

Javascript

Function, Object, Array
Methods



Javascript Function, Object, Array Methods

- ☐ Function
- ☐ Object
- ☐ Array Methods

Objektif sesi


- Memahami penggunaan Function pada Javascript
- Memahami penggunaan Object pada Javascript
- Memahami penggunaan Array pada Javascript

Function

Fungsi (function) dalam JavaScript adalah blok kode yang dirancang untuk melakukan tugas tertentu, dan fungsi tersebut dapat dipanggil (dijalankan) secara berulang. Fungsi membantu mengorganisir dan merapikan kode, serta mengizinkan kita untuk memecah program menjadi bagian-bagian yang lebih kecil dan mudah dikelola.

Fungsi Tanpa Parameter

javascript

 Copy code

```
// Fungsi tanpa parameter
function sapa() {
  console.log("Halo, selamat datang!");
}


// Memanggil fungsi
sapa();
```

Alur:

- Mendeklarasikan fungsi sapa tanpa parameter.
- Memanggil fungsi sapa, yang mencetak pesan sambutan ke konsol.

Fungsi Dengan Parameter

javascript

 Copy code

```
// Fungsi dengan parameter
function tambah(a, b) {
    return a + b;
}

// Memanggil fungsi
var hasil = tambah(5, 3);
console.log("Hasil penambahan:", hasil);
```

Alur:

- Mendeklarasikan fungsi tambah dengan dua parameter, a dan b.
- Fungsi mengembalikan hasil penambahan a dan b.
- Memanggil fungsi tambah dengan argumen 5 dan 3, dan menyimpan hasilnya dalam variabel hasil.
- Mencetak hasil penambahan ke konsol.

Fungsi Dengan Nilai Default

javascript

Copy code

```
// Fungsi dengan nilai default
function sapaNama(nama = "Teman") {
  console.log("Halo, " + nama + "!");
}

// Memanggil fungsi tanpa menyediakan argumen
sapaNama();

// Memanggil fungsi dengan menyediakan argumen
sapaNama("John");
```

Alur:


- Mendeklarasikan fungsi sapaNama dengan parameter nama yang memiliki nilai default "Teman".
- Memanggil fungsi sapaNama tanpa menyediakan argumen, sehingga nilai default "Teman" digunakan.
- Memanggil fungsi sapaNama dengan menyediakan argumen "John", sehingga nilai argumen digunakan.

Object

Objek (object) adalah tipe data di JavaScript yang mewakili sekumpulan nilai atau properti, serta fungsi atau metode yang dapat bekerja dengan nilai-nilai tersebut. Objek memungkinkan kita untuk mengelompokkan data dan perilaku terkait ke dalam satu entitas.

Objek Sederhana

javascript

 Copy code

```
// Mendefinisikan objek 'mobil'
var mobil = {
  merk: "Toyota",
  model: "Camry",
  tahun: 2022,
  warna: "Putih",
  hidupkanMesin: function() {
    console.log("Mesin dinyalakan!");
  }
};

// Mengakses properti dan memanggil metode objek
console.log("Merk Mobil:", mobil.merk);
console.log("Tahun Produksi:", mobil.tahun);
mobil.hidupkanMesin();
```

Alur:

- Membuat objek mobil dengan properti seperti merk, model, tahun, dan warna.
- Menambahkan metode hidupkanMesin ke dalam objek mobil.
- Mengakses properti objek menggunakan dot notation (mobil.merk).
- Memanggil metode objek (mobil.hidupkanMesin()).

Objek Dengan Metode dan this

javascript

Copy code

```
// Mendefinisikan objek 'orang'
var orang = {
  nama: "John",
  usia: 30,
  sapa: function() {
    console.log("Halo, nama saya " + this.nama + " dan saya berusia " +
  }
};


// Memanggil metode objek
orang.sapa();
```

Alur:

- Membuat objek orang dengan properti nama, usia, dan metode sapa.
- Menggunakan this untuk merujuk ke properti dalam metode objek.
- Memanggil metode objek (orang.sapa()).

