## **TUGAS INDIVIDU**

## LAPORAN PRAKTIKUM 8 PEMROGRAMAN WEB

Disusun untuk memenuhi tugas mata kuliah Pemrograman Web

Dosen Pengampu:

Dr. Eng. Agussalim, M.T.



### Disusun oleh:

Muhammad Surya Adhi Setiawan (21082010167)

https://github.com/suryaddede/Kuliah/tree/main/Pertemuan%208

Program Studi Sistem Informasi

Fakultas Ilmu Komputer

Universitas Pembangunan Nasional "Veteran" Jawa Timur

## Tujuan Praktikum

- 1. Memahami langkah-langkah koneksi PHP dengan MySQL.
- 2. Memahami perbedaan penggunaan mysql dengan mysqli.

#### Landasan Teori

#### PHP

PHP (Hypertext Preprocessor) adalah bahasa pemrograman server-side yang digunakan untuk membuat aplikasi web dinamis. PHP dapat berinteraksi dengan berbagai jenis database, termasuk MySQL.

#### • MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang digunakan untuk menyimpan dan mengelola data. MySQL sangat populer di kalangan pengembang web karena mudah digunakan dan gratis.

## • Koneksi antara PHP dan MySQL

Untuk menghubungkan PHP dengan MySQL, Anda perlu membuat koneksi ke server MySQL menggunakan fungsi mysqli\_connect() atau mysql\_connect(). Setelah koneksi berhasil dibuat, Anda dapat melakukan operasi pada database seperti SELECT, INSERT, UPDATE, dan DELETE menggunakan fungsi mysqli\_query() atau mysql\_query().

#### • Perbedaan penggunaan mysql dengan mysqli

mysql adalah ekstensi PHP yang digunakan untuk mengakses database MySQL. Namun, ekstensi ini sudah tidak disarankan lagi karena sudah tidak dikembangkan lagi dan rentan terhadap serangan SQL injection. mysqli adalah ekstensi PHP yang lebih aman dan memiliki fitur yang lebih lengkap dibandingkan mysql. Ekstensi ini juga mendukung koneksi ke server MySQL melalui protokol TCP/IP dan protokol Unix socket. Selain itu, mysqli juga mendukung prepared statement, transaksi, dan prosedur tersimpan.

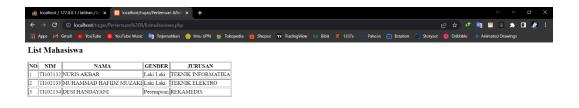
# Tools dan Software yang Digunakan

Untuk melakukan praktikum ini, beberapa tools dan software yang saya digunakan adalah sebagai berikut:

- 1. Text Editor: Seperti Visual Studio Code, Notepad++, Sublime Text, atau Brackets. Ini adalah alat utama untuk membuat dan mengedit file PHP.
- 2. Web Browser: Seperti Google Chrome, Mozilla Firefox, atau Microsoft Edge. Ini diperlukan untuk melihat hasil dari file PHP yang dibuat.

## **Hasil Latihan**

## List Mahasiswa



Alt+Q

Kode tersebut merupakan sebuah program yang digunakan untuk menampilkan data mahasiswa pada sebuah tabel. Program tersebut ditulis menggunakan bahasa pemrograman PHP dan memanfaatkan database MySQL untuk mengakses data yang akan ditampilkan pada tabel.

Pada awal program, terdapat tag HTML yaitu tag table, dimana pada tag tersebut terdapat atribut border yang diberikan nilai 1, sehingga garis tepi pada tabel akan terlihat. Kemudian, terdapat tag PHP yang berfungsi untuk memanggil file koneksi.php. File ini berisi koneksi ke database MySQL yang akan digunakan untuk mengakses data mahasiswa.

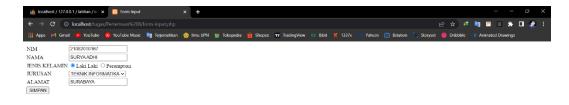
Selanjutnya, program akan mengeksekusi query MySQL dengan perintah SELECT \* FROM mahasiswa untuk mengambil seluruh data yang ada pada tabel mahasiswa. Data tersebut disimpan pada variabel \$\\$\text{mahasiswa}\$ dan akan diakses pada perulangan foreach untuk ditampilkan pada tabel.

Pada perulangan foreach, terdapat variabel \$row yang menyimpan data dari setiap baris pada tabel mahasiswa. Kemudian, terdapat variabel \$jenis\_kelamin yang digunakan untuk mengubah nilai jenis\_kelamin dari P menjadi Perempuan atau L menjadi Laki Laki. Hal ini dilakukan dengan menggunakan operator ternary.

Selanjutnya, program akan menampilkan data mahasiswa pada tabel dengan memanfaatkan tag HTML yaitu tag tr dan td. Tag tr digunakan untuk membuat baris pada tabel, sedangkan tag td digunakan untuk menampilkan data pada setiap sel di dalam baris tersebut. Variabel \$no digunakan untuk memberikan nomor urut pada setiap baris.

