



ANNAMACHARYA UNIVERSITY

Estd. under Andhra Pradesh

Private Universities (Establishment and Regulation) Act, 2016

(University listed in UGC as per section 2(f) of the UGC Act, 1956)

LECTURE NOTES

ON

Full Stack Development (24ACSE34L)

B.TECH II Year - I Sem (2026-27)

DEPARTMENT OF AIML

EXPERIMENT—1

1A) LIST AND TYPES OF LISTS

```
<html>
<head>
<title>HTML LISTS<title>    //Bug1
</head>
<body>
<h1>UN ORDERED LIST</h1>
<ul>
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato    //Bug2
<li>Radish</li>
</ul>
ORDERED LIST</h1>//Bug3
<ol>
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
<ol>
<h1>HTML Definition List<h1>//Bug4
<dl>
<dt><b>HTML</b>//bug4
<dd>This stands for Hyper Text Markup Language</dd>
<dt><b>HTTP</b></dt>
This stands for Hyper Text Transfer Protocol</dd>//bug5
</dl>
</body>
</html>
```

ERRORS:

1. Unclosed <title> tag

Description: This causes browsers to misinterpret the page title.

- **Solution:**
- `<title>HTML Lists</title>`

2. Missing closing for Potato

- **Description:** "Potato" isn't closed properly; browsers may auto-correct but the structure is invalid.
- **Solution:**

```
<li>Potato</li>
<li>Radish</li>
```

3. Mismatched heading for Ordered List

- **Bug:**
- **Description:** No opening <h1> tag before "ORDERED LIST".

Solution:

```
<h1>Ordered List</h1>
```

4. Unclosed Ordered List ()

- **Bug:**
- **Description:** The closing tag should be , not .

Solution:

```
</ol>
```

5. Unclosed <h1> in Definition List

- **Bug:**

```
<h1>HTML Definition List<h1>
```

- **Description:** The second <h1> should be a closing tag.

Solution:

```
<h1>HTML Definition List</h1>
```

6. Improper <dt> and <dd> structure

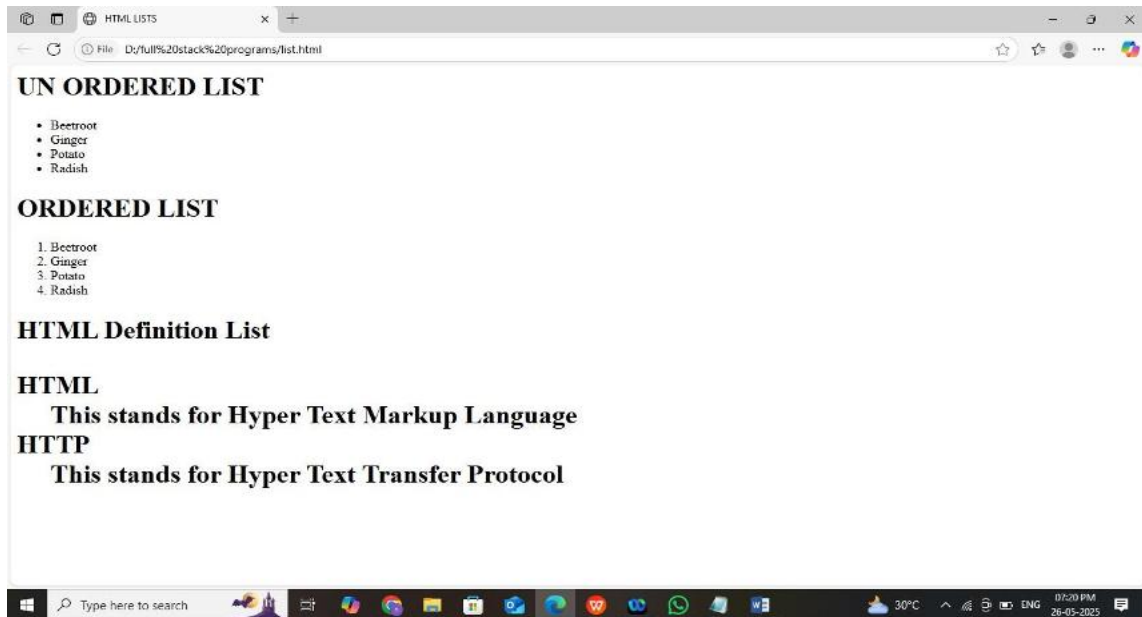
- **Bug:**

```
<dt><b>HTML</b>
<dd>This stands for Hyper Text Markup Language</dd>
<dt><b>HTTP</b></dt>
This stands for Hyper Text Transfer Protocol</dd>
```

- **Descriptions:**

- <dt> for "HTML" is not closed.
- "HTTP" has description outside of <dd>.
- Last description has extra </dd>.

OUTPUT:



1b)HYPERLINK

```
<html>
```

```
<title>Hyperlink Example</title> //Bug1
```

```
</head>
```

```
<p>Click following link</p> //Bug2
```

```
<a href="http://www.tutorialspoint.com" target="_self">full stack tutorials point<a> //Bug3
```

```
<body> //Bug 4
```

```
</html>
```

ERRORS:

1. Missing <head> opening tag

- Bug:**

```
<title>Hyperlink Example</title>  
</head>
```

You closed </head> but never opened <head>.

- Fix:**

```
<head>  
  <title>Hyperlink Example</title>  
</head>
```

2. Paragraph <p> outside <body>

- **Bug:**

<p>Click following link</p>

appears before <body>.

- **Fix:** Move it **inside** <body>.

3. Unclosed <a> tag

- **Bug:**

full stack tutorials point<a>

Last <a> should be .

- **Fix:**

Full Stack Tutorials Point

4. Improper HTML structure

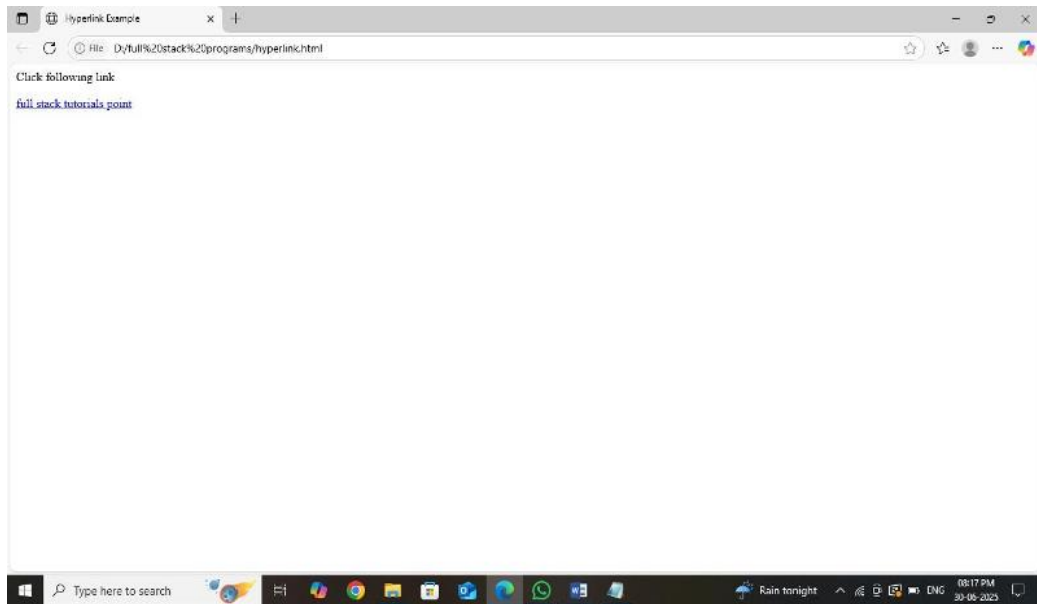
- **Bug:**

The <body> is declared **after the** <a> tag.

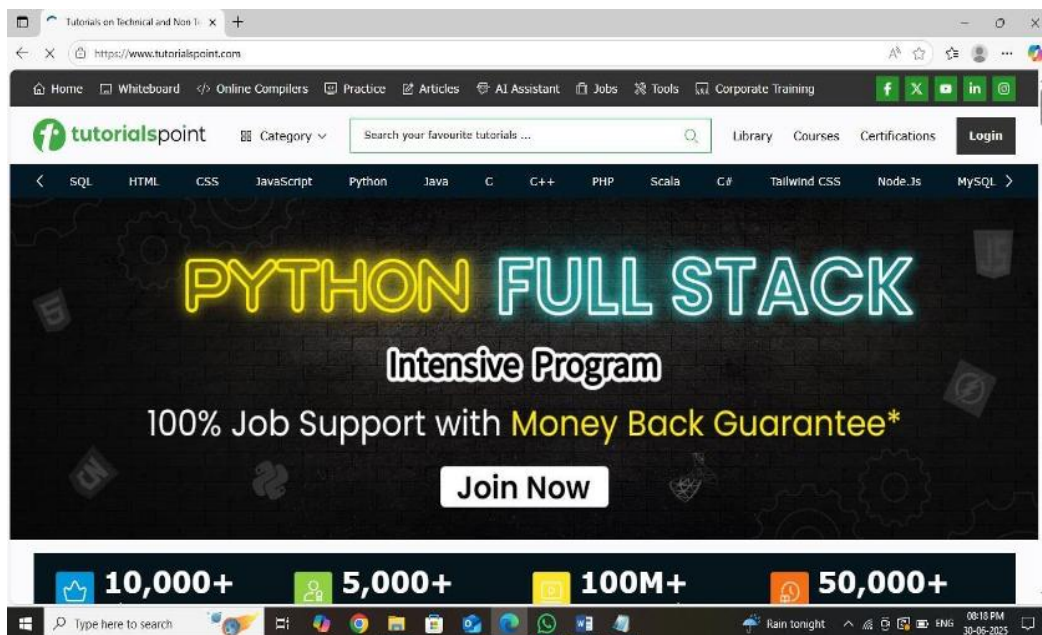
- **Fix:** Always structure HTML as:

```
<html>
  <head>...</head>
  <body>...</body>
</html>
```

OUTPUT:



After Clicking The Link This Page Will Be Open:



1C)YOUR IMAGE AND YOUR FRIEND IMAGE:

```
<html>
```

```
<head>
```

```
<title>HTML IMAGES</title>
```

```
<body> //Bug1
```

```
<h1>MY IMAGE</h1>
```

```
//bug3 //bug4
```

```
<h1>MY FRIEND IMAGE<h1> //bug2
```

```
<img href="C:\Users\WELCOME\Pictures\Saved Pictures\same.jpeg" height="500" width="500"/  
///bug5  
<body>  
</html>
```

Errors:

1. Missing </head> closing tag

You opened <head> but never closed it.

- **Fix:**

```
<head>  
  <title>HTML IMAGES</title>  
</head>
```

2. Improper closing of <h1>

- **Bug:**

```
<h1>MY FRIEND IMAGE<h1>
```

The second <h1> is wrongly opened instead of closed.

- **Fix:**

```
<h1>MY FRIEND IMAGE</h1>
```

3. Wrong attribute for image

- **Bug:**

```
<img href="C:\Users\WELCOME\Pictures\Saved Pictures\same.jpeg" ...>
```

Images use src, **not** href.

- **Fix:**

```

```

4. Unclosed tag

- **Bug:**

<img ... /

Missing the final >.

- **Fix:**

5. Improper <body> closure

- **Bug:**

<body>

...

<body>

</html>

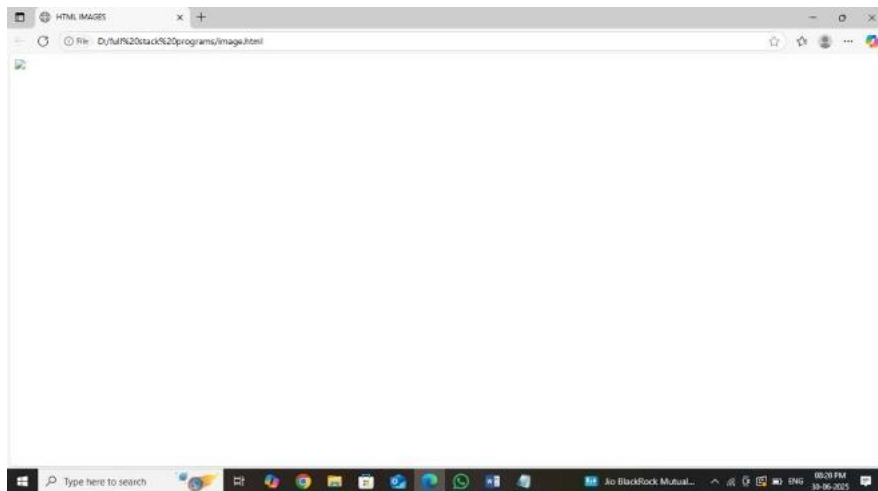
Second <body> is extra and missing </body>.

- **Fix:**

</body>

</html>

OUTPUT:



Here The Images Will Be Displayed

NOTE: (C:\Users\WELCOME\Pictures\Saved Pictures\same.jpeg) IMAGE MUST BE SAVED IN YOUR LOCAL DEVICE

1D)THUMBNAIL IMAGE

<html>

<head>

```
<style>
```

```
img {
```

```
  border: 1px solid #ddd;
```

```
  border-radius: 4px;
```

```
  padding: 5px;
```

```
  width: 150px;
```

```
} //bug1
```

```
</head>
```

```
<h2>Thumbnail Image</h2>//bug 5
```

```
<p>Click on the image to enlarge it.</p>
```

```
  //bug2
```

```
  
```

```
</a>
```

```
<a target="_blank" href="C:\Users\WELCOME\Pictures\Saved Pictures\same.jpeg">
```

```
  
```

```
</a>
```

```
<a target="_blank" href="C:\Users\WELCOME\Pictures\Saved Pictures\nature2.jpeg">
```

```
  
```

```
</a>
```

```
<a target="_blank" href="C:\Users\WELCOME\Pictures\Saved Pictures\nature3.jpeg">
```

```
  
```

```
</a>
```

```
<a target="_blank" href="C:\Users\WELCOME\Pictures\Saved Pictures\nature4.jpeg">
```

```
  
```

```
</a>
```

```
<a target="_blank" src="C:\Users\WELCOME\Pictures\Saved Pictures\nature5.jpeg"> //bug4
```

```
  <img href="C:\Users\WELCOME\Pictures\Saved Pictures\nature5.jpeg"> //bug3
```

```
</a>
```

```
</body>
```

```
</html>
```

Errors:

1. Unclosed <style> tag

2. Invalid src="_blank" inside

- **Bug:**

```

```

src="_blank" is not valid. _blank is used in .

- **Fix:**

Remove it:

```

```

3. Wrong attributes for

- **Bug:**

```
<img href="...">
```

 uses src, **not** href.

- **Fix:**

```

```

4. Wrong <a> attribute

- **Bug:**

```
<a target="_blank" src="...">
```

<a> uses href, **not** src.

