**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 1: Write a simple JavaScript program to perform arithmetic operations**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Practical 1</title>

</head>

<body>

<script>

let a = 10;

let b = 5;

console.log("a + b = ", a+b);

console.log("a - b = ", a-b);

console.log("a / b = ", a/b);

console.log("a \* b = ", a\*b);

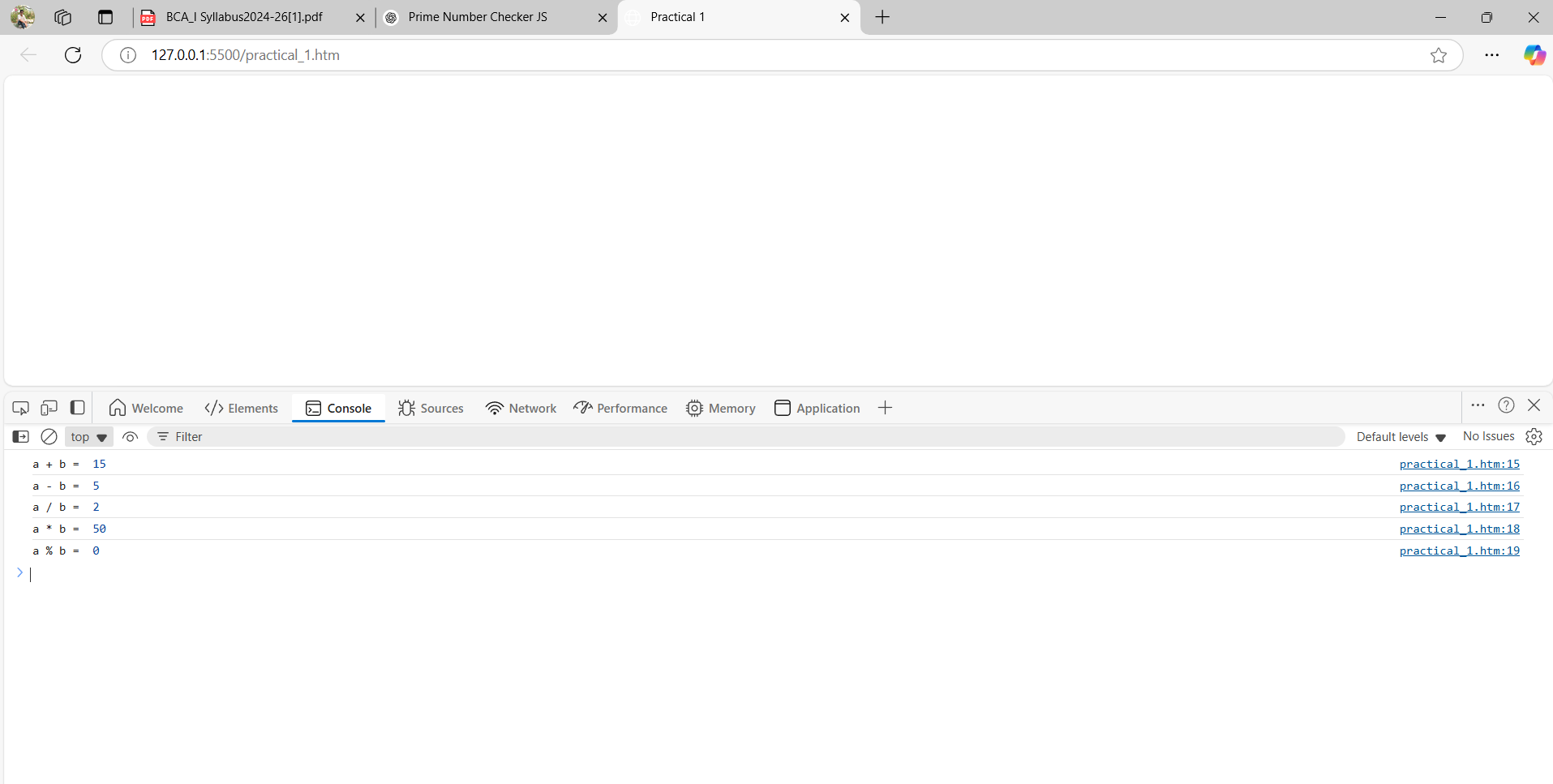
console.log("a % b = ", a%b);

</script>

</body>

</html>

**OUTPUT:**

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 2: Implement a JavaScript program using if-else statements and loops.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Practical 2</title>

</head>

<body>

<script>

let age = 5;

if(age<18){

console.log("You cannot drive");

}else if(age>65){

console.log("Please ask someone to drive for you");

}else{

console.log("You can drive");

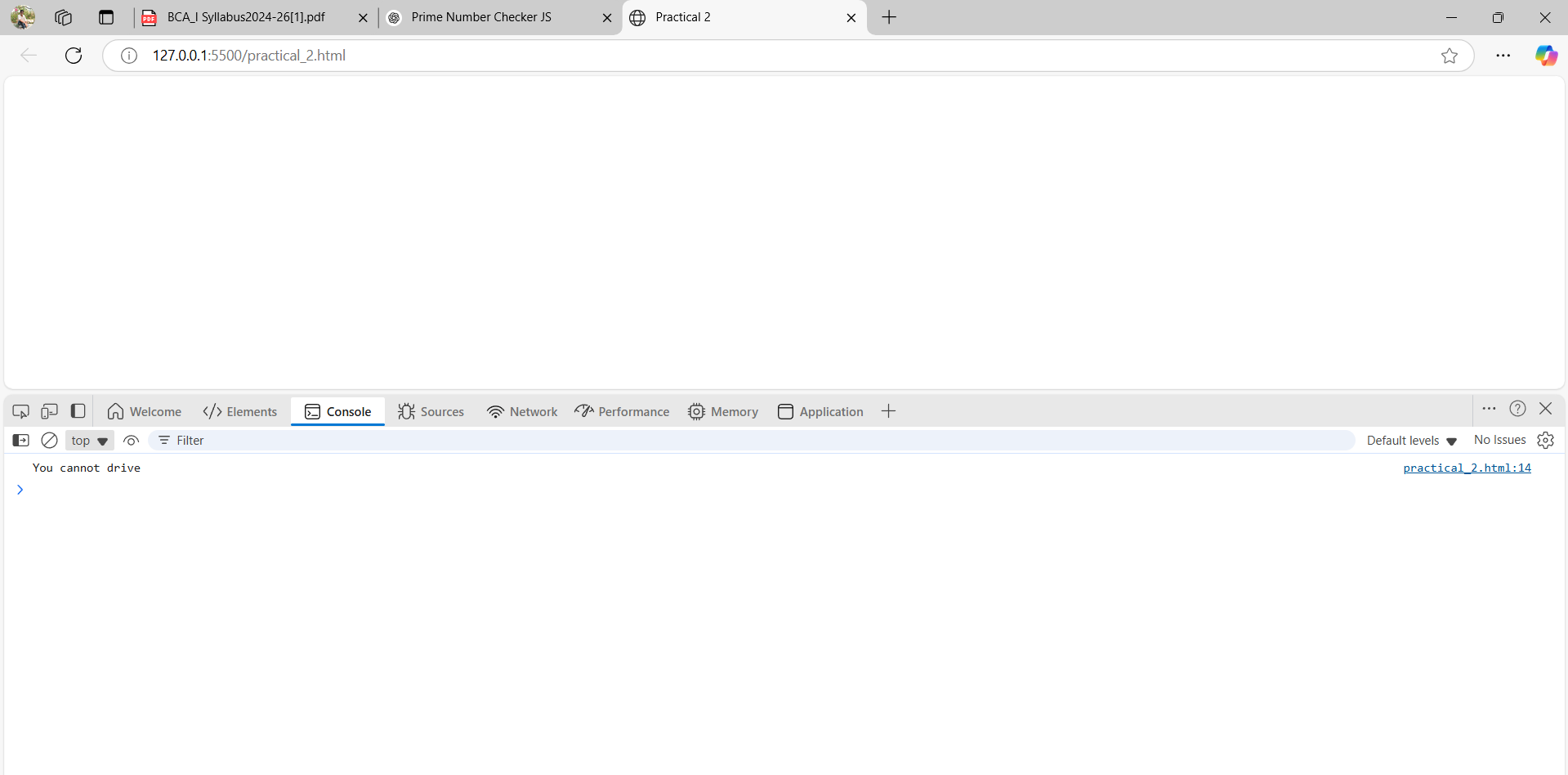
}

</script>

</body>

</html>

**OUTPUT:**

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 3: Create and invoke functions that perform specific tasks.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

</head>

<body>

<script>

function addNumbers(a, b) {

return a + b;

}

function calculateCircleArea(radius) {

return Math.PI \* radius \* radius;

}

function greetUser(name) {

console.log("Hello, " + name + "!");

}

function introduce(name = "Guest") {

console.log("Welcome, " + name + "!");

}

function isEven(number) {

if (number % 2 === 0) {

console.log(number + " is an even number.");

