

Flask-CRUD

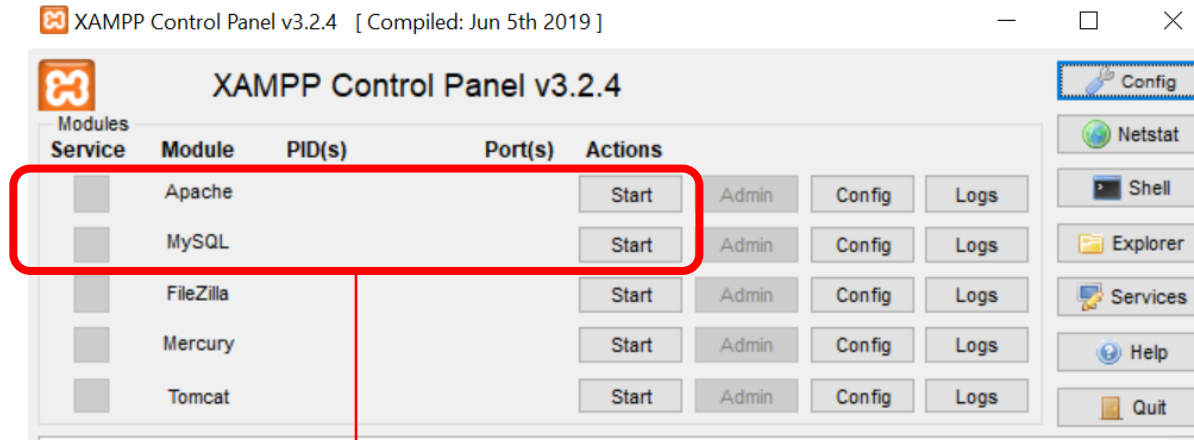
Create, Read, Update, Delete

Pre-requisite

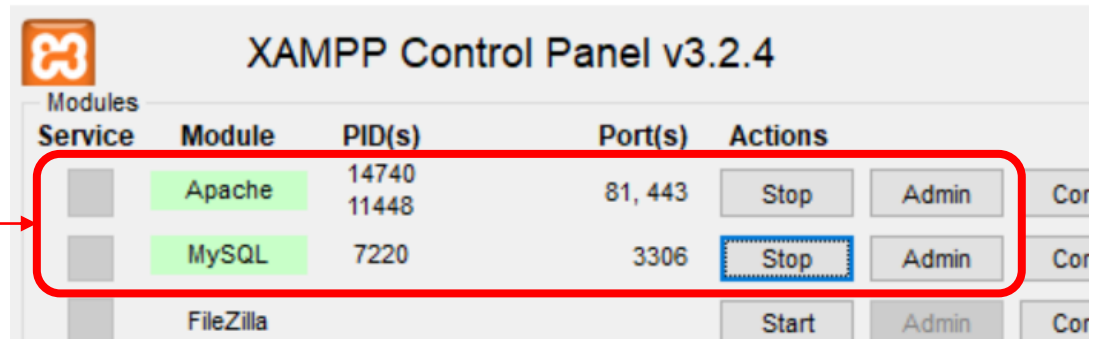
- SQL Engine. [Services: active]
- Database used. [**db_kuliah.sql**]
- SQL Connection
- Flask Installation

Mengaktifkan SQL

- Buka XAMPP Control Panel
- Aktifkan Service Apache dan Mysql.

