**TUGAS BESAR**

**PEMROGRAMAN FUNGSIONAL**

**High Order Function**

**The Functional Approach to Web Services**

****

**OLEH :**

**Mochammad Bagus Insan (19102128)**

**Regi Apriandi (19102283)**

**Fahri Rizmawan (19102138)**

**INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

**FAKULTAS INFORMATIKA**

**PRODI S1 INFORMATIKA**

**KATA PENGANTAR**

Puji syukur kita panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat dan hidayah-Nya kita dapat menyelesaikan laporan ini dengan tepat waktu. Yang mana laporan ini adalah tugas besar untuk mata kuliah Pemrograman Fungsional yang diampu oleh bapak Agus Priyanto, S.Kom., M.Kom. Laporan ini merupakan hasil dari tugas bagi para mahasiswa, untuk mempelajari dan memahami bahasa Pemrograman Phyton yang dapat diimplementasikan untuk masa depan yang akan datang.

Laporan ini disusun sebagai pengganti Ujian Akhir Semester mata kuliah Pemrograman Fungsional. Semoga laporan ini dapat bermanfaat dan senantiasa menjadi pembelajaran untuk meraih prestasi yang gemilang. Kritik dan saran dari dosen pengajar mata kuliah dan juga bagi semua pembaca, sangat penulis harapkan untuk perbaikan dan penyempurnaan dalam pembelajaran pada masa mendatang. Karena penulis sadar masih banyak kekurangan yang ada pada laporan ini.

**DAFTAR ISI**

**KATA PENGANTAR**  i

**DAFTAR ISI**  ii

**BAB I PENDAHULUAN**

1.1 Latar Belakang 1

2.1 Rumusan Masalah 1

3.1 Tujuan 2

**BAB II TINJAUAN PUSTAKA**

2.1 Python 3

2.2 CSV 3

2.3 Lambda 4

2.4 Map() 4

2.5 Filter() 4

2.6 Sorted() 4

2.7 Serialisasi 4

2.8 JSON 4

2.9 Postman 5

2.10 Ngrok 5

2.11 Flask 5

**BAB III PEMBAHASAN**

3.1 High Order Function 6

3.2.1 Penjelasan Code 6

3.2.2 Hasil Implementasi Program 12

3.2 The Functional Approach to Web Services 14

3.2.1 Sintaks dan Struktur JSON 14

3.2.2 Penjelasan Code 15

3.2.3 Alur kerja dari web service menggunakan Flask dan Postman 17

3.2.4 Hasil Implementasi Program 18

**BAB IV KESIMPULAN**

4.1 Kesimpulan 19

4.1.1 High Order Function 19

4.1.2 JSON (JAVA SCRIPT OBJECT NOTATION) 19

4.2 Saran 19

**DAFTAR PUSTAKA**  20

**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

Saat ini Informatika memiliki perkembangan yang sangat pesat. Informatika sendiri yaitu disiplin ilmu teknik yang menangani masalah transformasi atau pengolahan fakta simbolik dengan menggunakan komputer melalui proses-proses logika. Dalam informatika, hal yang paling berhubungan adalah bahasa pemrograman atau bahasa komputer.

Bahasa pemrograman dibagi dalam empat kelompok namun dari sumber lain ada yang hanya menyebutkan tiga kelompok bahasa pemrograman. Empat jenis bahasa pemrograman yaitu Object Oriented Language, High Level, Mid Level, dan Low level. Sebenarnya, keempat pengelompokan bahasa tersebut sama saja intinya, seperti bahasa manusia yang memiliki bermacam-macam bahasa namun sebenarnya bermakna sama, yang membedakan adalah simbol atau istilah yang dipakai.

Bahasa pemrograman saat ini jumlahnya sangat banyak. Python merupakan salah satu bahasa pemrograman populer yang digunakan oleh banyak developer. Menurut survei bahasa pemrograman versi www.tiobe.com, Python berada di peringkat ke-5 pada tahun 2016. Selain itu, Python juga bisa digunakan untuk enterprise. Dalam tingkatan bahasa pemrograman, Python termasuk high level language. Python menjadi salah satu bahasa pemrograman yang dapat digunakan untuk membangun aplikasi, baik itu berbasis desktop, web ataupun berbasis mobile.

Payton juga dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan diberbagai platform sistem operasi. seperti halnya bahasa pemrograman dinamis, phyton seringkali digunakan sebagai bahasa skrip dengan interpreter yang terintegrasi dalam sistem operasi. Saat ini kode python dapat dijalankan pada sistem berbasis Linux atau Unix, Windows, Mac OS X, Java Virtual Machine, OS/2, Amiga, Palm, dan Symbian(untuk produk-produk Nokia). Phyton didistribusikan dengan beberapa lisensi yang berbeda dari beberapa versi, namun pada prinsipnya Phyton dapat dapat diperoleh dan dipergunakan secara bebas, bahkan untuk kepentingan komersial. Lisensi Phyton tidak bertentangan baik menurut definisi Open Source maupun General Public License(GPL).