} else {

console.log(number + " is an odd number.");

}

}

let sum = addNumbers(10, 20);

console.log("Sum of 10 and 20: " + sum);

let area = calculateCircleArea(5);

console.log("Area of the circle: " + area.toFixed(2));

greetUser("Alice");

greetUser("Bob");

introduce("John");

introduce();

isEven(12);

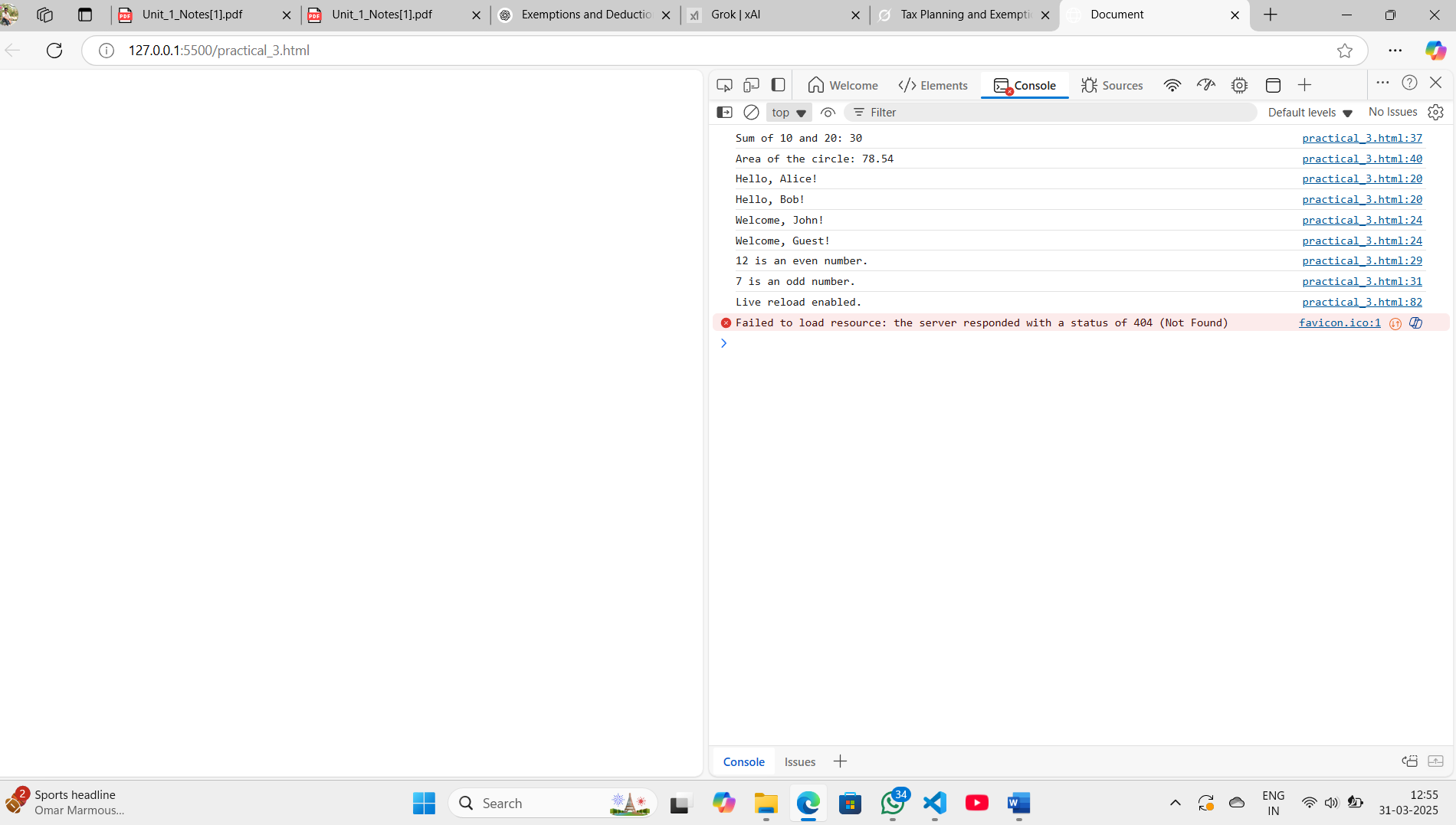
isEven(7);

</script>

</body>

</html>

**OUTPUT:**

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 4**: **Event Handling: Develop a web page where JavaScript responds to user events (e.g., button clicks).**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Practical 3</title>

</head>

<body>

<button id="button">Click Me</button>

<script>

document.getElementById("button").addEventListener("click", function(){

console.log("Button has been clicked");

});

document.getElementById("button").addEventListener("mouseover", function() {

document.getElementById("button").style.backgroundColor = "lightblue"

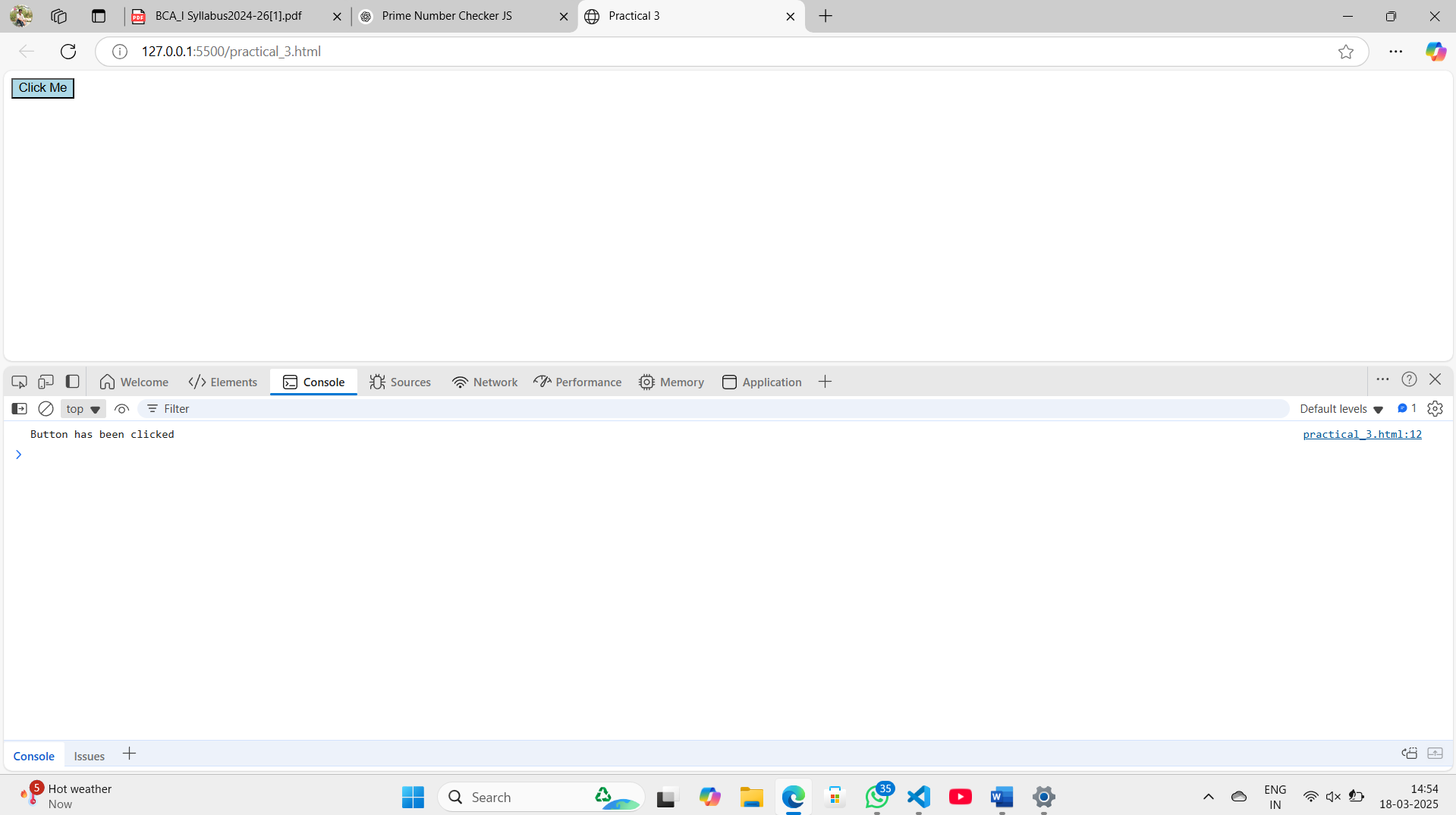
})

</script>

</body>

</html>

**OUTPUT:**

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 6**: **Create a basic HTTP server using Node.js that responds with "Hello, World!" to any request.**

const http = require("http");

const port = 3000;

const host = '127.0.0.1'

const server = http.createServer((req,res) => {

res.statusCode = 200;

res.setHeader('Content-Type', 'text/plain');

res.end('Hello, World!');

