Types of Functions in JavaScript

@_codevalley

Zuhaib Asif

Types of Functions:

- Named Functions
- Anonymous Functions
- Arrow Functions
- IIFE (Immediately Invoked Function Expression)
- Higher-Order Functions
- Constructor Functions

Named Functions:

These are regular functions that are defined with a name and can be called by that name.

```
function greet(name) {
  console.log(`Hello, ${name}!`);
}
greet("CodeValley"); // Output: Hello, CodeValley!"));
```

Anonymous Functions:

These are functions that do not have a name and are often used as arguments to other functions or assigned to variables.

```
const sum = function(a, b) {
  return a + b;
}
console.log(sum(3, 5)); // Output: 8
```

Arrow Function:

Introduced in ECMAScript 6 (ES6), arrow functions provide a concise syntax for writing functions, especially when you want to use a simple one-liner.

```
const multiply = (x, y) => x * y;
console.log(multiply(4, 6)); // Output: 24
```

IIFE (Immediately Invoked Function Expression):

These are self-invoking functions that execute as soon as they are defined.

```
(function() {
console.log("I am invoked immediately!");
})(); // Output: I am invoked immediately!
```

Higher-Order Functions:

Functions that can accept other functions as arguments and/or return functions.
eg: map(), filter(), reduce()

```
let arr = [5,10,15,20,25];
let newarr = arr.map((element)=>{
  return element + 5;
})
console.log(newarr);  // Output: [10,15,20,25,30]
```

Constructor Function:

Functions used to create instances of objects, often used for defining classes and creating objects with shared properties and methods.

```
function Person(name, age) {
  this.name = name;
  this.age = age;
}
const person = new Person("xyz", 30);
console.log(person.name); // Output: xyz
```