQCoNGlonE8o3AkbW11Ky-BaE6wr6UL2MaoRA&usqp=CAU"/>

<p><b><a href="https://www.pinterest.com.au/juliejoy63/butterfly-pictures/" >Click Me For More</a></b></p>

</body>

</html>

Output:



**Experiment 6: Demonstrate the use of flex box in CSS.**

<html>

<head>

<style>

.flex-box {

display:flex;

float:left;

}

.flex-box {

display: flex;

flex-wrap: wrap;

background-color:purple;

}

.flex-box {

display:flex;

flex-direction:column;

}

.flex-box {

display:flex;

flex-direction:row;

}

.flex-box > div {

background-color:powderblue;

width: 100px;

margin: 10px;

text-align: center;

line-height: 75px;

font-size: 30px;

}

</style>

<div class="flex-box">

<div>1</div>

<div>2</div>

<div>3</div>

</div>

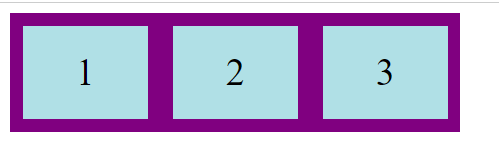
</head>

<body>

</body>

</html>

Output:



**Experiment 7: Demonstrate uses of arrays and strings in javascript.**

<!DOCTYPE html>

<html>

<body>

<h1>JavaScript Arrays</h1>

<p id="demo"></p>

<script>

const cars = [];

cars[0]= "Saab";

cars[1]= "Volvo";

cars[2]= "BMW";

document.getElementById("demo").innerHTML = cars;

</script>

<h1>Some functions in Strings</h1>

<label>The length of the string is:</label>

<p id='len'></p>

<label>The string in Lower case</label>

<p id='low'></p>

<label>Concatenated String</label>

<p id='con'></p>

<label>Sliced string</label>

<p id='slice'></p>

<label>Replaced String</label>

<p id='rep'></p>

<script>

let str="Sayyed";

let str1="Rina";

let str2=str1.concat(" ",str);

document.getElementById("len").innerHTML=str.length;

document.getElementById("low").innerHTML=str.toLowerCase();

document.getElementById("con").innerHTML=str2;

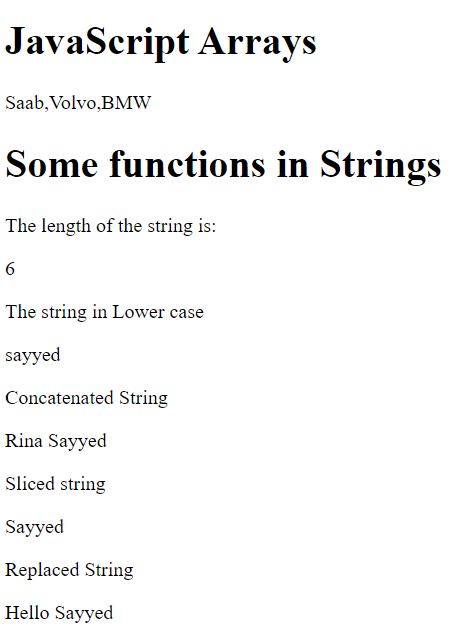
document.getElementById("slice").innerHTML=str.slice(0,7);

document.getElementById("rep").innerHTML=str2.replace("Rina","Hello");

</script>

</body></html>

Output:



**Experiment 8: Write a javascript program to demonstrate functions.**

<html>

<body>

<form>

<label>num1</label><input type="text" placeholder="enter a num" id="num1"></input>

<br/><br/>

<label>num2</label><input type="text" placeholder="enter a num" id="num2"></input>

<br/>

<br/>

<label>Result:</label><input type="text" disabled id="r"/>

<br/>

<br/>

<input type="button" onclick="add()" value="add"/>

<input type="button" onclick="sub()" value="sub"/>

<input type="button" onclick="mul()" value="mul"/>

<input type="button" onclick="div()" value="div"/>

</form>

<script>

function add() {

let a,b;

a=parseInt(document.getElementById("num1").value);

b=parseInt(document.getElementById("num2").value);

document.getElementById("r").value=(a+b);

}

function sub() {

let a,b;

a=parseInt(document.getElementById("num1").value);

b=parseInt(document.getElementById("num2").value);

document.getElementById("r").value=(a-b);

