

Kerjakan 3 problem di bawah ini !

Urutkan Data

0Deskripsi

Anda mendapatkan data produk dengan bentuk seperti di bawah ini (beserta 3 contoh).

nama	harga	rating	likes
Indomie	3000	5	150
Laptop	4000000	4.5	123
Aqua	3000	4	250
Smart TV	4000000	4.5	42
Headphone	4000000	3.5	90
Very Smart TV	4000000	3.5	87

Urutkan array di atas dengan prioritas sebagai berikut

1. harga terendah
2. jika harga sama maka urutkan berdasarkan rating tertinggi
3. jika rating sama maka urutkan berdasarkan likes terbanyak.

Pastikan program yang anda buat dapat menghandle sekitar ratusan data.

Aplikasi Data Produk Deskripsi

Anda diminta untuk membuat aplikasi sederhana daftar produk. Aplikasi tersebut dapat melakukan 5 operasi sebagai berikut

- Lihat daftar produk
- Lihat detail sebuah produk
- Buat Produk
- Edit Produk
- Hapus Produk

Aplikasi dapat dibuat menggunakan tools dan bahasa yang paling familier bagi anda

Kalkulator Sederhana

Deskripsi

Anda diminta untuk membuat fungsi kalkulator sederhana dengan input dalam bentuk string yang terdiri dari 3 bagian:

1. Operand 1 (angka maksimal 1 juta)
2. Operator (+, -, *, /)
3. Operand 2 (angka maksimal 1 juta)

Antar operand dan operator akan dipisahkan oleh spasi. Output dari aplikasi anda adalah hasil kalkulasi dari input yang diberikan.

Method signature dari fungsi yang dibuat adalah sebagai berikut

```
function kalkulator(string input) : integer
```

Contoh Input 1

13 + 187

Contoh Output 1

200

Contoh Input 2

134 - 11

Contoh Output 3

123

Contoh Input 1

8 * 7

Contoh Output 1

56

Contoh Input 1

16 / 4

Contoh Output 1

4

Prosedur Pengumpulan

1. Pengumpulan hasil test melalui google form <https://forms.gle/9h6JmY6vcdXPfEt5A>
2. Waktu pengerjaan soal test maksimal 7 hari

Jawaban

1. Jawaban

A. # array produk

```
produk = [  
    {'nama': 'Indomie', 'harga': 3000, 'rating': 5, 'likes': 150},  
    {'nama': 'Laptop', 'harga': 4000000, 'rating': 4.5, 'likes': 123},  
    {'nama': 'Aqua', 'harga': 3000, 'rating': 4, 'likes': 250},  
    {'nama': 'Smart TV', 'harga': 4000000, 'rating': 4.5, 'likes': 42},  
    {'nama': 'Headphone', 'harga': 4000000, 'rating': 3.5, 'likes': 90},  
    {'nama': 'Very Smart TV', 'harga': 4000000, 'rating': 3.5, 'likes': 87}  
]
```

fungsi untuk mengurutkan produk berdasarkan harga terendah

```
def sort_produk_by_harga_terendah(arr):  
    return sorted(arr, key=lambda x: x['harga'])
```

contoh penggunaan fungsi sort_produk_by_harga_terendah

```
sorted_produk = sort_produk_by_harga_terendah(produk)  
print(sorted_produk)
```

```
[ {'nama': 'Indomie', 'harga': 3000, 'rating': 5, 'likes': 150}, {'nama': 'Aqua', 'harga': 3000,  
'rating': 4, 'likes': 250}, {'nama': 'Laptop', 'harga': 4000000, 'rating': 4.5, 'likes': 123}, {'nama':  
'Smart TV', 'harga': 4000000, 'rating': 4.5, 'likes': 42}, {'nama': 'Headphone', 'harga': 4000000,  
'rating': 3.5, 'likes': 90}, {'nama': 'Very Smart TV', 'harga': 4000000, 'rating': 3.5, 'likes': 87}]
```

A. Indomie, Aqua, Laptop, Smart TV, Headphone, Very Smart TV.

B. Laptop, Headphone, Smart TV, Very Smart TV, Aqua, Indomie.

C. Indomie, Laptop, Aqua, Smart TV, Headphone, Very Smart TV

2. Jawaban

Untuk membuat aplikasi sederhana daftar produk, saya akan menggunakan bahasa pemrograman Python dan framework Flask.

