

# APLIKASI PENGOLAH DATA STATISTIK

**Irhamna Mahdi<sup>1</sup>, Salwa Farhanatussaidah<sup>2</sup>, Ganiya Syazwa<sup>3</sup>, Reynaldi Rahmad<sup>4</sup>,  
Sylaisha Andina Putriansyah<sup>5</sup>**

Program Studi Sains Data, Fakultas Sains, Institut Teknologi Sumatera

Email : [irhamna.122450049@student.itera.ac.id](mailto:irhamna.122450049@student.itera.ac.id)<sup>1</sup> , [salwa.122450055@student.itera.ac.id](mailto:salwa.122450055@student.itera.ac.id)<sup>2</sup> ,  
[ganiya.122450073@student.itera.ac.id](mailto:ganiya.122450073@student.itera.ac.id)<sup>3</sup> , [reynaldi.122450088@student.itera.ac.id](mailto:reynaldi.122450088@student.itera.ac.id)<sup>4</sup> ,  
[sylaisha.122450121@student.itera.ac.id](mailto:sylaisha.122450121@student.itera.ac.id)<sup>5</sup>

## 1. Pendahuluan

Seiring dengan pesatnya perkembangan aplikasi perangkat lunak komputer dan internet, maka manajemen dan penyebaran informasi data akan menjadi lebih efisien dan efektif. Hal ini membuat banyak perusahaan atau instansi yang memanfaatkan peluang tersebut untuk mengumpulkan dan mengelola data dengan menggunakan aplikasi pengolahan data statistik.(Nadinda et al., 2020)

Statistika meliputi pengumpulan data, pengolahan data, analisis data, dan penarikan kesimpulan berdasarkan data yang ada. Data merupakan kumpulan informasi yang dapat memberikan gambaran keadaan kritis suatu permasalahan.

Dalam pengaplikasian pengolahan data statistik kali ini, bahasa pemrograman yang digunakan adalah Python dengan menggunakan fungsi Map, Lambda dan Reduce. Fungsi Lambda digunakan untuk membuat dan menggunakan fungsi dalam satu baris yang pendek dan hanya akan digunakan sekali. Fungsi Map digunakan untuk mengaplikasikan fungsi pada anggota array atau struktur data. Sedangkan fungsi Reduce digunakan untuk menghitung jumlah dari elemen di dalam list tersebut.

Dengan menerapkan fungsi-fungsi tersebut, dalam aplikasi pengolahan data statistik ini, dataset excel yang *user* inputkan akan dapat dihitung dan diolah dengan menghasilkan output data statistik dasar seperti median, mean, modus, jumlah dan produk elemen data dengan efektif maupun efisien.

## 2. Metode

### 2.1 Fungsi

#### 2.1.1 Fungsi Lambda

Fungsi lambda adalah fungsi kecil satu baris yang digunakan untuk menggantikan fungsi sederhana. Sayangnya, fungsi lambda hanya dapat memuat satu ekspresi saja dan tidak dapat digunakan untuk menggantikan fungsi yang kompleks dan rumit. Penulisan lambda diawal adalah sebuah keharusan agar program mengetahui bahwa fungsi yang dimasukan menggunakan fungsi lambda sedangkan ekspresi dapat diubah sesuai dengan kebutuhan user.

#### 2.1.2 Fungsi Map

Fungsi map merupakan fungsi built-in dalam bahasa pemrograman python yang digunakan untuk memproses dan mengubah setiap item yang ada ke dalam perulangan tanpa menggunakan loop. Cara kerja fungsi ini adalah dengan

menerapkan fungsi ke dalam setiap item dan melakukan pembaruan pada variabelnya yang berbentuk list. (Ramandhany & Kunang, 2021)

### 2.1.3 Fungsi Reduce

Fungsi reduce dalam Python merupakan fungsi bawaan yang digunakan untuk menerapkan suatu fungsi tertentu ke semua elemen dalam sebuah iterable secara urut dengan mengambil hasil operasi sebelumnya dan elemen berikutnya sebagai argumen. Fungsi ini umumnya digunakan untuk melakukan agregasi data atau operasi pemrosesan data yang melibatkan urutan elemen. (Kuraś et al., 2023)

## 2.2 Modul

### 2.2.1 Modul 'Pandas'

Modul 'pandas' merupakan salah satu pustaka Python yang paling umum digunakan untuk menganalisis data dan membangun model machine learning. Dibuat berdasarkan paket Numpy, pandas mendukung array multidimensi, dan dapat dioptimalkan bersama dengan berbagai modul data science lainnya dalam ekosistem Python. (Lathifah et al., 2021).

