**JOBSHEET 3**

Web Design and Programming

Agna Putra Prawira (2341720065 )

Class 2I / 01

**STUDY PROGRAM D-IV TEKNIK INFORMATIKA**

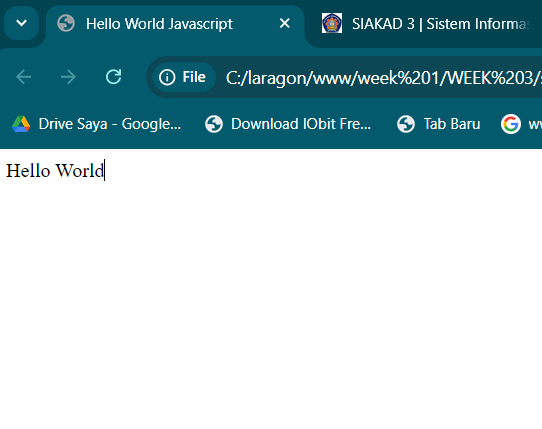
**INFORMATION TECHNOLOGY MAJOR**

**POLITEKNIK NEGERI MALANG**

Jl. Soekarno Hatta No. 9 Malang 65144

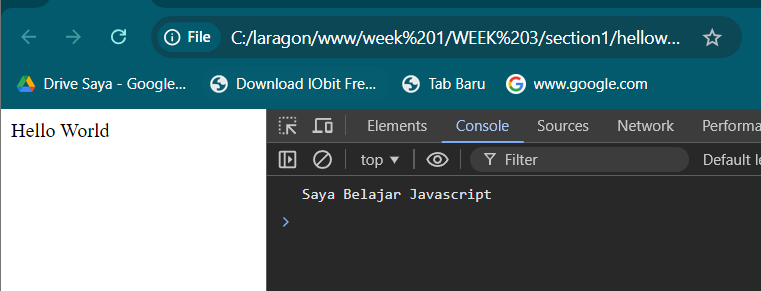
**Practical Section 1: Creating a Javascript Program**

1. Observe what happens to the browser, then record your observations (Question No. 1)



The browser shows **"Hello World"** on the webpage itself. The developer console shows **"Saya Belajar Javascript"** as a log message.

2.Observe what happens in the Console tab, then record your results! (Question No. 2)



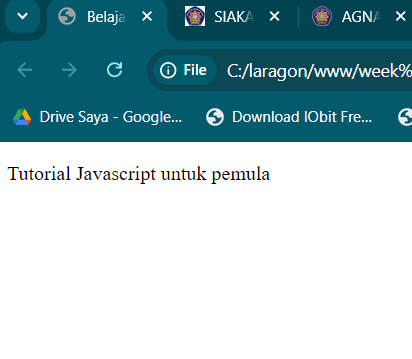
The Console tab displays the log message **"Saya Belajar Javascript"** as expected. No errors or additional messages are shown in the console.

3. Why do you think the command is not displayed? (Question No. 3)

-The **console.log()** function is used to output messages to the browser's **console**, not directly to the webpage itself. This method is primarily used for debugging and testing purposes by developers to see specific information while interacting with the web page.

**Practical Section 2: How to Write Javascript Code in HTML**

4.Observe what happens to the browser? Record your observations (Question No. 4)



-The message **"Hello JS dari Head"** is logged to the **JavaScript console** because of the console.log("Hello JS dari Head"); statement inside the <head> tag. You can view this output in the Console tab of the developer tools.

-The visible content on the webpage is the <p> element with the text:

Tutorial Javascript untuk pemula

5.Which do you think is better, written in the or tag? (Question No. 5)

**JavaScript in the <head>:**

* The code in the <head> is executed **before the page's content (HTML) is fully loaded**. This is useful for tasks that must run as early as possible, such as setting global variables or initializing configurations.

