PYTHON MODUL 4

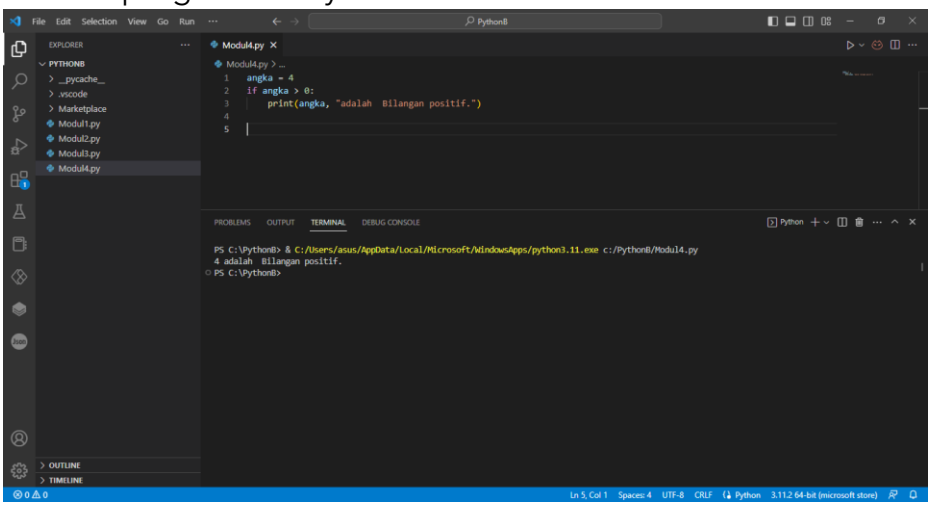
A. Percabangan

Percabangan yaitu cara yang digunakan untuk mengambil keputusan apabila di dalam program terdapat kondisi tertentu. Jumlah kondisinya bisa satu, dua atau lebih terdapat 3 jenis pernyataan yang digunakan untuk percabangan, yaitu if, if else dan if elif else

1. Pernyataan if

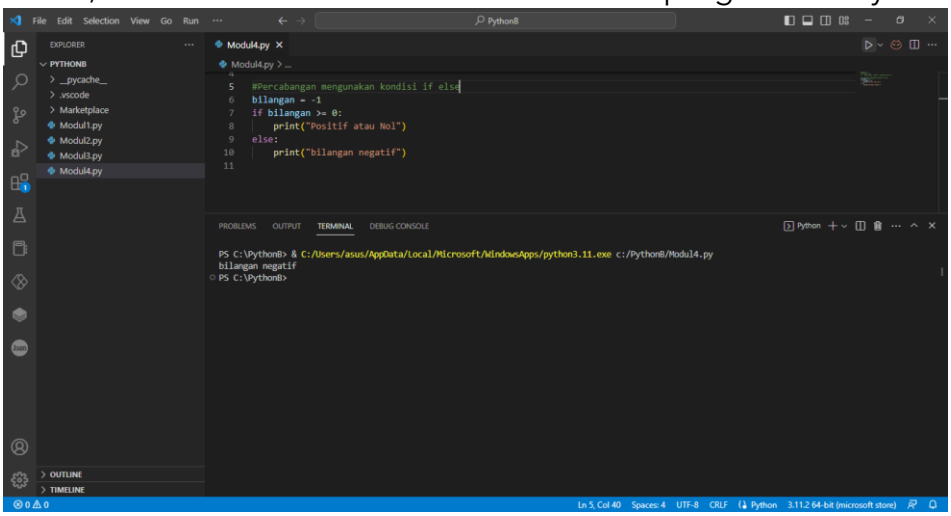
   Pernyataan if menjalankan satu buah kondisi bila hasilnya benar maka pernyataan di dalam blok if tersebut dieksekusi jika salah, maka pernyataan tidak dieksekusi.

   Contoh program Pernyataan if :



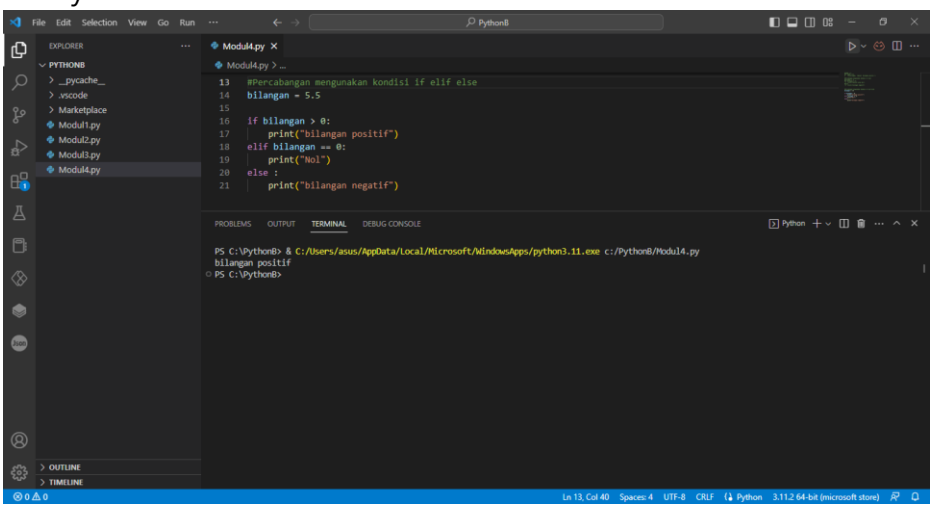
2. Pernyataan if else

   Pernyataan if else menjalankan dua kondisi. Kondisi pertama kalau benar, dan kondisi kedua kalau salah. Contoh program Pernyataan if else:



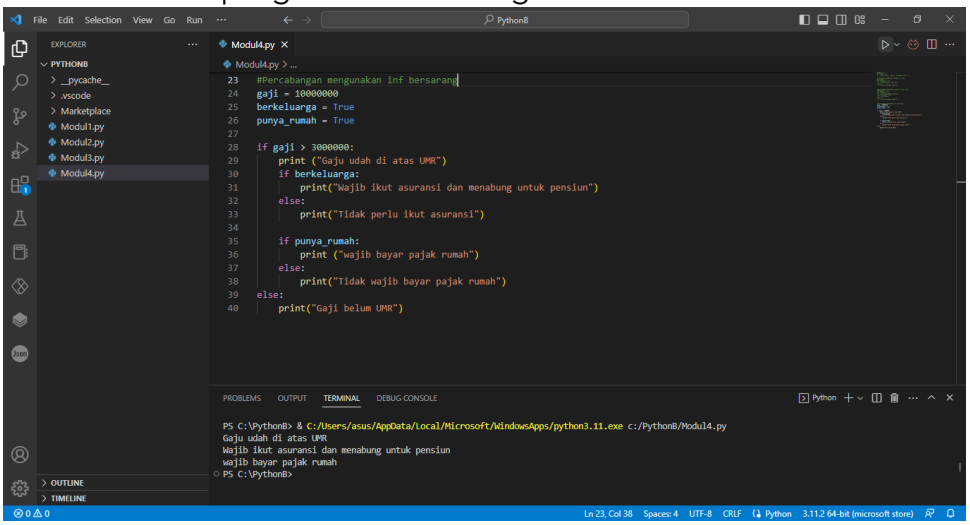
3. Pernyataan if elif else

Pernyataan if elif else digunakan untuk menjalankan lebih dari dua kondisi. Bila kondisi pada if benar, maka pernyataan di dalamnya yang dieksekusi. Bila salah, maka masuk ke kondisi elif. Dan bila tidak ada if atau elif yang benar, maka yang dijalankan adalah yang else. Contoh Pernyataan if elif else :



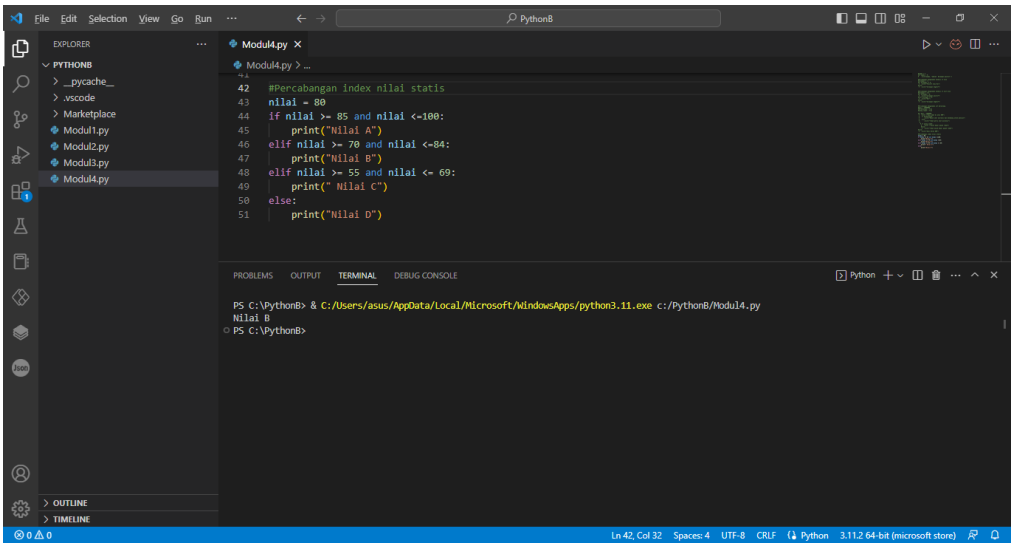
4. Tambahan : if Bersarang

Sebuah kondisional dapat disimpan di dalam if lain. Berikut ini adalah contoh program if bersarang :



5. Program Percabangan Indeks Nilai Statis

Ketentuan : Nilai 85 s/d 100 indeks A, nilai 70 s/d 84 indeks B, nilai 55 s/d 69 indeks C, nilai dibawah 55 indeks D.Contoh Program Percabangan Indeks Nilai Statis :



B. Perulangan

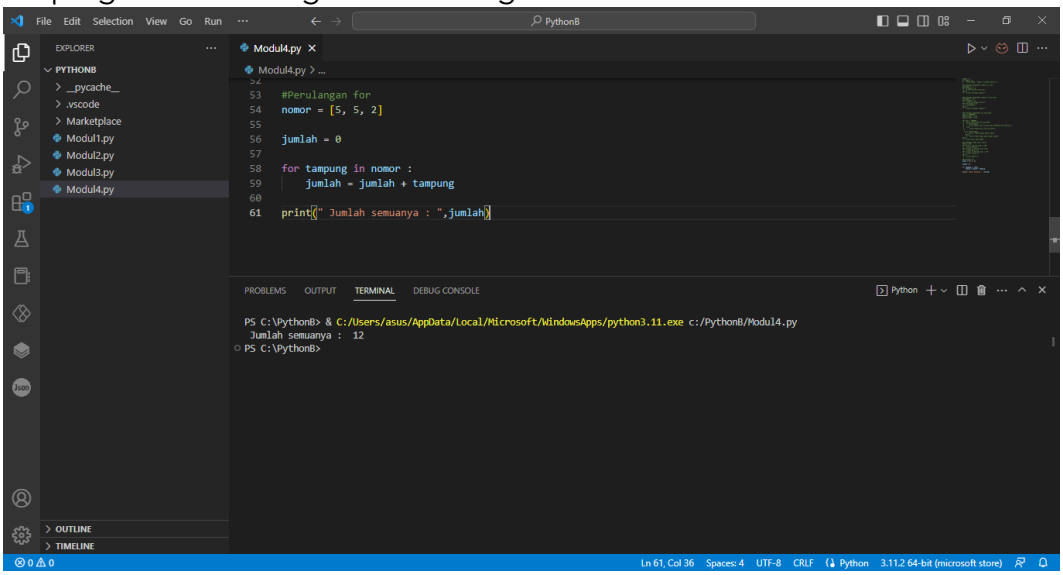
Secara umum, python mengeksekusi program baris perbaris. Mulai dari baris satu, dua, dan seterusnya. perulangan bisa dilakukan dengan dua cara atau metode, yaitu menggunakan for dan menggunakan while.