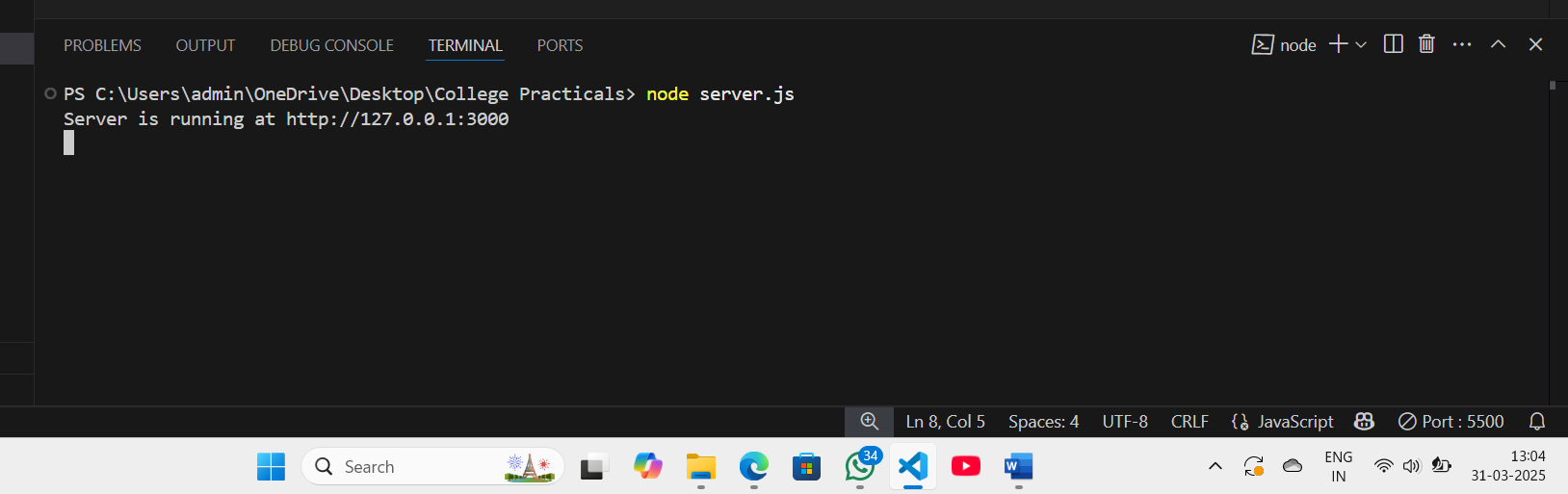
});

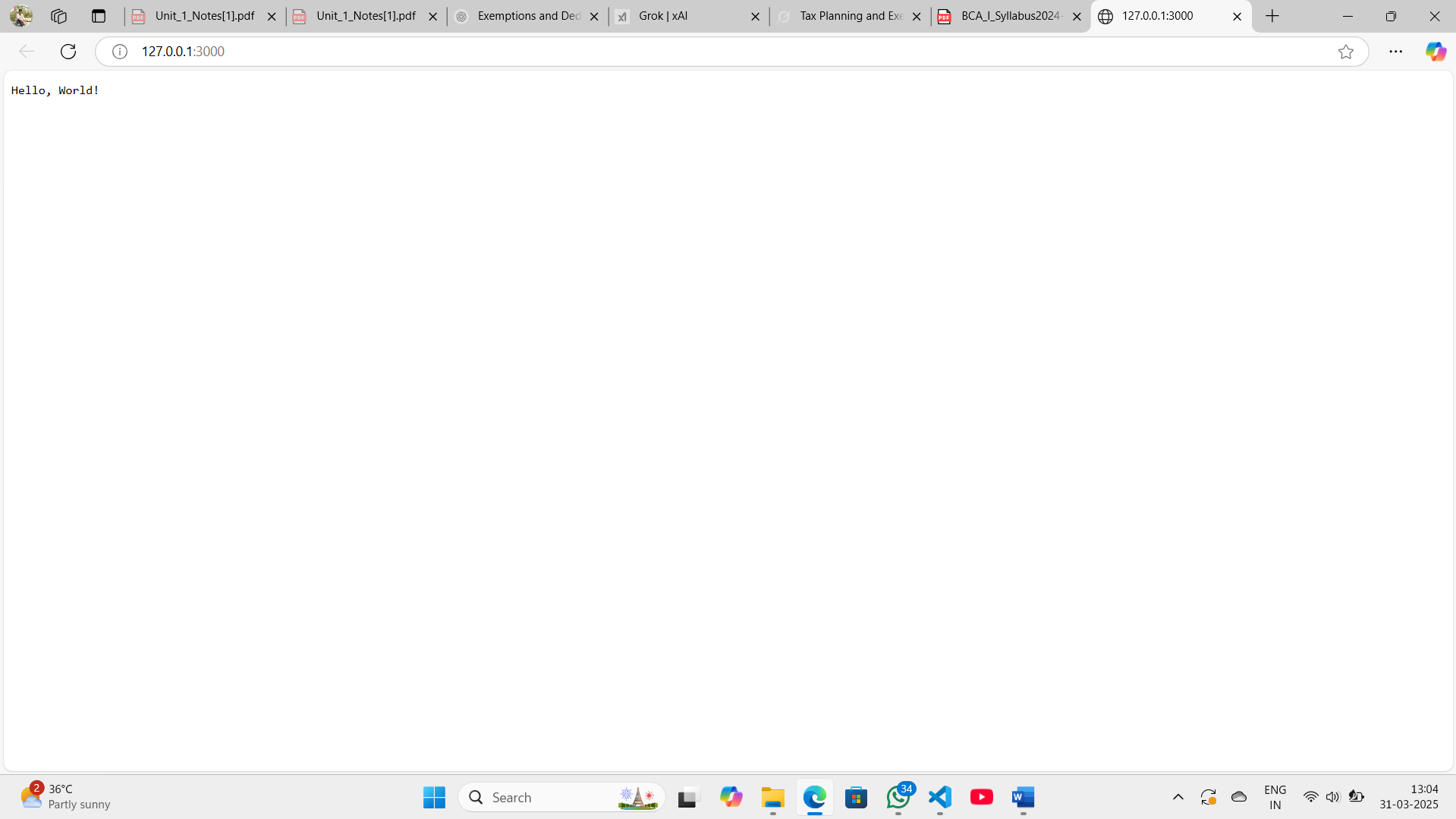
server.listen(port,host,() => {

console.log(`Server is running at http://${host}:${port}`);

});

**OUTPUT:**

****

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 7**: **Develop a basic REST API using Node.js that performs CR (Create, Read) operations on a simple data set.**

const express = require('express');

const app = express();

const port = 3000;

app.use(express.json());

let users = [

{ id: 1, name: 'Alice', age: 30 },

{ id: 2, name: 'Bob', age: 25 }

];

app.post('/users', (req, res) => {

const { name, age } = req.body;

const newUser = { id: users.length + 1, name, age };

users.push(newUser);

res.status(201).json(newUser);

});

app.get('/users', (req, res) => {

res.status(200).json(users);

});

app.get('/users/:id', (req, res) => {

const userId = parseInt(req.params.id, 10);

const user = users.find(u => u.id === userId);

if (user) {

res.status(200).json(user);

} else {

res.status(404).json({ message: 'User not found' });

}

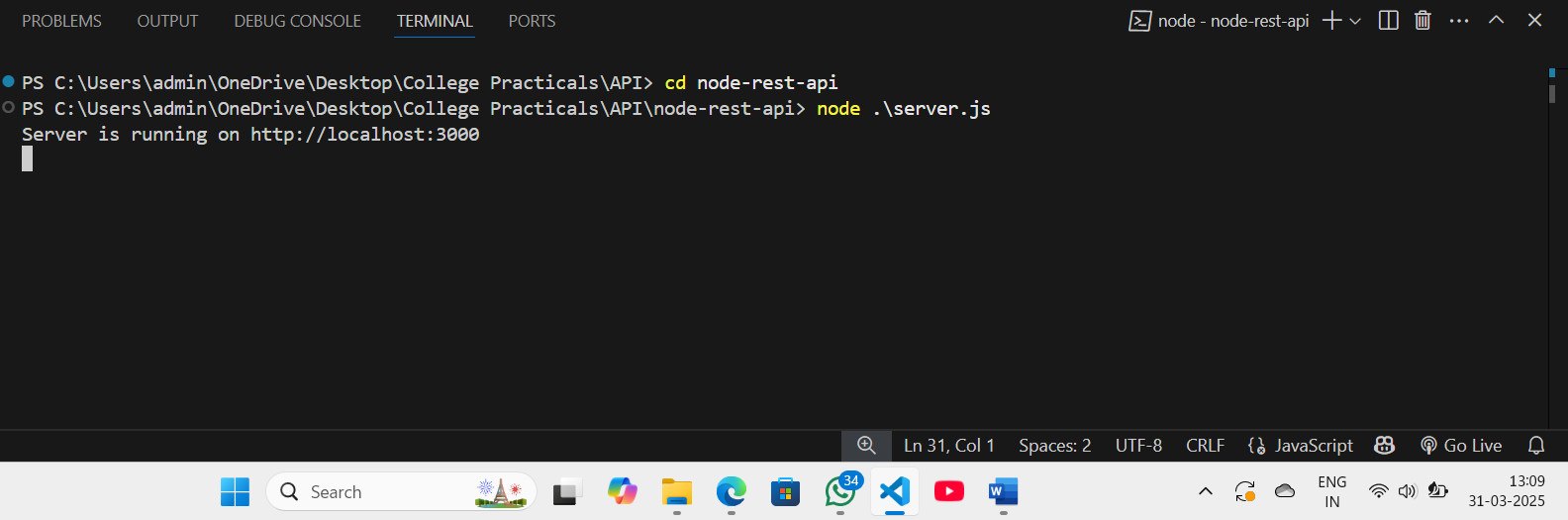
});