- **Fix:**

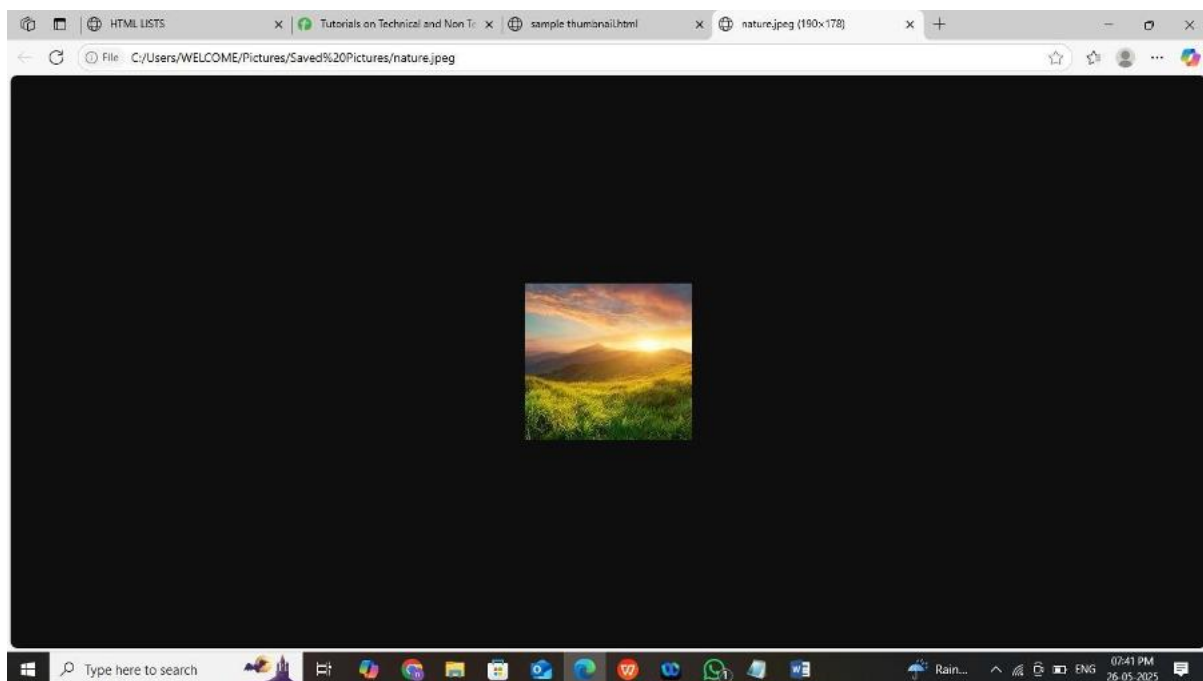
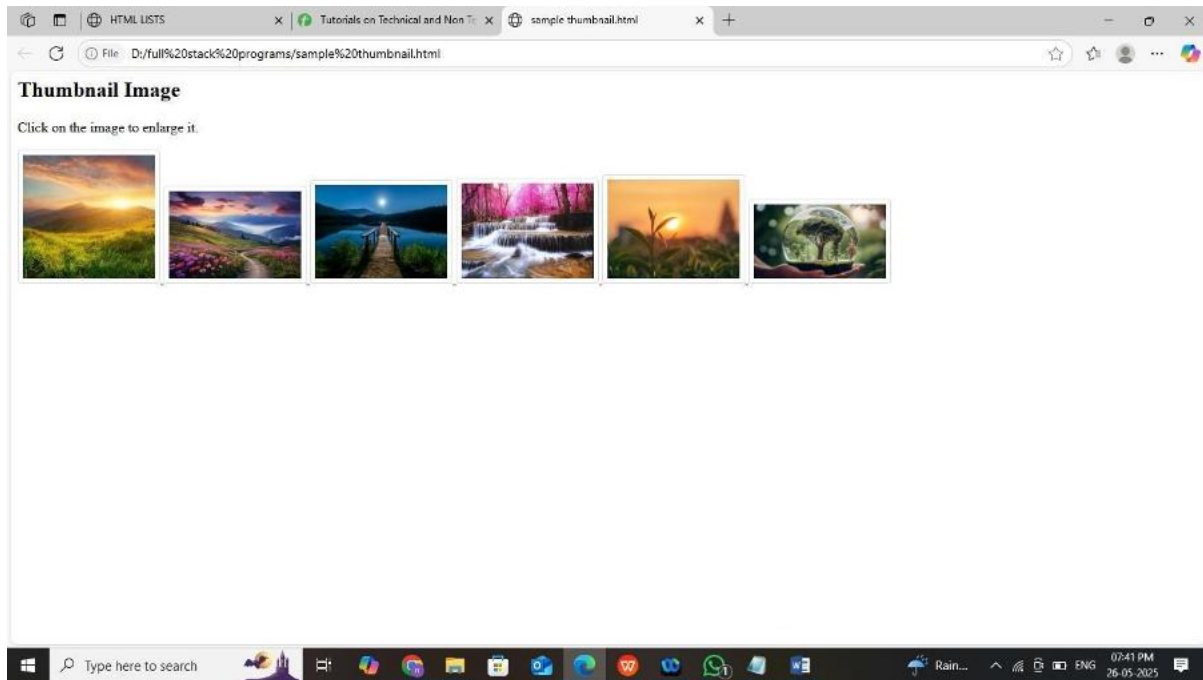
```
<a target="_blank" href="C:\Users\WELCOME\Pictures\Saved  
Pictures\nature5.jpeg">
```

5. Missing <body> opening tag

- **Bug:** You never opened `<body>`.
- **Fix:**

`<body>`

Output:



EXPERIMENT-2

2A)SIMPLE TABLE:

```
<html>

<head>

<title>HTML TABLES<title> //Bug1

<sty> //Bug2

table, th, td {

    border: 1px solid black;

    border-collapse: collapse;

}

</style>

</head>

<body>

<table>

<p>ANNAMACHARYA UNIVERSITY-BRANCHS //Bug3

<tr>

    <th rowspan="2">SI.N</th>

    <th colspan="2">BRANCH</th>

    <th>NUMBER OF STUDENTS</th> //Bug4


<tr>

    <td><b>MAIN BRANCH</b></td>

    <td><b>SUB BRANCH</b></td>

</tr>

    <td>1</td>

    <td>CSE</td>

    <td>CSE-AIDS //Bug5

    <td>2000</td>

</tr>
```

```
<td>2</td>
<td>CSE</td>
<td>CSE-AIML</td>
<td>2000</td>
</tr>
<tr>
<td>3</td>
<td>CSE</td>
<td>CSE(DS)</td>
<td>3000</td>
</tr>
<tr>
<td>4</td>
<td>CSE</td>
<td>CSE(ML)</td>
<td>1000</td>
</tr> //Bug6
</table>
</html>
```

Errors:

1. Incorrect closing of <title>

- **Bug:**

```
<title>HTML TABLES<title>
```

The second <title> should be </title>.

- **Fix:**

```
<title>HTML TABLES</title>
```

2. Wrong <style> tag

- **Bug:**

`<sty>`

should be `<style>`.

Fix:

`<style>`

3. `<p>` inside `<table>`

- **Bug:**
You put a `<p>` before the first `<tr>`. A `<p>` is not valid inside a table.
- Fix: Move `<p>` **above the table**.

4. Unclosed `<tr>`

- **Bug:**
The first `<tr>` isn't closed before the next `<tr>`.
- Fix: Add `</tr>` after the first row.

5. Unclosed `<td>`

- **Bug:**
`<td>CSE-AIDS`

is missing `</td>`.
- Fix:

`<td>CSE-AIDS</td>`

6. Missing `<body>` closing

- **Bug:**
No `</body>` before `</html>`.
- Fix: Add it.

OUTPUT:

ANNAMACHARYA UNIVERSITY:BRANCHS

SL.N	BRANCH		NUMBER OF STUDENTS
	MAIN BRANCH	SUB BRANCH	
1	CSE	CSE-AIDS	2000
2	CSE	CSE-AIML	2000
3	CSE	CSE(DS)	3000
4	CSE	CSE(ML)	1000

2B) TIME TABLE:

```

<html>

<head>

</head>

<body>

<table border="2" width=50% height= 50% //Bug1

<tr>

<th colspan ="7">TIME TABLE

</tr>


<th>TIMINGS</th>

<th> MON</th>

<th> TUE</th>

<th> WED</th>

<th> THUR</th>

<th> FRI</th>

<th> SAT</th>

<tr>

<td><b>9:15-10:15<b></td> //Bug2

<td>Science</td>

```

<td>Maths</td>

<td>Pysics</td> //Bug3

<td>Maths</td>

<td>Arts</td>

<td>Chemistry</td>

<tr>

<td>10:15-11:15</td>

<td>Social</td>

<td>Hindi</td>

<td>English</td>

<td>Social</td>

<td>Maths</td>

<td>Sports</td>

</tr>

<th colspan="7">LUNCH 12:00 to 2:00</th> //Bug4

</tr>

<tr>

<td>2:15-3:15</td>

<td>Science</td>

<td>Maths</td>

<td>Science</td>

<td>Maths</td>

<td>Telugu</td>

<td rowspan="2">Project</td> //bug5

</tr>

<tr>

<td>3:15-4:15<td>

<td>Social</td>

<td>Hindi<td>

<td>English</td>

```
<td>Social</td>
<td>English</td>
</tr>
```

```
<body> //Bug6
</body>
</html>
```

Errors:

1. Missing > after <table>

```
<table border="2" width=50% height= 50%
```

Missing closing >.

Fix:

```
<table border="2" width="50%" height="50%">
```

2. Unclosed tags

```
<td><b>9:15-10:15<b></td>
```

Second should be .

Fix:

```
<td><b>9:15-10:15</b></td>
```

3. Spelling mistakes

- Pysics → Physics

4. Improper <tr> and <th> usage

- Some <tr> tags not properly closed.
- <th colspan="7">LUNCH...</th> should be inside a <tr>.

5. Rowspan/colspan consistency

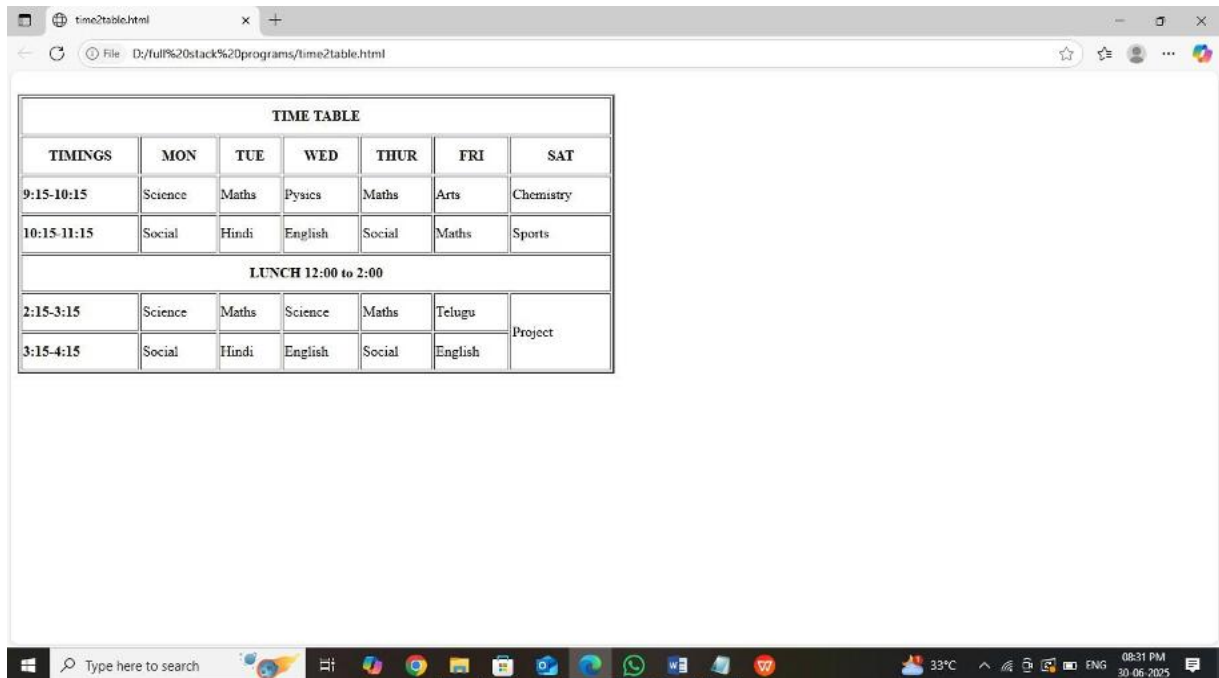
- <td rowspan="2">Project</td> is correct, but some <td> tags are left unclosed (missing </td>).

6. Improper <body> closing

```
<body>
</body>
```

Extra <body> before closing.

OUTPUT:



TIME TABLE						
TIMINGS	MON	TUE	WED	THUR	FRI	SAT
9:15-10:15	Science	Maths	Physics	Maths	Arts	Chemistry
10:15-11:15	Social	Hindi	English	Social	Maths	Sports
LUNCH 12:00 to 2:00						
2:15-3:15	Science	Maths	Science	Maths	Telugu	Project
3:15-4:15	Social	Hindi	English	Social	English	

2C)REGISTRATION FORM:

```
<html>
```

```
  <title>Student Registration Form<title>  //BUG1
```

```
</head>
```

```
<body>
```

```
  <h1>Student Registration Form<h1>  //BUG2
```

```
  <form>
```

```
    <label >Full Name:<label> //BUG3
```

```
    <input type="text" id="name" name="name"><br><br>
```

```
    <label// >Phone Number:</label>
```

```
    <input type="number" id="phone" name="phone"><br><br>
```

```
    <label >Date of Birth:</label> //BUG3
```

```
    <input type="date" id="dob" name="dob"><br><br>
```

```
    <label for="pwd">Password:</label>
```

```
    <input type="password" id="pwd" name="pwd"><br><br>
```

```
<label>Hobbies:</label><br>
<input type=" " id="sports" name="interests" value="Sports">//BUG4
<label>Sports</label><br>
<input type="checkbox" id="music" name="interests" value="Music">
<label>Music</label><br>
<input type="checkbox" id="reading" name="interests" value="Reading">
<label>Reading</label><br><br>

<label>Gender:</label><br>
<input type="radio" id="male" name="gender" value="Male">
<label>Male</label><br>
<input type="radio" id="female" name="gender" value="Female">
<label>Female</label><br><br>

<label>Grade:</label>
<select id="grade" name="grade">
  <option value="1">1st Grade</option>
  <option value="2">2nd Grade</option>
  <option value="3">3rd Grade</option>
  <option value="4">4th Grade</option>
  <option value="5">5th Grade</option>
</select><br><br>

<label>Address:</label><br>
<textarea id="address" name="address" rows="5" cols="50"></textarea><br><br>

<input type="submit" value="Submit">
<input type="r" value="Reset">//BUG6

</form>
</body>
```

</html>

Errors:

1. Title tag not closed properly

<title>Student Registration Form<title>

Wrong → <title> should be closed with </title>.

Fix:

<title>Student Registration Form</title>

2. Heading <h1> not closed properly

<h1>Student Registration Form<h1>

Wrong → Second <h1> should be </h1>.

Fix:

<h1>Student Registration Form</h1>

3. Label tags broker

<label >Full Name:<label>

Fix:<label>Phone Number:</label>

<label >Date of Birth:</label>

Wrong </label> invalid.

Fix:

<label>Date of Birth:</label>

4. Checkbox for sports has empty type

<input type=" " id="sports" name="interests" value="Sports">

Wrong → type=" " invalid.

Fix:

<input type="checkbox" id="sports" name="interests" value="Sports">

5. Reset button incorrect

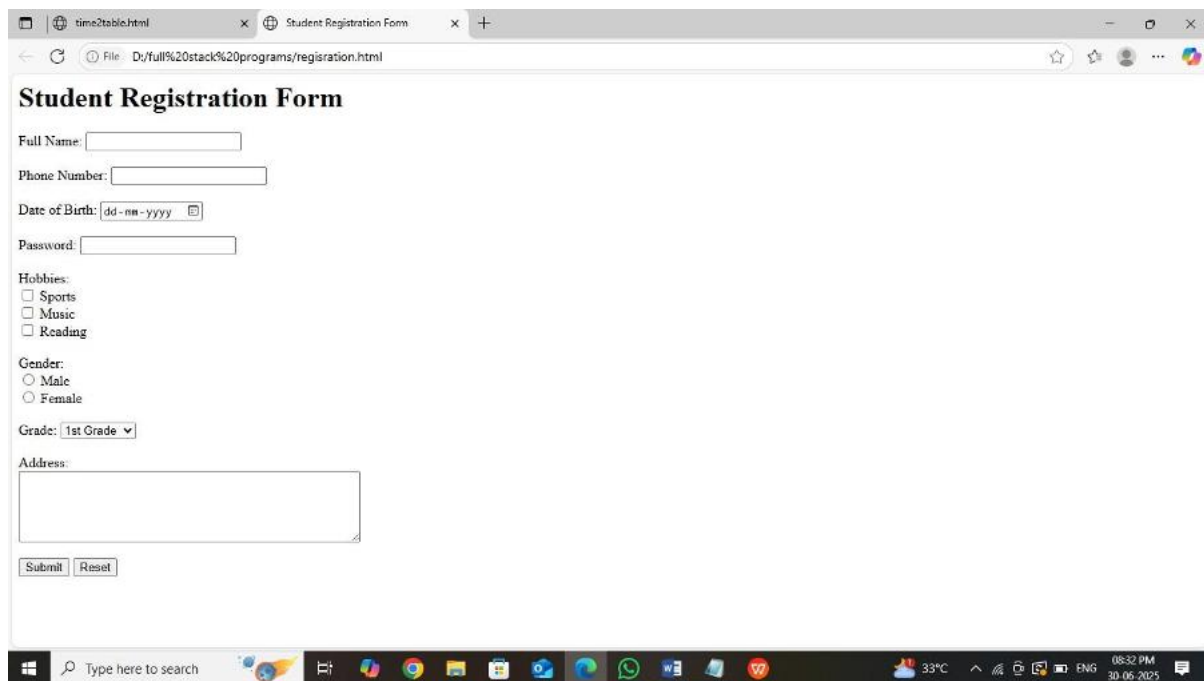
`<input type="r" value="Reset">`

✗ Wrong → type="r" is invalid.

✓ Fix:

`<input type="reset" value="Reset">`

OUTPUT:

A screenshot of a web browser window displaying a 'Student Registration Form'. The browser's address bar shows the file path 'D:/full%20stack%20programs/registration.html'. The form includes fields for 'Full Name', 'Phone Number', 'Date of Birth' (with a calendar icon), 'Password', and a 'Hobbies' section with checkboxes for 'Sports', 'Music', and 'Reading'. There is also a 'Gender' section with radio buttons for 'Male' and 'Female', a 'Grade' dropdown menu currently set to '1st Grade', and a large 'Address' text area. At the bottom of the form are 'Submit' and 'Reset' buttons. The Windows taskbar is visible at the bottom of the screen.