1. Perulangan dengan Menggunakan for

Perulangan dengan menggunakan for memiliki sintaks seperti berikut :

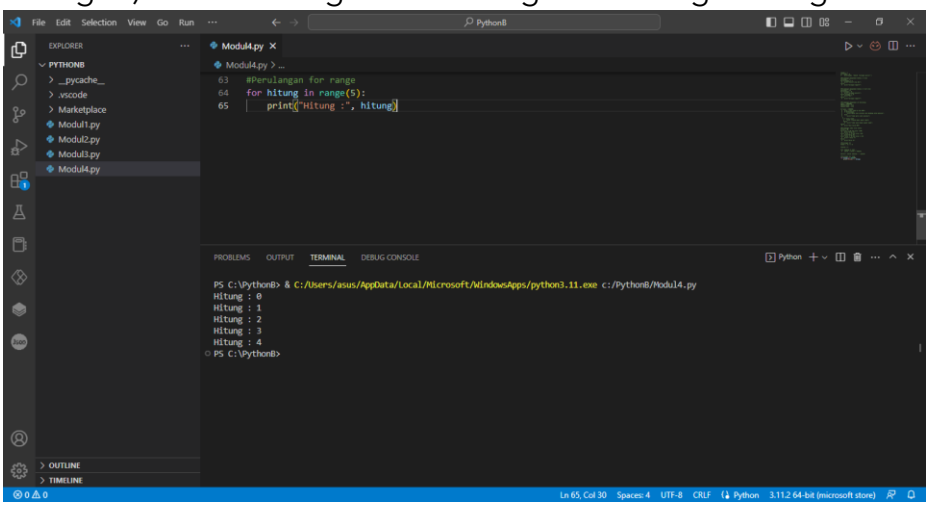
for var in sequence: body of

for var adalah variabel digunakan untuk menampung sementara nilai dari sequence pada saat terjadi perulangan. Sequence adalah tipe data berurut seperti string, list, dan tuple. Perulangan terjadi sampai looping mencapai elemen atau anggota terakhir dari sequence. Bila loop sudah sampai ke elemen terakhir dari sequence, maka program akan keluar dari looping. Contoh Program Perulangan for :



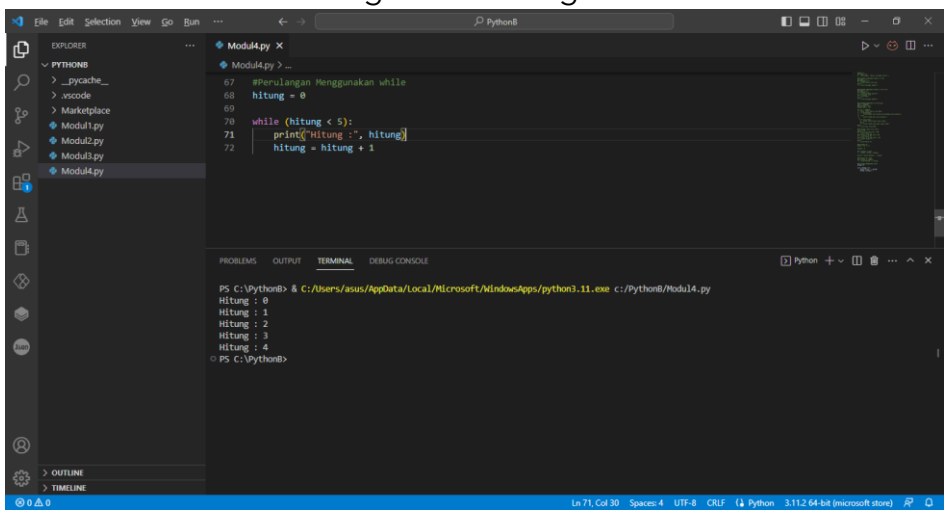
2. Perulangan for dengan range

Fungsi range() digunakan untuk menghasilkan deret bilangan. range(10) akan menghasilkan bilangan dari 0 sampai dengan 9 (10 bilangan). Contoh Program Perulangan for dengan Range :