Berikut adalah langkah-langkah yang dapat dilakukan untuk membuat aplikasi tersebut:

Instalasi Flask dan dependencies yang dibutuhkan:

```
pip install flask
```

Buat file app.py dan import modul yang dibutuhkan:

```
from flask import Flask, request, jsonify
```

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

```
products = [  
    {'id': 1, 'name': 'Produk 1', 'description': 'Deskripsi produk 1', 'price': 1000},  
    {'id': 2, 'name': 'Produk 2', 'description': 'Deskripsi produk 2', 'price': 2000},  
    {'id': 3, 'name': 'Produk 3', 'description': 'Deskripsi produk 3', 'price': 3000}  
]
```

Tambahkan route untuk masing-masing operasi:

Lihat daftar produk

```
@app.route('/products', methods=['GET'])
```

```
def get_products():
```

```
    return jsonify({'products': products})
```

Lihat detail sebuah produk

```
@app.route('/products/<int:product_id>', methods=['GET'])
```

```
def get_product(product_id):
```

```
    product = [product for product in products if product['id'] == product_id]
```

```
    if len(product) == 0:
```

```
        abort(404)
```

```
    return jsonify({'product': product[0]})
```

Buat produk

```
@app.route('/products', methods=['POST'])
```

```
def create_product():
```

```
    if not request.json or not 'name' in request.json:
```

```
        abort(400)
```

```
    product = {
```

```
        'id': products[-1]['id'] + 1,
```

```
        'name': request.json['name'],
```

```
        'description': request.json.get('description', ''),
```

```
        'price': request.json.get('price', 0)
```

```
    }
```

```
    products.append(product)
```

```
    return jsonify({'product': product}), 201
```

Edit produk

```
@app.route('/products/<int:product_id>', methods=['PUT'])
```

```
def update_product(product_id):
```

```
    product = [product for product in products if product['id'] == product_id]
```

```
    if len(product) == 0:
```

```
        abort(404)
```

```
    if not request.json:
```

```
        abort(400)
```

```
product[0]['name'] = request.json.get('name', product[0]['name'])
product[0]['description'] = request.json.get('description', product[0]['description'])
product[0]['price'] = request.json.get('price', product[0]['price'])
return jsonify({'product': product[0]})
```

Hapus produk

```
@app.route('/products/<int:product_id>', methods=['DELETE'])
```

```
def delete_product(product_id):
```

```
    product = [product for product in products if product['id'] == product_id]
```

```
    if len(product) == 0:
```

```
        abort(404)
```

```
    products.remove(product[0])
```

```
    return jsonify({'result': True})
```

Jalankan aplikasi dengan menambahkan kode berikut pada akhir file app.py:

```
if __name__ == '__main__':
```