2D)FRAMES:

FRAME.HTML:

`<html>`

`<head>`

`<title>Frames Example</title>`

`</head> //BUG4`

`<frameset rows="33%,33%,33%"> //BUG1`

`<frame href="imag2.html" name="imageFrame" //BUG2 //BUG3`

`<frame href="para.html" name="paragraphFrame">`

`<frame href="list.html" name="linkFrame">`

`</frameset>`

`</noframes>`

```
<body>
```

```
    <p>Your browser doesn't support frames. Please update your browser or use another one that supports frames.</p>
```

```
    <p>Here are the contents that would typically be displayed:</p>//BUG5
```

```
</body>
```

```
</html>
```

IMAGE.HTML

```
<html>
```

```
<head>
```

```
<title>HTML IMAGES</title>
```

```
</head>
```

```
<body>
```

```
<h1>MY IMAGE</h1>
```

```

```

```
<h1>MY FRIEND IMAGE</h1>
```

```

```

```
</body>
```

```
</html>
```

PARAGRAPH .HTML

```
<html>
```

```
<body>
```

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

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

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

```
</body>
```

```
</html>
```

List.html:

```
<html>
```

```
<head>
```

```
<title>HTML LISTS</title>
```

```
</head>

<body>

<h1>UN ORDERED LIST</h1>

<ul>

<li>Beetroot</li>

<li>Ginger</li>

<li>Potato</li>

<li>Radish</li>

</ul>

<h1>ORDERED LIST</h1>

<ol>

<li>Beetroot</li>

<li>Ginger</li>

<li>Potato</li>

<li>Radish</li>

</ol>

<h1>HTML Definition List</h1>

<dl>

<dt><b>HTML</b></dt>

<dd>This stands for Hyper Text Markup Language</dd>

<dt><b>HTTP</b></dt>

<dd>This stands for Hyper Text Transfer Protocol</dd>

</dl>

</body>

</html>
```

Errors:

1. Frameset not closed properly

```
<frameset rows="33%,33%,33%">
...
<frameset>
```

</frameset>

2. Frame tag uses href instead of src

```
<frame href="imag2.html" name="imageFrame"
<frame src="imag2.html" name="imageFrame">
```

3. Unclosed <frame> tag

```
<frame href="imag2.html" name="imageFrame"
<frame src="imag2.html" name="imageFrame">
```

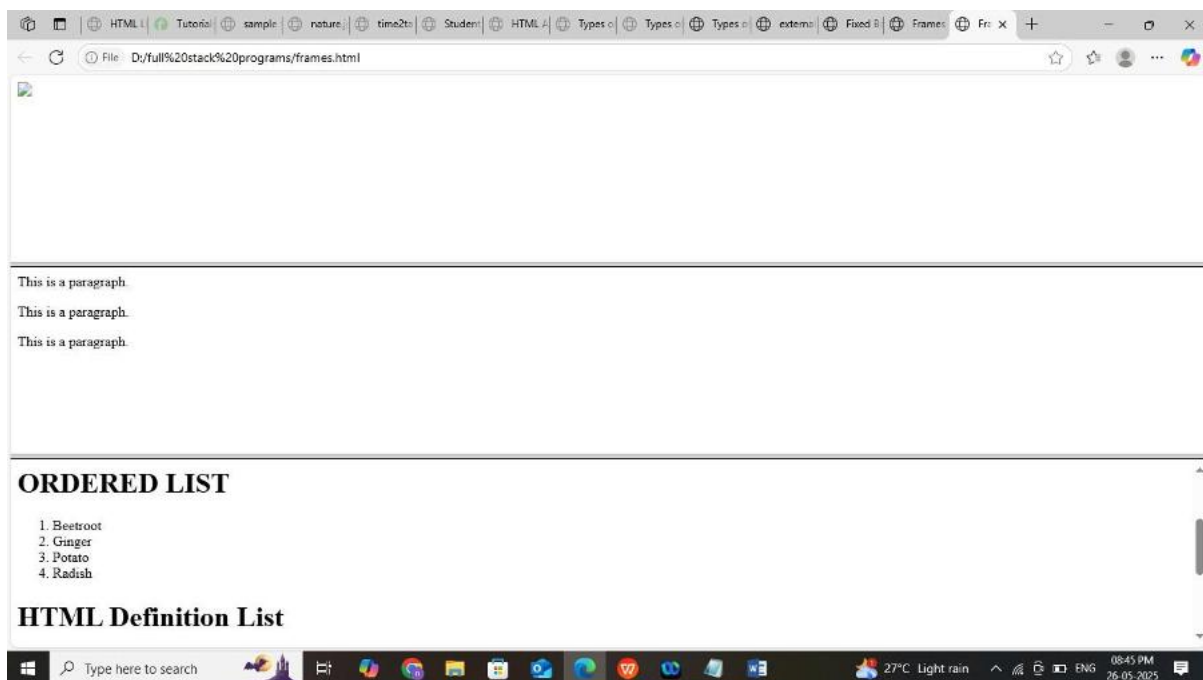
4. Head & body mismatch

You opened <head> but didn't close it before <frameset>. </head>

5. noframes section not closed properly

Missing </noframes> before </html>.

Output:



EXPERIMRNT-3

3A) SEMANTIC ELEMENTS:

```
<html>
```

```
<head>
```

```
<title>HTML semantic tags</title> //BUG1
```

```

<style>
aside {
    width: 30%;
    padding-left: 15px;
    margin-left: 15px;
    float: right;
    font-style: italic;
    background-color: lightgray;
}

{//BUG2
    border: 5px outset black;
    background-color: lightblue;
    text-align: center;
</style>
</head>
<body>  //BUG5

    <h1><u>Ex-header</h1></u>

    <p><h1>EXAMPLE OF SEMANTIC TAGS</p><h1>//BUG3 //BUG4
</header>

    <article>

    <h1><u>Ex-Article</h1></u>

    <h2>AU</h2>

    <p>Welcome to Annamacharya University!!!!</p>

    <section>

    <h1><u>Ex-section</p></u>//BUG6

    <h1>AITS</h1>

    <p>Annamacharya Institute of Technology & Sciences, Rajampet is an AUTONOMOUS institution and
    is affiliated to JNTUA, Anantapuramu</p>

    </section>

    <h1><u>Ex-aside</h1></u>

    <aside>

```

<p>Nature consists of everything that we can see and feel. It is a home to different kinds of living and non-living beings..</p>

</aside>

<p>Environmental Protection</p>

<p>Placing regulations upon cutting down specific trees.

Limiting the consumption of fossil fuels hydrogen fuel cells.

</p>

<p>soo.... </p>

<p> Please save water</p>

<h1><u>Ex-figure and figure caption</h1></u>

<h2>AI</h2>

<p>technology that enables computers and machines to simulate human learning, comprehension, problem solving, decision making, creativity and autonomy.</p>

<figure>

<center></center>

<figcaption><center>Fig.1 - AIDS</figcaption></center>

</figure>

<main>

<h1><u>Ex-main</h1></u>

<h1>Most Popular Browsers</h1>

<p>Chrome, Firefox, and Edge are the most used browsers today.</p>

</main>

<nav>

<h1><u>Ex-Nav</h1></u>

HTML |

CSS |

JavaScript |

Python

</nav>

<h1><u>Ex-Div</h1></u>

<div>

<h2>HTML</h2>

```
<p>Hyper Text Markup Language</p>

</div>

<h1><u>Ex-span</h1></u>

<p><span style="color:blue;font-weight:bold">ANNAMACHARYA</span> <span
style="color:red;font-weight:bold">UNIVERSITY</span></p>

<h1><u>Ex-footer</h1></u>

<footer>

  <p>Author:XXXXXXX<br>

  <a href="mailto:XXXXX@example.com">XXXX@example.com</a></p>

</footer>

</body>

</html>
```

Errors:

1)Title tag not closed properly

```
<title>HTML semantic tags<title>
<title>HTML semantic tags</title>
```

2. CSS error: extra { } block

```
{
  border: 5px outset black;
  background-color: lightblue;
  text-align: center;
}
body {
  border: 5px outset black;
  background-color: lightblue;
  text-align: center;
}
```

3. Incorrect nesting of <h1> and <u>

```
<h1><u>Ex-header</h1></u>
<h1><u>Ex-header</u></h1>
```

4. Paragraph & heading wrongly nested

```
<p><h1>EXAMPLE OF SEMANTIC TAGS</p><h1>
<h1>EXAMPLE OF SEMANTIC TAGS</h1>
```

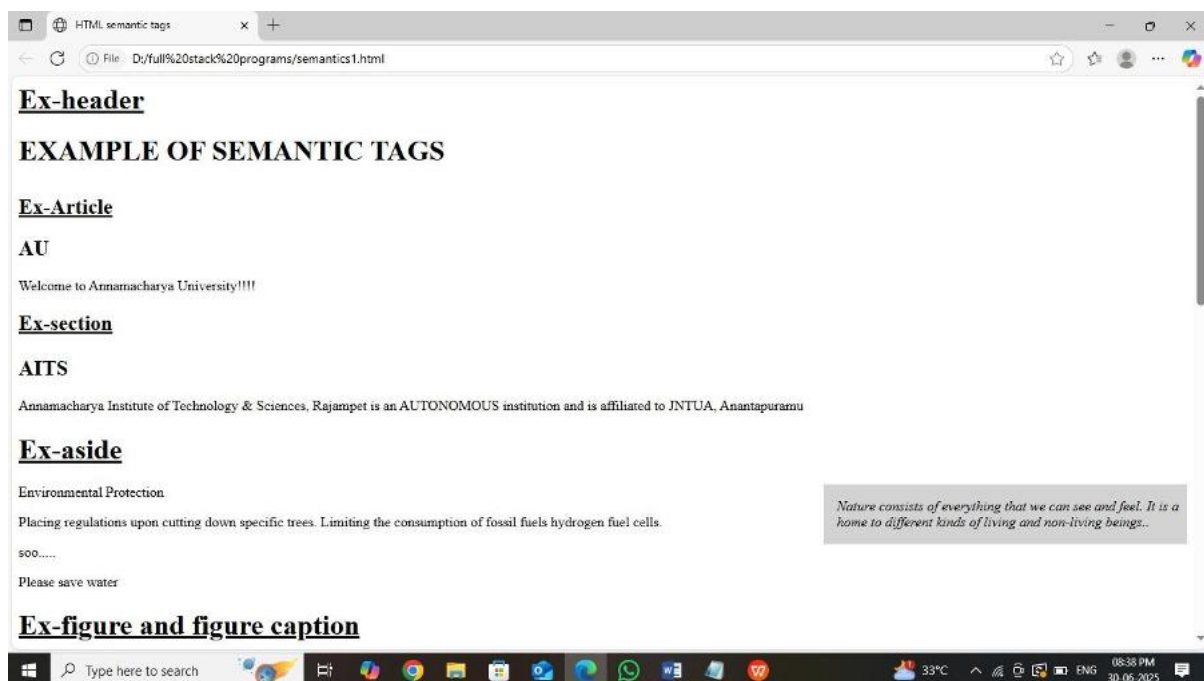
5. Missing <header> opening tag

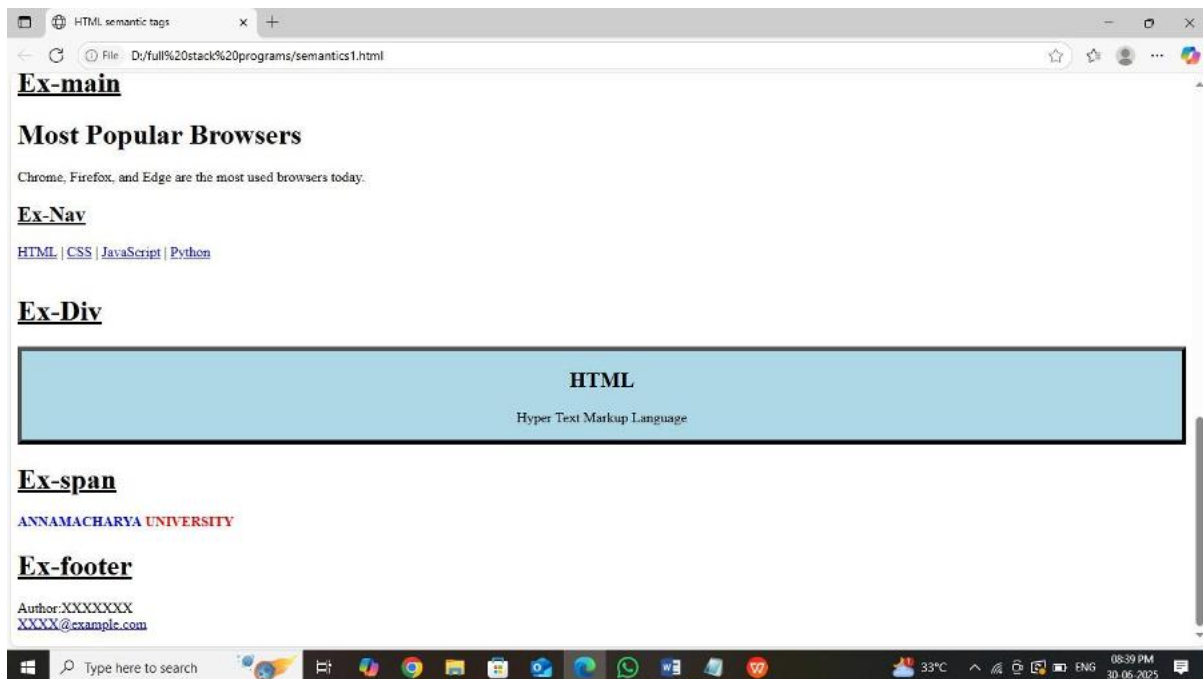
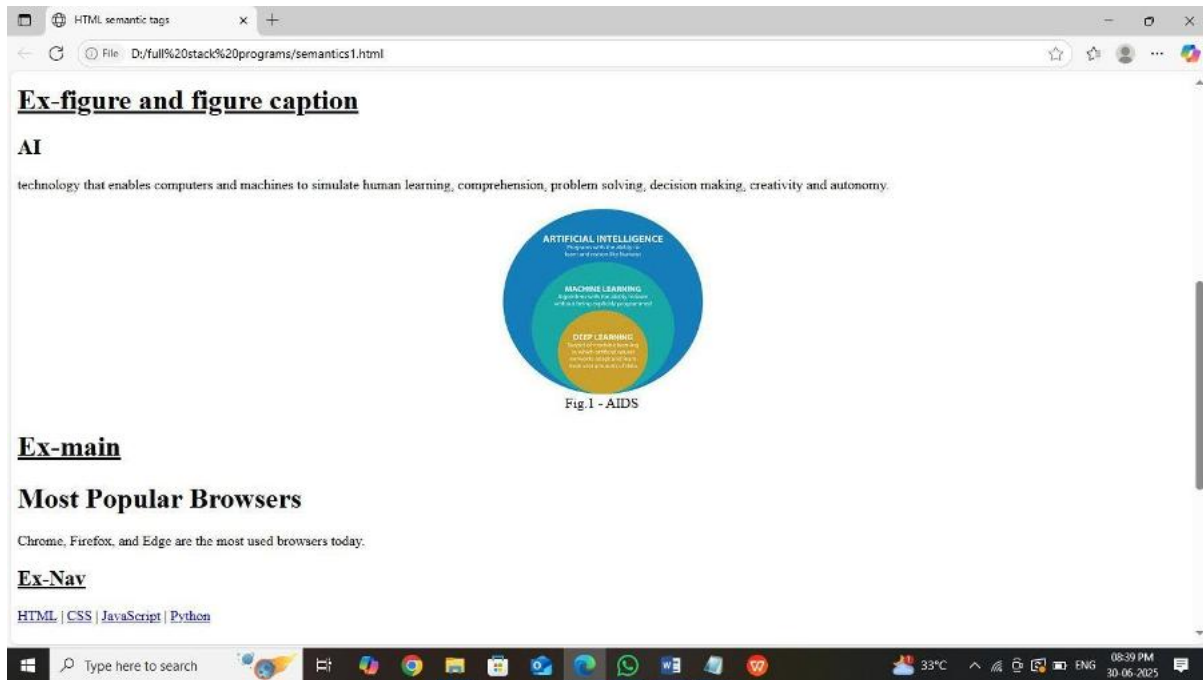
```
</header>
<header>
  <h1>Ex-header</h1>
</header>
```

6. Section heading wrong nesting

```
<h1><u>Ex-section</p></u>
<h1><u>Ex-section</u></h1>
```

OUTPUT:





3B)AUDIO AND VIDEO FILES:

```
<html>
<head>
<title>HTML Audio and video files</title>
</head>
<body>
<h1>Audio and Video</h1>
<p>Click here to play Audio</p>
```

```

<audios> //BUG1
<source src="D:\full stack programs\watersound.mp3">//BUG2
<p>Your browser does not support HTML audio.</p>
</audio>
<p>Click here to play Video</p>
<video>//BUG3
<source heef="D:\full stack programs\video.mp4">
<p>Your browser does not support HTML video.</p>
</video>
</body>
<html>//BUG5

```

Errors:

1. Wrong tag name

```

<audios>
<audio controls>

```

2. File path issue (Windows local path)

```
src="D:\full stack programs\watersound.mp3"
```

Won't work in browsers unless run locally with correct permissions.

