

Membuat Aplikasi CRUD : Pendaftaran Siswa Baru - Part 1

Persiapan Database, Koneksi dan Menampilkan Data dari Tabel

PHP sering diintegrasikan dengan MySQL untuk membangun aplikasi web. MySQL berperan sebagai tempat menyimpan data (basis data) sedangkan PHP adalah program yang mengelola data tersebut.

Sebenarnya PHP tidak hanya bisa dijoin dengan MySQL saja. PHP juga bisa dijoin dengan SQLite, PostgreSQL, MongoDB, Ms. Access, dan sebagainya.

Pada kesempatan ini, kita akan belajar menggunakan PHP dan MySQL untuk membuat aplikasi CRUD Pendaftaran Siswa Baru dengan model prosedural.

Bagaimana PHP dan MySQL dihubungkan?

PHP dan MySQL dapat saling berhubungan melalui perantara API (*Application Programming Interface*).

Ada tiga jenis API MySQL yang disediakan di PHP:

1. **mysql** API ini sudah tidak didukung/dikembangkan lagi (kadalwarsa).
2. **MySQLi** (*MySQL improvement*) peningkatan dari API **MySQL**
3. **PDO** (*PHP Data Object*)

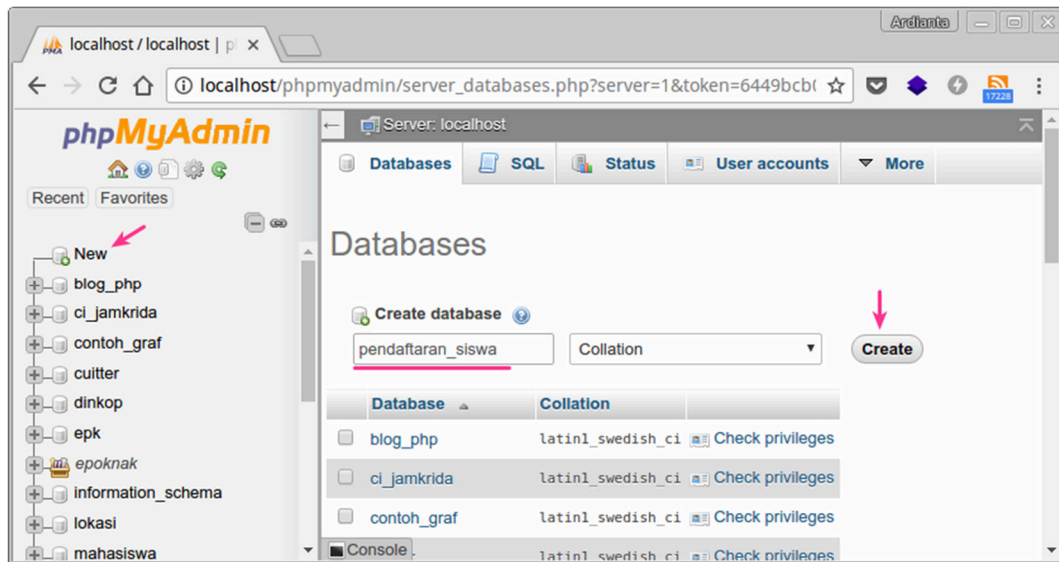
API yang pertama sudah tidak didukung lagi sejak PHP versi 5.5.x dan digantikan dengan **MySQLi** yang lebih canggih. Karena **MySQLi** dapat digunakan secara **OOP** dan prosedural.

Sedangkan API PDO digunakan untuk menghubungkan PHP dengan vendor database yang lainnya. Jadi tidak hanya untuk MySQL saja.

Pada tutorial ini, kita akan belajar cara menghubungkan PHP dan MySQL menggunakan API **MySQLi** secara prosedural.