}

function mul() {

let a,b;

a=parseInt(document.getElementById("num1").value);

b=parseInt(document.getElementById("num2").value);

document.getElementById("r").value=(a\*b);

}

function div() {

let a,b;

a=parseInt(document.getElementById("num1").value);

b=parseInt(document.getElementById("num2").value);

document.getElementById("r").value=(a/b);

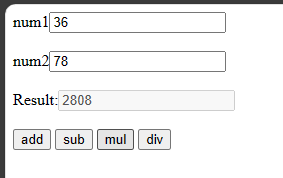
}

</script>

</body>

</html>

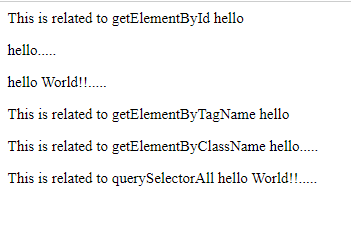
Output:



**Experiment 9: Write a javascript to demonstrate DOM elements,events.**

<html>  
<body>  
<p id="a">hello</p>  
<p class="c">hello.....</p>  
<p class="c">hello World!!.....</p>  
<p id="demo"></p>  
<p id="h"></p>  
<p id="s"></p>  
</body>  
<script>  
let element=document.getElementById("a");  
let b=document.getElementsByTagName("p");  
let d=document.getElementsByClassName("c");  
let q=document.querySelectorAll("p.c");  
document.getElementById("demo").innerHTML='This is related to getElementByTagName '+b[0].innerHTML;  
document.getElementById("a").innerHTML='This is related to getElementById '+element.innerHTML;  
document.getElementById("h").innerHTML='This is related to getElementByClassName '+d[0].innerHTML;  
document.getElementById("s").innerHTML='This is related to querySelectorAll '+q[1].innerHTML;  
</script>  
</html>

Output:



**Experiment 10: Create a “registration form” with the following fields.**

* **Name(text field)**
* **Passsword(password field)**
* **E-mail id(text field)**
* **Phone number(text field)**
* **Gender(radio button)**
* **Languages known(checkboxes-English,Telugu,Hindi,Tamil)**
* **Address(tesxt area)**

<html>

<head>

<h1>REGISTRATION FORM</h1>

</head>

<body>

<form>

<table>

<tr><td>

<b>Name:</b>

</td>

<td><input type="text"></td>

</tr>

<tr><td>

<b>Password:</b>

</td>

<td><input type="password"></td>

</tr>

<tr><td>

<b>E-mail id:</b>

</td>

<td><input type="text"></td>

</tr>

<tr><td>

<b>Phone number:</b>

</td>

<td><input type="number"></td>

</tr>

<tr><td>

<b>Gender:</b>

</td>

<td><input type="radio" value="Male" id="a" name="a">Male</td>

<td colspan="3"><input type="radio" value="Female" id="a" name="a">Female</td>

</tr>

<tr><td>

<b>Languagues known:</b>

</td>

<td><input type="Checkbox">Telugu</td>

<td><input type="Checkbox">English</td>

<td><input type="Checkbox">Hindi</td>

<td><input type="Checkbox">Tamil</td>

</tr>

<tr><td>

<b>Address:</b>

</td>

<td><input type="textarea"></td>

</tr>

<tr>

<td colspan="2"><input type="Submit"/></td>

</tr>

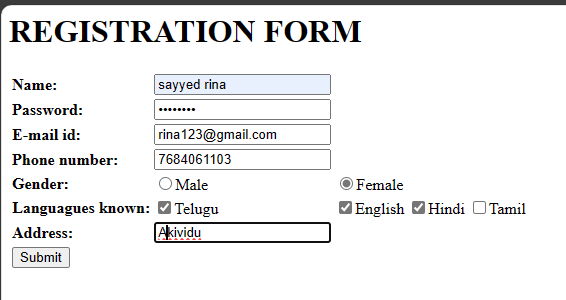
</table>

</form>

</body>

</html>

Output:



**Experiment 11:Write javascript to validate the following fields of the above registration page.**

<html>

<head>

<script>

function Registrationform(){

var name=document.myform.name.value;

var password=document.myform.password.value;

var email=document.myform.email.value;

var phone number=document.myform.phonenumber.value;

if (name==null || name==""){

alert("Name should contain alphabets and length should not be less than 6 characters");

return false;

