

MATERI AJAR 7

Form

1.1. Materi Pembelajaran

Form HTML

1.2. Kemampuan Akhir Pembelajaran

Sub CPMK 7 : Mahasiswa mampu membuat form pada halaman HTML

1.3. Materi Ajar

a. Form

Form merupakan sebuah tag yang kita gunakan untuk mengkoleksi inputan dari user, konsep ini sama seperti konsep formulir di dunia nyata. Form digunakan untuk menginputkan data sebelum diproses oleh sistem. Form ini biasanya hanya berupa interface yang disediakan untuk mengumpulkan data dari user, dan akan diproses dengan bahasa pemrograman web seperti JavaScript atau PHP.

Tag-tag Pada Form

Berikut ini tag-tag yang digunakan dalam pembuatan form

Nama Tag	Keterangan/Fungsi
<form>	Tag digunakan untuk membuat sebuah form HTML untuk input pengguna.
<input>	Tag digunakan untuk membuat sebuah kontrol input.
<textarea>	Tag digunakan untuk membuat sebuah kontrol input multi baris (text area).
<button>	Tag digunakan untuk membuat sebuah tombol yang dapat diklik.
<select>	Tag digunakan untuk membuat sebuah daftar drop-down.
<optgroup>	Tag digunakan untuk membuat sebuah kelompok pilihan yang terkait dalam daftar drop-down.
<option>	Tag digunakan untuk membuat sebuah pilihan dalam daftar drop-down.
<label>	Tag digunakan untuk membuat sebuah label untuk sebuah elemen <input>.

Nama Tag	Keterangan/ Fungsi
<fieldset>	Tag digunakan untuk membuat grup unsur terkait dalam bentuk.
<legend>	Tag digunakan untuk merepresentasikan pengelompokan daftar input pada sebuah form.
<datalist>	Tag digunakan untuk membuat sebuah caption untuk sebuah elemen <fieldset, <figure>, atau <details>.
<keygen>	Tag digunakan untuk menentukan daftar pilihan yang ditetapkan untuk kontrol input, (Tag baru HTML5).
<output>	Tag digunakan untuk membuat key-pair generator kolom input, (Tag baru HTML5).

sebuah form dalam HTML harus berada di dalam tag <form>, yang diawali dengan tag pembuka <form> dan diakhiri dengan tag penutup </form>. Dalam pembuatan sebuah form, tag <form> juga membutuhkan beberapa atribut yang digunakan sebagai pendukung dalam pembuatan form. Berikut ini beberapa atribut yang digunakan dalam pembuatan form.

- **action**, untuk menentukan alamat dari aksi yang akan dilakukan saat data dikirim.
- **id**, digunakan untuk mengatur style css dan javascript di masing – masing kotak inputan form menggunakan id.
- **class**, digunakan untuk mengatur style css dan javascript di masing – masing kotak inputan form menggunakan class.
- **method**, menentukan metode yang digunakan oleh browser untuk pengiriman data form. Ada dua nilai method yang digunakan.
 - **post**: berkaitan dengan HTTP POST method. Metode ini digunakan untuk mengirim data form yang disisipkan pada body form dan dikirim pada server.
 - **get**: berkaitan dengan HTTP GET method. Metode ini digunakan untuk mengirim data form yang disisipkan

pada URI **action** attribute dengan tanda tanya (?) sebagai pemisah kemudian dikirim ke server. karakter yang digunakan adalah ASCII.

- **name**, merupakan nama dari field yang akan menjadi kunci dan variabel di dalam program.
- **placeholder**, untuk menampilkan teks sementara (placeholder)/ Panduan pengisian form
- **type**, merupakan tipe dari field.
- **value**, untuk memberikan nilai default pada field.

Tag <input>

Tag <input> adalah tag yang digunakan dalam form pengisian dan memiliki banyak bentuk, mulai dari isian text biasa, text password, checkbox, radio, sampai dengan tombol submit, semuanya diatur pada tag <input>.

Bentuk-bentuk dari keluarga tag input ini dibedakan berdasarkan atribut type:

- **<input type="text" />** atau bisa juga **<input />** adalah textbox inputan biasa yang menerima input berupa text, contohnya digunakan untuk inputan nama, username, dan inputan yang berupa teks pendek. Input type text ini juga bisa memiliki atribut value yang bisa diisi nilai tampilan awal dari text.
- **<input type="password" />** tampilan inputan teks pada typer password sama seperti type text, namun teks yang diinput tidak akan terlihat, tetapi akan berupa bintang atau titik. Textbox ini biasanya digunakan pada inputan password.
- **<input type="submit" />** tipe inputan ini akan menampilkan tombol (button) untuk memproses data inputan form.
- **<input type="checkbox" />**