1. Menyiapkan Database

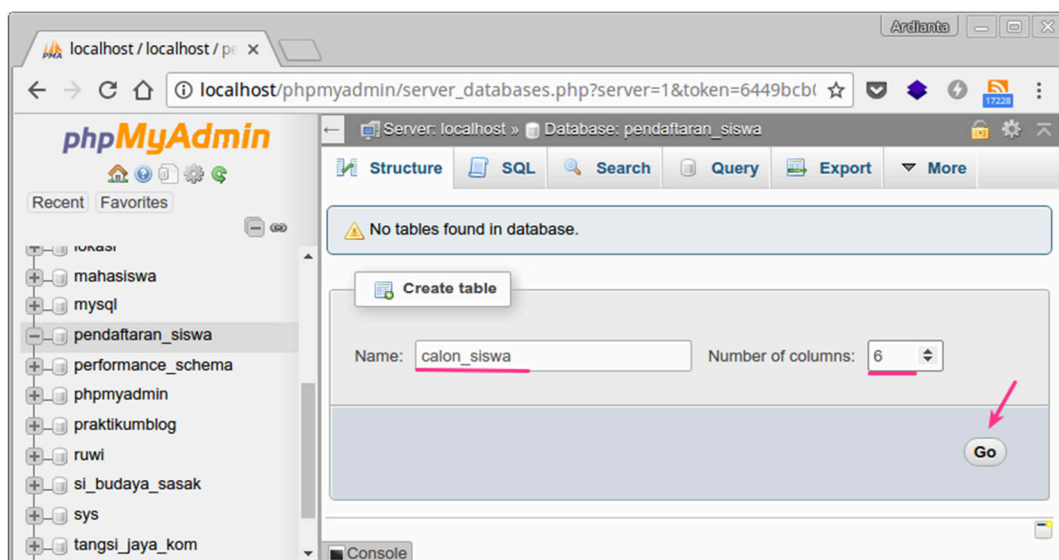
Pertama, kita membuat database bernama **pendaftaran_siswa**, yang dilakukan dengan menggunakan phpMyAdmin yang tersedia dalam XAMPP Anda.



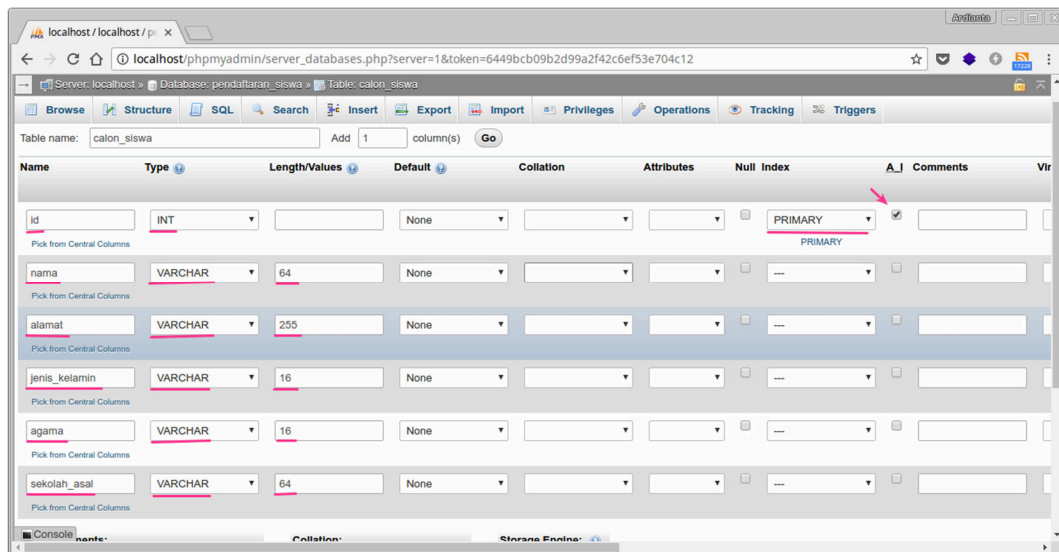
Anda juga bisa membuat database dengan perintah dari menu SQL dengan perintah berikut:

```
CREATE DATABASE `pendaftaran_siswa`;
```

Setelah itu, buatlah tabel di dalam database tersebut dengan nama **calon_siswa**.