### 2.2.2 Modul 'Functools'

Modul 'functools' merupakan salah satu modul *built-in* yang dimiliki oleh pemrograman python yang digunakan untuk bekerja pada fungsi-fungsi yang digunakan sebagai objek. Berikut fungsi utama modul 'functools':

- Partial() : untuk membuat fungsi parsial dari parameter tertentu.
- Partialmethod() : mendefinisikan metode dari fungsi yang didefinisikan untuk suatu parameter tertentu.
- Reduce() : untuk menerapkan fungsi dengan dua parameter secara berulang kali sehingga urutan elemennya berkurang menjadi satu nilai saja.
- Cache() : digunakan untuk membuat 'pembungkus tipis' pada sekitaran pencarian dictionary untuk tiap parameter dalam fungsi.
- Cached\_property() : untuk mengubah metode class menjadi properti dengan nilai yang dihitung sekali kemudian di-cache sebagai atribut normal.

## 3. Pembahasan

### 3.1 Kode Program

#### 3.1.1 Import Modul

```
import pandas as pd
from functools import reduce
```

Blok kode di atas merupakan awalan dalam membuat aplikasi pengolahan data statistik yaitu meng-*import* modul pandas serta fungsi reduce pada modul functools.

### 3.1.1 Fungsi

Dalam membuat aplikasi pengolah data statistik ada beberapa fungsi *user-defined* yang dibuat, yaitu sebagai berikut:

```
def load_data_from_file(file_path):  
    try:  
        data = pd.read_excel(file_path, header=None)  
        data_list = list(map(int, data.values.flatten()))  
        return data_list  
    except FileNotFoundError:  
        print("File tidak ditemukan.")  
        return None
```

Fungsi `load_data_from_file(file_path)` yang digunakan untuk membaca tipe data file Excel. `Pd_read_excel(file_path, header=None)` berfungsi untuk membaca file dari path yang diberikan dengan format file excel tersebut tidak memiliki header. `data_list = list(map(int, data.values.flatten()))` digunakan untuk mengubah tipe data yang berada dalam file menjadi integer. Lalu, `return data_list` digunakan untuk mengembalikan nilai data dalam bentuk integer. Fungsi ini menggunakan blok *try-except*, dimana akan mencetak pesan “File tidak ditemukan” dan mengembalikan nilai `None` apabila file tidak ditemukan.

```
def get_user_input():  
    try:  
        n = int(input("Masukkan jumlah data: "))  
        data = []  
  
        for i in range(n):  
            value = float(input(f"Masukkan nilai ke-{i + 1}: "))  
            data.append(value)  
        return data  
  
    except ValueError:  
        print("Mohon masukkan angka yang valid.")  
        return get_user_input()
```

Fungsi `get_user_input()`, dimana fungsi ini akan meminta *user* memasukkan sejumlah data. `for i in range(n)` berfungsi untuk memasukkan data sebanyak jumlah data yang diminta. Kemudian nilai yang dimasukkan dikonversi menjadi tipe data `float` menggunakan kode `float(input(f"MASukkan nilai-{i + 1}: "))` dan apabila data yang dimasukkan tidak valid maka *user* menginputkan data dari awal.

```
def calculate_statistics(data):
    mean = lambda data: reduce(lambda x, y: x + y, data) /
len(data)
    median = lambda data: sorted(data)[len(data)//2] if
len(data)%2 != 0 else (sorted(data)[len(data)// 2-1] +
sorted(data)[len(data)//2])/2
    return {
        "max": max(data),
        "min": min(data),
        "mean": mean(data),
        "median": median(data)
    }
```

Blok kode di atas merupakan pendefinisian fungsi `calculate_statistics()` dengan argumen `data` didalamnya. Fungsi tersebut melakukan operasi perhitungan rata-rata dan median. Selain itu, fungsi ini juga mengembalikan sebuah dictionary yang berisi statistik dasar (min, max, mean, dan median) dari data yang diberikan.

```
def main():
    while True:
        print("\n====Welcome to Program====")
        print("\nPilih data yang akan dimasukkan:")
        print("1. Input data")
        print("2. Upload file")
        print("3. Keluar")

        menu = input("\nMasukkan pilihan Anda (1/2/3): ")

        if menu == "3":
            print("\n====Sampai Jumpa====")
            break
```

Fungsi `main()` digunakan untuk menampilkan menu pilihan serta meminta *user* memilih menu pilihan yang telah disediakan dan program akan berhenti serta menampilkan pesan "====Sampai Jumpa====" apabila *user* memilih menu ke-3.

```

data = None
while data is None:
    if menu == "1":
        data = get_user_input()

    elif menu == "2":
        file_path = input("Masukkan file path: ")
        data = load_data_from_file(file_path)
    else:
        print("Pilihan tidak valid.")
        break

    if data is None:
        print("Maaf, data yang Anda masukkan tidak cocok. Silakan coba lagi.")

    if data:
        # Menampilkan beberapa baris awal data
        print("\nData Awal:")
        print(data)
        data = calculate_statistics(data)
        print(data)
if __name__ == "__main__":
    main()

```

Kode program di atas merupakan interaksi pengguna dengan program, dimana apabila *user* memilih opsi “1” maka program akan menjalankan fungsi `get_user_input()` dan akan menjalankan fungsi `load_data_from_file(file_path)` apabila *user* memilih opsi “2”. Setelah data berhasil terbaca, program akan menampilkan data awal serta statistik dasar dari data yang dimasukkan menggunakan fungsi `calculate_statistics(data)`.