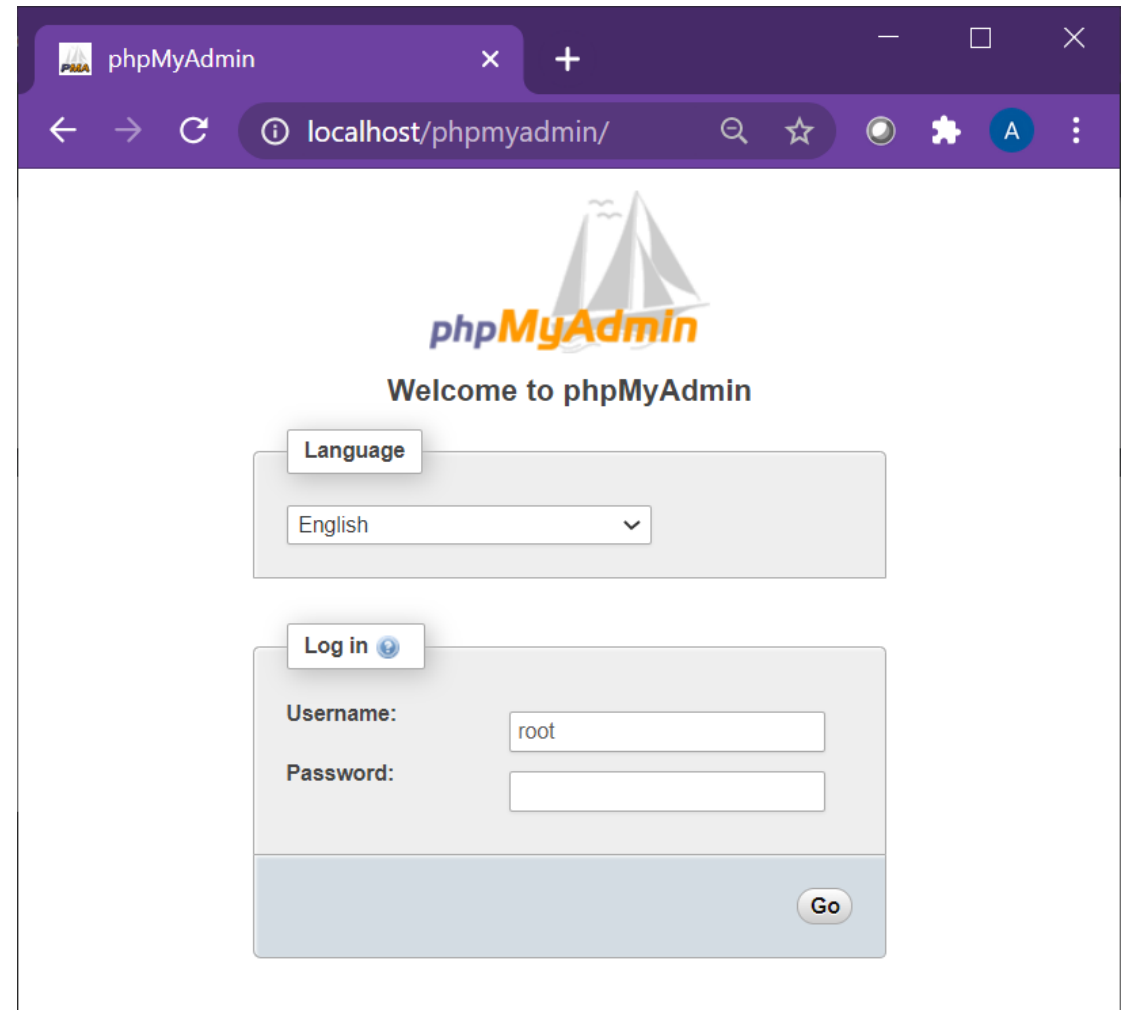


XAMPP Control Panel v3.2.4 [Compiled: Jun 5th 2019]



Mengakses MySQL

- Buka browser
- Pada address bar ketikkan alamat url: <http://localhost/phpmyadmin>
- Default account
 - Username : root
 - Password : (no passwd)



Tampilan Awal Halaman PhpMyAdmin

The screenshot displays the PhpMyAdmin web interface in a browser window. The address bar shows the URL `localhost/phpmyadmin/`. The interface features a top navigation bar with tabs for `Databases`, `SQL`, `Status`, `User accounts`, `Export`, `Import`, `Settings`, `Replication`, `Variables`, `Charsets`, `Engines`, and `Plugins`. On the left, a sidebar contains a tree view of databases and schemas, including `18.83.1234`, `db_movies`, `information_schema`, `mysql`, `performance_schema`, `phpmyadmin`, and `test`. The main content area is divided into several panels:

