

Nama : Akmal Muhamad Firdaus

NIM : 1301204188

Praktikum ABP Modul 5

1. Pengenalan PHP

Web server adalah perangkat lunak yang berjalan di atas sistem operasi dan bertanggung jawab untuk menangani permintaan HTTP dari klien (seperti browser web) dan memberikan respon sesuai dengan permintaan tersebut. Web server biasanya digunakan untuk menyajikan halaman web, gambar, atau file lainnya ke pengguna.

Server-side scripting adalah teknologi yang memungkinkan pengembang web untuk mengeksekusi kode di server dan menghasilkan respon dinamis berdasarkan permintaan pengguna. Kode yang dieksekusi di sisi server biasanya digunakan untuk memproses data yang dikirim oleh pengguna, mengambil data dari basis data, menghasilkan halaman web dinamis, atau melakukan tugas lainnya yang terkait dengan logika bisnis aplikasi web.

Contoh server-side scripting yang umum digunakan adalah PHP, Python, Ruby, dan JavaScript (Node.js). Teknologi ini memungkinkan pengembang untuk membuat aplikasi web dinamis dan interaktif dengan lebih mudah dan efisien, serta memungkinkan aplikasi web untuk menangani jumlah data yang lebih besar dan menghasilkan respon yang lebih cepat daripada client-side scripting.

2. Pengenalan PHP

Merupakan singkatan rekursif dari PHP : Hypertext Preprocessor. Pertama kali diciptakan oleh Rasmus Lerdorf pada tahun 1994. PHP sendiri harus ditulis diantara tag :

- <? dan ?>
- <?php dan ?>
- <script language = "php"> dan </script>
- <% dan %>

Setiap satu statement (perintah) biasanya diakhiri dengan titik-koma (;). PHP juga case sensitive untuk nama identifier yang dibuat oleh user sedangkan identifier bawaan dari PHP tidak case sensitive.

2.1. Variabel

Variabel digunakan untuk menyimpan sebuah value (nilai), data atau informasi. Nama variabel pada PHP diawali dengan tanda \$. Panjang dari suatu variabel tidak terbatas dan variabel tidak perlu dideklarasikan terlebih dahulu sebelumnya. Setelah tanda \$, dapat diawali dengan huruf atau under-score (_). Karakter berikutnya bisa terdiri dari huruf, angka dan atau karakter tertentu yang diperbolehkan (karakter ASCII dari 127 – 255). Variabel pada PHP bersifat case sensitive artinya besar kecilnya suatu karakter berpengaruh pada variabel tersebut. Suatu karakter pada PHP tidak boleh mengandung spasi.

```
<?php
//Variabel

$nim = "1301204188";
$nama = "Akmal Muhamad Firdaus";
//Tampilkan Variabel
echo "NIM : ".$nim."<br>". "Nama : ".$nama."<br><br>";
```

Keluaran:

NIM : 1301204188

Nama : Akmal Muhamad Firdaus

Pada PHP, tipe data dari suatu variabel tidak didefinisikan langsung oleh programmer, akan tetapi secara otomatis akan ditentukan oleh interpreter PHP. Namun demikian, PHP mendukung 8 (delapan) buah tipe data primitif, yaitu :

1. Boolean
2. Integer
3. Float
4. String
5. Array
6. Object
7. Resource
8. NULL

2.2. Konstanta

Konstanta merupakan variabel konstan yang nilainya tidak berubah-ubah. Untuk mendefinisikan konstanta pada PHP, dapat menggunakan fungsi define() yang telah tersedia pada PHP.

```
define("NAMA", "Akmal Muhamad Firdaus");  
define("NIM", "1301204188");  
echo "NIM : ".NIM."<br>". "Nama : ".NAMA;
```

