**Q.01) Write HTML Code to Create Resume/CV.**

**CODE:**

<html>

<head>

    <title>Himanshu- Resume</title>

</head>

<body>

    <header style="text-align: center; background-color: #333; color: white; padding: 10px;">

        <h1>Himanshu</h1>

        <p>Computer Science Sophomore</p>

    </header>

<div style="width: 80%; margin: auto; padding: 20px;">

        <section>

            <h2 style="border-bottom: 2px solid #333;">Contact Information</h2>

            <p>Email: himanshu@outlook.com</p>

            <p>Phone: (123) 456-7890</p>

            <p>LinkedIn: linkedin.com/in/himanshu</p>

            <p>GitHub : github.com/himanshu </p>

        </section>

  <section>

            <h2 style="border-bottom: 2px solid #333;">Experience</h2>

            <div>

                <h3>Intern - XYZ Tech</h3>

                <p>June 2023 - August 2023</p>

                <p>Worked on developing web applications and improving user interfaces.</p>

            </div>

            <div>

                <h3>Project Lead - University Project</h3>

                <p>January 2024 - Present</p>

                <p>Leading a team to develop a mobile application for campus events.</p>

            </div>

        </section>

<section>

            <h2 style="border-bottom: 2px solid #333;">Education</h2>

            <div>

                <h3>Bachelor of Science in Computer Science</h3>

                <p>University Name</p>

                <p>GPA: X.X/X.X</p>

                <p>Expected Graduation: May 2026</p>

            </div>

        </section>

        <section>

            <h2 style="border-bottom: 2px solid #333;">Skills</h2>

            <ul>

            <li>Programming Languages: C, C++, Python </li>

                <li>Web Development: HTML, CSS, JavaScript, MarkDown </li>

                <li>Tools: NeoVim, Git, Shell</li>

            </ul>

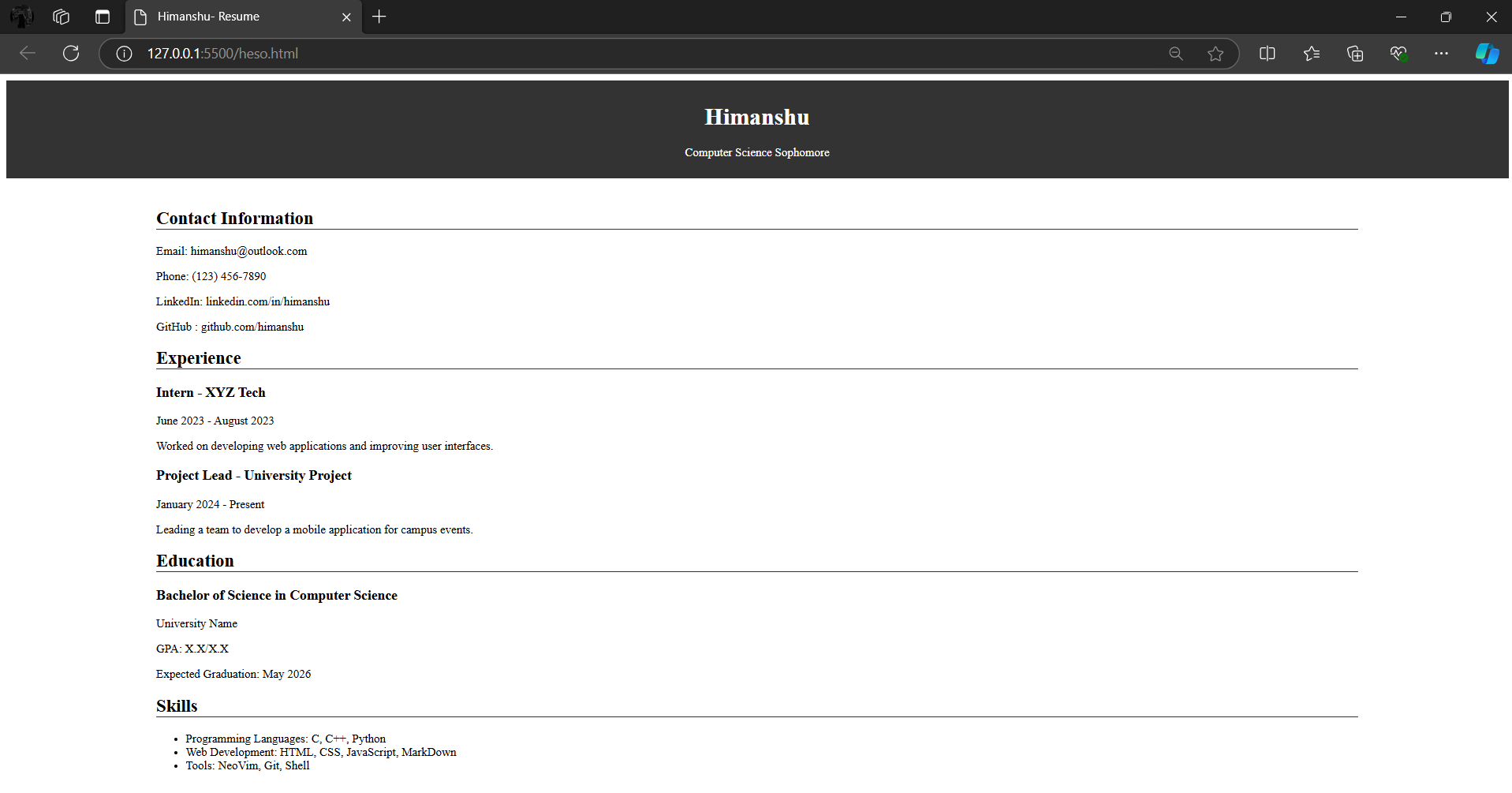
        </section>

    </div>

</body>

</html>

**Output:**



**Q.02) Write a Html Code to create Marksheet using <table> in Html.**

**CODE :**

<html>

<head>

<title>MarkSheet </title>

<style>

th,td{

border: 2px solid black ;

text-align: center;

}

body{

font-family:'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif ;

}

</style>

</head>

<body>

<hr width="75%" size="7px" color="black">

<h1 align="center">Graphic Era Hill University</h1>

<p align="center">

<b>Society Area, Clement Town Dehradun-248002, Uttarakhand (India)</b>

<br> <b>www.gehu.in</b>

</p>

<p align="center">

<h3 align="center">STATEMENT OF MARKS (PROVISIONAL)</h3>

<h3 align="center">BACHELOR OF COMPUTER APPLICATIONS - I SEMESTER EXAMINATION (2023-2024)</h3>

</p>

<p align="center"><b>Name of Student:</b> Name &nbsp; &nbsp; &nbsp; <b>Father's Name:</b> Father's Name &nbsp; &nbsp; &nbsp; <b>Enrollment No:</b> 102345678 &nbsp; &nbsp; &nbsp; <b>Roll No: </b> 350289 </p>

<table align="center" border="1" style="border-collapse:collapse; border: 2px solid black" >