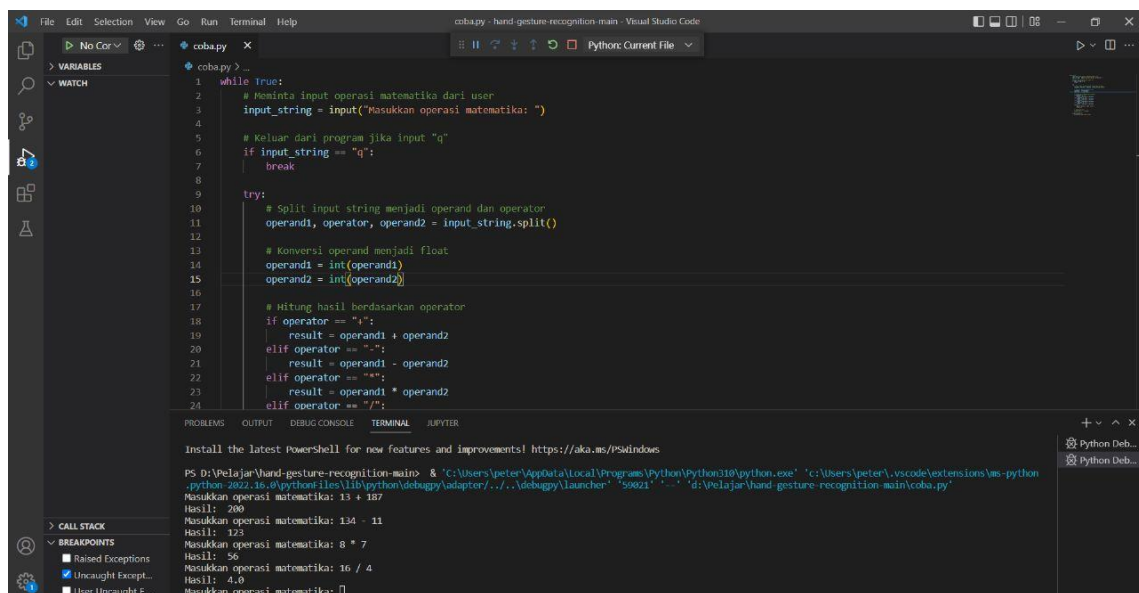
```
    app.run(debug=True)
```

Aplikasi dijalankan dengan perintah :

```
python app.py
```

Setelah dapat diakses API dengan menggunakan alamat <http://localhost:5000/> pada browser atau menggunakan tools API client seperti Postman.

3. Jawaban



```
1 while True:
2     # Meminta input operasi matematika dari user
3     input_string = input("Masukkan operasi matematika: ")
4
5     # Keluar dari program jika input "q"
6     if input_string == "q":
7         break
8
9     try:
10        # Split input string menjadi operand dan operator
11        operand1, operator, operand2 = input_string.split()
12
13        # Konversi operand menjadi float
14        operand1 = float(operand1)
15        operand2 = float(operand2)
16
17        # Hitung hasil berdasarkan operator
18        if operator == "+":
19            result = operand1 + operand2
20        elif operator == "-":
21            result = operand1 - operand2
22        elif operator == "*":
23            result = operand1 * operand2
24        elif operator == "/":
```

Install the latest Powershell for new features and improvements! <https://aka.ms/psinstall>

```
PS Dr:\Pelajar\hand-gesture-recognition-main > & 'C:\Users\peter\AppData\Local\Programs\Python\Python310\python.exe' 'C:\Users\peter\.vscode\extensions\ms-python.python-2022.16.0\python-files\lib\python\debugpy\adapter\..\..\debugpy\launcher' '59021' '-x' 'd:\Pelajar\hand-gesture-recognition-main\coba.py'
Masukkan operasi matematika: 13 + 187
Hasil: 200
Masukkan operasi matematika: 134 - 11
Hasil: 123
Masukkan operasi matematika: 8 * 7
Hasil: 56
Masukkan operasi matematika: 16 / 4
Hasil: 4.0
Masukkan operasi matematika: []
```

while True:

```
# Meminta input operasi matematika dari user
```

```
input_string = input("Masukkan operasi matematika: ")
```

```
if input_string == "q":  
    break
```

```
try:
```

```
    operand1, operator, operand2 = input_string.split()
```

```
    operand1 = int(operand1)  
    operand2 = int(operand2)
```

```
    if operator == "+":  
        result = operand1 + operand2  
    elif operator == "-":  
        result = operand1 - operand2  
    elif operator == "*":  
        result = operand1 * operand2  
    elif operator == "/":  
        result = operand1 / operand2  
    else:  
        print("Operator tidak valid")  
        continue
```

```
    print("Hasil: ", result)
```

```
except ValueError:  
    print("Operand tidak valid")
```