Contoh Program form1.html

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Mengenal Form </title>
</body>
<form action="form1.html" method="get">
  <p>Silahkan isi data berikut :</p>
  Nama      : <input type="text" name="nama"> <br />
  Jenis Kelamin :
    <input type="radio" name="jenkel" value="laki-laki">Laki-laki
    <input type="radio" name="jenkel" value="perempuan">Perempuan
    <br />
  Username   : <input type="text" name="username"> <br />
  Password   : <input type="password" name="password"> <br />
  Course yang diikuti:
  <input type="checkbox" name="html">HTML & CSS
  <input type="checkbox" name="javascript">Javascript
  <input type="checkbox" name="react">React JS
  <input type="checkbox" name="node">Node Js
  <br /><br />
  <button type="submit" name="submit" value="simpan">Simpan</button>
</form>
</body>
</html>

```

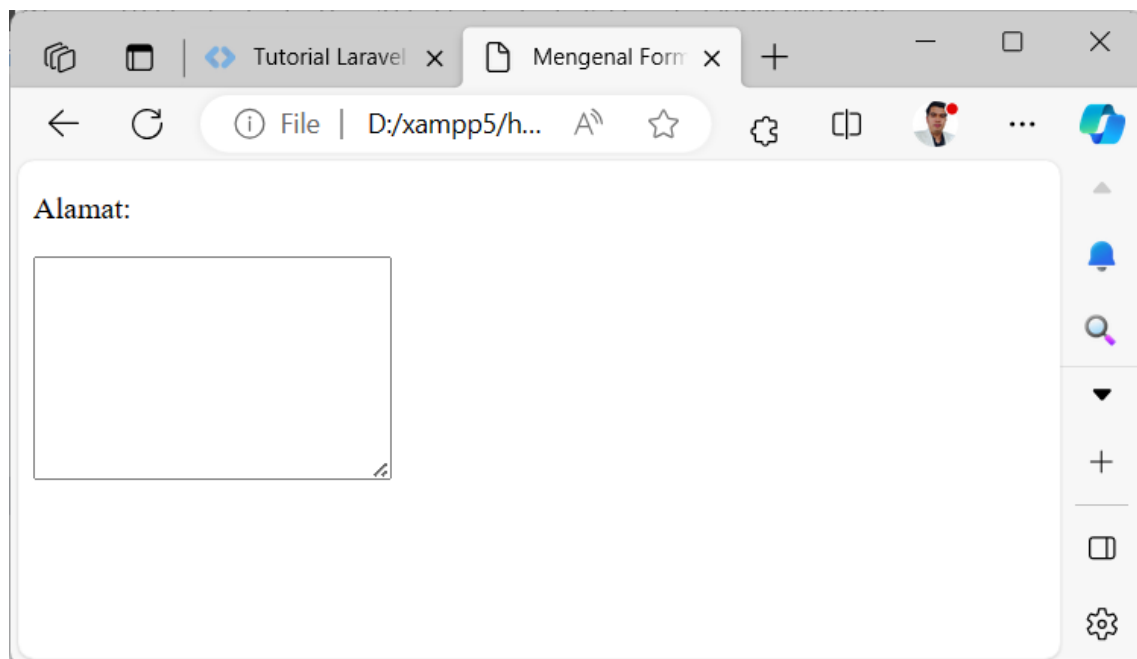
Hasil :

Tag <textarea>

Tag <textarea> ini sama dengan input Type Text, namun ukurannya lebih besar dari form text biasa dan dapat berisi banyak baris. Panjang dan banyak baris untuk text area bisa diatur melalui atribut Rows dan Cols, atau melalui CSS.

Contoh Program form2.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Mengenal </title>
  <style>
    #alamat{
      font-size: 10px;
      color: red;
      font-style: normal;
    }
  </style>
</head>
<body>
<form action="latihan.html" method="get">
  <p>Alamat:</p>
  <textarea name="alamat" id="alamat" cols="30"
rows="10"></textarea>
</form>
</body>
</html>
```

