



SMK NEGERI  
4 MALANG

# **MODUL** **PRAKTIKUM** **JAVASCRIPT** KELAS X

---

Penyusun:  
Ria Febrianti  
Dwi Fahmi Muhammad



**JS**

## Daftar Isi:

Daftar Isi:.....	i
Capaian Pembelajaran .....	ii
MATERI 5.....	1
A. <i>Function</i> .....	1
B. Objek Input (Elemen Formulir) dalam HTML:.....	2
C. Object Literal :.....	8
D. Constructor : .....	9

## **Capaian Pembelajaran**

Pada akhir fase E peserta didik mampu melakukan pemrograman terstruktur, antara lain penerapan struktur data yang terdiri dari data statis (array baik dimensi, panjang, tipe data, pengurutan) dan data dinamis (list, stack), penggunaan tipe data, struktur kontrol perulangan dan percabangan pada proyek pengembangan perangkat lunak sederhana dan gim.

## **Tujuan Pembelajaran**

1. Peserta didik mampu menjelaskan konsep bahasa pemrograman JavaScript melalui tanya jawab dengan benar.
2. Peserta didik mampu menyimpulkan tentang objek function dan objek input melalui tanya jawab dengan benar.
3. Peserta didik mampu menjalankan program dengan bahasa pemrograman JavaScript melalui latihan praktikum dengan benar.
4. Peserta didik mampu memecahkan studi kasus menggunakan bahasa pemrograman JavaScript setelah melakukan latihan praktikum dengan benar..

## **Petunjuk Umum**

1. Awali setiap aktivitas dengan doa semoga mendapat kemudahan dalam mempelajari modul dan keberkahan dalam mempelajari ilmu animasi digital puppeteer.
2. Pahami tujuan, petunjuk umum, dan langkah-langkah pembuatan dengan baik dan benar.
3. Perhatikan dengan baik langkah-langkah yang terdapat dalam modul.
4. Ikuti petunjuk dengan benar tanpa terlewati maupun dilewati.
5. Kerjakan tugas eksplorasi dengan baik, sabar, dan sesuai dengan petunjuk.
6. Tanyakan kepada guru apabila terdapat kesulitan mengenai latihan maupun hal-hal yang kurang jelas.

## MATERI 5

### Objek

Objek adalah struktur data kompleks yang dapat menyimpan data dan metode (fungsi) yang bekerja pada data tersebut. Dalam JavaScript, hampir semua hal adalah objek atau dapat dianggap sebagai objek.

#### A. Function

*Function* dalam JavaScript dianggap sebagai objek karena dapat dideklarasikan, disimpan dalam variabel dan digunakan sebagai argumen atau nilai kembalian fungsi lain. Objek fungsi dapat didefinisikan dengan tiga cara utama: **Function Declaration**, **Function Expression**, dan **Arrow Function**. **Function Declaration** menggunakan sintaks function diikuti dengan nama fungsi dan parameter, **Function Expression** menyimpan fungsi dalam variabel, dan **Arrow Function** menyediakan sintaksis yang lebih ringkas.

*Function* dalam JavaScript digunakan untuk mengelompokkan kode menjadi blok yang dapat dipanggil dan dieksekusi secara berulang. Ada beberapa cara untuk mendefinisikan fungsi dalam JavaScript, di antaranya:

##### 1. Function Declaration

Mendefinisikan fungsi dengan menggunakan kata kunci function diikuti oleh nama fungsi dan blok kode.

```
function name (parameter) {  
    statements;  
}
```

Contoh:

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

##### 2. Function Expression

Mendefinisikan fungsi sebagai ekspresi yang disimpan dalam variabel. Fungsi ini dapat disimpan dalam variabel dan digunakan seperti nilai lainnya.

```
const name = function (parameter) {  
    statements;  
}
```

Contoh:

```
let greet = function(name) {  
  console.log("Hello, " + name + "!");  
};  
  
greet("John"); // Output: Hello, John!
```

