**Elective module 3**

**Web Development**

**PRACTICAL**

**Name : - Prateek Kumar**

**Registration No : - ADIT22AP00152**

**NSTI Name : - NSTI Noida**

**Course : - ADIT (IBM)**

**Date : - 25-10-2024**

**Module : - elective module 3**

**Practical : - web development**

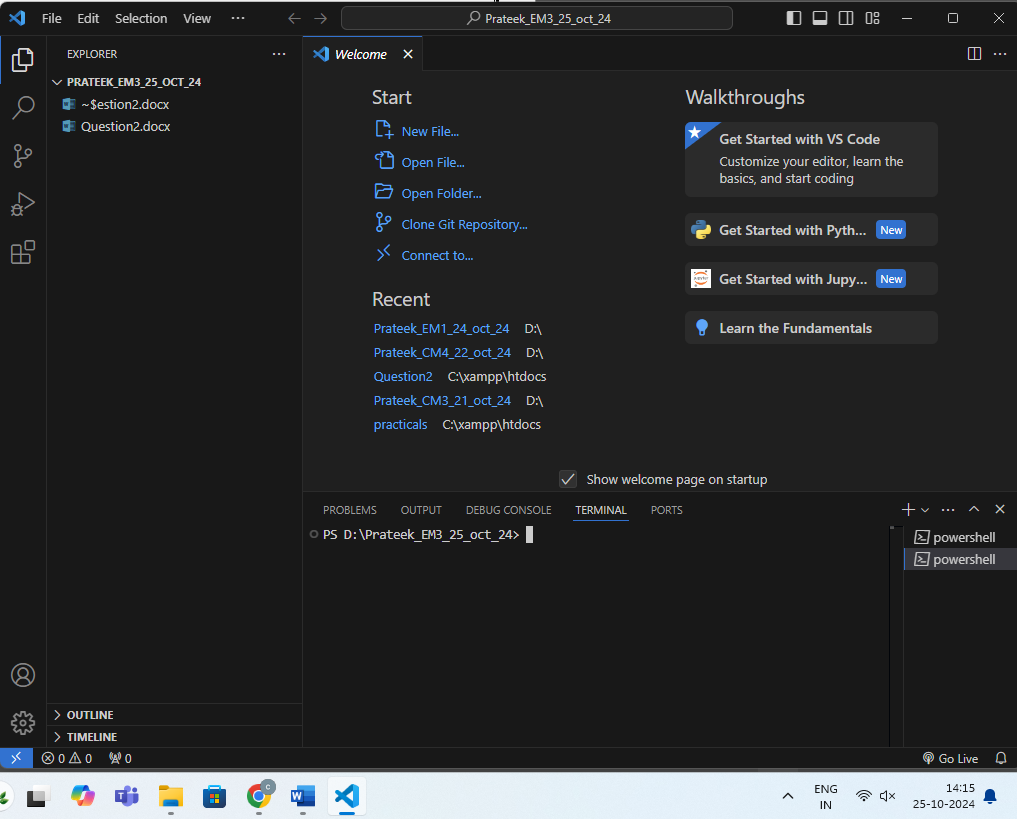
**Requirements/tools : -**

1. **Hardware: -** 
   * 1. **Working PC**
     2. **Internet connection**
2. **Software: -**
   * 1. **Browser**
     2. **Code editor (vs code)**

**Question 1: - Create an angular application that accepts input data using @Input and displays it on page**

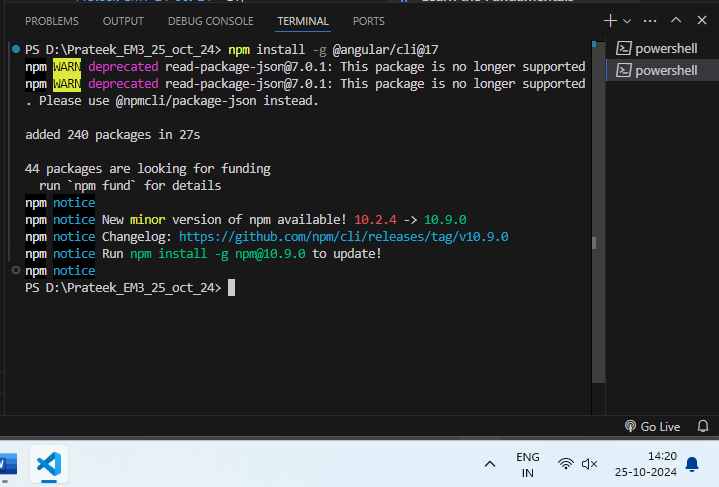
**solution**

**STEP1: - Open visual studio code and open the folder where you want to create the angular project and open the terminal using “ctrl + shift + `”**