Objek sebagai Fungsi Konstruktor

javascript

 Copy code

```
// Mendefinisikan fungsi konstruktor 'Mahasiswa'
function Mahasiswa(nama, jurusan) {
  this.nama = nama;
  this.jurusan = jurusan;
  this.sapa = function() {
    console.log("Halo, nama saya " + this.nama + " dari jurusan " + this.jurusan);
  };
}

// Membuat objek menggunakan fungsi konstruktor
var mahasiswa1 = new Mahasiswa("Alice", "Informatika");
var mahasiswa2 = new Mahasiswa("Bob", "Ekonomi");

// Memanggil metode objek
mahasiswa1.sapa();
mahasiswa2.sapa();
```

Alur:


- Mendefinisikan fungsi konstruktor Mahasiswa untuk membuat objek mahasiswa.
- Fungsi konstruktor ini memiliki parameter nama dan jurusan.
- Membuat objek mahasiswa1 dan mahasiswa2 menggunakan kata kunci new.
- Memanggil metode objek (mahasiswa1.sapa() dan mahasiswa2.sapa()).

Array Methods

Array methods pada JavaScript adalah fungsi bawaan yang dapat diterapkan pada objek array untuk melakukan berbagai operasi, seperti memanipulasi elemen, menyaring, mencari, atau mengubah isi array.

push() dan pop()

javascript

 Copy code

```
// Mendefinisikan array 'buah'
var buah = ["apel", "pisang", "jeruk"];

// Menambahkan elemen baru ke array menggunakan push
buah.push("mangga");

// Menghapus elemen terakhir dari array menggunakan pop
buah.pop();


console.log(buah); // Output: ["apel", "pisang", "jeruk"]
```

Alur:

- Menambahkan elemen baru "mangga" ke array menggunakan push().
- Menghapus elemen terakhir dari array menggunakan pop().

shift() dan unshift()

javascript

 Copy code

```
// Mendefinisikan array 'mobil'
var mobil = ["Toyota", "Honda", "Nissan"];

// Menghapus elemen pertama dari array menggunakan shift
mobil.shift();

// Menambahkan elemen baru di awal array menggunakan unshift
mobil.unshift("Ford");

console.log(mobil); // Output: ["Ford", "Honda", "Nissan"]
```

Alur:

- Menghapus elemen pertama "Toyota" dari array menggunakan shift().
- Menambahkan elemen baru "Ford" di awal array menggunakan unshift().

map()

javascript

Copy code

```
// Mendefinisikan array 'angka'
var angka = [1, 2, 3, 4, 5];

// Menggunakan map untuk menggandakan setiap elemen dalam array
var hasilKaliDua = angka.map(function(item) {
    return item * 2;
});


console.log(hasilKaliDua); // Output: [2, 4, 6, 8, 10]
```

Alur:

- Menggunakan map() untuk membuat array baru (hasilKaliDua) dengan nilai setiap elemen dari array angka dikalikan 2.

filter()

javascript

 Copy code

```
// Mendefinisikan array 'nilai'
var nilai = [80, 45, 90, 60, 75];

// Menggunakan filter untuk menyaring nilai di atas 70
var nilaiLulus = nilai.filter(function(item) {
  return item > 70;
});

console.log(nilaiLulus); // Output: [80, 90, 75]
```

Alur:

- Menggunakan filter() untuk membuat array baru (nilaiLulus) yang berisi nilai di atas 70 dari array nilai.

reduce()

javascript

Copy

```
// Mendefinisikan array 'angka'
var angka = [1, 2, 3, 4, 5];

// Menggunakan reduce untuk menjumlahkan semua elemen dalam array
var jumlah = angka.reduce(function(acc, current) {
    return acc + current;
}, 0);

console.log(jumlah); // Output: 15
```

Alur:

Menggunakan reduce() untuk menjumlahkan semua elemen dalam array angka.



Thank you