NIM : 1301204188

Nama : Akmal Muhamad Firdaus

2.3. Kondisi & Control Flow

1. If Else

If-else adalah struktur kontrol yang umum digunakan dalam pemrograman untuk memeriksa kondisi tertentu dan melakukan tindakan yang berbeda tergantung pada apakah kondisi tersebut benar atau salah.

```
$nilai = 70;  
if ($nilai >= 60) {  
    echo "Nilai Anda $nilai, Anda Lulus";  
} else {  
    echo "Nilai Anda $nilai, Anda Tidak Lulus";  
}
```

Dalam contoh di atas, kita menggunakan struktur if-else untuk mengevaluasi apakah \$nilai lebih dari sama dengan 60 atau tidak. Jika \$nilai lebih dari sama dengan, maka akan mencetak "Nilai Anda \$nilai, Anda Lulus" dan jika tidak maka akan mencetak "Nilai Anda \$nilai, Anda Tidak Lulus"

Nilai Anda 70, Anda Lulus

Pada contoh diatas, \$nilai adalah 70, Maka sesuai deskripsi if-else yang disebutkan diatas benar bahwa yang tertampil adalah "Nilai Anda 70, Anda Lulus".

2. Looping

Looping adalah konstruksi program yang memungkinkan untuk mengeksekusi sebuah blok kode secara berulang-ulang selama kondisi tertentu terpenuhi. Looping sangat berguna untuk mengeksekusi tugas yang sama berulang-ulang atau untuk memproses kumpulan data seperti array atau database. Berikut adalah salah satu contoh looping, yaitu menggunakan For Loop:

```
//belajar looping
echo "<br><br><br>";
for ($i=1; $i <= 10; $i++) {
    echo "Aku akmal, clonangan ke-$i <br>";
}
```

```
Aku akmal, clonangan ke-1
Aku akmal, clonangan ke-2
Aku akmal, clonangan ke-3
Aku akmal, clonangan ke-4
Aku akmal, clonangan ke-5
Aku akmal, clonangan ke-6
Aku akmal, clonangan ke-7
Aku akmal, clonangan ke-8
Aku akmal, clonangan ke-9
Aku akmal, clonangan ke-10
```

2.4. Array

Array adalah sebuah struktur data pada pemrograman yang digunakan untuk menyimpan kumpulan data atau nilai dalam satu variabel tunggal. Dalam PHP, array dapat berisi tipe data apapun seperti angka, string, boolean, objek, atau bahkan array lain.

```
//belajar array
echo "<br><br><br>";
$nama = array("Akmal", "Muhamad", "Firdaus");
echo $nama[0]. "<br>";
echo $nama[1]. "<br>";
echo $nama[2]. "<br>";
```

```
Akmal
Muhamad
Firdaus
```

3. Eksplorasi PHP Lebih Dalam

Note: Codingan berikut adalah lanjutan codingan (1 codebase) pada codingan sebelumnya

Studi Case: Login menggunakan AJAX (POST) jQuery

1. Form Structure

```
<form id="login-form" method="post">
  <div class="col-md-12">
    <div class="mb-3">
      <label for="username" class="form-label">Username</label>
      <input type="text" class="form-control" id="username" name="username" placeholder="Masukan username">
    </div>

    <div class="mb-3">
      <label for="password" class="form-label">Password</label>
      <input type="password" class="form-control" id="password" name="password" placeholder="Masukan password">
    </div>

    <button class="btn btn-teal mt-5" type="submit" id="btn-login">Login</button>
  </div>
</form>
```

Use Uakmal P:1301204188 for success response

Username

Masukan username

Password

Masukan password

Login

2. jQuery code

```
$('#login-form').on('submit', function(event) {
    event.preventDefault();
    console.log($(this));
    $.ajax({
        url: 'login.php',
        type: 'POST',
        dataType: 'json',
        data: $(this).serialize(),
        beforeSend: function() {
            $('#login-btn').prop('disabled', true);
        },
        success: function(response) {
            if (response.status == 'success') {
                $('#login-message').html('<div class="alert alert-success" role="alert">Login berhasil</div>');
            } else {
                $('#login-message').html('<div class="alert alert-danger" role="alert">' + response.message + '</div>');
            }
        },
        error: function(jqXHR, textStatus, errorThrown) {
            var errorData = jqXHR.responseJSON;
            $('#login-message').html('<div class="alert alert-danger" role="alert">' + errorData.message + '</div>');
        },
        complete: function() {
            $('#login-btn').prop('disabled', false);
        }
    });
});
```

