

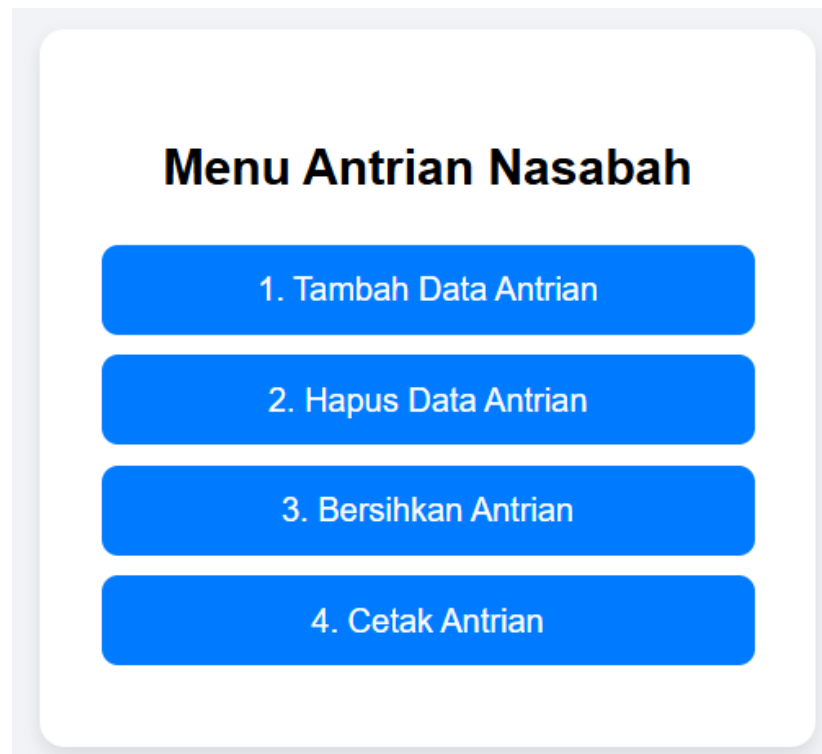
Laporan Praktikum

Tugas Modul 5

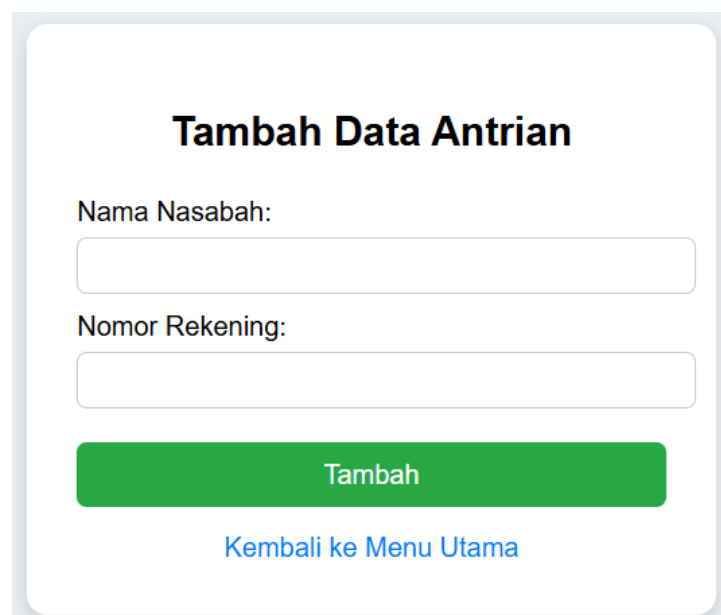
Nama : Muhammad Hadist Rifannan

Kelas : TI 1D

NPM : 240202107



Tampilan awal web , user dapat melakukan tambah data antrian , hapus data antrian , bersihkan antrian lalu mencetak antrian.



Tambah Data Antrian

Nama Nasabah:

Nomor Rekening:

Tambah

[Kembali ke Menu Utama](#)

Ini merupakan tampilan untuk menambah data antrian , user diminta memasukan nama dan nomor rekening yang akan ditambahkan ke dalam antrian.

Data ditambah
[Kembali](#)

Tampilan ketika data berhasil ditambah.