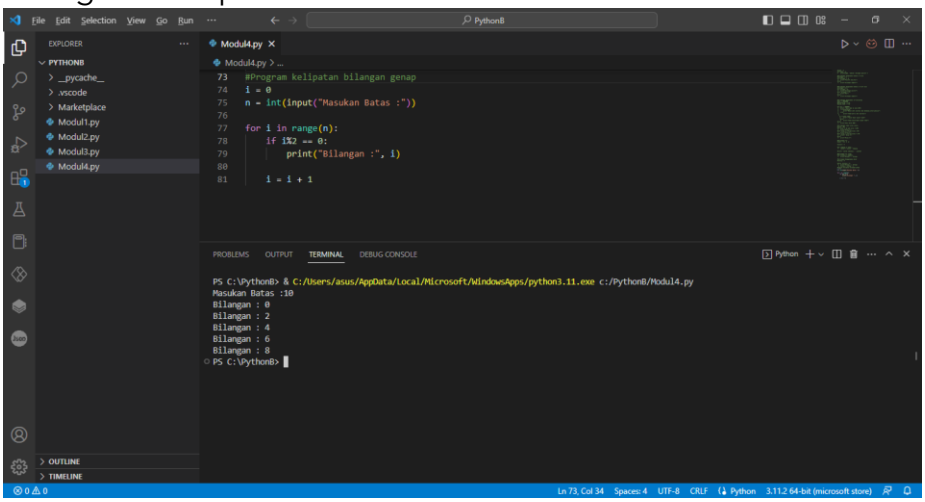
3. Perulangan Menggunakan while

Perulangan menggunakan while akan menjalankan blok pernyataan terus menerus selama kondisi bernilai benar. Bila kondisi salah, maka program akan keluar dari while dan lanjut ke baris pernyataan di luar while. Contoh Program Perulangan while :



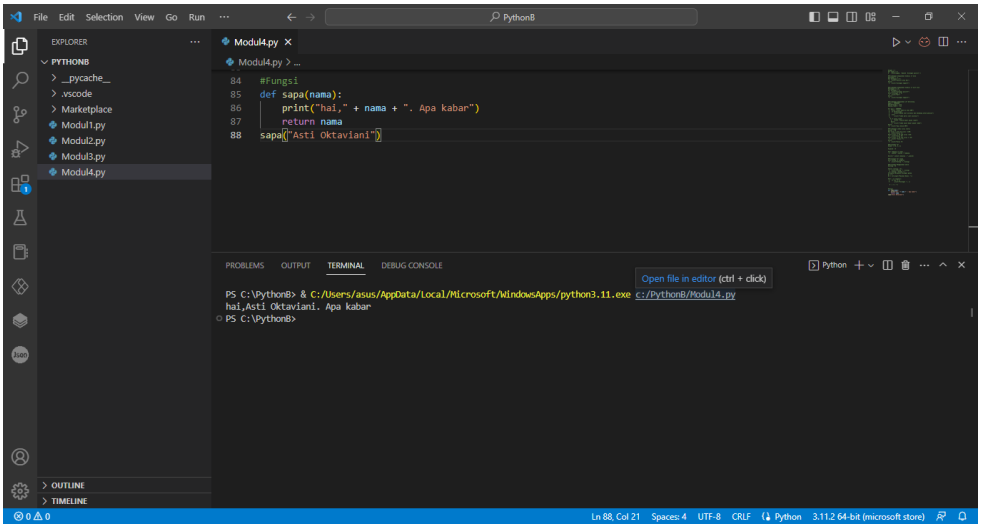
4. Program Kelipatan Bilangan Genap

Program pengulangan dengan for. Contoh Program Kelipatan Bilangan Genap :



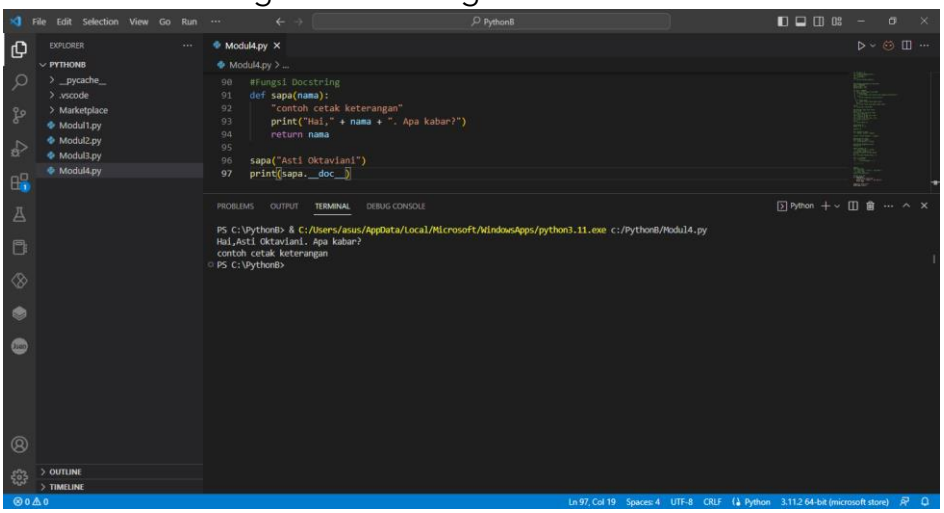
C. Fungsi

Fungsi adalah grup/blok program untuk melakukan tugas tertentu yang berulang. Fungsi membuat kode program menjadi reusable, artinya hanya di definisikan sekali saja, dan kemudian bisa digunakan berulang kali dalam program. Contoh Program Fungsi :