Secara keseluruhan, program tersebut cukup sederhana dan mudah dipahami. Namun, program tersebut juga perlu diperhatikan dari segi keamanan, terutama dalam hal penggunaan query MySQL yang rentan terhadap serangan SQL injection. Oleh karena itu, perlu dilakukan sanitasi data pada input yang diterima sebelum dieksekusi pada query MySQL.

## **Form Input**



Alt+Q

 Alt+Q

```
| Recording | Section | New | Got Run | Remind | Help | Some input/ptp | Taylor | Flagor | Fl
```

Kode tersebut merupakan sebuah program yang digunakan untuk membuat form input data mahasiswa pada sebuah halaman web. Program tersebut ditulis menggunakan bahasa pemrograman HTML dan berisi elemen-elemen form seperti input text, radio button, dan select option.

Pada awal program, terdapat tag HTML yaitu tag form, dimana pada tag tersebut terdapat atribut method yang diberikan nilai post dan action yang diberikan nilai simpan.php. Atribut method ini menunjukkan bahwa data yang dikirim melalui form akan diolah menggunakan metode HTTP POST, sedangkan atribut action menunjukkan bahwa data tersebut akan diproses pada file simpan.php.

Selanjutnya, terdapat tag HTML yaitu tag table yang digunakan untuk membuat tabel pada halaman web. Pada tabel tersebut, terdapat beberapa baris yang merepresentasikan elemen-elemen form seperti NIM, NAMA, JENIS KELAMIN, JURUSAN, dan ALAMAT. Setiap elemen tersebut diwakili oleh tag td dan input yang berbeda-beda, tergantung dari tipe input yang dibutuhkan.

Pada elemen JENIS KELAMIN, terdapat tag input dengan tipe radio dan atribut name yang sama yaitu jenis\_kelamin. Hal ini dilakukan agar pengguna hanya dapat memilih salah satu opsi dari radio button tersebut. Opsi tersebut merepresentasikan jenis kelamin Laki Laki dan Perempuan, dan masing-masing memiliki nilai yang berbeda yaitu L dan P.

Pada elemen JURUSAN, terdapat tag select yang memungkinkan pengguna untuk memilih salah satu opsi dari beberapa pilihan yang disediakan. Pilihan tersebut direpresentasikan oleh tag option yang berada di dalam tag select. Setiap option memiliki nilai yang berbeda-beda yaitu TEKNIK INFORMATIKA, TEKNIK MESIN, dan TEKNIK KIMIA.

Pada elemen terakhir yaitu ALAMAT, terdapat tag input dengan tipe text yang memungkinkan pengguna untuk memasukkan alamat sesuai dengan kebutuhan.

Setelah elemen-elemen form terbentuk, terdapat tombol SIMPAN yang merepresentasikan aksi yang akan dilakukan ketika pengguna menekannya. Tombol tersebut direpresentasikan oleh tag button dengan atribut type yang diberikan nilai submit. Ketika tombol ini ditekan, data yang telah diisi akan dikirimkan ke file simpan.php untuk diproses lebih lanjut.

## Simpan



Alt+Q

```
** Bit Selection View Go Run terminal Help sempanty ** minoralization ** minoralizat
```

Kode tersebut adalah script PHP yang digunakan untuk memproses data yang dikirim melalui form input pada file HTML sebelumnya. Script ini berfungsi untuk mengambil nilai dari setiap elemen form yang disubmit oleh pengguna dan menyimpannya ke dalam variabel. Kemudian, data tersebut akan di-insert ke dalam database menggunakan query SQL.

Pertama, script ini memanggil file koneksi.php yang berisi konfigurasi untuk menghubungkan dengan database. Selanjutnya, script mengambil data dari setiap elemen form menggunakan fungsi \$\_POST['nama\_input'] dan menyimpannya ke dalam variabel yang sesuai.

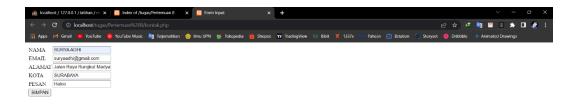
Setelah variabel yang berisi data telah disiapkan, script ini menggunakan query SQL untuk menambahkan data baru ke dalam tabel mahasiswa pada database. Query INSERT INTO digunakan dengan sintaks SET untuk menentukan nilai untuk setiap kolom pada tabel. Variabel-variabel yang berisi nilai dari elemen form ditentukan sebagai nilai kolom-kolom pada tabel tersebut.

Setelah data berhasil ditambahkan ke dalam database, script akan mengalihkan pengguna ke halaman listmahasiswa.php menggunakan fungsi header("location:listmahasiswa.php").

Kode tersebut merupakan contoh implementasi dari fungsi CRUD (Create, Read, Update, Delete) dalam pengembangan web. Dengan menggunakan script seperti ini, pengembang web dapat membuat aplikasi yang dapat mengolah data pengguna dengan mudah. Namun, perlu diingat bahwa script seperti ini harus dikembangkan dengan mempertimbangkan masalah keamanan, seperti SQL injection dan validasi input pengguna.

