Before reading past below instructions:

1. Create an account in Github using your name in this format: lastname\_firstname\_section
2. Request access to [Lycevm<3Alabang · GitHub](https://github.com/Lycevm-3Alabang)
3. Upload this file ON YOUR GITHUB ACCOUNT with answer under the title / file name : E3\_Assessment\_\_[Section]\_[LastnameFirstName]  
   example: E3\_Assessment\_\_BSCS32E1\_AlamoNinoFrancisco

Help: [Get started with GitHub documentation - GitHub Docs](https://docs.github.com/en/get-started)

**Sample Assessment for Introduction to Programming**

This assessment is designed to evaluate your understanding of basic programming concepts in C#, HTML, CSS, and JavaScript.

Instructions: Read each question carefully and provide complete and clear answers. Avoid multiple-choice format responses. Focus on demonstrating your understanding through code, explanations, and discussions.

**Part 1: C# (30 points)**

(10 points) Write a C# program that calculates the area of a triangle given its base and height. Include user input for both values and display the calculated area.

**Answer:**

A computer screen with text and images

Description automatically generated

* **So here Is my code wherein it just a basic code for the finding the area of the triangle by using an arithmetic operators to find the area of it, and by using a console writeline to print out the output and by input of the user.**

**(10 points) Declare an array of 5 integers and fill it with values based on a user-defined formula (e.g., n^2). Then, print the largest element in the array.**

**Answer:**

**A computer screen shot of a program

Description automatically generated**

* **So heres my code wherein it display the largest element in the array Im just using foreach loop to easily comprehend my code.**

**(10 points) Implement a simple for loop that iterates from 1 to 10 and prints each number along with its square root.**

**Answer:**

**A computer screen shot of a program

Description automatically generated**

* **Heres my code displaying the number from 1-10 with its own square root. By using a for loop and incrementing it to display the output of it**

**Part 2: HTML, CSS, and JavaScript (30 points)**

**HTML (10 points):** You are provided with the following incomplete HTML code snippet:

**HTML**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>My Website</title>**

**</head>**

**<body>**

**<h1>Welcome to...</h1>**

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

**<ul>**

**<li>Item 1</li>**

**<li>Item 2</li>**

**</ul>**

**</body>**

**</html>**

Complete the code snippet by adding the following elements:

An image within the <body> tag with a relevant src attribute.

An ordered list (<ol>) with three items.

A hyperlink within a <p> tag that points to an external website.

A CSS styling rule using an inline style attribute to change the font color of the <h3> heading.

CSS (10 points): Create a CSS stylesheet that defines the following styles:

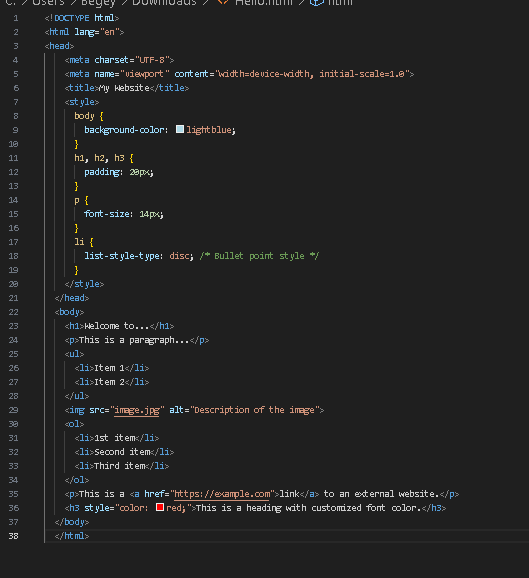
Change the background color of the body element to light blue.

Apply a padding of 20px to all headings (h1, h2, h3).

Set the font size of the <p> tag to 14px.

Make the list items (li) have a bullet point style instead of the default numbers.