**JavaScript in the <body>:**

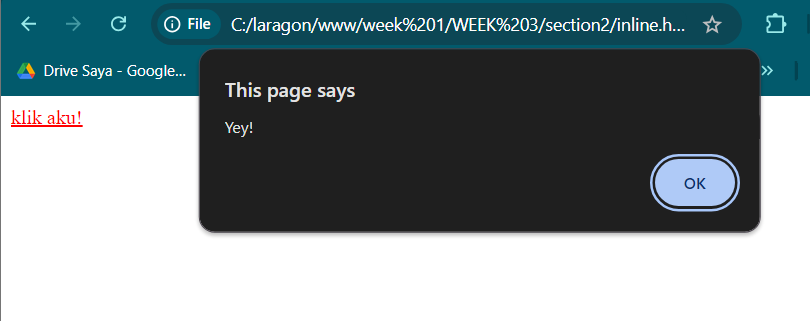
* The code in the <body> tag is executed **after the page's content is loaded**. This is generally better for scripts that need to manipulate or interact with page elements because those elements are already available.

Conclusion:

For **initialization scripts** that don't need to interact with the page's content, the <head> tag is fine.For scripts that need to **modify or access DOM elements**, it's better to place them **at the bottom of the <body>** tag to ensure all content is loaded before the script runs.

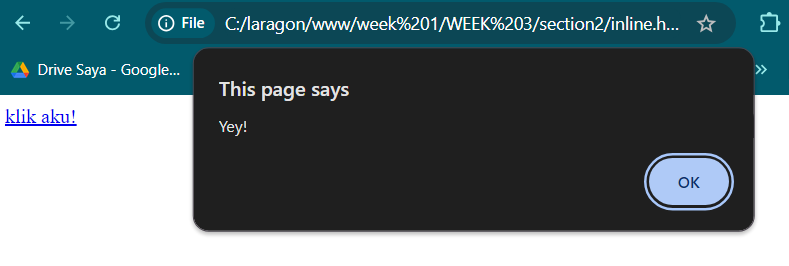
6. Observe what happens to the browser! Record your observations (Question No. 6)

-First Code (onclick attribute):



* When the link is clicked, a pop-up alert box appears with the message **"Yey!"**.
* The href="#" keeps the user on the same page, so after the alert is closed, the page does not refresh or navigate.
* The event is triggered by the onclick attribute, which specifies what to do when the link is clicked.

-Second Code (JavaScript in href):



* Similarly, when the link is clicked, a pop-up alert box with the message **"Yey!"** appears.
* Instead of using the onclick attribute, this code directly places the JavaScript in the href attribute, executing the alert() function when the link is clicked.

7. What is the difference between the two program codes (Question No. 7)

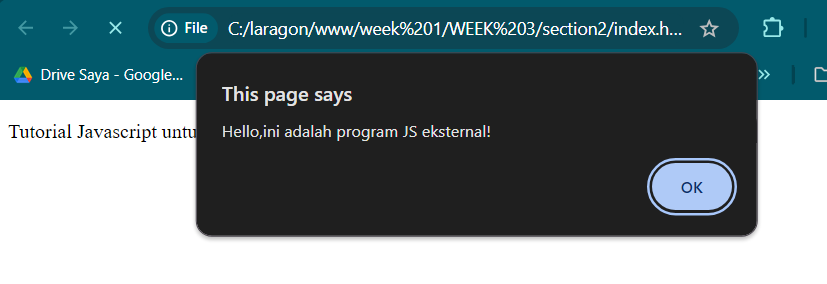
1. **First Code (Using onclick Event Handler):**

* The JavaScript is placed inside the **onclick attribute**.
* This is more structured and uses the event handler (onclick) to trigger the function.
* It keeps the href="#" for navigation, which is good for separation of concerns (HTML structure vs. JavaScript behavior).
* This approach is **better for accessibility** and maintainability.

2. **Second Code (Using javascript: in href):**

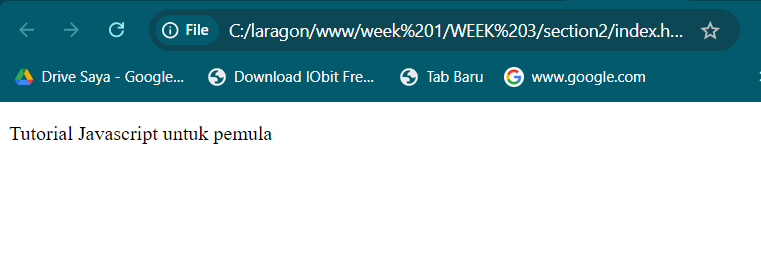
* The JavaScript is placed directly in the **href attribute** with the javascript: protocol.
* This method is an older, less clean way of handling events and embeds the JavaScript directly into the link’s navigation.
* It mixes behavior (executing JavaScript) with the link's natural function (navigation), which can lead to potential issues like broken accessibility features and security vulnerabilities (e.g., Cross-Site Scripting).

