

LAPORAN PRAKTIKUM
PERANCANGAN DAN PEMROGRAMAN WEB

MODUL VI
(JAVASCRIPT & JQUERY)



Oleh:

(Faishal Arif Setiawan)

(2311104066)

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
DIREKTORAT KAMPUS PURWOKERTO
UNIVERSITAS TELKOM
2025

BAB I

PENDAHULUAN

A. Dasar Teori

Javascript, seperti namanya, merupakan bahasa pemrograman *scripting*. Dan seperti bahasa *scripting* lainnya, Javascript umumnya digunakan hanya untuk program yang tidak terlalu besar, biasanya hanya beberapa ratus baris. Javascript pada umumnya mengontrol program yang berbasis Java. Jadi memang pada dasarnya Javascript tidak dirancang untuk digunakan dalam aplikasi skala besar.

Meskipun dibuat dengan tujuan awal untuk mengendalikan program Java, komunitas Javascript menggunakan bahasa ini untuk tujuan lain, memanipulasi gambar dan isi dari dokumen HTML. Singkatnya, pada akhirnya Javascript digunakan untuk satu tujuan utama, “menghidupkan” dokumen HTML dengan mengubah konten statis menjadi dinamis dan interaktif. Bersamaan dengan perkembangan Internet dan dunia *web* yang pesat, Javascript akhirnya menjadi bahasa utama dan satu-satunya untuk membuat HTML menjadi interaktif di dalam browser.

B. Tujuan


1. Mahasiswa mampu memahami konsep dan implementasi Javascript pada *web*.
2. Mahasiswa mampu memahami sintaks, elemen, dan fungsi pada Javascript.
3. Mahasiswa mampu memahami konsep dan implementasi jQuery pada *web*.
4. Mahasiswa mampu memahami sintaks jQuery.

BAB II

HASIL

A. GUIDED (Praktikum Terbimbing)

- a. Tipe Data pada JavaScript:
membuat beberapa variabel dengan tipe data yang berbeda-beda seperti angka (number), teks (string), logika benar/salah (boolean), kosong (null), belum diisi (undefined), daftar (array), tanggal (date), pola huruf besar (regex), dan juga objek. Semua hasilnya kemudian ditampilkan ke halaman menggunakan innerHTML.



```
1  var angka = 42;
2  var teks = "Ahmad Ruslan";
3  var benar = true;
4  var salah = false;
5  var kosong = null;
6  var tidakDidefinisikan;
7  var daftar = ["apel", "pisang", "mangga"];
8  var tanggal = new Date();
9  var pola = /[A-Z]/;
10 var orang = { nama: "Ruslan", umur: 21 };
11
```

Penjelasan:

Kode ini menunjukkan kalau JavaScript itu fleksibel kita bisa menaruh apa saja di dalam variabel dan langsung menampilkannya di browser.

Output:

```
Number: 42
String: Ahmad Ruslan
Boolean 1: true
Boolean 0: false
Null: null
Undefined: undefined
Array: apel, pisang, mangga
Date: Thu Oct 30 2025 04:08:17 GMT+0700 (Western Indonesia Time)
RegExp: /[A-Z]/, Ini Sirsak = true
Object: nama = Ruslan, umur = 21
```

b. Variabel dan Array

membuat beberapa variabel sederhana (nama, umur, status mahasiswa) dan juga membuat array satu dimensi dan dua dimensi.

Kemudian saya menambahkan dan menghapus data dari array menggunakan metode `.push()` dan `.pop()`.

```
1  var nama = "Dewi";
2  var umur = 21;
3  var isMahasiswa = true;
4  document.getElementById("test").innerHTML =
5  "Nama: " + nama + "<br>" +
6  "Umur: " + umur + "<br>" +
7  "Mahasiswa: " + isMahasiswa;
8
9  var data = ["Satu", 2, true, "Tiga", 4];
10 var arr2 = [["satu", "dua"], ["tiga", "empat"]];
11 document.getElementById("array").innerHTML =
12 "Elemen pertama: " + data[0] + "<br>" +
13 "Elemen ketiga: " + data[2] + "<br>" +
14 "Isi arr2: " + arr2 + "<br>" +
15 "Elemen arr2[0][1]: " + arr2[0][1] + "<br>" +
16 data[10];
17
18 var data2 = ["pisang", "apel", "mangga"];
19 document.getElementById("array2").innerHTML =
20 "Isi awal array: " + data2 + "<br>" +
21 "Panjang awal array: " + data2.length + "<br><br>" +
22 "Menambahkan durian (push): " + data2.push("durian") + "<br>" +
23 "Isi array setelah push: " + data2 + "<br>" +
24 "Menghapus elemen terakhir (pop): " + data2.pop() + "<br>" +
25 "Isi array setelah pop: " + data2 + "<br>";
```

Penjelasan:

Array itu seperti “lemari data” bisa menampung banyak nilai dalam satu variabel.