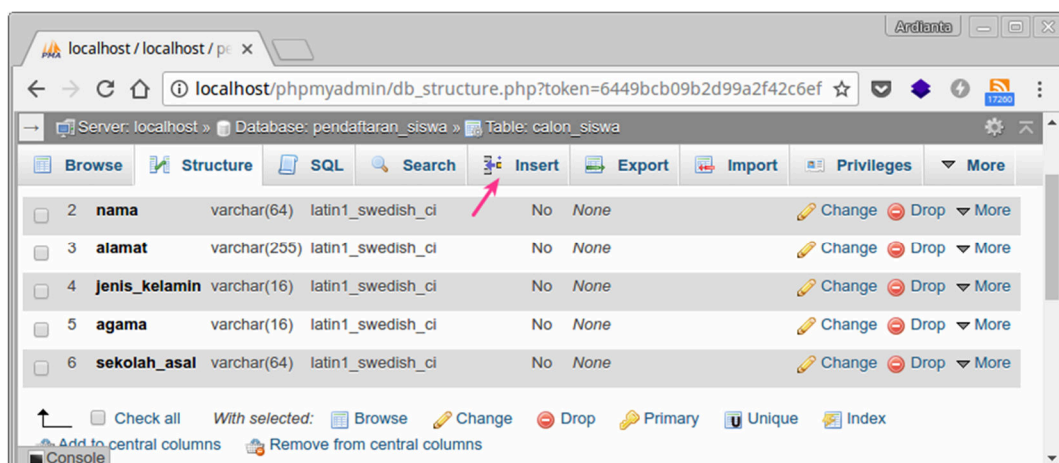
Berikan tipe data kolomnya seperti ini.



Setelah itu klik Save.

```
CREATE TABLE `pendaftaran_siswa`.`calon_siswa` (  
  `id` INT NOT NULL AUTO_INCREMENT ,  
  `nama` VARCHAR(64) NOT NULL ,  
  `alamat` VARCHAR(255) NOT NULL ,  
  `jenis_kelamin` VARCHAR(16) NOT NULL ,  
  `agama` VARCHAR(16) NOT NULL ,  
  `sekolah_asal` VARCHAR(64) NOT NULL ,  
  PRIMARY KEY (`id`)  
) ENGINE = InnoDB;
```

Kemudian kita coba isi dengan sebuah data, klik Insert.



Silahkan isi seperti berikut:

The screenshot shows a database management interface for a table named 'calon_siswa' in a database named 'pendaftaran_siswa'. The table has the following columns: id (int(11)), nama (varchar(64)), alamat (varchar(255)), jenis_kelamin (varchar(16)), agama (varchar(16)), and sekolah_asal (varchar(64)). The data entry form is filled with the following values: id is empty, nama is 'Lia', alamat is 'Jl. Mangga No. 3, Mataram', jenis_kelamin is 'perempuan', agama is 'islam', and sekolah_asal is 'SMPN 32 Ampenan'. A red arrow points to a 'Go' button at the bottom right of the form.

Column	Type	Function	Null	Value
id	int(11)			
nama	varchar(64)			Lia
alamat	varchar(255)			Jl. Mangga No. 3, Mataram
jenis_kelamin	varchar(16)			perempuan
agama	varchar(16)			islam
sekolah_asal	varchar(64)			SMPN 32 Ampenan

Klik Go untuk menyimpan.

```
INSERT INTO `calon_siswa` (`id`, `nama`, `alamat`,  
`jenis_kelamin`, `agama`, `sekolah_asal`) VALUES (NULL, 'Lia',  
'Jl. Mangga No. 3, Mataram', 'perempuan', 'islam', 'SMPN 32  
Ampenan');
```

2. Membuat Proyek Baru

Buat folder baru dengan nama **pendaftaran-siswa** dalam direktori **htdocs** pada web server XAMPP Anda. Selanjutnya buatlah 8 buah file kosong dalam folder tersebut dengan ekstensi **.php**, dengan nama file sebagai berikut:

1. **config.php** — untuk menyimpan konfigurasi database;
2. **index.php** — halaman utama;
3. **list-siswa.php** — halaman untuk menampilkan data siswa;
4. **form-daftar.php** — halaman formulir pendaftaran;
5. **proses-pendaftaran.php** — yang memproses pendaftaran;
6. **form-edit.php** — form untuk edit data siswa;
7. **proses-edit.php** — skrip untuk memproses edit/update;
8. **hapus.php** — skrip untuk menghapus data dari database;

Selanjutnya akan dilakukan pembuatan aplikasi Pendaftaran Siswa Baru dengan tahapan pembuatan file di atas secara berurutan.