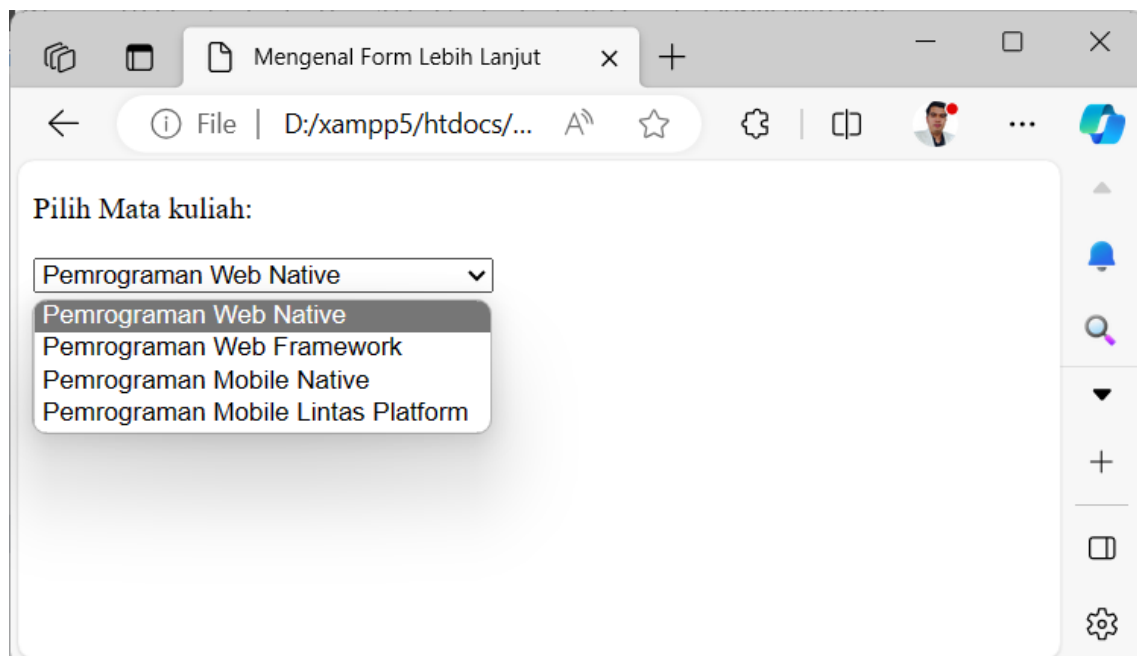


Tag <select>

Tag <select> ini digunakan untuk memberikan pilihan pada user dari data yang sudah tersedia. Dalam tag <select> terdapat tag <option> yang digunakan untuk membuat pilihan data.

Contoh Program form3.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Mengenal Form Lebih Lanjut</title>
</head>
<body>
<form action="form3.html" method="get">
  <p>Pilih Mata kuliah:</p>
  <select name="select">
    <option value="web1">Pemrograman Web Native</option>
    <option value="web2">Pemrograman Web Framework</option>
    <option value="Mobile1">Pemrograman Mobile Native</option>
    <option value="Mobile2">Pemrograman Mobile Lintas
Platform</option>
  </select>
</form>
</body>
</html>
```



Contoh Program form4.html

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Form Biodata</title>
</body>
<form action="form4.html" method="POST">
  <fieldset>
    <legend><h1>Formulir Peserta </h1> </legend>
    <p>
      <label>Nama :</label>
      <input type="text" name="nama" placeholder="Masukan nama
lengkap">
    </p>
    <p>
      <label>Username :</label>
      <input type="text" name="username" placeholder="Masukan
username">
    </p>
    <p>
      <label>Password :</label>
      <input type="text" name="password" placeholder="Masukan
password">
    </p>
    <p>
      <label>Jenis Kelamin :</label>
      <input type="radio" name="jenkel" value="laki-laki">Laki-laki
      <input type="radio" name="jenkel" value="perempuan">Perempuan
    </p>
    <p>
      <label>Tanggal Lahir :</label>
      <input type="date" name="tanggal">
    </p>
    <p>
      <label>Agama :</label>
      <select name="agama">
        <option value="budha">Budha</option>
        <option value="hindu">Hindu</option>
        <option value="islam">Islam</option>
        <option value="katholik">Kristen Katholik</option>
        <option value="protestan">Kristen Protestan</option>
      </select>
    </p>
    <p>
      <label>Email :</label>

```

```

        <input type="email" name="email" placeholder="Masukkan email">
    </p>
    <p>
        <label>Pilih Course:</label>
        <input type="checkbox" name="web native">Web Native
        <input type="checkbox" name="web framework">Web Framework
        <input type="checkbox" name="mobile native">Mobile Native
        <input type="checkbox" name="web lintas platform">Mobile
Lintas Platform
    </p>
    <p>
        <p>Alamat :</p>
        <textarea name="alamat"cols="50" rows="10"></textarea>
    </p>
    <p>
        <input type="submit" name="submit" value="Daftar">
    </p>
</fieldset>
</form>
</body>
</html>

```

Formulir Peserta

Nama :

Username :

Password :

Jenis Kelamin : Laki-laki Perempuan

Tanggal Lahir :

Agama :

Email :

Pilih Course : Web Native Web Framework Mobile Native Mobile Lintas Platform

Alamat :

Penanganan Form dengan Javascript

Untuk mendapatkan data formulir ini melalui JavaScript, harus melampirkan atribut tertentu ke kolom masukan formulir dan formulir itu sendiri. Atribut ini bisa berupa id, a class, atau bahkan dengan name. Ini akan membantu mendapatkan data dalam JavaScript menggunakan metode dokumennya.