- General settings:** Includes a dropdown for `Server connection collation` set to `utf8mb4_unicode_ci`.
- Appearance settings:** Includes a dropdown for `Language` set to `English`, a dropdown for `Theme` set to `pmahomme`, and a dropdown for `Font size` set to `82%`. A `More settings` link is also present.
- Database server:** Lists server details:
 - Server: 127.0.0.1 via TCP/IP
 - Server type: MariaDB
 - Server connection: SSL is not being used
 - Server version: 10.4.6-MariaDB - mariadb.org binary distribution
 - Protocol version: 10
 - User: root@localhost
 - Server charset: cp1252 West European (latin1)
- Web server:** Lists web server details:
 - Apache/2.4.39 (Win64) OpenSSL/1.1.1c PHP/7.3.8
 - Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - \$Id: 7cc7cc96e675f6d72e5cf0f267f48e167c2abb23 \$
 - PHP extension: mysqli, curl, mbstring
 - PHP version: 7.3.8
- phpMyAdmin:** Lists version information and links:
 - Version information: 4.9.0.1, latest stable version: 5.0.4
 - Documentation
 - Official Homepage
 - Contribute
 - Get support
 - List of changes
 - License

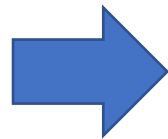
A notification banner at the bottom states: `A newer version of phpMyAdmin is available and you should consider upgrading. The newest version is 5.0.4, released on 2020-10-15.`






Import SQL

- Siapkan database yang akan digunakan.
 - Buat database dengan nama **db_kuliah**.
 - Import database **db_kuliah.sql**.













db_kuliah.sql




	Database	Collation	Action
<input type="checkbox"/>	1234_prodi	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/>	1234_reviewpbdtk01	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/>	db_cucimotor	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/>	db_kuliah	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/>	db_prodi	utf8mb4_general_ci	 Check privileges

Import SQL

- Pastikan database db_kuliah memiliki struktur table berikut:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	nim 	varchar(15)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	2	nama	varchar(50)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	3	asal	varchar(30)	latin1_swedish_ci		No	None			 Change  Drop  More

- Dan data sebagai berikut:

nim	nama	asal 
18.83.1233	Salsabila	Bantul
18.83.1234	Rahmadi	Sleman
18.83.1235	Sukarwo	Kulon Progo
18.83.1236	Pambudi	Kulon Progo
18.83.1237	Putri Amalia	Yogyakarta

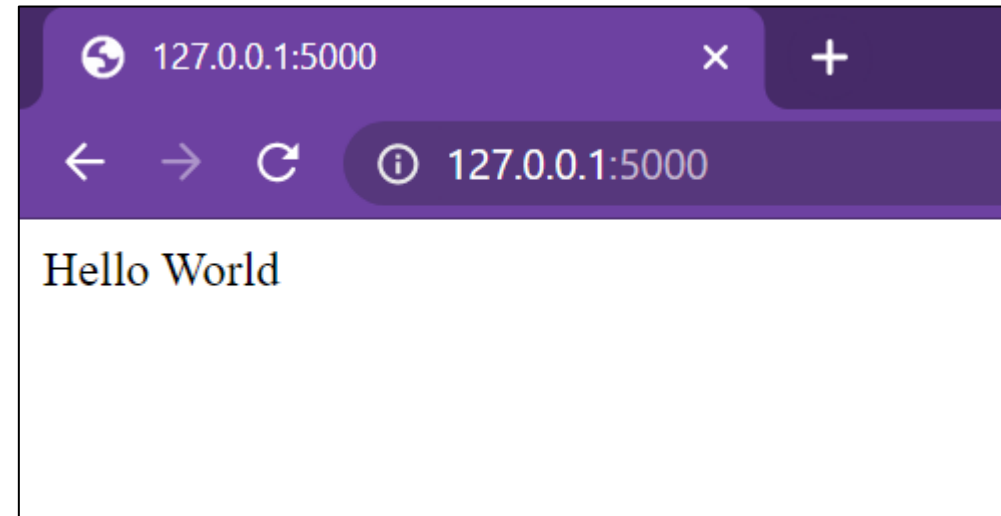
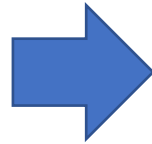
Flask Installation