2.1. Cara Membuat Koneksi ke Dabase MySQL

Buka file `config.php`, silahkan isi dengan kode berikut:

```
<?php

$server = "localhost";
$user = "root";
$password = "kopi";
$nama_database = "pendaftaran_siswa";

$db = mysqli_connect($server, $user, $password, $nama_database);

if( !$db ){
    die("Gagal terhubung dengan database: " .
    mysqli_connect_error());
}

?>
```

Untuk menghubungkan PHP dengan MySQL, kita menggunakan fungsi `mysqli_connect()` dengan parameter alamat server, user, password, dan nama database.

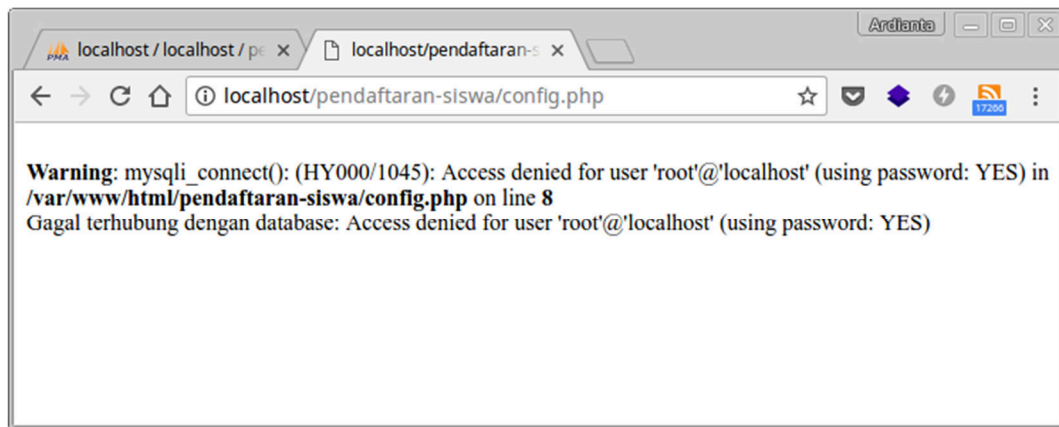
Silahkan isi password sesuai dengan yang dibuat di servernya. Jika tidak menggunakan password, maka dikosongkan saja.

```
$password = "";
```

Jika koneksi gagal, variabel `$db` akan bernilai `false`. Karena itu, kita bisa mengeceknya dengan ekspresi `if`.