AJAX (Asynchronous JavaScript and XML) adalah teknik pengembangan web yang memungkinkan kita untuk melakukan permintaan data ke server dan meng-update halaman web tanpa perlu melakukan refresh halaman secara keseluruhan. Dengan teknik ini, kita dapat membuat aplikasi web yang lebih interaktif, responsif dan cepat. Penggunaan AJAX biasanya melibatkan pemrograman client-side dengan menggunakan JavaScript, sedangkan server-side menggunakan bahasa pemrograman seperti PHP.

Untuk memudahkan penggunaan AJAX, kita bisa menggunakan library JavaScript seperti jQuery. jQuery menyediakan fungsi AJAX yang siap pakai dan mudah digunakan. Fungsi AJAX pada jQuery yang paling sering digunakan adalah \$.ajax(). Fungsi ini memiliki banyak opsi yang dapat dikonfigurasi untuk mengatur cara permintaan AJAX dilakukan dan cara respons yang diterima. Selain itu, jQuery juga menyediakan beberapa fungsi pendekatan seperti get dan post yang memudahkan kita dalam melakukan permintaan HTTP GET atau POST ke server. Dengan menggunakan jQuery, kita dapat mengembangkan aplikasi web yang interaktif dengan lebih cepat dan mudah.

Tracing dan penjelasan kode :

1. `$('#login-form').on('submit', function(event) {`
Kode ini menetapkan event handler untuk elemen HTML dengan id #login-form ketika form login dikirimkan / dilakukn proses submit.
2. `event.preventDefault();`
Secara default, form akan melakukan action (Contohnya post / get) terhadap url yang dituju, tetapi dengan adanya preventDefault akan menghentikan perilaku default dari formulir login untuk melakukan aksi refresh atau redirect tersebut.
3. `$.ajax({`
Untuk memulai permintaan ajax ke url tertentu, lalu disertai beberapa atribut lainnya sebagai pengaturan tambahan
4. `success: function(response) {`
Kode blok ini akan menangani respon yang diterima dari server jika permintaan ajax berhasil (response code 200-299) dilakukan
5. `error: function(jqXHR, textStatus, errorThrown) {`
Kode blok ini akan menangani respon yang diterima dari server jika terjadi error saat melakukan permintaan ajax.
6. `complete: function() { $('#login-btn').prop('disabled', false); }`
Kode ini mengaktifkan tombol login kembali setelah permintaan ajax selesai dilakukan.

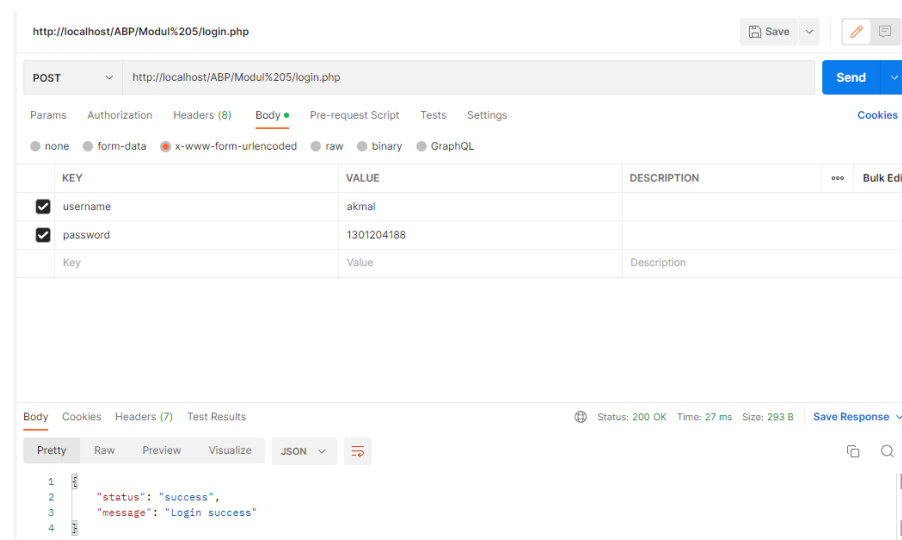
3. PHP Code

```
<?php
if(isset($_POST["username"]) && isset($_POST["password"])) && ($_POST["username"] != "" && $_POST["password"] != ""){
    $username = $_POST['username'];
    $password = $_POST['password'];
    if($username == 'akmal' && $password == '1301204188'){
        http_response_code(200);
        $response = array(
            'status' => 'success',
            'message' => 'Login success',
        );
    } else {
        http_response_code(401);
        $response = array(
            'status' => 'error',
            'message' => 'Username or password is wrong!',
        );
    }
} else {
    http_response_code(400);
    $response = array(
        'status' => 'error',
        'message' => 'Username or password is required!',
    );
}
header("Content-Type: application/json");
echo json_encode($response);
```