}else if(password.length<6){

alert("Password should not be less than 6 characters length);

return false;

}

else if (email == "" || !regEmail.test(email)) {

alert("Should not contain any invalid and must follow the standard pattern name@domain.com");

return false;

}

else (phone == "" || !regPhone.test(phone)) {

alert("phone number should contain 10 digits only");

return false;

}

}

</script>

</head>

<body>

<fieldset>

<legend>REGESTRATION FORM</legend>

<form name="myform" method="post" action="exp11.html" onsubmit="return Registrationform()" >

<table>

<tr>

<td><label>NAME</label></td>

<td><input type="text" name="name" placeholder="name" size="30" width="30"></td>

</tr>

<td><label>PASSWORD</label></td>

<td><input type="password" name="password" placeholder="password" size="30" width="30"></td>

</tr>

<tr>

<td><label>EMAIL</label></td>

<td><input type="text" name="email id" placeholder="email id" size="30" width="30"></td>

</tr>

<tr>

<td><label>PHONE NO.</label></td>

<td><input type="phone number" name="phone number" placeholder="phn no." size="30" width="30"></td>

</tr>

</tr>

<td><label>GENDER</label></td>

<td><input type="radio" name="gender" value= "male" >MALE

<input type="radio" name="gender" value= "female" >FEMALE

</td></tr>

<tr>

<td><label>LANGUAGES KNOWN</label></td>

<td><input type="checkbox" name="english" value="english" >ENGLISH

<input type="checkbox" name="telugu" value="telugu" >TELUGU

<input type="checkbox" name="hindi" value="hindi" >HINDI

<input type="checkbox" name="tamil" value="tamil" >TAMIL

</td></tr>

<tr>

<td><label>ADDERSS</label></td>

<td><textarea rows="5" cols="30"></textarea></td>

</tr>

<tr>

<td colspan="2" align = "center"><button><input type="reset"></button></td>

</tr></table>

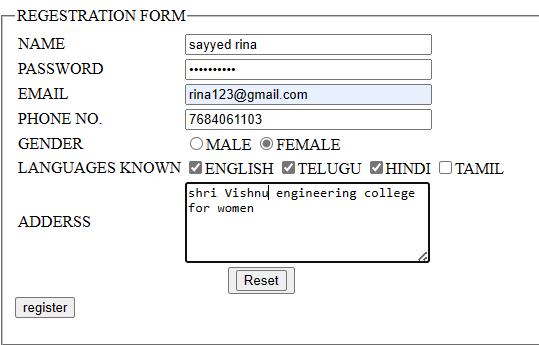
<input type="submit" value="register">

</form>

</fieldset>

</html>

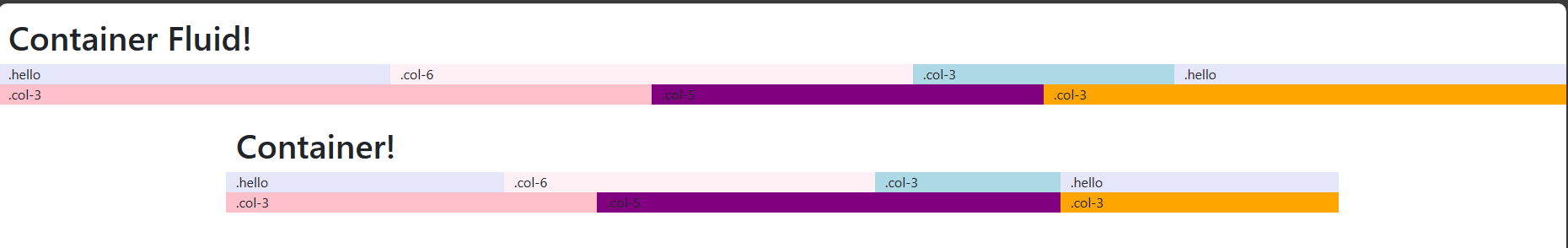
Output:



**Experiment 12:Demonstrate Grid system in bootstrap.**

<!DOCTYPE html>  
<html lang="en">  
<head>  
  <title>Bootstrap Example</title>  
  <meta charset="utf-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1">  
  <link href="<https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css>" rel="stylesheet">  
  <script src="<https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js>"></script>  
</head>  
<body>  
<div class="container-fluid mt-3">  
  <h1>container fluid!</h1>  
  <div class="row">  
    <div class="col-3 " style="background-color:lavender;">.hello</div>  
    <div class="col-4" style="background-color:lavenderblush;">.col-6</div>  
    <div class="col-2" style="background-color:lightblue;">.col-3</div>  
    <div class="col-3 " style="background-color:lavender;">.hello</div>  
  </div>  
   <div class="row">  
    <div class="col-5" style="background-color:pink;">.col-3</div>  
    <div class="col-3 " style="background-color:purple;">.col-5</div>  
    <div class="col-4 " style="background-color:orange;">.col-3</div>  
  </div></div>  
&nbsp;  
<div class="container">  
   <h1>container!</h1>  
  <div class="row">  
    <div class="col-3 " style="background-color:lavender;">.hello</div>  
    <div class="col-4" style="background-color:lavenderblush;">.col-6</div>  
    <div class="col-2" style="background-color:lightblue;">.col-3</div>  
    <div class="col-3 " style="background-color:lavender;">.hello</div>  
  </div>  
   <div class="row">  
    <div class="col-4" style="background-color:pink;">.col-3</div>  
    <div class="col-5 " style="background-color:purple;">.col-5</div>  
    <div class="col-3 " style="background-color:orange;">.col-3</div>  
  </div></div>  
</body>  
</html>

Output:



**Experiment 13:Apply different button styles in Bootstrap.**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Bootstrap Example</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>

</head>

<body>

<div class="container mt-3">

<h2>Button Styles</h2>

<button type="button" class="btn">Basic</button>

<button type="button" class="btn btn-primary">Primary</button>

<button type="button" class="btn btn-secondary">Secondary</button>

<button type="button" class="btn btn-success">Success</button>

<button type="button" class="btn btn-info">Info</button>

<button type="button" class="btn btn-warning">Warning</button>

<button type="button" class="btn btn-danger">Danger</button>

<button type="button" class="btn btn-dark">Dark</button>

<button type="button" class="btn btn-light">Light</button>

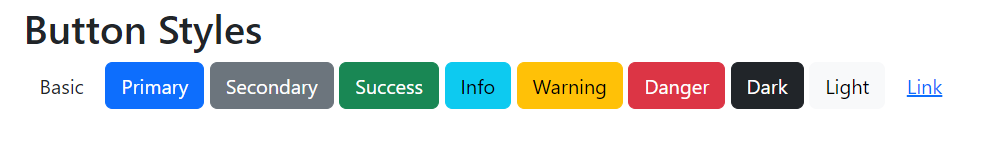
<button type="button" class="btn btn-link">Link</button>

</div>

</body>

</html>

Output:



**Experiment 14: Demonstrate Bootstrap Cards.**

<!DOCTYPE html>

<head>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>

</head>

<body>

<div class="container">

<div class="card" style="width:300px">

<img class="card-img-top" src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcS9SE2\_BrnCOcoU5GdVizvWuogejS-d96FNSQ&usqp=CAU"style="width:100%">

<div class="card-body">

<h4 class="card-title">Profile...!</h4>

<p class="card-text">click to view more details...</p>

<a href="#" class="btn btn-primary">See More</a>

</div> </div>

<br>

<div class="card" style="width:300px">

<div class="card-body">

<h4 class="card-title">Profile...!</h4>

<p class="card-text">click to view more details...</p>

<a href="#" class="btn btn-primary">See More</a> </div>

<img class="card-img-bottom" src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcS9SE2\_BrnCOcoU5GdVizvWuogejS-d96FNSQ&usqp=CAU" style="width:100%">

</div></div>

<br>

<div class="card img-fluid" style="width:400px">

<img class="card-img-top" src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcS9SE2\_BrnCOcoU5GdVizvWuogejS-d96FNSQ&usqp=CAU"style="width:100%">