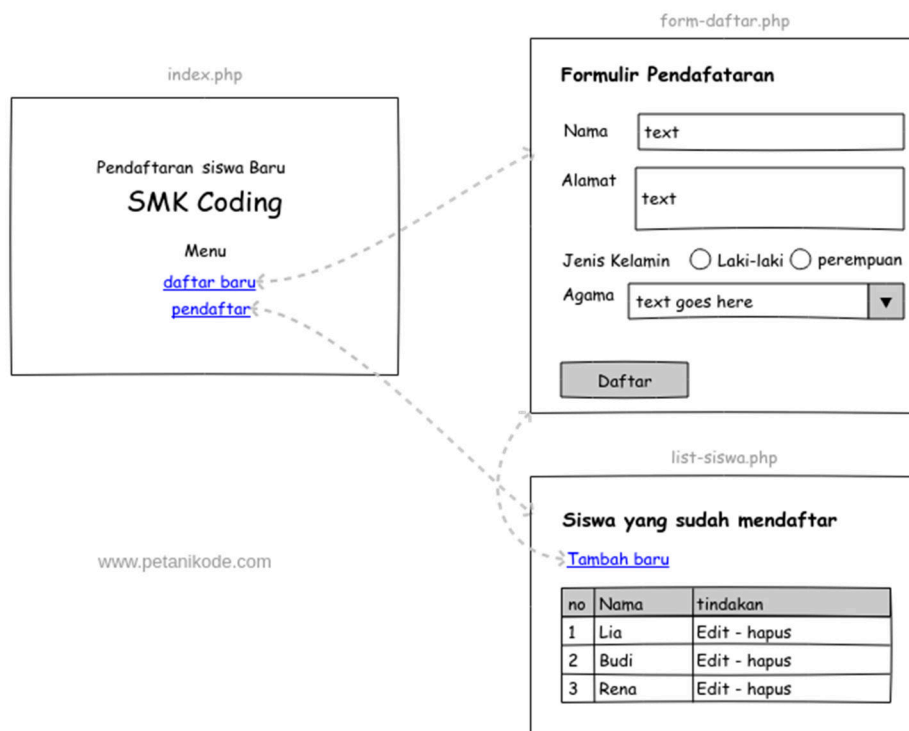
```
if( !$db ){
    die("Gagal terhubung dengan database: " .
    mysqli_connect_error());
}
```

Berikut ini tampilan saat koneksinya gagal, karena password-nya salah.



2.2. Membuat Halaman Indeks

Rencananya kita akan menaruh dua link di halaman indeks. Link pertama untuk menuju ke formulir pendaftaran dan link yang kedua untuk menuju ke halaman data siswa yang sudah mendaftar. Ilustrasi cara kerja aplikasi PSB seperti gambar berikut:



Sekarang, silahkan isi file **index.php** dengan kode berikut:

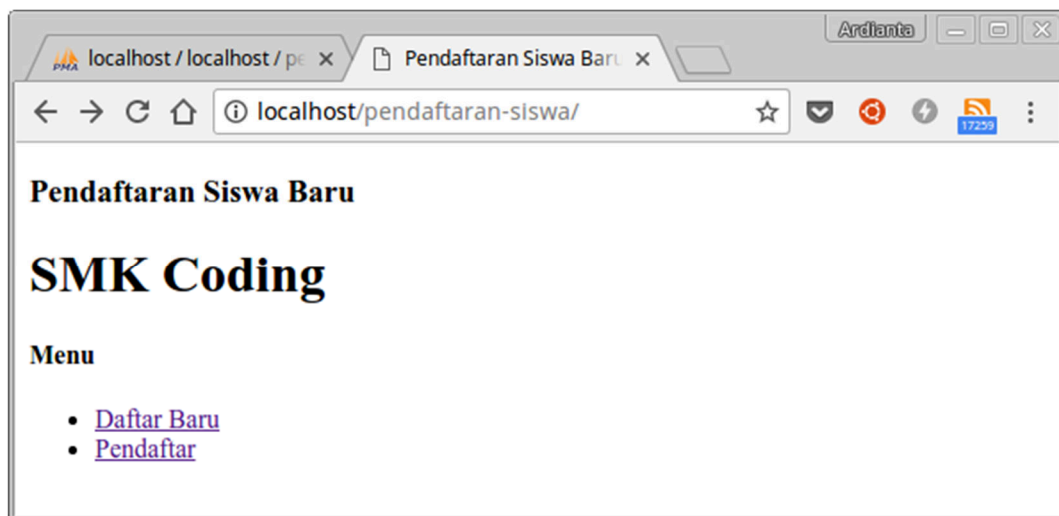
```
<!DOCTYPE html>
<html>
<head>
  <title>Pendaftaran Siswa Baru | SMK Coding</title>
</head>

<body>
  <header>
    <h3>Pendaftaran Siswa Baru</h3>
    <h1>SMK Coding</h1>
  </header>

  <h4>Menu</h4>
  <nav>
    <ul>
      <li><a href="form-daftar.php">Daftar Baru</a></li>
      <li><a href="list-siswa.php">Pendaftar</a></li>
    </ul>
  </nav>

</body>
</html>
```