Fix → Keep the audio/video file in the **same folder** as your HTML file and use a **relative path**:

```
src="watersound.mp3"
```

3. Missing controls attribute

Without it, the play/pause buttons won't appear.

Add:

```

<audio controls> ... </audio>
<video controls> ... </video>

```

4. Wrong attribute name in <video>

```

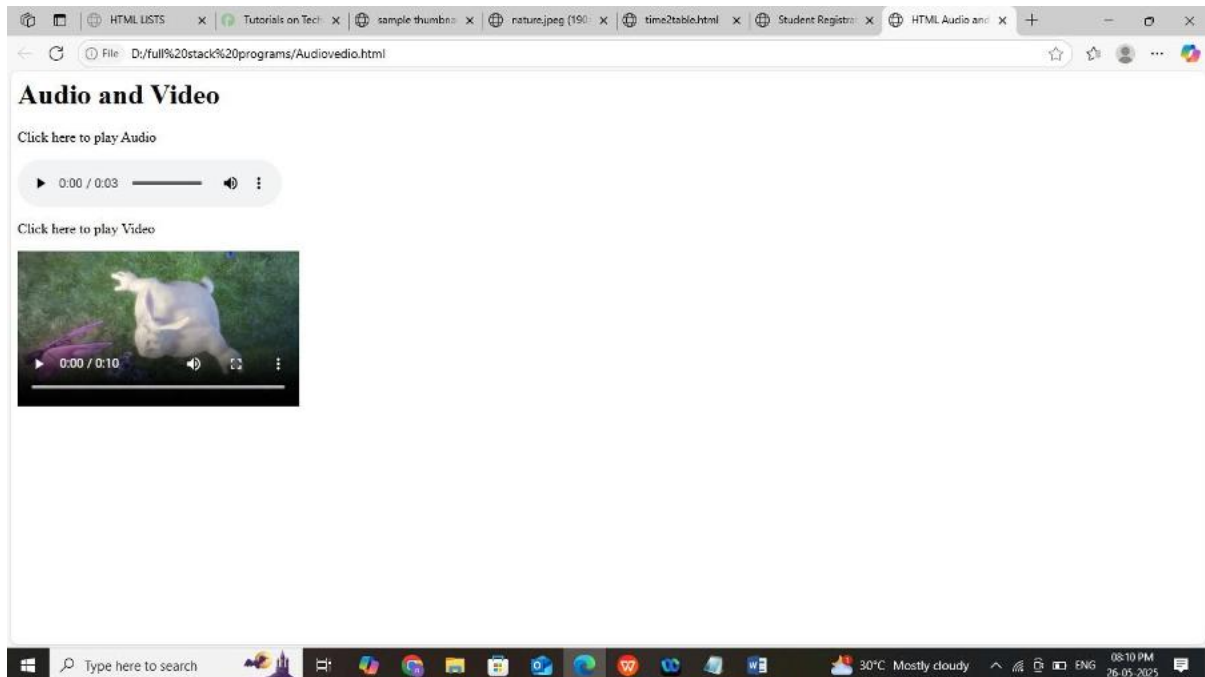
<source heef="video.mp4">
<source src="video.mp4" type="video/mp4">

```

5. Unclosed <html> properly

At the end, you wrote <html> instead of </html>.

Output:



3C) TYPES OF CSS:

Inline:

```
<html>
```

```
<head>
```

```
<title>Types of CSS</title> //BUG1
```

```
<body>
```

```
<h3><u>Ex-Inline CSS</h3>
```

```
<h1 src="color:blue;">HYPER TEXT MARKUP LANGUAGE</h1> //BUG2
```

```
<p src="color:red;"> CASCADING STYLE SHEETS</p> //BUG3
```

```
</body> //BUG4
```

```
</html>
```

Error:

1. Unclosed <title> tag

```
<title>Types of CSS<title>
```

<title>Types of CSS</title>

2. Wrong attribute used (src instead of style)

```
<h1 src="color:blue;">HYPER TEXT MARKUP LANGUAGE</h1>
<h1 style="color:blue;">HYPER TEXT MARKUP LANGUAGE</h1>
```

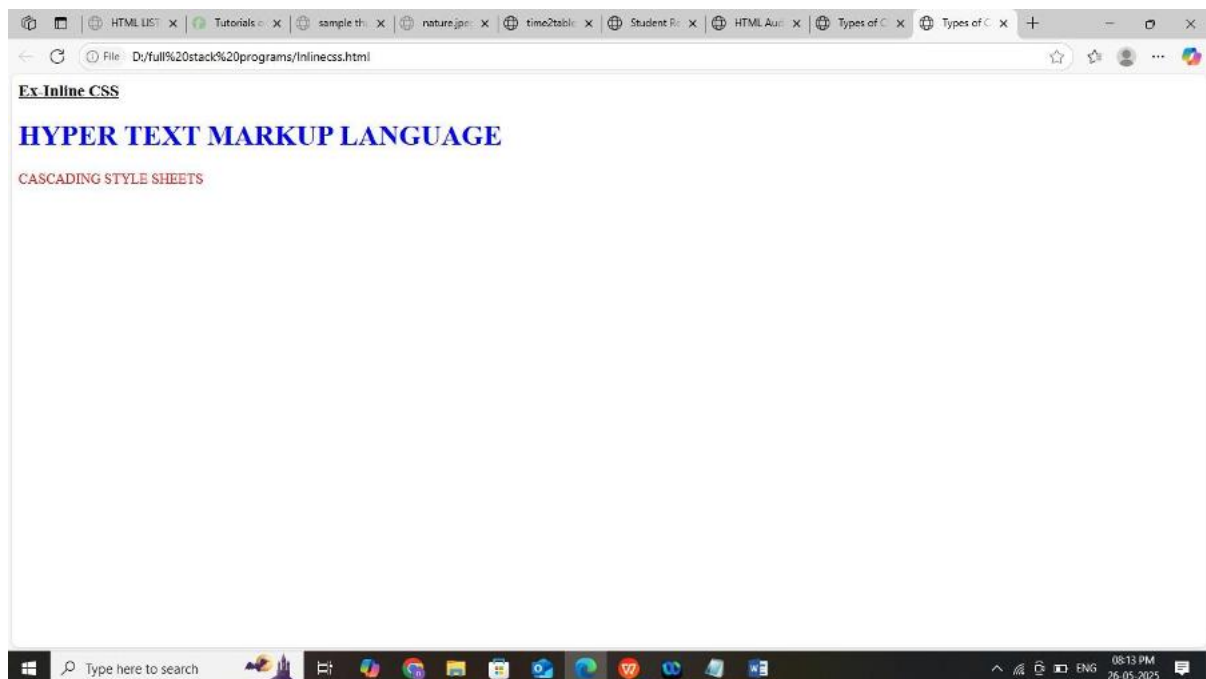
3. Paragraph tag not closed properly

```
<p src="color:red;"> CASCADING STYLE SHEETS<p>
<p style="color:red;">CASCADING STYLE SHEETS</p>
```

4. Missing proper closing of <body> and <html>

You wrote <body> again instead of </body> and <html> instead of </html>.

Output:



EXPERIMENT-4

4) SELECTOR FORMS:

<html>

<head>

<title>CSS Selector Demo<title> //BUG1 //BUG2

```
/* Simple Selectors */
```

```
h1 {
```

```
    color: blue; /* Element selector */
```

```
}
```

```
#my-id {
```

```
    background-color: yellow; /* ID selector */
```

```
}
```

```
.my-class {
```

```
    font-weight: bold; /* Class selector */
```

```
}
```

```
h2, h3 {
```

```
    color: green; /* Group selector */
```

```
}
```

```
* {
```

```
    margin: 0; /* Universal selector */
```

```
}
```

```
/* Combinator Selectors */
```

```
div p {
```

```
    color: red; /* Descendant selector */
```

```
}
```

```
div > p {
```

```
    font-style: italic; /* Child selector */
```

```
}
```

```
h2 + p {
```

```
    color: purple; /* Adjacent sibling selector */
```

```
}
```

```
h1 ~ p {
```

```
    text-decoration: underline; /* General sibling selector */
```

```
}
```

```
/* Pseudo-Class Selectors */
a:hover {
    color: orange; /* Hover state */
}
p:nth-child(2) {
    background-color: lightgray; /* Pseudo-class for the second child */
}

/* Pseudo-Element Selectors */
p::first-line {
    font-weight: bold; /* Bold the first line of a paragraph */
}
p::before {
    content: "Note: "; /* Insert content before the paragraph */
    font-weight: bold;
}

/* Attribute Selectors */
a[target="_blank"] {
    color: red; /* Attribute selector for links opening in new tab */
}
input[type="text"] {
    border: 2px solid blue; /* Attribute selector for text input */
}
</style>
</head>
<body>

<h1 id="my-id">Hello, CSS Selectors!</h1>
<h2 class="my-class">Simple CSS Example</h2>
```

<h3>Selectors in Action</h3>

<div>

<p>This paragraph is a descendant of the div.</p>

<p>This paragraph is a child of div and styled differently.</p>

</div>

<h2>Sibling Selectors<h2>

<p>This paragraph appears after the heading (adjacent sibling).</p>

<p>This paragraph shares the same parent as the heading (general sibling).</p>

<h3>Links and Inputs<h3>

Open Example //BUG4

<input type="t" placeholder="Type here"> //BUG5

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

<p>This is the second paragraph to demonstrate nth-child pseudo-class styling.</p>

</html> //BUG6

Errors:

1. Unclosed <title> tag

```
<title>CSS Selector Demo<title>
<title>CSS Selector Demo</title>
```

2. Missing <style> tag for CSS

```
<style>
/* CSS here */
</style>
```

3. Wrong HTML closing tags

```
<h2>Sibling Selectors<h2>
```

<h2>Sibling Selectors</h2>

(Same issue with <h3> tags later.)

4. Broken <a> tag (typo in attribute)

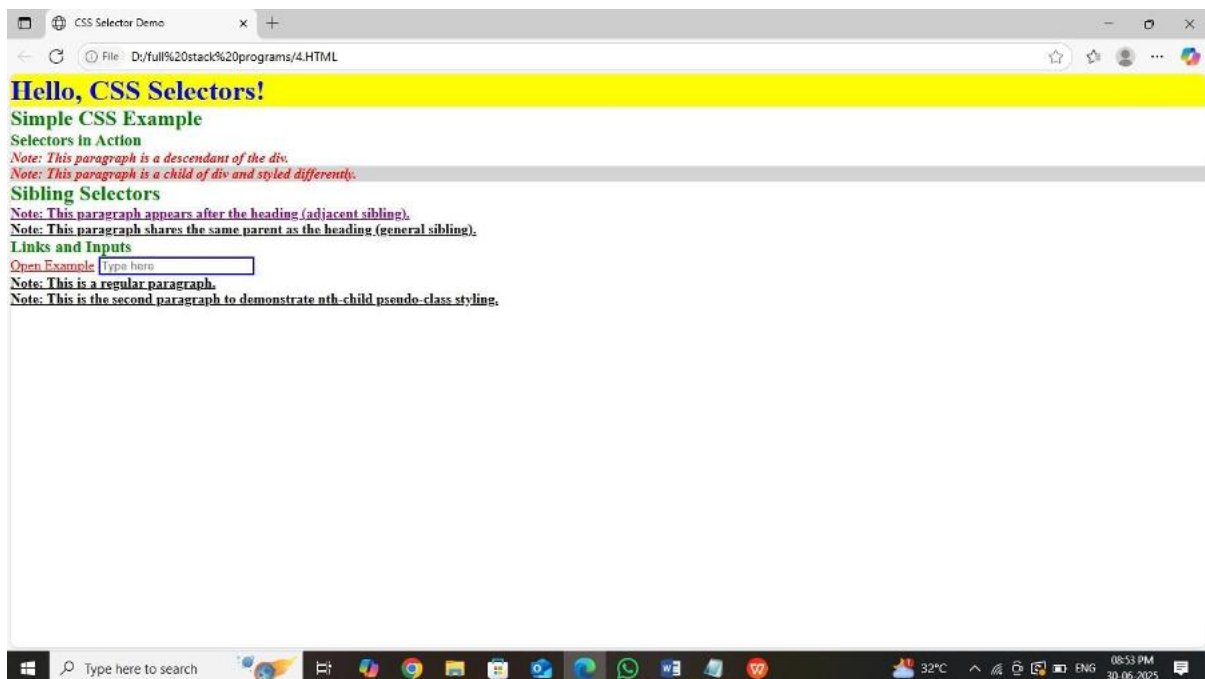
```
<a srcf=https://example.com" target="_blank">Open Example</a>
<a href="https://example.com" target="_blank">Open Example</a>
```

5. Incorrect <input> type

```
<input type="t" placeholder="Type here">
<input type="text" placeholder="Type here">
```

6. Some tags not closed properly (</body>, </html> missing).

OUTPUT:



EXPERIMENT-5

5.a) colors.html

<html>

<head>

<title>Color Reference in CSS</title> //BUG1

<style>

```
body {  
    font-family: Arial, sans-serif;  
    background-color: pink;  
    padding: 20px;  
}  
color-box { //BUG2  
    width: 100px;  
    height: 100px;  
    margin: 10px;  
    display: inline-block;  
    border: 1px solid #000;  
}  
.color-name {  
    background-color: aqua; /* Named Color */  
}  
.color-hex //BUG3  
    background-color: #ff5733; /* HEX Color */  
}  
.color-rgb {  
    background-color: rgb(75, 190, 95); /* RGB Color */  
}  
.color-rgba {  
    background-color: rgba(255, 255, 0, 0.5); /* RGBA Color */  
}  
.color-hsl {  
    background-color: hsl(240, 100%, 50%); /* HSL Color */  
}  
color-hsla { //BUG4  
    background-color: hsla(120, 100%, 50%, 0.5); /* HSLA Color */  
}  
</style>
```

```
</head>

<body>

  <h1>Demonstrating Various Ways to Reference Colors in CSS</h1>

  <div class="color-box color-name"></div>

  <p>Named Color: aqua</p>

  <div class="color-box color-hex"></div>

  <p>HEX Color: #ff5733</p>

  <div id="color-box color-rgb"></div> //BUG5

  <p>RGB Color: rgb(75, 190, 95)</p>

  <div class="color-box color-rgba"></div>

  <p>RGBA Color: rgba(255, 255, 0, 0.5)</p>

  <div class="color-box color-hsl"></div>

  <p>HSL Color: hsl(240, 100%, 50%)</p>

  <div cid="color-box color-hsla"></div> //BUG6

  <p>HSLA Color: hsla(120, 100%, 50%, 0.5)</p>

</body>

</html>
```

Errors:

1. Unclosed <title> tag

```
<title>Color Reference in CSS<title>
<title>Color Reference in CSS</title>
```

2. Wrong selector (color-box instead of .color-box)

```
color-box {
  width: 100px;
```

```

...
}
.color-box {
  width: 100px;
  ...
}

```

3. Missing { in .color-hex rule

```

.color-hex
  background-color: #ff5733;
}
.color-hex {
  background-color: #ff5733;
}

```

4. Wrong selector (color-hsla instead of .color-hsla)

```

color-hsla {
  background-color: hsla(120, 100%, 50%, 0.5);
}
.color-hsla {
  background-color: hsla(120, 100%, 50%, 0.5);
}

```

5. Wrong attributes in <div> elements

- RGB box:

```

<div id="color-box color-rgb"></div>
<div class="color-box color-rgb"></div>

