**Summary Explanation for JavaScript Topics**

**Classes 15 & 16: Control Flow Statements (if/else, loops)**

**\*\*Topic Introduction:\*\***

**- \*\*Conditional Statements (if/else)\*\*: Teach how to use `if` and `else` statements in JavaScript to make decisions based on conditions. For example, checking if a number is even or odd.**

**```javascript**

**let number = 5;**

**if (number % 2 === 0) {**

**console.log(number + " is even");**

**} else {**

**console.log(number + " is odd");**

**}**

**```**

**5/2 = 2 r 1**

**- \*\*Loops (for, while)\*\*: Explain how loops repeat tasks until a condition is met. Introduce the `for` loop and `while` loop for iterating through arrays or performing repetitive actions.**

**```javascript**

**let array = [1, 2, 3, 4, 5];**

**for (let i = 0; i < array.length; i++) {**

**console.log(array[i]);**

**}**

**```**

**\*\*Classwork:\*\***

**- Practice exercises using `if/else` statements and loops. For example, writing a program to check if numbers are even or odd, and iterating through an array.**

**\*\*Assignment:\*\***

**- Create a JavaScript program using a loop to display numbers 1 to 10 on a webpage using `innerHTML`.**

**Classes 17 & 18: Introduction to DOM Mcript**

**document.getElementById('myElement').innerText = 'New Text';**

**anipulation and Event Handling**

**\*\*Topic Introduction:\*\***

**- \*\*DOM (Document Object Model)\*\*: Explain how JavaScript interacts with webpage elements. Cover basic techniques like accessing elements and modifying content.**

**```javas**

**```**

**- \*\*Event Handling\*\*: Introduce handling user interactions like clicks and mouseovers.**

**```javascript**

**document.getElementById('myButton').addEventListener('click', function() {**

**document.body.style.backgroundColor = 'blue';**

**});**

**```**

**\*\*Classwork:\*\***

**- Practice exercises manipulating DOM elements and handling events. For example, changing text content or dynamically adding/removing elements.**

**\*\*Assignment:\*\***

**- Create a webpage with a button that changes the background color when clicked.**

**Classes 19 & 20: Object-Oriented Programming (OOP) Concepts in JavaScript**

**\*\*Topic Introduction:\*\***

**- \*\*Objects\*\*: Describe how objects represent real-world entities with properties and methods.**

**```javascript**

**let book = {**

**title: 'JavaScript Basics',**

**author: 'John Doe',**

**genre: 'Programming',**

**displayInfo: function() {**

**console.log(this.title + ' by ' + this.author);**

**}**

**};**

**book.displayInfo();**

**```**

**\*\*Classwork:\*\***

**- Practice creating objects and using methods. For example, creating an object representing a product with properties like name and price.**

**\*\*Assignment:\*\***

**- Create a JavaScript program defining an object representing a book with properties like title, author, and genre, and include methods to display this information on a webpage.**

**Classes 21 & 22: Working with Arrays and Objects**

**\*\*Topic Introduction:\*\***

**- \*\*Arrays\*\*: Explain how arrays store collections of data and cover common methods like adding, removing, and iterating through elements.**

**```javascript**

**let fruits = ['apple', 'banana'];**

**fruits.push('orange'); // Adds 'orange' to the end**

**console.log(fruits);**

**```**

**- \*\*Objects\*\*: Discuss accessing and modifying object properties and methods.**

**```javascript**

**let person = {**

**name: 'Alice',**

**age: 25**

**};**

**person.age = 26;**

**console.log(person.age); // 26**

**```**

**\*\*Classwork:\*\***

**- Practice exercises using arrays and objects. For example, performing operations on arrays like sum and average, and updating object properties.**

**By providing explanations and examples at each step, students will develop a solid understanding of JavaScript concepts and be able to apply them effectively.**