### 3. Arrow Function

Cara baru dan lebih ringkas untuk menulis fungsi dalam JavaScript, diperkenalkan dalam ECMAScript 6. Menggunakan tanda panah => untuk menunjukkan fungsi, yang menghilangkan kebutuhan akan kata kunci function dan dapat memiliki sintaks yang lebih singkat.

```
(parameter)=> {  
  statements;  
}
```

Contoh:

```
let greet = (name) => {  
  console.log("Hello, " + name + "!");  
};  
  
greet("John"); // Output: Hello, John!
```

## B. Objek Input (Elemen Formulir) dalam HTML:

Elemen formulir/input dalam HTML adalah bagian dari antarmuka pengguna yang memungkinkan pengguna memberikan masukan atau memilih opsi. Beberapa elemen formulir/input umum meliputi **Text Input** untuk memasukkan teks, **Radio Input** untuk memilih satu opsi dari beberapa opsi, **Checkbox Input** untuk memilih atau menghapus opsi, **Textarea** Input untuk memasukkan teks multiline, dan **Select** Input untuk menyediakan pilihan dropdown. Setiap elemen formulir/input memiliki atribut dan properti yang dapat dimanipulasi menggunakan JavaScript untuk mengakses atau mengubah nilainya.

Objek input dalam JavaScript digunakan untuk mengambil dan memanipulasi nilai yang dimasukkan oleh pengguna melalui elemen-elemen input dalam formulir HTML. Beberapa jenis elemen input yang umum digunakan adalah:

## 1. Latihan 1

Text Input: digunakan untuk memasukkan teks atau string oleh pengguna.

Contoh:

```
<html>
<head>
  <title>Form Pengiriman</title>
  <script language="JavaScript">
    function kirimData() {
      var nama = document.formPengiriman.nama.value;
      var alamat =
document.formPengiriman.alamat.value;
      var kota = document.formPengiriman.kota.value;
      var kodePos =
document.formPengiriman.kode_pos.value;

      // Menetapkan nilai input ke elemen output
      document.getElementById("outputNama").innerText =
"Nama: " + nama;
      document.getElementById("outputAlamat").innerText
= "Alamat: " + alamat;
      document.getElementById("outputKota").innerText =
"Kota: " + kota;

document.getElementById("outputKodePos").innerText = "Kode
Pos: " + kodePos;
    }
  </script>
</head>

<body>

  <form name="formPengiriman">
    <h2>Detail Pengiriman</h2>
    <pre>
Nama          : <input type="text" size="25" name="nama">
Alamat        : <input type="text" size="25" name="alamat">
Kota          : <input type="text" size="25" name="kota">
Kode Pos      : <input type="text" size="25"
name="kode_pos">
</pre>
    <input type="button" value="Kirim"
onclick="kirimData()" />
    <input type="reset" value="Ulang" />
  </form>

  <div id="output">
    <h2>Hasil Pengiriman</h2>
    <p id="outputNama"></p>
    <p id="outputAlamat"></p>
    <p id="outputKota"></p>
    <p id="outputKodePos"></p>
  </div>
```

```
</body>
</html>
```

Output:

### Detail Pengiriman

Nama :   
Alamat :   
Kota :   
Kode Pos :

### Hasil Pengiriman

## 2. Latihan 2

Radio Input: digunakan untuk memilih satu opsi dari beberapa opsi yang disediakan.

```
<!DOCTYPE html>
<html lang="en">

<head>
  <title>Pilih Metode Pembayaran</title>
</head>

<body>
  <script lang="javascript">
    const pilihMetodePembayaran = function (form) {
      var metode = "";
      if (form.metode_pembayaran.value == "transfer") {
        metode = "Transfer Bank";
      } else if (form.metode_pembayaran.value ==
"kartu_kredit") {
        metode = "Kartu Kredit";
      } else if (form.metode_pembayaran.value ==
"e_wallet") {
        metode = "E-Wallet";
      } else {
        metode = "Belum dipilih";
      }
      alert('Anda telah memilih metode pembayaran: ' +
metode);
    }
  </script>
  <form>
    <h1>Pilih Metode Pembayaran:</h1>
    <hr />
    <p><input type="radio" value="transfer"
name="metode_pembayaran" />Transfer Bank</p>
    <p><input type="radio" value="kartu_kredit"
```

```

name="metode_pembayaran" />Kartu Kredit</p>
    <p><input type="radio" value="e_wallet"
name="metode_pembayaran" />E-Wallet</p>
    <hr />
    <p>
        <input type="button" value="CONFIRM"
onclick="pilihMetodePembayaran(this.form)" />
        <input type="reset" value="RESET" />
    </p>
</form>
</body>

</html>

