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:**

using System;

class Program

{

static void Main()

{

Console.WriteLine("Enter the base of the triangle:");

double baseLength = double.Parse(Console.ReadLine());

Console.WriteLine("Enter the height of the triangle:");

double height = double.Parse(Console.ReadLine());

double area = CalculateTriangleArea(baseLength, height);

Console.WriteLine("The area of the triangle is: " + area);

}

static double CalculateTriangleArea(double baseLength, double height)

{

return 0.5 \* baseLength \* height;

}

}

**(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:**

using System;

class Program

{

static void Main()

{

int[] array = new int[5];

for (int i = 0; i < array.Length; i++)

{

array[i] = (i + 1) \* (i + 1); // n^2 formula

}

Console.WriteLine("Array elements:");

foreach (int num in array)

{

Console.WriteLine(num);

}

int max = array[0];

for (int i = 1; i < array.Length; i++)

{

if (array[i] > max)

{

max = array[i];

}

}

Console.WriteLine("The largest element in the array is: " + max);

}

}

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

**Answer:**

using System;

class Program

{

static void Main()

{

Console.WriteLine("Number\tSquare Root");

for (int i = 1; i <= 10; i++)

{

double squareRoot = Math.Sqrt(i);

Console.WriteLine($"{i}\t{squareRoot}");

}

}

}

**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.

**ANSWER:**

<!DOCTYPE html>

<html>

<head>

<title>KSlady</title>

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

</head>

<body>

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

<h3 style=” color: violet;”> This is a blue heading</h3>

<p>This is a paragraph with a <a href= https://www.facebook.com/people/KSLady-keepstylinlady >hyperlink to an external website</a></p>

<ul>

<li>Item 1</li>

<li>Item 2</li>

</ul>

<img src= “example.jpg” alt= “Image” />

<ol>

<li>Item 1</li>

<li>Item 2</li>

<li>Item 3</li>

</ol>

</body>

</html>

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:**

/\* Style.css \*/

body {

background color: blue;

}

h1, h2, h3 {

padding: 20px;

}

p {

font-size: 14px;

}

li {

list-style-type: disc;

}

**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:**

<!DOCTYPE html>

<html>

<head>

<title>Even/Odd Checker</title>

</head>

<body>

<button onclick="checkEvenOdd()">Check Even/Odd</button>

<p id="resultParagraph"></p>

<script>

function checkEvenOdd() {

var number = parseInt(prompt("Enter a number:"));

var resultParagraph = document.getElementById("resultParagraph");

if (isNaN(number)) {

resultParagraph.textContent = "Invalid input. Please enter a valid number.";

} else {

if (number % 2 === 0) {

resultParagraph.textContent = number + " is even.";

} else {

resultParagraph.textContent = number + " is odd.";

}

}

}

</script>

</body>

</html>

**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:**

Object-oriented programming (OOP) fundamentally transforms the way software is developed by encapsulating data and behaviors within objects, fostering code organization and maintainability. Through the principles of inheritance and composition, OOP enables developers to efficiently reuse code, constructing classes that either build upon existing ones or compose functionalities from simpler components.

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.  
  
**The Key Principles of OOP:**

* **Encapsulation:**

EXAMPLE: Email class with methods like send (), receive (), and delete (), encapsulating email data and operations while hiding the underlying network protocols and message handling mechanisms.

* **Inheritance:**

EXAMPLE: Animal superclass with subclasses such as Dog, Cat, and Bird, inheriting common attributes and behaviors like eat (), sleep (), and make Sound () from the superclass.

* **Polymorphism:**

EXAMPLE: Shape superclass with subclasses Square, Circle, and Triangle, each implementing a calculate Area () method differently based on the specific geometric properties of the shape.

* **Abstraction:**

EXAMPLE: Database Connection class with methods like connect (), execute Query (), and close Connection (), abstracting away the details of database connection management and query execution.

**Real-world scenarios or cases where OOP is particularly valuable:**

* Mobile App Development: OOP is widely employed in mobile app development, particularly in platforms like Android (Java or Kotlin) and iOS (Swift or Objective-C). It helps in creating modular, reusable components for building complex mobile applications with ease.
* Enterprise Software Systems: OOP is extensively utilized in building large-scale enterprise software systems. It enables developers to encapsulate business logic into objects, promoting code organization, maintainability, and scalability.
* E-commerce Platforms: OOP is valuable in e-commerce platforms for managing product catalogs, user accounts, shopping carts, and order processing. It allows for the creation of reusable components to handle various aspects of e-commerce functionality efficiently.
* Financial Systems: OOP is crucial in financial systems for modeling financial instruments, transactions, and market data. It facilitates the implementation of complex calculations, risk analysis, and reporting functionalities in a structured and maintainable manner.
* Healthcare Systems: OOP is beneficial in healthcare systems for managing patient records, medical procedures, and healthcare workflows. It helps in organizing healthcare-related data and functionalities into reusable objects, ensuring data integrity and system reliability.
* Educational Software: OOP is employed in educational software for developing interactive learning materials, quizzes, and educational games. It allows for the creation of reusable components that facilitate personalized learning experiences for students.
* Content Management Systems (CMS): OOP is used in CMS platforms like Word Press, Drupal, and Joomla for managing content, user permissions, and website customization. It enables developers to create modular plugins and themes, extending the functionality of the CMS efficiently.
* Internet of Things (IoT) Applications: OOP is crucial in IoT applications for modeling connected devices, sensors, and actuators. It helps in building scalable and interoperable IoT solutions by encapsulating device functionalities into reusable objects.
* Telecommunication Systems: OOP is employed in telecommunication systems for managing network infrastructure, communication protocols, and subscriber services. It facilitates the development of reliable and scalable telecommunication solutions by organizing code into reusable and manageable components.
* Supply Chain Management Systems: OOP is valuable in supply chain management systems for modeling inventory, orders, shipments, and logistics operations. It enables developers to create modular components for tracking and managing various aspects of the supply chain efficiently.