Dengan `.push()` kita bisa menambah isi di akhir array, sedangkan `.pop()` berfungsi untuk mengambil dan menghapus data terakhir dari array.

Output:

Isi awal array: pisang,apel,mangga

Panjang awal array: 3

Menambahkan durian (push): 4

Isi array setelah push: pisang,apel,mangga,durian

Menghapus elemen terakhir (pop): durian

Isi array setelah pop: pisang,apel,mangga

c. Percabangan (If-Else)

membuat kondisi sederhana untuk menentukan gelar akademik berdasarkan variabel pendidikan.

Kalau nilainya S1 maka hasilnya “Sarjana”, kalau S2 jadi “Master”, dan seterusnya.

```
1  var pendidikan = "S2";
2  var gelar;
3  if (pendidikan === "S1") {
4      gelar = "Sarjana";
5  } else if (pendidikan === "S2") {
6      gelar = "Master";
7  } else if (pendidikan === "S3") {
8      gelar = "Doktor";
9  } else {
10     gelar = "Tidak diketahui";
11 }
12 document.getElementById("outputPercabangan").innerHTML =
13 "Sekarang kamu: " + pendidikan + ", gelar kamu: " + gelar;
```

Penjelasan:

Percabangan if-else digunakan ketika program harus mengambil keputusan. Contohnya, ketika kita ingin menentukan hasil yang berbeda tergantung isi suatu variabel.

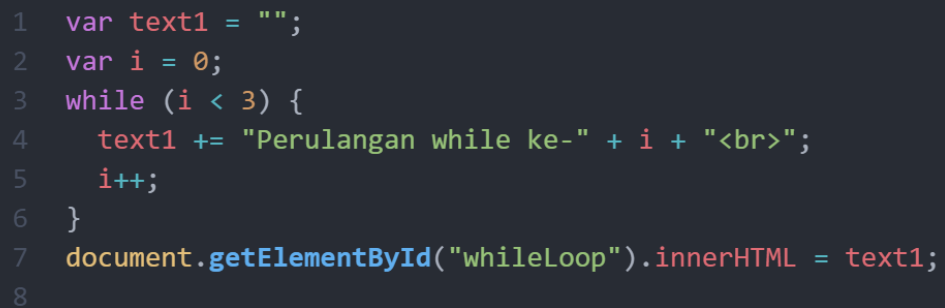
Output:

Sekarang kamu: S2, gelar kamu: Master

d. Perulangan While

membuat perulangan menggunakan while. Nilai i dimulai dari 0 dan akan terus bertambah sampai mencapai angka 3.

Setiap kali perulangan, program menampilkan teks di browser.

A code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a light-colored font and is as follows:

```
1  var text1 = "";
2  var i = 0;
3  while (i < 3) {
4      text1 += "Perulangan while ke-" + i + "<br>";
5      i++;
6  }
7  document.getElementById("whileLoop").innerHTML = text1;
8
```

Penjelasan:

Perulangan while digunakan saat kita ingin mengulang suatu perintah beberapa kali.

Selama kondisi masih benar ($i < 3$), program akan terus mengulang.

Output:

Perulangan while ke-0

Perulangan while ke-1

Perulangan while ke-2

e. Object dan Nested Object

membuat objek mobil yang punya beberapa informasi seperti merk dan tahun.