Pada kode diatas mengecek apakah variabel `$_POST["username"]` dan `$_POST["password"]` ada dan tidak kosong. Jika ada, maka kode akan melanjutkan untuk memproses login. Lalu pada masing masing blok if-else statement terdapat `http_response_code`. `http_response_code` adalah sebuah fungsi built-in di dalam PHP yang digunakan untuk mengatur kode status HTTP pada sebuah response. Kode status HTTP digunakan untuk memberikan informasi tentang bagaimana sebuah permintaan berhasil atau gagal dilakukan. Contoh kode status HTTP yang digunakan pada kode diatas adalah:

1. 200 : Ok
2. 400 : Bad Request
3. 401 : Unauthorized

Lalu setelah keluar dari blok if-else statement (2 baris terakhir) adalah untuk menampilkan response sesuai dengan logic if, sebagai contoh jika user memasukan username = akmal dan password = 1301204188, maka akan mengembalikan sebuah response berbentuk json seperti gambar berikut:



JSON (JavaScript Object Notation) dan XML (eXtensible Markup Language) keduanya adalah format data yang digunakan untuk pertukaran data antar aplikasi pada web. Namun, JSON lebih banyak digunakan dibandingkan XML untuk beberapa alasan berikut:

1. Lebih mudah dibaca dan ditulis: JSON menggunakan format notasi objek dan array yang sudah dikenal dalam pemrograman, sehingga lebih mudah dibaca dan ditulis oleh manusia maupun mesin.
2. Ukuran file yang lebih kecil: JSON memiliki ukuran file yang lebih kecil dibandingkan dengan XML. Hal ini karena JSON menggunakan sintaksis yang lebih sederhana dan tidak memerlukan banyak tag seperti pada XML.
3. Parsing yang lebih cepat: JSON dapat diparsing lebih cepat daripada XML. Hal ini karena struktur data JSON lebih sederhana dan lebih mudah diproses oleh komputer.
4. Dukungan penuh oleh bahasa pemrograman modern: JSON didukung oleh hampir semua bahasa pemrograman modern, termasuk JavaScript, PHP, Python, C#, dll. Sementara XML memerlukan pengolahan yang lebih rumit dan membutuhkan library tambahan untuk diproses oleh bahasa pemrograman tertentu.

4. Output

Use U:akmal P:1301204188 for success response

Username or password is required!

Username

Masukan username

Password

Masukan password

Login

Use U:akmal P:1301204188 for success response

Username or password is wrong!

Username

unknown

Password

Login

Use U:akmal P:1301204188 for success response

Login berhasil

Username

akmal

Password

Login

Studi Case 2: Get users data (Dummy) menggunakan AJAX (GET) jQuery dan menerapkannya pada table

Selain menggunakan http POST (pada studicase 1), terdapat method lain yang biasanya paling sering digunakan, yaitu contohnya menggunakan http GET.