- pastikan Flask telah terinstall dengan baik, sehingga Ketika servicenya diaktifkan menggunakan perintah:

flask run

- Maka Flask dapat menampilkan **Hello World** pada Browser.

```
1  from flask import Flask
2
3  app = Flask(__name__)
4
5  @app.route('/')
6  def index():
7      return 'Hello World'
8
9  if __name__ == '__main__':
10     app.run()
11
```



Open Connection SQL - Flask

- Pastikan Package **mysql-connector** telah terinstall.

```
pip install mysql-connector
```



Pastikan menginstal mysql-connector
pada virtualenvironment

```
(flaskCrud) C:\Users\Win-10 Presensi\Documents\FlaskCRUD>pip install mysql-connector
Collecting mysql-connector
  Downloading mysql-connector-2.2.9.tar.gz (11.9 MB)
    |████████████████████████████████████████| 11.9 MB 79 kB/s
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: mysql-connector
  Building wheel for mysql-connector (setup.py) ... /
```

Open Connection SQL - Flask

```
1  from flask import Flask
2
3  from mysql import connector
4
5  app = Flask(__name__)
6
7  #open connection
8  db = connector.connect(
9      host      = 'localhost',
10     user      = 'root',
11     passwd    = '',
12     database= 'db_kuliah'
13 )
14
15 if db.is_connected():
16     print('open connection successful')
17
18 @app.route('/')
19 def index():
20     return 'Hello World'
21
22 if __name__ == '__main__':
23     app.run()
```

Import library mysql-connector

Melakukan **open connection**, menggunakan perintah **connector.connect** diikuti dengan informasi mengenai **host**, **user**, **password** untuk mengakses mysql. Serta **nama database** yang akan digunakan pada web apps.

Untuk mengetahui keberhasilan proses **open connection**.

```
(flaskCrud) C:\Users\Win-10 Presensi\Documents\FlaskCRUD>flask run
* Serving Flask app 'app.py' (lazy loading)
* Environment: development
* Debug mode: on
open connection successful
* Restarting with stat
open connection successful
* Debugger is active!
* Debugger PIN: 940-927-594
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Flask CRUD

- Goal : Dashboard [Administration Page]

Halaman Data Mahasiswa x Page Tittle

127.0.0.1:5000

Data Mahasiswa Teknik Komputer

Tambah Data

Content Title **Main Content**

NIM	Nama	Asal	Aksi
18.83.1233	Salsabila	Bantul	<button>Ubah</button> <button>Hapus</button>
18.83.1234	Rahmadi	Sleman	<button>Ubah</button> <button>Hapus</button>
18.83.1235	Sukarwo	Kulon Progo	<button>Ubah</button> <button>Hapus</button>
18.83.1236	Pambudi	Kulon Progo	<button>Ubah</button> <button>Hapus</button>
18.83.1237	Putri Amalia	Yogyakarta	<button>Ubah</button> <button>Hapus</button>

Halaman Tambah Data Mahasiswa x

127.0.0.1:5000/tambah/

Data Mahasiswa Teknik Komputer

Tambah Data Mahasiswa

No Induk Mahasiswa

nim

Nama Lengkap

nama

Asal

kota asal

Cancel Submit

Halaman Ubah Data Mahasiswa x

127.0.0.1:5000/ubah/18.83.1233

Data Mahasiswa Teknik Komputer

Ubah Data Mahasiswa

No Induk Mahasiswa

18.83.1233

Nama Lengkap

Salsabila

Asal

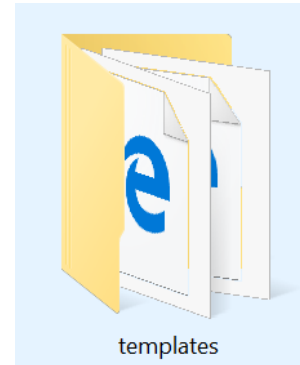
Bantul

Cancel Submit

Flask CRUD

- Administration Page - 4 File HTML

- base.html
- index.html
- tambah.html
- ubah.html



- Karena akan mengakses file html, maka perlu mempersiapkan terlebih dahulu directory **templates**, dan meletakkan ke-4 file html [*base.html*, *index.html*, *tambah.html*, dan *ubah.html*] pada directory tersebut
- Untuk menampilkan hasilnya, perlu menggunakan fungsi `render_template` sehingga perlu import fungsi tersebut pada main file [*app.py*]