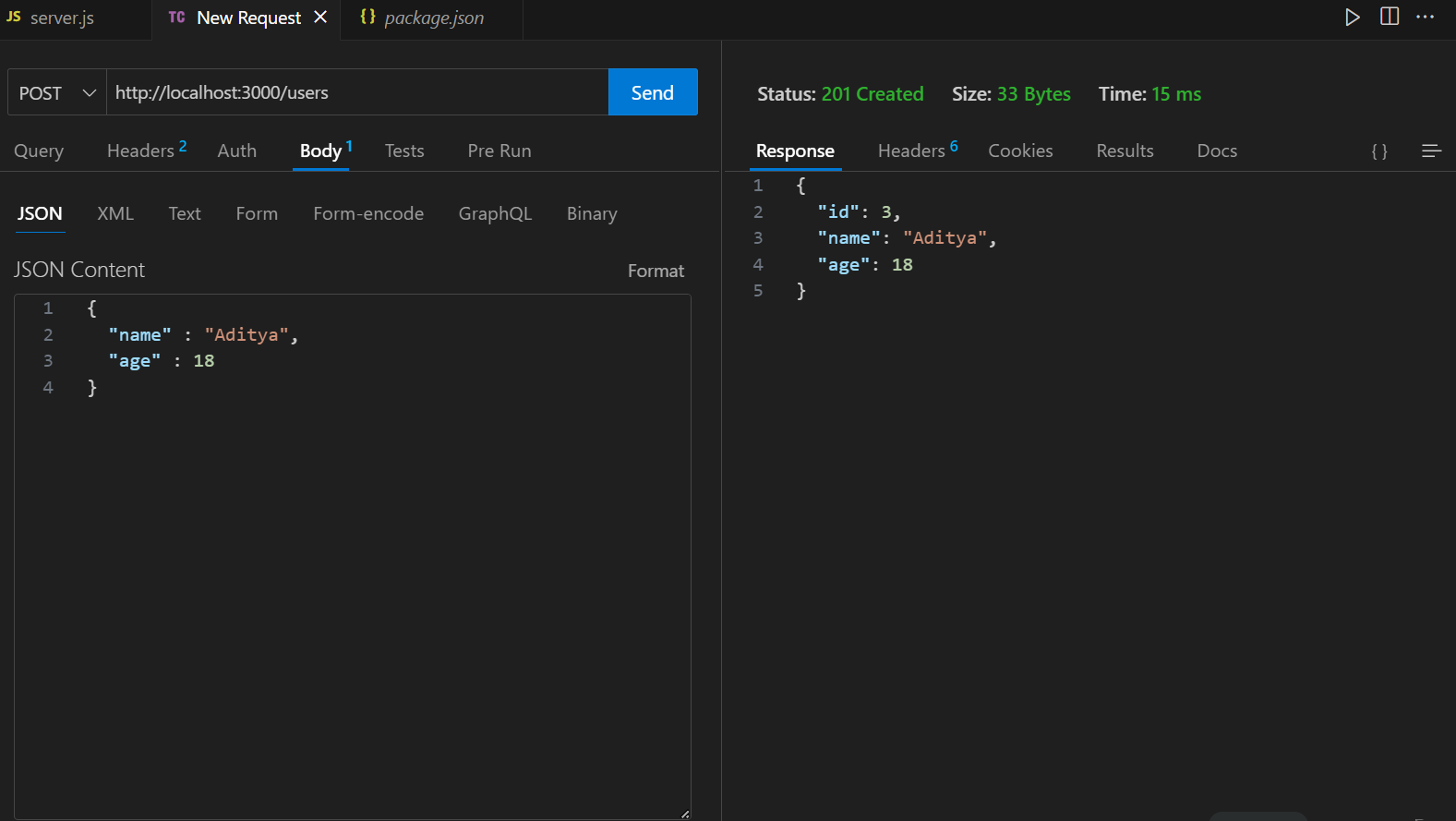
app.listen(port, () => {

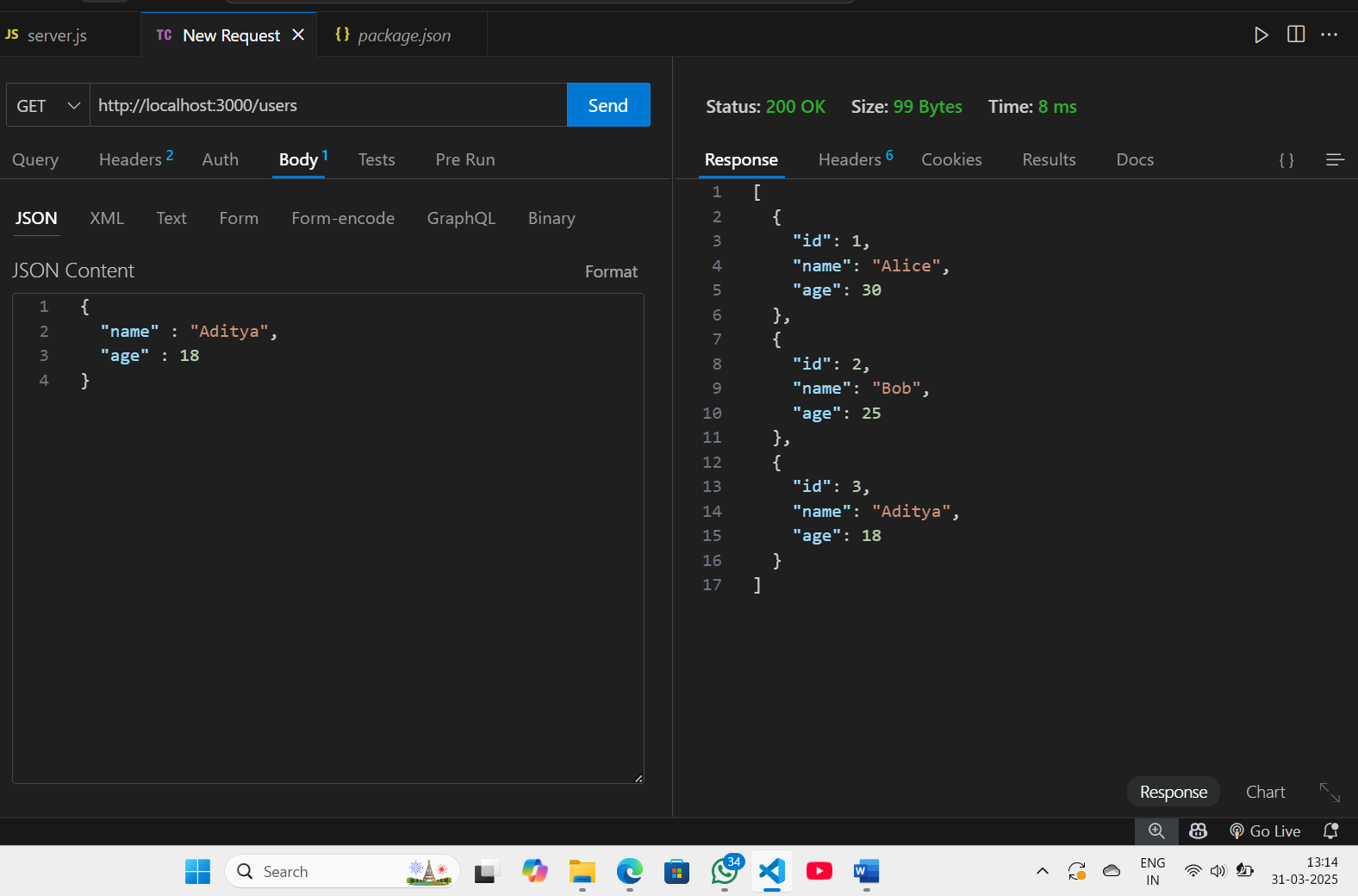
console.log(`Server is running on http://localhost:${port}`);

});