Misalnya, jika menggunakan id atribut pada kolom masukan, Anda dapat mengakses data kolom masukan dan nilai lainnya menggunakan metode dokumen `getElementById('idName')`:

```
// HTML
<input type="text" id="username" class="form-control" placeholder="Enter your
Username...">

// JS
let myUsername = document.getElementById('username');
console.log(myUsername);
```

Jika menggunakan name atribut, akan menggunakan `getElementsByName(name)`. Hal ini mirip dengan cara kerja atribut id karena ia juga mengembalikan array yang dapat Anda ulangi atau akses dengan nomor indeksinya:

```
// HTML
<input type="text" name="username" class="form-control" placeholder="Enter your
Username...">

// JS
let myUsername = document.getElementsByName('username');
console.log(myUsername[0]);
```

Cara Mengirim Formulir dengan JavaScript

Langkah pertama adalah melampirkan atribut pilihan Anda ke formulir, yang bisa Anda gunakan untuk melacak kapan formulir dikirimkan. Ini bisa berupa id, class atau name atribut.

```

<form action="" id="loginForm">
  <h1>Login</h1>
  <input type="text" id="username" class="form-control" placeholder="Enter your
Username...">
  <input type="password" id="password" class="form-control" placeholder="Enter your
Password...">
  <button type="submit">Submit</button>
</form>

```

sekarang dapat menangani pengiriman formulir dengan JavaScript. Pertama-tama Anda mendapatkan formulir dengan atribut pilihan, yang bisa berupa id, dan menyimpannya dalam variabel:

```
let loginForm = document.getElementById("loginForm");
```

Kemudian dapat melampirkan `addEventListener` ke variabel form dan mendengarkan acara pengiriman. Pendengar peristiwa ini memungkinkan Anda melampirkan fungsi panggilan balik yang dipicu setelah formulir dikirimkan:

```
loginForm.addEventListener("submit", (e) => {
  e.preventDefault();

  // handle submit
});
```

sekarang bisa mendapatkan data formulir dan menangani operasi apa pun yang Anda inginkan. Untuk artikel ini, pertama-tama mari kita validasi data dengan memeriksa apakah inputnya kosong sebelum melakukan operasi apa pun:

```
loginForm.addEventListener("submit", (e) => {
  e.preventDefault();

  let username = document.getElementById("username");
  let password = document.getElementById("password");

```

```

if (username.value == "" || password.value == "") {
    // throw error
} else {
    // perform operation with form input
}
});

```

Contoh Program formjs.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Mengenal Form Nusantech Academy</title>
    <style>
        #alamat{
            font-size: 10px;
            color: red;
            font-style: normal;
        }
    </style>
</body>
    <form action="" id="loginForm">
        <h1>Login</h1>
        <input type="text" id="username" class="form-control"
placeholder="Enter your Username...">
        <input type="password" id="password" class="form-control"
placeholder="Enter your Password...">
        <button type="submit">Submit</button>
    </form>
</body>
<script type="text/javascript">
    let loginForm = document.getElementById("loginForm");

    loginForm.addEventListener("submit", (e) => {
        e.preventDefault();

        let username = document.getElementById("username");
        let password = document.getElementById("password");

        if (username.value == "" || password.value == "") {
            alert("Ensure you input a value in both fields!");
        } else {
            // perform operation with form input
            alert("This form has been successfully submitted!");
        }
    });

```

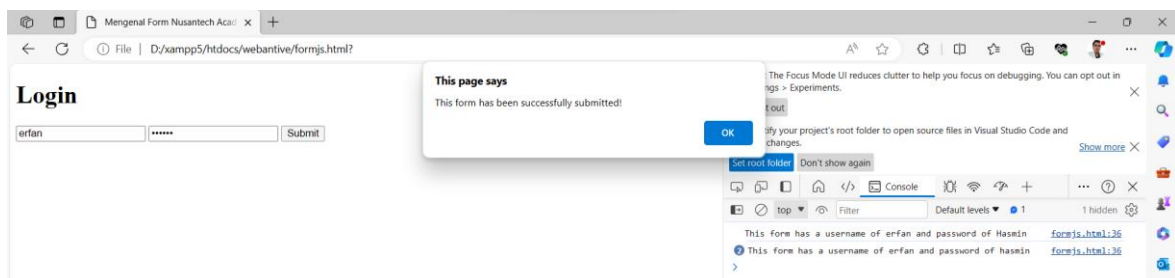
```

        console.log(
            `This form has a username of ${username.value} and password
of ${password.value}`
        );

        username.value = "";
        password.value = "";
    }
});

</script>
</html>

```



Refrence :

<https://www.freecodecamp.org/news/how-to-submit-a-form-with-javascript/>