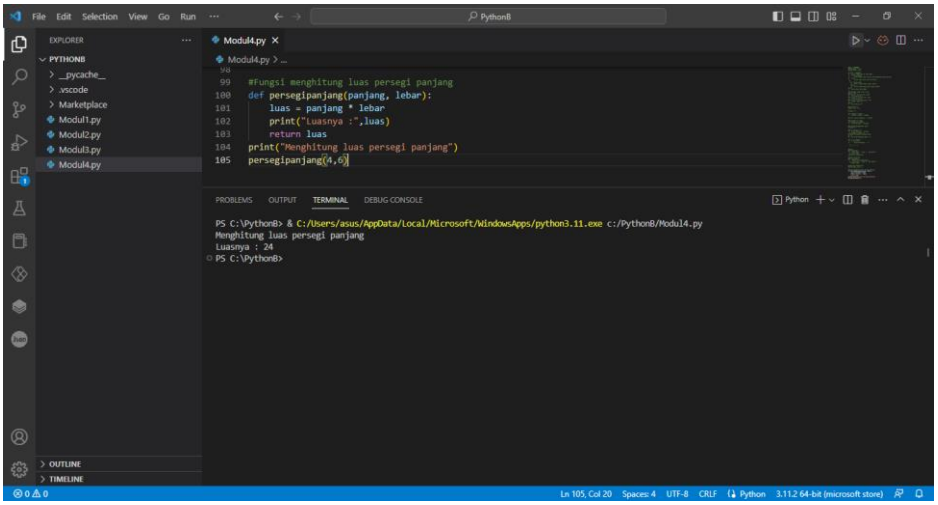


1. Docstring

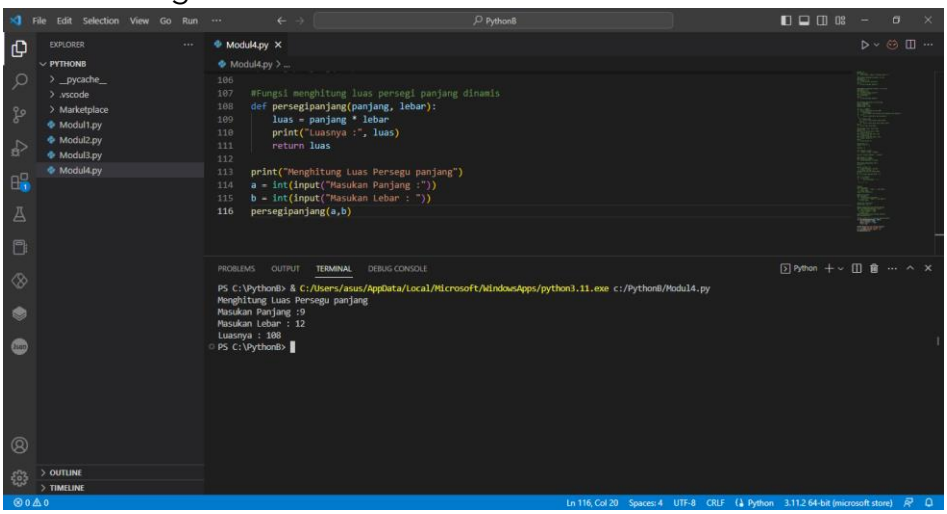
   Docstring adalah singkatan dari documentation string berfungsi sebagai dokumentasi atau keterangan singkat tentang fungsi yang kita buat. Contoh Program Docstring



2. Contoh Program Luas Persegi Panjang dengan Fungsi. Contoh Program Statis



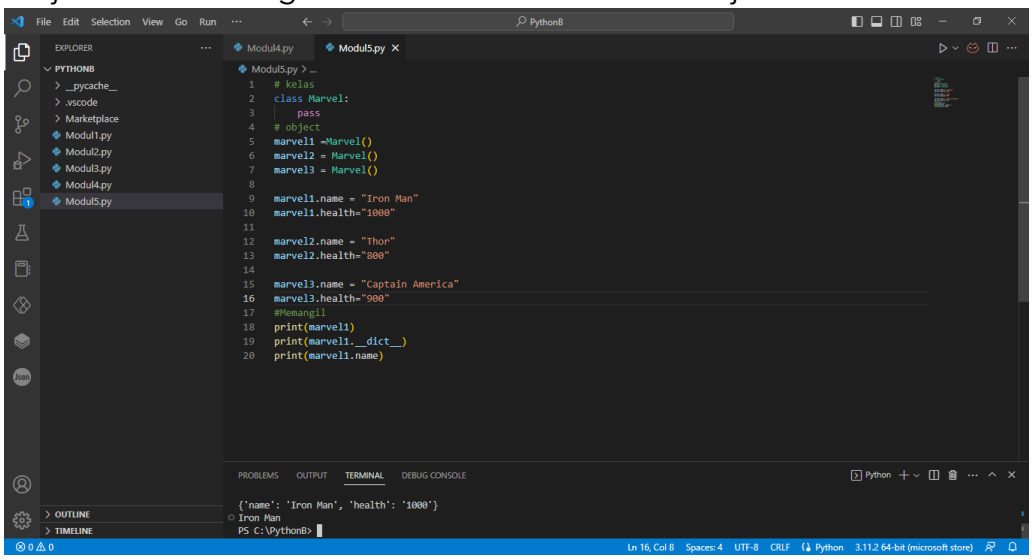
Contoh Program Dinamis :