```

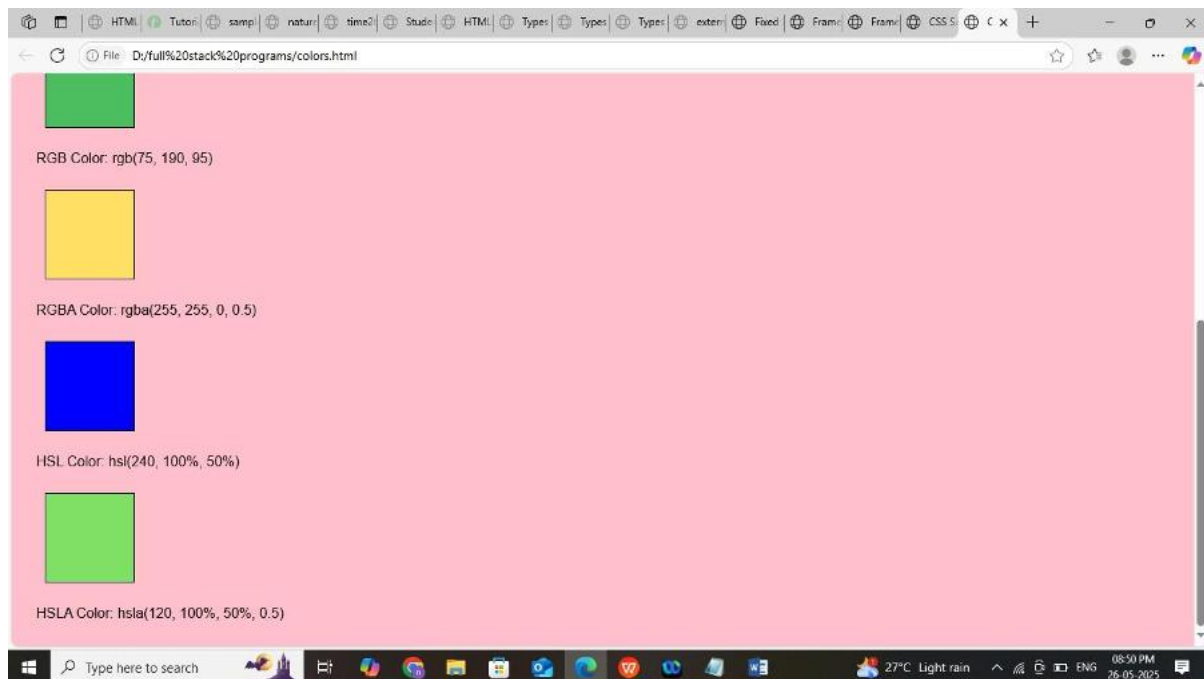
- HSLA box:

```

<div cid="color-box color-hsla"></div>
<div class="color-box color-hsla"></div>

```

Output:



5) B BACKGROUND.HTML

```
<html>
```

```
<head>
```

```
<title>Fixed Background Image</title> //BUG1
```

```
<style> //BUG2
```

```
body {
```

```
margin: 0;
```

```
padding: 0;

height: 50vh;

overflow: hidden;

}
```

```
.background-image {

    background-image: url('https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcTIlFPkEarzbeyY89hsnmbLWloiQrn7LT6XQ&s');

    background-size: cover;

    min-height: 500px;

    background-repeat: no-repeat ;

    background-attachment: fixed;

    background-position: center;

    transform: rotate(0deg);

    position: relative;

    height: 100vh;

    top: 10%;

    transform: translateY(20%) rotate(0deg);

}
```

```
content { //BUG3

    position: relative;

    z-index: 1;

    color: red;

    text-align: center;

    padding: 10px;

}
```

</style>

<head>

<body>

```
<div class="background-image"></div>

<div class="content">

  <h1>Welcome to Annamcharya university</h1>

  <p>Knowledge is power</p>

</div>  //BUG4


</body>

</html>
```

Errors:

1. Unclosed <title> tag

```
<title>Fixed Background Image<title>
<title>Fixed Background Image</title>
```

2. Unclosed <head> tag

```
<head> ... <head>
</head>
```

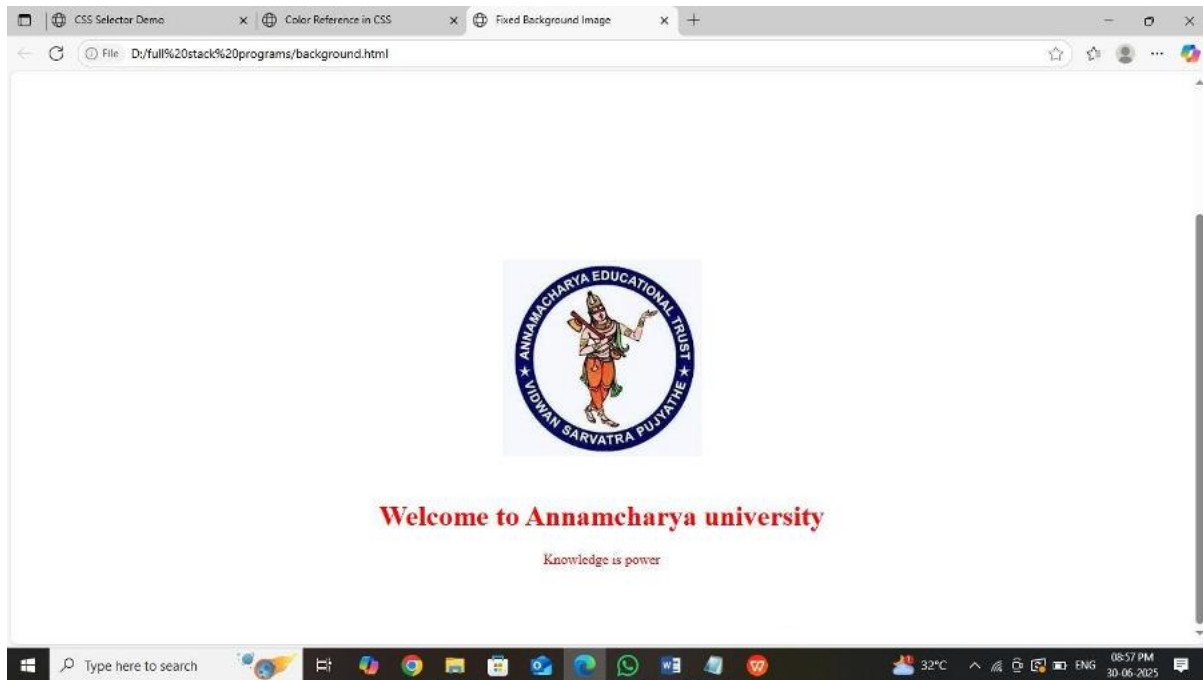
3. Wrong CSS selector content { ... }

```
content {
  ...
}
.content {
  ...
}
```

4. Extra <div> not closed properly

```
<div class="content">
  ...
<div>
</div>
```

OUTPUT:



5.c) font.html

```
<html>
```

```
<head>
```

```
<title>CSS Font and Text Properties</title> //BUG1
```

```
<style>
```

```
body {
```

```
    font-family: Arial, sans-serif;
```

```
    padding: 20px;
```

```
    background-color: green;
```

```
}
```

```
.font-size {
```

```
    font-size: 40px;
```

```
}
```

```
.font-weight {
```

```
    font-weight: bold;
```

```
}
```

```
.font-style {
```

```
    font-style: italic;
```

```
}
```

```

    .text-decoration {
        text-decoration: underline;
    }

    .text-transform {
        text-transform: uppercase;
    }

    .text-alignment {
        text-align: center;
    }
</style>
</head>
<body>

    <h1>Demonstrating CSS Font and Text Properties</h1>

    <p id="font-size">Artificial Intelligence and Data Science(AIDS)</p>//BUG3 //BUG4

    <p cid="font-weight">Artificial Intelligence and Data Science(AIDS)</p>//BUG2

    <p cid="font-style">Artificial Intelligence and Data Science(AIDS)</p>

    <p class="text-decoration">Artificial Intelligence and Data Science(AIDS)</p>

    <p cid="text-transform">Artificial Intelligence and Data Science(AIDS)</p>

    <h2 class="text-alignment">Artificial Intelligence and Data Science(AIDS)</h2>//BUG5

</html>

```

Errors:

1. Unclosed <title> tag

```
<title>CSS Font and Text Properties</title>
<title>CSS Font and Text Properties</title>
```

2. Wrong attribute cid instead of class or id

```
<p cid="font-weight">...</p>
```

- Use either id or class. Since you already defined **classes** in CSS (.font-size, .font-weight, etc.), you should use class.

Example fix:

```
<p class="font-weight">Artificial Intelligence and Data Science(AIDS)</p>
```

3. Inconsistent usage of id vs class

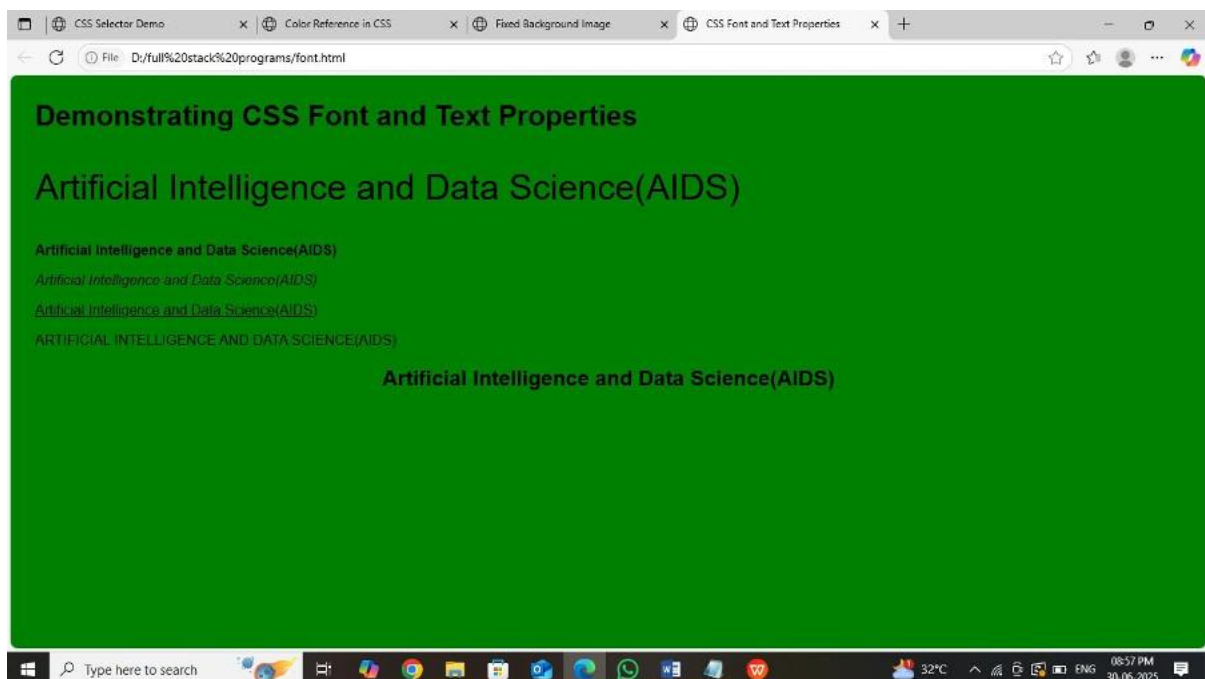
```
<p class="font-size">Artificial Intelligence and Data Science(AIDS)</p>
```

4. Unclosed <body> and <html> tags

You ended with only </html> but forgot </body> before it.

```
</body>
</html>
```

OUTPUT:



5.d)box.html

```
<html>

<head>

  <title>CSS Box Model</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      background-color: #f4f4f4; /* Light grey background for visibility */

    }

    .box //BUG1

      background-color: #ffffff; /* White background for the box */

      color: #333; /* Dark text color */

      width: 300px; /* Width of the content area */

      text-align: center; /* Center the text */

      margin: 20px; /* Space outside the border */

      border: 5px solid #4CAF50; /* Green border */

      padding: 20px; /* Space inside the box, between content and border */

    .content {

      background-color: #e7f3fe; /* Light blue background for content */

      padding: 15px; /* Padding inside the content area */

      margin: 10px; /* Margin outside the content */

    }

    .border {

      border: 4px dashed #f44336; /* Red dashed border */

      padding: 10px; /* Padding inside the border */

      width: 100%; /* Full width of the box */

      margin: 10px 0; /* Margin for separation */

    }

    .padding {
```

```

padding: 20px; /* Increased padding */

background-color: #ffe0b2; /* Light orange background for padding section */
}

.margin {

background-color: #fff3e0; /* Light peach background for margin section */

padding: 10px; /* Padding inside the margin section */

margin: 20px 0; /* Margin space *////BUG2

</style>
</head>
<body>

<h1>Understanding the CSS Box Model</h1>

<div class="box" //BUG3

<h2>Box Model Example</h2>

<div class="content">AIML</div>

<div class="border">AIDS</div>

<div class="padding">CSE(DS)</div>

<div class="margin">CSE(AI)</div //BUG4

</div>

<p>

The CSS Box Model is essential for layout design in web development. It consists of:

<ul>

<li><strong>AIML:</strong>Artificial Intelligence and Machine Learning</li>

<li><strong>AIDS:</strong> Artificial Intelligence and Data Science</li>

<li><strong>CSE(DS):</strong>Computer science and engineering(Data Science)</li>

<li><strong>CSE(AL):</strong>Computer science and engineering( Machine Learning) </li>

</body>

```

</html>

Errors:

1. Missing curly braces {} in .box

```
.box
  background-color: #ffffff;
.box {
  background-color: #ffffff;
  ...
}
```

2. Unclosed .margin CSS rule

```
.margin {
  background-color: #fff3e0;
  padding: 10px;
  margin: 20px 0;
.margin {
  background-color: #fff3e0;
  padding: 10px;
  margin: 20px 0;}
```

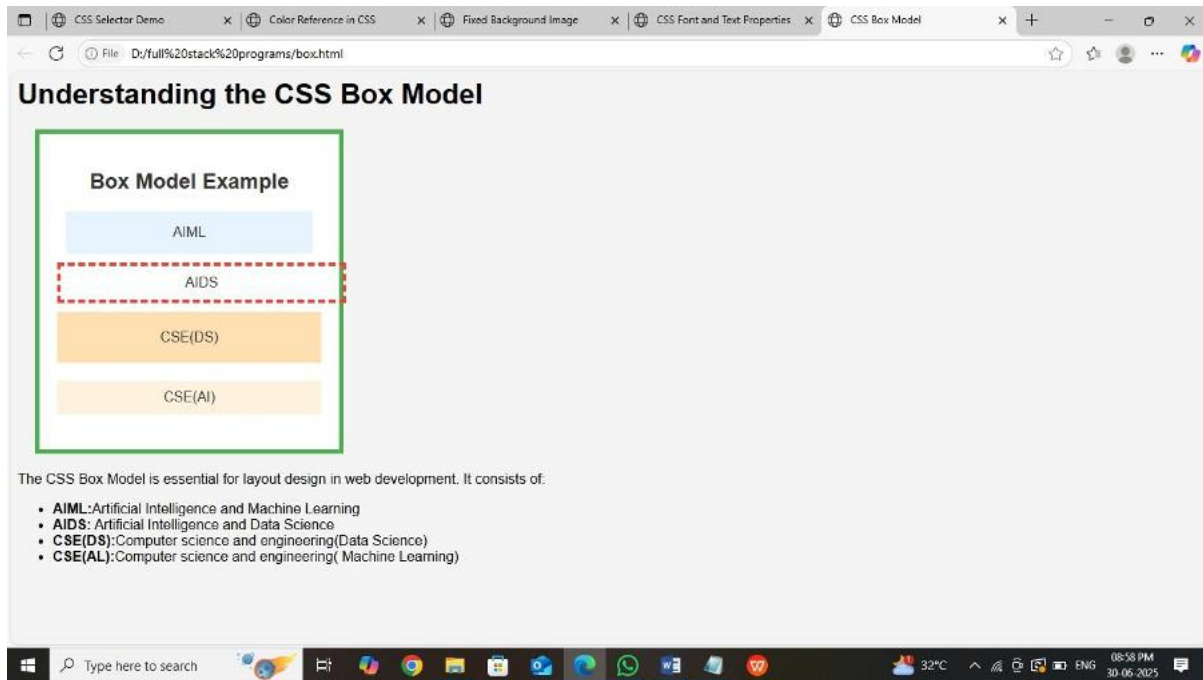
3. Broken <div> tag

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

4. Broken closing tag

```
<div class="margin">CSE(AI)</div
<div class="margin">CSE(AI)</div>
```

OUTPUT:



Exercise--6

6A))internal and external js:

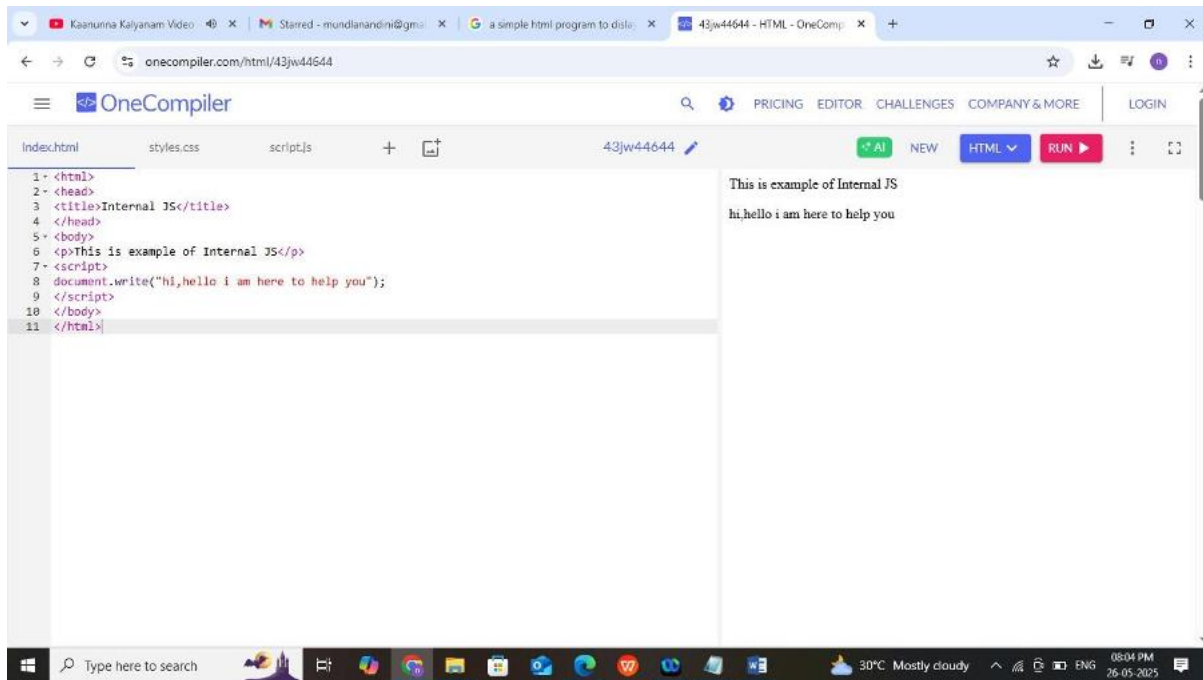
Internal js:

```
<html>//BUG1
<title>Internal JS</title>
</head>
<p>This is example of Internal JS //BUG2 //BUG4
document.write("hi,hello i am here to help you"); //BUG3
</script>
</body>
</html>
```

Errors:

1. You forgot to wrap your JavaScript code inside `<script>...</script>`.
2. `<head>` is not closed properly.
3. The `<p>` tag is opened but never closed.
4. You should place JS inside `<body>` or `<head>`.

