

Nama : Ade Amanah Putri  
NIM : K3524018  
Kelas : B  
Mata Kuliah: DESAIN WEB

## **LAPORAN PRAKTIKUM DESAIN WEB 8**

### **A. JUDUL**

Praktikum Web Storage.

### **B. TUJUAN PRAKTIKUM**

1. Mahasiswa mengenal mekanisme penyimpanan data pada HTML.
2. Mahasiswa dapat menggunakan local storage HTML.
3. Mahasiswa dapat menggunakan session storage HTML.

### **C. ALAT DAN BAHAN**

- Laptop/PC
- Text Editor seperti VS Code
- Web Browser
- Internet

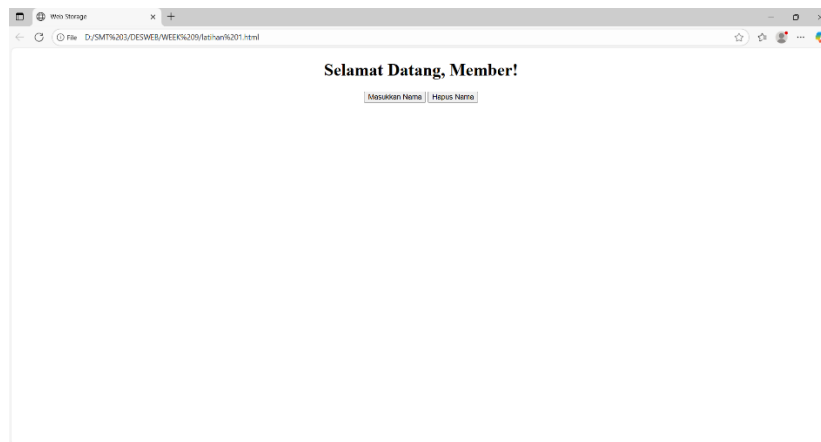
### **D. RINGKASAN MATERI**

1. Web Storage merupakan metode yang digunakan dalam website untuk menyimpan data pada sisi client. Berguna untuk menyimpan referensi, status aplikasi, data sementara, atau cache yang diperlukan selama menjelajahi web.
2. Ada 2 jenis web storage yang umum digunakan pada HTML5, pertama ada localStorage dimana data akan tetap ada bahkan setelah web ditutup. Kedua ada sessionStorage, dimana data hanya akan disimpan selama sesi browser berlangsung.
3. Fungsi-fungsi yang digunakan untuk berinteraksi dengan keduanya ada setItem(key,value), getItem(key), removeItem(key), clear, dan length.

4. Keunggulan menggunakan web storage adalah kapasitasnya besar, efisien, tidak termasuk permintaan http, penanganan data mudah, dan tersedia selama sesi.
5. Kelemahan menggunakan web storage adalah penyimpanannya terbatas, tidak cocok untuk data sensitive, tidak selalu tersedia, tidak mendukung sinkronisasi, dan tidak mendukung struktur data kompleks.

## E. OUTPUT PROGRAM

### Latihan 1:



### Latihan 2:



## F. LANGKAH KERJA

### Latihan 1:

1. Masuk ke teks editor (VS Code).
2. Mengetikkan scrip HTML untuk program web storage sesuai pada modul bagian Latihan 1.

```
<> latihan 1.html > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1">
6      <title>Web Storage</title>
7  </head>
8  <body>
9      <center>
10         <h1>Selamat Datang, Member!</h1>
11         <button onclick="gantiNama(event)">Masukkan Nama</button>
12     </center>
13
14     <script type="text/javascript">
15         const namaMember = document.querySelector('h1');
16
17         const namaTersimpan = localStorage.getItem('nama');
18         if (namaTersimpan) {
19             namaMember.innerHTML = 'Halo ' + namaTersimpan + '!';
20         }
21
22         function gantiNama(e) {
23             let nama = prompt('Masukkan nama Anda:');
24
25             if (nama && nama.trim() !== '') {
26                 localStorage.setItem('nama', nama);
27                 namaMember.innerHTML = 'Halo ' + nama + '!';
28             } else {
29                 alert('Nama tidak boleh kosong!');
30             }
31         }
32     </script>
33 </body>
34 </html>
```

- Ketika website dijalankan, halaman web akan menampilkan "Selamat Datang, Member!" kepada pengguna dan menampilkan tombol masukkan nama pada bagian tengah.
- Nama dan sapaan "Halo (nama)" akan muncul di halaman web, menggantikan sapaan default web setelah pengguna menekan tombol dan memasukkan nama.
- Ketika halaman ditutup lalu dibuka kembali (tanpa memasukkan nama), website akan menampilkan sapaan personal (dengan nama) karena data telah disimpan di local storage.

