

FUNCTIONS IN JAVASCRIPT

Sai Vardhan T

FUNCTIONS IN JS

- Functions in JavaScript are reusable blocks of code that perform a specific task.
- They allow you to encapsulate code, making it easier to manage, reuse, and maintain.
- Here are some clear types of functions in JavaScript:
 - Function declaration
 - Function Expression
 - Arrow Function
 - Immediately invoked Function Expression (IIFE)
 - Named Function Expression

FUNCTION DECLARATION

- A Function declaration defines a named function
- SYNTAX:

```
function functionName(parameters) {  
    // code block  
}
```

- EXAMPLE:

```
// Function to add two numbers  
function add(a, b) {  
    return a + b;  
}  
  
// Call the function  
var result = add(5, 3);  
console.log(result); // Output: 8
```

EXAMPLE

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Function Declaration Example</title>
</head>
<body>
  <h1>Function Declaration Example</h1>
  <p>Click the button to display a greeting message.</p>
  <button onclick="displayGreeting()">Display Greeting</button>
  <script>
    // Function declaration
    function displayGreeting() {
      alert("Hello, welcome to our website!");
    }
  </script>
</body>
</html>
```

FUNCTION EXPRESSION

- A Function expression defines a function as part of a larger expression
- SYNTAX:

```
var functionName = function(parameters) {  
    // code block  
};
```

- EXAMPLE:

```
// Function expression to multiply two numbers  
var multiply = function(a, b) {  
    return a * b;  
};  
  
// Call the function  
var result = multiply(5, 3);  
console.log(result); // Output: 15
```

EXAMPLE

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Function Expression Example</title>
</head>
<body>
  <h1>Function Expression Example</h1>
  <p>Click the button to display a greeting message.</p>
  <button onclick="displayGreeting()">Display Greeting</button>

  <script>
    var displayGreeting = function() {
      alert("Hello, welcome to our website!");
    };
  </script>
</body>
</html>
```

ARROW FUNCTION

- An arrow function is a concise way to write anonymous functions
- SYNTAX:

```
var functionName = (parameters) => {  
    // code block  
};
```

- EXAMPLE:

```
// Arrow function to calculate the square of a number  
var square = (x) => {  
    return x * x;  
};  
  
// Call the function  
var result = square(5);  
console.log(result); // Output: 25
```

EXAMPLE

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Arrow Function Example</title>
</head>
<body>
  <h1>Arrow Function Example</h1>
  <p>Click the button to display a greeting message.</p>
  <button onclick="displayGreeting()">Display Greeting</button>
  <script>
    // Arrow function
    var displayGreeting = () => {
      alert("Hello, welcome to our website!");
    };
  </script>
</body>
</html>
```


Immediately Invoked Function Expression (IIFE)

- An IIFE is a function that is executed immediately after it's defined.
- SYNTAX:
- EXAMPLE:

```
(function() {  
    // code block  
})();
```

```
// IIFE to display a message  
(function() {  
    console.log("Hello from IIFE!");  
})();  
// Output: Hello from IIFE!
```

EXAMPLE

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>IIFE Example</title>
</head>
<body>
  <h1>IIFE Example</h1>
  <p>Open the console to see the output.</p>

  <script>
    // Immediately Invoked Function Expression (IIFE)
    (function() {
      var message = "This is an IIFE example.";
      alert(message);
    })();
  </script>
</body>
</html>
```

Named Function Expression

- A named function expression allows a function expression to have a name.
- SYNTAX:

```
var functionName = function functionName(parameters) {  
    // code block  
};
```

- EXAMPLE:

```
// Named function expression to calculate the factorial of a number  
var factorial = function factorial(n) {  
    if (n <= 1) {  
        return 1;  
    }  
    return n * factorial(n - 1);  
};  
  
// Call the function  
var result = factorial(5);  
console.log(result); // Output: 120
```

EXAMPLE

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Named Function Expression Example</title>
</head>
<body>
  <h1>Named Function Expression Example</h1>
  <p>Open the console to see the output.</p>

  <script>
    // Named Function Expression
    var displayMessage = function showMessage() {
      alert("This is a Named Function Expression example.");
    };
    displayMessage();
  </script>
</body>
</html>
```

TYPES OF USER DEFINED FUNCTIONS

- Here is the break down of functions:
 - With Parameters With Return type
 - With Parameters Without Return type
 - Without Parameters with Return type
 - Without Parameters without Return type

With Parameters With Return type

- This type of function has parameters and return type.

```
<!doctype html>
<html>
  <head>
    <script>
      function add(a, b){
        var c = a+b;
        return "Sum = "+c;
      }
      function calculate(){
        document.getElementById("sum").innerHTML = add(10, 20);
      }
    </script>
  </head>
  <body>
    <button onclick="calculate()">Add</button>
    <p id="sum"> </p>
  </body>
</html>
```

With Parameters Without Return type

- This type of function takes input, performs operations, and display results from the function itself.

```
<!doctype html>
<html>
  <head>
    <script>
      function add(a, b){
        var c = a+b;
        document.write("Sum is : " + c);
      }
    </script>
  </head>

  <body>
    <button onclick="add(10, 20)">Add</button>
  </body>
</html>
```

Without Parameters with Return type

- This function will not have parameters but have a return type.

```
<!doctype html>
<html>
  <head>
    <script>
      function PI(){
        return "PI value is = 3.142";
      }
      function getPI(){
        document.getElementById("pi").innerHTML = PI();
      }
    </script>
  </head>
  <body>
    <button onclick="getPI()">PI Value</button>
    <p id="pi"> </p>
  </body>
</html>
```


Without Parameters without Return type

- This type of function is used to perform operations like clear(), removeAll(), displayAll() etc.

```
<!doctype html>
<html>
  <head>
    <script>
      function sayHello(){
        document.write("Hello All, Welcome");
      }
    </script>
  </head>
  <body>
    <button onclick="sayHello()">Say Hello to All</button>
  </body>
</html>
```