**PRACTICAL-03**

**PROJECT STATUS: COMPLETED**

Basics of JavaScript with project status checking

JavaScript is a versatile programming language that is primarily used for web development. It allows you to add interactivity, manipulate the content of a webpage, and respond to user actions. Here are the basics of JavaScript:

1. Variables:

Variables are used to store and manipulate data. You declare a variable using the `var`, `let`, or `const` keyword.

```javascript

var x = 10;

let y = "Hello, ";

const pi = 3.14;

```

- `var` is function-scoped.

- `let` and `const` are block-scoped. Use `let` for variables that can be reassigned, and `const` for constants.

2. Data Types:

JavaScript has several data types, including numbers, strings, booleans, objects, arrays, and more.

```javascript

var num = 42;

var text = "Hello, World!";

var isTrue = true;

var person = { name: "John", age: 25 };

var fruits = ["apple", "banana", "orange"];

```

3. Operators:

JavaScript supports various operators for performing operations on variables and values.

```javascript

var sum = 5 + 3;

var concat = "Hello" + " " + "World";

var isGreaterThan = 10 > 5;

```

4. Functions:

Functions allow you to group and reuse blocks of code. You can define functions and call them later in your code.

```javascript

function greet(name) {

return "Hello, " + name + "!";

}

var greeting = greet("John");

```

5. Control Flow:

JavaScript supports conditional statements (if, else if, else) and loops (for, while, do-while) for controlling the flow of your program.

```javascript

if (condition) {

// code to execute if the condition is true

} else if (anotherCondition) {

// code to execute if the second condition is true

} else {

// code to execute if no conditions are true

}

for (var i = 0; i < 5; i++) {

// code to repeat five times

}

```

6. Events:

JavaScript enables you to respond to user actions, such as clicks, keypresses, or mouse movements, through event handling.

```javascript

document.getElementById("myButton").addEventListener("click", function() {

alert("Button clicked!");

});

```

7. DOM Manipulation:

The Document Object Model (DOM) represents the structure of an HTML document. JavaScript can be used to manipulate the DOM, updating and modifying the content dynamically.

```javascript

document.getElementById("myElement").innerHTML = "New content";

```

8. Asynchronous JavaScript:

JavaScript supports asynchronous programming using features like callbacks, Promises, and async/await for handling tasks that might take some time to complete, such as fetching data from a server.

```javascript

fetch("https://api.example.com/data")

.then(response => response.json())

.then(data => console.log(data))

.catch(error => console.error("Error:", error));

```

These basics provide a foundation for working with JavaScript. As you delve deeper into JavaScript development, you'll explore advanced topics like object-oriented programming, AJAX, modularization, and various libraries/frameworks such as React, Angular, or Vue.js for building interactive web applications.