```
from flask import render_template
```

Flask CRUD

- base.html

```
1  <!DOCTYPE html>
2  ▼ <html lang="en">
3  ▼ <head>
4      <meta charset="UTF-8">
5      <!-- title area -->
6      <title> Halaman {% block judul%} {% endblock %} </title>
7
8      <!-- CSS area -->
9      <link rel="stylesheet" type="text/css" href="{{url_for('static',
10      filename='css/bootstrap.min.css')}}">
11      <link rel="stylesheet" type="text/css" href="{{url_for('static',
12      filename='css/AdminLTE.css')}}">
13
14      <!-- CSS font area -->
15      <link rel="stylesheet" type="text/css" href="{{url_for('static',
16      filename='font/sansPro.css')}}">
17  </head>
```

Page Tittle

Flask CRUD

- base.html

```
16 ▼ <body>
17   <!-- content area -->
18 ▼   <div class="content-wrapper">
19 ▼     <section class="content-header">
20 ▼       <h1>
21 ▼         <a href="/">
22           
24         </a>
25         Data Mahasiswa
26         <small>Teknik Komputer</small>
27       </h1>
28     </section>
```

Flask CRUD

- base.html

```
29 ▼      <section class="content">
30 ▼          <div class="row">
31 ▼              <div class="col-xs-12">
32 ▼                  <div class="box">
33 ▼                      <div class="box-header">
34                          {% block JudulKonten %} {% endblock %}
35                      </div>
36
37                      <!-- data table (database) area -->
38 ▼                      <div class="box-body">
39                          {% block konten %} {% endblock %}
40                      </div>
41                  </div>
42              </div>
43          </div>
44      </section>
45  </div>
46
```

Content Tittle

Main Content

Flask CRUD

- base.html

```
47  <!-- Javascript area -->
48  <script src="{{url_for('static', filename='js/bootstrap.min.js')}}"></script>
49  <script src="{{url_for('static', filename='js/adminlte.min.js')}}"></script>
50
51  </body>
52  </html>
```


index.html

```
1 {% extends 'base.html' %}
2
3 {% block judul %} Data Mahasiswa {% endblock %}
4
5 {% block JudulKonten %}
6     <a href="/tambah/" class="btn btn-info"> Tambah Data </a>
7 {% endblock %}
8
9 {% block konten %}
10     <table class="table table-bordered table-hover">
11         <thead>
12             <tr>
13                 <th>NIM</th>
14                 <th>Nama</th>
15                 <th>Asal</th>
16                 <th>Aksi</th>
17             </tr>
18         </thead>
19         <tbody>
20             {% for row in hasil %}
21                 <tr>
22                     <td>{{row.0}}</td>
23                     <td>{{row.1}}</td>
24                     <td>{{row.2}}</td>
25                     <td>
26                         <a href="/ubah/{{row.0}}" class="btn btn-warning"> Ubah </a>
27                         <a href="/hapus/{{row.0}}" class="btn btn-danger"
28                             onclick="return confirm('yakin akan menghapus {{row.1}} ?')"> Hapus </a>
29                     </td>
30                 </tr>
31             {% endfor %}
32         </tbody>
33     </table>
34 {% endblock %}
```

Flask CRUD

- app.py

Menampilkan data
dari Database

