

Nama : La Ode Muhammad Gazali
NIM : 222212696
Kelas : 2KS2

MODUL 11 PRAKTIKUM PEMROGRAMAN BERBASIS WEB

(Pengenalan AJAX dan JSON)

Penugasan

Kerjakan sesuai dengan yang dijelaskan pada bagian Kegiatan Praktikum. Simpan tangkapan layar hasil pekerjaan Anda (source code dan halaman web) dalam file docx. Simpan ulang file tersebut dalam format pdf, dan beri nama dengan format <<nim>>_modul11.pdf, contoh: 192191234_modul11.pdf. Kumpulkan file tersebut sebagai responsi melalui Google Classroom.

Penyelesaian

- Membuat AJAX Sederhana

kita akan coba membuat aplikasi Ajax sederhana menggunakan PHP sebagai aplikasi backend-nya. Kita akan membuat satu halaman web yang akan mengambil waktu server dan menampilkannya tanpa memuat ulang seluruh halaman. Agar menyesuaikan dengan waktu di Jakarta (GMT+7) maka perlu diatur default time zone sebelum memanggil fungsi date()

telltime.php

```
<?php
date_default_timezone_set('Asia/Jakarta');
echo date('H:i:s');
?>
```

20:52:47

Selanjutnya akan dibuat sebuah file untuk melihat kotak berisi tampilan waktu server yang telah mengakomodasi fungsi autoUpdateTime() agar dapat menampilkan waktu secara otomatis setiap detik.

Showtime.html

```
<!DOCTYPE html>
<html lang="en-ID">
  <head>
    <title>Ajax Demonstration</title>
    <meta charset="utf-8" />
  </head>
  <style>
```

```

.displaybox {
    display: table-cell;
    vertical-align: middle;
    width: 250px;
    height: 40px;
    background-color: #ffffff;
    border: 2px solid #000000;
    padding: 10px;
    font: 1.5em normal verdana, helvetica, arial, sans-serif;
}
</style>
<script>
    var ajaxRequest;

    function ajaxResponse() {
        if (ajaxRequest.readyState != 4) {
            return;
        } else {
            if (ajaxRequest.status == 200) {
                var plholder = document.getElementById("showtime");
                plholder.innerHTML = ajaxRequest.responseText;
            } else {
                alert("Request failed: " + ajaxRequest.statusText);
            }
        }
    }

    function getServerTime() {
        ajaxRequest = new XMLHttpRequest();
        var svcUrl = "telltme.php";
        ajaxRequest.onreadystatechange = ajaxResponse;
        ajaxRequest.open("GET", svcUrl);
        ajaxRequest.send(null);
    }

    function autoUpdateTime() {
        getServerTime(); // Call getServerTime() initially
        setInterval(getServerTime, 1000); // Call getServerTime() every
second (1000 milliseconds)
    }
</script>

<body onload="autoUpdateTime()">
    <h1>Ajax Demonstration</h1>
    <h2>Getting the server time without refreshing the page</h2>

```

```

<form>
  <input type="button" value="Get Server Time"
onclick="getServerTime()" />
</form>
<br />
<div id="showtime" class="displaybox" style="white-space:
nowrap"></div>
</body>
</html>

```

Saat dijalankan, program tersebut dapat menampilkan waktu nyata secara otomatis setiap detiknya

Ajax Demonstration

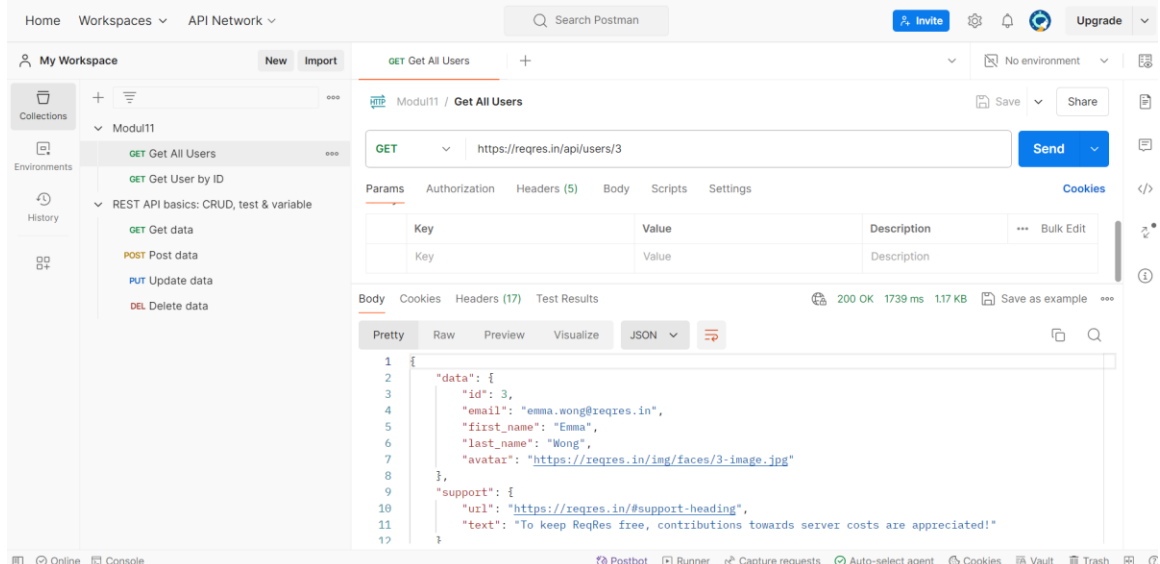
Getting the server time without refreshing the page