8. Observe what happens to the browser! Record your observations (Question No. 8)



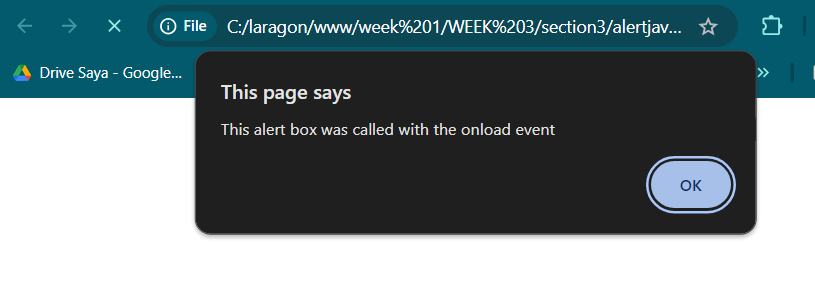
The browser correctly loads the external JavaScript file and displays the alert with the message "Hello, ini adalah program JS eksternal!".

9. Move kode-program.js file to another folder, what will happen if the javascript file is in a different folder? Observe and record your observations (Question No. 9)



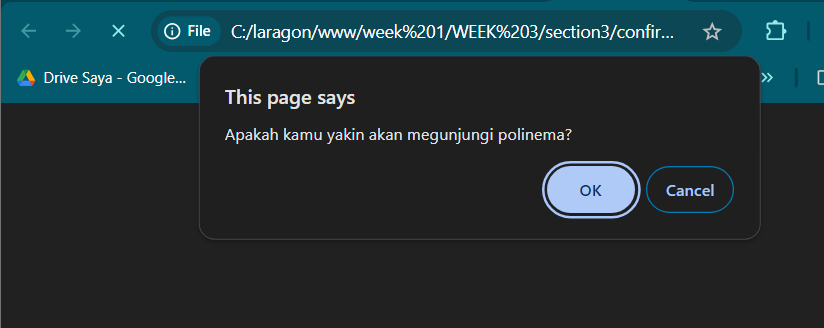
When the kode-program.js file is moved to a different folder without updating the HTML, the JavaScript fails to load, no alert is displayed, and an error is logged in the console indicating that the file could not be found.

10. Record your observations (Question No. 10)



Upon loading the page, an alert box with the message "This alert box was called with the onload event" is displayed, triggered by the onload event in the <body> tag. There is no other visible content on the page, and the script runs without errors.

11. Record your observations (Question No. 11)



 **Confirmation Dialog**:

* The confirm function is used to ask the user if they are sure about visiting Polinema.
* The confirm dialog returns a boolean value based on the user's response.

 **Redirection**:

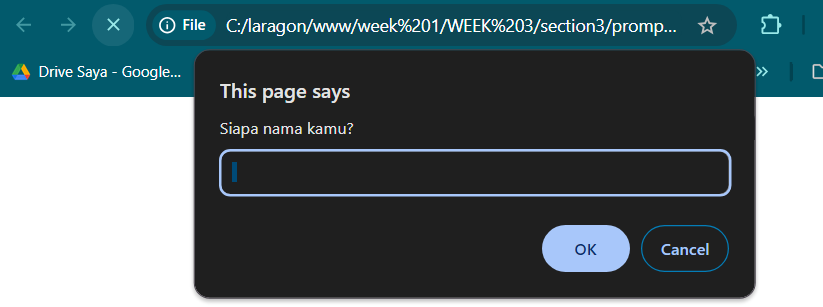
* If the user confirms (true), they are redirected to "<https://www.polinema.ac.id>" using window.location.

