**LAPORAN RESMI**

**MODUL IV**

**JavaScript**

**PEMROGRAMAN BERBASIS WEB**



**NAMA : DONY EKA OCTAVIAN PUTRA**

**N.R.P : 230441100041**

**DOSEN : FIRLI IRHAMNI,S.T., M.KOM.**

**ASISTEN : MUHAMMAD IRHAM ALKARIM**

**TGL PRAKTIKUM : MARET 2024**

**Disetujui : .. APRIL 2024**

**Asisten**

**MUHAMMAD IRHAM ALKARIM 20.04.411.00111**

**LABORATORIUM BISNIS INTELIJEN SISTEM**

**PRODI SISTEM INFORMASI**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

# **BAB I PENDAHULUAN**

## **Latar Belakang**

JavaScript adalah bahasa pemrograman yang dikembangkan oleh Brendan Eich pada tahun 1995 di Netscape Communications Corporation. Awalnya disebut Mocha, kemudian LiveScript, dan akhirnya dinamai JavaScript. Bahasa ini memungkinkan pengembang web menciptakan halaman web yang dinamis dan interaktif, dengan kemampuan untuk mengubah konten secara real-time tanpa perlu memuat ulang halaman. Dukungan dari browser utama seperti Netscape Navigator dan Internet Explorer membantu menjadikan JavaScript sebagai standar de facto untuk pengembangan web interaktif.

Pada tahun 1997, JavaScript distandardisasi oleh Ecma International sebagai ECMA-262, yang kemudian dikenal sebagai ECMAScript. Standardisasi ini memastikan konsistensi dan kompatibilitas antar berbagai browser. Sejak itu, ECMAScript telah mengalami banyak revisi dan pembaruan, dengan versi terbaru yang memperkenalkan fitur-fitur canggih dan efisien. JavaScript telah berevolusi dari bahasa scripting sederhana menjadi bahasa pemrograman yang kuat dan serbaguna, digunakan tidak hanya untuk pengembangan front-end tetapi juga untuk back-end melalui platform seperti Node.js.

JavaScript memiliki ekosistem yang kaya dengan banyak pustaka dan kerangka kerja seperti React, Angular, dan Vue.js yang mempercepat dan menyederhanakan proses pengembangan. Selain itu, komunitas pengembang yang besar dan aktif terus berkontribusi dengan berbagai alat, pustaka, dan sumber daya yang menjaga JavaScript tetap relevan dan inovatif. Dengan fleksibilitas dan kemampuannya untuk beradaptasi dengan kebutuhan teknologi yang terus berkembang, JavaScript diprediksi akan tetap menjadi bahasa utama dalam pengembangan aplikasi web dan teknologi lainnya di masa depan.

## 1.2 Tujuan

1. Mahasiswa mampu memahami dasar pemrograman JavaScript.
2. Mahasiswa mampu memahami Event dan DOM Manipulation.

# **BAB II DASAR TEORI**

## **Pengenalan dan Sejarah Singkat JavaScript**

JavaScript, bahasa pemrograman yang sangat populer dalam pengembangan web, pertama kali dikembangkan oleh Brendan Eich di Netscape Communications pada tahun 1995. Awalnya dikenal sebagai Mocha dan kemudian LiveScript, sebelum akhirnya dinamai JavaScript. Pada tahun 1997, JavaScript diajukan sebagai standar ECMAScript kepada European Computer Manufacturers Association (ECMA). Sejak itu, JavaScript terus berkembang dengan penambahan fitur baru dan peningkatan kinerja melalui rilis ECMAScript yang berkala, seperti ES6 (ECMAScript 2015).

Dengan kemampuannya untuk menambahkan interaktivitas ke halaman web, JavaScript menjadi bahasa pemrograman yang sangat populer. Selain digunakan dalam pengembangan web, JavaScript juga digunakan dalam pengembangan aplikasi mobile, desktop, permainan, dan bahkan di bidang pengembangan server (Node.js). JavaScript saat ini menjadi salah satu elemen kunci dalam ekosistem pengembangan web modern, bersama dengan HTML dan CSS, memungkinkan pembuatan aplikasi web yang kompleks dan interaktif.