Get Server Time

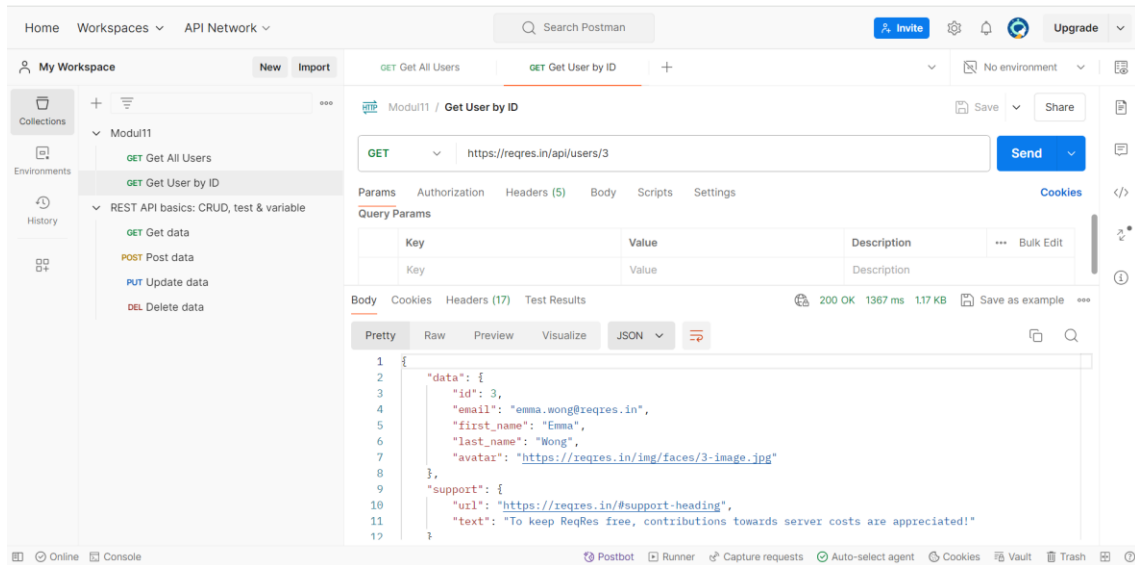
20:55:56

- Memproses data JSON dari API

Pertama untuk melakukan GET API digunakan website Postman. Dengan URL <https://reqres.in/api/users?page=1> dicoba untuk mendapatkan seluruh data user pada postman. Hasilnya seperti berikut



Kemudian, akan dilakukan GET data User By ID



Selain menggunakan Postman, kita juga dapat mengakses API melalui PHP menggunakan fungsi `file_get_contents()` atau library `cURL`. Kita akan mencoba fungsi `file_get_contents()` terlebih dahulu seperti berikut

php11A.php

```
<?php
$data = file_get_contents("https://reqres.in/api/users?page=1");
$parse_data = json_decode($data);

if ($parse_data && isset($parse_data->data)) {
    $users = $parse_data->data;
?>







<table border="1">
    <thead>
        <tr>
            <th>ID</th>
            <th>Nama</th>
            <th>Email</th>
            <th>Gambar</th>
        </tr>
    </thead>
    <tbody>
        <?php foreach ($users as $user) { ?>
            <tr>
                <td><?php echo $user->id; ?></td>
                <td><?php echo $user->first_name . ' ' . $user->last_name; ?></td>
                <td><?php echo $user->email; ?></td>
                <td></td>
```

```

        </tr>
        <?php } ?>
    </tbody>
</table>
<?php
} else {
    echo "Data tidak tersedia atau terjadi kesalahan dalam mengambil
data.";
}
?>