```
1  from flask import Flask, render_template
2  from flask import request, redirect, url_for
3  from mysql import connector
4
5  app = Flask(__name__)
6
7  ▼ db = connector.connect(
8      host      = "localhost",
9      user      = "root",
10     passwd    = "",
11     database  = "db_kuliah"
12 )
13 ▼ if db.is_connected():
14     print("Berhasil Terhubung ke Database")
15
16 @app.route('/')
17 ▼ def halaman_awal():
18     cur = db.cursor()
19     cur.execute("select * from mahasiswa")
20     res = cur.fetchall()
21     cur.close()
22     return render_template('index.html', hasil=res)
23
24 if __name__ == '__main__':
25     app.run()
```

Running Flask

- Jalankan flask untuk melihat hasilnya

Tambah Data ke Database – [Insert Data]

- Buat routing di **app.py** untuk mengakses halaman Tambah Data.

```
@app.route('/tambah/')  
def tambah_data():  
    return render_template('tambah.html')
```

- Pastikan pada folder **templates** sudah ada file html dengan nama **tambah.html**

tambah.html

```
1  {% extends 'base.html' %}
2
3  {% block judul %} Tambah Data Mahasiswa {% endblock %}
4
5  {% block JudulKonten %}
6      <h3 class="box-title"> Tambah Data Mahasiswa </h3>
7  {% endblock %}
8
9  {% block konten%}
10     <form action="{url_for('proses_tambah')}}" method="POST">
11     <div class="form-group">
12         <label>No Induk Mahasiswa</label>
13         <input type="text" name="nim" class="form-control" placeholder="nim">
14     </div>
15     <div class="form-group">
16         <label>Nama Lengkap</label>
17         <input type="text" name="nama" class="form-control" placeholder="nama">
18     </div>
19     <div class="form-group">
20         <label>Asal</label>
21         <input type="text" name="asal" class="form-control" placeholder="kota asal">
22     </div>
23     <div class="box-footer">
24         <button type="reset" class="btn btn-default"> Cancel </button>
25         <button type="submit" class="btn btn-primary pull-right"> Submit </button>
26     </div>
27 </form>
28 {% endblock %}
```

Proses Tambah Data

- Import **request** dan **redirect** dan **url_for** pada **app.py**

```
from flask import request, redirect, url_for
```

- Buat **routing** untuk proses ketika data akan ditambahkan ke database melalui app.py.

```
@app.route('/proses_tambah/', methods=['POST'])
def proses_tambah():
    nim = request.form['nim']
    nama = request.form['nama']
    asal = request.form['asal']
    cur = db.cursor()
    cur.execute('INSERT INTO mahasiswa (nim, nama, asal) VALUES (%s, %s, %s)', (nim, nama, asal))
    db.commit()
    return redirect(url_for('halaman_awal'))
```

Running Flask

- Jalankan flask untuk melihat hasilnya

Mengubah Data pada Database – [Update Data]

- Buat routing pada **app.py** untuk mengarahkan ke halaman ubah data dan pemrosesan ubah data.

```
@app.route('/ubah/<nim>', methods=['GET'])
def ubah_data(nim):
    cur = db.cursor()
    cur.execute('select * from mahasiswa where nim=%s', (nim,))
    res = cur.fetchall()
    cur.close()
    return render_template('ubah.html', hasil=res)
```


Mengubah Data pada Database – [Update Data]

- Buat halaman **ubah.html** pada folder **templates** yang digunakan untuk menampilkan data yang akan diubah.