## **2.2 Variable dan Tipe Data**

1. **Variable**

Variabel digunakan untuk menyimpan data dalam JavaScript. Mendeklarasikan variabel dilakukan dengan menggunakan kata kunci var, let, atau const.

Penjelasan :

* var dan let: digunakan untuk mendefinisikan bahwa suatu variabel tersebut nilainya dapat berubah.
* const : digunakan untuk mendefinisikan bahwa suatu variabel tersebut nilainya tidak dapat berubah (konstan). Syntax umum :
* var/let/const [nama\_variabel] = nilai;

Contoh :

→ var namaLengkap = “otong surotong”;

→ let nama\_panggilan = ‘surot;

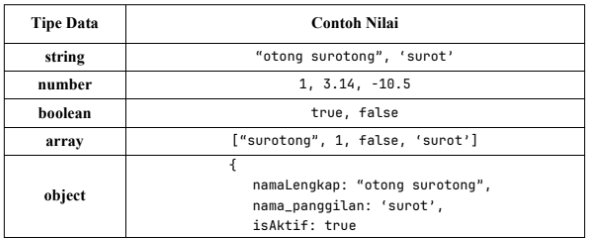
→ const PI = 3.14;

1. **Tipe Data**

JavaScript memiliki beberapa tipe data dasar seperti string, number, boolean, null, undefined, array, dan object. Setiap tipe data memiliki peran dan karakteristiknya sendiri.

Penjelasan :

* string : digunakan untuk merepresentasikan teks atau urutan karakter. String dituliskan di dalam tanda kutip, baik tunggal ('') maupun ganda ("").
* number : digunakan untuk merepresentasikan nilai numerik. Ini bisa berupa bilangan bulat (integer) atau bilangan pecahan (floating point).
* boolean : digunakan untuk merepresentasikan 2 nilai yaitu (true) atau (false).
* null : digunakan untuk menunjukkan bahwa sebuah variabel tidak memiliki nilai atau tidak memiliki nilai yang valid. Secara harfiah, null berarti "tidak ada nilai".
* undefined : digunakan untuk menunjukkan bahwa sebuah variabel telah dideklarasikan tetapi tidak memiliki nilai yang diberikan.
* array : digunakan untuk menyimpan kumpulan nilai dalam satu variabel. Elemen-elemen dalam array diindeks, dimulai dari indeks 0.
* •object : digunakan untuk merepresentasikan sebuah objek, yang terdiri dari kumpulan pasangan kunci-nilai (key-value pairs). Objek dapat memiliki properti dan metode.





Setiap tipe data dalam JavaScript memiliki karakteristik dan penggunaan masing-masing, dan memahami mereka secara mendalam sangat penting dalam pengembangan aplikasi JavaScript yang efektif.

## **2.3 Operator dan Ekspresi**

1. **Operator**

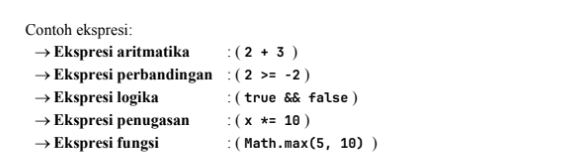
Operator adalah simbol atau kata kunci yang digunakan untuk melakukan operasi pada operand (nilai atau variabel). Di JavaScript, ada beberapa jenis operator yang berbeda.

Diantaranya :

* Operator aritmatika : digunakan untuk melakukan operasi matematika pada operand. Contoh ( +, -, \*, /, % ).
* Operator perbandingan : digunakan untuk membandingkan dua nilai atau variabel dan menghasilkan nilai boolean. Contoh ( <, <=, >, >=, ==, !=, ===, !== ).
* Operator logika : digunakan untuk menggabungkan atau membalikkan kondisi logika dan menghasilkan nilai boolean. Contoh ( &&, ||, ! ).
* Operator penugasan : digunakan untuk memberikan nilai kepada variabel. Contoh ( =, +=, -=, \*=, dll ).