1. **Rumusan Masalah**

Berdasarkan latar belakang Tugas Akhir diatas maka dapat ditetapkan identifikasi masalah sebagai berikut :

1. Apa yang dimaksud dengan pengertian dari python, Pandas, Matplotlib, CSV, Lambda, Map, Filter, dan Sorted didalam bahasa pemrograman Phyton.
2. Membuat program studi kasus dengan data-data yang real, data real yang dimaksud adalah data yang benar ada di internet, disini kita memakai data tentang covid-19 yang berada di Indonesia.
3. Apa yang dimaksud dengan JSON ( JAVASCRIPT OBJECT NOTATION).
4. Apa yang dimaksud dengan serialisasi data?
5. Apa yang dimaksud dengan Postman?
6. **Tujuan**

Adapun tujuan dari pembuatan makalah ini adalah:

1. Untuk mengetahui pengertian dari Python, Pandas, Matplotlib, CSV, Lambda, Map, Filter, Sorted didalam bahasa pemrograman Phyton.
2. Mempermudah user untuk mengakses data covid-19 di Indonesia per tanggal 21 Januari.
3. Untuk mengetahui pengertian dari JSON (JavaScript Object Notation) dan memberikan pengetahuan baru tentang pemrograman.
4. Untuk mengetahui pengertian dari Serialisasi data.
5. Untuk mengetahui pengertian dari Postman.

**BAB II**

**TINJAUAN PUSTAKA**

1. **Python**

Python adalah salah satu bahasa pemrograman yang dapat melakukan eksekusi sejumlah instruksi multi guna secara langsung (interpretatif) dengan metode orientasi objek. Python adalah bahasa pemrograman yang paling mudah dipahami. Python dibuat oleh programmer Belanda yang bernama Guido Van Rossum.

1. **Python PIP**

PIP merupakan program untuk manajemen paket di Python. Tugasnya untuk menginstal, menghapus, upgrade paket Python, dll. Paket Pyhon itu merupakan sebuah modul yang berisi kode-kode python dan isi paket ini bisa kita impor ke dalam program kita.

1. **Pandas**

Pandas adalah sebuah library di Python yang berlisensi BSD dan open source yang menyediakan struktur data dan analisis data yang mudah digunakan. Pandas biasa digunakan untuk membuat tabel, mengubah dimensi data, mengecek data, dan lain sebagainya. Struktur data dasar pada Pandas dinamakan DataFrame, yang memudahkan kita untuk membaca sebuah file dengan banyak jenis format seperti file .txt, .csv, dan .tsv. Fitur ini akan menjadikannya table dan juga dapat mengolah suatu data dengan menggunakan operasi seperti join, distinct, group by, agregasi, dan teknik lainnya yang terdapat pada SQL.

1. **Matplotlib**

Matplotlib adalah library Python yang fokus pada visualisasi data seperti membuat plot grafik. Matplotlib pertama kali diciptakan oleh John D. Hunter dan sekarang telah dikelola oleh tim developer yang besar. Awalnya matplotlib dirancang untuk menghasilkan plot grafik yang sesuai pada publikasi jurnal atau artikel ilmiah. Matplotlib dapat digunakan dalam skrip Python, Python dan IPython shell, server aplikasi web, dan beberapa toolkit graphical user interface (GUI) lainnya.

1. **CSV**

CSV (Comma Separated Value) merupakan format basis data sederhana dimana setiap record yang ada dipisahkan dengan tanda koma (,) atau titik koma (;).Format data CSV ini dapat diolah dengan berbagai text editor dengan mudah.

Dalam bahasa pemrograman Python telah disediakan modul CSV yang khusus untuk mengolah data berformat CSV. Untuk memanipulasi data CSV dengan python tentunya pertama dilakukan adalah mengimport modul CSV dengan perintah import CSV.

1. **Lambda**

Lambda expression di python adalah sebuah ekspresi untuk membuat fungsi. Lambda sendiri berasal dari teori kalkulus, yakni Lambda Calculus yang dikenalkan oleh Alonzo Church di tahun 1930. Dengan lambda, kita bisa membuat fungsi tanpa nama atau dikenal dengan anonymous function. dengan Lambda tidak perlu menggunakan def dan return. Lambda memiliki lebih dari satu argumen atau parameter, tetapi hanya bisa memiliki satu ekspresi atau isi.

1. **Map()**

Fungsi Map() digunakan untuk menerapkan suatu fungsi yang diberikan ke setiap item atau anggota iterable (list, tuple, dan lain-lain) dan mengembalikannya dalam bentuk objek map. Objek map() ini nantinya dapat diubah ke bentuk list, tuple, dan lain-lain yang mana anggotanya adalah hasil dari proses anggota iterable lama.

1. **Filter()**

Fungsi filter() digunakan untuk menyaring elemen, item, atau anggota iterable dengan bantuan fungsi yang bertugas menguji setiap anggota iterable apakah bernilai True atau False. hasil atau nilai yang dikembalikan berupa iterable baru dari anggota iterable lama yang sudah melewati proses pengujian (dari fungsi yang dibuat) dan bernilai True.

1. **Sorted()**

Fungsi sorted() digunakan untuk mengurutkan iterable baik secara ascending (naik) atau descending (turun). Fungsi sorted() menerima tiga buah parameter yang setidaknya satu diantaranya wajib diisi yaitu :

* Iterable - sequence (string, list, tuple), collection (set, dictionary, frozenset), atau iterator lainnya.
* key (opsional) - fungsi kunci yang digunakan sebagai pembanding pengurutan. Defaultnya adalah None.
* Reverse (opsional) - menentukan apakah iterable diurut secara terbalik (descending) atau tidak (ascending). Jika bernilai true maka iterable akan diurut secara terbalik. defaultnya adalah false.

1. **Serialisasi Data**

Serialisasi Data adalah sebuah proses mengubah object menjadi byte stream yang direpresentasikan dalam sebuah string. Proses serialisasi object digunakan untuk menyimpan suatu object ke dalam file atau memori dalam komputer atau untuk keperluan transmisi object. Kebalikan dari proses serialisasi adalah proses deserialisasi.