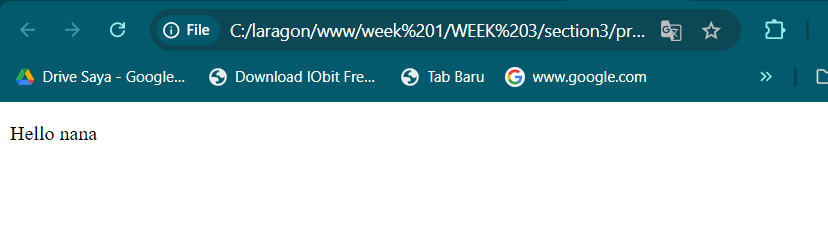
 **User Cancellation**:

* If the user declines (false), the page displays the message "Baiklah, tetap disini saja ya :) " using document.write.

 **Improvement Suggestions**:

* Instead of document.write, consider updating a specific element's content or using other methods like alert for user feedback.

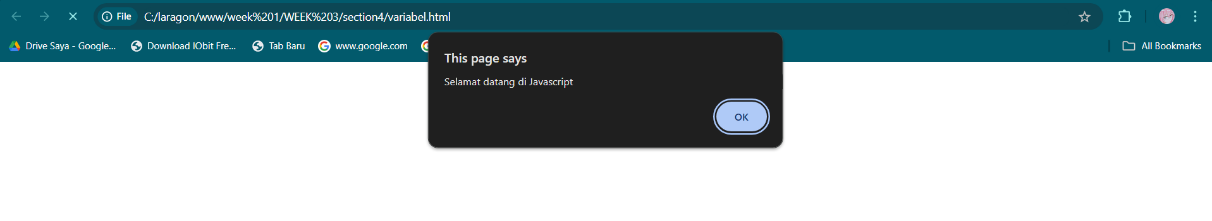
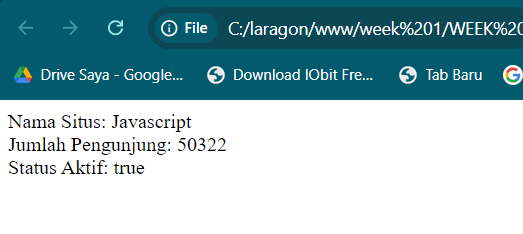
12. Record your observations (Question No. 12)



 When the user provides their name, the message "Hello [name]" is displayed on the webpage.

 If the user cancels the prompt, document.write will still attempt to display a message, potentially leading to undesirable results like showing "Hello null" if the user does not input anything.

13 Record your observations (Question No. 13)



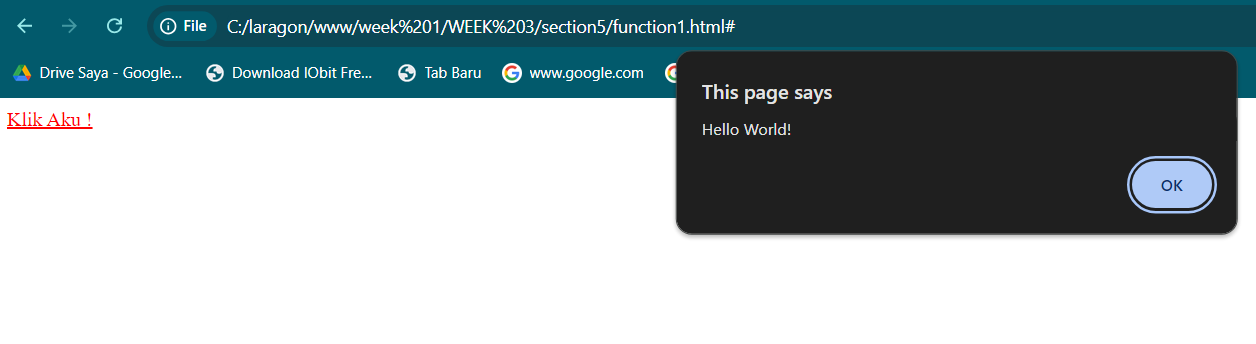
 The script creates three variables (name, visitorCount, and isActive).

 It displays a welcome message using alert().

 It then outputs the variable values directly into the HTML body using document.write().

 There is a typo in the closing HTML tag (</htm1> should be </html>).