Tampilan awalnya akan menjadi seperti ini:



2.3. Cara Menampilkan Data Tabel MySQL

Silahkan buka **list-siswa.php** dan isi dengan kode berikut:

```
<?php include("config.php"); ?>

<!DOCTYPE html>
<html>
<head>
  <title>Pendaftaran Siswa Baru | SMK Coding</title>
</head>
```

```

<body>
    <header>
        <h3>Siswa yang sudah mendaftar</h3>
    </header>

    <nav>
        <a href="form-daftar.php">[+] Tambah Baru</a>
    </nav>

    <br>

    <table border="1">
    <thead>
        <tr>
            <th>No</th>
            <th>Nama</th>
            <th>Alamat</th>
            <th>Jenis Kelamin</th>
            <th>Agama</th>
            <th>Sekolah Asal</th>
            <th>Tindakan</th>
        </tr>
    </thead>
    <tbody>

        <?php
        $sql = "SELECT * FROM calon_siswa";
        $query = mysqli_query($db, $sql);

        while($siswa = mysqli_fetch_array($query)){
            echo "<tr>";

            echo "<td>".$siswa['id']. "</td>";
            echo "<td>".$siswa['nama']. "</td>";
            echo "<td>".$siswa['alamat']. "</td>";
            echo "<td>".$siswa['jenis_kelamin']. "</td>";
            echo "<td>".$siswa['agama']. "</td>";
            echo "<td>".$siswa['sekolah_asal']. "</td>";

            echo "<td>";
            echo "<a href='form-";
edit.php?id=".$siswa['id']."'>Edit</a> | ";
            echo "<a";
href='hapus.php?id=".$siswa['id']."'>Hapus</a>";
            echo "</td>";

            echo "</tr>";

        }
        ?>

    </tbody>
    </table>

    <p>Total: <?php echo mysqli_num_rows($query) ?></p>

```



```
</body>
</html>
```

Pertama kita membutuhkan koneksi ke database, maka kita harus mengimpor file `config.php` agar variabel `$db` dapat dibaca.

```
<?php include("config.php"); ?>
```

Setelah itu, kita melakukan query untuk mengambil data siswa pada tabel `calon_siswa`.

```
$sql = "SELECT * FROM calon_siswa";
$query = mysqli_query($db, $sql);
```

Hasil dari query tersebut, kemudian kita ubah menjadi array dengan fungsi `mysqli_fetch_array()` dan hasilnya disimpan dalam variabel `$siswa`.

```
$siswa = mysqli_fetch_array($query)
```

Perulangan `while` akan mengulang selama masih ada isi dalam tabel dan selama perulangan datanya ditampilkan.

```
while($siswa = mysqli_fetch_array($query)) {
    echo "<tr>";

    echo "<td>".$siswa['id']. "</td>";
    echo "<td>".$siswa['nama']. "</td>";
    echo "<td>".$siswa['alamat']. "</td>";
    echo "<td>".$siswa['jenis_kelamin']. "</td>";
    echo "<td>".$siswa['agama']. "</td>";
    echo "<td>".$siswa['sekolah_asal']. "</td>";

    echo "<td>";
    echo "<a href='form-edit.php?id=".$siswa['id']."'>Edit</a> | ";
    echo "<a href='hapus.php?id=".$siswa['id']."'>Hapus</a>";
    echo "</td>";

    echo "</tr>";
}
```

Perhatikan array `$siswa`, indeksnya sesuai dengan nama kolom pada database dan bersifat *case sensitive*.

Terakhir, kita menampilkan total data yang ada di dalam tabel dengan fungsi `mysqli_num_rows()`.

```
echo mysqli_num_rows($query);
```

Maka hasilnya sebagai berikut:

Siswa yang sudah mendaftar

[\[+\] Tambah Baru](#)

No	Nama	Alamat	Jenis Kelamin	Agama	Sekolah Asal	Tindakan
1	Lia	Jl. Mangga No. 3, Mataram	perempuan	islam	SMPN 32 Ampenan	Edit Hapus

Total: 1