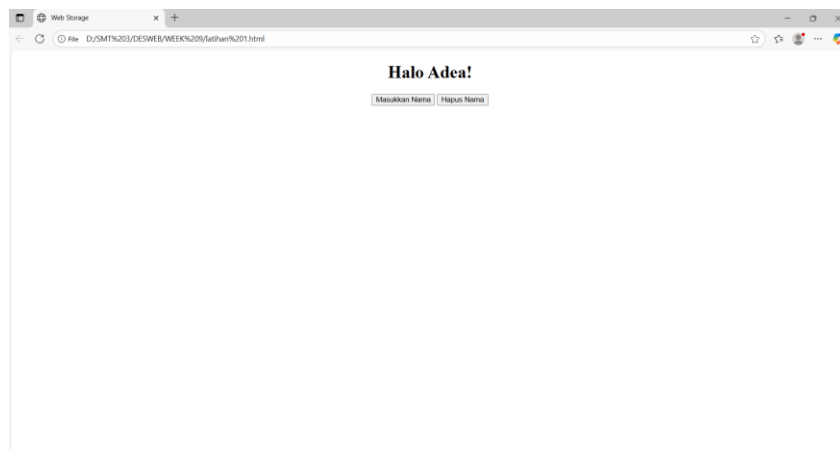
3. Menambahkan tombol hapus nama dari member.

```
8 <body>
9   <center>
10    <h1 id="namaMember">Selamat Datang, Member!</h1>
11    <button onclick="gantiNama(event)">Masukkan Nama</button>
12    <button id="hapusBtn" onclick="hapusNama()">Hapus Nama</button>
13  </center>
14
```

- Menambahkan id="namaMember" pada tag <h1> agar dapat diakses untuk seluruh manipulasi text.
- Menambahkan button dengan fungsi hapusNama( ) yang akan dipanggil saat tombol diklik untuk menghapus nama.

```
33 function hapusNama() {
34   localStorage.removeItem('nama');
35   const h1Element = document.getElementById('namaMember');
36   h1Element.innerHTML = 'Selamat Datang, Member!';
37   alert('Nama member telah dihapus.');
```

- Fungsi hapusNama( ) ditambahkan setelah fungsi gantiNama( ). Dimana Ketika nama dihapus, tampilan akan kembali lagi kepada sapaan default member.
4. Javascript tidak perlu dimodifikasi untuk membuat data nama akan tetap ada sebelum dihapus, karena script diatas sudah memuat hal tersebut. Sehingga script hanya perlu disimpan dengan Ctrl+S.
5. Jalankan kode melalui web browser.



## Latihan 2:

1. Masuk ke teks editor (VS Code).
2. Mengetikkan scrip HTML untuk program web storage sesuai pada modul bagian Latihan 2.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device width, initial-scale=1">
6   <title>Web Storage 2</title>
7   <style type="text/css">
8     @import url('https://fonts.googleapis.com/css2?family=Josefin+Sans:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;1,100;1,200;1,300;1,400;1,500&family=1111+Pr1s1sd');
9
10    body {
11      font-family: 'Josefin Sans', sans-serif;
12    }
13
14    .label {
15      display: block;
16      text-align: center;
17    }
18
19    .tulisan {
20      text-align: center;
21      margin: auto;
22      border: 2px solid;
23      width: 50%;
24      position: absolute;
25      top: 50%;
26      left: 50%;
27      transform: translate(50%, 50%);
28    }
29
30    .switch {
31      position: relative;
32      display: inline-block;
33      width: 60px;
34      height: 34px;
35    }
36
37    .switch input {
38      opacity: 0;
39      width: 0;
40      height: 0;
41    }
42
43    .slider {
44      position: absolute;
45      cursor: pointer;
46      top: 0;
47      left: 0;
48      right: 0;
49      bottom: 0;
50      background-color: #808080;
51      -webkit-transition: .4s;
52      transition: .4s;
53    }
54
55    .slider:before {
56      position: absolute;
57      content: "";
58      height: 26px;
59      width: 26px;
60      left: 4px;
61      bottom: 4px;
62      background-color: white;
63      -webkit-transition: .4s;
64      transition: .4s;
65    }
66
67    input:checked + .slider {
68      background-color: #2196F3;
69    }
70
71    input:focus + .slider {
72      box-shadow: 0 0 1px #2196F3;
73    }
74
75    input:checked + .slider:before {
76      -webkit-transform: translateX(26px);
77      -ms-transform: translateX(26px);
78      transform: translateX(26px);
79    }
80  </style>
81 </head>
82 <body>
83   <div class="tulisan">
84     <h1>Universitas Sebelas Maret</h1>
85     <h2>Pendidikan Teknik Informatika dan Komputer</h2>
86   </div>
87   <div class="label">
88     <h3>Dark Mode</h3>
89     <div class="switch">
90       <input type="checkbox" onchange="ubah(event)">
91       <span class="slider"></span>
92     </div>
93   </div>
94 </body>
95 </html>
```

- Script juga memuat import font dari Google Fonts, yaitu Josefin Sans.
- Ketika halaman website dibuka, tampilan utama akan menampilkan label “Dark Mode” beserta sebuah tombol switch/checkbox berbentuk slider yang dibuat seperti saklar untuk mengubah mode.

- Dibawahnya ada teks utama yaitu “Universitas Sebelas Maret” dan “Pendidikan Teknik Informatika dan Komputer” didalam sebuah kotak/frame. Bagian ini berada di tengah layar karena sudah diposisikan menggunakan CSS.
  - Saat tombol dark mode ini ditekan (digeser ke kanan), event onchange="ubahTema(event)" akan terpanggil. Namun karena belum ada fungsi untuk mengubah tema, menekan tombol tersebut belum menyebabkan perubahan tema pada halaman, hanya perubahan warna tombol yang terjadi.
3. Menambahkan css untuk tampilan saat mode dark setelah body dan .tulisan.