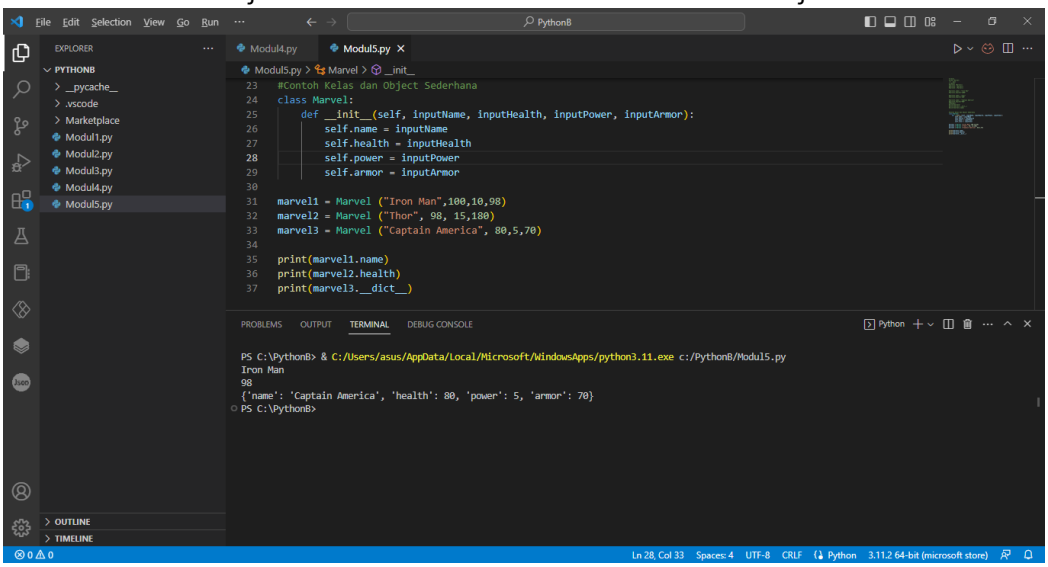
PYTHON MODUL 5

A. Object Oriented Programming

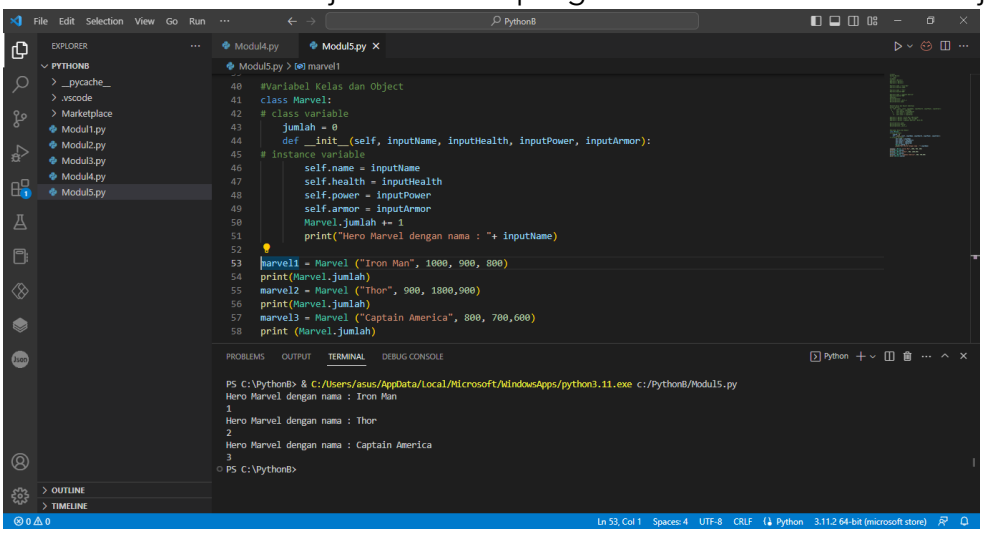
Object Oriented Programming (OOP) merupakan suatu konsep pemrograman yang menekankan pada paradigma atau cara pandang terhadap suatu masalah berdasarkan object. Istilah dalam OOP yaitu sebagai berikut : Kelas, Variabel, Data member. Overloading, Overloading Operator, Variabel, Pewarisan/Inheritansi (Inheritance),Instance, Instansiasi , Metode, Objek. Contoh Program Perkenalan Kelas dan Object



1. Kelas dan Object Sederhana. Contoh Kelas dan Object Sederhana



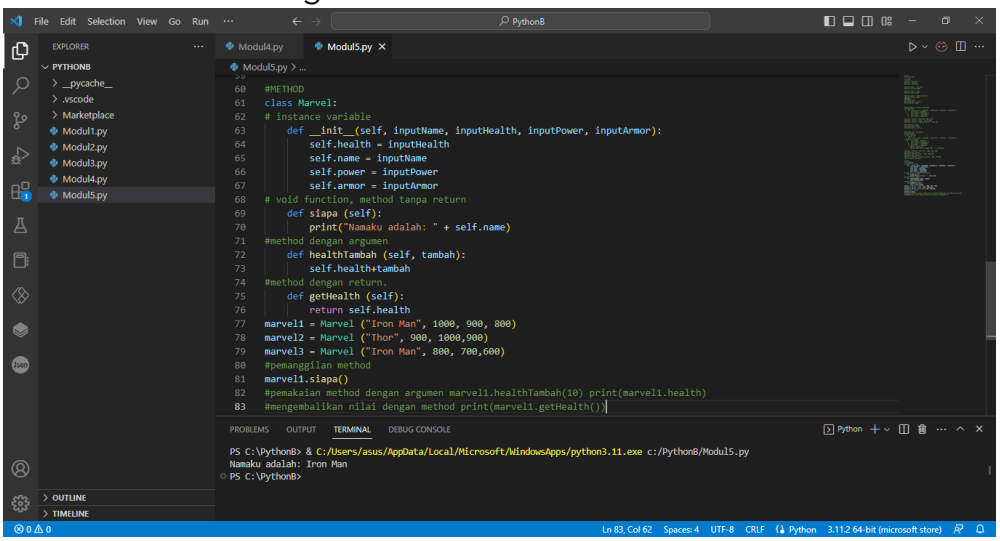
2. Variabel Kelas dan Object. Contoh program Variabel Kelas dan Object



```
#Variabel Kelas dan Object
class Marvel:
# class variable
    jumlah = 0
    def __init__(self, inputName, inputHealth, inputPower, inputArmor):
# instance variable
        self.name = inputName
        self.health = inputHealth
        self.power = inputPower
        self.armor = inputArmor
        Marvel.jumlah += 1
        print("Hero Marvel dengan nama : "+ inputName)

marvel1 = Marvel ("Iron Man", 1000, 900, 800)
print(Marvel.jumlah)
marvel2 = Marvel ("Thor", 900, 1800,900)
print(Marvel.jumlah)
marvel3 = Marvel ("Captain America", 800, 700,600)
print (Marvel.jumlah)
```

```
PS C:\PythonB> & C:/Users/asus/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/PythonB/Modul5.py
Hero Marvel dengan nama : Iron Man
1
Hero Marvel dengan nama : Thor
2
Hero Marvel dengan nama : Captain America
3
PS C:\PythonB>
```