<tr>

<th> </th>

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

<th rowspan="2">CREDITS</th>

<th colspan="2">MID SEMESTER EXAM</th>

<th colspan="2">SESSIONAL MARKS</th>

<th colspan="2">END SEMESTER EXAM</th>

<th rowspan="2">TOTAL MARKS OBTAINED</th>

<th rowspan="2">GRADE</th>

<th rowspan="2">GRADE POINT</th>

</tr>

<tr>

<th> </th>

<th>CODE </th>

<th>NAME</th>

<th>Max.</th>

<th>Obt</th>

<th>Max.</th>

<th>Obt</th>

<th>Max.</th>

<th>Obt</th>

</tr>

<tr>

<th rowspan="5">Theory</th>

<td>TBC101</td>

<td>Computational Thinking and Foundation of IT </td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>TBC102</td>

<td>Foundation of Computer Programming</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>TBC103</td>

<td>Mathematical Foundation of Computer Science</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>THU101</td>

<td>Professional Communication</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>GE101</td>

<td>Generative Elective</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<th rowspan="4">Practical</th>

<td>PBC101</td>

<td>C Programming Lab</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>PBC102</td>

<td>Digital Productivity tool for Modern Workplaces</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>SBC101</td>

<td>MOOCS Based Seminar on Environmental Science</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