**OUTPUT:**

****

****

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 8: Create a REST API with Node.js**

const express = require('express');

const app = express();

const port = 3000;

app.use(express.json());

let users = [

{ id: 1, name: 'John Doe', age: 30 },

{ id: 2, name: 'Jane Doe', age: 25 },

{ id: 3, name: 'Jim Brown', age: 35 },

];

app.get('/users', (req, res) => {

res.status(200).json(users);

});

app.put('/users/:id', (req, res) => {

const userId = parseInt(req.params.id);

const updatedData = req.body;

let user = users.find(u => u.id === userId);

if (!user) {

return res.status(404).send({ error: 'User not found' });

}

user.name = updatedData.name || user.name;

user.age = updatedData.age || user.age;

res.status(200).json(user);

});

app.delete('/users/:id', (req, res) => {

const userId = parseInt(req.params.id);

const userIndex = users.findIndex(u => u.id === userId);

if (userIndex === -1) {

return res.status(404).send({ error: 'User not found' });

}

const deletedUser = users.splice(userIndex, 1);

res.status(200).json({ message: 'User deleted successfully', deletedUser });

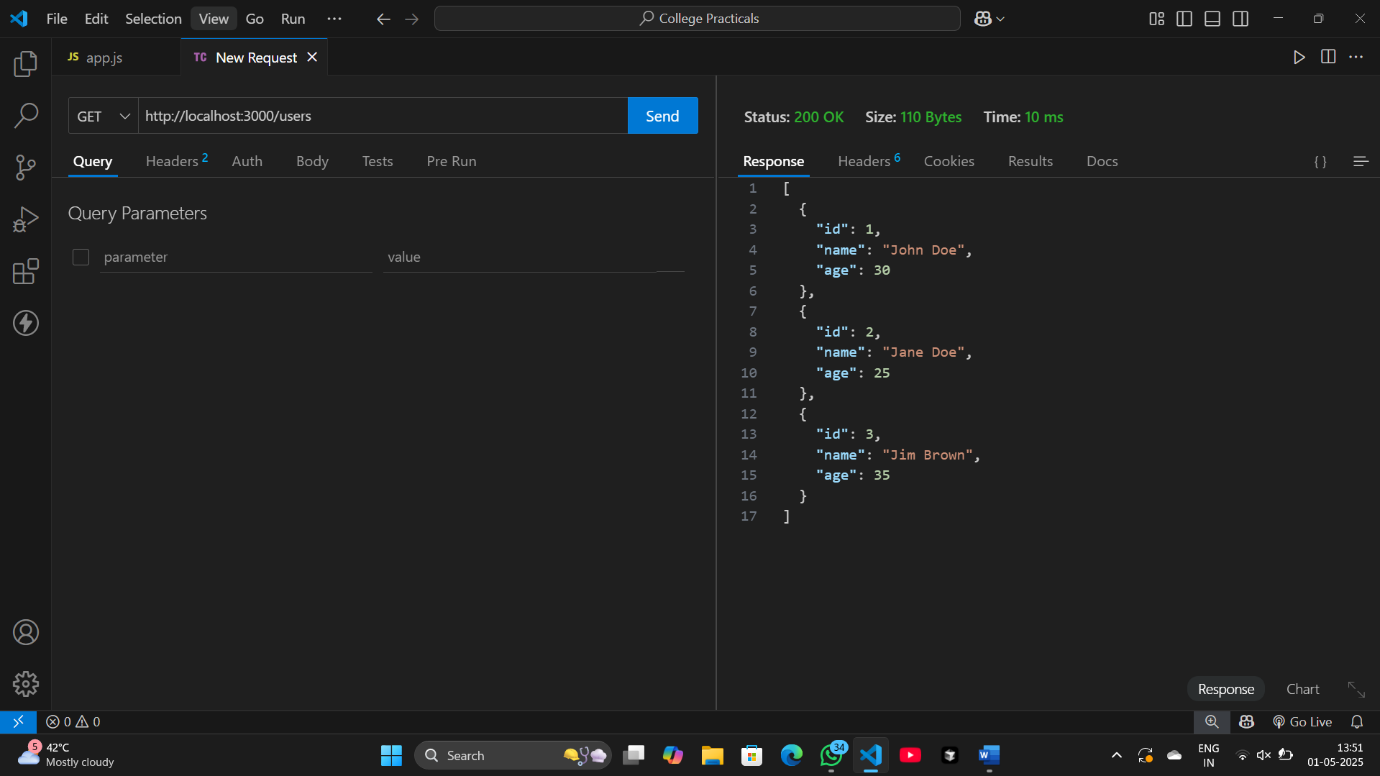
});