```

Output:

**Pilih Metode Pembayaran:**

☐ Transfer Bank  
☐ Kartu Kredit  
☐ E-Wallet

### 3. Latihan 3

Checkbox Input: digunakan untuk memilih satu atau beberapa opsi dari beberapa opsi yang disediakan.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Pilih Produk</title>
</head>
<body>
    <form>
        <h1>Pilih Produk yang Ingin Dibeli:</h1>
        <hr>
        <label><input type="checkbox" value="ON" name="baju">
Baju</label><br>
        <label><input type="checkbox" value="ON"
name="celana"> Celana</label><br>
        <label><input type="checkbox" value="ON"
name="sepatu"> Sepatu</label><br>
        <hr>
        <input type="button" value="CONFIRM"
onclick="pilihProduk(this.form)">

```



```

        <input type="reset" value="RESET">
    </form>

    <script>
        let pilihProduk = (form) => {
            var produk = [];
            if (form.baju.checked == true) {
                produk.push("Baju");
            }
            if (form.celana.checked == true) {
                produk.push("Celana");
            }
            if (form.sepatu.checked == true) {
                produk.push("Sepatu");
            }
            if (produk.length === 0) {
                alert('Silakan pilih setidaknya satu
produk!');
            } else {
                alert('Anda telah memilih produk: ' +
produk.join(', ');
            }
        }
    </script>
</body>
</html>

```

Output:

### Pilih Produk yang Ingin Dibeli:

- ☐ Baju  
☐ Celana  
☐ Sepatu

CONFIRM

RESET

#### 4. Latihan 4

Textarea: digunakan untuk memasukkan teks atau string dalam jumlah yang lebih besar daripada yang dapat diterima oleh elemen input teks biasa.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Alamat Pengiriman</title>
</head>

```

```

<body>
  <script lang="javascript">
    function kirim() {
      var alamatStr = document.form.alamat.value;
      document.getElementById("outputAlamat").innerText
= alamatStr;
    }
  </script>
  <form name="form">
    <h1>Form Alamat Pengiriman</h1>
    <hr />
    <h3>Masukkan Alamat Pengiriman:</h3>
    <textarea name="alamat" rows="3"
cols="30"></textarea>
    <br /><br />
    <input type="button" value="KIRIM" onclick="kirim()"
/>

    <input type="reset" value="ULANG" />
    <h3>Alamat Pengiriman yang Akan Dikirim:</h3>
    <p id="outputAlamat"></p>
  </form>
</body>
</html>

```

Output:

## Form Alamat Pengiriman

---

**Masukkan Alamat Pengiriman:**

**Alamat Pengiriman yang Akan Dikirim:**

## 5. Latihan 5

Select: digunakan untuk memilih satu opsi dari daftar opsi yang disediakan dalam bentuk dropdown.