<tr>

<td>GP101</td>

<td>General Proficiency</td>

<td>x</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>xx</td>

<td>B+</td>

<td>x</td>

</tr>

</table>

<p>

<b>Result : </b> Pass <br>

<b>Total No. of Credits registered :</b> 22 <br>

<b>Total No. of Credits earned :</b> 22 <br>

<b>SGPA : </b> X.xx

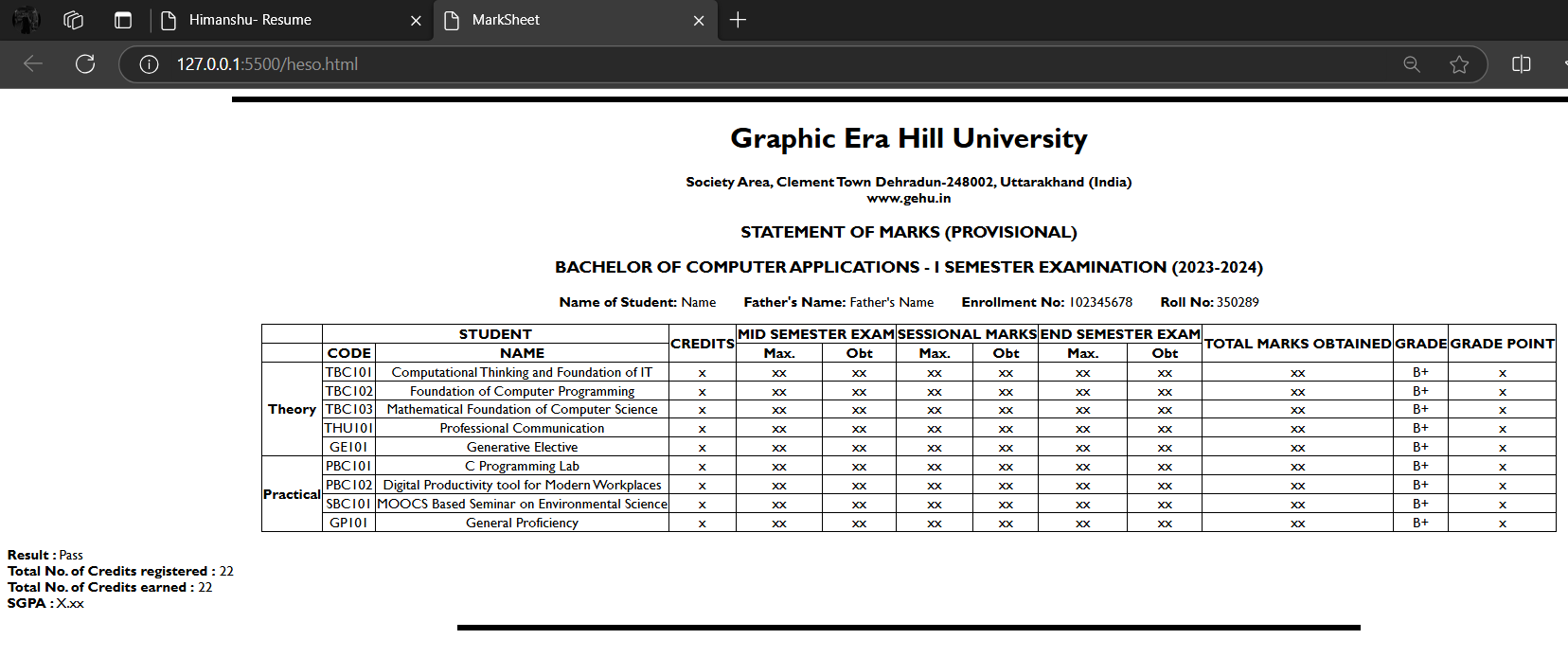
</p>

<hr width="50%" size="7px" color="black">

</body>

</html>

**Output:**



**Q.03) Write Html Code to create a Student Registration form using form control.**

**CODE:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Student Registration Form</title>

</head>

<body>

<h1>Student Registration Form</h1>

<hr>

<form action="/submit-registration" method="post">

<fieldset>

<legend>Personal Information</legend>

<label for="first-name">First Name:</label>

<input type="text" id="first-name" name="first-name" required>

<br><br>

<label for="last-name">Last Name:</label>

<input type="text" id="last-name" name="last-name" required>

<br><br>

<label for="dob">Date of Birth:</label>

<input type="date" id="dob" name="dob" required>

<br><br>

<label>Gender:</label>

<input type="radio" id="male" name="gender" value="male" required>

<label for="male">Male</label>

<input type="radio" id="female" name="gender" value="female" required>

<label for="female">Female</label>

<input type="radio" id="other" name="gender" value="other" required>

<label for="other">Other</label>

</fieldset>

<hr>

<fieldset>

<legend>Contact Details</legend>

<label for="email">Email Address:</label>

<input type="email" id="email" name="email" required>

<br><br>

<label for="phone">Phone Number:</label>

<input type="tel" id="phone" name="phone">

<br><br>

<label for="address">Home Address:</label>

<br>

<textarea id="address" name="address" required></textarea>

</fieldset>

<hr>

<fieldset>

<legend>Academic Details</legend>

<label for="student-id">Student ID:</label>

<input type="text" id="student-id" name="student-id" required>

<br><br>

<label for="department">Department:</label>

<select id="department" name="department" required>

<option value="">Select Department</option>

<option value="cs">Computer Science</option>

<option value="ee">Electrical Engineering</option>

<option value="me">Mechanical Engineering</option>

<option value="ce">Civil Engineering</option>

</select>

<br><br>

<label for="year">Year of Study:</label>

<select id="year" name="year" required>

<option value="">Select Year</option>

<option value="1">First Year</option>

<option value="2">Second Year</option>

<option value="3">Third Year</option>

<option value="4">Fourth Year</option>

</select>

</fieldset>

<hr>

<fieldset>

<legend>Preferences</legend>

<label for="language">Preferred Programming Language:</label>

<select id="language" name="language" required>

<option value="">Select Language</option>

<option value="python">Python</option>

<option value="java">Java</option>

<option value="cpp">C++</option>

<option value="javascript">JavaScript</option>

</select>

<br><br>

<label for="interests">Interests:</label>

<br>

<textarea id="interests" name="interests"></textarea>

</fieldset>

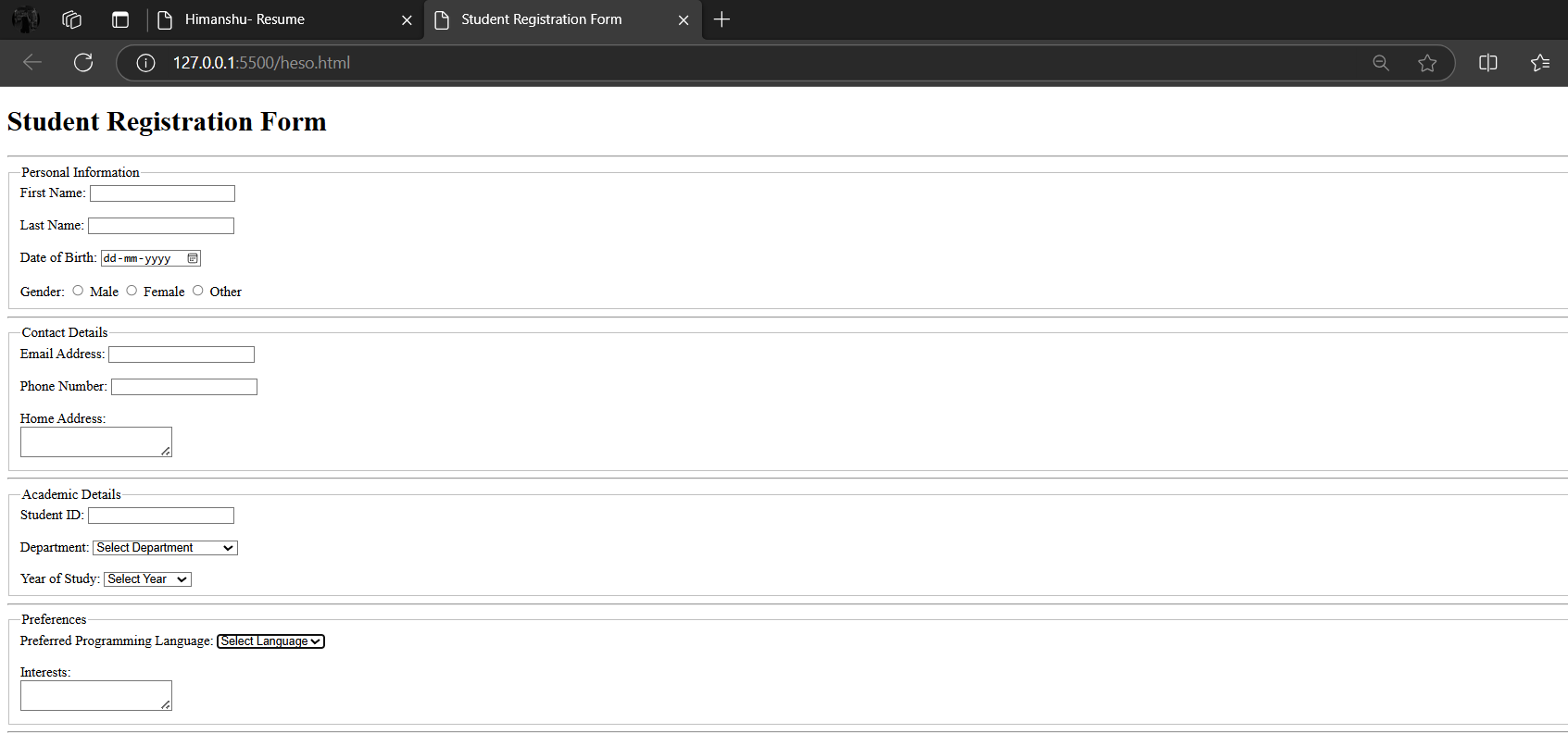
<hr>

</form>

</body>

</html>

**Output:**

****

**Q.04 Write an HTML code to show different type of list and use various list attributes of CSS.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Different Types of Bullet Points</title>

