**Functions in JavaScript**

Functions are one of the most fundamental building blocks. A function is a block of code designed to perform a specific task and can be executed when called. Functions are powerful tools in JavaScript, enabling abstraction, modularity, and reusability in our code.

**Types of Functions in JavaScript.**

1. **Function Declaration**

A named function that can be called anywhere in the script (due to hoisting).

Example

function greet(name) {

return `Hello, ${name}!`;

}

console.log(greet("Alice"));

1. **Function Expression**

A function assigned to a variable, not hoisted.

Example

const greet = function(name) {

return `Hello, ${name}!`;

};

console.log(greet("Bob"));

1. **Arrow** **Function**

A concise syntax for writing functions (introduced in ES6).

Example

const greet = (name) => `Hello, ${name}!`;

console.log(greet("Charlie"));

1. **Anonymous Functions**

Functions without a name, often used as callbacks.

Example

setTimeout(function() {

console.log("This runs after 1 second");

}, 1000);

1. **Immediately Invoked Function Expression (IIFE)**

A function that runs as soon as it is defined.

Example

(function() {

console.log("IIFE executed!");

})();

1. **Generator Function**

Functions that can be paused and resumed, marked by an asterisk \*.

Example

function\* generateNumbers() {

yield 1;

yield 2;

yield 3;

}

const gen = generateNumbers();

console.log(gen.next().value); // 1

console.log(gen.next().value); // 2

1. **Async Function**

Functions that return a Promise and allow asynchronous code with await.

Example

async function fetchData() {

const response = await fetch('https://api.example.com/data');

const data = await response.json();

return data;

}