1. **Ekspresi**

Ekspresi adalah kombinasi variabel, nilai, operator, dan/atau fungsi yang dievaluasi menjadi nilai tunggal. Ekspresi dapat berupa ekspresi aritmatika, ekspresi logika, atau ekspresi yang lebih kompleks.



Ekspresi dievaluasi menjadi nilai tunggal sesuai dengan aturan dan operasi yang dinyatakan di dalamnya. Ekspresi ini sering digunakan dalam kondisi, perulangan, dan pernyataan lainnya dalam JavaScript untuk mengontrol alur eksekusi program.

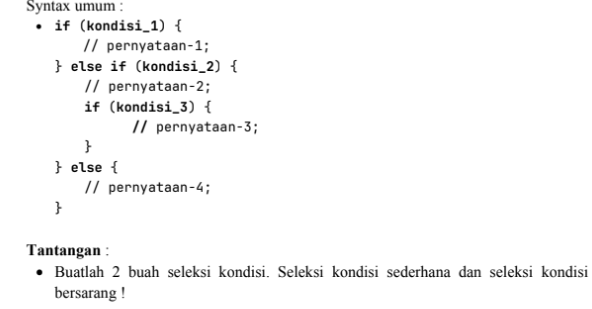
Memahami operator dan ekspresi adalah kunci dalam pemrograman JavaScript, karena mereka memungkinkan kita untuk melakukan berbagai macam operasi dan mengendalikan alur eksekusi program secara efektif.

## **2.4 Struktur Kontrol**

Dalam pemrograman JavaScript, struktur kontrol digunakan untuk mengontrol alur eksekusi program berdasarkan kondisi tertentu atau untuk mengulangi blok kode tertentu. Dengan memahami struktur kontrol, Anda dapat membuat program yang lebih dinamis dan responsif. Berikut adalah beberapa konsep struktur kontrol yang penting:

1. **Seleksi Kondisi (if, else if, else)**

Seleksi kondisi digunakan untuk menjalankan blok kode tertentu berdasarkan kondisi yang diberikan. Pernyataan if digunakan untuk mengevaluasi ekspresi boolean, jika ekspresi tersebut benar (true), maka blok kode di dalam if akan dieksekusi. Jika ekspresi tersebut salah (false), maka blok kode di dalam else akan dieksekusi. else if dapat digunakan untuk mengevaluasi beberapa kondisi.



1. **Pengulangan (for, while, do-while)**

Pengulangan digunakan untuk mengeksekusi blok kode berulang kali. JavaScript menyediakan beberapa jenis pengulangan, yaitu for, while, dan do-while. Loop for digunakan ketika Anda mengetahui berapa kali iterasi yang dibutuhkan, while digunakan ketika Anda hanya ingin menjalankan loop selama kondisi tertentu terpenuhi, dan do-while digunakan ketika Anda ingin menjalankan loop setidaknya sekali sebelum mengevaluasi kondisi.

Syntax umum :

* for → for (inisialisasi; kondisi; iterasi) {

// pernyataan yang ingin diulang;

}

* while → while (kondisi) {

// Blok kode yang akan diulang

// Perhatikan bahwa kondisi akan dievaluasi sebelum setiap iterasi

}

* do-while → do {

// Blok kode yang akan diulang

// Perhatikan bahwa blok kode akan dijalankan

setidaknya sekali sebelum kondisi dievaluasi

} while (kondisi);

**Tantangan :**

* Praktikan 3 macam perulangan, dan coba pahami cara kerjanya Dengan memahami dan menguasai struktur kontrol ini, Anda dapat mengendalikan alur eksekusi program dengan lebih efisien dan fleksibel dalam pengembangan aplikasi JavaScript.

## **2.5 Fungsi**

Fungsi adalah blok kode yang dirancang untuk melakukan tugas tertentu. Dengan menggunakan fungsi, Anda dapat mengorganisir kode Anda menjadi bagian-bagian yang lebih kecil dan dapat digunakan kembali. Dalam JavaScript, fungsi sangat penting dan dapat digunakan untuk berbagai tujuan, mulai dari mengeksekusi tugas sederhana hingga mengelola logika yang kompleks dalam aplikasi Anda.