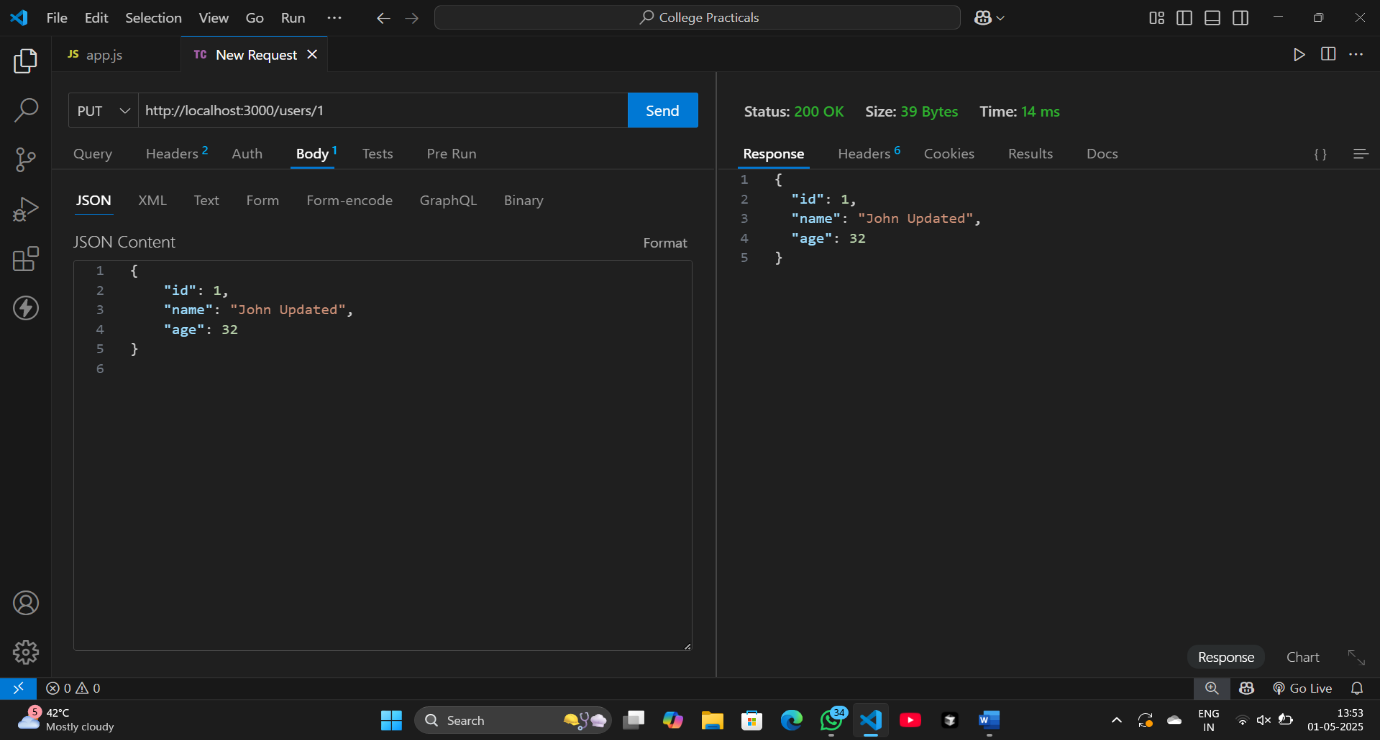
app.listen(port, () => {

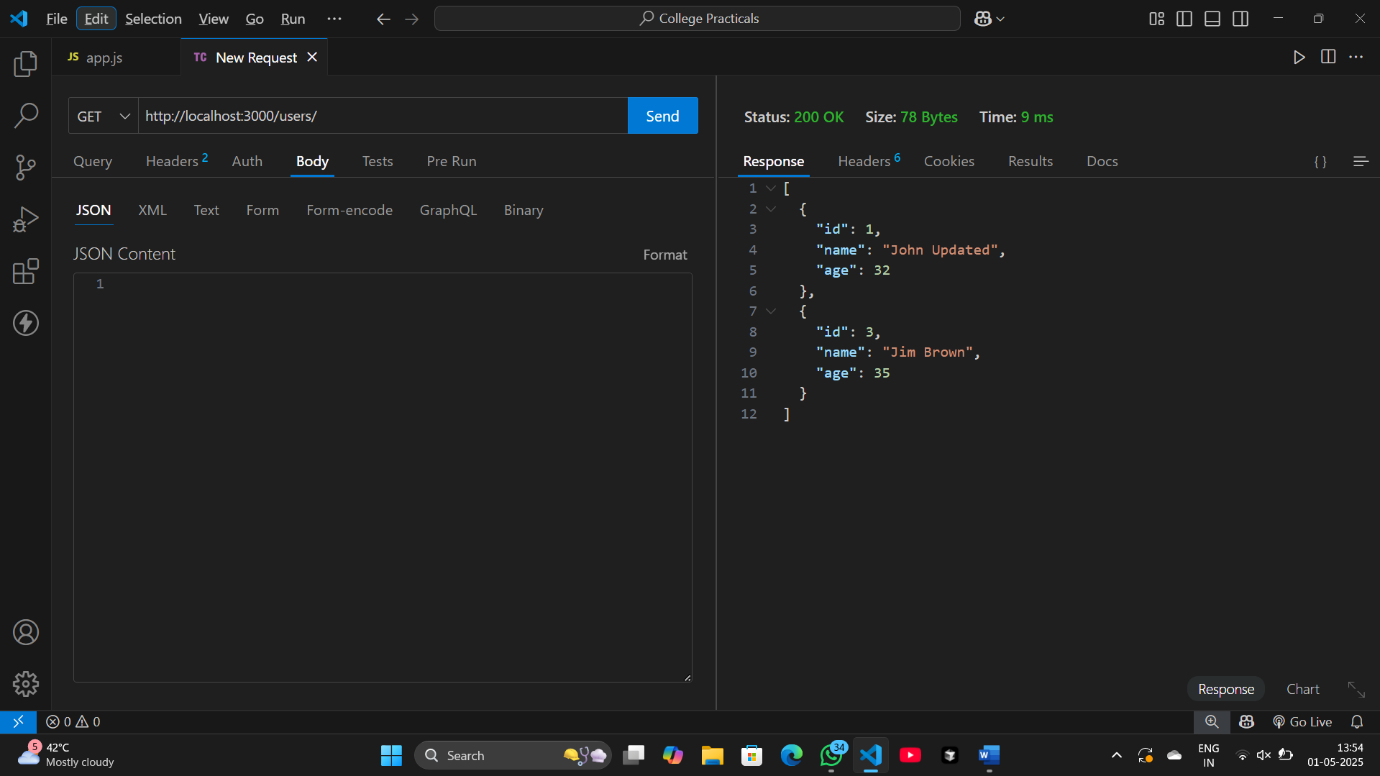
console.log(`Server running on http://localhost:${port}`);

});

**Output:**







**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 9: Perform a practical on bootstrap setup**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Bootstrap Example</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

<div class="container">

<h1 class="text-center my-4">Welcome to Bootstrap</h1>

<div class="row">

<div class="col-md-4">

<div class="card h-100">

<img src="ravan-enemy-of-aryavarta.jpg" class="card-img-top img-fluid" style="height: 300px; object-fit: cover;" alt="Ravan: Enemy of Aryavarta">

<div class="card-body">

<h5 class="card-title">Ravan: Enemy of Aryavarta</h5>

<p class="card-text">A historical fiction novel about the legendary king of Lanka.</p>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card h-100">

<img src="rich-dad-poor-dad.jpg" class="card-img-top img-fluid" style="height: 300px; object-fit: cover;" alt="Rich Dad Poor Dad">

<div class="card-body">

<h5 class="card-title">Rich Dad Poor Dad</h5>

<p class="card-text">A personal finance book by Robert Kiyosaki about financial literacy.</p>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card h-100">

<img src="the\_god\_of\_small\_things.jpg" class="card-img-top img-fluid" style="height: 300px; object-fit: cover;" alt="The God of Small Things">

<div class="card-body">

<h5 class="card-title">The God of Small Things</h5>

<p class="card-text">A novel by Arundhati Roy that won the Booker Prize in 1997.</p>

</div>

</div>

</div>

</div>

<button class="btn btn-primary mt-4">Click Me!</button>

</div>

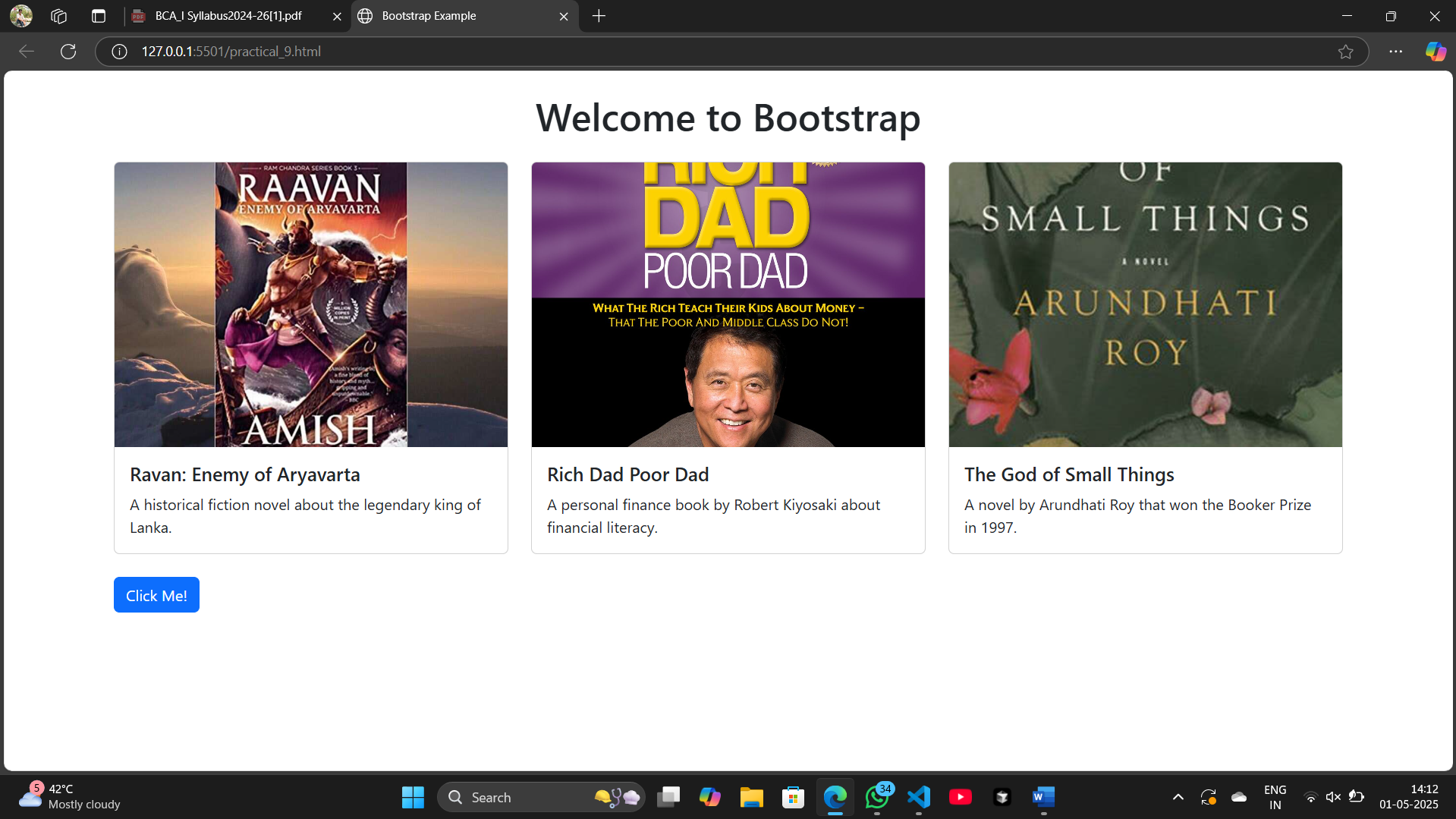
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.min.js"></script>

</body>

</html>

**Output:**



**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 10: Bootstrap Grid Layout**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Responsive Web Page Layout</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

<header class="bg-primary text-white text-center py-4">

<h1>My Responsive Web Page</h1>

</header>

<div class="container my-4">

<div class="row">

<aside class="col-lg-3 col-md-4 col-sm-12">

<div class="card">

<div class="card-body">

<h5 class="card-title">Sidebar</h5>

<p>This is the sidebar section. It will be displayed on the left side on larger screens, but stack on smaller screens.</p>

</div>

</div>

</aside>

<main class="col-lg-9 col-md-8 col-sm-12">

<div class="card">

<div class="card-body">

<h5 class="card-title">Main Content</h5>

<p>This is the main content section. On large screens, it takes up more space. On smaller screens, both the sidebar and content section will stack vertically.</p>