3. Method. Contoh Program Method

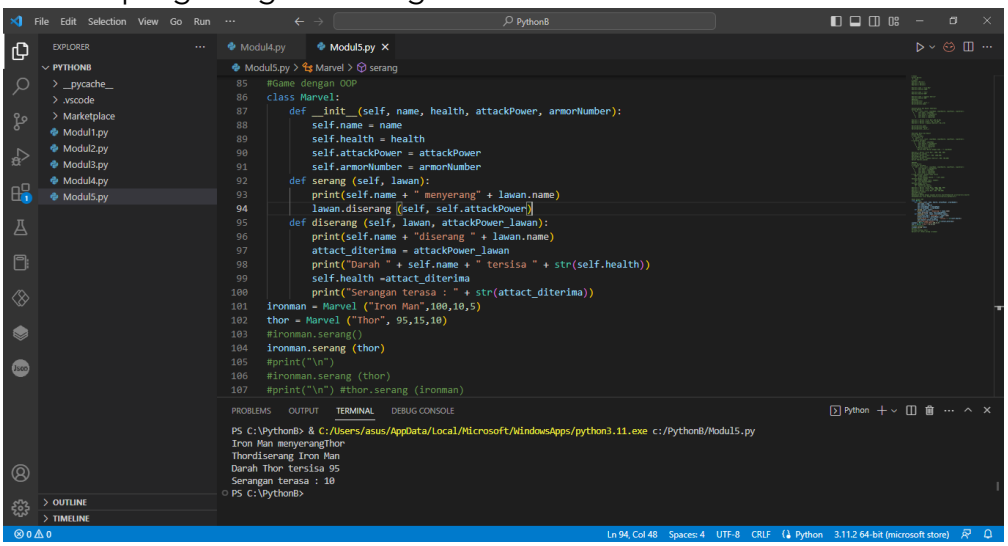


```
#METHOD
class Marvel:
# instance variable
    def __init__(self, inputName, inputHealth, inputPower, inputArmor):
        self.health = inputHealth
        self.name = inputName
        self.power = inputPower
        self.armor = inputArmor
# void function, method tanpa return
    def siapa (self):
        print("Namaku adalah: " + self.name)
#method dengan argumen
    def healthTambah (self, tambah):
        self.health+tambah
#method dengan return.
    def getHealth (self):
        return self.health
marvel1 = Marvel ("Iron Man", 1000, 900, 800)
marvel2 = Marvel ("Thor", 900, 1000,900)
marvel3 = Marvel ("Iron Man", 800, 700,600)
#pemanggilan method
marvel1.siapa()
#pemakaian method dengan argumen marvel1.healthTambah(10) print(marvel1.health)
#mengembalikan nilai dengan method print(marvel1.getHealth())
```

```
PS C:\PythonB> & C:/Users/asus/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/PythonB/Modul5.py
Namaku adalah: Iron Man
PS C:\PythonB>
```

4. Game dengan OOP

Contoh program game dengan OOP :



```
#Game dengan OOP
class Marvel:
    def __init__(self, name, health, attackPower, armorNumber):
        self.name = name
        self.health = health
        self.attackPower = attackPower
        self.armorNumber = armorNumber
    def serang (self, lawan):
        print(self.name + " menyerang" + lawan.name)
        lawan.diserang (self, self.attackPower)
    def diserang (self, lawan, attackPower_lawan):
        print(self.name + "diserang " + lawan.name)
        attact_diterima = attackPower_lawan
        print("Darah " + self.name + " tersisa " + str(self.health))
        self.health =attact_diterima
        print("Serangan terasa : " + str(attact_diterima))
ironman = Marvel ("Iron Man",100,10,5)
thor = Marvel ("Thor", 95,15,10)
#ironman.serang()
ironman.serang (thor)
#print("\n")
#ironman.serang (thor)
#print("\n") #thor.serang (ironman)
```

```
PS C:\PythonB> & C:/Users/asus/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/PythonB/Modul5.py
Iron Man menyerangThor
Thordiserang Iron Man
Darah Thor tersisa 95
Serangan terasa : 10
PS C:\PythonB>
```

MODUL 6

PYTHON (FOR DATA SCIENCE)

A. Data Science

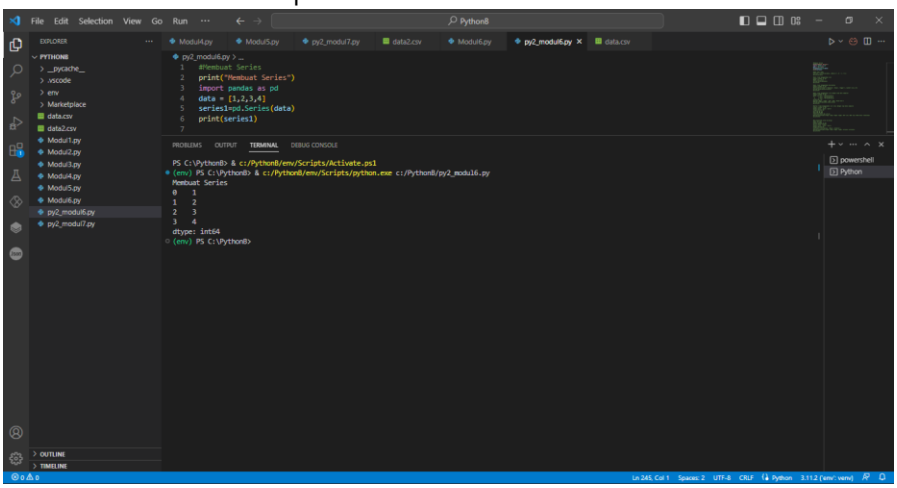
Adalah ilmu yang secara khusus mempelajari soal data terutama data kuantitatif atau data numerik. Secara umum data science adalah penggalian atau bisa juga disebut mengekstrak data agar dapat difilter serta didapatkan data yang benar untuk menghasilkan produk data yang sebenar-benarnya.