Output:



External1. js:

```
document.write("hi <br>");
```

```
document.write("hello,<br>");
```

```
document.write("students!!!!<br>");
```

```
document.write("welcome to AITS");
```

html:

```
<html>//BUG3
```

```
<title> externall JS</title>
```

```
</head>
```

```
<body>
```

```
<script src ="ext.js" //BUG 1 //BUG2
```

```
</script>
```

```
</body>//BUG4
```

Errors:

- 1) In your HTML, <script src="ext.js" is **missing a closing >**.
- 2) You named your file External1.js but in HTML you wrote ext.js → mismatch.
- 3) <title> is outside <head>.
- 4) <html> doesn't have <head> and </html> closing tags.

Output:

Hi
Hello
Students
Welcome to aits!!!

6D)VOTER ELIGIBILITY OR NOT

```
<html>

<head>

    <title>Voter Information</title> //bug3
body { //BUG1

    font-family: Arial, sans-serif;

    padding: 20px;

}
table {

    border-collapse: collapse;

width: 50%;

margin-top: 20px;

}
th, td {

    border: 1px solid #ddd;

padding: 8px;

text-align: left;

}
th {

    background-color: #f2f2f2;
```

```
}
```

```
</head>
```

```
<body>
```

```
  <h1>Voter Registration</h1>
```

```
  <button onclick="getVoterInfo()">Register Voter</button>
```

```
  <table id="voterTable" style="display:none;">
```

```
    <thead>
```

```
      <tr>
```

```
        <th>Name</th>
```

```
        <th>Age</th>
```

```
        <th>Eligibility to Vote</th>
```

```
      </tr>
```

```
    </thead>
```

```
    <tbody>
```

```
      <tr id="voterRow"></tr>
```

```
    </tbody>
```

```
</table>
```

```
<script>
```

```
  function getVoterInfo() {
```

```
    // Prompt user for name and age
```

```
    let name = prompt("Please enter your name:");
```

```
    let age = prompt("Please enter your age:");
```

```
    // Validate age input
```

```
    if (isNaN(age) || age === null || age.trim() === "")
```

```
    {
```

```
      alert("Please enter a valid age.");
```

```
      return;
```

```
    }
```

```
    // Check voting eligibility
```

```

        let eligibility = age >= 18 ? "Eligible to Vote" : "Not Eligible to Vote"; //bug6

        // Display the information in table format
document.getElementById("voterRow").innerHTML = `

        <td>${name}</td>

        <td>${age}</td>

        <td>${eligibility}</td>

    `;

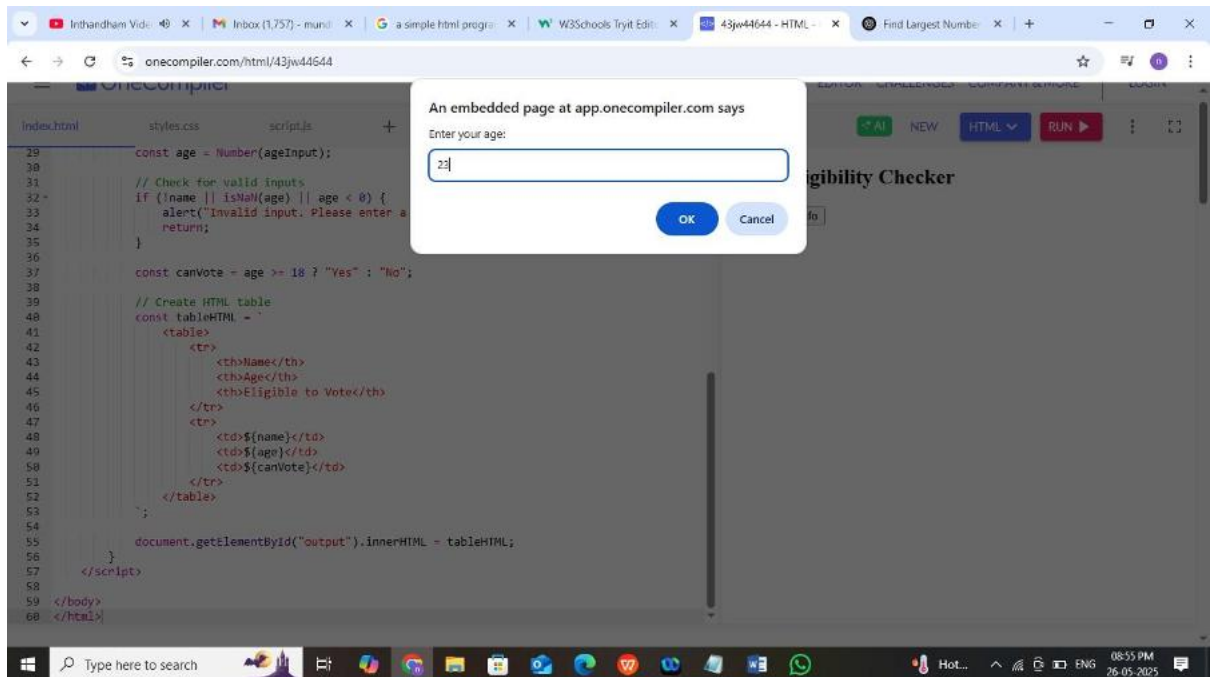
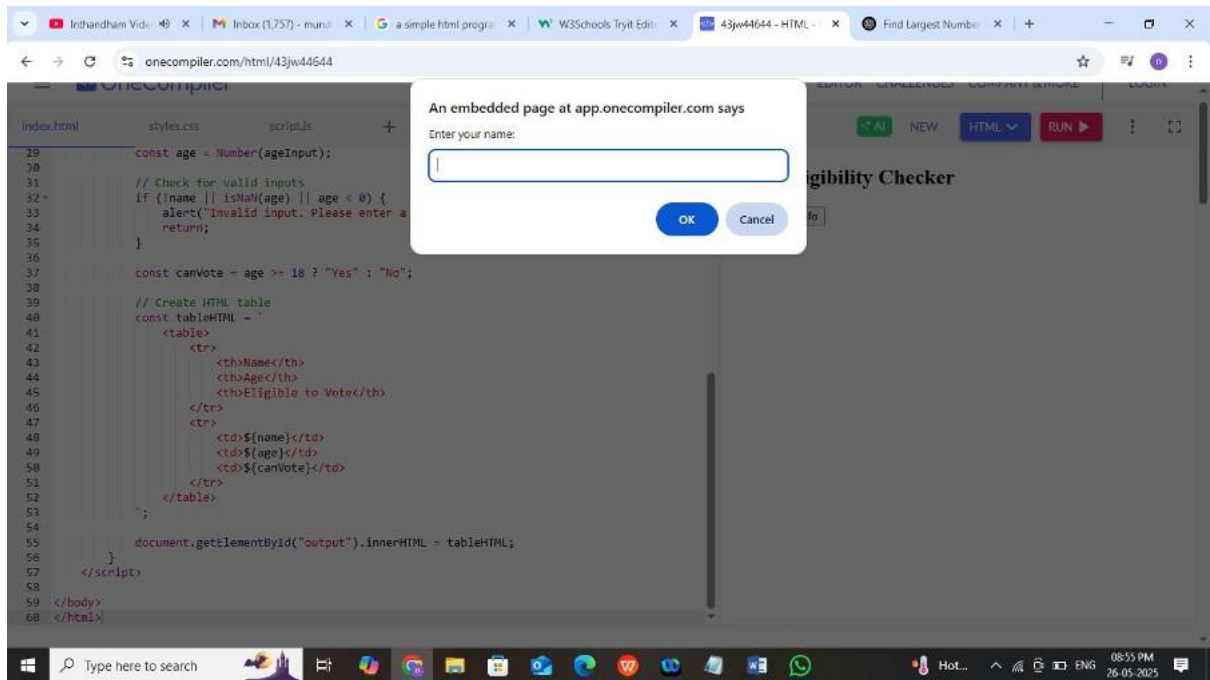
    document.getElementById("voterTable").style.display = "table"; // Show the table
}
</body> //BUG4

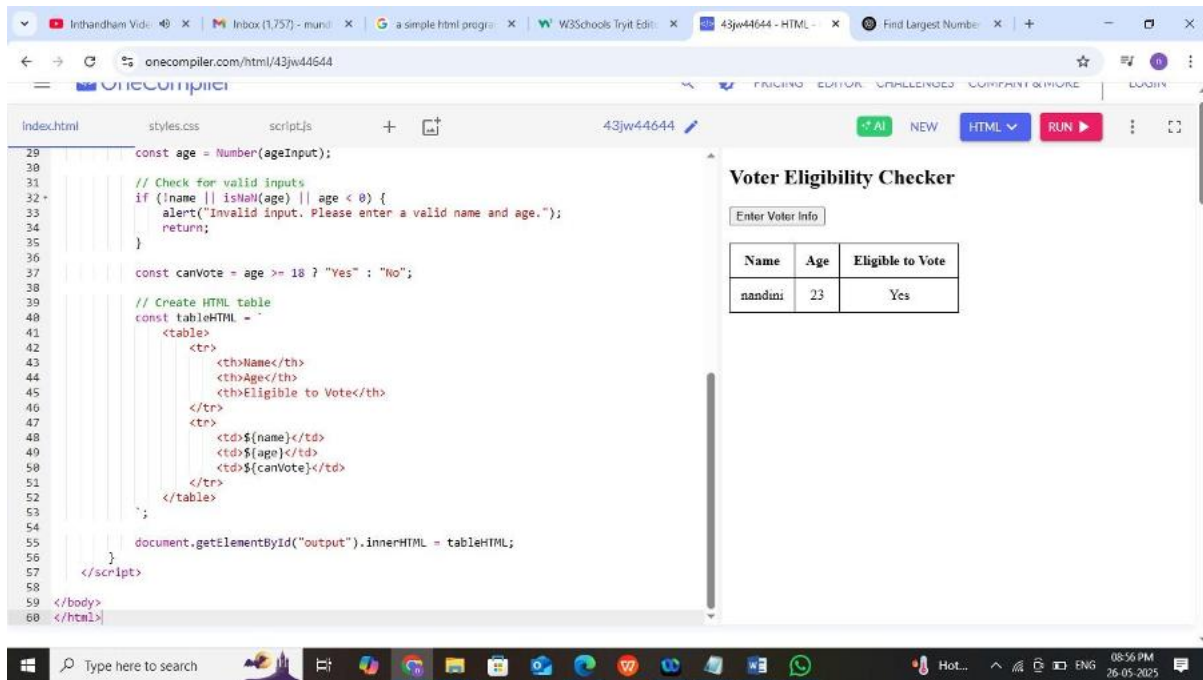
```

Errors:

- 1) You wrote raw CSS inside <head> without <style> tags.
- 2) <body> CSS selector is placed outside <style>.
- 3) <script> is opened but **not closed properly**.
- 4) The last </html> closing tag is missing.
- 5) age is a string from prompt(), so comparison age >= 18 may behave unexpectedly unless converted to a number.

OUTPUT:





Experiment-7

7.a)Documentobject html

<html>

<head>

<title>Document Object in JavaScript</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

background-color: #f9f9f9;

color: #333;

}

h1, h2 {

color: #4CAF50;

}

#results {

font-size: 18px;

margin-top: 20px;

}

```

</style>
</head>
<body>

<h1>Document Object in JavaScript</h1>
<h2>Examples of Document Object Properties and Methods</h2>

<button onclick="showDocumentProperties()">Show Document Properties</button>
<button onclick="changeDocumentTitle()">Change Document Title</button>//BUG3
<button onclick="addNewElement()">Add New Element</button>

<div id="results"></div>//BUG1

function showDocumentProperties{//BUG2
    // Get document properties
    const title = document.title; // Document title
    const bodyContent = document.body.innerHTML; // Body content
    const numElements = document.getElementsByTagName("*").length; // Total
number of elements

    // Display document properties
    resultsHTML = "<strong>Document Properties:</strong><br>";//BUG5
    resultsHTML += `Title: ${title}<br>`;
    resultsHTML += `Number of elements: ${numElements}<br>`;
    resultsHTML += `Body Content: ${bodyContent.length} characters<br>`;

    // Display results in the div
    document.getElementById("results").innerHTML = resultsHTML;
}

function DocumentTitle() {

```

```

// Change the document title
const newTitle = prompt("Enter a new title for the document:");//BUG4
if () {
    document.title = newTitle;
    alert("Document title changed to: " + newTitle);
}
}

function addNewElement() {
    // Create a new paragraph element
    const newParagraph = document.createElement("p");
    newParagraph.textContent = "This is a new paragraph added to the document!";

    // Append the new paragraph to the body
    document.body.appendChild(newParagraph);
    alert("New paragraph added!");
}
</script>

</body>
</html>

```

Errors:

1. Your <script> tag is missing before defining the functions.
2. Function declaration for showDocumentProperties is wrong → missing parentheses ().
3. In changeDocumentTitle() you wrote function DocumentTitle() → mismatch with the button's onclick.
4. In DocumentTitle() you wrote if () { ... } → incomplete condition.
5. Missing let/const before resultsHTML variable.

7.b) windowobject. html

```

<html>
<head>

```

```
<title>Window Object in JavaScript</title>//BUG1
```

```
body {  
    font-family: Arial, sans-serif;  
    margin: 20px;  
    background-color: #f9f9f9;  
    color: #333;  
}  
h1, h2      color: #4CAF50;//BUG2  
}  
results { //BUG3  
    font-size: 18px;  
    margin-top: 20px;  
}
```

```
<style>
```

```
</head>
```

```
<body>
```

```
<h1>Window Object in JavaScript</h1>
```

```
<h2>Examples of Window Object Properties and Methods</h2>
```

```
<button onclick="WindowProperties()">Show Window Properties</button>//BUG4
```

```
<button onclick="openWindow()">Open New Window</button>//BUG5
```

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

```
<script>
```

```
function showWindowProperties() {  
    // Get window properties  
    const width = window.innerWidth; // Width of the window  
    const height = window.innerHeight; // Height of the window
```

```

const url = window.location.href; // Current URL
const title = window.document.title; // Document title


// Display window properties
resultsHTML = "<strong>Window Properties:</strong><br>";
resultsHTML += `Width: ${width}px<br>`;
resultsHTML += `Height: ${height}px<br>`;
resultsHTML += `Current URL: ${url}<br>`;
resultsHTML += `Document Title: ${title}<br>`;


// Display results in the div
document.getElementById("results").innerHTML = resultsHTML;
}


function openNewWindow() {
  // Open a new window
  newWindow = window.open("", "New Window", "width=400,height=300");
  newWindow.document.write("<h2>This is a New Window</h2>");
  newWindow.document.write("<p>You can close this window!</p>");
}
</script>


</body>
</html>

```

Errors:

1. Your <style> tag is misplaced → you opened it at the bottom but CSS rules are outside it.
2. h1, h2 color: #4CAF50; → missing {} braces.
3. results { ... } is invalid → should be #results { ... } (ID selector).
4. Button calls WindowProperties() but your function is named showWindowProperties().
5. Button calls openWindow() but function name is openNewWindow().

6. resultsHTML is not declared with let/const.
7. newWindow should also be declared with let or const.