1. **JSON**

JSON (JavaScript Object Notation) adalah sebuah format data yang digunakan untuk pertukaran dan penyimpanan data. JSON merupakan bagian (subset) dari Javascript. JSON bisa dibaca dengan berbagai macam bahasa pemrograman seperti C, C++, C#, Java, Javascript Perl, Python, dan banyak lagi.

1. **Postman**

Postman adalah sebuah aplikasi yang berfungsi sebagai REST Client untuk uji coba REST API. Postman biasa digunakan oleh developer pembuat API sebagai tools untuk menguji API yang telah mereka buat

1. **Ngrok**

Ngrok adalah proxy server untuk membuat / membuka jaringan private melalui NAT atau firewall untuk menghubungkan localhost ke internet dengan tunnel yang aman. Kalau teman-teman masih bingung, ngrok itu ibaratnya jaringan yang ada di localhost bisa diakses secara publik, jadi ga hanya di laptop temen-temen aja.

1. **Flask**

Flask adalah kerangka kerja aplikasi web bersifat kerangka kerja mikro yang ditulis dalam bahasa pemrograman Python dan menggunakan dependensi Werkzeug dan Jinja2. Aplikasi yang menggunakan Flask antara lain adalah Pinterest, LinkedIn, dan halaman web komunitas situs Flask itu sendiri.

**BAB III**

**PEMBAHASAN**

1. **High Order Function**

High Order Function atau fungsi orde tinggi adalah fungsi yang menerima fungsi lain sebagai parameternya, dan mengembalikan fungsi lain sebagai keluarannya. Di dalam High Order Function sesuai study case yang diminta mencakupi beberapa fungsi, yaitu fungsi lambda, map, filter dan sorted.

Pada pemrograman Python tentunya membutuhkan data real, yang mana kita menggunakan CSV sebagai data yang nantinya akan di import ke project. di dalam CSV terdapat data yang berisi seputar informasi Covid-19.

1. **Penjelasan Code**

untuk pertama-tama kita mengimpor perpustakaan python yang dibutuhkan oleh program. perpustakaan yang dibutuhkan oleh program yaitu pandas yang berfungsi untuk mengolah data dan menjadikannya dictionary python agar dapat di akses oleh fungsi sesuai studi kasus, yang kedua yaitu perpustakaan os dimana berfungsi untuk membuat fungsi clear\_screen() dalam python, yang ketiga yaitu mengimpor perpustakaan matplotlib dimana berfungsi untuk menampilkan grafik. setelah kita mengimpor beberapa perpustakaan python yang kita butuhkan, lalu kita membuat variabel df yang berisi data csv covid-19 indonesia.

import pandas as pd  
import os  
import matplotlib.pyplot as plt  
df = pd.read\_csv('covid\_19\_indonesia\_time\_series\_all.csv')

selanjutnya kita membuat fungsi menu dengan menggunakan def, fungsi tersebut berisi pemanggilan fungsi clear\_screen dan menu yang akan ditampilkan ke user dan berisi pengkondisian if else yang berfungsi untuk pemilihan menu oleh user yang nantinya akan mengeksekusi fungsi-fungsi yang akan dibuat.

def menu():  
 clear\_screen()  
 print("APLIKASI CEK DATA COVID-19 INDONESIA JANUARI 2021")  
 print("1. CEK KASUS COVID-19 (filter(lambda))")  
 print("2. CEK URUTAN KASUS COVID-19 (sorted(lambda))")  
 print("3. CEK GRAFIK KASUS COVID-19 (map(lambda))")  
 selected\_menu = input("PILIH MENU : ")  
 if selected\_menu == "1":  
 cari\_data()  
 elif selected\_menu == "2":  
 urutkan\_data()  
 elif selected\_menu == "3":  
 grafik\_data()  
 exit()  
 else:  
 print("PILIHAN MENU SALAH!")  
 kembali\_ke\_menu()

setelah kita membuat fungsi menu, kita membuat fungsi kembali ke menu dimana berfungsi untuk mengembalikan ke menu awal jika fungsi-fungsi berhasil atau telah dieksekusi dan fungsi clear screen untuk menghapus atau membersihkan teks atau output yang ada di workspace agar memudahkan user untuk membaca pada workspace.

def kembali\_ke\_menu():  
 print("\n")  
 input("TEKAN ENTER UNTUK KEMBALI KE MENU")  
 menu()  
  
def clear\_screen():  
 os.system('cls' if os.name == 'nt' else 'clear')

untuk fungsi berikutnya yang kita buat adalah fungsi cari\_data, disini kita membuat fungsi dengan penamaan yang disesuaikan dengan eksekusi fungsi di pemilihan fungsi menu, didalam fungsi cari\_data terdapat pemanggilan fungsi clear\_screen dan berisi pemilihan menu yang berisi pengkondisian. untuk pemilihan menu yang pertama, kita membuat variabel kolom yang berfungsi untuk mengakses kolom date dan kolom total cases di variabel df yang berisi data csv.

