

Modul 4 : JavaScript (2)

4.1 Waktu Pelaksanaan Praktikum

Durasi kegiatan praktikum adalah **170 menit**, dengan rincian sebagai berikut.

- 15 menit untuk pengerjaan Tes Awal atau wawancara Tugas Pendahuluan
- 60 menit untuk penyampaian materi
- 45 menit untuk pengerjaan jurnal, tes akhir atau tugas
- 50 menit **pengayaan**

4.2 Tujuan

Setelah mengikuti praktikum ini, mahasiswa diharapkan dapat:

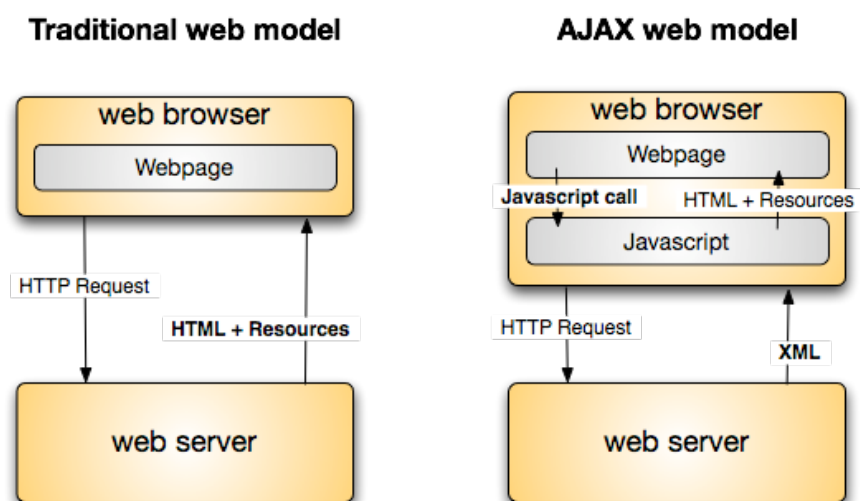
1. Memahami gambaran umum aplikasi web.
2. Memahami kerangka program PHP.
3. Memahami tipe data, variabel, dan operator pada PHP.
4. Memahami sintaks untuk mengatur alur program PHP.

4.3 Dasar Teori

a) AJAX

AJAX (Asynchronous JavaScript and XML) adalah cara berkomunikasi antara *browser (client)* dan *server* secara asinkron. Artinya, komunikasi terjadi pada *background process* pada *browser* sehingga *browser* dapat melakukan banyak komunikasi secara simultan. Pada saat terjadi komunikasi dengan AJAX, halaman pada *browser* tidak berubah. Komunikasi seperti ini terjadi saat kita, misalnya, mengunggah status ke Facebook dan mengirim komentar ke Instagram. Sehingga, interaksi antara pengguna dengan aplikasi dapat terjadi dengan lebih mudah dan cepat.

Tanpa AJAX, *browser* melakukan komunikasi dengan *server* secara sinkron. Saat dilakukan komunikasi, terjadi pergantian halaman pada *browser* dan hanya satu komunikasi saja yang dapat dilakukan pada satu waktu. Perbedaan kedua jenis komunikasi ini dapat dilihat pada Gambar 4.1.



Gambar 4.1 Perbedaan antara komunikasi dengan dan tanpa AJAX

4.4 Prosedur Praktikum

a) Latihan 1

Buat direktori “greet-ajax” di dalam direktori httdocs. Tulis kode di bawah ini ke *file* greet-ajax/index.html.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>AJAX</title>
5   </head>
6   <body>
7     <label>
8       Name: <input type="text" id="name">
9     </label>
10    <button id="greet">Greet</button>
11    <div id="message"></div>
12    <script src="ajax-greet.js"></script>
13  </body>
14 </html>
```

Tulis kode di bawah ini ke *file* greet-ajax/ajax-greet.js. Perhatikan baris 14 yang menggunakan karakter *backtick* (`), bukan tanda petik biasa (').

```
1 const button = document.querySelector('#greet');
2 const message = document.querySelector('#message');
3
4 button.addEventListener('click', () => {
5   const name = document.querySelector('#name').value;
6
7   message.textContent = 'Please wait...';
8
9   fetch('http://localhost/ajax/ajax.php', {
10     method: 'post',
11     headers: {
12       'Content-Type': 'application/x-www-form-urlencoded',
13     },
14     body: `name=${name}`
15   })
16   .then(res => res.text())
17   .then(data =>
18     message.textContent = data;
19   );
20 });
```

Tulis kode di bawah ini ke *file* greet-ajax/greet.php.

```
1 <?php
2
3 sleep(1);
4 $name = $_POST['name'];
5 echo "Hello $name!";
```

Buka <http://localhost/greet-ajax>, isi nama pada *text field*, lalu klik tombol Greet.

Pada *file* greet-ajax/ajax-greet.js, jelaskan tujuan dari *statement* `then` pada baris 6 s.d. 9.

b) Latihan 2

Buat direktori “chat” di dalam direktori htdocs. Tulis kode di bawah ini ke *file* chat/chat.js.

```
1  const chat = document.querySelector('#chat');
2  const message = document.querySelector('#message');
3  const baseUrl = 'http://localhost/chat';
4
5  function readChat() {
6    fetch(`${baseUrl}/chat-read.php`)
7      .then(res => res.text())
8      .then(data => {
9        chat.value = data;
10      });
11
12    setTimeout(readChat, 1000);
13  };
14
15  readChat();
16
17  message.addEventListener('keyup', e => {
18    if (e.keyCode === 13) {
19      fetch(`${baseUrl}/chat-write.php`, {
20        method: 'post',
21        headers: {
22          'Content-Type': 'application/x-www-form-urlencoded',
23        },
24        body: `text=${message.value}`
25      });
26
27      message.value = '';
28    }
29  });
```

Tulis kode di bawah ini ke file chat/index.html.

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>AJAX Chat App</title>
5    </head>
6    <body>
7      <textarea id="chat" rows="20" cols="31"></textarea><br>
8      <label>
9        Message: <input type="text" id="message">
10     </label>
11     <script src="chat.js"></script>
12   </body>
13 </html>
```

Tulis kode berikut ke *file* chat/chat-read.php.

```
1 <?php
2
3 echo file_get_contents('chat.txt');
```

Tulis kode berikut ke *file* chat/chat-write.php.

```
1 <?php
2
3 $message = $_POST['text'];
4
5 file_put_contents('chat.txt', "$message\n", FILE_APPEND);
```

Buka <http://localhost/chat> pada dua *browser window* yang berbeda lalu lakukan *chatting*.

Jelaskan tujuan dari statement `setTimeout(readChat, 1000);`

Pada file chat/chat.js, ubah argumen kedua pada *statement* `“setTimeout(readChat, 1000);”` dari 1000 menjadi 100 lalu lakukan *chatting*. Apakah yang berbeda? Jelaskan.

Pada file chat/chat-write.php, hapus argumen `“FILE_APPEND”` lalu lakukan *chatting*. Apakah yang berbeda? Apakah tujuan dari argumen `“FILE APPEND”`?
