

MI3103  
Praktikum Antar Muka Komputer  
Pengenalan Flask: Python *Web Framework*

Fadjar Fathurrahman

2018

## 1 Tujuan

- Dapat merancang website sederhana dengan menggunakan Flask

## 2 Perangkat lunak yang diperlukan

- Linux OS
- Distribusi Anaconda untuk Python 3
- Browser
- Editor teks seperti gedit, VSCode, Atom

## 3 Flask

Pada bagian ini, kita akan mempelajari Flask, yang merupakan suatu framework dalam bahasa Python yang dapat digunakan untuk membuat website dinamik. Flask sudah menyediakan web server built-in yang dapat digunakan pada saat pengembangan web.

## 4 Contoh aplikasi sederhana

Buat sebuah direktori untuk menyimpan pekerjaan Anda, misalkan dengan nama ContohFlask. Di dalam direktori ini buatlah file dengan nama ContohFlask.py. Tuliskan kode berikut ini dalam file tersebut.

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello ..."
```

Setelah itu, bukalah terminal Anda pada direktori ContohFlask (dengan menggunakan perintah `cd` yang sesuai). Kemudian ketiklah perintah berikut ini pada terminal

```
export FLASK_APP=ContohFlask.py
flask run
```

**Catatan:** Anda mungkin perlu menggunakan *full path* jika Anda tidak menambahkan direktori yang berisi perintah flask, misalnya:

```
/home/students/anaconda3/bin/flask run
```

Anda akan melihat keluaran yang mirip seperti berikut ini.

```
* Serving Flask app "ContohFlask.py"
* Environment: production
  WARNING: Do not use the development server in a production environment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Dengan menggunakan internet browser, ketikkan URL `http://127.0.0.1:5000/` pada isian lokasi URL.

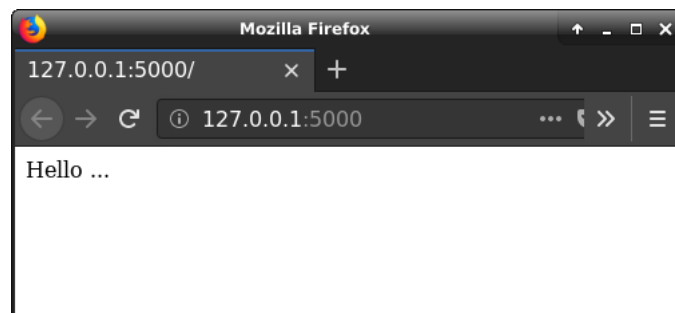


Figure 1: Tampilan ContohFlask

## 4.1 Penjelasan kode

```
from flask import Flask
```

Kode ini akan memuat fungsi Flask pada modul flask. Fungsi Flask digunakan untuk inisialisasi objek app yang merupakan pengontrol dari website yang akan kita buat. Inisialisasi objek app dilakukan pada kode berikut ini.

```
app = Flask(__name__)
```

Setelah app diinisialisasi, kita perlu membuat fungsi-fungsi yang akan merespon pada jika ada request yang datang pada webserver. Pada kode sebelumnya, fungsi ini bernama hello yang akan merespon jika URL yang diminta adalah root ("/"). Bentuk dasar dari fungsi ini adalah:

```
@app.route("/link-name")
def nama_fungsi():
    # berikan respon yang sesuai di sini.
```

Fungsi inilah yang akan merespon jika ada webserver diakses dengan format `http://alamat_server:nomor_port/link-name`. Secara default nama `alamat_server:nomor_port` yang digunakan pada Flask adalah `127.0.0.1:5000`

Sebagai contoh yang kita gunakan sebelumnya.

```
@app.route("/")
def hello():
    return "Hello ..."
```

## 4.2 Latihan

Coba tambahkan fungsi berikut ini dalam file `ContohFlask.py`.

```
@app.route("/about-me")
def about_me():
    return """
    My name is Jono.<br>
    I am a student of MI3103"<br>
    I am learning web development using Flask.<br>
    """
```

Restart webserver Anda, kemudian akses `http://127.0.0.1:5000/about-me`. Anda seharusnya akan melihat tampilan seperti ini.

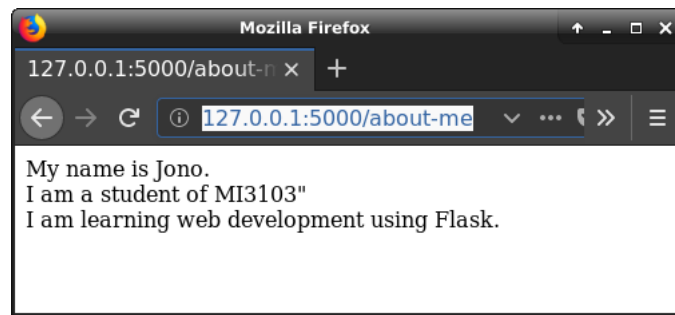


Figure 2: Tampilan ContohFlask

## 4.3 Agar dapat diakses dari komputer lain

Secara default web yang Anda buat hanya dapat diakses dalam komputer Anda sendiri. Agar dapat diakses dari komputer lain Anda harus menggunakan perintah berikut ini

```
flask run --host=0.0.0.0
```

atau (jika flask tidak ada dalam PATH)

```
/home/students/anaconda3/bin/flask run --host=0.0.0.0
```

Sekarang coba akses web Anda dari komputer lain dengan menggunakan browser. Dari komputer tersebut Anda harus mengganti 127.0.0.1 dengan alamat IP komputer Anda di mana webserver Flask dijalankan.

## 5 HTTP Client dari baris perintah

Internet browser bukanlah satu-satunya cara untuk mengakses website (HTTP server). Kita juga dapat berinteraksi dengan cara lain, misalnya dengan menggunakan program `wget` atau sejenisnya. Pada bagian ini kita akan menggunakan pustaka Python bernama `Requests` untuk berinteraksi dengan web server.

Buat file dengan nama `simple_request.py`. Ketikkan kode berikut ini pada file tersebut.

```

from __future__ import print_function
import requests
import sys
Nargs = len(sys.argv)
print("Nargs = ", Nargs)
if Nargs < 2:
    url_text = "http://localhost:5000/" # default pada Flask web server
else:
    url_text = sys.argv[1]
print("url_text = ", url_text)
r = requests.get(url_text)
print("Status = %s" % r.status_code)
print("Reason = %s" % r.reason)
print("HTML response follows\n\n")
print(r.text)

```

Pada terminal ketik perintah-perintah berikut:

```
python simple_request.py
```

jika Flask web server Anda sedang berjalan.

Coba juga

```
python simple_request.py http://tf.itb.ac.id
```

atau dengan alamat URL yang Anda inginkan.

**Tugas** Coba jelaskan kode Python yang baru saja Anda buat. Anda dapat menggunakan sumber berikut (hanya sebagai contoh, Anda dapat menggunakan sumber lainnya).

- <http://docs.python-requests.org/en/master/user/quickstart/>
- <https://www.pythonforbeginners.com/requests/the-awesome-requests-module>