pada kolom tersebut terdapat group by untuk menjadikan satu value dengan beberapa data berdasarkan kolom date. Selanjutnya ada TotalCases.sum yang berfungsi untuk mejumlahkan semua value berdasarkan kolom Date. kemudian reset\_index() berfungsi untuk mengatur ulang index ke default. selanjutnya ke variabel dict yang dimana variabel tersebut untuk merubah data dari dataframe ke dictionary python

def cari\_data():  
 clear\_screen()  
 print("CEK KASUS BERDASARKAN :")  
 print("1. TANGGAL (JANUARI)")  
 print("2. PROVINSI")  
 sub\_menu = input("PILIH MENU : ")  
 pilihan = input("MASUKAN DATA : ")  
 if sub\_menu == "1":  
 kolom = df.groupby("Date").TotalCases.sum().reset\_index()  
 dict = kolom.to\_dict('Records')  
 fungsi = list(filter(lambda x: x['Date'] == int(pilihan), dict))  
 hasil = pd.DataFrame.from\_dict(fungsi)  
 print(hasil)  
  
 elif sub\_menu == "2":  
 lokasi = df.groupby("Location").TotalCases.max().reset\_index()  
 dict = lokasi.to\_dict('Records')  
 fungsi = list(filter(lambda x: x['Location'] == pilihan, dict))  
 hasil = pd.DataFrame.from\_dict(fungsi)  
 print(hasil)  
 kembali\_ke\_menu()

Selanjutnya kita membuat fungsi urutkan\_data, disini kita membuat fungsi dengan penamaan yang telah disesuaikan dengan eksekusi fungsi di pemilihan menu, di dalam fungsi urutkan\_data berisi pemanggilan fungsi clear\_scren dan menampilkan list menu dengan menggunakan print,setelah itu berisi pemilihan menu dan berisi pengkondisian. Untuk pemilihan menu pertama, kita membuat variabel kolom yang berfungsi untuk mengakses kolom location dan kolom total kematian, dengan mengambil data terbesar total kematian berdasarkan lokasi di variabel df yang berisi data csv. Pada kolom tersebut terdapat group by untuk menjadikan satu value dengan beberapa data berdasarkan kolom lokasi. Selanjutnya, ada TotalDeath.max yang berfungsi untuk menjumlahkan semua value berdasarkan kolom location. Kemudian, reset index () berfungsi untuk mengatur ulang index ke default. Selanjutnya, ada variabel dict variabel dict ini befungsi untuk merubah data dari dataframe ke dictionary python.

Untuk pemilihan menu kedua, kita membuat variabel kolom yang berfungsi untuk mengakses kolom location dan kolom total kesembuhan, dengan mengambil data terbesar total kesembuhan berdasarkan lokasi di variabel df yang berisi data csv. Pada kolom tersebut terdapat group by untuk menjadikan satu value dengan beberapa data berdasarkan kolom lokasi. Selanjutnya, ada TotalRecovered.max yang berfungsi untuk menjumlahkan semua value berdasarkan kolom location. Kemudian, reset index () berfungsi untuk mengatur ulang index ke default. Selanjutnya, ada variabel dict variabel dict ini befungsi untuk merubah data dari dataframe ke dictionary python.

Di menu ketiga, kita membuat variabel kolom yang berfungsi untuk mengakses kolom location dan kolom total kasus, dengan mengambil data terbesar dari total kasus tersebut berdasarkan lokasi di variabel df yang berisi data csv. Pada kolom tersebut terdapat group by untuk menjadikan satu value dengan beberapa data berdasarkan kolom location. Selanjutnya, ada TotalCases.max yang berfungsi untuk menjumlahkan semua value berdasarkan kolom lokasi. Kemudian, reset index () berfungsi untuk mengatur ulang index ke default. Selanjutnya, ada variabel dict variabel ini berfungsi untuk merubah data dari dataframe ke dictionary python.

def urutkan\_data():  
 clear\_screen()  
 print("URUTAN DATA BERDASARKAN :")  
 print("1. TOTAL KEMATIAN")  
 print("2. TOTAL KESEMBUHAN")  
 print("3. TOTAL KASUS")  
 sub\_menu = input("PILIH MENU : ")  
 if sub\_menu == "1":  
 lokasi = df.groupby("Location").TotalDeaths.max().reset\_index()  
 dict = lokasi.to\_dict('Records')  
 fungsi = list(sorted(dict, key=lambda x: x['TotalDeaths'], reverse=True))  
 hasil = pd.DataFrame.from\_dict(fungsi)  
 print(hasil)  
 elif sub\_menu == "2":  
 lokasi = df.groupby("Location").TotalRecovered.max().reset\_index()  
 dict = lokasi.to\_dict('Records')  
 fungsi = list(sorted(dict, key=lambda x: x['TotalRecovered'], reverse=True))  
 hasil = pd.DataFrame.from\_dict(fungsi)  
 print(hasil)  
 elif sub\_menu == "3":  
 lokasi = df.groupby("Location").TotalCases.max().reset\_index()  
 dict = lokasi.to\_dict('Records')  
 fungsi = list(sorted(dict, key=lambda x: x['TotalCases'], reverse=True))  
 hasil = pd.DataFrame.from\_dict(fungsi)  
 print(hasil)  
 kembali\_ke\_menu()

setelah membuat fungsi urutkan\_data, lalu kita akan membuat fungsi yang diberi nama grafik\_data. Difungsi ini pertama kita memanggil fungsi clear\_screen yang telah dibuat lalu setelah itu kita menampilkan beberapa list menu pilihan menggunakan fungsi print yang dapat menampilkan output string ke user. Sama seperti fungsi-fungsi yang kita telah buat sebelumnya, di fungsi grafik\_data ini, terdapat pemilihan menu dengan menginisiasi variable inputan untuk user agar dapat menginput menu kedalam program.