</head>

<body>

<h2>Unordered List with Circle Bullets</h2>

<ul class="circle-bullets">

<li>Apple</li>

<li>Banana</li>

<li>Cherry</li>

<li>Date</li>

</ul>

<h2>Unordered List with Disc Bullets</h2>

<ul class="disc-bullets">

<li>Dog</li>

<li>Cat</li>

<li>Rabbit</li>

<li>Hamster</li>

</ul>

<h2>Unordered List with Square Bullets</h2>

<ul class="square-bullets">

<li>Red</li>

<li>Blue</li>

<li>Green</li>

<li>Yellow</li>

</ul>

<h2>Ordered List with Lowercase Letters</h2>

<ol class="lower-alpha">

<li>First Item</li>

<li>Second Item</li>

<li>Third Item</li>

<li>Fourth Item</li>

</ol>

<h2>Ordered List with Lowercase Roman Numerals</h2>

<ol class="lower-roman">

<li>Item One</li>

<li>Item Two</li>

<li>Item Three</li>

<li>Item Four</li>

</ol>

<h2>Ordered List with Uppercase Letters</h2>

<ol class="upper-alpha">

<li>Item A</li>

<li>Item B</li>

<li>Item C</li>

<li>Item D</li>

</ol>

<h2>Ordered List with Decimal Numbers</h2>

<ol class="decimal-numbers">

<li>Step One</li>

<li>Step Two</li>

<li>Step Three</li>

<li>Step Four</li>

</ol>

<h2>Definition List</h2>

<dl class="custom-definition">

<dt>HTML</dt>

<dd>HyperText Markup Language</dd>

<dt>CSS</dt>

<dd>Cascading Style Sheets</dd>

<dt>JavaScript</dt>

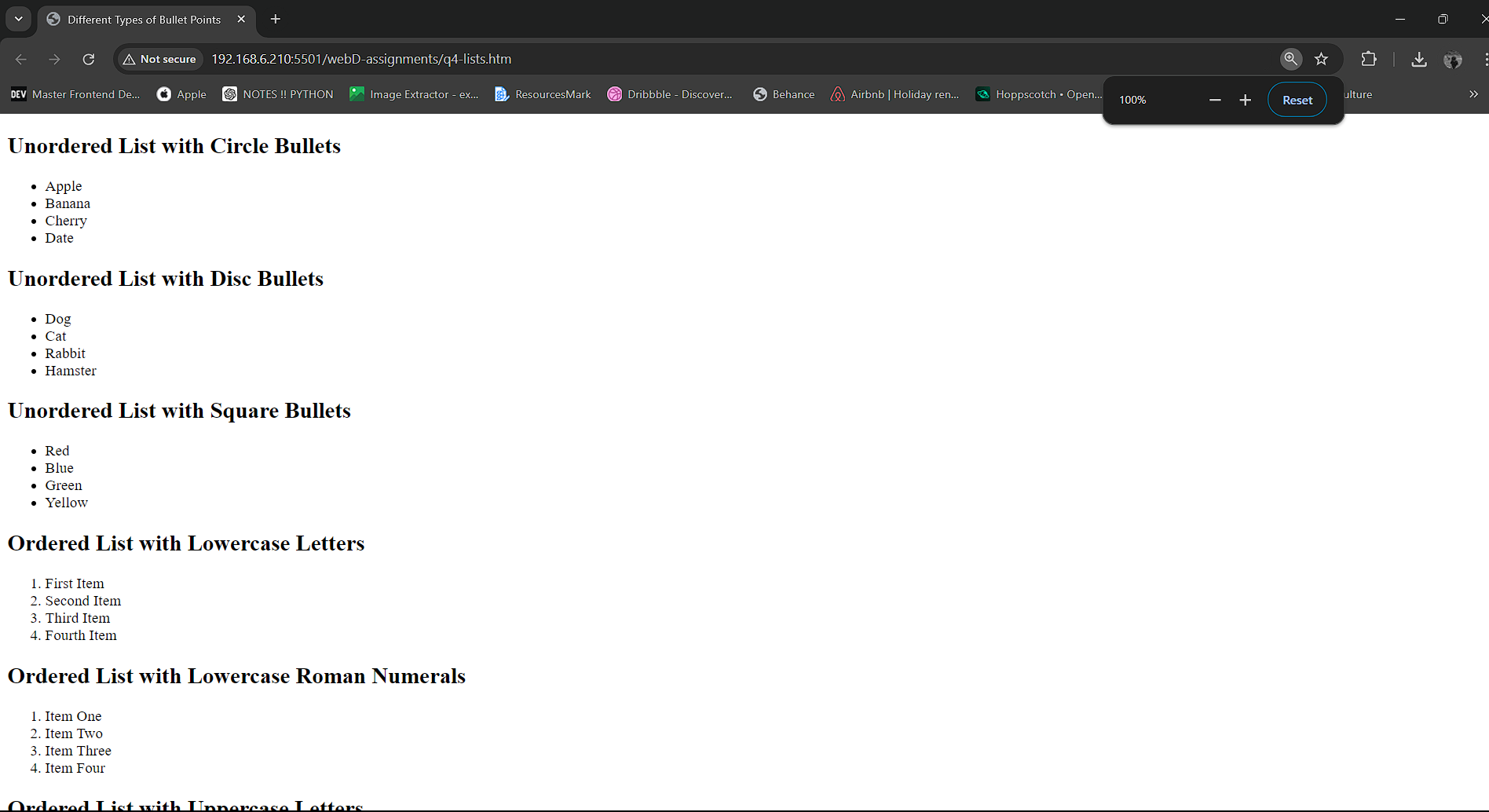
<dd>A programming language for the web</dd>

</dl>

</body>

</html>

**OUTPUT 04:**

****

**Q.05: Write an HTML code to demonstrate audio and video tag of HTML files.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>HTML Audio and Video Demonstration</title>

</head>

<body>

<h1>HTML Audio and Video Tags</h1>

<!-- Audio Section -->

<h2>Audio Example</h2>

<p>Here is an audio file playing a song:</p>

<audio controls>

<source src="audio-file.mp3" type="audio/mpeg">

<source src="audio-file.ogg" type="audio/ogg">

Your browser does not support the audio element.

</audio>

<!-- Video Section -->

<h2>Video Example</h2>

<p>Here is a video file with controls:</p>

<video width="600" height="340" controls>

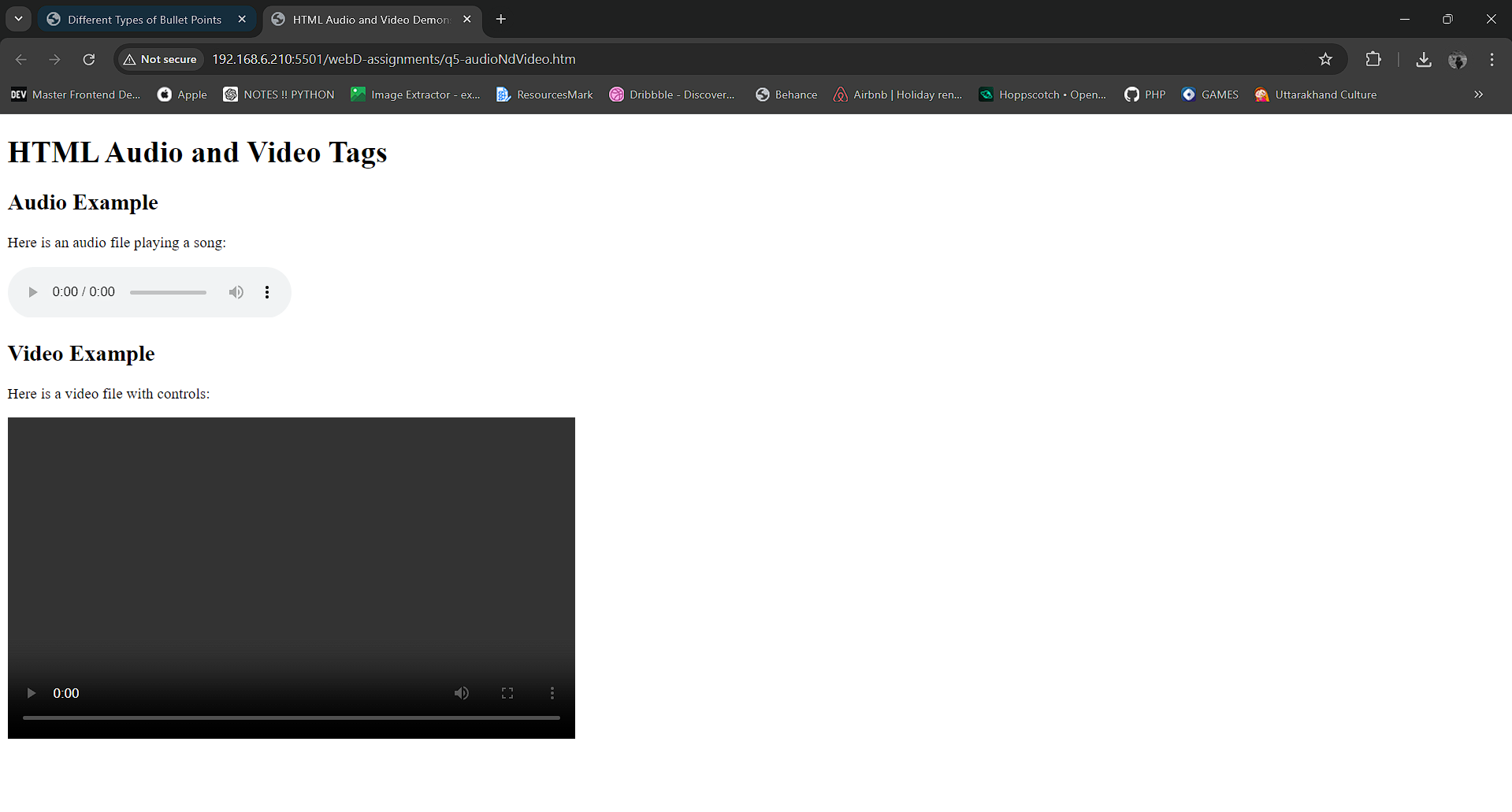
<source src="video-file.mp4" type="video/mp4">