```

Maka diperoleh list data user sebagai berikut

ID	Nama	Email	Gambar
1	George Bluth	george.bluth@reqres.in	
2	Janet Weaver	janet.weaver@reqres.in	
3	Emma Wong	emma.wong@reqres.in	
4	Eve Holt	eve.holt@reqres.in	
5	Charles Morris	charles.morris@reqres.in	
6	Tracey Ramos	tracey.amos@reqres.in	

- Membuat Aplikasi AJAX untuk Pencarian

kita akan membuat aplikasi suggestion pada fitur pencarian. Kita akan menggunakan table meetings yang telah dibuat pada praktikum 9. Kita akan membuat fitur pencarian dimana pengguna dapat mencari data berdasarkan nama, lalu aplikasi akan menampilkan daftar suggestion atau saran pencarian.

Kita akan membuat file php11F.php sebagai interface yang diperoleh dari file php10F.php pada praktikum 10 (termasuk dengan seluruh file pendukungnya), kemudian untuk mengakomodasi fitur suggest dengan function showHint() maka akan dibuat script dalam file php11F_suggestion.js. Terakhir kita akan membuat file php11_gethint untuk mendapatkan saran pencarian, kita perlu mengambil nama-nama di tabel meetings yang sesuai dengan kata kunci yang diinput oleh user pada textfield pencarian dalam format JSON. Berikut programnya

php11F.php

```
<?php
    session_start(); // Memulai session
    if (!isset($_SESSION['username'])){
        header("Location: php11D.php");
    }
?>
<!DOCTYPE html>
<html lang='en-GB'>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PHP11 F</title>
    <style>
        body {
            font-family: Arial, sans-serif;
        }

        h1 {
            text-align: center;
        }

        table {
            width: 80%;
            border-collapse: collapse;
            margin : 0 auto;
        }

        th, td {
            border: 1px solid #ddd;
            padding: 8px;
            text-align: left;
        }

        th {
            background-color: #f2f2f2;
        }

        .btn-container {
            display: flex;
            justify-content: space-between;
        }

        .btn-container a {
            margin: 0 5px;
        }
    </style>
</head>
<body>
```

```

        .success-message {
            background-color: #4CAF50;
            color: white;
            padding: 10px;
            margin-bottom: 20px;
            border-radius: 5px;
            text-align: center;
        }

        /* Style for the tambah data button */
        .btn-tambah {
            display: block;
            width: 150px;
            margin-left: auto;
            margin-top: 20px;
            margin-right: 130px;
            padding: 10px;
            text-align: center;
            background-color: #007bff;
            color: #fff;
            border: none;
            border-radius: 5px;
            cursor: pointer;
            transition: background-color 0.3s ease;
            text-decoration: none;
            justify-content: right;
        }

        .btn-tambah:hover {
            background-color: #0056b3;
        }
    </style>
    <script>
        function showHint(str) {
            if (str.length == 0) {
                document.getElementById("txtHint").innerHTML = "";
                return;
            } else {
                var xmlhttp = new XMLHttpRequest();
                xmlhttp.onreadystatechange = function() {
                    if (this.readyState == 4 && this.status == 200) {
                        document.getElementById("txtHint").innerHTML =
this.responseText;
                    }
                };
            }
        }
    </script>

```

```

        xmlhttp.open("GET", "gethint.php?q=" + str, true);
        xmlhttp.send();
    }
}
</script>
</head>
<body>
    <?php include('php11F_header.php'); ?>

    <h1>Data Meeting</h1>
    <?php
        // Tampilkan pemberitahuan jika ada parameter success di URL
        if (isset($_GET['success']) && $_GET['success'] == 1) {
            echo "<div class='success-message'>Data berhasil
dihapus.</div>";
        }
    ?>

    <!-- Form pencarian -->
    <form>
        Name: <input type="text" id="txt1" onkeyup="showHint(this.value)">
    </form>
    <p>Suggestions: <span id="txtHint"></span></p>

    <?php
        // Pengaturan koneksi ke database
        $db_hostname = "localhost"; // Ganti dengan server database Anda
        $db_database = "praktikum9"; // Ganti dengan nama database Anda
        $db_username = "root"; // Ganti dengan username database Anda
        $db_password = ""; // Ganti dengan password database Anda

        // Koneksi ke database
        $pdo = new PDO("mysql:host=$db_hostname;dbname=$db_database",
$db_username, $db_password);
        $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

        // Mendapatkan data dari tabel meetings
        $stmt = $pdo->query("SELECT * FROM meetings ORDER BY slot");
        $rows = $stmt->fetchAll();

        if (count($rows) > 0) {

            echo "<table>";

