**JavaScript Functions**

In JavaScript a function allows us to define a block of code designed to perform a specific task. They are a fundamental part of JavaScript programming. The code we write can be executed as many times as we want.

Syntax:

// defining a function

Function <function name>()

{

//code to be executed

};

//calling a function

<function name>();

**Function Parameters:**

A function can have one or more parameters, which will be supplied by the calling code and can be used inside the function.

JavaScript is a dynamic type scripting language, so function parameters can have of any data type.

Function show Message(firstname,lastname)

{

Return (“Hello” + first name+ ““+last name);

}

**Functions in Java Script**

**1.Function Declarations:**  
These are defined using the function keyword and can be called before they are defined due to hoisting.

// Function Declaration

function addNumbers(a, b) {

return a + b;

}

// Calling the function

const result = addNumbers(5, 3);

console.log(result); // Output: 8

**2.Function Expressions**

Functions can be assigned to variables and used as values.

javascript

Copy code

const greet = function(name) {  
 return `Hello, ${name}!`;  
};  
console.log(greet("Bob")); // Hello, Bob!

**3.Arrow Functions**

Introduced in ES6, they are a concise way to write functions and do not have their own this binding.

javascript

Copy code

const greet = (name) => `Hello, ${name}!`;  
console.log(greet("Charlie")); // Hello, Charlie!

**4.Anonymous Functions**

Functions without a name, often used as arguments to other functions.

javascript

Copy code

setTimeout(function() {  
 console.log("This runs after 2 seconds");  
}, 2000);

**5.Immediately Invoked Function Expressions (IIFE)**

* These functions are executed immediately after their definition.

javascript

Copy code

(function() {  
 console.log("This is an IIFE");  
})();

**6.Constructor Functions**

Functions used to create objects.

javascript

Copy code

function Person(name, age) {  
 this.name = name;  
 this.age = age;  
}  
const alice = new Person("Alice", 25);  
console.log(alice.name); // Alice

**7.Generator Functions**

Functions that can pause execution and yield values using the function\* syntax.

javascript

Copy code

function\* generateNumbers() {  
 yield 1;  
 yield 2;  
 yield 3;  
}  
const gen = generateNumbers();  
console.log(gen.next().value); // 1

**8.Async Functions**

Functions that work with asynchronous operations using async and await.

JavaScript

Copy code

async function fetchData() {  
 const response = await fetch('https://api.example.com/data');  
 const data = await response.Json();  
 return data;  
}  
fetchData().then(console.log);