```
16      body.dark {
17          background-color: #121212;
18          color: white;
19      }
20
```

- Warna background menjadi hitam saat mode dark.

```
39      body.dark .tulisan {
40          border-color: white;
41      }
42
```

- Border berwarna putih saat mode dark.
4. Menambahkan Javascript setelah class label untuk mengecek tema terakhir yang tersimpan di local storage sebelum keluar website.

```
102    <script>
103        const temaTersimpan = localStorage.getItem('tema');
104
105        if (temaTersimpan === 'dark') {
106            document.body.classList.add('dark');
107            document.getElementById('toggle').checked = true;
108        }

```

5. Selanjutnya menambahkan fungsi untuk mengubah tampilan saat dark mode on.

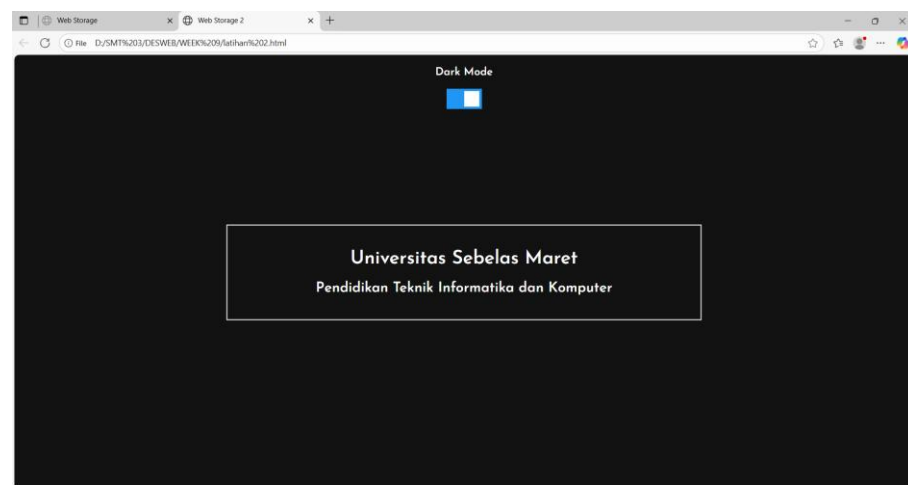
```
111    function ubahTema(e) {
112        if (e.target.checked) {
113            document.body.classList.add('dark');
114            localStorage.setItem('tema', 'dark');
115        } else {
116            document.body.classList.remove('dark');
117            localStorage.setItem('tema', 'light');
118        }
119    }

```

6. Menambahkan id="toggle" pada tag <input> agar otomatis on di mode gelap saat terakhir sebelum meninggalkan website berada di mode gelap.

```
94 <div class="label">
95   <h3>Dark Mode</h3>
96   <label class="switch">
97     <input id="toggle" type="checkbox" onchange="ubahTema(event)">
98     <span class="slider"></span>
99   </label>
100 </div>
101
```

7. Menyimpan scrip dengan Ctrl+S.
8. Menjalankan kode di web browser, coba aktifkan dark mode.



## G. KESIMPULAN

Dari praktikum Web Storage dapat disimpulkan bahwa penggunaan web storage memungkinkan sebuah website menyimpan data pada sisi pengguna tanpa perlu server. Melalui localStorage dan sessionStorage, data seperti nama pengguna maupun preferensi tampilan (dark mode) dapat dipertahankan meskipun halaman ditutup ataupun selama sesi berlangsung. Dengan demikian, pengalaman pengguna menjadi lebih personal karena tampilan terakhir sebelum meninggalkan website dapat dimunculkan kembali saat halaman dibuka lagi.