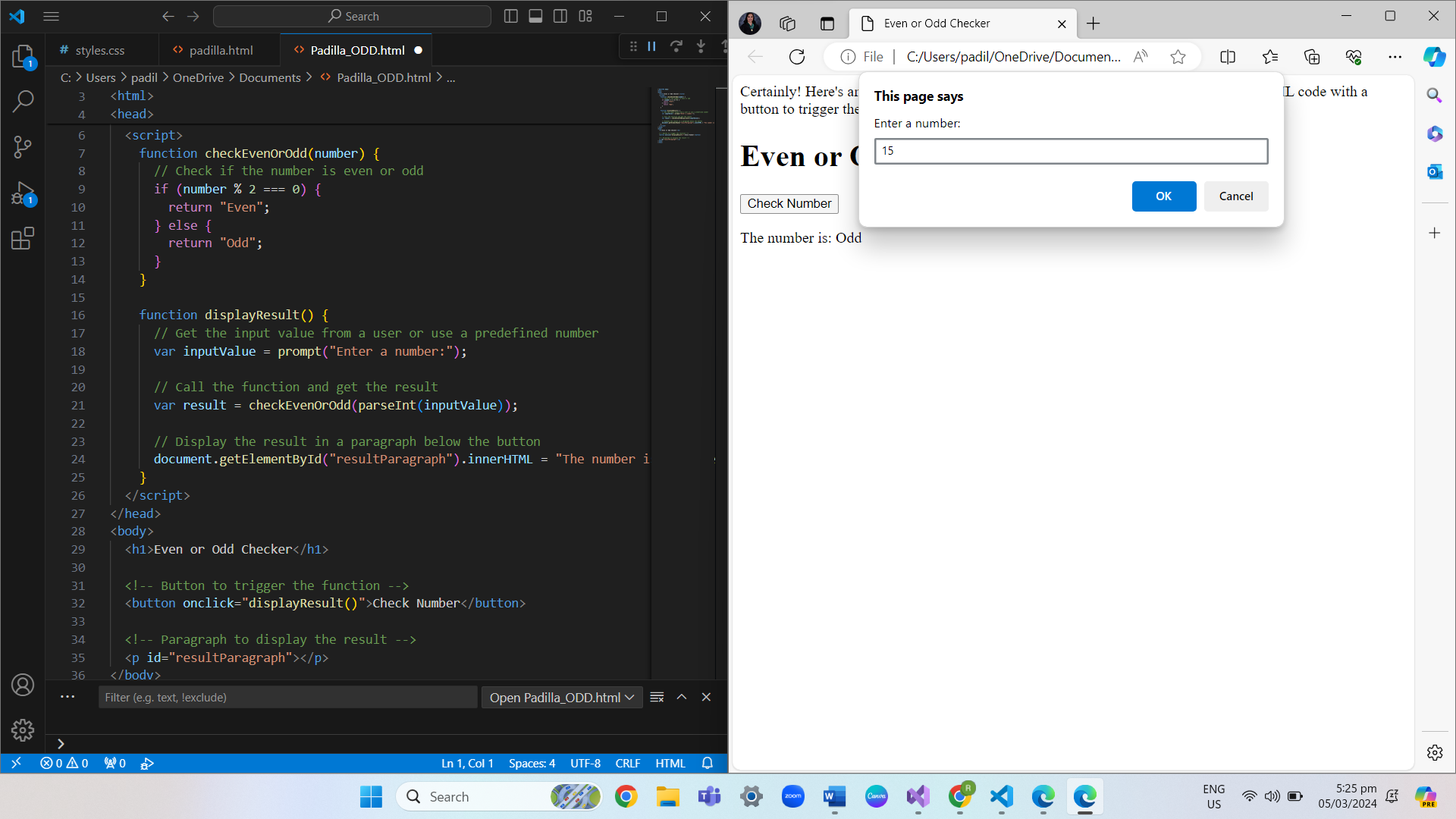
**Answer:**

****

* **Heres my answer providing the instructions or snippets below on the html. By some adding some providing a code snippet it will change the display of this html compare to the unfinished html so by adding some css it will change the color, size and etc of the html for having it’s a design and by following the instructions it will change it into simple website with a design.**

**JavaScript (10 points):** Write a JavaScript function that takes a number as input and returns a string indicating whether the number is even or odd. Then, add a button to your HTML page that, when clicked, calls this function and displays the result (even or odd) in a paragraph element below the button.

**Answer:**



* This code indicates whether the number being entered is even or odd. Utilizing HTML and a JavaScript script, it will generate a webpage with the ability to show the right response.

**Part 3: Essay Question (40 points)**

Discuss the importance of object-oriented programming (OOP) concepts in software development. Explain the key principles of OOP (encapsulation, inheritance, polymorphism, abstraction) and provide examples of how they can be used to create more efficient, maintainable, and reusable code. Include real-world scenarios or cases where OOP is particularly valuable.

**Answer:**

**Encapsulation** refers to the bundling of data and methods that operate on that data into a single unit, known as a class. By encapsulating data, you hide the internal state of an object and only expose the necessary functionality through methods. This helps in reducing complexity and preventing unintended manipulation of data.

**Inheritance**: Inheritance allows a class (subclass) to inherit properties and behavior (methods) from another class (superclass). This promotes code reuse and allows for creating a hierarchy of classes, where subclasses can specialize or extend the functionality of the superclass.

**Polymorphism**: Polymorphism allows objects of different classes to be treated as objects of a common superclass. It enables flexibility in code design and allows for writing code that can work with objects of various types without needing to know their specific class. Polymorphism is often achieved through method overriding, where subclasses provide their own implementation of methods defined in the superclass.

**Abstraction**: Abstraction involves hiding the complex implementation details and showing only the essential features of an object. It allows developers to focus on what an object does, rather than how it does it, which improves code readability and reduces complexity. Abstraction is often achieved through abstract classes or interfaces, which define a blueprint for classes to follow.

* **Software development teams**: OOP promotes modular design, allowing teams to work on different components of a system independently. Classes encapsulate functionality, reducing dependencies and facilitating collaboration.
* **Graphical user interface (GUI) development**: OOP concepts like inheritance and polymorphism are used to create reusable GUI components. For example, a button class can inherit from a generic UI component class, and polymorphism allows different types of buttons (e.g., submit, cancel) to share common behavior.
* **Database management systems**: OOP principles are applied in ORM (Object-Relational Mapping) frameworks, where database tables are mapped to classes, and objects represent rows in the database. Encapsulation helps in managing data integrity, inheritance allows for modeling relationships between entities, and polymorphism enables querying and manipulating data in a uniform way.
* **Game development**: OOP facilitates the creation of game objects with distinct properties and behaviors. For example, a game engine may define a base class for game entities (e.g., characters, enemies) with common functionality, and subclasses can extend or override this functionality to create different types of entities.

Points Distribution:

Each part carries equal weight (30 points).

Code clarity, functionality, and explanations will be considered in grading.

The essay question focuses on understanding and application of OOP concepts.