

Functions in JavaScript are blocks of reusable code that can be defined and invoked to perform a specific task. There are several ways to create functions in JavaScript. I'll provide examples for the most common methods:

1. Function Declaration:

A function declaration is a statement that defines a function.

```
```javascript
// Function Declaration
function greet(name) {
 console.log(`Hello, ${name}!`);
}

// Function Invocation
greet("John"); // Output: Hello, John!
```
```

2. Function Expression:

A function expression defines a function as part of an expression. It can be named or anonymous.

```
```javascript
// Named Function Expression
const greet = function sayHello(name) {
 console.log(`Hello, ${name}!`);
};

// Function Invocation
greet("Jane"); // Output: Hello, Jane!
```
```

3. Arrow Function:

Arrow functions provide a concise syntax for writing functions. They are often used for short, one-liner functions.

```
```javascript
// Arrow Function
const greet = (name) => {
 console.log(`Hello, ${name}!`);
};
```

```
// Function Invocation
greet("Alice"); // Output: Hello, Alice!
````
```

4. Function with Return:

Functions can return values using the `return` keyword.

```
````javascript
// Function with Return
function add(a, b) {
 return a + b;
}

// Function Invocation
const result = add(3, 4);
console.log(result); // Output: 7
````
```

5. Function with Default Parameters:

You can specify default values for function parameters.

```
````javascript
// Function with Default Parameters
function greet(name = "Guest") {
 console.log(`Hello, ${name}!`);
}

// Function Invocation
greet(); // Output: Hello, Guest!
greet("Bob"); // Output: Hello, Bob!
````
```

6. Function with Rest Parameter:

The rest parameter allows a function to accept an indefinite number of arguments as an array.

```
````javascript
// Function with Rest Parameter
function sum(...numbers) {
 return numbers.reduce((total, num) => total + num, 0);
}
```

```
// Function Invocation
const total = sum(1, 2, 3, 4, 5);
console.log(total); // Output: 15
...
```

These are basic examples to illustrate the different ways you can define and use functions in JavaScript. Functions are a fundamental building block in JavaScript, enabling code organization, reusability, and modularity.