<div class="card-img-overlay">

<h4 class="card-title">Profile..!</h4>

<p class="card-text">click to view more details...</p>

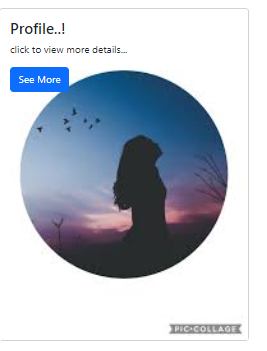
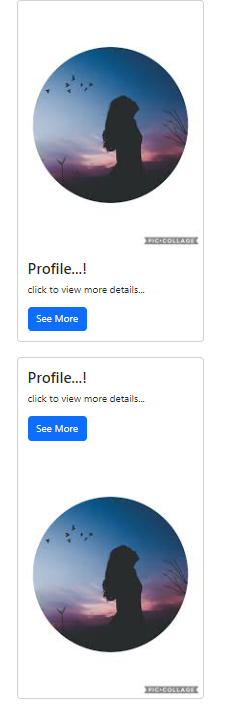
<a href="#" class="btn btn-primary">See More</a>

</div> </div>

<br>

</body></html>

Output:



**Experiment 15:Write a program to create a tabbed navigation menu in Bootstrap.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Tabbed Navigation Menu</title>

<!-- Add Bootstrap CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

</head>

<body>

<div class="container">

<ul class="nav nav-tabs" id="myTab" role="tablist">

<li class="nav-item">

<a class="nav-link active" id="home-tab" data-toggle="tab" href="#home" role="tab" aria-controls="home" aria-selected="true">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" id="profile-tab" data-toggle="tab" href="#profile" role="tab" aria-controls="profile" aria-selected="false">Profile</a>

</li>

<li class="nav-item">

<a class="nav-link" id="contact-tab" data-toggle="tab" href="#contact" role="tab" aria-controls="contact" aria-selected="false">Contact</a>

</li> </ul>

<div class="tab-content" id="myTabContent">

<div class="tab-pane fade show active" id="home" role="tabpanel" aria-labelledby="home-tab">

<!-- Content for the Home tab goes here -->

<h1>Home</h1>

<p>This is home tab!!..</p>

</div>

<div class="tab-pane fade" id="profile" role="tabpanel" aria-labelledby="profile-tab">

<!-- Content for the Profile tab goes here -->

<h1>Profile</h1>

<p>I'm pranavi of 2nd btech</p>

</div>

<div class="tab-pane fade" id="contact" role="tabpanel" aria-labelledby="contact-tab">

<!-- Content for the Contact tab goes here -->

<h1>Contact</h1>

<p>Contact me here no-9568475621.</p>

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

<!-- Add Bootstrap JS and jQuery -->

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

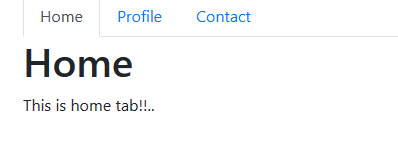
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.3/dist/umd/popper.min.js"></script>

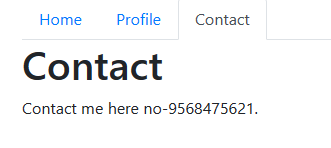
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

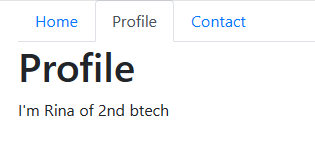
</body>

</html>

Output:

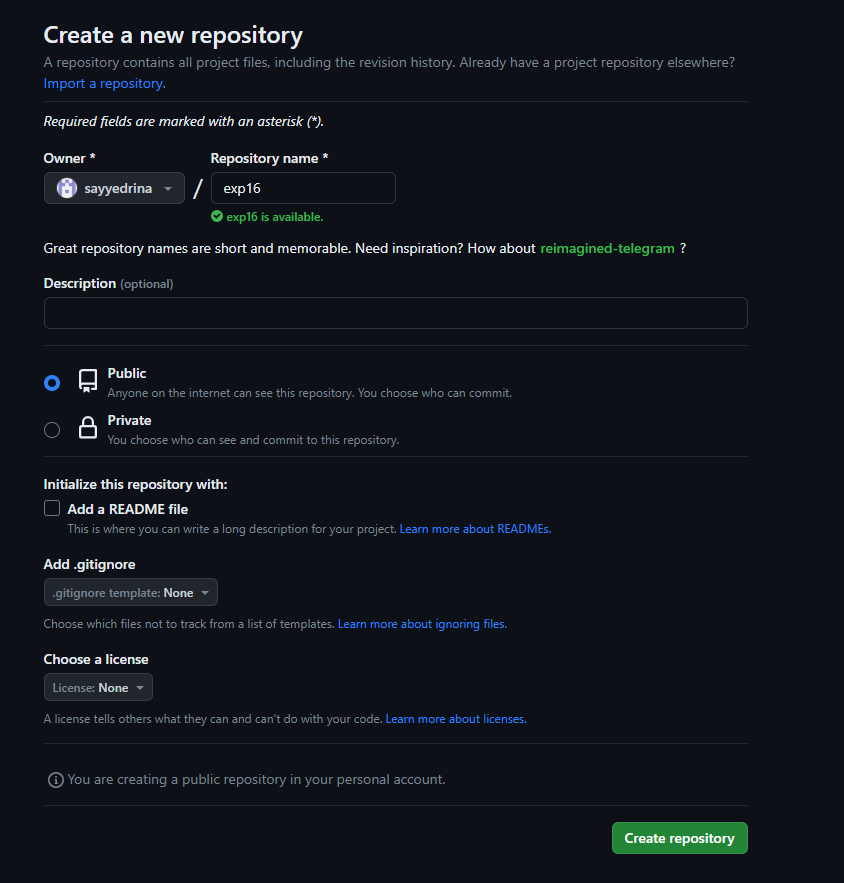






**Experiment 16: Demonstrate the following with respect to GIT/GitHub.**

* **Install Git and create Git Repository.**
* **Git Configuration (User Name and Email).**
* **Copy the Repository from GitHub to the Local Computer(git clone).**
* **Publishing Changes back to GitHub(push).**
* **Adding and removing from a Git repository.**
* **Creating pull requests.**
* **Branching repositories**
* **Deploy a web page to GitHub pages.**
* To install the Git we need to go to browser and download the official Brochure and download it, and run the file.
* Set the path and setup Git and GitBash.
* After installing the Git then Create GitHub account .
* Follow the link ”https://github.com” and create the account in GitHub and click on "create new repository" ,set private or public ,but when we work in real time ,we keep it private and name the repository name.
* Click on "Create repository".



* ……………………. Go to GitBash and pose commands change the directory to your local directory in your computer and give commands as "git config --global user.name "SREESAMHITA"" and then give email commandas "git config --global user.email[rina072009@gmail.com](mailto:rina072009@gmail.com)and after give command as "git config --list",and so that we get user.name and user.email.
* …………………………. Copy the link of the repository and go to Git bash and give command as "git clone (repository link)" , now your GitHub account is linked with GitBash.Change the directory to GitHub filename. Give command as "git add (content)", and finally give commit command that is "git commit -m "commit".
* As now we need to push changes into GitHub, by giving command as "git push origin main".Now if we check file in GitHub account the changes will be changed.
* To create pull request,open your file and click on edit symbol and then click on pull request and commit changes.now pull request is activated in this way we create discussion while updating a code and then close the request.
* To deploy webpages to GitHub pages ,first we need to click on already existing and then upload all files into it.Go to settings and click on pages select branch as main and /root ,now a link is created for the repository.
* Now click that we can deploy webpage into GitHub pages……………..