### 3.2 Hasil Program

Berikut merupakan tampilan hasil program aplikasi pengolahan data statistik:

<pre> ====Welcome to Program====  Pilih data yang akan dimasukkan: 1. Input data 2. Upload file 3. Keluar  Masukkan pilihan Anda (1/2/3): 1 Masukkan jumlah data: 4 Masukkan nilai ke-1: 3 Masukkan nilai ke-2: 3 Masukkan nilai ke-3: 6 Masukkan nilai ke-4: 8 </pre>	<pre> Data Awal: [3.0, 3.0, 6.0, 8.0] {'max': 8.0, 'min': 3.0, 'mean': 5.0, 'median': 4.5}  ====Welcome to Program====  Pilih data yang akan dimasukkan: 1. Input data 2. Upload file 3. Keluar  Masukkan pilihan Anda (1/2/3):  </pre>
--	---

```

====Welcome to Program====
Pilih data yang akan dimasukkan:
1. Input data
2. Upload file
3. Keluar

Masukkan pilihan Anda (1/2/3): 2
Masukkan file path: /content/datapbf.xlsx

Data Awal:
[40, 70, 50, 42, 45, 21, 45, 90, 48, 70, 70, 75, 80, 77, 42, 43]
{'max': 90, 'min': 21, 'mean': 56.75, 'median': 49.0}

====Welcome to Program====
Pilih data yang akan dimasukkan:
1. Input data
2. Upload file
3. Keluar

Masukkan pilihan Anda (1/2/3):

```

Masukkan pilihan Anda (1/2/3): 3

====Sampai Jumpa====

Gambar 1. Ouput dan contoh penggunaan program

#### 4. Kesimpulan

Pada tugas kali ini, kami mengembangkan aplikasi pengolahan data statistik dengan menggunakan bahasa pemrograman python. Aplikasi ini memiliki fungsi seperti Lambda, Map, dan Reduce. Aplikasi ini menggunakan modul Pandas dan Functools yang digunakan untuk analisis dan pemrosesan data. Modul Pandas digunakan untuk menganalisis data dan membangun model pada machine learning, sedangkan functools digunakan sebagai objek pada fungsi. Hasil program menunjukkan bahwa aplikasi bekerja dengan baik untuk mengolah data statistik seperti median, mean, modus, jumlah, dan produk elemen data. Fungsi lambda digunakan untuk menggantikan fungsi sederhana, namun fungsi ini hanya dapat digunakan untuk memuat satu ekspresi saja. Fungsi Map digunakan memproses serta mengubah setiap item ke dalam perulangan tanpa looping. Fungsi Reduce digunakan melakukan operasi pada pemrosesan data yang melibatkan urutan elemen. Pada pengaplikasiannya, user diminta memasukkan data baik secara manual maupun dalam file dengan format Excel, dimana program akan memberikan jawaban mengenai hasil dari statistik dasar pada data yang telah di-input.

## DAFTAR PUSTAKA

- Kuraś, P., Strzalka, D., Kowal, B., & Mazurek, J. (2023). REDUCE – A Python Module for Reducing Inconsistency in Pairwise Comparison Matrices. *Advances in Science and Technology Research Journal*, 17(4), 227–234. <https://doi.org/10.12913/22998624/170187>
- Lathifah, L., Handoyo, E., & Adi Soetrisno, Y. A. (2021). Sistem Crawling Data Instrumen Akreditasi Berbasis Selenium Dan Pandas. *Transient: Jurnal Ilmiah Teknik Elektro*, 10(1), 84–91. <https://doi.org/10.14710/transient.v10i1.84-91>
- Nadinda, D., Priyanto, H., & Muhandi, H. (2020). Aplikasi Pengolahan Data Statistik Sektoral pada Dinas Komunikasi dan Informatika Provinsi Kalimantan Barat. *Jurnal Sistem Dan Teknologi Informasi (Justin)*, 8(2), 140. <https://doi.org/10.26418/justin.v8i2.31997>
- Ramandhany, D., & Kunang, Y. N. (2021). Visualisasi Heat Map Data Kecelakaan Di Kota Palembang. *Bina Darma Conference on Computer Science*, 304–311.