

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Algoritma Dan Pemrograman 2B
Kelas : 1IA24
Praktikum Ke- : 5
Tanggal : Selasa, 11 2022
Materi : Pembuatan API endpoint dengan flask
NPM : 50422818
Nama : Lutfi Robbani
Ketua asisten : Gede Oke
Nama asisten :
Paraf asisten :
Jumlah Lembar : 10 Lembar



LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2022

LISTENING

act5.py > show_menus

```
1  from flask import Flask, request, render_template, redirect, url_for
2
3  app = Flask(__name__)
4
5  menus = []
6
7
8  def generate_id():
9      return max([menu["id"] for menu in menus]) + 1 if menus else 1
10
11 @app.route("/")
12 def show_menus():
13     return render_template("index.html", menus=menus)
14
15 @app.route("/add", methods=["GET", "POST"])
16 def add_menu():
17     if request.method == "POST":
18         name = request.form["name"]
19         price = int(request.form["price"])
20         menus.append({"id": generate_id(), "name": name, "price": price})
21         return redirect(url_for("show_menus"))
22     else:
23         return render_template("add.html")
24
25 @app.route("/edit/<int:id>", methods=["GET", "POST"])
26 def edit_menu(id):
27     menu = next((menu for menu in menus if menu["id"] == id), None)
28     if menu is None:
29         return redirect(url_for("show_menus"))
```

```

30     if request.method == "POST":
31         menu["name"] = request.form["name"]
32         menu["price"] = int(request.form["price"])
33         return redirect(url_for("show_menus"))
34     else:
35         return render_template("edit.html", menu=menu)
36
37 @app.route("/delete/<int:id>")
38 def delete_menu(id):
39     menu = next((menu for menu in menus if menu["id"] == id), None)
40     if menu is not None:
41         menus.remove(menu)
42     return redirect(url_for("show_menus"))
43
44 if __name__ == "__main__":
45     app.run(debug=True, port=8000)

```

```

1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <title>Edit Menu</title>
6      <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
7      <style>
8          body {
9              background-color: #ff00c8;
10         }
11
12         .container {
13             margin-top: 50px;
14         }
15     </style>
16 </head>
17
18 <body>
19     <div class="container">
20         <h1>Edit Menu</h1>
21         <form method="post">
22             <div class="form-group">
23                 <label for="name">Name</label>
24                 <input type="text" class="form-control" id="name" name="name" value="{{ menu.name }}"
25                     required>
26             </div>
27             <div class="form-group">
28                 <label for="price">Price</label>
29                 <input type="number" class="form-control" id="price" name="price" value="{{ menu.price }}"
30                     required>
31             </div>
32             <button type="submit" class="btn btn-primary">Save</button>
33             <a href="{{ url_for('show_menus') }}" class="btn btn-secondary">Cancel</a>
34         </form>
35     </div>
36 </body>
37
38 </html>

```

```

1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7      <link
8        rel="stylesheet"
9        href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
10       integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
11       crossorigin="anonymous"
12     />
13     <title>Add Menu</title>
14   </head>
15   <body>
16     <div class="container">
17       <h1>Add Menu</h1>
18       <form method="post">
19         <div class="form-group">
20           <label for="name">Name</label>
21           <input
22             type="text"
23             class="form-control"
24             id="name"
25             name="name"
26             required
27           />
28         </div>
29         <div class="form-group">
30           <label for="price">Price</label>
31           <input
32             type="text"
33             class="form-control"
34             id="price"
35             name="price"
36             required
37           />
38         </div>
39         <button type="submit" class="btn btn-primary">Add</button>
40         <a href="{url_for('show_menus')}}" class="btn btn-secondary">Cancel</a>
41       </form>
42     </div>
43   </body>
44 </html>
45

```

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Menu List</title>
5      <link
6        rel="stylesheet"
7        href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
8      />
9      <style>
10       body {
11         background-color: #6facc3;
12       }
13
14       .container {
15         margin-top: 50px;
16       }
17     </style>
18   </head>
19
20   <body>
21     <div class="container">
22       <h1>Menu List</h1>
23       <table class="table">
24         <thead>
25           <tr>
26             <th>ID</th>
27             <th>Name</th>
28             <th>Price</th>
29             <th></th>
30           </tr>
31         </thead>
32         <tbody>
33           {% for menu in menus %}
34           <tr>
35             <td>{{ menu.id }}</td>
36             <td>{{ menu.name }}</td>
37             <!-- <td>{{ menu.price }}</td> -->
38             <td>Rp {{ '{:,.0f}'.format(menu.price) }}</td>
39             <!-- <td><a href="{{ url_for('edit_menu', id=menu.id) }}" class="btn btn-warning">Edit</a></td>
40                  <td><a href="{{ url_for('delete_menu', id=menu.id) }}" class="btn btn-danger">Delete</a></td>
41             </td>
42             <td>
43               <a
44                 href="{{ url_for('edit_menu', id=menu.id) }}"
45                 class="btn btn-warning d-inline-block mr-2"
46               >Edit</a>
47               <a
48                 href="{{ url_for('delete_menu', id=menu.id) }}"
49                 class="btn btn-danger d-inline-block"
50               >Delete</a>
51             </td>
52             </td>
53           </tr>
54           {% endfor %}
55         </tbody>
56       </table>
57       <a href="{{ url_for('add_menu') }}" class="btn btn-primary">Add coy</a>
58     </div>
59   </body>
60 </html>

```

LOGIKA

act5.py > show_menus

```
1  from flask import Flask, request, render_template, redirect, url_for
2
3  app = Flask(__name__)
4
5  menus = []
```

Pertama tama import beberapa modul dari modul flask yaitu modul Flask, request, render_template, url_for.

Setelah itu buat instance aplikasi Flask. Ubah modul flask menjadi variabel (__name__), dan nilai tersebut simpan dalam variabel app. Kemudian buat list kosong dengan nama menu.