Untuk pemilihan menu yang pertama atau di bagian perulangan if sub\_menu == “1”, kita membuat variable kolom yang berfungsi untuk mengambil kolom TotalCases dengan menjumlahkan semua valuenya menggunakan fungsi sum() berdasarkan kolom Date lalu mereset indexnya Kembali ke default. Setelah kita berhasil membuat variable kolom, lalu kita membuat variable dict yang berfungsi mengubah tipe data variable kolom yaitu dataframe ke dictionary supaya bisa dieksekusi di fungsi map dan lambda.

Setelah berhasil membuat variable dict, lalu kita membuat fungsi menggunakan lambda dan map untuk dapat mengambil kedua kolom tersebut agar dapat dijadikan grafik menggunakan library python matplotlib. Untuk yang pertama kita membuat fungsi menggunakan lambda dan map yang diberi nama date dimana fungsi tersebut mengambil value dari kolom Date menggunakan get() dengan hasil berupa tipe data list, dan untuk yang kedua kita membuat fungsi menggunakan lambda dan map yang diberi nama death dimana fungsi tersebut mengambil value dari kolom TotalCases menggunakan get() dengan hasil berupa tipe data list.

Setelah kita mendapatkan dua fungsi yaitu date dan death, lalu kita menginisasi grafik matplotlib dengan membuat variable ax, fungsi variable ax disini adalah untuk mengatur tanggal pada grafik. Setalah membuat variable ax, kita menginisiasi label pada bagian bawah untuk keterangan tanggal menggunakan plt.xlabel() dilanjutkan dengan penamaan menggunakan string didalam kurung dan menginisiasi label pada bagian samping untuk keterangan jumlah kasus menggunakan plt.ylabel() dilanjutkan penamaan menggunakan tipe data string didalam kurung.

Setelah membuat label untuk bagian samping dan bawah, lalu kita membuat judul untuk grafik tersebut menggunakan plt.title() dilanjutkan dengan penamaan judul menggunakan string didalam kurung dan mengatur tanggal agar terlihat jelas dengan inisiasi ax.set\_xticks() dilanjutkan menginput tanggal yang sesuai, disini berawal tanggal 1 sampai 21 dengan menggunakan list didalam kurungnya. Untuk dapat menampilkan grafik tersebut, kita memanggilnya dengan perintah plt.plot().

Untuk pemilihan menu yang ke dua, kita menggunakan cara yang sama seperti sebelumnya dengan mengganti kolom untuk menu kedua yaitu menggantinya menjadi kolom NewCase.

def grafik\_data():  
 clear\_screen()  
 print("LIHAT GRAFIK :")  
 print("1. TOTAL KASUS")  
 print("2. PENAMBAHAN KASUS PERHARI")  
 sub\_menu = input("PILIHAN MENU : ")

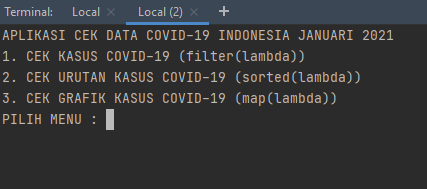
if sub\_menu == "1":  
 kolom = df.groupby("Date").TotalCases.sum().reset\_index()  
 dict = kolom.to\_dict('Records')  
 date = list(map(lambda x: x.get('Date'), dict))  
 kasus = list(map(lambda x: x.get('TotalCases'), dict))  
 # grafik matplotlib  
 ax = plt.subplot()  
 plt.xlabel("tanggal (Januari)")  
 plt.ylabel("jumlah kasus")  
 plt.title("Grafik Jumlah Kasus Covid-19")  
 ax.set\_xticks([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21])  
 plt.plot(date, kasus)  
 plt.show()  
 elif sub\_menu == "2":  
 kolom = df.groupby("Date").NewCases.sum().reset\_index()  
 dict = kolom.to\_dict('Records')  
 date = list(map(lambda x: x.get('Date'), dict))  
 kasus = list(map(lambda x: x.get('NewCases'), dict))  
 #grafik matplotlib  
 ax = plt.subplot()  
 plt.xlabel("tanggal (Januari)")  
 plt.ylabel("jumlah kasus")  
 plt.title("Grafik Penambahan Kasus Harian Covid-19")  
 ax.set\_xticks([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21])  
 plt.plot(date, kasus)  
 plt.show()  
 kembali\_ke\_menu()

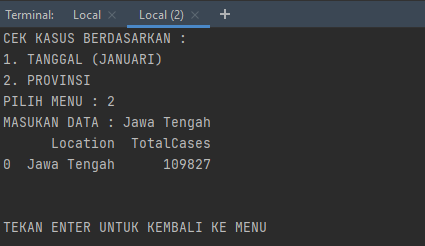
Setelah kita berhasil membuat semua fungsi yang dibutuhkan, lalu diakhir kode kita membuat main loop programnya supaya program dapat dijalankan dengan eksekusi pertama di fungsi menu(), karena fungsi menu disini adalah untuk menapilkan menu-menu yang akan dipilih oleh user atau fungsi menu disini merupakan fungsi yang utamanya.