7.c) arrayobject. html

```
<html>
<head>
  <title>Array Object in JavaScript</title>//BUG1

  body {
    font-family: Arial, sans-serif;
    margin: 20px;
    background-color: #f9f9f9;
    color: #333;
  }
  h1 h2 { //BUG2
    color: #4CAF50;
  }
  #results {
    font-size: 18px;
    margin-top: 20px;
  }
</style>
</head>
<body>

<h1>Array Object in JavaScript</h1>
<h2>Examples of Array Object Properties and Methods</h2>
```

```
<label if="inputArray">Enter a list of numbers (comma-separated):</label><br>//BUG3
```

```
<input type="text" src="inputArray" placeholder="e.g., 10, 20, 30"><br><br>//BUG4
```

```
<button onclick="performArrayOperations()">Process Array</button>
```

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

```
<script>
```

```
function ArrayOperations() {  
    // Get the input value and convert it into an array  
    const input = document.getElementById("inputArray").value;  
    const array = input.split(',').map(Number); // Convert to numbers  
  
    // Array properties and methods  
    const length = array.length; // Length of the array  
    const reversedArray = array.slice().reverse(); // Reversed array  
    const sortedArray = array.slice().sort((a, b) => a - b); // Sorted array  
    const maxVal = Math.max(...array); // Maximum value in the array  
    const minVal = Math.min(...array); // Minimum value in the array  
    const includesVal = array.includes(20); // Check if 20 is included  
    const joinedString = array.join(', '); // Join array into a string//BUG5
```

```
    // Display results
```

```
    let resultsHTML = "<strong>Results<br>";  
    resultsHTML += `Original Array: [${joinedString}]<br>`;   
    resultsHTML += `Length: ${length}<br>`;   
    resultsHTML += `Reversed Array: [${reversedArray.join(', ')}]<br>`;   
    resultsHTML += `Sorted Array: [${sortedArray.join(', ')}]<br>`;   
    resultsHTML += `Maximum Value: ${maxVal}<br>`;   
    resultsHTML += `Minimum Value: ${minVal}<br>`;
```

```

resultsHTML += `Includes 20: ${includesValue}`;

// Display results in the div
document.getElementById("results").innerHTML = resultsHTML;
}
</script>

</body>
</html>

```

Errors:

1. CSS is not inside <style> tag.
2. h1 h2 { ... } is wrong → should be h1, h2 { ... }.
3. <label if="inputArray"> → should be for="inputArray".
4. <input src="inputArray"> → should be id="inputArray".Function name
5. joinedString not declared with let.

7.d) mathobject .html

```

<html>
<head>
  <title>Math Object in JavaScript<\/title>\/\/BUG1
  <\/style>\/\/BUG2
  body {
    font-family: Arial, sans-serif;
    margin: 20px;
    background-color: #f9f9f9;
    color: #333;
  }
  h1> h2 {\/\/BUG3
    color: #4CAF50;
  }
  #results {

```

```
        font-size: 18px;
        margin-top: 20px;
    }
</style>
</head>
<body>

<h1>Math Object in JavaScript</h1>
<h2>Examples of Math Object Properties and Methods</h2>

<label if="number1">Enter first number:</label><br>//BUG4
<input id="number" id="number1"><br><br>

<label for="number2">Enter second number:</label><br>
<input type="number" id="number2"><br><br>//BUG5

<button onclick="performMathOperations">Calculate</button>//BUG6

<div id="results"></div>

<script>
    function performMathOperations { //BUG7
        // Retrieve the numbers from input fields
        const num1 = parseFloat(document.getElementById("number1").value);
        const num2 = parseFloat(document.getElementById("number2").value);

        // Math operations
        const sum = num1 + num2;
        const difference = num1 - num2;
        const product = num1 * num2;
```

```

    const quotient = num2 !== 0 ? (num1 / num2) : 'Division by zero!'; // handle
division by zero

    const power = Math.pow(num1, num2); // num1 raised to the power of num2
    const squareRoot1 = Math.sqrt(num1); // square root of num1
    const squareRoot2 = Math.sqrt(num2); // square root of num2
    const max = Math.max(num1, num2); // maximum of num1 and num2
    const min = Math.min(num1, num2); // minimum of num1 and num2


    // Display results
    resultsHTML = "<strong>Results:</strong><br>";//BUG8
    resultsHTML += `Sum: ${sum}<br>`;
    resultsHTML += `Difference: ${difference}<br>`;
    resultsHTML += `Product: ${product}<br>`;
    resultsHTML += `Quotient: ${quotient}<br>`;
    resultsHTML += `Power (num1^num2): ${power}<br>`;
    resultsHTML += `Square Root of num1: ${squareRoot1}<br>`;
    resultsHTML += `Square Root of num2: ${squareRoot2}<br>`;
    resultsHTML += `Maximum: ${max}<br>`;
    resultsHTML += `Minimum: ${min}`;


    // Display results in the div
    document.getElementById("results").innerHTML = resultsHTML;
}
</script>

</body>
</html>

```

Errors:

1. <title>Math Object in JavaScript</title> → wrong closing tag.
2. Extra </style> before your CSS → breaks styling.
3. CSS selector h1> h2 → wrong, should be h1, h2.

4. <label if="number1"> → should be for="number1".
5. <input id="number" id="number1"> → duplicate id. Should be id="number1" and add type="number".
6. <button onclick="performMathOperations"> → missing parentheses ().
7. Function declaration function performMathOperations { → missing ().
8. resultsHTML not declared with let.

7.e) stringobject .html

```
<html>
<head>
  <title>String Object in JavaScript</title>
  <style>
    #body {//BUG1
      font-family: Arial, sans-serif;
      margin: 20px;
      background-color: #f5f5f5;
      color: #333;
    }
    h1+ h2 {//BUG2
      color: #4CAF50;
    }
    results {//BUG3
      font-size: 18px;
      margin-top: 20px;
    }
  </style>
</head>

<h1>String Object in JavaScript</h1>
<h2>Examples of String Object Properties and Methods</h2>

<label for="inputString">Enter a string:</label><br>
```

```
<input type=" " id="inputString" value="Hello World!"><br><br>//BUG4
```

```
<button onclick="showStringInfo()">Show String Info</button>
```

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

```
<script>
```

```
function showStringInfo() {
```

```
    // Get the input string
```

```
    const str = document.getElementById("inputString").value;
```

```
    // String properties and methods
```

```
    length = str.length; // Length of the string
```

```
    const upperCase = str.toUpperCase(); // Convert to uppercase
```

```
    const lowerCase = str.toLowerCase(); // Convert to lowercase
```

```
    const firstChar = str.charAt(0); // First character of the string
```

```
    const indexOfO = str.indexOf('o'); // Index of first occurrence of 'o'
```

```
    const substring = str.substring(0, 5); // Substring from index 0 to 5
```

```
    const includesWorld = str.includes("World"); // Check if string includes "World"
```

```
    // Display results
```

```
    let resultsHTML = "<strong>Results:</strong><br>";
```

```
    resultsHTML += `Original String: "${str}"<br>`;
```

```
    resultsHTML += `Length: ${length}<br>`;
```

```
    resultsHTML += `Uppercase: "${upperCase}"<br>`;
```

```
    resultsHTML += `Lowercase: "${lowerCase}"<br>`;
```

```
    resultsHTML += `First Character: "${firstChar}"<br>`;
```

```
    resultsHTML += `Index of 'o': ${indexOfO}<br>`;
```

```
    resultsHTML += `Substring (0-5): "${substring}"<br>`;
```

```
    resultsHTML += `Includes "World": ${includesWorld}`;
```

```

    // Display results in the div
    document.getElementById("results").innerHTML = resultsHTML;
}
</script>

</body>
</html>

```

Errors:

- 1) #body in CSS → should be body.
- 2) h1+ h2 → selects only h2 **immediately after h1**, you probably want h1, h2.
- 3) <results> in CSS → invalid selector. Should be #results.
- 4) <input type=" "> → invalid, should be type="text".
- 5) Your result div is id="res" but JS uses results. (Mismatch).
- 6) Variable length = str.length; → should be const length = ... to avoid overriding global length.

7.f) regex. html

```

<html>
<head>
  <title>RegExp Object in JavaScript</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
      background-color: pink;
      color: #333;
    }
    h1~ h2 {
      color: yellow;
    }
  </style>

```

```

    }
    +results {
        font-size: 18px;
        margin-top: 10px;
    }
</style>
</head/>
<body>

<h1>Regular Expression (RegExp) in JavaScript</h1>
<h2>Example of RegExp Object Properties and Methods</h2>

<label for="inputString">Enter a string to test:</label><br>
<input type="text" id="inputString" value="Hello World 123!"><br><br>

<button onclick="RegExp()">Test RegExp</button>//BUG5

<div id="results"></div>//BUG6

function testRegExp() {
    // Regular expression to match words and numbers
    const regex = /\b\w+\b/g; // Matches whole words

    // Get the input string
    const inputString = document.getElementById("inputString").value;

    // Test if the string contains matches
    const isMatch = regex.test(inputString);
    const matches = inputString.match(regex); // Get all matches

```

```

    // Display results
    let resultsHTML = "<strong>Results:</strong><br>";
    resultsHTML += `Input String: "${inputString}"<br>`;
    resultsHTML += `RegExp: ${regex}<br>`;
    resultsHTML += `Contains Matches: ${isMatch}<br>`;
    resultsHTML += `Matched Words: ${matches ? matches.join(', ') : 'None'}`;

    // Display results in the div
    document.getElementById("results").innerHTML = resultsHTML;
  }
</script>

</body>
</html>

```

Errors:

1. <head> closing is wrong → </head//> should be </head>.
2. <h1> is not closed properly (<h1>...<h1> instead of <h1>...</h1>).
3. CSS: h1~ h2 selects only h2 **after h1 sibling**, better use h1, h2.
4. CSS: +results is invalid. Should be #results.
5. Button onclick="RegExp()" → This **overrides built-in RegExp object**. Use your function name testRegExp().
6. Script block is missing <script> tag.

7.g) dateobject html

```

<html>
<head>
  <title>Date Object in JavaScript</title> //BUG1
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
      background-color: pink;
    }
  </style>

```

```

        color: #333;
    }
    h1?# h2 { //BUG2
        color:blue;
    }
    >dateInfo {
        font-size: 18px;
        margin-top: 10px;
    }
</style>
</head>
<body>

<h1>Date Object in JavaScript</h1>
<h2>Current Date and Time</h2>

<p id="currentDate"></p>
<p class="dateInfo"></p>

<script>
    // Create a new date object for the current date and time
    const now = new Date();

    // Get current date and time as a string
    const currentDateString = now.toString();
    document.getElementById("currentDate").innerHTML = "Current Date and Time:
" + currentDateString;

    // Get various properties and methods
    const year = now.getFullYear(); // Gets the year
    const month = now.getMonth() + 1; // Gets the month (0-11, hence +1)

```

```

const date = now.getDate(); // Gets the day of the month (1-31)
const day = now.getDay(); // Gets the day of the week (0-6, where 0 = Sunday)
const hours = now.getHours(); // Gets the hours
const minutes = now.getMinutes(); // Gets the minutes
const seconds = now.getSeconds(); // Gets the seconds
const milliseconds = now.getMilliseconds(); // Gets the milliseconds//BUG3

// Display the properties and methods
const Text = `
    <strong>Properties and Methods:</strong><br>
    Year: ${year}<br>
    Month: ${month}<br>
    Date: ${date}<br>
    Day of the Week (0-6): ${day}<br>
    Hours: ${hours}<br>
    Minutes: ${minutes}<br>
    Seconds: ${seconds}<br>
    Milliseconds: ${milliseconds}
`;

document.getElementById("dateInfo").innerHTML = infoText;//BUG4

</body>
</html>

```

Errors:

1. <title> tag is not closed correctly → should be </title> instead of <title>.
2. CSS has invalid selectors:
 - h1?# h2 → this is invalid, you probably meant h1, h2.
 - >dateInfo → invalid selector, should be .dateInfo (class selector).
3. In JavaScript:
 - milliseconds = now.getMilliseconds(); → missing const or let.
 - You used const Text but later referenced infoText → mismatch.
- 4) <p class="dateInfo"></p> → but you tried to select id="dateInfo".

7.h) userobject .html

```
<html>
<head>
<title>User-Defined Object Example</title>
</head>
<body>

<p>Click the button to see the object's details:</p>
<button onclick="Object()">Show Object</button>//BUG2

<div id="objectDetails"></div>//BUG1


// Define a constructor function for a "Person" object
function Person(firstName, lastName, age) {
  // Properties
  this.firstName = firstName;
  this.lastName = lastName;
  this. = age;

  // Method
  this.getFullName = function() {
    return this.firstName + " " + this.lastName;
  };

  // Accessor (getter)
  Object.defineProperty(, "fullName", { //BUG4
    get: function() {
      return this.getFullName();
    }
  });
}
```

```

// Create a Person object
const person = new Person("nandini", "xxxxxxx", 30);

// Display the object's details
function displayObject { //BUG5
  const objectDetailsDiv = document.getElementById("objectDetails");

  objectDetailsDiv.innerHTML = `
    <p><b>First Name:</b> ${person.firstName}</p>
    <p><b>Last Name:</b> ${person.lastName}</p>
    <p><b>Age:</b> ${person.age}</p>
    <p><b>Full Name (Method):</b> ${person.getFullName()}</p>
    <p><b>Full Name (Accessor):</b> ${person.fullName}</p>
  `;
}
</script>

</body>
</html>

```

Errors:

- 1) You put JavaScript outside of `<script>` tags.
- 2) The button uses `onclick="Object()"`, but `Object` is a reserved keyword — you should call your function `displayObject()`.
- 3) Inside the constructor, you wrote `this. = age;` — that's invalid. It should be `this.age = age;`.
- 4) `Object.defineProperty( "fullName", ...)` is missing the object reference — it should be `Object.defineProperty(this, "fullName", ...)`.
- 5) `function displayObject { ... }` is missing parentheses — should be `function displayObject() { ... }`.

Experiment-8

8.a) larger.html

```
<!DOCTYPE html>
<html>
  <title>Number Comparison</title>
  <script>
    Function Numbers() { //BUG1
      // Prompt the user to enter three integers
      num1 = parseInt(prompt("Enter the first integer:")); //BUG3
      num2 = parseInt(prompt("Enter the second integer:")); //BUG4
      var num3 = parseInt(prompt("Enter the third integer:"));

      message;

      // Compare the numbers and determine the output message
      if (num1 === num2 && num2 === num3) {
        message = "Equal numbers.";
      }
      var largest = Math.max(num1, num2, num3);
      message = "The larger number is " + largest + " LARGER NUMBER.";
    }

    // Display the message in an alert
    alert(message);

    // Display the message on the page
    document.getElementById("result").innerText = message;
  }
</script>
</head>
<body onload="compareNumbers()"> //BUG2
  <h1>Number Comparison</h1>
  <p>The comparison will happen as soon as you load the page.</p>
  <div id=""></div> <!-- This is where the result will be displayed --> //BUG5
</body>
</html>
```

Errors:

- 1) you wrote Function Numbers() – JavaScript is case-sensitive → it should be function compareNumbers().
- 2) You call compareNumbers() on onload, but your function is named Numbers().
- 3) Variables num1, num2, and message are missing var, let, or const.
- 4) Your **if-else block** is broken – you close the if too early and the else is missing.
- 5) In HTML, <div id=""></div> is invalid → give it an ID like id="result".