```
8  def generate_id():
9      return max([menu["id"] for menu in menus]) + 1 if menus else 1
```

Selanjutnya buat fungsi generate_id, lalu buat perintah jika list menus tidak kosong, maka akan mengambil nilai maksimum dari semua id menu yang sudah ada dan menambahkan 1 untuk mendapatkan id yang baru. jika list menus kosong, maka akan mengembalikan angka 1 sebagai id yang pertama kali dibuat.

```
11 @app.route("/")
12 def show_menus():
13     return render_template("index.html", menus=menus)
```

Kemudian tambahkan route ke aplikasi dengan menggunakan @app.route(). Route ini akan menentukan URL dan fungsi yang akan dijalankan ketika URL tersebut dipanggil.

Setelah itu buat fungsi show_menus(), didalam nya Kembali kan fungsi render_template dari kelas flask, lau render_template ini akan menerima 2 argumen yaitu index.html / memanggil nama template, dan menus / list yang berisi semua menu yang sudah ditambahkan ke dalam aplikasi.

```
15 @app.route("/add", methods=["GET", "POST"])
16 def add_menu():
17     if request.method == "POST":
18         name = request.form["name"]
19         price = int(request.form["price"])
20         menus.append({"id": generate_id(), "name": name, "price": price})
21         return redirect(url_for("show_menus"))
22     else:
23         return render_template("add.html")
```

Kemudian tambahkan route ke aplikasi dengan menggunakan @app.route(/add). Route ini akan menentukan URL dan fungsi yang akan dijalankan ketika URL tersebut dipanggil.

Tambahkan method juga di dalam nya yaitu method post, dan get. Kemudian buat fungsi didalam nya yaitu add_menu().

Jika metode HTTP adalah POST, maka fungsi akan mengambil data pengguna dari form HTML dengan memanggil request.form["name"] dan int(request.form["price"]) yang berbentuk interger. Kemudian,

fungsi akan memeriksa apakah username yang dimasukkan oleh pengguna sudah ada di dalam dictionary users. kemudian data akan ditambahkan ke list menus menggunakan fungsi append().

Setelah itu, program akan mengalihkan user kembali ke halaman utama menggunakan fungsi redirect() dan argumen url_for("show_menus"). Dan jika metode http adalah get atau hanya mengambil data dari server, html hanya akan merender templete add.html.

```
25 @app.route("/edit/<int:id>", methods=["GET", "POST"])
26 def edit_menu(id):
27     menu = next((menu for menu in menus if menu["id"]== id), None)
28     if menu is None:
29         return redirect(url_for("show_menus"))
30     if request.method == "POST":
31         menu["name"] = request.form["name"]
32         menu["price"] = int(request.form["price"])
33         return redirect(url_for("show_menus"))
34     else:
35         return render_template("edit.html", menu=menu)
36
```

Kemudian Ketika user mengakses halaman /edit/<int:id>, fungsi edit_menu(id) akan dipanggil dengan argumen id yang diberikan oleh user pada URL. Jika tidak ditemukan menu dengan id yang sesuai, maka pengguna akan diarahkan kembali ke halaman utama menggunakan fungsi redirect() dan argumen url_for("show_menus").

Jika method request adalah "POST", artinya user telah mengirimkan data melalui form pada halaman /edit/<int:id>. Program mengambil nilai name dan price dari form menggunakan request.form["name"] dan request.form["price"], dan kemudian mengupdate nilai name dan price pada menu yang sesuai di dalam list menus. Setelah itu, program mengalihkan user kembali ke halaman utama menggunakan fungsi redirect() dan argumen url_for("show_menus").

Dan jika metode http adalah get atau hanya mengambil data dari server, html hanya akan merender templete edit.html.

```
37 @app.route("/delete/<int:id>")
38 def delete_menu(id):
39     menu = next((menu for menu in menus if menu["id"] == id), None)
40     if menu is not None:
41         menus.remove(menu)
42     return redirect(url_for("show_menus"))
43
```

Kemudian tambahkan route ke aplikasi dengan menggunakan @app.route(/delete/<int:id>). Route ini akan menentukan URL dan fungsi yang akan dijalankan ketika URL tersebut dipanggil.

Tambahkan method juga di dalam nya yaitu method post, dan get. Kemudian buat fungsi didalam nya yaitu delete_menu() dengan parameter id.

Jika menu yang sesuai ditemukan di dalam list menus, maka program akan menghapus menu tersebut dari list menus. Setelah itu, program akan mengalihkan pengguna kembali ke halaman utama untuk menampilkan daftar menu terbaru menggunakan fungsi `redirect()` dan argumen `url_for("show_menus")`.

```
--  
44  if __name__ == "__main__":  
45      app.run(debug=True, port=8000)
```

Selanjutnya buat statement Ketika file python dijalankan, program akan mengevaluasi kondisi jika `__name__ == "__main__"` maka debug akan berjalan dan bernilai true dengan port 8000.

OUTPUT

Menu List

ID	Name	Price
----	------	-------

Add Menu

Add Menu

Name

Lutfi Robbani

Price

2000000

Add

Cancel

Menu List

ID	Name	Price	
1	Lutfi Robbani	Rp 2,000,000	<div>EditDelete</div>

Add Menu

Edit Menu

Name

Lutfi Robbani 2B

Price

2000000

Save

Cancel