Kegunaan fungsi :

* Fungsi digunakan untuk mengelompokkan serangkaian pernyataan menjadi satu unit yang dapat dipanggil.
* Saat fungsi dipanggil, blok kode di dalamnya dieksekusi.

Syntax umum :

→ function namaFungsi(parameter1, parameter2, ...) {

// Blok kode yang akan dieksekusi saat fungsi dipanggil

}

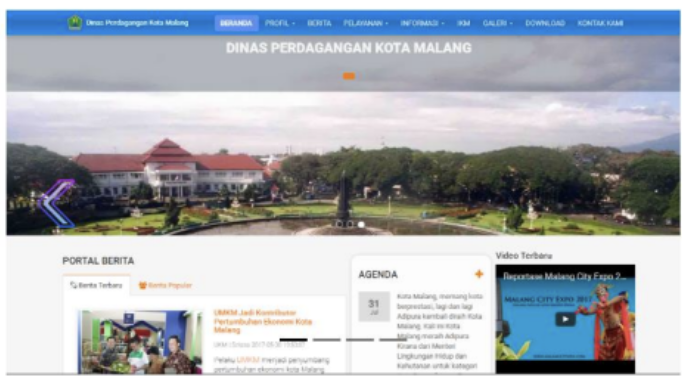
**Tantangan :**

* Buatlah sebuah fungsi yang berguna untuk menghitung keliling lingkaran dan luas lingkaran, dan panggil tersebut dengan nilai argument yang berbeda – beda.

# **BAB IV IMPLEMENTASI**

## **Soal**

1. Buat halaman web yang berisikan yang memuat tentang CSS Image, CSS Navigasi, CSSLayout dan CSS Dropdown, sesuai dengan apa yang telah dipelajari pada modul 3 ini. Contoh tampilan adalah seperti ini:



## **Source Code**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Navbar</title>

<style>

  body {

    margin: 0;

    font-family: Arial, sans-serif;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    background-image: url('BG.jpg');

    background-size: cover;

    background-position: center;

  }

  .navbar {

    background-color: #24a8e0;

    text-align: center;

    padding: 20px 0;

    width: 100%;

    position: fixed;

    top: 0;

  }

  .navbar a {

    color: white;

    text-decoration: none;

    margin: 0 20px;

    padding: 10px;

    transition: all 0.3s ease;

  }

  .navbar a:hover {

    background-color: #ffffff;

    color: #24a8e0;

    transform: scale(1.1);

  }

  .dropdown {

    position: relative;

    display: inline-block;

    color: white;

  }

  .dropdown-content {

    display: none;

    position: absolute;

    background-color: #24a8e0;

    min-width: 100px;

    z-index: 1;

    text-align: left; /\* Text sejajar ke kiri \*/

    padding: 10px 0; /\* Padding atas bawah pada dropdown \*/

    margin-left: -20px; /\* Menyesuaikan posisi dropdown-content dengan BERANDA \*/

  }

  .dropdown-content a {

    color: #ffffff;

    display: block;

    padding: 10px 20px; /\* Padding kiri kanan pada setiap link \*/

    text-decoration: none;

    transition: all 0.3s ease;

  }

  .dropdown-content a:hover {

    background-color: #ffffff;

    color: #24a8e0;

  }

  .dropdown:hover .dropdown-content {

    display: block;

  }

  .content {

    text-align: center;

    color: white;

    display: flex;

    flex-wrap: wrap;

    justify-content: center;

  }

  .profile {

    width: 100vw;

    height: 100vh;

  }

  .profile img {

    width: 100%;

    height: 100%;

    object-fit: cover;

  }

  .TULISAN {

    position: absolute;

    top: 15%;

    left: 50%;

    transform: translate(-50%, -50%);

    z-index: 10;

    color: white;

    font-size: 20px;

    text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.5);

