Portfolio (Week-1)

In our first lab for Web and Mobile Application, we focused on setting up the development environment necessary for MERN (MongoDB, Express, React, Node.js) stack development. This included installing key tools such as NodeJS, MongoDB, Visual Studio Code, and POSTMAN. These tools are fundamental for building and testing modern web applications.

After the installations, our tutor provided an overview of the lecture topics. They explained the concepts of web applications and the history of their evolution, emphasizing how web technologies have progressed over time. We also learned about the N-tier architecture, a design pattern used for creating scalable and maintainable applications. This architecture divides an application into three layers: the presentation layer, business logic layer, and data layer. Additionally, the tutor introduced us to the concept of stacks and their significance in web development. The MERN stack, for instance, integrates various tools and technologies to create seamless web applications.

Lab Exercise: Arithmetic Operations Program

Following the lecture, we moved on to our Week 1 lab exercise. The task was to write a program that could perform addition, subtraction, multiplication, and division of two or more numbers. Below is the step-by-step process of how I approached this task:

Defining Variables

To begin, I created two variables to store the numbers that would be used for performing arithmetic operations.A screenshot of a computer

Description automatically generated

Formulating Operations

Next, I defined the formulas for each operation (addition, subtraction, multiplication, and division). This step involved writing simple expressions using JavaScript syntax.

A black screen with a black border

Description automatically generated

Printing Results

To display the results of the operations, I added print statements that would output the computed values for each operation when the program runs.A screen shot of a computer program

Description automatically generated

Printing Results

To display the results of the operations, I added print statements that would output the computed values for each operation when the program runs.A screen shot of a computer

Description automatically generated

Viewing the Output

After running the program, the results of all the arithmetic operations were displayed in the terminal window, confirming that the code worked as expected.A black screen with white text

Description automatically generated

**Learning Objectives**

To understand the process of setting up a development environment for MERN stack applications.

To become familiar with fundamental web development tools such as NodeJS, MongoDB, Visual Studio Code, and POSTMAN.

To grasp the basics of web application architecture and the evolution of web technologies.

To gain hands-on experience writing and executing a simple JavaScript program to perform arithmetic operations.

**Key Points**

Setting up a proper development environment is crucial for efficient coding and testing.

The N-tier architecture divides responsibilities across layers, making applications more maintainable.

Practical coding exercises help reinforce theoretical concepts, like basic JavaScript operations in this lab.

**Problems Faced**

Issue Running the Code: Initially, when I tried to execute the program using the node index.js command, I encountered an error. The terminal indicated that the file index.js could not be found.

Solution: I realized that I had mistakenly saved the file with a different name. I renamed the file to index.js and retried the command, which resolved the issue.

Syntax Error: While writing the formulas for arithmetic operations, I forgot to include a closing parenthesis in one of the expressions. This caused the program to throw a syntax error when I attempted to run it.

Solution: I carefully reviewed my code and added the missing parenthesis. To avoid similar issues in the future, I decided to use Visual Studio Code’s built-in linting feature, which highlights syntax errors as I type.

Confusion with Division Operation: During the calculation of division, I initially used an incorrect operator, which resulted in an unexpected output.

Solution: I referred to the JavaScript operator documentation and corrected the operator from % (modulus) to / (division).

By addressing these challenges, I gained a deeper understanding of the importance of attention to detail in coding and the usefulness of debugging tools.

This first lab was a great introduction to the MERN stack environment and served as a foundational step in my journey toward developing web and mobile applications.