## Hasil Praktikum

## Kontak



Alt+Q

Kode di atas adalah sebuah halaman form yang digunakan untuk menginput data kontak. Pada baris ke-5, form mengarahkan aksinya ke file simpankontak.php saat tombol submit ditekan. Kemudian pada baris ke-8 sampai 17, dibuat sebuah tabel dengan beberapa kolom yang akan diisi oleh pengguna. Kolom-kolom tersebut terdiri dari nama, email, alamat, kota, dan pesan. Setiap kolom memiliki input field yang berbeda, seperti pada baris ke-10 yang menggunakan input type "text" untuk email, dan baris ke-14 yang menggunakan input type "textarea" untuk pesan. Setelah semua kolom diisi, pengguna dapat menyimpan datanya dengan menekan tombol "SIMPAN" pada baris ke-18.

Saat tombol "SIMPAN" ditekan, form akan mengirimkan data yang telah diisi ke file simpankontak.php melalui method POST. Pada file tersebut, data yang dikirimkan akan disimpan pada variabel-variabel yang bersesuaian dengan nama kolom pada tabel yang digunakan untuk menyimpan data kontak di database. Setelah data disimpan pada variabel-variabel tersebut, sebuah query SQL akan dibuat pada baris ke-14 untuk memasukkan data ke dalam tabel kontak dengan menggunakan perintah INSERT INTO. Query tersebut menggunakan variabel-variabel yang telah diisi sebelumnya sebagai nilainya. Selanjutnya, pada baris ke-16, fungsi mysqli\_query() digunakan untuk menjalankan query yang telah dibuat. Kemudian pada baris ke-19, pengguna akan dialihkan kembali ke halaman listkontak.php setelah data berhasil disimpan.

Dalam keseluruhan, kode di atas menunjukkan cara membuat sebuah form input untuk menginput data kontak, dan cara memproses data yang diinputkan melalui form tersebut untuk kemudian disimpan ke dalam tabel kontak di database.

# **Tampil Kontak**



Alt+Q

Kode di atas adalah tampilan halaman web yang menampilkan daftar kontak yang tersimpan dalam sebuah database. Halaman ini terdiri dari sebuah tabel yang berisi daftar kontak, dan masing-masing kontak memiliki kolom-kolom seperti nama, gender, email, alamat, kota, dan pesan.

Untuk mengambil data kontak dari database, digunakan fungsi 'mysqli\_query' dari file koneksi.php, yaitu '\$kontak = mysqli\_query(\$koneksi, "SELECT \* FROM kontak"); '. Setiap kontak diambil dengan menggunakan perulangan 'foreach', dan hasilnya ditampilkan dalam bentuk tabel menggunakan perintah 'echo'. Dalam hal ini, variabel '\$row' berisi data kontak, dan masingmasing kolom ditampilkan dalam sel-sel tabel.

Pada kolom gender, digunakan data jenis kelamin dalam bentuk singkatan, yaitu "L" untuk laki-laki dan "P" untuk perempuan. Selain itu, dalam tabel ini juga terdapat nomor urut pada kolom pertama.

Untuk menampilkan tabel kontak ini, diperlukan file koneksi.php yang berfungsi untuk menghubungkan halaman web dengan database. Koneksi.php diinclude pada kode ini dengan perintah 'include 'koneksi.php';'.

Dalam tampilan ini, terdapat beberapa kekurangan pada kode, seperti tidak adanya validasi input dan tidak adanya tombol untuk mengedit atau menghapus data kontak. Namun, secara umum, kode ini dapat dijadikan dasar untuk membuat halaman tampilan daftar kontak yang lebih lengkap dan fungsional.

## Simpan Kontak



Alt+Q

```
| The Data Selection View Go Run Rumminal Help | simple and provided | Selection View Go Run Rumminal Help | simple and provided | Selection View Go Run Rumminal Help | Selection View Go Rum Rumminal Help | Selection View Rumminal H
```

Kode tersebut merupakan sebuah script PHP yang bertujuan untuk menyimpan data yang diinput oleh pengguna melalui formulir pada database. Script tersebut berisi beberapa variabel, yaitu '\$nama', '\$jkel', '\$email', '\$alamat', '\$kota', dan '\$pesan' yang diisi dengan nilai yang diinput oleh pengguna melalui metode '\$\_POST'. Selanjutnya, script tersebut menjalankan sebuah query SQL yang berisi perintah untuk menyimpan data yang diinput oleh pengguna ke dalam database. Query tersebut akan menjalankan perintah INSERT INTO untuk memasukkan data ke dalam tabel kontak pada database. Setelah query dijalankan dan data berhasil disimpan ke dalam database, pengguna akan dialihkan ke halaman tampilkontak.php menggunakan fungsi 'header()'.

