JavaScript Fundamentals – Part 1

Topic: Values and Variables:

- Declare variables called 'country', 'continent' and 'population' and assign their values according to your own country (population in millions)
- Log their values to the console.

Topic : Data Types :

- Declare a variable called 'isIsland' and set its value according to your country. The variable should hold a Boolean value. Also declare a variable 'language', but don't assign it any value yet
- Log the types of 'isIsland', 'population', 'country' and 'language' to the console.

- Create a string variable containing your full name.
- Write the string manipulation code using template literals to include variables inside strings.

5. **Basic Arithmetic Operations:**

- Declare two variables with numeric values.
- Perform and log the results of addition, subtraction, multiplication, division, and modulus operations.

Floating-Point Precision:

- Provide an example where adding two decimal numbers results in a precision error.
- Show how to solve or mitigate this issue using toFixed, parseFloat, or another method.

6. Boolean data type :

1. **Boolean Values:**

- Create variables with true and false values.
- Demonstrate the use of logical operators (&&, ||, !) with these boolean values.

Topic : Non primitive data types

Assignment on Objects

1. **Creating and Accessing Objects:**

    - Create an object named person with properties: firstName, lastName, age, and address (which should itself be an object with properties: street, city, state).
    - Write code to access and print each property of the person object.

Assignment on Array :

1. **Creating and Manipulating Arrays:**

    - Create an array named colors containing the values "red", "green", "blue".
    - Add "yellow" to the end of the array.
    - Remove the first element of the array.
    - Insert "purple" at the beginning of the array.

Topic : **Implicit Type Conversion (Type Coercion)**

1. **Automatic Conversion in Arithmetic Operations:**
    - Write examples where JavaScript implicitly converts data types in arithmetic operations.
    - Explain the result of each operation.
2. **Comparison Operators:**
    - Provide examples of how JavaScript uses implicit conversion when comparing different types using == and ===.

Boolean Coercion in Conditionals:

- Write examples where JavaScript implicitly converts values to boolean in conditional statements (if, while, etc.).
- List values that are considered "falsy" in JavaScript

Topic : Explicit Type Conversion

1. **Converting to String:**

    - Demonstrate how to explicitly convert various data types to a string using String() and .toString().
    - Show the difference between the two methods, if any.

2. **Converting to Number:**
    - Write examples to convert different data types (string, boolean, etc.) to numbers using Number(), parseInt(), and parseFloat().
    - Explain the difference between Number() and the other two methods.
3. **Converting to Boolean:**
    - Show how to explicitly convert different data types to boolean using Boolean() function.
    - Provide examples of truthy and falsy values.

Topic : Scope of variable :

Assignment: Exercises to demonstrate your understanding of how **var, let,** and **const** work in JavaScript. Provide code examples and explanations for each part of the assignment.

Topic : Operator

1. Arithmatic operator :
    - Write code to perform addition, subtraction, multiplication, division, and modulus operations using JavaScript. Explain each operation.
    - Demonstrate the use of the increment (++) and decrement (--) operators, both in prefix and postfix forms. Explain the difference between prefix and postfix.

2. Comparison Operators :
    - Write examples using the equality (==) and inequality (!=) operators. Explain how type coercion affects the comparison.
    - Demonstrate the use of strict equality (===) and strict inequality (!==) operators. Explain why these are generally preferred over == and !=.
    - Provide examples using relational operators (>, <, >=, <=) to compare numbers. Explain how they work.

3. Logical Operators :

AND (&&) and OR (||) Operators:

- Write examples using the logical AND (&&) and OR (||) operators. Explain how they combine boolean expressions.

NOT (`!`) Operator:

- Demonstrate the use of the logical NOT (`!`) operator to invert a boolean value. Provide an example and explanation.

4 Assignment Operators

- Demonstrate how to use the basic assignment operator (`=`) and compound assignment operators (`+=`, `-=`, `*=`, `/=`, `%=`). Provide examples and explanations.

3. Ternary (Conditional) Operator

Using the Ternary Operator:

- Write an example using the ternary operator (`condition ? expr1 : expr2`). Explain how it works as a shorthand for `if-else`.

4. Bitwise Operators (Optional)

- Provide a brief explanation of bitwise operators and their use. Write examples using `&`, `|`, `^`, `~`, `<<`, and `>>`.

Topic : Branching statement :

Write a JavaScript program that prompts the user for their age and outputs a message based on the age entered:

- If the age is less than 13, output "You are a child."
- If the age is between 13 and 19, output "You are a teenager."
- If the age is 20 or more, output "You are an adult."
- Use both `if...else` and `switch` statements to demonstrate branching.

Loop :

- Given an array `['a', 'b', 'c', 'd', 'e']`, use a `for` loop to create a new array with the elements in reverse order and print the new array.

- Write a `do...while` loop that calculates and prints the sum of the first N natural numbers. For example, if N = 5, the sum would be 1 + 2 + 3 + 4 + 5 = 15.
- Write a `do...while` loop to calculate the factorial of a given number (e.g., 5! = 5 × 4 × 3 × 2 × 1 = 120). Ensure that the loop runs at least once.
- Write a `while` loop to reverse a given integer number. For example, if the input is `12345`, the output should be `54321`.
- Given an array of numbers `[3, 5, 7, -2, 4, -1, 8]`, write a `while` loop to find and print the first negative number in the array.

Topic : Function :

- Write a function named `greet` that takes a name as a parameter and prints "Hello, [name]!" to the console.
- Write a function named `average` that takes an array of numbers as an argument and returns their average.

  Example: `average([2, 4, 6, 8, 10])` should return `6`.
- Write a function named `factorial` that takes a non-negative integer and returns its factorial.

Topic : Array methods :

- Write a code and covers all methods like `push`, `pop`, `shift`, `unshift`, `map`, `filter`, `reduce`, `forEach`, `find`, `includes`, `indexOf`, `splice`, `slice`, and `concat`.