<p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Temporibus, nemo.</p>

</div>

</div>

</main>

</div>

</div>

<footer class="bg-dark text-white text-center py-3">

<p>&copy; 2025 My Website | All Rights Reserved</p>

</footer>

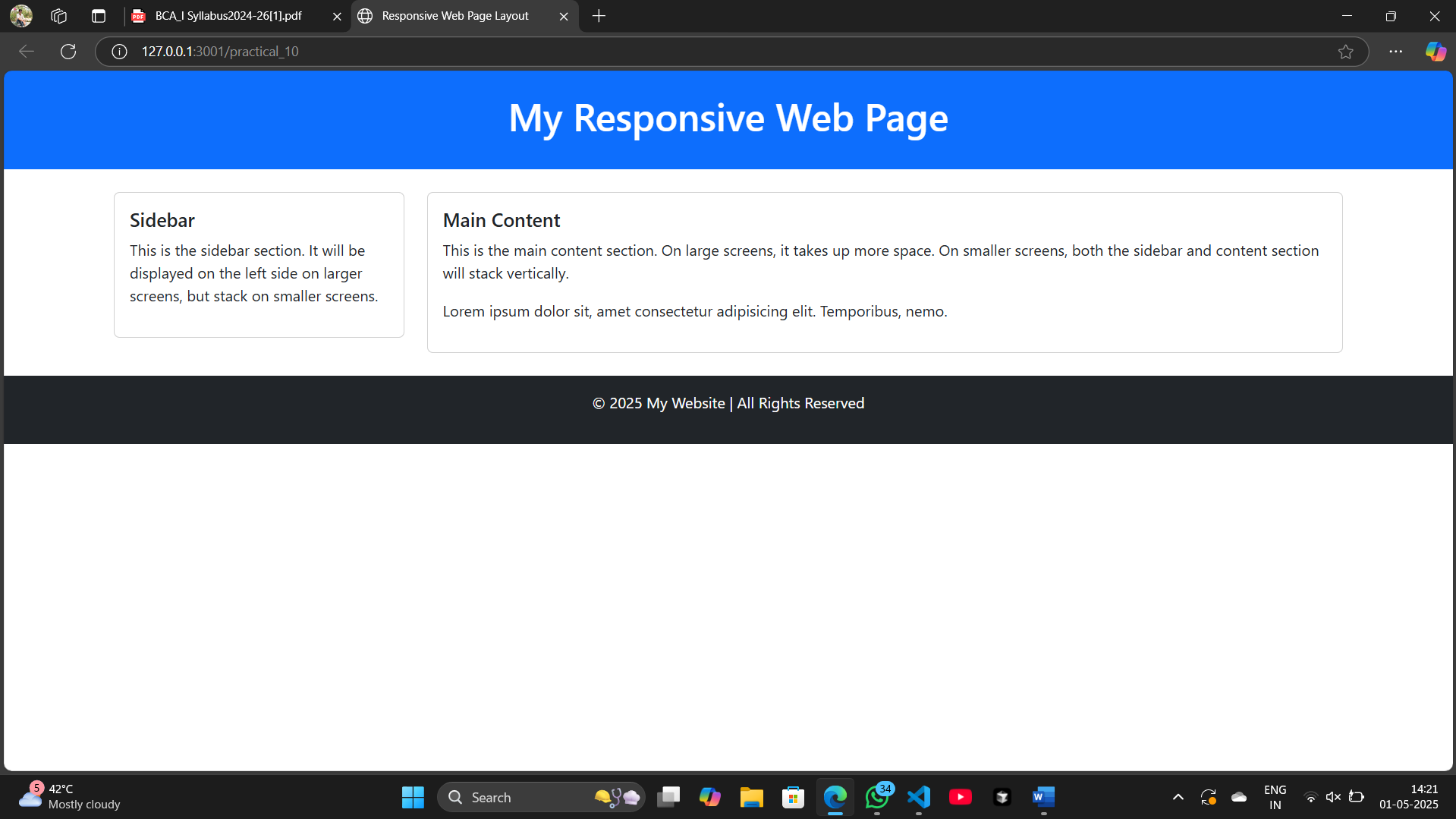
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.min.js"></script>

</body>

</html>

**Output:**

****

**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 11: Bootstrap Components**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Navigation Bar and Modal</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">

<div class="container-fluid">

<a class="navbar-brand" href="#">My Website</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">726

<ul class="navbar-nav ms-auto">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="#">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">About</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Services</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Contact</a>

</li>

</ul>

</div>

</div>

</nav>

<div class="container my-5">

<h2>Welcome to My Website</h2>

<p>This is a simple example of a navigation bar and a modal using Bootstrap components.</p>

<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exampleModal">

Open Modal

</button>

</div>

<div class="modal fade" id="exampleModal" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<h5 class="modal-title" id="exampleModalLabel">Modal Title</h5>

<button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>

</div>

<div class="modal-body">

<p>This is a simple modal window. You can add any content here like forms, images, or text.</p>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>

<button type="button" class="btn btn-primary">Save changes</button>

</div>

</div>

</div>

</div>

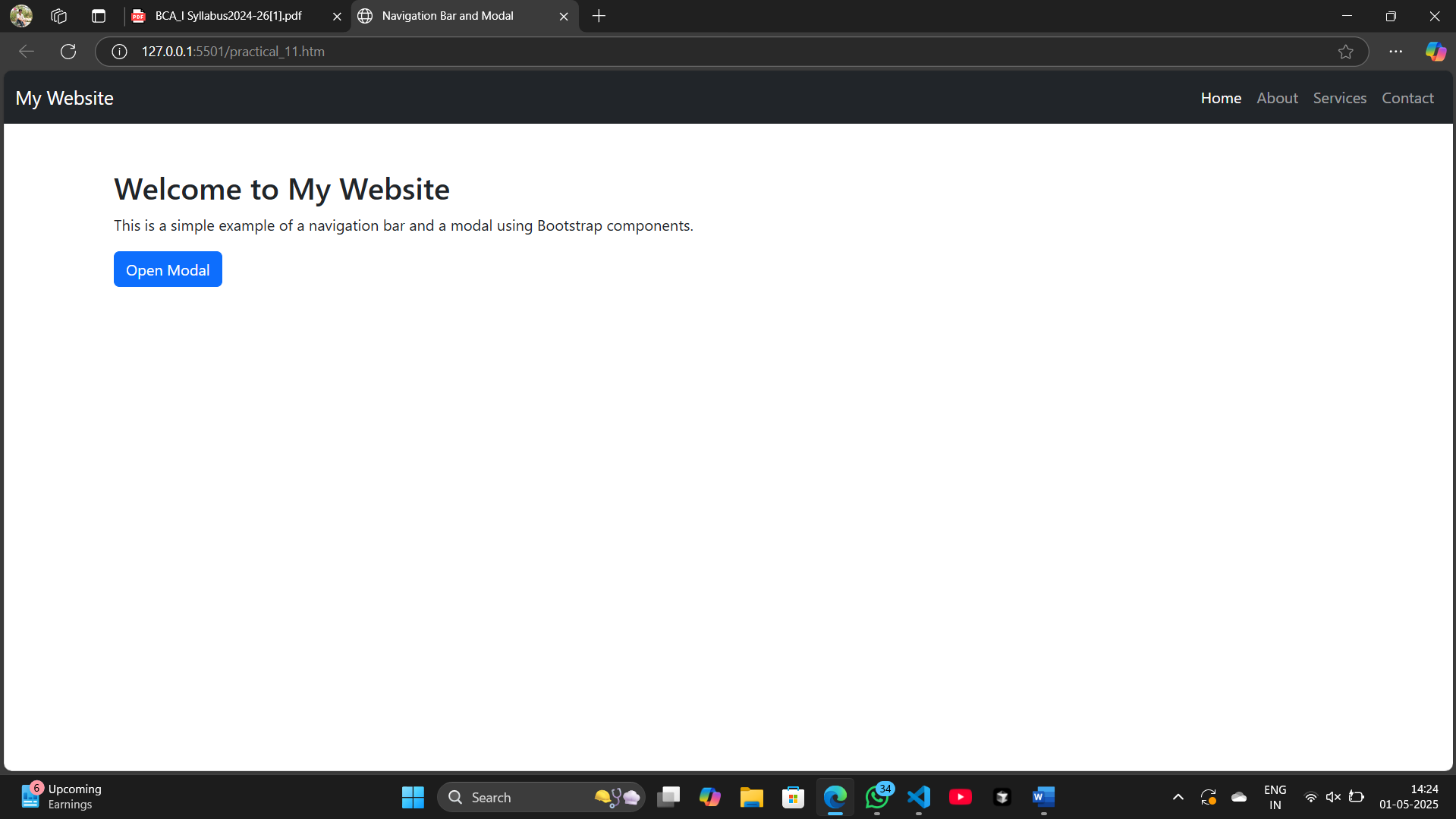
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>

