NAME: TANKISO MASOEBE

WEB DESIGN WEEK 9

Scenario Analysis:

In a web development project, JavaScript is used to validate a user registration form before the data is sent to the server. The form includes fields such as username, email, password, and phone number. When a user clicks the "Submit" button, a JavaScript function checks whether all required fields are filled correctly. For example, it ensures that the email address contains "@" and ".com", the password is at least eight characters long, and the phone number contains only digits. If any errors are found, JavaScript displays error messages next to the respective input fields, prompting the user to correct them. This process prevents the submission of incomplete or incorrect data, saving time and reducing server load. By performing real-time validation, JavaScript enhances user experience, improves data accuracy, and ensures the website functions smoothly without unnecessary server requests.

Concept Research:

JavaScript operators are symbols used to perform operations on values and variables. They are essential for calculations, comparisons, and logic in programming. The main types include **arithmetic operators** (like `+`, `-`, `*`, `/`, `%`) for math operations, **assignment operators** (like `=`, `+=`, `-=`) for assigning values, and **comparison operators** (like `==`, `===`, `!=`, `>`, `<`) for comparing values. **Logical operators** (`&&`, `||`, `!`) combine or invert conditions, while **ternary operators** (`condition ? value1 : value2`) provide quick decision-making. JavaScript also includes **bitwise**, **string**, and **type** operators, allowing developers to build interactive and dynamic web applications efficiently.

Tool Practice:

Script:

```
let number = prompt("Enter a number:");
if (number % 2 === 0) {
  alert(number + " is even.");
} else {
  alert(number + " is odd.");
}
```

Reflection:

This simple JavaScript script checks whether a user's input number is even or odd. I learned how to use the `prompt()` function to get user input, the `if-else` statement for decision-making, and the `alert()` function to display results. It helped me understand how JavaScript interacts with users through the browser and how logic can control program flow. Writing and testing this script improved my confidence in using conditional statements and operators, which are important for creating interactive and responsive web pages.

Application Practice:

Code:

```
<button id="myButton">Click Me</button>
<p id="message"></p>

<script>
document.getElementById("myButton").addEventListener("click", function() {
  document.getElementById("message").innerText = "Button clicked!";
});
</script>
```

Explanation:
This code creates a button that displays a message when clicked. The `addEventListener()` method detects the "click" event and runs a function that changes the text inside the paragraph with the ID `message`. This exercise helped me understand how JavaScript handles user interactions on a webpage. It showed me the importance of event listeners in making web pages dynamic and responsive. I learned that combining HTML elements with JavaScript allows users to interact directly with content, improving both usability and user experience.