  }

  .informasi {

    display: flex;

    justify-content: space-between; /\* Mengatur jarak antar kolom \*/

    width: 80%;

    margin: 80px auto; /\* Memberikan margin atas dan bawah, serta mengatur posisi horizontal menjadi di tengah \*/

    overflow-y: auto;

    height: 50vh;

  }

  .informasi > div {

    background-color: rgba(27, 23, 23, 0.5);

    padding: 20px;

    border-radius: 10px;

    text-align: center;

    width: 30%; /\* Lebar kolom diatur 48% agar ada jarak antar kolom \*/

  }

  .colom {

    background-color: rgba(27, 23, 23, 0.5);

    padding: 20px;

    border-radius: 10px;

    text-align: center;

    width: 50%; /\* Lebar kolom diatur 48% agar ada jarak antar kolom \*/

    position: relative;

  }

  .informasi > div img,

  .colom img {

    width: 50%;

    height: 50%;

    object-fit: cover; /\* Memastikan foto tetap proporsional \*/

    margin: 20px auto;

  }

  @media (max-width: 768px) {

    .profile {

      display: none;

    }

    .informasi {

      width: 100%;

    }

    .informasi > div {

      width: 100%;

    }

  }

</style>

</head>

<body>

<div class="navbar">

  <div class="dropdown">

    <a href="">BERANDA</a>

    <div class="dropdown-content">

      <a href="">FOTOGRAFI</a>

      <a href="">VIDEOGRAFI</a>

      <a href="">PORTOFOLIO</a>

    </div>

  </div>

  <div class="dropdown">

    <a href="">PROFIL</a>

    <div class="dropdown-content">

      <a href="">SENDIRI</a>

      <a href="">BERSAMA</a>

    </div>

  </div>

  <a href="">BERITA</a>

  <a href="">KECELAKAAN</a>

  <a href="">PELANGGAN</a>

  <a href="">INFORMASI</a>

  <a href="">BERITA</a>

  <a href="">KONTAK</a>

</div>

<div class="content">

  <div class="profile">

    <img src="4527182.jpg" alt="profile" />

  </div>

  <div class="TULISAN">

    <h1>RUMAH MAKAN DONZZ</h1>

  </div>

  <div class="informasi">

    <div>

      <h2>Informasi 1</h2>

      <p>

        Rendang adalah masakan tradisional Indonesia yang berasal dari Minangkabau, Sumatera Barat.

        Makanan ini terkenal di Indonesia bahkan hingga mancanegara sebagai salah satu masakan terenak di dunia.

        Rendang terbuat dari potongan daging sapi yang dimasak dalam waktu yang cukup lama dengan campuran rempah-rempah dan santan,

        hingga kuahnya mengering dan bumbunya meresap ke dalam daging.

      </p>

      <img src="rndang.png" alt="Foto Informasi 1" />

    </div>

  <div class="colom">

    <div>

      <h2>Informasi 2</h2>

      <p>

          Makanan gule adalah masakan khas Indonesia yang berasal dari Jawa Tengah.

           Makanan ini terbuat dari daging yang dimasak dengan bumbu rempah-rempah khas Indonesia seperti bawang merah,

           bawang putih, jahe, lengkuas, serai, dan daun salam. Bumbu-bumbu ini kemudian dihaluskan dan dimasak bersama dengan santan,