Lalu saya membuat jadwal yang berisi dua objek di dalamnya: asal dan tujuan.

```

1  var namaObject = { key1: "nilai1", key2: "nilai2", key3: "nilai3" };
2  var objek_kosong = {};
3  var mobil = {
4      "warna-badan": "merah",
5      "nomor-polisi": "BK1234AB",
6      merk: "Toyota",
7      tahun: 2020
8  };
9  document.getElementById("object").innerHTML =
10 "Merk mobil: " + mobil.merk + "<br>" +
11 "Warna badan: " + mobil["warna-badan"] + "<br>" +
12 "Nomor polisi: " + mobil["nomor-polisi"] + "<br>" +
13 "Tahun: " + mobil.tahun + "<br>";
14
15 // =====
16 // NESTED OBJECT
17 // =====
18 var jadwal = {
19     platform: 34,
20     telah_berangkat: false,
21     asal: { kode_kota: "MDN", nama_kota: "Medan", waktu: "2023-12-29 14:00" },
22     tujuan: { kode_kota: "JKT", nama_kota: "Jakarta", waktu: "2023-12-29 17:30" }
23 };
24 document.getElementById("outputObject").innerHTML =
25 "Platform: " + jadwal.platform + "<br>" +
26 "Asal: " + jadwal.asal.nama_kota + "<br>" +
27 "Tujuan: " + jadwal.tujuan.nama_kota + "<br>" +
28 "Waktu Berangkat: " + jadwal.asal.waktu;

```

Penjelasan:

Object bisa dibilang seperti “wadah data” yang berisi pasangan nama dan nilai (key-value).

Nested object berarti objek di dalam objek, berguna untuk data yang lebih terstruktur seperti jadwal keberangkatan.

Output:

Object Pada JavaScript

Merk mobil: Toyota
Warna badan: merah
Nomor polisi: BK1234AB
Tahun: 2020

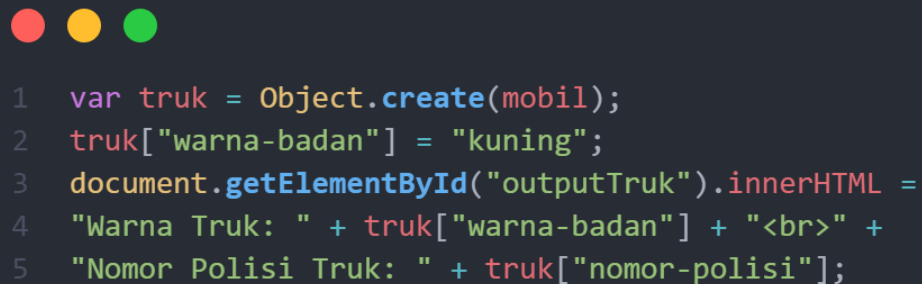
Nested Object

Platform: 34
Asal: Medan
Tujuan: Jakarta
Waktu Berangkat: 2023-12-29 14:00

F. Prototype pada JavaScript

membuat objek baru bernama truk yang dibuat dari objek mobil menggunakan `Object.create()`.

Lalu saya ubah warnanya supaya tidak sama dengan mobil aslinya.



```
1  var truk = Object.create(mobil);
2  truk["warna-badan"] = "kuning";
3  document.getElementById("outputTruk").innerHTML =
4  "Warna Truk: " + truk["warna-badan"] + "<br>" +
5  "Nomor Polisi Truk: " + truk["nomor-polisi"];
```

Penjelasan:

Prototype di JavaScript digunakan untuk mewarisi properti dari objek lain. Dengan cara ini, truk punya semua atribut mobil tanpa harus menulis ulang, tapi tetap bisa dimodifikasi.


Output:

Warna Truk: kuning
Nomor Polisi Truk: BK1234AB

g. Fungsi Pada JavaScript

membuat dua fungsi, tambah() dan kali(), yang digunakan untuk menghitung penjumlahan dan perkalian.

Setelah itu, saya memanggil fungsi tersebut dengan beberapa nilai berbeda.



```
1  function tambah(a, b) {  
2      return a + b;  
3  }  
4  var kali = function (a, b) {  
5      return a * b;  
6  };  
7  document.getElementById("hasilTambah").innerHTML =  
8  "tambah(3,5) = " + tambah(3,5) + "<br>" +  
9  "tambah(5,2) = " + tambah(5,2);  
10 document.getElementById("hasilKali").innerHTML =  
11 "kali(4,5) = " + kali(4,5);
```

Penjelasan:

Fungsi digunakan untuk mengelompokkan logika agar mudah digunakan berulang-ulang. Daripada menulis rumus berulang kali, kita cukup panggil fungsi dengan nilai baru.

Output:

tambah(3,5) = 8

tambah(5,2) = 7

kali(4,5) = 20

h. jQuery

menggunakan jQuery untuk mengganti teks dan warna pada elemen HTML. jQuery dijalankan ketika halaman selesai dimuat dengan `$(document).ready()`.

```
1  $(document).ready(function() {  
2    // 1. Text  
3    var hobi = ["Membaca", "Olahraga", "Ngoding"];  
4    $("#hasil").text("Hobi kedua saya: " + hobi[1]);  
5  
6    // 2. If-Else  
7    var nilai = 75;  
8    if (nilai >= 70) {  
9      $("#nilai").text("Selamat, kamu lulus!").css("color", "green");  
10   } else {  
11     $("#nilai").text("Maaf, kamu belum lulus.").css("color", "red");  
12   }  
}
```

Penjelasan:

jQuery membuat pengaturan elemen HTML jadi lebih mudah.