</video>

</body>

</html>

**Output:**

****

**Q.07 Write HTML code and demonstrate the implementation of internal, external & inline css  
  
CODE:**<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>CSS Demonstration</title>

<!-- External CSS (simulating the external CSS with an internal style block) -->

<link rel="stylesheet" href="styles.css">

<!-- Internal CSS -->

<style>

.internal-style {

background-color: lightblue;

padding: 20px;

text-align: center;

font-size: 18px;

color: darkblue;

}

</style>

</head>

<body>

<h1>Internal, External & Inline CSS Demo</h1>

<!-- Inline CSS -->

<div style="background-color: lightgreen; padding: 20px; text-align: center; font-size: 18px;">

This div is styled using Inline CSS

</div>

<!-- Internal CSS (applied to this div) -->

<div class="internal-style">

This div is styled using Internal CSS

</div>

<!-- External CSS (you'll link it to an actual external stylesheet) -->

<div class="external-style">

This div is styled using External CSS (simulated in the example)

</div>

</body>

</html>

**EXTERNAL CSS :**/\* External CSS \*/

.external-style {

background-color: lightcoral;

padding: 20px;

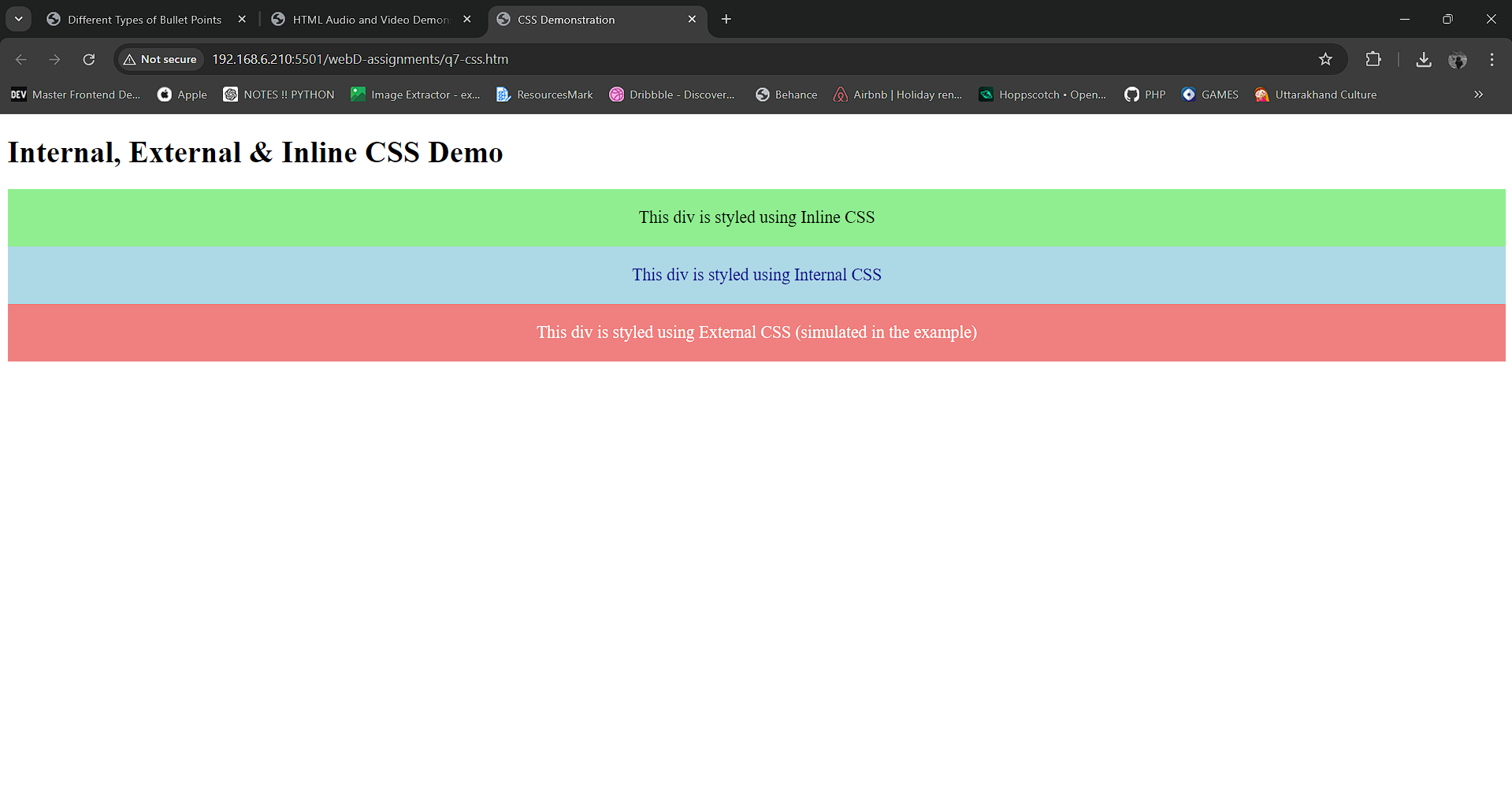
text-align: center;

font-size: 18px;

color: white;

}

**OUTPUT 07 :**



**Q.08 : Design sample home page for Graphic Era University using frame and demonstrate internal hyperlinking and external hyperlinking.**

**CODE :**

**HTML :**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Graphic Era Hill University</title>

<link rel="stylesheet" href="style-gehu-erp.css">

</head>

<body>

<form action="#">

<div class="form\_\_head">

<img src="./showClientLoginPageLogo.png" alt="college-logo">

</div>

<div class="form\_\_body">

<input type="text" placeholder="user id" />

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

<button>Log In</button>

<div class="links">

<p> <a href="">Forgot password?</a></p>

<p> <a href="">Forgot ID? </a> </p>

</div>

<hr>

<p class="cyborg">Powered by <a href=""> Cyborg IT Service(P)Ltd. </a></p>

</div>

</form>

</body>

</html>

**CSS :**

\*{

border: 0 ;

padding: 0 ;

margin : 0 ;

box-sizing: border-box ;

}

body{

background-image: url("./gehuHome.jpg");

background-size: cover ;

background-repeat: no-repeat;

}

form{

position: absolute;

top : 70px ;

right: 60px ;

height: 80vh ;

width: 21vw ;

border: 2px solid grey ;

border-radius: 7px;

}

.form\_\_head{

height: 20% ;

width: auto ;

background-color: #E6F4F7 ;

display: flex;

align-items: center ;

justify-content: center;

border-top-left-radius: 7px ;

border-top-right-radius: 7px ;

}

.form\_\_head img{

height: 55% ;

width: auto ;

align-items: center;

background-position: center;

background-size: contain;

}

.form\_\_body{

height: 80% ;

display: flex ;

flex-direction: column ;

flex-wrap: wrap;

justify-content: space-evenly;

align-content: space-evenly;

background-color: rgba(133, 133, 133, 0.527) ;

color: black ;

