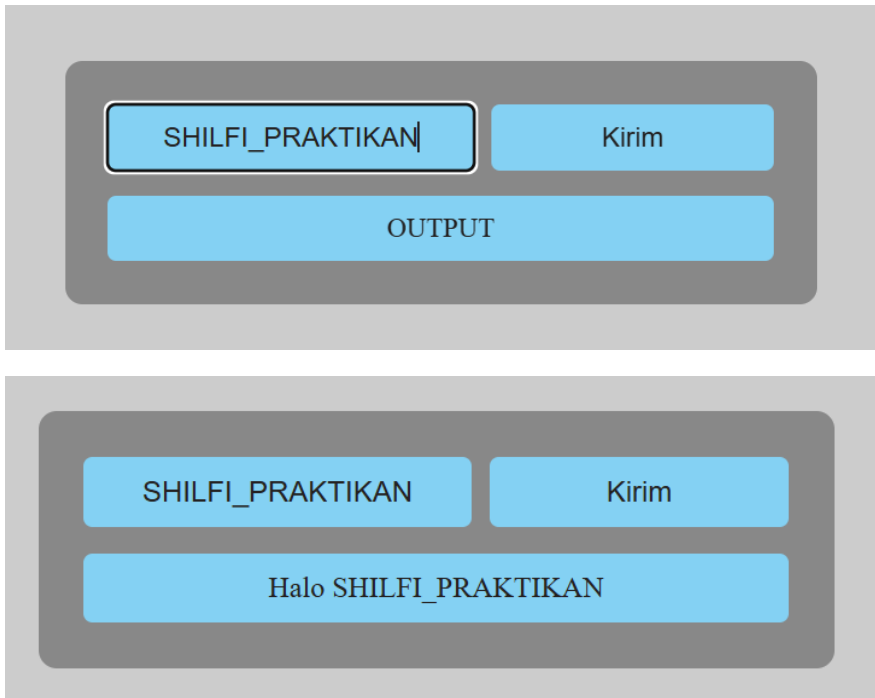


TP3_2311104002_Shilfi Habibah_SE0701

➔ Link github repository

https://github.com/shilfihabibah/KPL_Shilfi-Habibah_2311104002_SE07-01

➔ Screenshot hasil run (sebelum dan sesudah tombol submit ditekan)



➔ Penjelasan singkat dari kode implementasi yang dibuat (beserta screenshot dari potongan source code yang dijelaskan).

1. Server Side (Node.js + Express.js)

Kode di bawah ini mendefinisikan server menggunakan **Express.js**:

```
const express = require("express");
const bodyParser = require("body-parser");

const app = express();
const port = 3000;
```

- ❖ Mengimpor **Express** untuk membuat server.
- ❖ Mengimpor **body-parser** untuk menangani data dari form HTML.
- ❖ **app** adalah instance dari Express.
- ❖ **port** adalah port tempat server berjalan (**3000**).

2. Konfigurasi Middleware

```
app.set("view engine", "ejs");
app.use(bodyParser.urlencoded({ extended: true }));
```

- ❖ **app.set("view engine", "ejs")** → Mengatur EJS sebagai template engine untuk merender halaman HTML.
- ❖ **bodyParser.urlencoded({ extended: true })** → Memungkinkan pengambilan data dari form HTML melalui req.body.

3. Routing

Menampilkan Halaman Utama

```
app.get("/", (req, res) => {
  res.render("index", { output: "" });
});
```

- ❖ Ketika pengguna mengakses /, server akan merender file **index.ejs** dengan **output** kosong.

Menangani Form Submission

```
app.post("/submit", (req, res) => {
  const inputText = req.body.inputText;
  const outputText = `Halo ${inputText}`;
  res.render("index", { output: outputText });
});
```

- ❖ **req.body.inputText** mengambil nilai dari input form.
- ❖ **outputText** menyimpan teks hasil ("Halo [Nama]").
- ❖ Halaman di-render kembali dengan output yang diperbarui.

4. Menjalankan Server

```
app.listen(port, () => {
  console.log(`Server berjalan di http://localhost:${port}`);
});
```

5. Client Side (HTML, CSS, JS)

Form Input & Button

```
<div class="row">
  <input type="text" id="inputNama" placeholder="Masukkan Nama">
  <button onclick="tampilkanNama()">Kirim</button>
</div>
<p id="output">OUTPUT</p>
```

- ❖ Pengguna memasukkan nama ke dalam input.
- ❖ Tombol **Kirim** memicu fungsi **tampilkanNama()**.

Fungsi JavaScript untuk Menampilkan Nama

```
function tampilkanNama() {  
    let nama = document.getElementById("inputNama").value;  
    if (nama.trim() !== "") {  
        document.getElementById("output").innerText = "Halo " + nama;  
    } else {  
        document.getElementById("output").innerText = "OUTPUT";  
    }  
}
```

- ❖ Jika input tidak kosong, teks **"Halo [Nama]"** akan ditampilkan.
- ❖ Jika kosong, **OUTPUT** tetap default.