```
1  {% extends 'base.html' %}
2
3  {% block judul %} Ubah Data Mahasiswa {% endblock %}
4
5  {% block JudulKonten %}
6  <h3 class="box-title"> Ubah Data Mahasiswa </h3>
7  {% endblock %}
```

```
9  {% block konten%}
10 ▼ <form action="{{url_for('proses_ubah')}}" method="POST">
11     {% for row in hasil %}
12 ▼     <div class="form-group">
13         <label>No Induk Mahasiswa</label>
14         <input type="hidden" name="nim_ori" value="{{ row.0 }}">
15         <input type="text" name="nim" class="form-control" placeholder="nim" value="{{row.0}}">
16     </div>
17 ▼     <div class="form-group">
18         <label>Nama Lengkap</label>
19         <input type="text" name="nama" class="form-control" placeholder="nama" value="{{row.1}}">
20     </div>
21 ▼     <div class="form-group">
22         <label>Asal</label>
23         <input type="text" name="asal" class="form-control" placeholder="kota asal" value="{{row.2}}">
24     </div>
25     {% endfor %}
26
27 ▼     <div class="box-footer">
28         <button type="reset" class="btn btn-default"> Cancel </button>
29         <button type="submit" class="btn btn-primary pull-right"> Submit </button>
30     </div>
31 </form>
32 {% endblock %}
```

Proses ubah Data

```
@app.route('/proses_ubah/', methods=['POST'])
def proses_ubah():
    no_mhs = request.form['nim_ori']
    nim = request.form['nim']
    nama = request.form['nama']
    asal = request.form['asal']
    cur = db.cursor()
    sql = "UPDATE mahasiswa SET nim=%s, nama=%s, asal=%s WHERE nim=%s"
    value = (nim, nama, asal, no_mhs)
    cur.execute(sql, value)
    db.commit()
    return redirect(url_for('halaman_awal'))
```

app.py

Running Flask

- Jalankan flask untuk melihat hasilnya

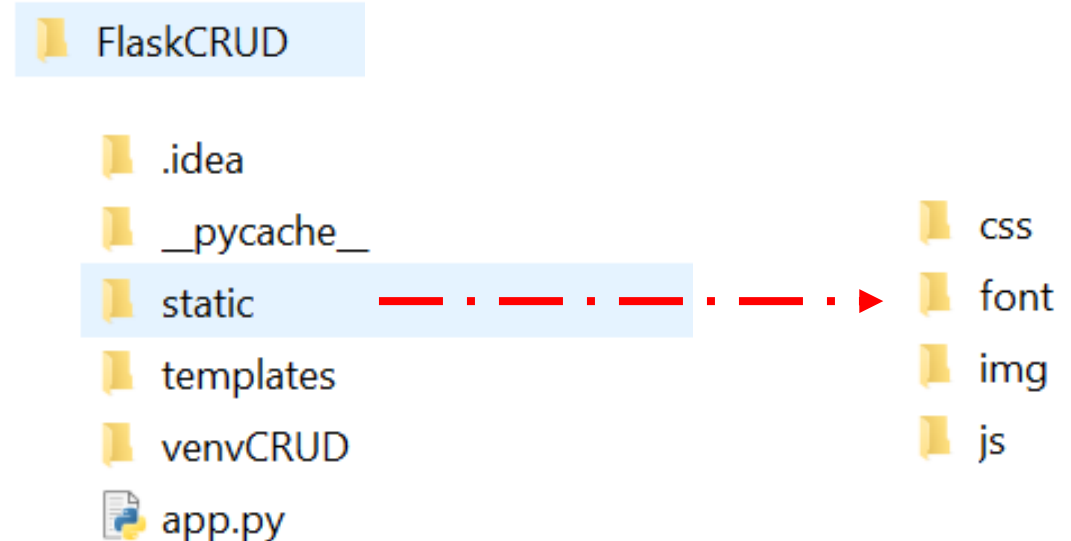
Menghapus Data di Database – [Delete Data]

- Buat **URL routing** pada **app.py** yang digunakan untuk mengarahkan ke proses hapus data

```
@app.route('/hapus/<nim>', methods=['GET'])
def hapus_data(nim):
    cur = db.cursor()
    cur.execute('DELETE from mahasiswa where nim=%s', (nim,))
    db.commit()
    return redirect(url_for('halaman_awal'))
```

Mempercantik Tampilan Web

- Buat directory dengan nama **static**.
- Copy file **css**, **js**, **img** dan **font** yang ada pada directory assets ke folder **static**.



Mempercantik Tampilan Web

- Pada file **base.html** kita menggunakan file **CSS**, **font**, **JS** serta **Image**.