****

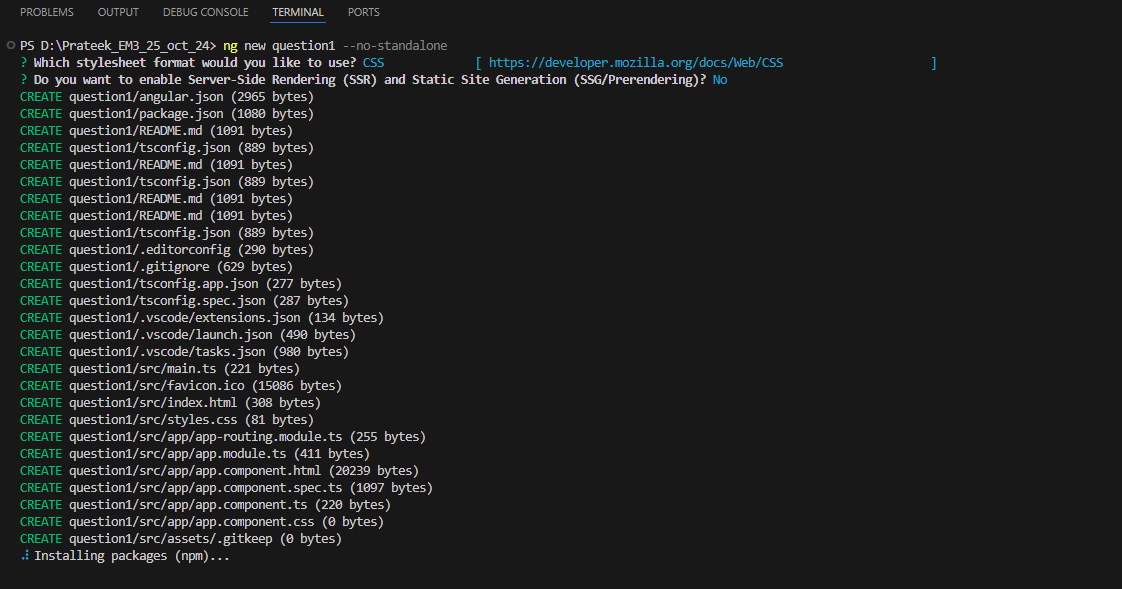
**STEP2: - Install latest angular CLI using**

**npm install -g @angular/cli@17**

****

**STEP3: - Create angular project using ng new question1 and select options to customize your project according to your needs and let it create the application and installing packages**