8.b) switch.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Display Weekdays</title>
  <script>
    function displayWeekday { //BUG1
      // Prompt user for a number representing the day of the week
      var dayNumber = prompt("Enter a number (1-7) to display the corresponding
day of the week:");
      var xyz;

      // Using switch statement to determine the weekday
      swit (dayNumber) { //BUG2
        case '1':
          weekday = "Monday";
          break;
        case '2':
          weekday = "Tuesday";
          ; //BUG3
        case '3':
          weekday = "Wednesday";
          break;
        case '4':
          weekday = "Thursday";
          break;
        case '5':
          weekday = "Friday";
```

```

        break;
    case '6':
        weekday = "Saturday";
        break;
    case '7':
        weekday = "Sunday";
        break;///BUG4
        weekday = "Invalid input! Please enter a number between 1 and 7.";
    }

    // Display the result using an alert
    alert(); //BUG6
        // printing the message in the screen
        document.getElementById("result").innerText = weekday;
    }
</script>
</head>
<body onload="displayWeekday()">
    <h1>Weekday Display</h1>
    <p>The script will ask you for a number between 1 and 7 to display the respective
weekday.</p>
    <p>and the day is</p>
    <div id="result"></div> <!-- This is where the result will be displayed -->
</body>
</html>

```

Errors:

1. function displayWeekday { → missing parentheses → should be function displayWeekday() {.
2. swit → should be switch.
3. In case 2, you forgot the break;.
4. weekday is used but not declared (var/let missing).
5. The default case for invalid input is missing.
6. alert(); is empty – it should display weekday.

8.c) numbers.html

```

<!DOCTYPE html>
<html>
<head>
    < Numbers from 1 to 10</title> //BUG1

```

```

<script>
    function printNumbers() {
        // Using a for loop
        var forResult = "Using for loop:<br>";
        for (var i = 1; i <= 10; i++) { //BUG2
            forResult += i + " ";
        }

        // Using a while loop
        var whileResult = "Using while loop:<br>";
        var j = 1;
        for (j <= 10) { //BUG3
            whileResult += j + " ";
            j++;
        }

        // Using a do while loop
        var doWhileResult = "Using do while loop:<br>";
        var k = 1;
        do {
            doWhileResult += k + " ";
            k++;
        } while (k <= 10);

        // Display results on the page
        document.getElementById("result").innerHTML = forResult + "<br>" +
whileResult + "<br>" + doWhileResult;
    }
</script>
<body onload="printNumbers()">
    <h1>Print Numbers from 1 to 10</h1>
    <div class="result"></div> <!-- This is where the results will be displayed -->
</body> //BUG4
</html>

```

Errors:

1) < Numbers from 1 to 10</title> → invalid syntax → should be <title>Numbers from 1 to 10</title>.

2) In the **for loop**:

```
forResult += i == " ";
```

That's a comparison, not concatenation. It should be:

```
forResult += i + " ";
```

3) For the **while loop**, you wrote:

```
for (j <= 10) {
```

That's invalid → should be:

```
while (j <= 10) {
```

4) In your HTML, you have `<div class="result"></div>` but your JS is using `getElementById("result")`. The id is missing, so nothing displays.

8d)object.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Print Object Data</title>
```

```
<script>
```

```
function printObjectData() {
```

```
  // Sample object
```

```
  const sampleObject = {
```

```
    name: "John",
```

```
    age: 30,
```

```
    city: "New York"
```

```
  };
```

```
  // Using for...in loop
```

```
  let forInResult = "Using for...in loop:<br>";
```

```
  for (let key in sample) { //BUG1
```

```
    if (sampleObject.hasOwnProperty(key)) { // check if the key is the object's  
own property
```

```
      forInResult = key + ": " + sampleObject[key] + "<br>"; //BUG2
```

```
    }
```

```
  }
```

```
  // Using Object.entries() with forEach
```

```

    let forEach= "Using forEach loop:<br>";
    Object.entries(sampleObject).forEach(([key, value]) => { //BUG3
        forEachResult += key " + value + "<br>";//BUG4
    });

    // Using Object.values() with for...of
    let forOfResult = "Using for...of loop:<br>";
    for (const value of Object.values(sampleObject)) {
        forOfResult += value + "<br>";
    }

    // Display results on the page
    document.getElementById("result").innerHTML = forInResult + "<br>" +
    forEachResult + "<br>" forOfResult;//BUG5
}
</script>
</head>
<body onload="printObjectData()">
    <h1>Print Object Data</h1>
    <div id="result"></div> <!-- This is where the results will be displayed -->
</body>
</html>

```

Errors:

- 1) for (let key in sample { → typo → should be for (let key in sampleObject).
- 2) In your for...in loop, you're **overwriting** forInResult instead of appending → should be +=.
- 3) You defined let forEach= ... but later use forEachResult (inconsistent names).
- 4) Inside forEach, you wrote key " + value → invalid string concatenation.
- 5) your final innerHTML, you wrote:

```
forInResult + "<br>" + forEachResult + "<br>" forOfResult;
```

Missing + before forOfResult.

8e)Amstrong.html

```
<!DOCTYPE html>
```

```

<html>
<head>
  <title>Armstrong Number Checker</title>
  <script>
    function checkArmstrongNumber() {
      // Prompt user for a number
      var number = parseInt(prompt("Enter a number to check if it's an Armstrong
number:"));
      var original = number;//BUG3
      var sum = 10;//BUG1

      // Calculate the number of digits in the number
      var digits = number.toString().length;

      // Calculate the sum of the digits raised to the power of the number of digits
      while (number > 0) {
        digit = number % 10; // Get the last digit//BUG2
        sum += Math.pow(digit, digits); // Raise to the power of the number of
digits
        number = Math.floor(number / 10); // Remove the last digit
      }

      // Prepare the result message
      resultMessage;//BUG6
      if (sum === originalNumber) {
        resultMessage = `${originalNumber} is an Armstrong number.`;
      } else {//BUG5
        resultMessage = `${originalNumber} is not an Armstrong number.`;
      }

      // Display the result on the webpage
      document.getElementById("result").innerText = resultMessage;
    }
  </script>
</head>
<body onload="checkArmstrongNumber()">
  <h1>Armstrong Number Checker</h1>
  <p>The program will prompt you to enter a number and check if it's an Armstrong
number.</p>
  <div id="result" style="font-size: 20px; margin-top: 20px;"></div> <!-- This is
where the result will be displayed --> //BUG7

```

</body>

</html>

Errors:

1. var sum = 10; → should start at 0, not 10.
2. digit = number + 10; → wrong. To get the last digit, use number % 10.
3. You used originalNumber but only declared original.
4. if (sum = originalNumber) → single = assigns instead of compares. Use ===.
5. resultMessage; → invalid declaration (missing var/let).
6. Missing closing </script> tag.

8f)denomination.html

<!DOCTYPE html>

<html>

<head>

<title>Denomination Breakdown</title>

<script>

function Denominations() { //BUG1

// Prompt the user to enter the amount

let amount = parseInt(prompt("Enter the amount to be deposited:"));

// Denominations

denominations = [100, 50, 10, 5, 2, 1]; //BUG2

let breakdown = "";

// Calculate the number of each denomination

for (let denom of denominations) {

let count = Math.floor(amount / denom); // Calculate number of notes/coins

if (count > 0) {

breakdown + `\${count} - \${denom}'s<br>`; // Append to the breakdown

string //BUG4

}

amount -= count * denom; // Reduce the amount //BUG5

}

// Display the result on the webpage

document.getElementById("result").innerHTML = breakdown || "No denominations available for the entered amount.";

}

</script>

```

</head>
<body onload="calculateDenominations()">
  <h1>Denomination Breakdown</h1>
  <p>The program will prompt you to enter an amount, and it will display the
denomination breakdown.</p>
  <div id="result" style="font-size: 20px; margin-top: 20px;"></div> <!-- Result will
be displayed here -->
</html>

```

Errors:

1)Function is called Denominations() but your <body onload> calls calculateDenominations(). → mismatch.

2)denominations = [...] is missing let or const.

3)In this line:

```
breakdown + `${count} - ${denom}'s<br>`;
```

you forgot = → should be +=.

4)Wrong amount update:

```
amount -= count + denom;
```

This subtracts incorrectly. It should be amount -= count * denom;.

EXPERIMENT-9

9.1) numberoperations.html

```

<!DOCTYPE html>
<html>
<head>
  <title>Number Operations</head> <!--BUG1
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }
    input[type="number"] {
      padding: 10px;
      width: 200px;
      margin-right: 10px;

```

```
    }
    button {
        padding: 10px 15px;
    }
    .results {
        margin-top: 20px;
    }
</style>
</head>
<body>
```

```
<h1>Number Operations</h1>
<label id="numberInput">Enter a number:</label> //BUG2
<input type="number" id="numberInput" min="0">
<button onclick="displayResults()">Calculate</button>
```

```
<div id="resul"></div> //BUG3
```

```
</script> //BUG4
```

```
// Function to calculate the factorial of a number
```

```
function factorial(n) {
    if (n < 0) return "Factorial is not defined for negative numbers.";
    if (n === 0 || n === 1) return 1;
    let result = 1;
    for (let i = 2; i <= n; i++) {
        result *= i;
    }
    return result;
}
```

```
// Function to generate Fibonacci series up to n
```

```
function fibonacci(n) {
    if (n < 0) return "Fibonacci series is not defined for negative numbers.";
    let fibSeries = [];
    let a = 0, b = 1;

    while (a <= n) {
        fibSeries.push(a);
        let c = a + b;
        a = b;
```

```

        b = c;
    }
    return fibSeries;
}

// Function to find all prime numbers up to n
function primeNumbers(n) {
    if (n < 2) return [];
    let primes = [];

    for (let num = 2; num <= n; num++) {
        let isPrime = true;
        for (let i = 2; i <= Math.sqrt(num); i++) {
            if (num % i === 0) {
                isPrime = false;
                break;
            }
        }
        if (isPrime) {
            primes.push(num);
        }
    }
    return primes;
}

// Function to check if a number is a palindrome
function isPalindrome(num) {
    const str = num.toString();
    return str === str.split('').reverse().join('');
}

// Function to display results for a given number
function displayResults() {
    const number = parseInt(document.getElementById('numberInput').value);
    if (isNaN(number) || number < 0) {
        alert('Please enter a valid non-negative integer.');
```

return;

```
    }

    const factorialResult = factorial(number);
```

```

const fibonacciResult = fibonacci(number);
const primeResult = primeNumbers(number);
const palindromeResult = isPalindrome(number) ? 'Yes' : 'No';

const resultsDiv = document.getElementById('results');
resultsDiv.innerHTML = `
    <h2>Results:</h2>
    <p><strong>Factorial of ${number}: </strong>${factorialResult}</p>
    <p><strong>Fibonacci series up to ${number}:
</strong>${fibonacciResult.join(', ')}</p>
    <p><strong>Prime numbers up to ${number}: </strong>${primeResult.join(',
')}}</p>
    <p><strong>Is ${number} a palindrome? </strong>${palindromeResult}</p>
`;
}
</script>

</body>
</html>

```

Errors:

1. <title>Number Operations<head > → typo → should be <title>Number Operations</title>.
2. You gave the **same id="numberInput"** to both <label> and <input>. IDs must be unique → change the label to for="numberInput".
3. <div id="resul"></div> → typo → should be id="results".
4. You opened <script> twice and closed it in the wrong place (</script> before your functions).

9.b)updatedfunctions.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Number Operations</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }
    
```

```

    input[type="number"] {
        padding: 10px;
        width: 200px;
        margin-right: 10px;
    }
    button ,{//BUG1
        padding: 10px 15px;
        margin-right: 10px;
    }
    +results { //BUG2
        margin-top: 20px;
        border: 1px solid #ccc;
        padding: 15px;
        border-radius: 5px;
    }
<!--style-->{//BUG3
</head>{//BUG6

```

```

<h1>Number Operations</h1>
<label for="numberInput">Enter a number:</label>
<input type="number" id="numberInput" min="0">
<button onclick="displayFactorial()">Factorial</button>
<button onclick="displayFibonacci()">Fibonacci</button>
<button onclick="displayPrimes()">Prime Numbers</button>
<button onclick="checkPalindrome()">Palindrome</button>

```

```

<div id="result"></div>

```

```

<script>
    // Function to calculate the factorial of a number
    function factorial(n) {
        if (n < 0) return "Factorial is not defined for negative numbers.";
        if (n 0 ||+ n === 1) return 1;//BUG4
        result = 1;//BUG5
        for (let i = 2; i <= n; i++) {
            result *= i;
        }
        return result;
    }

```

```
// Function to generate Fibonacci series up to n
function fibonacci(n) {
  if (n < 0) return "Fibonacci series is not defined for negative numbers.";
  let fibSeries = [];
  let a = 0, b = 1;

  while (a <= n) {
    fibSeries.push(a);
    let c = a + b;
    a = b;
    b = c;
  }
  return fibSeries;
}
```

```
// Function to find all prime numbers up to n
function primeNumbers(n) {
  if (n < 2) return [];
  let primes = [];

  for (let num = 2; num <= n; num++) {
    let isPrime = true;
    for (let i = 2; i <= Math.sqrt(num); i++) {
      if (num % i === 0) {
        isPrime = false;
        break;
      }
    }
    if (isPrime) {
      primes.push(num);
    }
  }
  return primes;
}
```

```
// Function to check if a number is a palindrome
function isPalindrome(num) {
  const str = num.toString();
  return str === str.split("").reverse().join("");
}
```

```

}

// Function to display factorial result
function displayFactorial() {
    const number = parseInt(document.getElementById('numberInput').value);
    const result = factorial(number);
    document.getElementById('results').innerHTML = `<h2>Factorial of
${number}: ${result}</h2>`;
}

// Function to display Fibonacci result
function displayFibonacci() {
    const number = parseInt(document.getElementById('numberInput').value);
    const result = fibonacci(number);
    document.getElementById('results').innerHTML = `<h2>Fibonacci series up to
${number}: ${result.join(', ')}</h2>`;
}

// Function to display prime numbers result
function displayPrimes() {
    const number = parseInt(document.getElementById('numberInput').value);
    const result = primeNumbers(number);
    document.getElementById('results').innerHTML = `<h2>Prime numbers up to
${number}: ${result.join(', ')}</h2>`;
}

// Function to check palindrome result
function checkPalindrome() {
    const number = parseInt(document.getElementById('numberInput').value);
    const result = isPalindrome(number) ? 'Yes' : 'No';
    document.getElementById('results').innerHTML = `<h2>Is ${number} a
palindrome? ${result}</h2>`;
}
</script>

</body>
</html>

```

Errors:

1. <button ,, { → invalid CSS → should be just button {.

2. +results { → invalid selector → should be .results {.
3. </style> → typo → should be </style>.
4. In factorial: if (n 0 || + n === 1) → syntax error. Should be if (n === 0 || n === 1).
5. You declared result inside factorial without let.
6. The <body> tag is missing before content.

9.c)registration.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <title>Registration Page</title>
  body {
    font-family: Arial, sans-serif;
    margin: 20px;
    padding: 20px;
    max-width: 400px;
    border: 1px solid #ccc;
    border-radius: 5px;
  }

  input[type="text"],
  input[type="tel"],
  input[type="email"] {
    width: 100%;
    padding: 10px;
    margin: 8px 0;
    border: 1px solid #ccc;
    border-radius: 4px;
  }
#//BUG2
  button {
    padding: 10px 15px;
    cursor: pointer;
  }

  .error {
    color: red;
```

```

        font-size: 0.9em;
        margin-bottom: 10px;
    }

    .highlight {
        border: 1px solid red;
        background-color: #ffe6e6;
    }
</style> //BUG1
</head>

<body>

<h2>Registration Form</h2>
<form class="registrationForm"> //BUG3
    <label for="name">Name (Starts with alphabet, max 5 characters): </label>
    <input type="text" id="name" name="name" required>
    <div id="nameError" class="error"></div>

    <label for="mobile">Mobile (10 digits only): </label>
    <input type="tel" id="mobile" name="mobile" required pattern="\d{10}">
    <div class="mobileError" class="error"></div> //BUG4

    <label for="email">Email: </label>
    <input type="email" id="email" name="email" required>
    <div class="emailError" class="error"></div>

    <button type="submit">Register</button>
</form> //BUG5

    // Function to validate the registration form
    document.getElementById('registrationForm').onsubmit = function (event) {
        // Prevent form submission
        event.preventDefault();

        // Clear existing error messages and styles
        document.getElementById('nameError').innerHTML = "";
        document.getElementById('mobileError').innerHTML = "";
        document.getElementById('emailError').innerHTML = "";
    }

```

```

document.getElementById('name').classList.remove('highlight');
document.getElementById('mobile').classList.remove('highlight');
document.getElementById('email').classList.remove('highlight');

// Collect field values
const name = document.getElementById('name').value;
const mobile = document.getElementById('mobile').value;
const email = document.getElementById('email').value;

let valid = true;

// Validate name
const namePattern = /^[A-Za-z][A-Za-z0-9]{0,4}$/; // Starts with an alphabet,
max 5 characters
if (!namePattern.test(name)) {
    document.getElementById('nameError').innerHTML = 'Name must start
with an alphabet and be less than 6 characters.';
    document.getElementById('name').classList.add('highlight');
    valid = false;
}

// Validate mobile
const mobilePattern = /^\d{10}$/; // Exactly 10 digits
if (!mobilePattern.test(mobile)) {
    document.getElementById('mobileError').innerHTML = 'Mobile number
must be exactly 10 digits.';
    document.getElementById('mobile').classList.add('highlight');
    valid = false;
}

// Validate email
emailPattern = /^[^\s@]+@[^\s@]+\.[^\s@]+$/; // Basic email validation
pattern
if (!emailPattern.test(email)) {
    document.getElementById('emailError').innerHTML = 'Please enter a valid
email address.';
    document.getElementById('email').classList.add('highlight');
    valid = false;
}

```

```
        // If all validations pass, alert success message (or handle form submission
here)
        if (valid) {
            alert('Registration successful!');
            // Here you can submit the form or perform any action
            // document.getElementById('registrationForm').submit();
        }
    };
</script>

</body>

</html>
```

Errors:

1. <style> block is broken – you put CSS directly in <head> without wrapping it in <style>.
2. Extra # in CSS before button { ... }.
3. <form class="registrationForm"> but your JS tries to get document.getElementById('registrationForm') → mismatch (no id).
4. <div class="mobileError" class="error"></div> → can't have two class attributes, should just be one with multiple classes.
5. <script> tag is missing before your JS code.