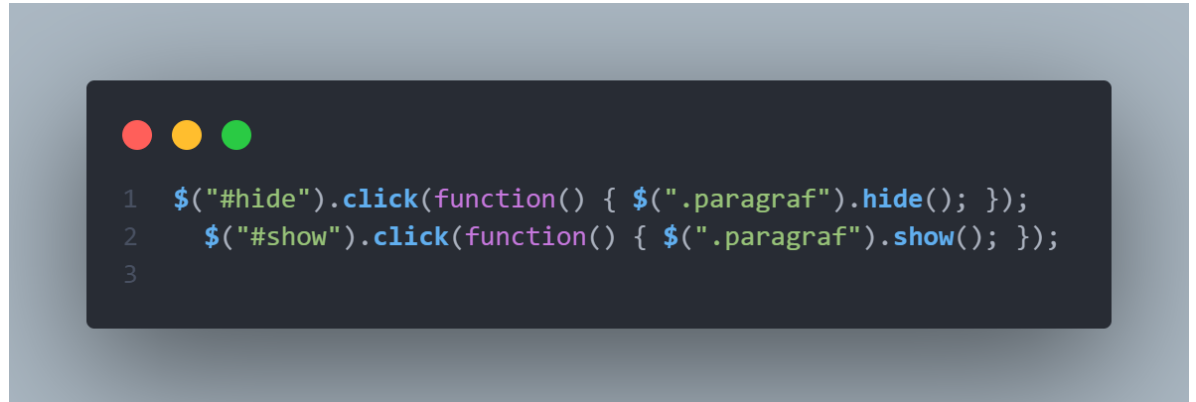
Dengan `.text()` kita bisa mengubah teks, dan `.css()` untuk mengganti warna atau tampilan lainnya.

Output:

Hobi kedua saya: Olahraga

Selamat, kamu lulus!

- i. Efek Hide/Show
menambahkan dua tombol untuk menyembunyikan (Hide) dan menampilkan (Show) paragraf.



Penjelasan:

jQuery bisa membuat efek visual di halaman tanpa CSS tambahan.

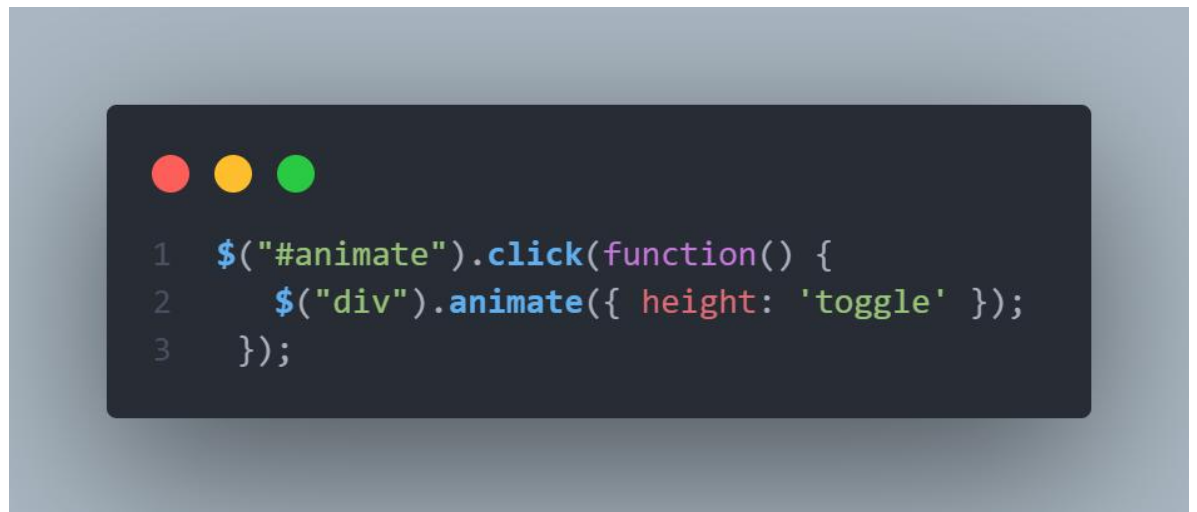
.hide() dan .show() mengontrol tampilan elemen,

Output:



- j. Animasi

tombol Start Animation untuk membuat kotak bergerak naik-turun menggunakan .animate().