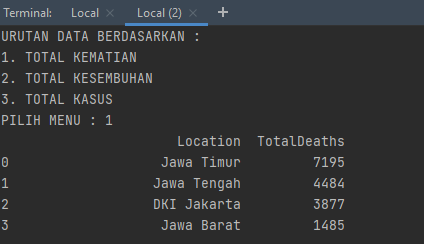
if \_\_name\_\_ == "\_\_main\_\_":  
 while True:  
 menu()

1. **Hasil Implementasi Program**

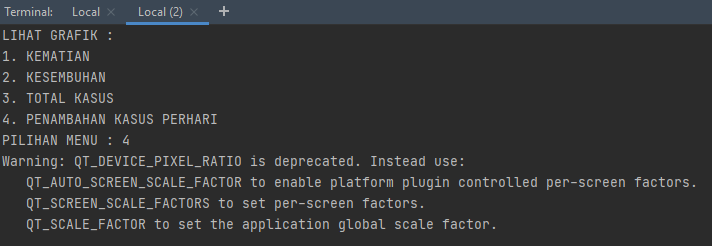
* **Menu Utama**

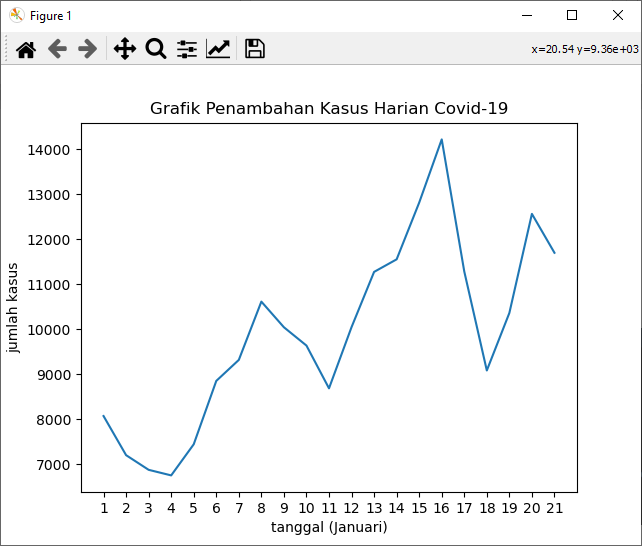


* **Menu = 1**
* **Menu = 2**



* **Menu = 3**





1. **The Functional Approach to Web Services**

Web service adalah sekumpulan fungsi, logika, atau method sebagai standar untuk pertukaran data antara aplikasi atau sistem dan dapat diakses secara remote dari berbagai device dengan bahasa pemrograman , arsitektur , atau platform yang berbeda-beda. Struktur dari sebuah web service bisa berbentuk Simple Object Access Protocol (SOAP) atau Representational State Transfer (REST), sedangkan fungsi tersebut direpresentasikan dalam bentuk text, XML atau JSON

1. **Sintaks dan Struktur JSON**

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atau associative array.

2. Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence). Pada dasarnya, semua bahasa pemprograman modern mendukung struktur data ini dalam bentuk yang sama maupun berlainan.

JSON menggunakan bentuk sebagai berikut

* 1. Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).
  2. Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [ (kurung kotak buka) dan diakhiri dengan ] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).
  3. Nilai (value) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat.
  4. String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.
  5. Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan.

1. **Penjelasan Code**

Untuk studi kasus yang kedua, kita harus mempersiapkan perpustakaan flask supaya kita dapat membangun web server menggunakan python. Kode dibawah merupakan cara untuk mengimpor perpustakaan python yaitu flask dan pandas.

from flask import Flask, request  
import pandas as pd  
  
app = Flask(\_\_name\_\_)

setelah kita mengimpor flask dan pandas, lalu selanjutnya akan membuat permintaan ke server web, yang memproses permintaan tersebut dan mengembalikan tanggapan kembali ke browser. Di aplikasi Flask kita, kita bisa membuat titik akhir untuk menangani berbagai permintaan. Permintaan dari URL yang berbeda dapat diarahkan ke titik akhir yang berbeda dalam proses yang disebut perutean.

Untuk membangun rute, pertama-tama kita perlu mendefinisikan sebuah fungsi, yang dikenal sebagai fungsi tampilan, yang berisi kode untuk memproses permintaan dan menghasilkan respons. Responsnya bisa sesederhana string teks. Kemudian, kita dapat menggunakan route() decorator untuk mengikat URL ke fungsi tampilan sehingga fungsi tersebut akan dipicu saat URL dikunjungi.

route()dekorator mengambil jalur URL sebagai parameter, atau bagian dari URL yang mengikuti nama domain. Semua jalur URL harus dimulai dengan garis miring di depan.

Dalam kode tersebut kita membuat fungsi postJsonHandler yang dimana didalamnya berfungsi untuk menampilkan output True jika tipe data yang diterima berupa JSON, jika bukan JSON maka program tersebut akan mengeluarkan output false. Berikutnya terdapat variable bernama content yang berfungsi mendapatkan data JSON yang di POST melalui Postman, yang berarti, variable konten yaitu variable yang menyimpan json Ketika sudah di dapatkan dari Postman

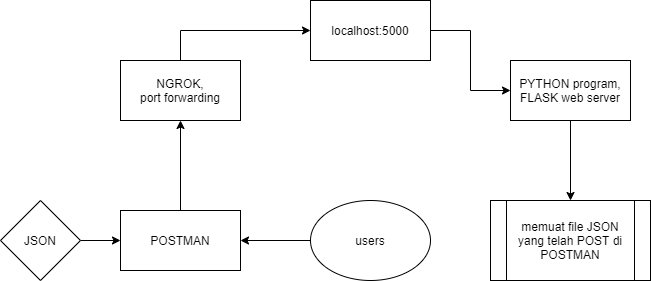
Selanjutnya terdapat variable dataframe yang berfungsi meruba tipe data dictionary atau list JSON ke dataframe pandas supaya dapat dibaca jelas oleh user. Setelah membuat variable dataframe, selanjutnya memanggilnya dengan fungsi print untuk menampilkannya di output program. Fungsi .head() di program adalah untuk menampilkan berdasarkan limit, untuk deprogram menginisiasi lima yang berarti data hanya dimuat lima saja.