Daftar Antrian Nasabah

| No | Nama Nasabah | No Rekening |
|----|--------------|-------------|
| 1 | Hadist | 0450656 |
| 2 | Yasir | 009349 |
| 3 | Ali | 0934023 |

Kembali ke Menu Utama

Tampilan pada menu cetak antrian. dan disini semua queue / antrian akan ditampilkan nama nasabah dan no rekening nya masing masing.

1 data antrian berhasil dihapus.

Kembali ke Menu Utama

Tampilan pada menu hapus data antrian. ini akan menghapus antrian paling awal di queue. Yang berarti menghapus data no 1. Sesuai dengan prinsip queue yakni FIFO (First In First Out).

Daftar Antrian Nasabah

| No | Nama Nasabah | No Rekening |
|----|--------------|-------------|
| 1 | Hadist | 0450656 |
| 2 | Yasir | 009349 |
| 3 | Ali | 0934023 |

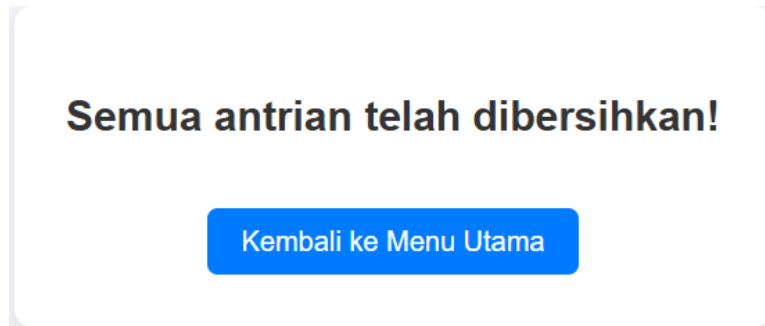
Kembali ke Menu Utama

Daftar Antrian Nasabah

| No | Nama Nasabah | No Rekening |
|----|--------------|-------------|
| 1 | Yasir | 009349 |
| 2 | Ali | 0934023 |

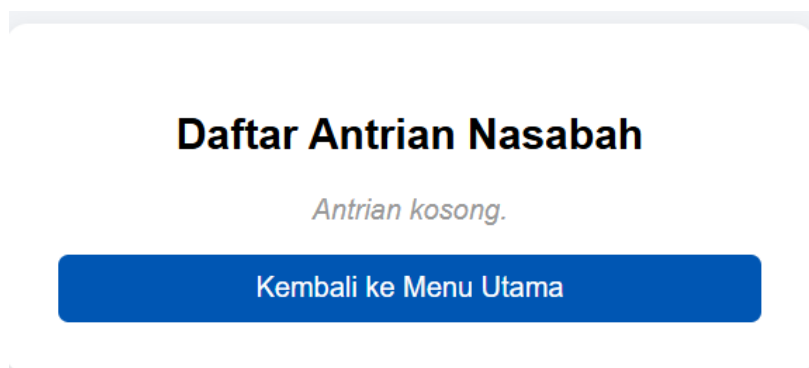
Kembali ke Menu Utama

Diatas adalah tampilan sebelum dan sesudah data antrian dihapus.



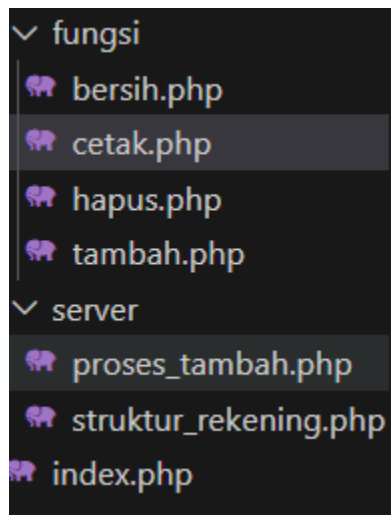
Ketika user menekan tombol bersihkan antrian , maka seluruh data antrian akan dihapus.