Penjelasan:

.animate() membuat efek gerakan yang lebih halus.

Output:

Click tombol untuk melihat animasi:

Start Animation



B. UNGUIDED (Tugas Mandiri)

(Bagian ini berisi latihan atau tugas tambahan yang diberikan di akhir modul.

Tuliskan nomor soal, isi soalnya, source code jawaban, screenshot output, dan penjelasan setiap nomor.)

1. Soal 1:

Sebuah toko ingin membuat program sederhana menggunakan **HTML dan JavaScript** untuk menghitung **total harga pembelian** dari beberapa barang. Setiap barang memiliki **harga satuan** dan **jumlah pembelian** yang sudah ditentukan. Berikut adalah daftar barang:

Nama Barang	Harga Satuan	Jumlah
Susu UHT	Rp 42.000	10
Roti Tawar	Rp 30.000	3
Mie Instan	Rp 20.000	5
Sosis	Rp 51.000	7

Jawaban:

Source Code:

```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <title>Tugas 1 - Total Harga Pembelian</title>
6 </head>
7 <body>
8   <h2>Program Total Harga Pembelian</h2>
9   <script>
10    const barang = [
11      { nama: "Susu UHT", harga: 42000, jumlah: 10 },
12      { nama: "Roti Tawar", harga: 30000, jumlah: 3 },
13      { nama: "Mie Instan", harga: 20000, jumlah: 5 },
14      { nama: "Sosis", harga: 51000, jumlah: 7 }
15    ];
16
17    let total = 0;
18    document.write("<table border='1' cellpadding='5'><tr><th>Nama Barang</th><th>Harga Satuan</th><th>Jumlah</th><th>Subtotal</th></tr>");
19
20    barang.forEach(item => {
21      const subtotal = item.harga * item.jumlah;
22      total += subtotal;
23      document.write("<tr>
24        <td>${item.nama}</td>
25        <td>Rp ${item.harga.toLocaleString()}</td>
26        <td>${item.jumlah}</td>
27        <td>Rp ${subtotal.toLocaleString()}</td>
28      </tr>");
29    });
30
31    document.write("</table><h3>Total Pembelian: Rp ${total.toLocaleString()}</h3>");
32  </script>
33 </body>
34 </html>
35
```

Output:

Program Total Harga Pembelian

Nama Barang	Harga Satuan	Jumlah	Subtotal
Susu UHT	Rp 42.000	10	Rp 420.000
Roti Tawar	Rp 30.000	3	Rp 90.000
Mie Instan	Rp 20.000	5	Rp 100.000
Sosis	Rp 51.000	7	Rp 357.000

Total Pembelian: Rp 967.000

Penjelasan:

Tugas ini menghitung total harga dari beberapa barang menggunakan JavaScript dasar.

Data barang disimpan dalam array of object yang berisi nama, harga satuan, dan jumlah.

Program melakukan perulangan untuk menghitung subtotal tiap barang, lalu menjumlahkannya menjadi total akhir.

Hasil perhitungan ditampilkan dalam bentuk tabel dengan kolom nama, harga, jumlah, dan subtotal.

Tujuannya agar pengguna bisa melihat total harga pembelian secara otomatis tanpa menghitung manual.

2. Soal 2:

(Easy) Buat halaman HTML yang menampilkan teks “Belajar jQuery itu asik!” di elemen `<p>` dengan `id="teks"`. Gunakan jQuery untuk mengganti isi teks tersebut menjadi “Halo Dunia dengan jQuery!” ketika halaman selesai dimuat.

Jawaban:

Source Code:



```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <title>Tugas 2 - jQuery Easy</title>
6   <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
7 </head>
8 <body>
9   <p id="teks">Belajar jQuery itu asik!</p>
10
11   <script>
12     $(document).ready(function() {
13       $("#teks").text("Halo Dunia dengan jQuery!");
14     });
15   </script>
16 </body>
17 </html>
18
```

Output:

Halo Dunia dengan jQuery!

Penjelasan:

Tugas ini menggunakan jQuery untuk mengubah isi teks setelah halaman selesai dimuat.

Elemen `<p>` dengan id “teks” awalnya menampilkan kalimat “Belajar jQuery itu asik!”.