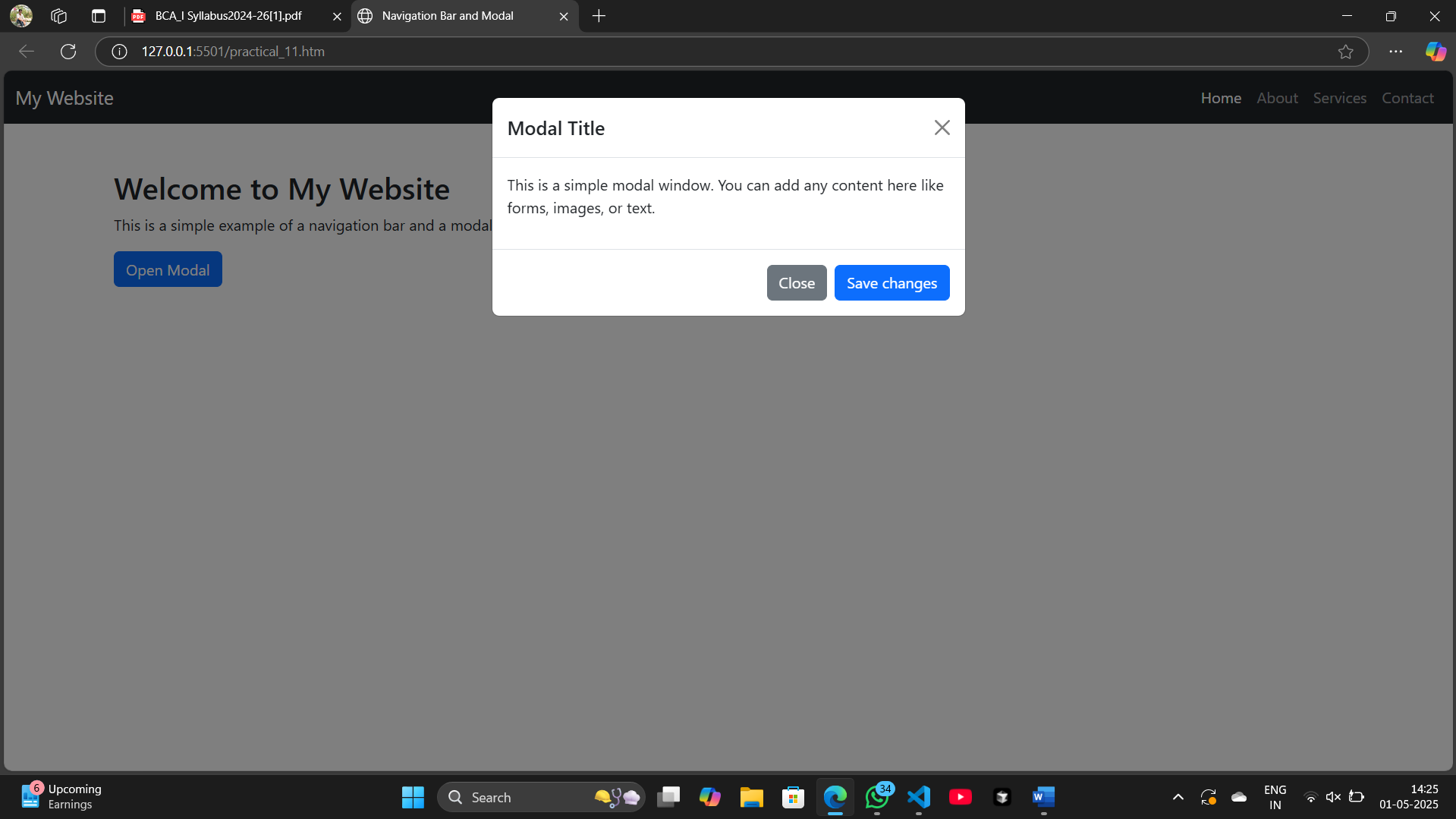
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.min.js"></script>

</body>

</html>

**Output:**

****



**Name: Aditya Dnyaneshwar Pawar**

**Class: FY BCA Roll No.: 157 Batch: B5**

**Practical 12: Create a Bootstrap Form**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Bootstrap Form Example</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

<div class="container mt-5">

<h2>Create an Account</h2>

<form action="#" method="POST">

<div class="mb-3">

<label for="name" class="form-label">Full Name</label>

<input type="text" class="form-control" id="name" placeholder="Enter your full name">

</div>

<div class="mb-3">

<label for="email" class="form-label">Email address</label>

<input type="email" class="form-control" id="email" placeholder="Enter your email">

</div>

<div class="mb-3">

<label for="password" class="form-label">Password</label>

<input type="password" class="form-control" id="password" placeholder="Create a password">

</div>

<div class="mb-3">

<label for="confirmPassword" class="form-label">Confirm Password</label>

<input type="password" class="form-control" id="confirmPassword" placeholder="Confirm your password">

</div>

<div class="mb-3">

<label class="form-label">Gender</label><br>

<input type="radio" class="form-check-input" name="gender" value="Male" id="male">

<label class="form-check-label" for="male">Male</label>

<input type="radio" class="form-check-input" name="gender" value="Female" id="female">

<label class="form-check-label" for="female">Female</label>

</div>

<div class="mb-3 form-check">

<input type="checkbox" class="form-check-input" id="terms">

<label class="form-check-label" for="terms">I agree to the terms and conditions</label>

</div>

<button type="submit" class="btn btn-primary">Submit</button>

</form>

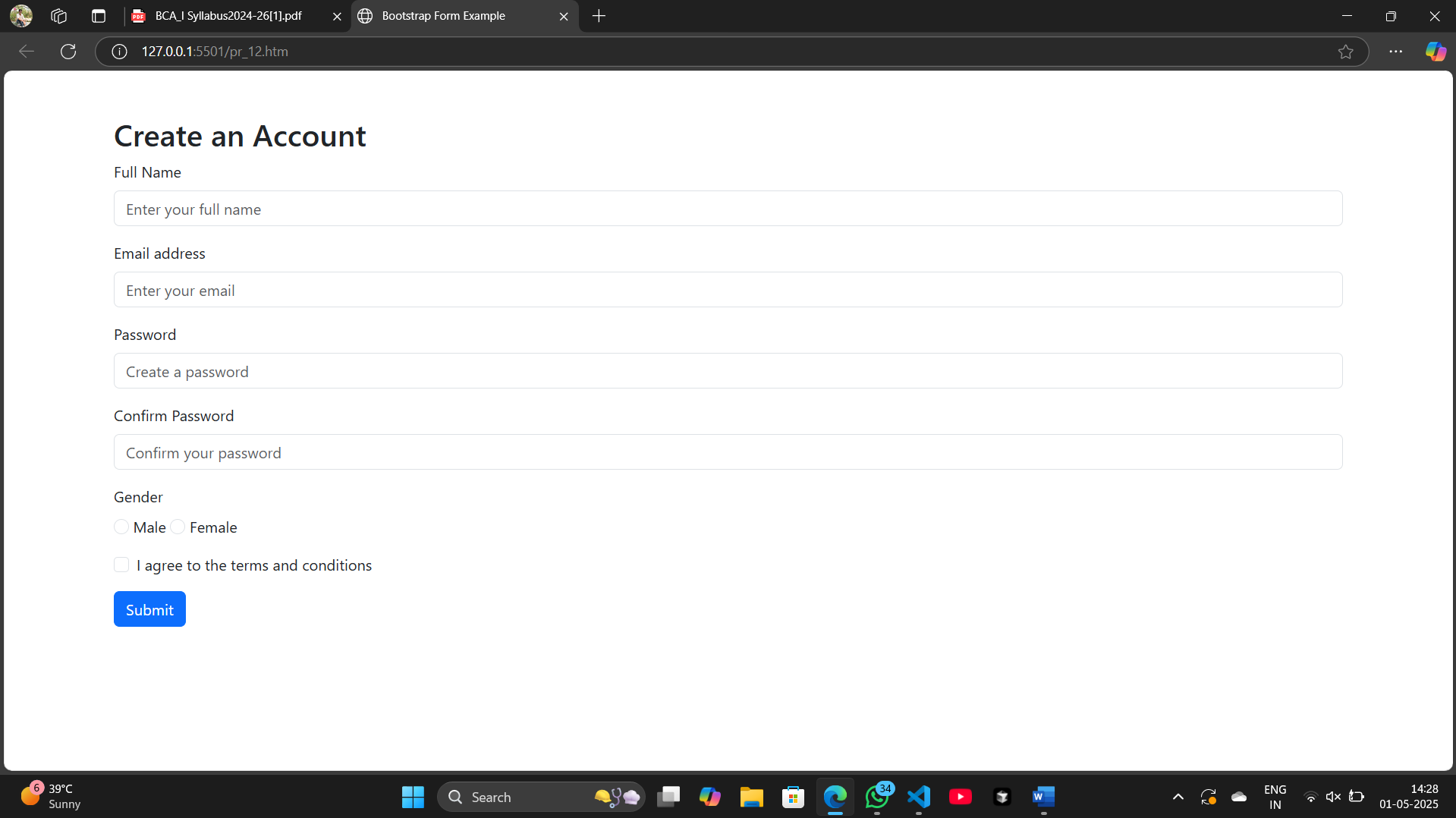
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

**Output:**

****