Script tersebut terdiri dari beberapa bagian, yaitu:

### 1. Include koneksi.php

Script tersebut memanggil file koneksi.php yang berisi informasi mengenai koneksi ke database yang dibutuhkan oleh script.

#### 2. Variabel

Script tersebut mendefinisikan beberapa variabel yang berisi nilai yang diinput oleh pengguna melalui formulir.

## 3. Query SQL

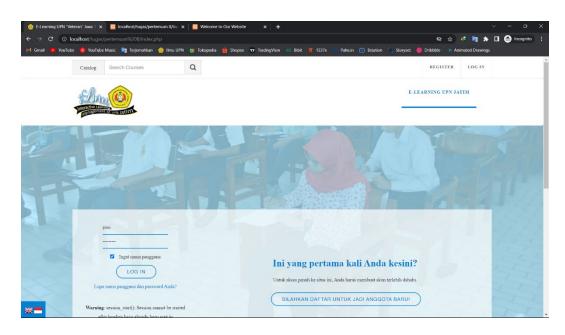
Script tersebut menjalankan sebuah query SQL yang berisi perintah INSERT INTO untuk menyimpan data yang diinput oleh pengguna ke dalam database.

#### 4. Mengalihkan ke halaman tampilkontak.php

Setelah query dijalankan dan data berhasil disimpan ke dalam database, pengguna akan dialihkan ke halaman tampilkontak.php menggunakan fungsi 'header()'.

# **Tugas Modul 7**

## Index



```
indexulp X is blooked view to no no hominal Help indexulps. Toges-Year Management (in the proper of the proper of
```

Kode tersebut merupakan duplikasi kode HTML dan PHP dari website ilmu.upnjatim.ac.id yang berfungsi untuk menampilkan halaman login pada website tersebut. Halaman login tersebut terdiri dari dua kolom, yaitu kolom untuk login dan kolom untuk membuat akun baru.

Pada bagian HTML, terdapat beberapa tag seperti div, section, span, form, label, input, dan script. Div digunakan untuk membuat container yang bisa diberi atribut seperti id dan class, sedangkan section digunakan untuk mengelompokkan konten yang terkait. Span digunakan untuk menampilkan pesan notifikasi atau pesan lainnya, sedangkan form digunakan untuk mengumpulkan data yang akan disubmit. Label digunakan untuk memberikan label pada input form, sedangkan input digunakan untuk menerima input dari pengguna, seperti username dan password. Script digunakan untuk menjalankan kode JavaScript pada halaman tersebut.

Pada bagian PHP, terdapat require 'login.php'; yang digunakan untuk memanggil file login.php yang berisi kode untuk memproses data login dari pengguna. Terdapat juga atribut action pada form yang diisi dengan login.php, sehingga data yang dimasukkan oleh pengguna akan dikirimkan ke file tersebut untuk diproses.

Pada halaman login tersebut, pengguna diminta untuk memasukkan nama pengguna dan kata sandi yang benar. Terdapat juga opsi untuk mengaktifkan fitur "Ingat nama pengguna" pada browser agar pengguna tidak perlu memasukkan nama pengguna setiap kali masuk ke halaman tersebut. Jika pengguna lupa nama pengguna atau kata sandi, terdapat link untuk membantu pengguna memulihkan akun tersebut.

Untuk pengguna tamu, terdapat juga opsi untuk masuk sebagai tamu dengan mengisi username dan password default "guest". Selain itu, terdapat opsi untuk membuat akun baru dengan mengklik tombol "Silahkan daftar untuk jadi anggota baru!".

Pada bagian akhir halaman, terdapat pesan bahwa cookie harus diaktifkan pada browser untuk dapat mengakses situs tersebut.

## Login

Kode di atas merupakan sebuah script PHP yang digunakan untuk melakukan login pada sebuah aplikasi web. Pertama, script ini memulai session menggunakan fungsi 'session\_start()'. Selanjutnya, script ini melakukan koneksi ke database dengan menggunakan informasi login seperti nama server, nama pengguna, password, dan nama database.

Setelah melakukan koneksi, script ini akan melakukan verifikasi login menggunakan metode POST. Pertama-tama, script akan mengambil nilai 'username' dan 'password' yang diinputkan oleh pengguna melalui formulir login pada halaman web. Password yang dimasukkan akan di-hash menggunakan fungsi 'hash()' dengan algoritma SHA-512 sebelum dilakukan query ke database.

Query tersebut akan mencari baris pada tabel 'users' dengan username dan password yang sesuai dengan nilai yang dimasukkan oleh pengguna. Jika hasil query mengembalikan 1 baris, maka login berhasil dan session username akan diset ke nilai 'username' yang dimasukkan. Kemudian, pengguna akan diarahkan ke halaman cetak.php dengan menggunakan fungsi 'header()'. Namun, jika hasil query tidak mengembalikan 1 baris, maka login gagal dan pesan error "Username atau password salah" akan ditampilkan pada halaman login.