```
<!-- CSS area -->
<link rel="stylesheet" type="text/css" href="{{url_for('static', filename='css/bootstrap.min.css')}}">
<link rel="stylesheet" type="text/css" href="{{url_for('static', filename='css/AdminLTE.css')}}">

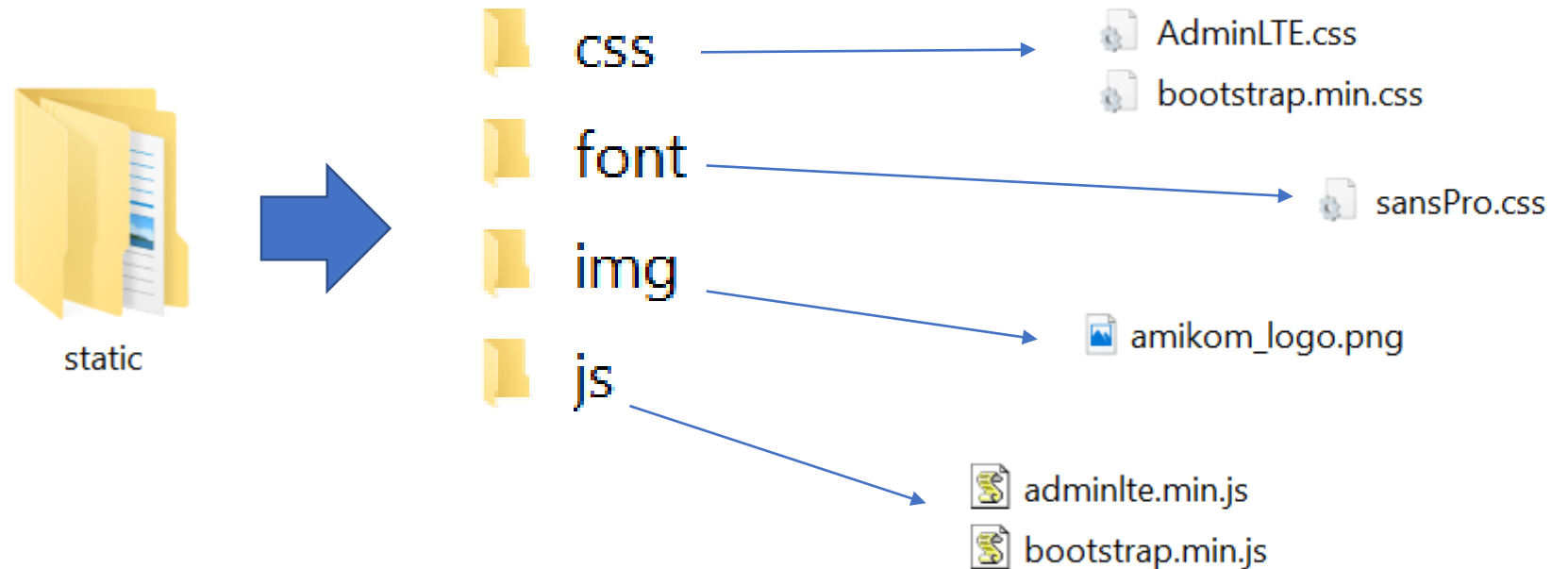
<!-- CSS font area -->
<link rel="stylesheet" type="text/css" href="{{url_for('static', filename='font/sansPro.css')}}">
```

```
<a href="/">
    
</a>
```

```
<!-- Javascript area -->
<script src="{{url_for('static', filename='js/bootstrap.min.js')}}"></script>
<script src="{{url_for('static', filename='js/adminlte.min.js')}}"></script>
```

Mempercantik Tampilan Web

- Pada file **base.html** kita menggunakan file **CSS**, **font**, **JS** serta **Image**.
- Maka letakkan file yang kita gunakan pada folder static, sesuai tempatnya masing-masing.



Running Flask

- Jalankan flask untuk melihat hasilnya