Ketika dokumen siap (`$(document).ready()`), jQuery mengganti teks tersebut menjadi “Halo Dunia dengan jQuery!”.

Ini menunjukkan kemampuan jQuery dalam memanipulasi konten HTML secara dinamis.

Tujuan tugas ini adalah mengenalkan dasar penggunaan jQuery untuk mengubah elemen di halaman web.

3. Soal 3:

Buat halaman HTML dengan satu tombol dan satu paragraf. Ketika tombol diklik:

- teks pada paragraf berubah jadi **“Tombol sudah diklik!”**
- warnanya berubah jadi **merah**

Source code:

```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <title>Tugas 3 - jQuery Medium</title>
6   <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
7 </head>
8 <body>
9   <button id="tombol">Klik Saya!</button>
10  <p id="pesan">Paragraf awal</p>
11
12  <script>
13    $(document).ready(function() {
14      $("#tombol").click(function() {
15        $("#pesan").text("Tombol sudah diklik!").css("color", "red");
16      });
17    });
18  </script>
19 </body>
20 </html>
21
```

Output:



Penjelasan:

Tugas ini membuat tombol yang bisa mengubah isi dan warna teks pada paragraf ketika diklik.

Dengan bantuan jQuery, event click() digunakan untuk mendeteksi aksi

pengguna.

Saat tombol ditekan, isi paragraf berubah menjadi “Tombol sudah diklik!” dan warnanya berubah menjadi merah.

4. Soal 4:

Buat satu elemen `<div>` berbentuk kotak (misalnya 100x100px, warna biru).

Tambahkan dua tombol:

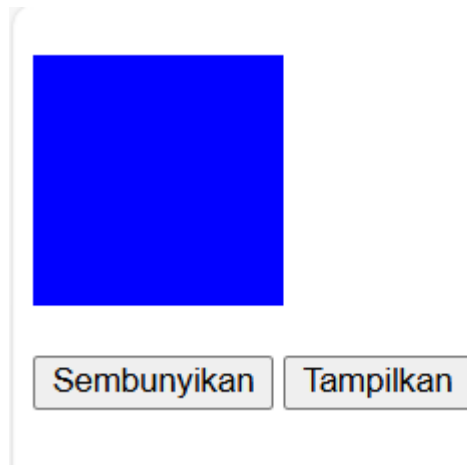
- Sembunyikan → membuat kotak hilang perlahan (fade out)
- Tampilkan → membuat kotak muncul perlahan (fade in)

Source code:



```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <title>Tugas 4 - jQuery Hard</title>
6   <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
7   <style>
8     #kotak {
9       width: 100px;
10      height: 100px;
11      background-color: blue;
12      margin: 20px 0;
13    }
14  </style>
15 </head>
16 <body>
17   <div id="kotak"></div>
18   <button id="sembunyi">Sembunyikan</button>
19   <button id="tampil">Tampilkan</button>
20
21   <script>
22     $(document).ready(function() {
23       $("#sembunyi").click(function() {
24         $("#kotak").fadeOut();
25       });
26       $("#tampil").click(function() {
27         $("#kotak").fadeIn();
28       });
29     });
30   </script>
31 </body>
32 </html>
33
```

Output:



Penjelasan:

Tugas ini membuat sebuah kotak berwarna biru yang bisa disembunyikan dan ditampilkan dengan efek halus.

Dua tombol digunakan untuk memanggil fungsi `fadeOut()` dan `fadeIn()` milik jQuery.

Ketika tombol “Sembunyikan” diklik, kotak perlahan menghilang, sedangkan “Tampilkan” membuatnya muncul kembali.

Efek transisi ini memperlihatkan kemampuan jQuery dalam membuat animasi sederhana tanpa CSS tambahan.

BAB III

PENUTUP

A. Kesimpulan

Dari hasil praktikum dan tugas mandiri JavaScript serta jQuery, dapat disimpulkan bahwa JavaScript berperan penting dalam mengatur logika dan proses interaktif di dalam website, seperti pengolahan data, percabangan, dan perulangan. Sementara itu, jQuery mempermudah manipulasi elemen HTML, penanganan event, serta pembuatan efek visual dengan sintaks yang lebih sederhana. Melalui kombinasi keduanya, pengembangan web menjadi lebih efisien, dinamis, dan menarik bagi pengguna. Praktikum ini juga membantu memahami dasar-dasar pemrograman client-side yang menjadi fondasi penting dalam pembuatan aplikasi web modern.