Secara keseluruhan, script ini melakukan proses autentikasi sederhana dengan mengambil nilai 'username' dan 'password' yang dimasukkan oleh pengguna melalui formulir login pada halaman web, melakukan hashing password menggunakan SHA-512, dan mencari baris pada tabel 'users' dengan username dan password yang sesuai. Script ini juga menggunakan session untuk menyimpan informasi login dan melakukan redirect ke halaman cetak.php jika login berhasil.

## Cetak



Kode di atas merupakan kode HTML dan PHP yang digunakan untuk membuat sebuah halaman website. Halaman tersebut memiliki beberapa bagian seperti header, about, services, contact, dan footer. Header berisi navbar, gambar, dan pesan selamat datang yang di-generate secara dinamis oleh kode PHP. Bagian about berisi teks deskripsi tentang perusahaan dengan gambar. Bagian services berisi tiga kotak yang berisi informasi tentang layanan yang ditawarkan oleh perusahaan. Bagian contact berisi formulir kontak dan informasi kontak perusahaan. Bagian footer berisi teks yang menandakan pencipta halaman.

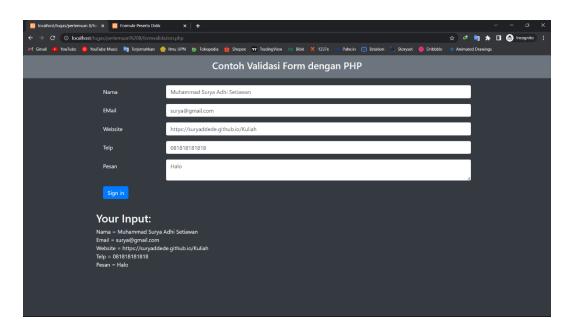
Kode PHP yang digunakan pada header bertujuan untuk memeriksa apakah pengguna sudah login atau belum. Jika belum, maka pengguna akan di-redirect ke halaman login. Jika pengguna sudah login, maka pesan selamat datang yang digenerate secara dinamis akan ditampilkan pada header.

Kode CSS juga digunakan pada halaman tersebut untuk mengatur tampilan halaman. Kode CSS yang digunakan diambil dari font-awesome dan cetak.css. File cetak.css berfungsi untuk mengatur tampilan halaman yang berkaitan dengan cetak.

Kode HTML dan PHP pada halaman tersebut terlihat rapi dan terstruktur dengan baik. Hal ini memudahkan pengembangan dan pemeliharaan halaman di masa depan. Tampilan halaman yang dibuat juga terlihat menarik dengan penggunaan warna yang tepat dan gambar yang sesuai. Selain itu, penggunaan kode PHP pada header juga menunjukkan bahwa pengembang memiliki pemahaman tentang keamanan website dan mempertimbangkan hal ini saat membuat halaman tersebut.

## Latihan Modul 10

#### Form Validation



```
cheads

(illes-Learning UPN EquotyVeteranEquot; Jana Timur: Massak ke situs</title>

(illes-Learning UPN EquotyVeteranEquot; Jana Timur: Massak ke situs* / FollowInk(Ctd+ dick)

(illes-Learning UPN EquotyVeteranEquot; Jana Timur: Massak ke situs* / FollowInk(Ctd+ dick)

(illes-Learning UPN EquotyVeteranEquot; Jana Timur: Massak ke situs* / FollowInk(Ctd+ dick)

(illes-Learning UPN EquotyVeteranEquot; Jana Timur: Massak ke situs* / FollowInk(Ctd+ dick)

(illes-Learning UPN EquotyVeteranEquoty Jana Timur: Massak ke situs* / FollowInk(Ctd+ dick)

(illes-Learning UPN EquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVeteranEquotyVete
//si[consi(
var N = []; N.yui = [];
N.pageloadstarttime = new Date();
N.pa
```

Kode yang diberikan merupakan sebuah contoh implementasi validasi form dengan menggunakan bahasa pemrograman PHP. Validasi form ini sangat penting untuk memastikan data yang masuk ke database tidak mengandung karakter yang tidak diinginkan atau tidak valid, sehingga data tersebut dapat diproses dengan benar oleh aplikasi.

Pertama-tama, kode ini mengecek apakah method yang digunakan dalam request adalah POST atau bukan. Jika iya, maka kode akan memproses data yang di-inputkan oleh pengguna. Kode kemudian melakukan validasi pada masing-masing input form, dimulai dari input nama. Jika input nama kosong, maka akan muncul pesan error "Nama tidak boleh kosong". Namun, jika input nama tidak kosong, maka kode akan memeriksa apakah input nama hanya terdiri dari huruf dan spasi. Jika input nama tidak memenuhi syarat, maka akan muncul pesan error "Inputan hanya boleh huruf dan spasi".

Selanjutnya, kode melakukan validasi pada input email. Jika input email kosong, maka akan muncul pesan error "Email tidak boleh kosong". Namun, jika input email tidak kosong, maka kode akan memeriksa apakah input email memenuhi format email yang valid. Jika input email tidak memenuhi syarat, maka akan muncul pesan error "Format email invalid".