14 Record your observations (Question No.14)

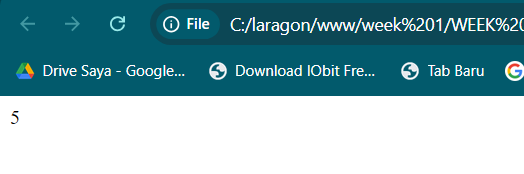


 **Function**: sayHello is an arrow function that shows an alert box with the message "Hello World!" when called.

 **Anchor Tag**: The <a> tag triggers the sayHello() function when clicked, showing the alert.

 **Fixes**: The code fixes include correcting the DOCTYPE, fixing typos in the <html> tags, removing the syntax error in the JavaScript string, and ensuring the anchor tag is properly closed.

15. Record your observations (Question No.15)



 A new HTML file is created that defines a function named total in JavaScript.

 The function total takes two parameters, numberA and numberB, and returns their sum.

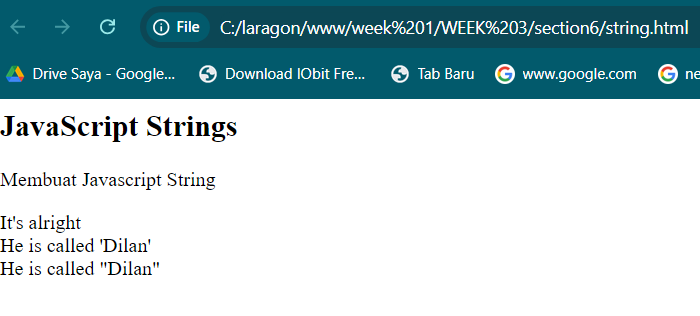
 In the body of the HTML, document.write(total(2, 3)); is used to call the function total with the values 2 and 3. The result, which is 5, is written directly to the HTML page.

16. Record your observations (Question No. 16)



When this code runs, it will display the text "John" in the paragraph with the id="demo" because the variable x is eventually assigned the string value "John"

17. Record your observations (Question No. 17)



 **Strings**: The code creates three variables:

* answer1 = "It's alright" (double quotes, single quote inside).
* answer2 = "He is called 'Dilan'" (double quotes, single quotes inside).
* answer3 = 'He is called "Dilan"' (single quotes, double quotes inside).

 **Display**: It uses document.getElementById("demo").innerHTML to display the concatenated string values in the HTML with line breaks (<br>).

18. Record your observations (Question No. 18)



 **Booleans**: A Boolean can either be true or false.

 **Comparison Operators**: The == operator checks if two values are equal. If they are, it returns true; otherwise, it returns false.

 **Display in HTML**: Using document.getElementById().innerHTML, the result of the comparisons is dynamically inserted into the HTML page.

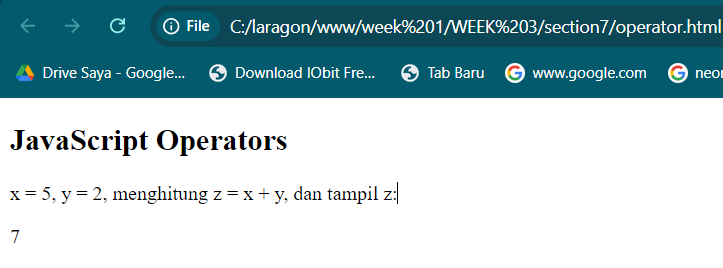
19. Record your observations (Question No. 19)

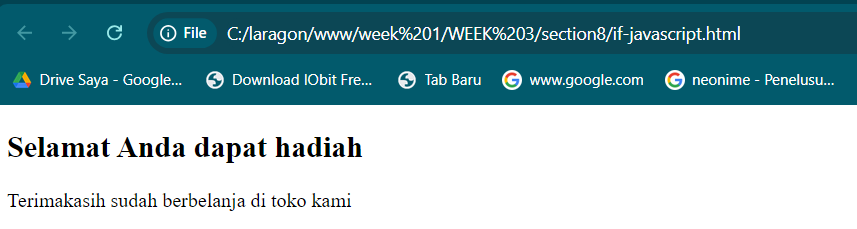


1. **HTML Structure**:
   * <html> and <body>: Define the basic structure.
   * <h2> and <p>: Display text on the page.
2. **JavaScript Code**:
   * var cars = ["Satu", "Dua", "Tiga"];: Creates an array with three string elements.
   * document.getElementById("demo").innerHTML = cars[0];: Sets the content of the paragraph with the id "demo" to the first item in the array, "Satu".

