Operators Tasks

Basic Math Operations

Objective: Write a program to perform basic arithmetic operations.

1. Take two numbers as input from the user.
2. Perform addition, subtraction, multiplication, and division.
3. Display the results.

Example:
Input: 5, 3
Output:

- Sum: 8
- Difference: 2
- Product: 15
- Quotient: 1.67

Logical Condition

Objective: Check if a number is greater than 10 and divisible by 2.

1. Take a number as input.
2. Use logical operators to determine if the conditions are met.
3. Display the result in the console.

Ternary Operator Task

Objective: Determine if a number is positive or negative using a ternary operator.

1. Take a number as input.
2. Use a ternary operator to check if it's positive or negative.
3. Display the result.

Example:
Input: -7
Output: Negative

Conditions and Looping Tasks

Check Odd or Even

Objective: Write a program to check if a number is odd or even.

1. Take a number as input.
2. Use an if-else statement to determine if it's odd or even.
3. Display the result.

Example:
Input: 4
Output: Even

Grade System

Objective: Create a grade system using a switch statement.

1. Take a percentage input from the user.
2. Use a switch statement to assign grades based on the range:
   - 90–100: A
   - 80–89: B
   - 70–79: C
   - <70: F
3. Display the grade.

Print Multiplication Table

Objective: Print the multiplication table of a given number using a for loop.

1. Take a number as input.
2. Use a for loop to generate and display the table (1 to 10).

Example:
Input: 5
Output:

- 5 x 1 = 5
- 5 x 2 = 10
- ...
- 5 x 10 = 50

Count Digits in a Number

Objective: Count the digits in a number using a while loop.

1. Take a number as input.
2. Use a while loop to count the digits.
3. Display the count.

Example:

Input: `12345`

Output: `5`

Dialog Box Tasks

Welcome Alert

Objective: Show an alert when the page loads.

1. Create a program that displays: `Welcome to my website!`

User Confirmation

Objective: Use a confirm dialog box to prompt the user.

1. Ask the user if they want to continue.
2. Display:
   - `You chose to continue!` if they confirm.
   - `You canceled!` if they decline.

Prompt for Age

Objective: Validate the user's age using a prompt box.

1. Ask the user for their age.
2. If the input is a number >= 18, display: `You are eligible!`

Simple BMI Calculator

Objective: Create a BMI calculator using prompt boxes.

1. Prompt the user for height and weight.
2. Calculate BMI using the formula:
   $$\text{BMI} = \frac{\text{weight (kg)}}{\text{height (m)}^2}$$
3. Display the BMI in an alert box.

String Methods Tasks

Reverse a String

Objective: Reverse a string using string methods.

1. Take a string as input.
2. Use string methods to reverse it.
3. Display the result.

Example:
Input: `hello`
Output: `olleh`

Count Vowels

Objective: Count the number of vowels in a string.

1. Take a string as input.
2. Count and display the vowels.

Example:
Input: `javascript`
Output: `3`

Check Palindrome

Objective: Determine if a string is a palindrome.

1. Take a string as input.
2. Check if it reads the same backward as forward.
3. Display the result.

Example:
Input: `madam`
Output: `true`

Extract Initials

Objective: Extract initials from a full name.

1. Take a full name as input.

2. Return the initials in uppercase.

Example:
Input: `John Doe`
Output: `J.D`

Replace Words

Objective: Replace a specific word in a sentence.

1. Take a sentence and a word to replace as input.
2. Replace the word and display the new sentence.

Example:
Input: `I love programming`
Replace `programming` with `JavaScript`.
Output: `I love JavaScript`

Split Sentence into Words

Objective: Split a sentence into an array of words.

1. Take a sentence as input.
2. Split and display the words as an array.

Example:
Input: `Hello world`
Output: `['Hello', 'world']`

Remove Spaces

Objective: Remove all spaces from a string.

1. Take a string as input.
2. Remove all spaces and display the result.

Example:
Input: `I love JavaScript`
Output: `IloveJavaScript`

Find Character Frequency

Objective: Count the frequency of a specific character in a string.

1. Take a string and a character as input.
2. Count and display how many times the character appears.

Example:

Input: `hello` (Character: `l`)

Output: `2`