}

input, button{

height: 3rem ;

width: 75% ;

/\* opacity: 0.7 ; \*/

padding: 15px ;

/\* text-align: center; \*/

border: .6px solid rgb(128, 128, 128) ;

border-radius: 2.5px;

}

button{

background-color: #03A9F4 ;

border-radius: 1.5px;

}

.links{

display: flex ;

flex-direction: column ;

gap: 7px ;

text-align: right;

}

.links a{

color: white;

text-decoration: none;

&:hover{

text-decoration: underline;

}

}

hr{

height: 1px ;

background-color: white ;

}

.cyborg{

color: white ;

}

.cyborg a{

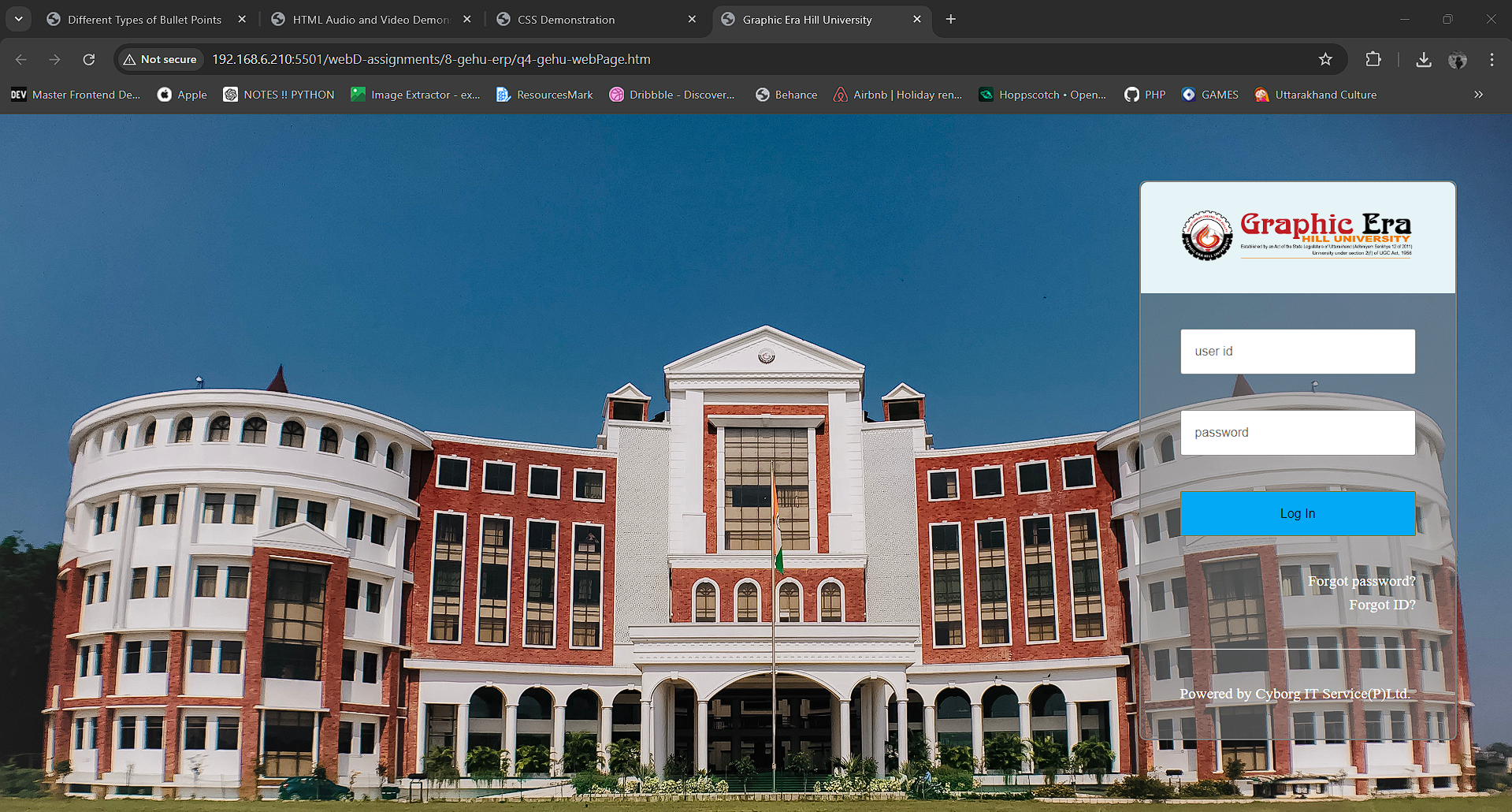
text-decoration: none ;

color: white ;

&:hover{

text-decoration: underline;

}}

**OUTPUT 08 :**

**Q.09 : Write HTML code to design a web page similar to Facebook front page.**

**CODE :**

**HTML :**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>fb login</title>

<link rel="stylesheet" href="style-fb-login.css">

</head>

<body>

<main>

<!-- <h1>facebook</h1> -->

<div class="logo" style="background-image: url('./fb-logo.svg');"> </div>

<div class="box">

<p>Log in to Facebook</p>

<input type="text" placeholder="Email address or phone number">

<input type="password" placeholder="Password">

<button>Log in</button>

<div class="creds">

<a href="#">Forgotten account?</a>

<a href="#">Sign up for Facebook</a>

</div>

</div>

</main>

<footer>

</footer>

</body>

</html>

**CSS :**

/\* including font \*/

@import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=swap');

\*{

margin: 0 ;

padding: 0 ;

border: 0 ;

font-family: 'roboto' ;

box-sizing: border-box ;

}

body{

background-color: #ffffff ;

}

main{

height: 70vh ;

width: 100vw ;

background-color: #F2F4F7 ;

display: flex;

flex-direction: column;

flex-wrap: wrap;

justify-content: center;

}

.logo{

margin: 0 auto ;

height: 100px ;

width: 320px;

background-size: contain;

background-repeat: no-repeat;

}

.box{

margin: 0 auto ;

height: 350px ;

width: 400px ;

border-radius: 12px ;

background-color: #ffffff ;

box-shadow: 0 0 15px 5px rgba(174, 174, 174, 0.395) ;

font-size: 1.2rem;

display: flex;

flex-direction: column;

flex-wrap: wrap ;

justify-content: space-evenly;

align-items: center;

}

button{

color: white ;

font-size: 1.5rem;

font-weight: 700;

background-color: #0037ff ;

border-radius: 1.5px;

}

input, button{

height: 3rem ;

width: 90% ;

padding: 15px ;

font-size: 1.1rem;

border: .6px solid rgba(128, 128, 128, 0.511) ;

border-radius: 2.5px;

opacity: .8;

}

.creds{

width: 100%;

text-align: center;

font-size: .8rem ;