Kode selanjutnya melakukan validasi pada input website. Jika input website kosong, maka akan muncul pesan error "Website tidak boleh kosong". Namun, jika input website tidak kosong, maka kode akan memeriksa apakah input website memenuhi format URL yang valid. Jika input website tidak memenuhi syarat, maka akan muncul pesan error "URL tidak valid".

Terakhir, kode melakukan validasi pada input telepon. Jika input telepon kosong, maka akan muncul pesan error "Telepon tidak boleh kosong". Namun, jika input telepon tidak kosong, maka kode akan memeriksa apakah input telepon hanya terdiri dari angka. Jika input telepon tidak memenuhi syarat, maka akan muncul pesan error "Nomor HP hanya boleh angka".

Untuk melakukan validasi pada setiap input, kode ini menggunakan fungsi cek\_input yang bertujuan untuk membersihkan data dari input pengguna agar tidak terdapat karakter-karakter yang tidak diinginkan seperti spasi atau karakter khusus. Fungsi ini menghilangkan spasi di awal dan akhir input, menghapus karakter backslash (\) dan karakter-karakter lain yang terkait dengan HTML, dan mengembalikan data yang sudah bersih.

Setiap input form disertai dengan tag span yang akan menampilkan pesan error jika terdapat kesalahan pada input tersebut. Pesan error ini diinisialisasi terlebih dahulu sebagai string kosong dan hanya akan diisi jika terdapat kesalahan pada input form.

Kode ini juga menggunakan CSS class "is-invalid" pada setiap input form yang mengalami kesalahan, sehingga tampilan form akan berubah menjadi merah pada bagian yang tidak valid.

Dalam halaman HTML, form ini dibungkus dalam sebuah div dengan class "card-body", yang kemudian ditempatkan di dalam div dengan class "col-md-6" dan "row". selanjutnya pada kode tersebut, terdapat fungsi cek\_input yang digunakan untuk membersihkan inputan yang diterima dari form. Fungsi ini menerima satu parameter yaitu \$data, kemudian mengembalikan data yang sudah bersih dengan menggunakan tiga fungsi bawaan PHP, yaitu trim(), stripslashes(), dan htmlspecialchars(). Fungsi trim() digunakan untuk menghilangkan whitespace pada awal dan akhir string. Fungsi stripslashes() digunakan untuk menghilangkan backslash () yang mungkin ada pada inputan. Sedangkan, fungsi htmlspecialchars() digunakan untuk mengkonversi karakter-karakter tertentu yang memiliki makna khusus pada HTML menjadi karakter-karakter yang aman untuk ditampilkan pada halaman web.

Kode pada bagian form ini menggunakan metode POST untuk mengirimkan data dari form ke server, yaitu dengan menggunakan method="post" pada tag form. Selanjutnya, pada bagian action pada tag form, menggunakan htmlspecialchars(\$ SERVER["PHP SELF"]), yang mengindikasikan bahwa form akan dikirimkan ke halaman itu sendiri ketika tombol "submit" pada form ditekan. Tuiuan dari penggunaan fungsi htmlspecialchars() pada \$ SERVER["PHP SELF"] adalah untuk mencegah serangan cross-site scripting (XSS) pada aplikasi web.

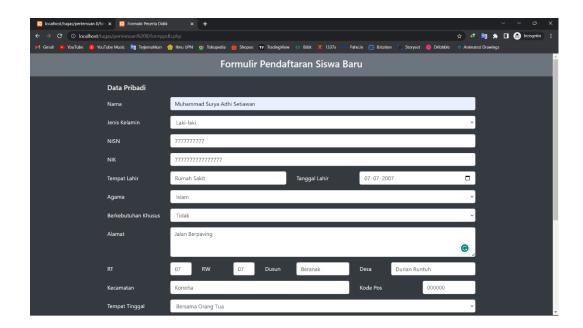
Selanjutnya, kode HTML dibangun dengan menggunakan div, label, dan input. Setiap inputan pada form memiliki atribut name dan id yang digunakan untuk mengidentifikasi inputan. Dalam contoh ini, inputan terdiri dari nama, email, website, nomor telepon, dan pesan yang ingin disampaikan. Untuk memvalidasi inputan, digunakan fungsi-fungsi bawaan PHP, seperti empty(), preg\_match(), dan filter\_var(). Fungsi empty() digunakan untuk memastikan inputan tidak kosong, sedangkan fungsi preg\_match() digunakan untuk memastikan inputan hanya terdiri dari huruf dan spasi. Fungsi filter\_var() digunakan untuk memvalidasi format email dan URL.