**ng new question1 –no-standalone**

****

**STEP4: - Generate the Child Component using**

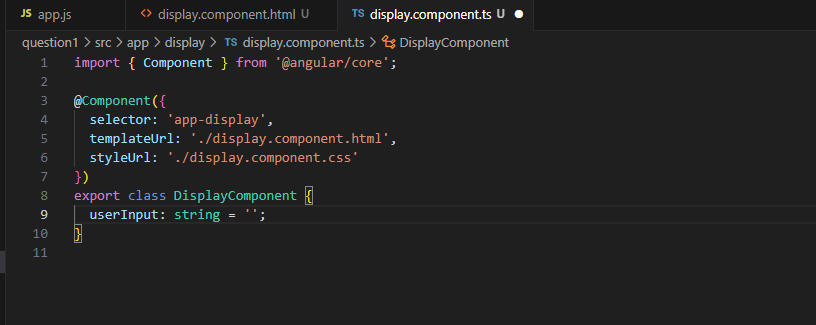
**ng generate component display**

**A screen shot of a computer program

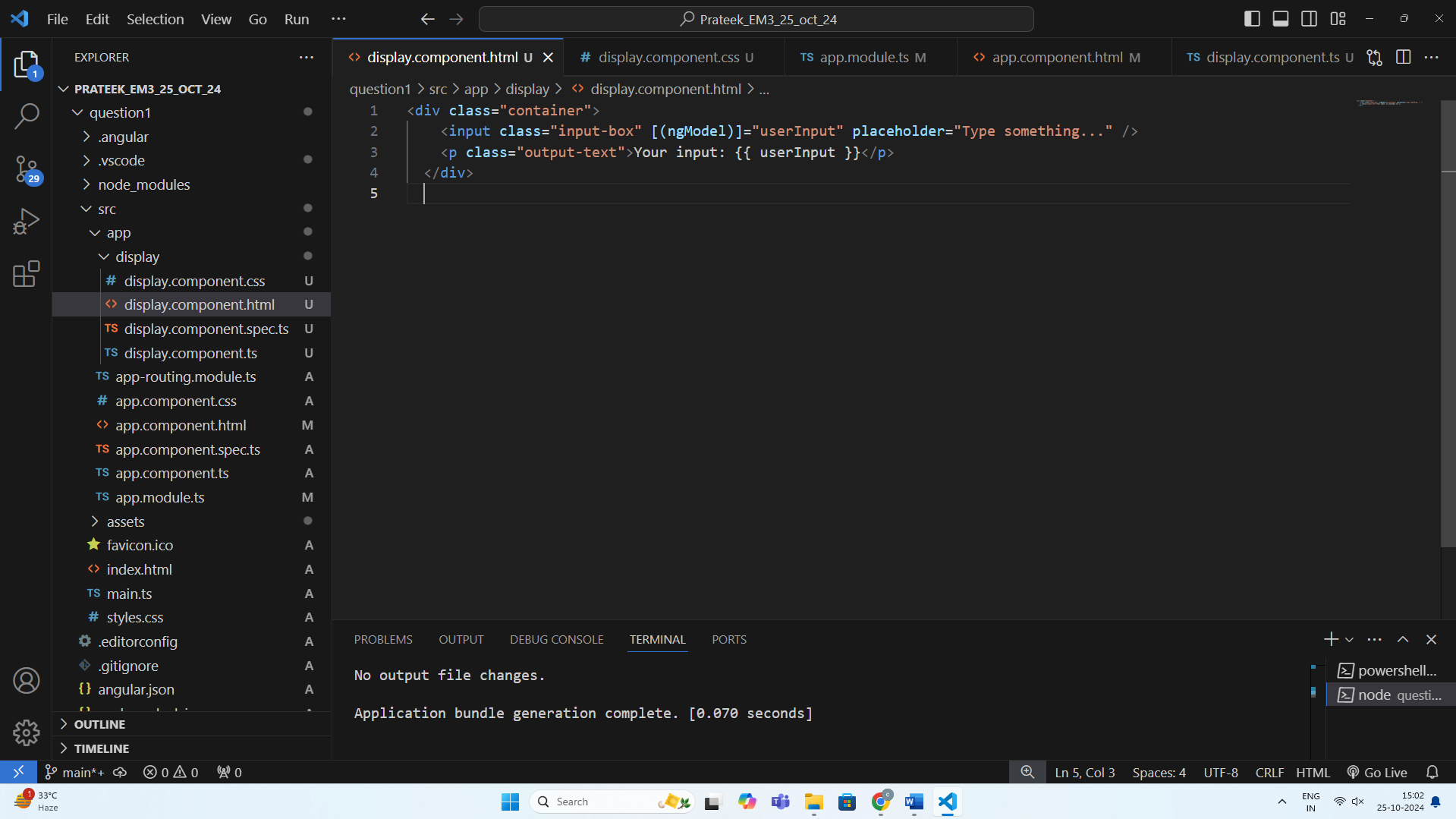
Description automatically generated**

**STEP5: - configure the Child Component**

**i)create a property to store input**

****

**ii)Go to display.component.html and write the code to take user input and show the output on the screen**

****

**Here is the code**

<div class="container">

    <input class="input-box" [(ngModel)]="userInput" placeholder="Type something..." />