@app.route('/postjson', methods=['POST'])  
def postJsonHandler():  
 print(request.is\_json)  
 content = request.get\_json()  
 dataframe = pd.DataFrame.from\_dict(content)  
 print(dataframe.head(5))  
 return 'JSON Berhasil Di Post'

melalui perintah dibawah, kita dapat mengatur port yang digunakan untuk mengakses web server flask yang telah dibuat menggunakan localhost.

app.run(host='0.0.0.0', port=5000)

1. **Alur kerja dari web service menggunakan Flask dan Postman**



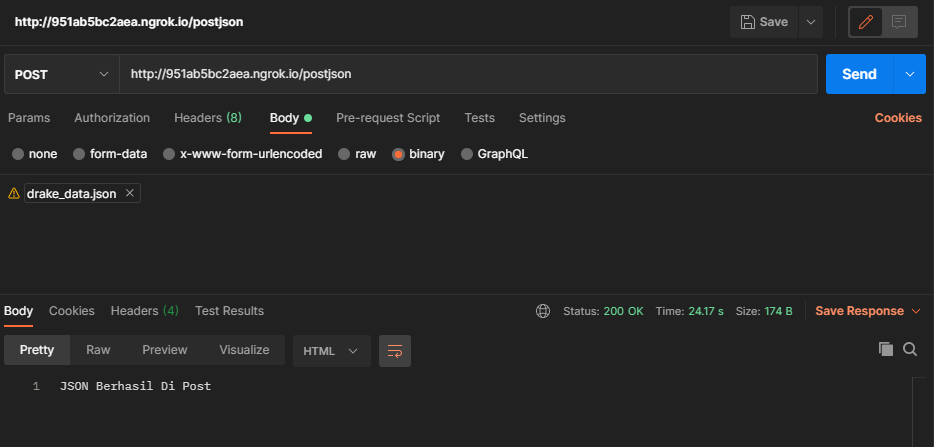
Pertama user login ke Postman dengan akunnya masing-masing, setelah itu User klik icon plus atau new disamping overview, kemudian input Alamat URL Ngrok yang telah dibuat, setelah itu rubah perintah GET ke perintah POST, kemudian arahkan ke menu Body, selanjutnya pilih binary, setelah itu select file dan pilih file JSON yang akan di POST ke Web Server flask yang telah dibuat dan telah dijalankan.

kemudian klik button send yang berwarna biru, setelah button send di klik maka postman akan mengakses Ngrok dengan alamat URL yang telah di input sebelumnya, jika Ngrok menerima dengan berhasil maka Ngrok akan melanjutkan proses ke localhost dengan port 5000.

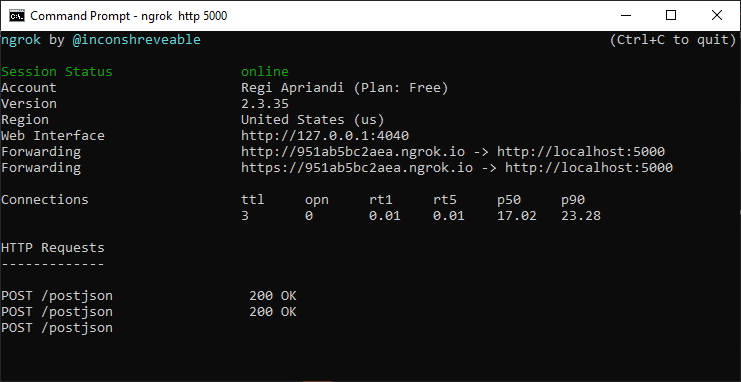
Web server flask akan otomatis menerima proses POST yang telah dilakukan oleh postman jika file yang diterima JSON maka web server akan langsung mengeksekusi file JSON tersebut dan menampilkannya dalam bentuk data frame, jika file yang diterima bukan JSON, maka web server akan menghentikan proses dan menampilkan output false dan output pada Postman yaitu “JSON Berhasil Di Post”

1. **Hasil Implementasi Program**

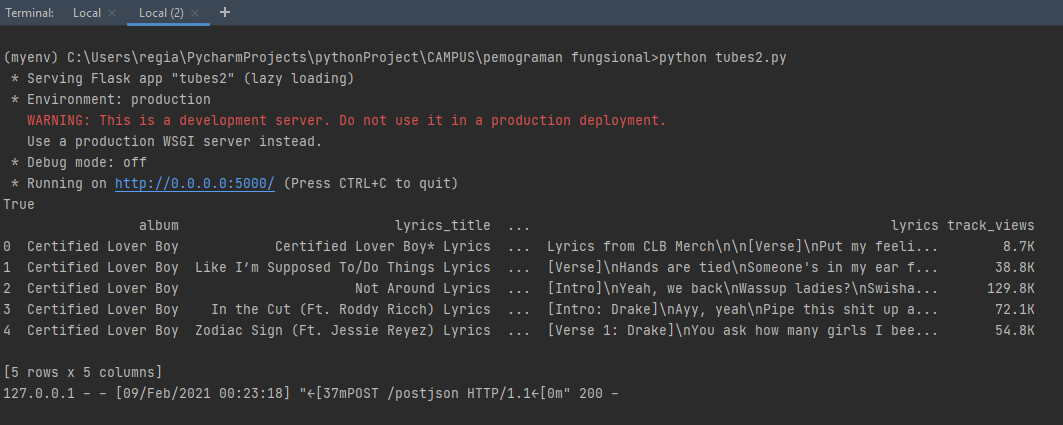
* **Postman**



* **Ngrok**



* **Output Program**



**BAB IV**

**KESIMPULAN**

**4.1 Kesimpulan**

**4.1.1 High-Order-Function**

High-Order-Function adalah fungsi yang dapat menerima parameter fungsi sebagaiargumen dan bahkan mengembalikan nilai dari fungsi. High-Order-Function sama seperti fungsi biasa dengan kemampuan tambahan untuk menerima dan mengembalikan fungsi lainnya dengan argumen dan output yang ditentukan.