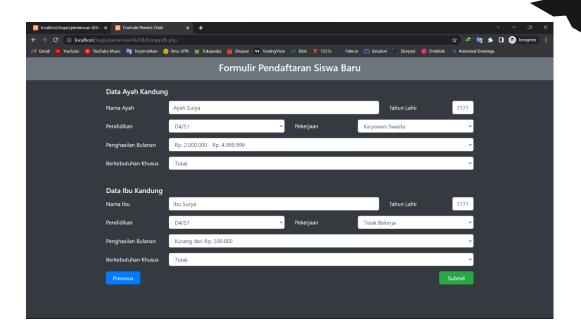
Terakhir, jika terdapat error pada inputan, maka akan ditampilkan pesan error pada masing-masing inputan. Pesan error tersebut disimpan pada variabel yang sudah dideklarasikan sebelumnya. Pesan error ditampilkan pada tag span dengan kelas warning. Sedangkan, untuk menandai inputan yang error, digunakan kelas CSS is-invalid pada tag input. Jika tidak ada error, maka nilai inputan akan ditampilkan kembali pada form ketika form di-refresh.

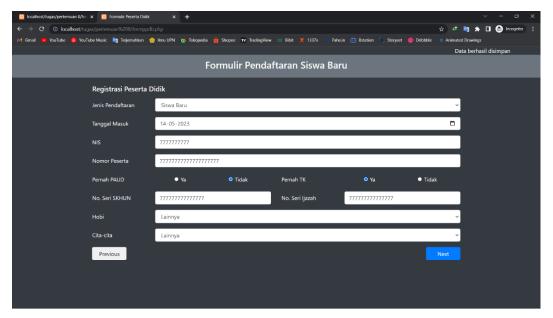
# **Tugas Modul 10**

## Formulir Pendaftaran Peserta Didik

☐ localhost/tugas/pertemuan 8/for >	X 👸 Formulir Peserta Didik	× +					v - o	×
← → C © localhost/tugss/pertenuen/6208/formppdb.php							📭 🖈 🔲 😁 Incogniti	
eri. Granii 🕟 Walfale 🧶 Walfale Music 🥦 Tojernahkan 🔞 Bray LIPN 🔞 Tokapada 👸 Shoper 👣 TadingYew 🗆 Bibli 🔀 1277a \cdots Rahein 🖰 Blation 🖒 Skrayet 🔞 Chibble 💀 Azimated Drawings								
Formulir Pendaftaran Siswa Baru								
	Registrasi Peserta D	idik						
	Jenis Pendaftaran	Siswa Baru				v		
	Tanggal Masuk	14-05-2023				D		
	NIS	7777777777						
	Nomor Peserta	7777777777						
	Pernah PAUD	○ Ya	O Tidak	Pernah TK	• Ya	● Tidak		
	No. Seri SKHUN	7777777777		No. Seri Ijazah	7777777777			
	Hobi	Lainnya				·		
	Cita-cita	Lainnya				·		
	Previous							







```
₽ □

■ formppdb.php ×

                                            div class-form-group row?

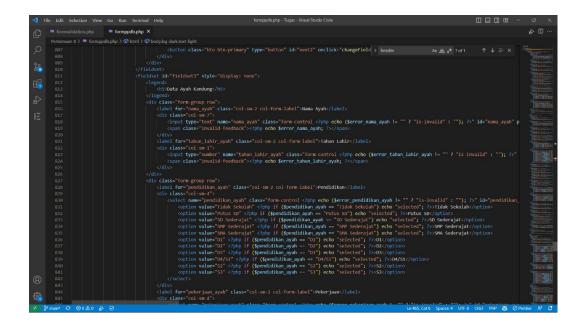
(label for="jents pendaftaran" class="col-sn-2 col-form-label">Jenis Pendaftaran</label>

(label for="jents pendaftaran" class="col-sn-2 col-form-label">Jenis Pendaftaran</label>

(div class-"col-sn-10")

(select name-"jenis pendaftaran" class-"form-control control 
(sphe cho (Serror_jenis pendaftaran != "" ? "is-invalid" : ""); ?>" id-"jer

(option value" "selected disabled>Polis Pendaftaran
(spiton value" "slowa Baru c/php if (Sjenis pendaftaran == "siswa Baru") echo "selected"; ?>>Siswa Baru
(option value" "Pindahan" c/php if (Sjenis pendaftaran == "labahan") echo "selected"; ?>>Pindahan
(spiton value" "Pindahan" cho Serror_jenis pendaftaran; ?>
(spiton value" "Pindahan" cho Serror_jenis pendaftaran; ?>
```



Kode yang diberikan merupakan sebuah contoh implementasi validasi form dengan menggunakan bahasa pemrograman PHP. Validasi form ini sangat penting untuk memastikan data yang masuk ke database tidak mengandung karakter yang tidak diinginkan atau tidak valid, sehingga data tersebut dapat diproses dengan benar oleh aplikasi.

Pertama-tama, kode ini mengecek apakah method yang digunakan dalam request adalah POST atau bukan. Jika iya, maka kode akan memproses data yang di-inputkan oleh pengguna. Kode kemudian melakukan validasi pada masing-masing input form, dimulai dari input jenis pendaftaran hingga input kebutuhan khusus ibu.