**Experiment 17: Demonstrate how to create designs/prototype using Figma.**

In Figma we mainly have 3 stages

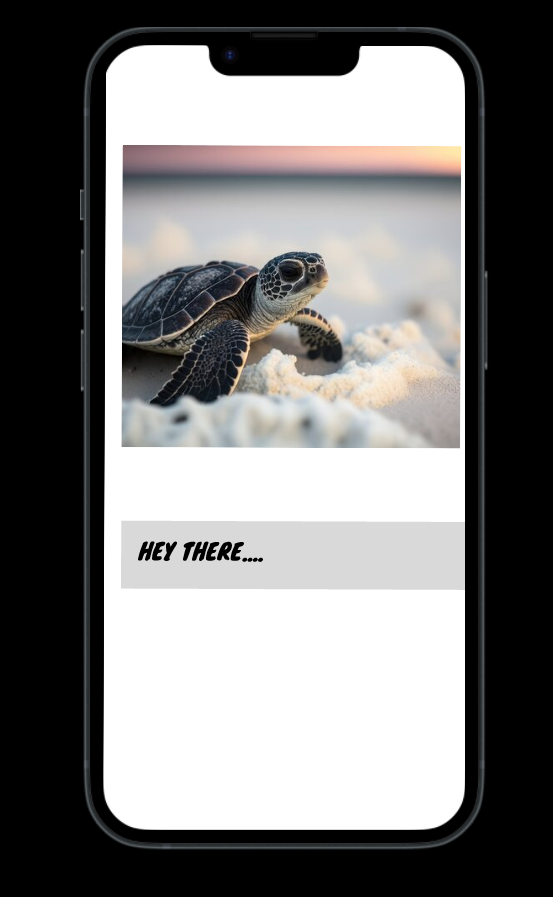
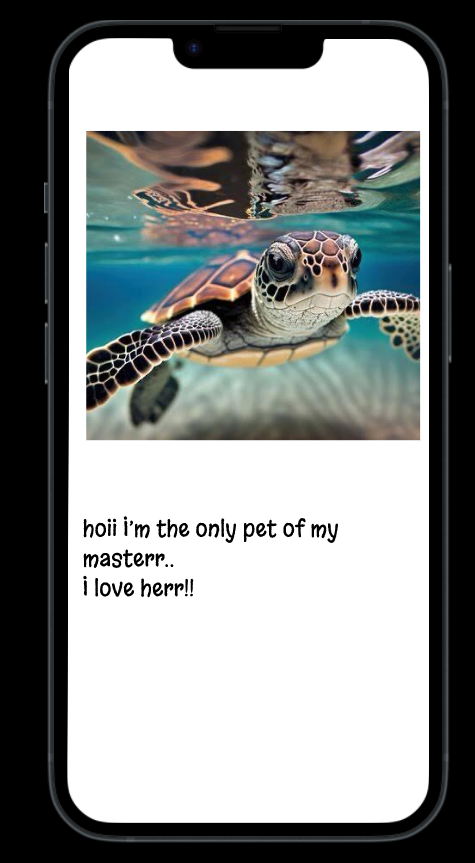
1.Design

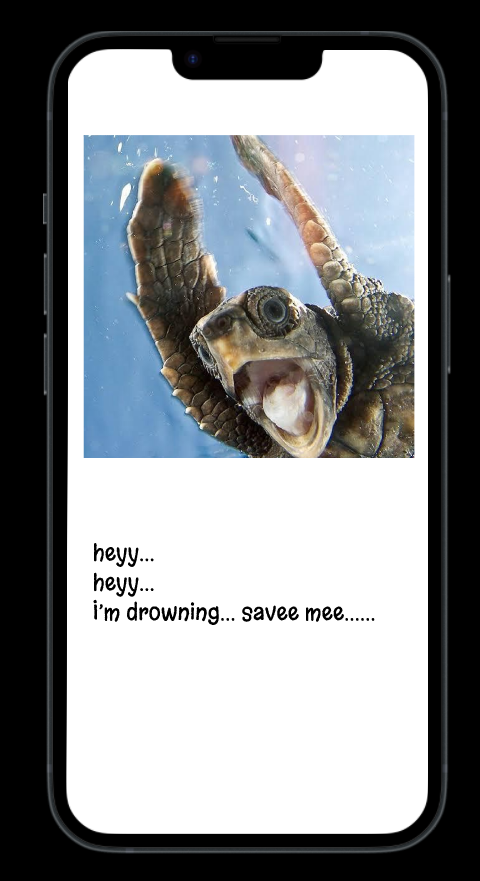
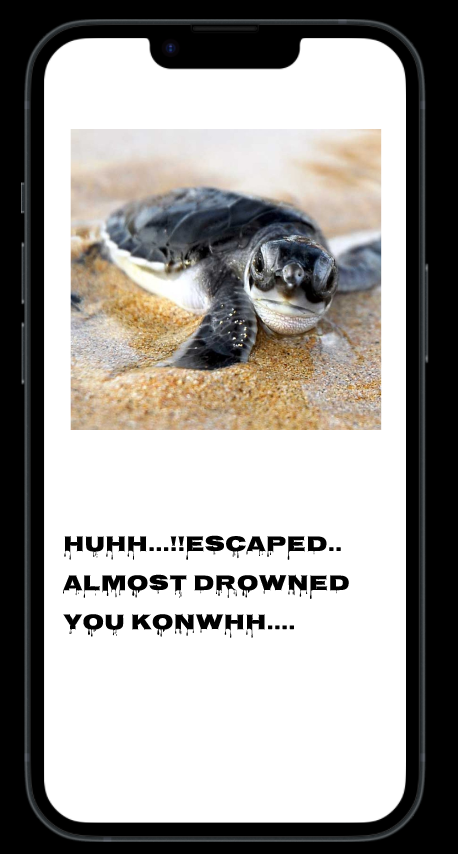
2.Prototype