**4.1.2 JSON (JAVA SCRIPT OBJECT NOTATION)**

JSON telah lama dipergunakan dalam dunia web. Tetapi selain itu, JSON dapat juga dipergunakan sebagai pengganti dari XML sebagai format penulisan data dalam file dengan beragam variasi penulisan yang anda dapat tentukan sendiri. Kelebihan JSON terletak pada kecepatan dan kemudahan penggunaanya. String dari JSON sendiri jauh lebih mudah dibaca daripada XML.

**4.2 Saran**

Terkait dengan hal tersebut, saya menyarankan beberapa hal untuk diperhatikan seperti berikut ini :

1. Program sederhana ini bisa dapat dijadikan referensi untuk pembuatan program yang lainnya menggunakan High-Order-Function
2. Dalam pengetikan sintaks pada JSON disarankan untuk teliti, karena sangat berpengaruh terhadap program jika ada kesalahan pada pengetikan.
3. Penulis juga mengharapkan kritik dan saran dalam penulisan makalah dikemudian hari.

**DAFTAR PUSTAKA**

[1] BAKTI.Bahasa Pemrograman Python Pengertian, Sejarah dan Kelebihan kekurangannya. Diakses pada tanggal 8 Februari 2021 melalui https://www.baktikominfo.id/id/informasi/pengetahuan/bahasa\_pemrograman\_python\_pengertian\_sejarah\_kelebihan\_dan\_kekurangannya-954#:~:text=Python%20adalah%20salah%20satu%20bahasa,Belanda%20bernama%20Guido%20Van%20Rossum.

[2]. Anggi Yulio P.2017.Manipulasi File CSV dengan Python.

Diakses pada tanggal 8 Februari 2021 melalui

<https://devtrik.com/python/manipulasi-file-csv-dengan-python/#:~:text=CSV%20(Comma%20Separated%20Value)%20merupakan,%2C)%20atau%20titik%20koma%20(%3B)>.

[3] Ahmad Muhardian.2019.Belajar Python Membuat fungsi dengan Lambda Expression.

Diakses pada tanggal 8 Februari 2021 melalui

<https://www.petanikode.com/python-lambda/>

[4] Kuro Neko.2020.Fungsi map() di Python.

Diakses pada tanggal 8 Februari 2021 melalui

<https://kopiding.in/fungsi-map-python/#:~:text=Fungsi%20map()%20digunakan%20untuk,mengembalikannya%20dalam%20bentuk%20objek%20map>.

[5] Kuro Neko.2020.Fungsi filter() di Python.

Diakses pada tanggal 8 Februari 2021 melalui

<https://kopiding.in/fungsi-filter-python/#:~:text=Fungsi%20filter()%20digunakan%20untuk,apakah%20bernilai%20True%20atau%20False>.

[6] Kuro Neko.2020.Fungsi sorted() di Python.

Diakses pada tanggal 8 Februari 2021 melalui

<https://kopiding.in/fungsi-sorted-python/#:~:text=key%20(opsional)%20–%20fungsi%20kunci%20yang%20digunakan%20sebagai%20pembanding%20pengurutan>.

[7] Eko Purnomo.2014.Tutorial PHP OOP part 26: Serialisasi Object.

Diakses pada tanggal 8 Februari 2021 melalui

<https://www.nulis-ilmu.com/tutorial-php-oop-part-26-serialisasi-object/>

[8] Ahmad Muhardian.2020.JSON itu apasih? Dan apa gunanya dalam pemrograman.

Diakses pada tanggal 8 Februari 2021 melalui

<https://www.petanikode.com/json-pemula/>

[9] Ahmad Muhardian.2020.JSON itu apasih? Dan apa gunanya dalam pemrograman.

Diakses pada tanggal 8 Februari 2021 melalui

<https://www.petanikode.com/json-pemula/>

[10] Faisal Al Isfahani.2020.Web Service, Pengertian, Sejarah dan Kegunaan.

Diakses pada tanggal 8 Februari 2021 melalui

<https://www.researchgate.net/publication/339662253_Web_Service_Pengertian_Sejarah_dan_Ke> gunaan

[11] NovanKurniawan.2020.Postman.

Diakses pada tanggal 8 Februari 2021 melalui

<https://medium.com/@novancimol12/postman-4f181d625fe1>

[12] Sekolahkoding.2020.Belajar Flask.

Diakses pada tanggal 8 Februari 2021 melalui

<https://sekolahkoding.com/belajar/flask>

[13] Alfin Chandra.2019.Catatan Laravel::laravel + ngrok.

Diakses pada tanggal 8 Februari 2021 melalui

https://alfinchandra4.medium.com/catata-laravel-laravel-ngrok-6f614d0b32db#:~:text=Ngrok%20adalah%20proxy%20server%20untuk,di%20laptop%20temen-temen%20aja