Jika data antrian kosong , maka tampilan cetak data antrian akan menjadi seperti ini



PENJELASAN SOURCE CODE :

Disini saya membuat beberapa script untuk meng handle beberapa fungsi diatas.yaitu fungsi tambah data , cetak data , hapus data , dan bersihkan data.



lalu menambahkan proses_tambah dan struktur_rekening sebagai handle server.proses_tambah untuk menangani fungsi penambahan data antrian.sementara struktur_rekening adalah berisi blueprint object / class untuk setiap elemen object data antrian.

Lalu ada index.php untuk halaman awal dari website tersebut

struktur_rekening.php berisi class yang menjadi blueprint struktur data antrian.

```
1 class Rekening{
2     public $nama;
3     public $nomorRekening;
4
5     function __construct($nama , $noRek){
6         $this->nama = $nama;
7         $this->nomorRekening = $noRek;
8     }
9 }
```

Berisi variable public nama , nomor rekening dan sebuah konstruktor kelas agar setiap pembuatan object selalu di assign ke dalam variable nama , dan nomor rekening.

```
1 include_once 'struktur_rekening.php';
2
3 session_start();
4
5 $data = isset($_SESSION['data']) ? $_SESSION['data'] : array();
6
7 if (isset($_POST['tambah'])){
8     array_push($data , new Rekening($_POST['nama'] , $_POST['rekening']));
9     $_SESSION['data'] = $data;
10    echo "Data ditambah<br>";
11    echo "<a href='../index.php'>Kembali</a>";
12 }
```

Disini script menyertakan struktur_rekening.php untuk mengambil object class yang telah kita buat.

Lalu ketika user menambah data antrian maka variable data akan di push dengan value object class menggunakan constructor dengan argument dari inputan nama dan nomor rekening.

Simpan data kedalam session agar data tersimpan di web.

```

1  include_once '../server/struktur_rekening.php';
2  session_start();
3
4  // Cek apakah antrian ada dan tidak kosong
5  if (isset($_SESSION['data']) && count($_SESSION['data']) > 0) {
6      // Hapus elemen pertama (antrian terdepan)
7      array_shift($_SESSION['data']);
8      $pesan = "1 data antrian berhasil dihapus.";
9  } else {
10     $pesan = "Antrian kosong. Tidak ada data yang dihapus.";
11 }

```

File **hapus.php** akan berisi seperti diatas. File ini akan menyertakan class Rekening. Kode ini akan mengecek terlebih dahulu jika data yang disimpan ada atau kosong , jika kosong maka akan muncul pesan “Antrian kosong. Tidak ada data yang dihapus.” Jika data tidak kosong maka akan menghapus data antrian pada urutan paling awal yakni index ke 0, dan memunculkan tulisan “1 data antrian berhasil dihapus”.

```

1  if (count($antrian) > 0): ?>
2      <table>
3          <thead>
4              <tr>
5                  <th>No</th>
6                  <th>Nama Nasabah</th>
7                  <th>No Rekening</th>
8              </tr>
9          </thead>
10         <tbody>
11             <?php for ($i = 0; $i < count($antrian); $i++){?>
12                 <tr>
13                     <td><?= $i + 1 ?></td>
14                     <td><?= $antrian[$i]->nama?></td>
15                     <td><?= $antrian[$i] ->nomorRekening?></td>
16                 </tr>
17             <?php } ?>
18         </tbody>
19     </table>

```

File **cetak.php** akan mencetak data antrian yang disimpan di session. Dengan mengecek terlebih dahulu apakah data tersebut ada. Jika ada maka lakukan perulangan untuk mengambil tiap data lalu ambil nama dan nomor rekening dari object class.

```
1  unset($_SESSION['data']);
```

File **bersih.php** akan menghapus seluruh data antrian yang tersimpan. Dengan meng unset / mereset session yang telah disimpan.

```
1  <form action="../server/proses_tambah.php" method="post">
2      <label>Nama Nasabah:</label>
3      <input type="text" name="nama" required>
4
5      <label>Nomor Rekening:</label>
6      <input type="text" name="rekening" required>
7
8      <button type="submit" name="tambah">Tambah</button>
9  </form>
```

File **tambah.php** berisi form yang akan dikirimkan ke proses_tambah.php , file ini mempunyai 2 inputan yaitu nama nasabah dan nomor rekening.