When the HTML is loaded in a browser, it shows "Satu" in the specified paragraph.

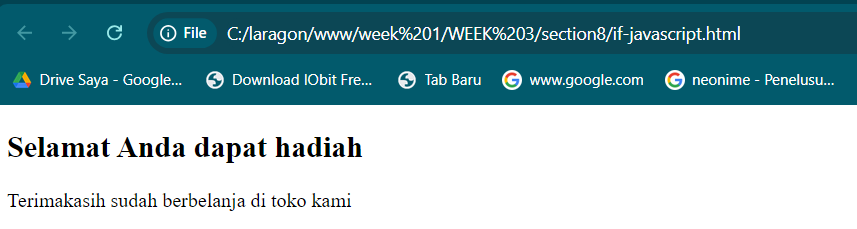
20. Record your observations (Question No.20)



1. **HTML Structure**:
   * <html> and <body>: Basic HTML structure.
   * <h2>: A heading that describes the content.
   * <p>: A paragraph explaining the operation being performed.
   * <p id="demo"></p>: An empty paragraph with the id "demo" where the result will be displayed.
2. **JavaScript Code**:
   * var x = 5;: Defines a variable x with a value of 5.
   * var y = 2;: Defines a variable y with a value of 2.
   * var z = x + y;: Adds x and y and stores the result in z.
   * document.getElementById("demo").innerHTML = z;: Sets the inner HTML of the paragraph with id "demo" to the value of z (which is 7).

**Outcome**: When you open this HTML file in a browser, it will display "7" in the paragraph with the id "demo".

21. Record your observations (Question No. 21)



 Prompts the user for their shopping total.

 If the total is over 30,000, it shows a prize message.

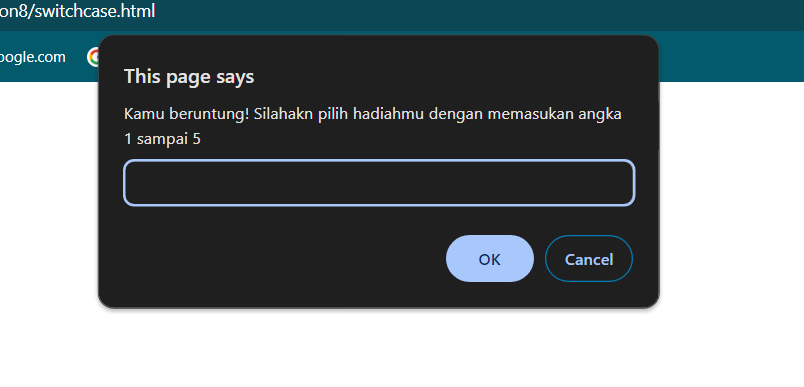
 Always shows a thank-you message.

22. Record your observations (Question No. 22)



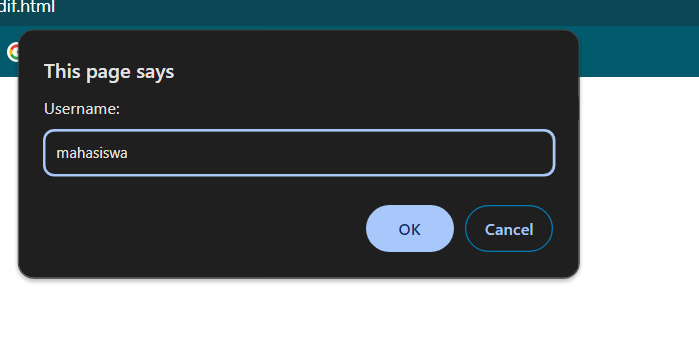
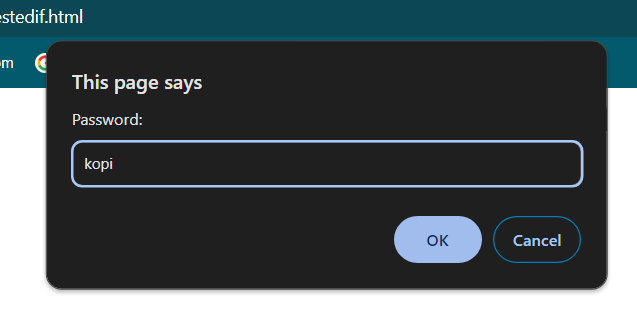
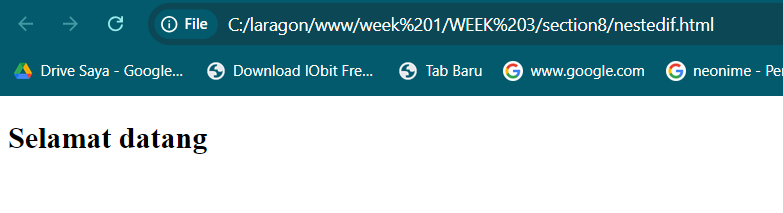
 Prompts the user to enter a password. If the password is "teh", it displays a welcome message. Otherwise, it shows an incorrect password message. Always displays a thank-you message.