}

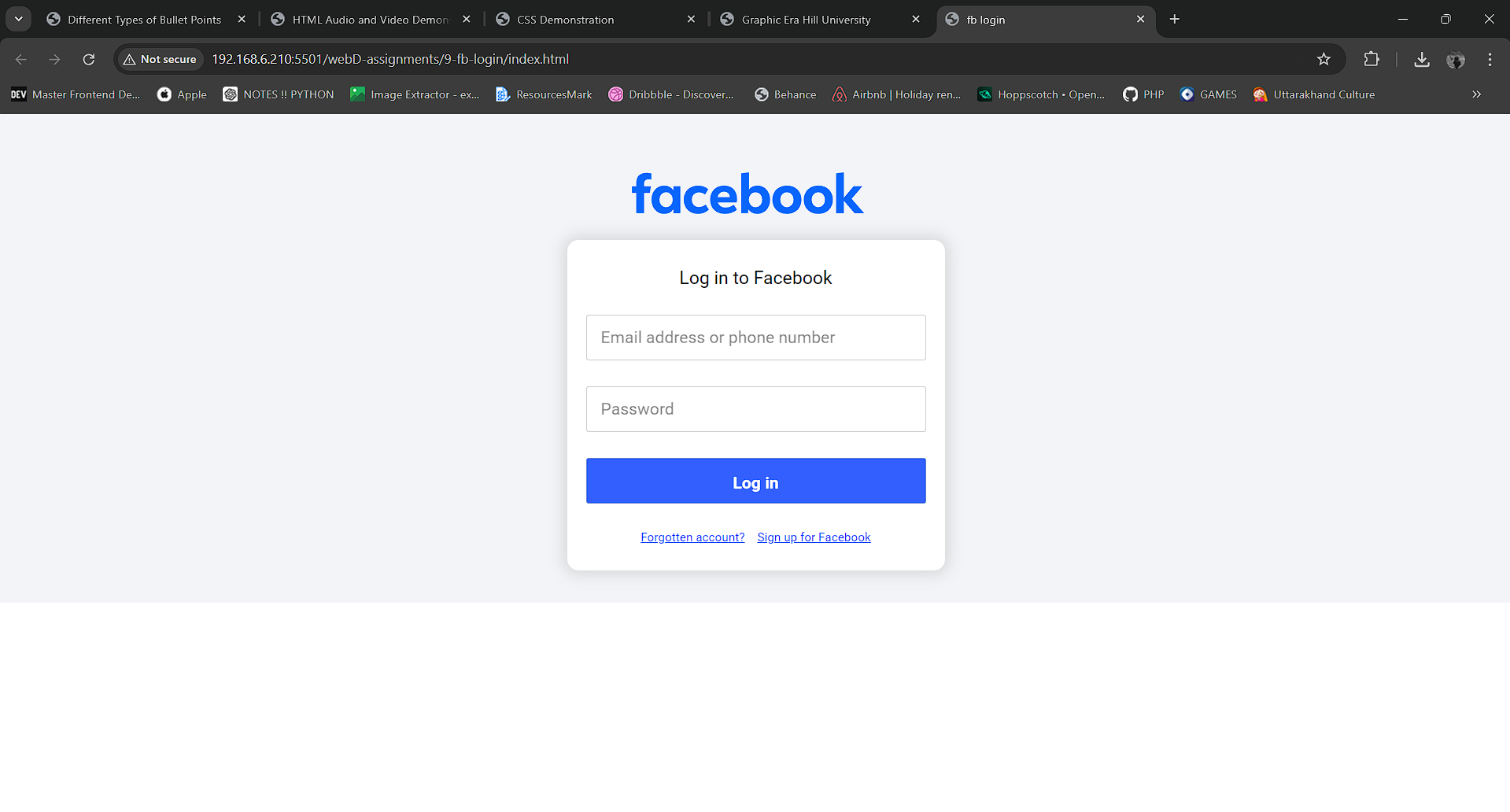
.creds a{

margin: 0 5px ;

color: #0037ff ;

}

**OUTPUT 09 :**

****

**Q10. Write a HTML code to demonstrate the implementation of internal JavaScript and external JavaScript**

**CODE :**

**HTML :**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>JS Implementation</title>

</head>

<body>

<script>

// internal javaScript

console.log("this log is created by internal JS. ")

</script>

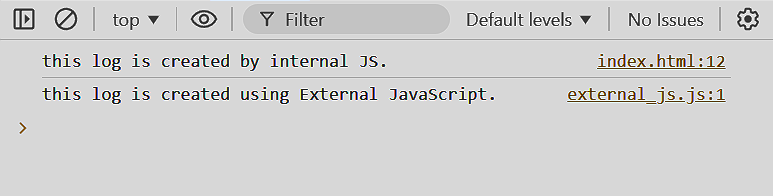
<script src="external\_js.js"></script>

</body>

</html>

**JS: (External JS) :**

console.log("this log is created using External JavaScript.")

**OUTPUT 10 :**

**Q11. Write Java Script Code to calculate the % obtained by the Student in Subject & assign grades as : A when % is 80-90, B when % is 70-80, C when % is 60-70 and D when % is below 60.**

**CODE:**

**JAVA SCRIPT:**

var marks = {

hindi : 80,

maths : 70,

english : 20,

science : 30,

history : 40,

}

var percentage = ( marks.hindi + marks.maths + marks.english + marks.science + marks.history) / 5 ;

var grade ;

if(percentage < 90 && percentage > 80){

grade = 'A' ;

}else if(percentage < 80 && percentage > 70){

grade = 'B' ;

}else if(percentage < 70 && percentage > 60){

grade = 'C' ;

}else if(percentage < 60){

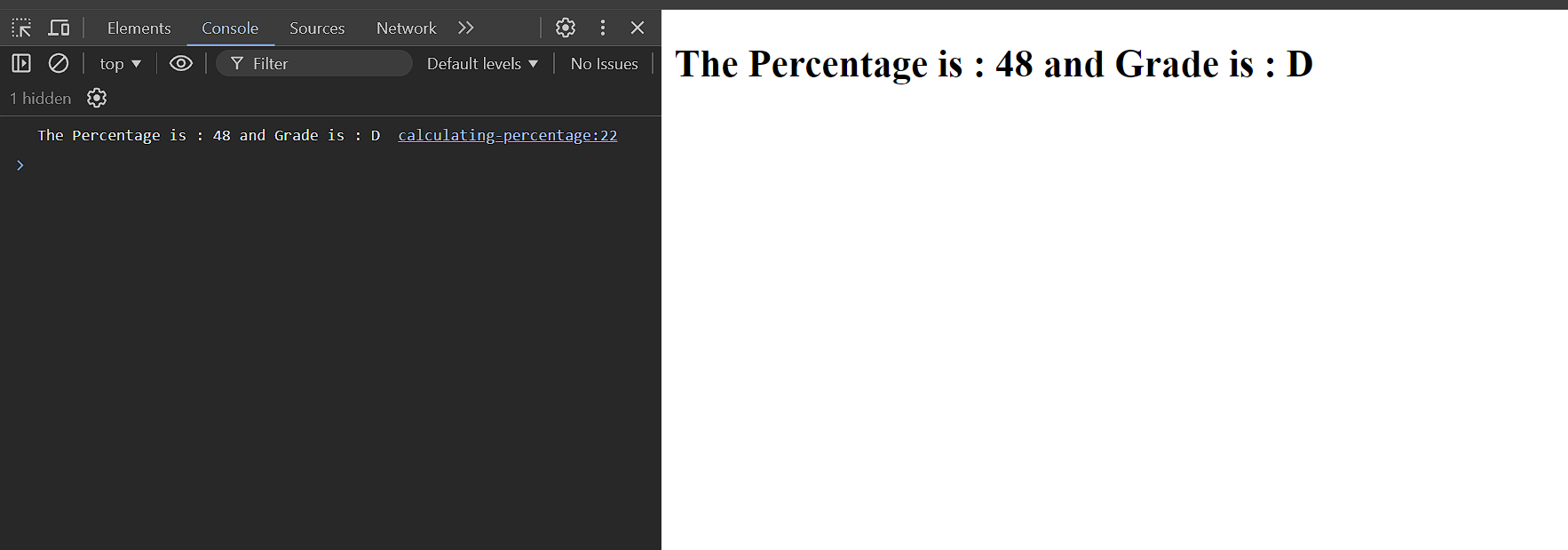
grade = 'D' ;