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Pilih Produk</title>
</head>
<body>

```

```

<script lang="javascript">
    function pilihProduk(form) {
        var produk = form.produk.value;
        document.getElementById("outputProduk").innerText
= "Anda memilih produk: " + produk;
    }
</script>
<form name="form">
    <h1>Pilih Produk yang Ingin Dibeli:</h1>
    <h3>Silakan pilih produk yang Anda inginkan:</h3>
    <select name="produk" size="1">
        <option value="Baju">Baju</option>
        <option value="Celana">Celana</option>
        <option value="Sepatu">Sepatu</option>
        <option value="Tas">Tas</option>
    </select>
    <br /><br />
    <input type="button" value="KIRIM"
onclick="pilihProduk(this.form)" />
    <input type="reset" value="ULANG" />
    <h3>Produk yang Anda Pilih:</h3>
    <p id="outputProduk"></p>
</form>
</body>
</html>

```

Output:

## Pilih Produk yang Ingin Dibeli:

Silakan pilih produk yang Anda inginkan:

Baju ▾

KIRIM ULANG

Produk yang Anda Pilih:

### C. Object Literal :

Object literal adalah cara untuk membuat objek langsung di dalam JavaScript tanpa menggunakan konstruktor.

Objek literal didefinisikan dengan menggunakan sepasang kurung kurawal `{}`. Properti dan nilai dari objek dipisahkan oleh titik dua `:` dan dipisahkan satu sama lain oleh koma `,`.

Contoh :

```
let Siswa = {
  nama: "Amir",
  umur: 16,
  kelas: "X RPL B",
  belajar: function() {
    console.log(this.nama + " sedang belajar.");
  }
};
```

Keuntungan :

- Sederhana dan mudah dipahami.
- Cocok untuk membuat objek tunggal atau objek sederhana.

#### D. Constructor :

Definisi :

Constructor adalah fungsi yang digunakan untuk membuat banyak objek dengan properti dan metode yang sama.

Constructor ditulis sebagai fungsi biasa, dengan awalan huruf besar (konvensi kecil). Dalam constructor, properti objek ditetapkan menggunakan `this`.

Note : Untuk membuat objek baru dari constructor, kita menggunakan operator `new`.

Contoh :

```
function Siswa(nama, umur, kelas) {
  this.nama = nama;
  this.umur = umur;
  this.kelas = kelas;
  this.belajar = function() {
    console.log(this.nama + " sedang belajar.");
  };
}
let andi = new Siswa("Amir", 16, "X RPL B");
```

Keuntungan :

- Memungkinkan pembuatan banyak objek dengan struktur yang sama.
- Cocok untuk kasus di mana Anda perlu membuat banyak instance objek dengan properti dan metode yang sama.
- Memungkinkan untuk menerapkan konsep warisan (inheritance) menggunakan prototipe.

### Perbedaan Utama:

1. Objek Literal vs. Constructor :
  - Objek literal digunakan untuk membuat objek tunggal atau objek sederhana.
  - Constructor digunakan untuk membuat banyak objek dengan struktur yang sama.
2. Penggunaan "this" :
  - Pada objek literal, tidak ada penggunaan `this` karena properti dan metode langsung ditentukan di dalam objek.
  - Pada constructor, `this` digunakan untuk merujuk ke objek yang sedang dibuat oleh constructor.
3. Pemanggilan :
  - Objek literal langsung didefinisikan.
  - Constructor membutuhkan operator `new` untuk membuat objek baru.

Pemilihan antara objek literal dan constructor tergantung pada kebutuhan spesifik Anda. Jika Anda hanya perlu membuat objek tunggal atau objek sederhana, objek literal cukup. Namun, jika Anda perlu membuat banyak instance objek dengan struktur yang sama, maka constructor lebih sesuai.

## **Latihan Praktikum 5**

Tuliskan program berikut dan jalankan!

```
// Objek literal untuk biodata siswa SMK
let student = {
  name: "Amir",
  age: 17,
  grade: 10,
  hobbies: ["membaca", "bermain musik"],
  displayInfo: function() {
    console.log("Nama: " + student.name);
    console.log("Usia: " + student.age);
    console.log("Kelas: " + student.grade);
    console.log("Hobi: " + student.hobbies.join(", "));
  }
};

// Menampilkan informasi siswa
student.displayInfo();
```

Dari program tersebut ubah menjadi objek menggunakan constructor dan tambahkan jumlah siswa yang di tampilkan sebanyak 5 teman kalian sesuai absen !