23. Record your observations (Question No. 23)



 Prompts the user to pick a number between 1 and 5. Uses a switch statement to assign a prize based on the input. Displays an error message for invalid choices and a thank-you message for valid choices.

24. Record your observations (Question No.24)



 Prompts the user for a username and password.

 Uses nested if statements to check:

* If the username is "mahasiswa" and the password is "kopi", displays a welcome message.
* If the password is incorrect, shows an error message.
* If the username is incorrect, displays a different error message.

25. Record your observations (Question No. 25)

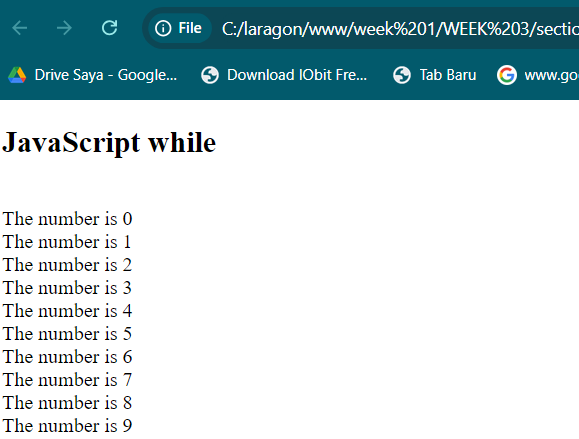


 Uses a for loop to iterate from 0 to 4.

 Concatenates a string "The number is X" for each value of i (where X is the current loop index) with a line break (<br>).

 Sets the inner HTML of the paragraph with the id "demo" to the accumulated string.

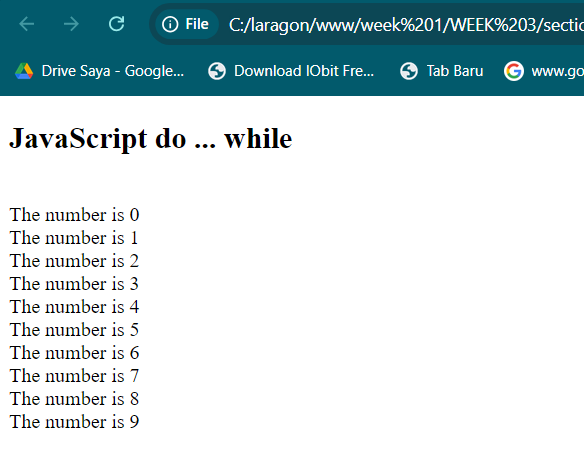
26. Record your observations (Question No. 26)



* + Uses a while loop to iterate while i is less than 10.
  + Concatenates a string "The number is X" (where X is the current value of i) with a line break (<br>) to the text variable.
  + Increments i by 1 on each iteration.
  + Sets the inner HTML of the paragraph with the id "demo" to the accumulated string.

The result is that the paragraph displays numbers 0 through 9, each on a new line.

27. Record your observations (Question No.27)



* + Uses a do ... while loop to ensure that the loop runs at least once.
  + Concatenates a string "The number is X" (where X is the current value of i) with a line break (<br>) to the text variable.
  + Increments i by 1 on each iteration.
  + Continues looping while i is less than 10.
  + Sets the inner HTML of the paragraph with the id "demo" to the accumulated string.

The result is that the paragraph displays numbers 0 through 9, each on a new line.

LINK GITHUB :

<https://github.com/Agnaputra/Desain-pemrograman-web>