3.Inspect

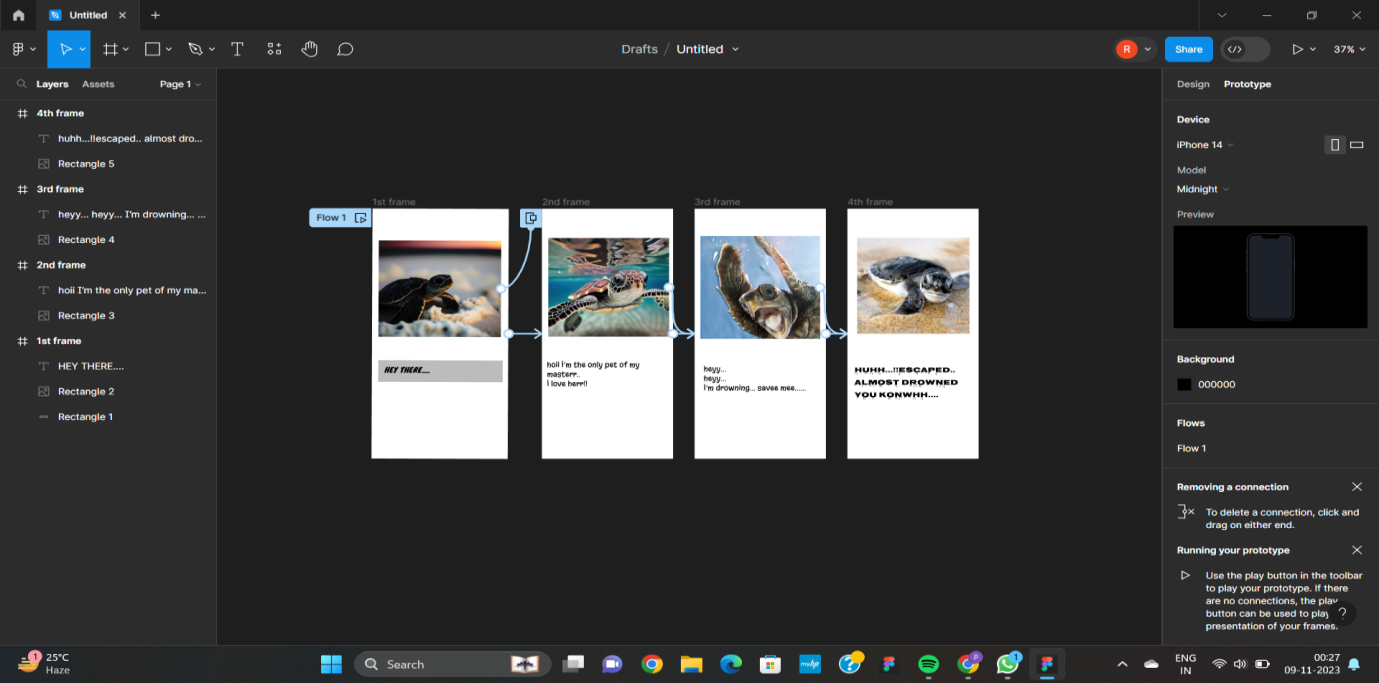
First, we need to download Figma app and login by giving credentials.

* Now, we need to insert a frame and select the type of frame we want to insert.
* After selecting the frame go to place a rectangle and click add photo/video so we select picture from the system and insert it there.
* Click T on the keyboard to add a text box and write some text we can even change font size , color in our desired manner.
* Now create 4 frames of your way.

* Until now, we completed design
* Now it’s time for link frames.
* We click prototype on the top right
* When we click on connection and drag from one frame to another, we have the option of how we can navigate like on click and type of animation like move in, move out.



* And finally click on inspect thus we can find the prototype.