    <p class="output-text">Your input: {{ userInput }}</p>

  </div>

**iii).Lets customize it with CSS open display.component.css add css to our html classes**

/\* Center the container and add some padding \*/

.container {

    display: flex;

    flex-direction: column;

    align-items: center;

    justify-content: center;

    background-color: #f9f9f9;

    padding: 20px;

    border-radius: 8px;

    max-width: 400px;

    margin: 0 auto;

    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

  }

  /\* Style the input box \*/

  .input-box {

    width: 100%;

    padding: 10px;

    margin-bottom: 15px;

    font-size: 16px;

    border: 1px solid #ccc;

    border-radius: 4px;

    outline: none;

    transition: border-color 0.3s;

  }

  .input-box:focus {

    border-color: #4CAF50;

  }

  /\* Style the output text \*/

  .output-text {

    font-size: 18px;

    color: #333;

    background-color: #e0f7e9;

    padding: 10px;

    border-radius: 4px;

    width: 100%;

    text-align: center;

  }

**STEP6: - Add FormsModule in app.module.ts file**

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { FormsModule } from '@angular/forms';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

import { DisplayComponent } from './display/display.component';

@NgModule({

  declarations: [

    AppComponent,

    DisplayComponent

  ],

  imports: [

    BrowserModule,

    AppRoutingModule,

    FormsModule

  ],

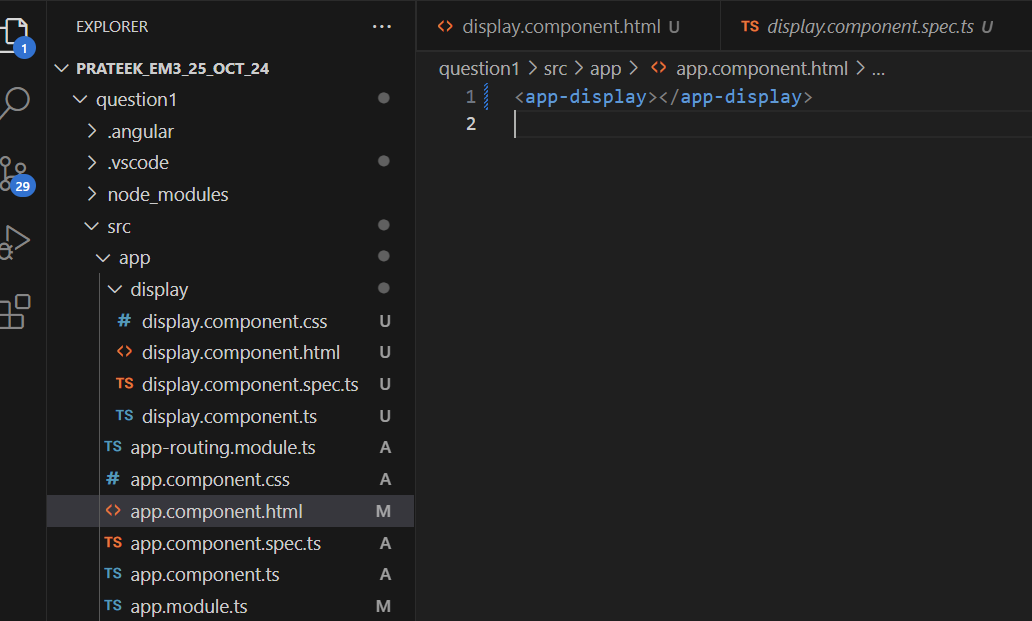
  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

**STEP7:-** **Add the display Component to appComponent**

****

**STEP8:- run the angular application and test the functionality**

1. **Run application using**

**ng serve**

1. **Lets test the output**

**A screenshot of a computer

Description automatically generated**

**As we can see the two-way data binding is working successfully**

**Question2:-** **create a basic express application that listens on port 3000 and responds with “welcome” when accessed.**

**STEP1: - create a folder named question2 in visual studio code and run the terminal using “ctrl + shift + `”**

**A screenshot of a computer

Description automatically generated**

**STEP2: - initialize it as npm directory using**

**npm init -y**

**A screenshot of a computer

Description automatically generated**

**STEP3: - Install express and using**

**npm install express**

**A computer screen with white text

Description automatically generated**

**STEP4: - create an app.js file and import express and Initialize it and show the output as welcome on the “/” homepage**

// Import express

const express = require('express');

// Initialize the app

const app = express();