           sehingga menghasilkan kuah kental yang kaya rasa.psum

      </p>

      <img src="GULE.png" alt="Foto Informasi 2" />

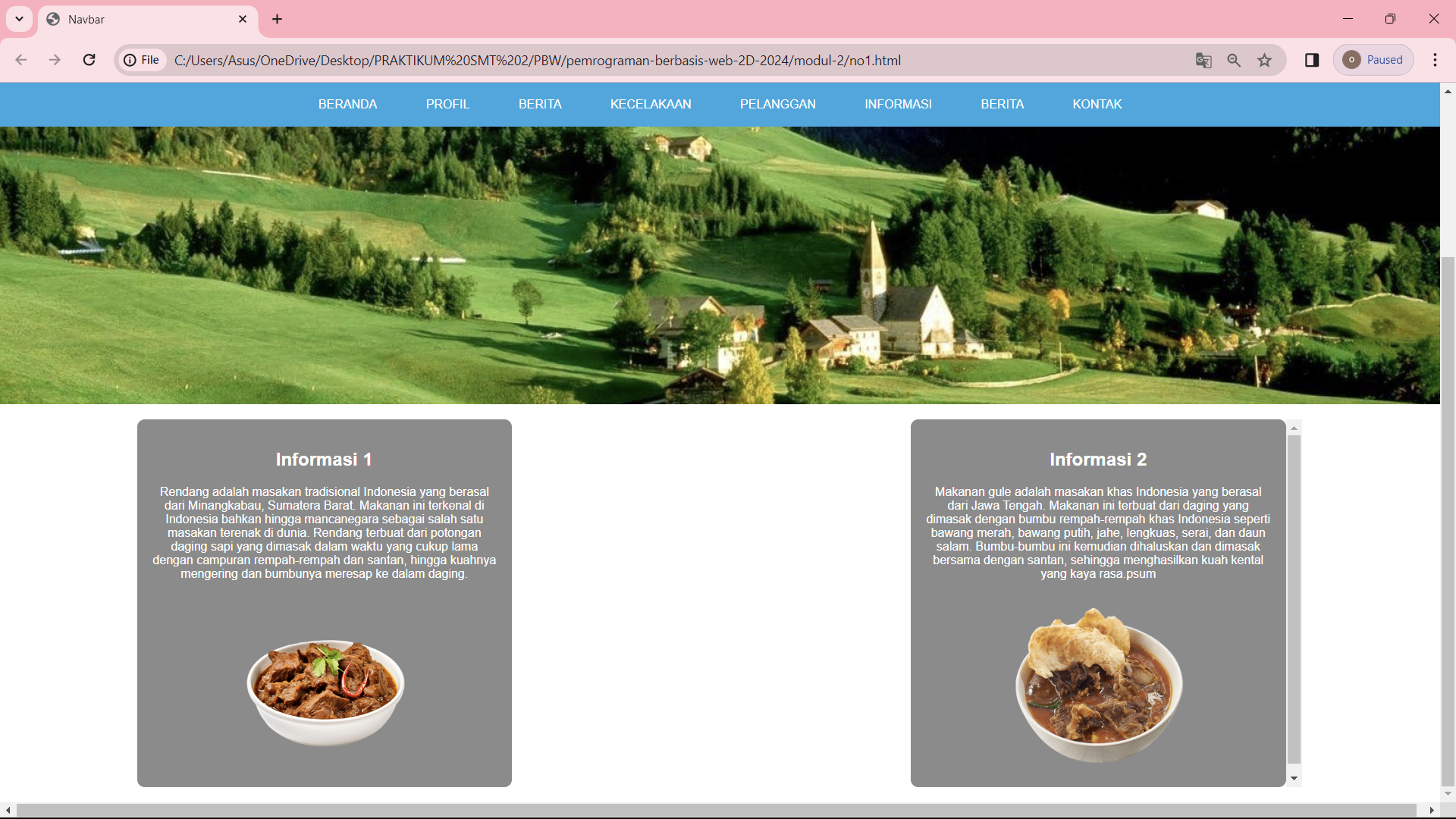
    </div>

  </div>

</body>

</html>

## **Hasil**



Penjelasan

HTML ini merupakan struktur dasar dari halaman web yang menentukan bagaimana konten akan ditampilkan. Halaman web dimulai dengan deklarasi <!DOCTYPE html> yang menunjukkan bahwa dokumen adalah dokumen HTML5. Selanjutnya, tag <html lang="en"> mengindikasikan bahwa bahasa yang digunakan adalah bahasa Inggris.