```



```

        echo
"<tr><th>Slot</th><th>Name</th><th>Email</th><th>Action</th></tr>";
        foreach ($rows as $row) {
            echo "<tr>";
            echo "<td>" . $row["slot"] . "</td>";
            echo "<td>" . $row["name"] . "</td>";
            echo "<td>" . $row["email"] . "</td>";
            echo "<td class='btn-container'>";
            echo "<a href='php11G.php?slot=" . $row["slot"] . "'><img
src='img/edit.png' style='width:30px;height:30px;' alt='Edit'
title='Edit'></a>";
            echo "<a href='php11H.php?slot=" . $row["slot"] . "'><img
src='img/remove.png' style='width:30px;height:30px;' alt='Delete'
title='Delete'></a>";
            echo "</td>";
            echo "</tr>";
        }
        echo "</table>";
    } else {
        echo "No data found.";
    }

    // Menutup koneksi database
    $pdo = null;
    ?>
    <!-- Tambah Data button -->
    <a href="php11E.php" class="btn-tambah">Tambah Data</a>

    <script src="php11F_suggestion.js"></script>
</body>
</html>

```

php11F_suggestion.js

```

function showHint(str) {
    if (str.length == 0) {
        document.getElementById("txtHint").innerHTML = ""; // Menghapus konten
        pada elemen txtHint jika text field kosong
        return; // Keluar dari fungsi jika text field kosong
    }
    xhttp = new XMLHttpRequest();

    // Membuat fungsi yang akan dieksekusi ketika respons server sudah siap
    xhttp.onreadystatechange = function () {
        if (this.readyState == 4 && this.status == 200) {

```

```

        document.getElementById("txtHint").innerHTML = this.responseText; //
        Menampilkan saran pencarian yang didapat dari server pada elemen txtHint
    }
};

// Mengirimkan permintaan ke file di server
xhttp.open("GET", "php11F_gethint.php?keyword=" + str, true); //
Parameter keyword berisi kata kunci pencarian pada txt1
xhttp.send();
}

```

php11F_gethint.php

```

<?php
// Pengaturan koneksi ke database
$db_hostname = "localhost"; // Ganti dengan server database Anda
$db_database = "praktikum9"; // Ganti dengan nama database Anda
$db_username = "root"; // Ganti dengan username database Anda
$db_password = ""; // Ganti dengan password database Anda
$db_charset = "utf8mb4"; // Opsional
$dsn =
"mysql:host=$db_hostname;dbname=$db_database;charset=$db_charset";
$opt = array(
    PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
    PDO::ATTR_EMULATE_PREPARES => false
);

try {
    // Membuat koneksi PDO
    $pdo = new PDO($dsn, $db_username, $db_password, $opt);

    // Mendapatkan kata kunci pencarian dari parameter GET
    $keyword = isset($_GET['keyword']) ? $_GET['keyword'] : '';

    // Query untuk mendapatkan nama-nama yang memuat kata kunci
    // pencarian yang diinput oleh user
    $stmt = $pdo->prepare("SELECT name FROM meetings WHERE name LIKE
:keyword");
    $stmt->execute(['keyword' => '%' . $keyword . '%']);

    // Menginisialisasi array untuk menyimpan hasil query
    $results = [];

```

```

// Memasukkan nama-nama yang memuat kata kunci pencarian ke dalam
array
while ($row = $stmt->fetch()) {
    $results[] = $row;
}

// Mengirimkan hasil sebagai JSON
if (!empty($results)) {
    echo json_encode($results);
} else {
    // Output "no suggestion" jika tidak ada saran yang ditemukan
    $response[] = array('name' => 'no suggestion');
    echo json_encode($response);
}

// Menutup koneksi database
$pdo = null;
} catch (PDOException $e) {
    exit("PDO Error: " . $e->getMessage() . "<br>");
}
?>







```

Data Meeting
Logout

Data Meeting

Name:

Suggestions: [{"name": "Trish Shelby"}]

Slot	Name	Email	Action
1	Michael North	m.north@foe.stis.ac.id	 
4	Trish Shelby	t.shelby@foe.stis.ac.id	 
5	Harry Garden	h.garden@foe.stis.ac.id	 

Tambah Data

Nampak bahwa saat mengetikan “tri” pada textfield, akan langsung muncul suggestion nama yang dicari dalam format JSON.