// Define a route

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

  res.send('Welcome');

});

// Start the server

app.listen(3000, () => {

  console.log('Server is running on http://localhost:3000');

});

**STEP5:- run the application and see the output**

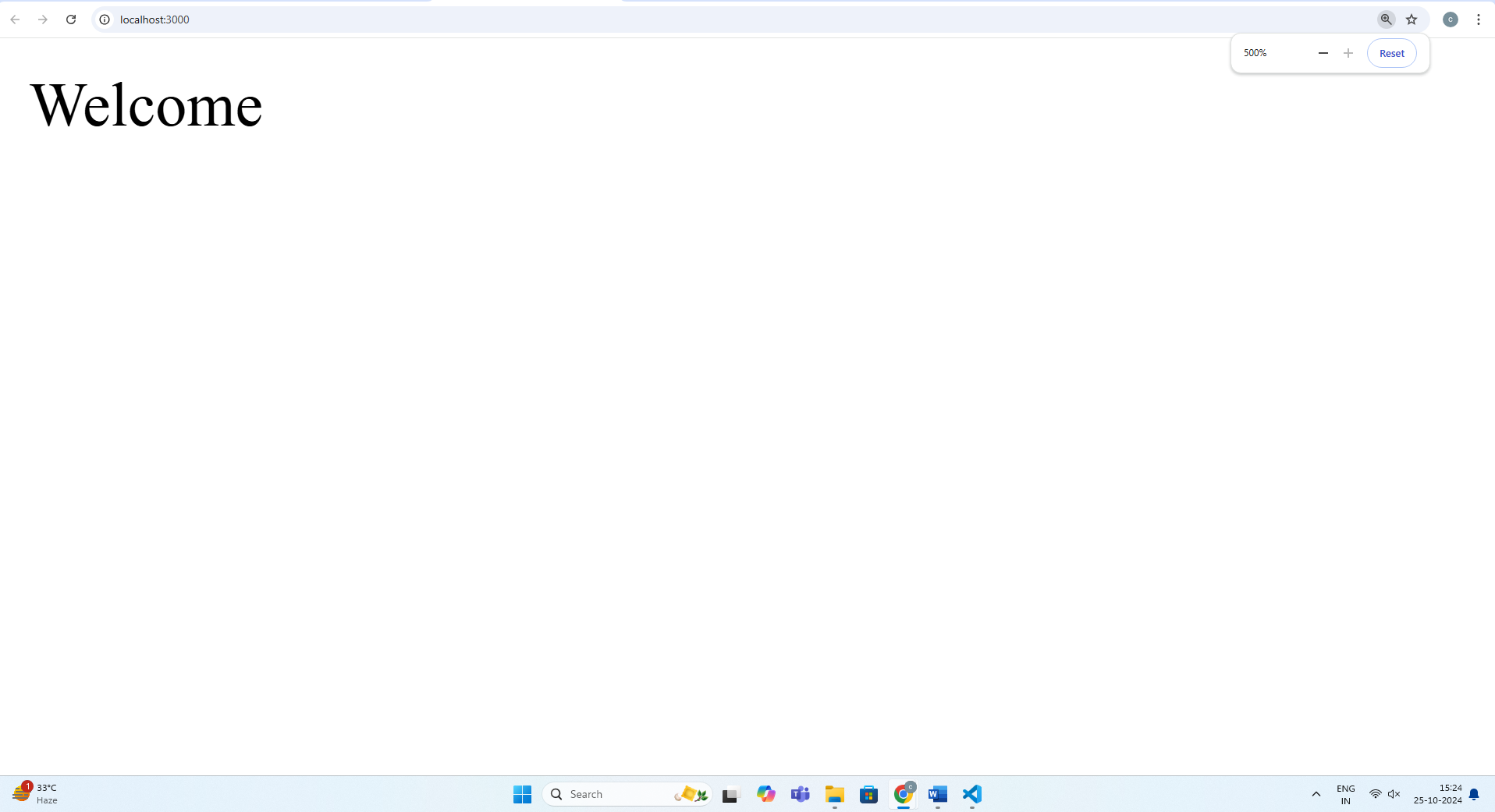
1. **run the application using**

**node app.js**

A screenshot of a computer

Description automatically generated

1. **check the output**

****