Bagian <head> berisi informasi-informasi terkait halaman web, seperti metadata (seperti charset, viewport, dan title) dan referensi ke file CSS eksternal. CSS ini digunakan untuk memperindah tampilan halaman dengan memberikan gaya pada elemen-elemen HTML.

Di dalam tag <body>, konten utama halaman web diletakkan. Terdapat sebuah navbar dengan beberapa menu navigasi, seperti Beranda, Profil, Berita, dan Kontak. Menu dropdown juga ditambahkan untuk submenu Fotografi, Videografi, dan Portofolio di menu Beranda, serta submenu Sendiri dan Bersama di menu Profil.

Setelah navbar, terdapat konten utama halaman yang berisi gambar profil, judul "RUMAH MAKAN DONZZ", dan informasi-informasi tambahan dalam bentuk kolom-kolom. Setiap kolom berisi judul, teks deskripsi, dan gambar terkait informasi tersebut.

CSS ini digunakan untuk mengatur tata letak dan gaya visual dari sebuah halaman web. Pertama, pengaturan umum seperti pengaturan font, margin, dan padding diatur agar konten terlihat rapi dan mudah dibaca. Selanjutnya, navbar (bar navigasi) dibuat dengan latar belakang biru (#24a8e0) dan link-linknya disusun secara horizontal. Efek hover diterapkan pada link-link navbar untuk memberikan respons visual saat dihover.

Dropdown menu juga ditambahkan dengan efek transisi yang halus saat menu dibuka. Selain itu, konten utama dibuat fleksibel untuk menyesuaikan tampilan pada berbagai ukuran layar. Gambar profil dan tulisan di tengah-tengah halaman diatur agar tetap terlihat menarik. Informasi-informasi tambahan diletakkan dalam kolom-kolom dengan latar belakang semi-transparan agar konten utama tetap terfokus.

Pengaturan responsivitas juga diterapkan dengan menggunakan media query untuk menyembunyikan gambar profil pada layar kecil dan membuat kolom informasi mengisi lebar layar penuh. Semua ini bertujuan untuk memberikan tampilan yang menarik dan fungsional bagi pengguna yang mengakses halaman web tersebut.

# **BAB V PENUTUP**

## **Analisa**

Pada Praktikum kali ini merupakan praktikum yang bertujuan untuk memahami dan mengimplementasikan pemahaman tentang CSS (Cascading Style Sheets) dalam pembuatan tata letak dan desain halaman web. Praktikum ini mencakup beberapa konsep dasar CSS, seperti penggunaan selektor, properti CSS, dan pengaturan tata letak.

Salah satu aspek penting dari praktikum ini adalah penerapan CSS untuk mengubah tampilan halaman web secara visual, seperti mengatur warna latar belakang, warna teks, jenis font, dan ukuran teks. Selain itu, praktikum ini juga memperkenalkan konsep penggunaan kelas (class) dan id dalam CSS untuk memisahkan gaya dari elemen HTML.

Praktikum ini memberikan pemahaman yang baik tentang bagaimana CSS dapat digunakan untuk memperindah tampilan halaman web secara efisien dan terstruktur. Melalui praktikum ini, mahasiswa dapat belajar bagaimana membuat halaman web yang responsif dan estetis dengan menggunakan CSS.

## **Kesimpulan**

1. Kesimpulan dari praktikum ini dengan judul "Modul CSS" adalah praktikum ini memberikan pemahaman yang baik tentang penggunaan CSS dalam desain dan tata letak halaman web. Melalui praktikum ini, mahasiswa dapat memahami konsep dasar CSS, seperti selektor, properti, dan nilai, serta bagaimana menggunakannya untuk mengubah tampilan halaman web.
2. Selain itu, praktikum ini juga mengajarkan penggunaan kelas dan id dalam CSS untuk memberikan gaya pada elemen HTML secara terpisah. Hal ini membantu dalam memisahkan struktur dan tampilan halaman web, sehingga memudahkan dalam pengelolaan dan pemeliharaan kode. konsep-konsep dasar CSS dan praktek langsung dalam praktikum, mahasiswa dapat mengembangkan keterampilan desain web yang lebih baik.