}

console.log(`The Percentage is : ${percentage} and Grade is : ${grade}`) ;

document.write(`<h2> The Percentage is : ${percentage} and Grade is : ${grade} </h2>`) ;

**OUTPUT – 11 :**

****

**Q12 : Write a JS Code to print Fibonacci Sequence.**

**CODE:**

let n = 10; // Number of terms you want in the Fibonacci sequence

let a = 0, b = 1, nextTerm;

document.write(" <h2> Fibonacci Sequence: </h2> ");

for (let i = 1; i <= n; i++) {

document.write(`${a}, `)

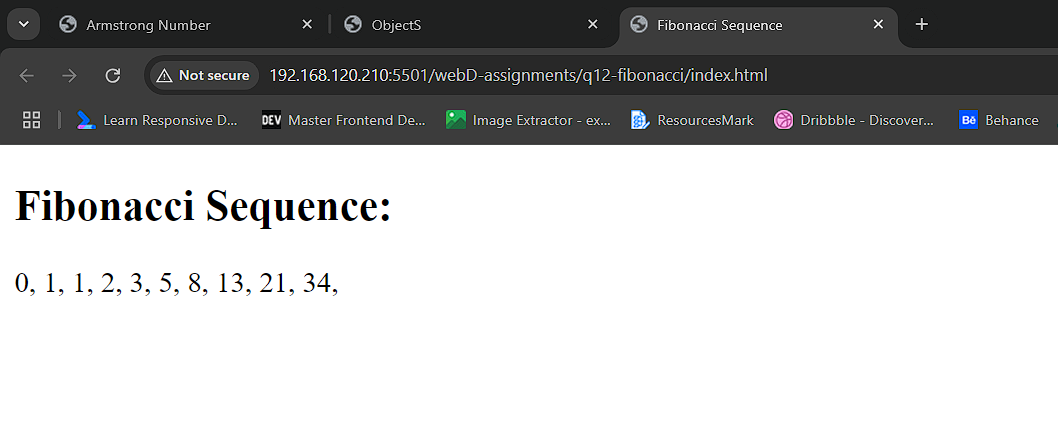
nextTerm = a + b;

a = b;

b = nextTerm;

}

**OUTPUT - 12:**

****

**Q13: Write JS Code to check whether the given no. is Armstrong or not.**

**CODE :**

let n = 10; // Number of terms you want in the Fibonacci sequence

let a = 0, b = 1, nextTerm;

document.write(" <h2> Fibonacci Sequence: </h2> ");

for (let i = 1; i <= n; i++) {

document.write(`${a}, `)

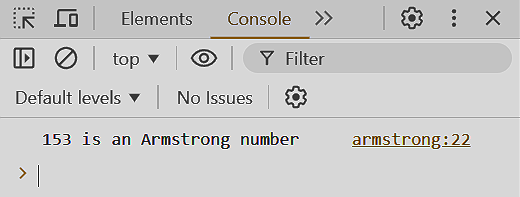
nextTerm = a + b;

a = b;

b = nextTerm;

}

**OUTPUT – 13 :**

****

**Q14: Write a JavaScript to create Student object and demonstrate all the three ways of creating object**

**CODE:**

// Object literal method

const student1 = {

name: "John",

age: 20,

course: "Computer Science",

details: function() {

return `${this.name} is ${this.age} years old and studies ${this.course}.`;

}

};

console.log(student1.details());

// Using the Object() constructor

const student2 = new Object();

student2.name = "Jane";

student2.age = 22;

student2.course = "Mathematics";

student2.details = function() {

return `${this.name} is ${this.age} years old and studies ${this.course}.`;

};

console.log(student2.details());

// Constructor function method

function Student(name, age, course) {

this.name = name;

this.age = age;

this.course = course;

this.details = function() {

return `${this.name} is ${this.age} years old and studies ${this.course}.`;

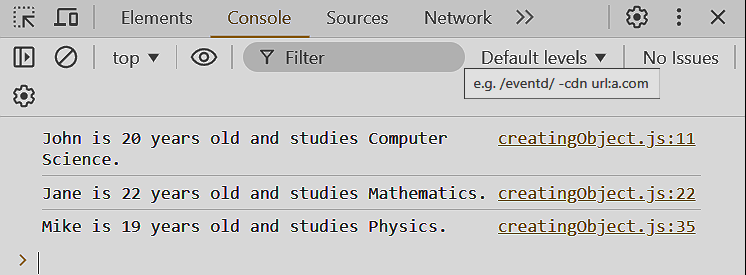
};

}

const student3 = new Student("Mike", 19, "Physics");

console.log(student3.details());

**OUTPUT–14 :**



**Q15: Write JS Code to Create Employee Object using constructor and update the salary of Employee using Update Salary Method.**

**CODE:**

// Constructor function to create Employee object

function Employee(name, age, salary, position) {

this.name = name;

this.age = age;

this.salary = salary;

this.position = position;

// Method to update the salary

this.updateSalary = function(newSalary) {

this.salary = newSalary;

console.log(`The new salary of ${this.name} is ${this.salary}`);

};

// Method to display employee details

this.details = function() {

return `${this.name}, age ${this.age}, holds the position of ${this.position} and earns a salary of ${this.salary}.`;

};

}

// Creating an employee object

let emp1 = new Employee("Alice", 30, 50000, "Software Engineer");

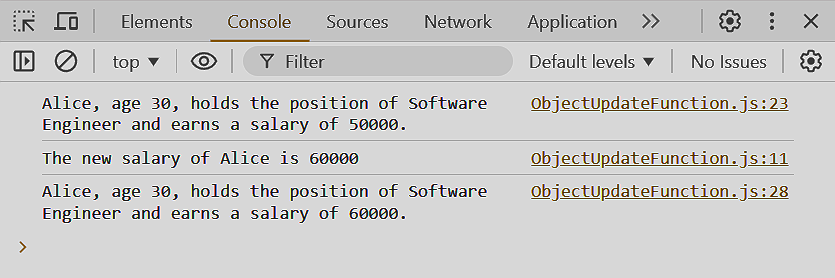
console.log(emp1.details());

// Updating the salary

emp1.updateSalary(60000);

console.log(emp1.details());

**OUTPUT-15 :**

****