Untuk melakukan validasi pada setiap input, kode ini menggunakan fungsi cek\_input yang bertujuan untuk membersihkan data dari input pengguna agar tidak terdapat karakter-karakter yang tidak diinginkan seperti spasi atau karakter khusus. Fungsi ini menghilangkan spasi di awal dan akhir input, menghapus karakter backslash (\) dan karakter-karakter lain yang terkait dengan HTML, dan mengembalikan data yang sudah bersih.

Setiap input form disertai dengan tag span yang akan menampilkan pesan error jika terdapat kesalahan pada input tersebut. Pesan error ini diinisialisasi terlebih dahulu sebagai string kosong dan hanya akan diisi jika terdapat kesalahan pada input form.

Kode ini juga menggunakan CSS class "is-invalid" pada setiap input form yang mengalami kesalahan, sehingga tampilan form akan berubah menjadi merah pada bagian yang tidak valid.

Dalam halaman HTML, form ini dibungkus dalam sebuah div dengan class "card-body", yang kemudian ditempatkan di dalam div dengan class "col-md-6" dan "row". selanjutnya pada kode tersebut, terdapat fungsi cek\_input yang digunakan untuk membersihkan inputan yang diterima dari form. Fungsi ini menerima satu parameter yaitu \$data, kemudian mengembalikan data yang sudah bersih dengan menggunakan tiga fungsi bawaan PHP, yaitu trim(), stripslashes(), dan htmlspecialchars(). Fungsi trim() digunakan untuk menghilangkan whitespace pada awal dan akhir string. Fungsi stripslashes() digunakan untuk menghilangkan backslash () yang mungkin ada pada inputan. Sedangkan, fungsi htmlspecialchars() digunakan untuk mengkonversi karakter-karakter tertentu yang memiliki makna khusus pada HTML menjadi karakter-karakter yang aman untuk ditampilkan pada halaman web.

Kode pada bagian form ini menggunakan metode POST untuk mengirimkan data dari form ke server, yaitu dengan menggunakan method="post" pada tag form. Selanjutnya, pada bagian action pada tag form, menggunakan htmlspecialchars(\$ SERVER["PHP SELF"]), yang mengindikasikan bahwa form akan dikirimkan ke halaman itu sendiri ketika tombol "submit" pada form ditekan. Tujuan penggunaan fungsi htmlspecialchars() \$ SERVER["PHP SELF"] adalah untuk mencegah serangan cross-site scripting (XSS) pada aplikasi web.

Selanjutnya, kode HTML dibangun dengan menggunakan div, label, dan input. Setiap inputan pada form memiliki atribut name dan id yang digunakan untuk mengidentifikasi inputan. Dalam contoh ini, inputan terdiri data pendaftar peserta didik baru. Untuk memvalidasi inputan, digunakan fungsi-fungsi bawaan PHP, seperti empty(), preg\_match(), dan filter\_var(). Fungsi empty() digunakan untuk memastikan inputan tidak kosong, sedangkan fungsi preg\_match() digunakan untuk memastikan inputan hanya terdiri dari huruf dan spasi. Fungsi filter\_var() digunakan untuk memvalidasi format email dan URL.

Terakhir, jika terdapat error pada inputan, maka akan ditampilkan pesan error pada masing-masing inputan. Pesan error tersebut disimpan pada variabel yang sudah dideklarasikan sebelumnya. Pesan error ditampilkan pada tag span dengan kelas warning. Sedangkan, untuk menandai inputan yang error, digunakan kelas CSS is-invalid pada tag input. Jika tidak ada error, maka nilai inputan akan ditampilkan kembali pada form ketika form di-refresh.

# Kesimpulan

Dalam praktikum ini, kita telah belajar bagaimana cara membuat koneksi antara PHP dan MySQL. Koneksi ini sangat penting karena memungkinkan kita untuk mengambil data dari database MySQL dan menampilkannya pada halaman web menggunakan PHP. Dalam praktikum ini, kita telah menggunakan fungsi mysqli\_connect() untuk membuat koneksi ke database MySQL dan fungsi mysqli\_query() untuk melakukan query pada database. Selain itu, kita juga telah belajar bagaimana cara menampilkan data dari database pada halaman web menggunakan PHP.

Pada saat membuat koneksi antara PHP dan MySQL, ada beberapa hal yang perlu diperhatikan, seperti memastikan informasi server, username, password, dan nama database yang digunakan sudah benar. Selain itu, kita juga perlu memastikan bahwa koneksi ke database berhasil terhubung dan query yang dilakukan menghasilkan data yang diinginkan.

Secara keseluruhan, pada praktikum ini kita telah berhasil membuat koneksi antara PHP dan MySQL dan menampilkan data dari database pada halaman web menggunakan PHP. Dengan memahami cara membuat koneksi antara PHP dan MySQL, kita dapat membuat aplikasi web yang lebih kompleks dan interaktif dengan menggunakan database sebagai sumber data.