1. Form Structure

```
<section class="bg-white pt-10 py-10">
  <div class="container px-5">
    <div class="text-center mb-5">
      <h2>Example use table get from server using ajax</h2>
      <button class="btn btn-teal mt-4" class="btn btn-primary" id="get-users">
        Click to get data
      </button>
    </div>
    <div class="row gx-5 z-1">
      <div class="col-md-12">
        <table class="table" id="users-tbl">
          <thead>
            <tr>
              <th scope="col">#</th>
              <th scope="col">Nama</th>
              <th scope="col">NIM</th>
              <th scope="col">Alamat</th>
            </tr>
          </thead>
          <tbody>
            <tbody>
          </tbody>
        </table>
      </div>
    </div>
  </div>
</section>
```

Example use table get from server using ajax

Click to get data

#	Nama	NIM	Alamat
---	------	-----	--------

Struktur tabel tersebut kosong, tetapi setelah user menekan Click to get data (#get-users), maka sistem akan melakukan ajax get terhadap server.

2. Javascript Code

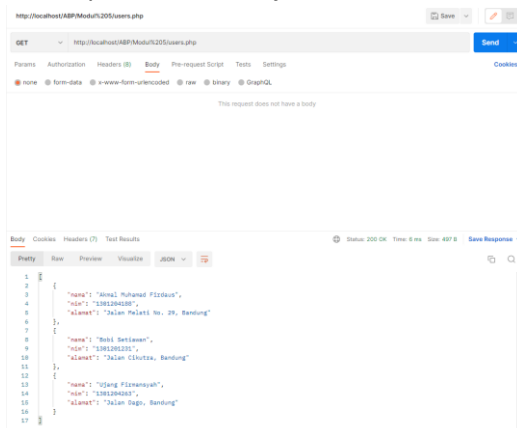
```
$("#get-users").click(function(e) {
  $.ajax({
    url: 'users.php',
    dataType: 'json',
    success: function(data) {
      let no = 1;
      $.each(data, function(index, user) {
        var row = '<tr>';
        row += '<th scope="row">' + no + '</th>';
        row += '<td>' + user.nama + '</td>';
        row += '<td>' + user.nim + '</td>';
        row += '<td>' + user.alamat + '</td>';
        row += '</tr>';
        $('#users-tbl tbody').append(row);
        no++;
      });
    },
    error: function(jqXHR, textStatus, errorThrown) {
      console.log('Error: ' + textStatus);
    }
  });
});
```

Sama seperti penjelasan ajax diatas, disini menggunakan ajax menggunakan metode yang berbeda, yaitu menggunakan get. Lalu setelah mendapat response dari server, maka akan dilakukan foreach untuk dilakukan DOM terhadap tabel (#users-tbl)

3. PHP Code

```
<?php
$users = [
    [
        "nama" => "Akmal Muhamad Firdaus",
        "nim" => "1301204188",
        "alamat" => "Jalan Melati No. 29, Bandung"
    ],
    [
        "nama" => "Bobi Setiawan",
        "nim" => "1301201231",
        "alamat" => "Jalan Cikutra, Bandung"
    ],
    [
        "nama" => "Ujang Firmansyah",
        "nim" => "1301204263",
        "alamat" => "Jalan Dago, Bandung"
    ]
];
header('Content-Type: application/json');
echo json_encode($users);
```

Pada kode php diatas, tidak ada logic apapun, melainkan hanya menampilkan data dummy dalam format json.



4. Ouput Before

Example use table get from server using ajax

Click to get data

#	Nama	NIM	Alamat
---	------	-----	--------

After

Example use table get from server using ajax

Click to get data

#	Nama	NIM	Alamat
1	Akmal Muhamad Firdaus	1301204188	Jalan Melati No. 29, Bandung
2	Bobi Setiawan	1301201231	Jalan Cikutra, Bandung
3	Ujang Firmansyah	1301204263	Jalan Dago, Bandung

Full Source Code : <https://github.com/codernewbie04/CII3H4-ABP-Praktikum>