B. Pandas DataFrame

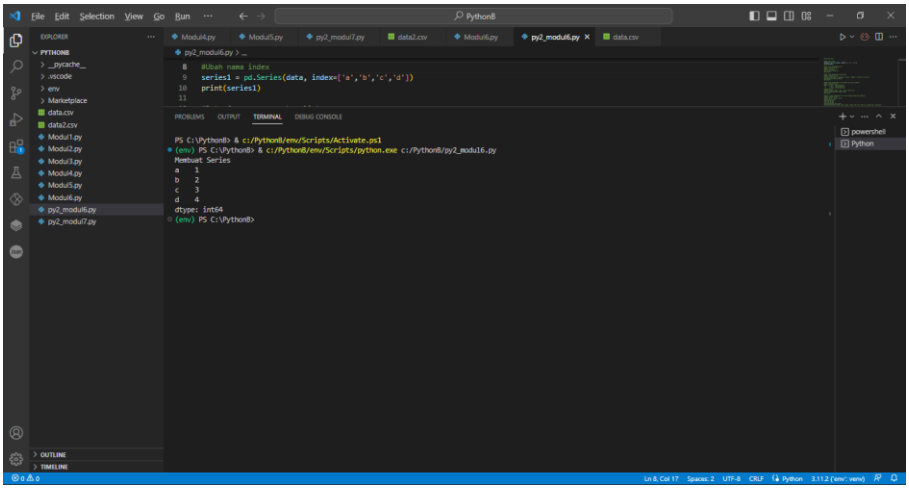
1. Pandas

Berasal dari kata python data analysis library mendukung data multi-dimensi yang artinya menggunakan 2 buah index. Sedangkan data satu dimensi adalah elemen pada data dapat diakses hanya dengan 1 buah index.

a) Series merupakan struktur data dasar dalam Pandas yang dapat berisi tipe data seperti integer, string, dan lain-lain, yang mendukung tipe data sama atau campuran. Contoh Membuat Series

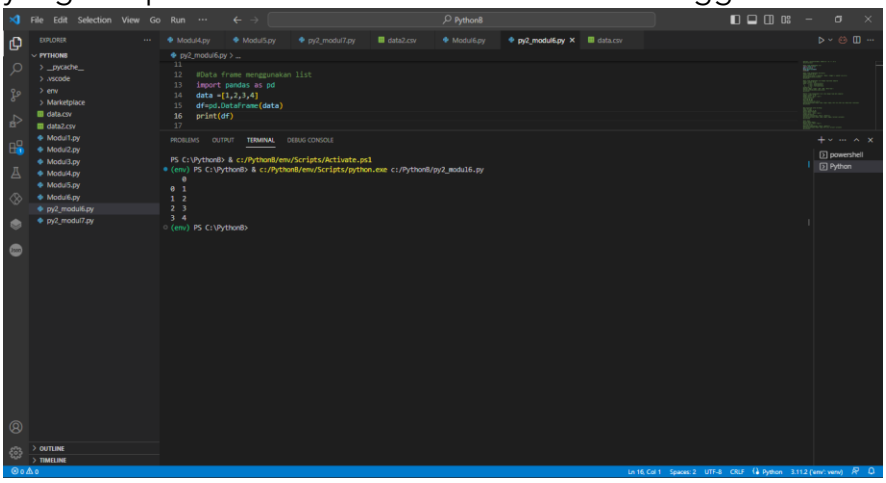


b) Ubah Nama Index. Contoh

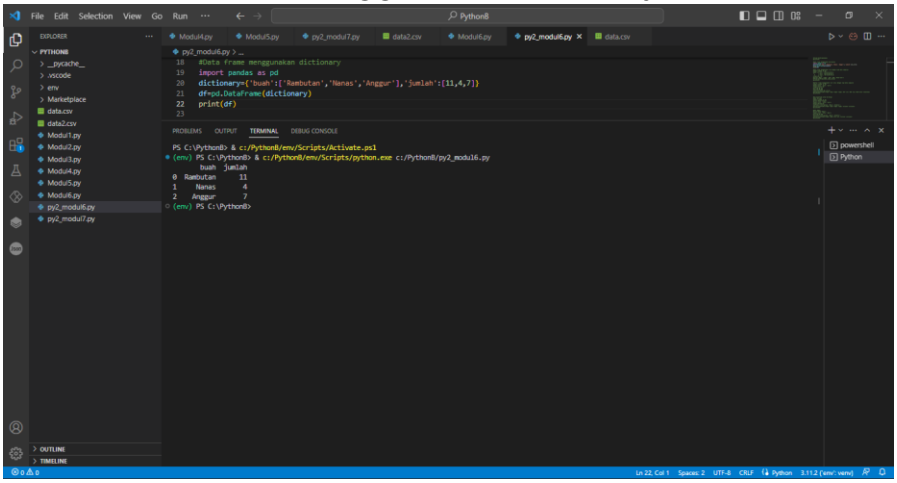


c) DataFrame

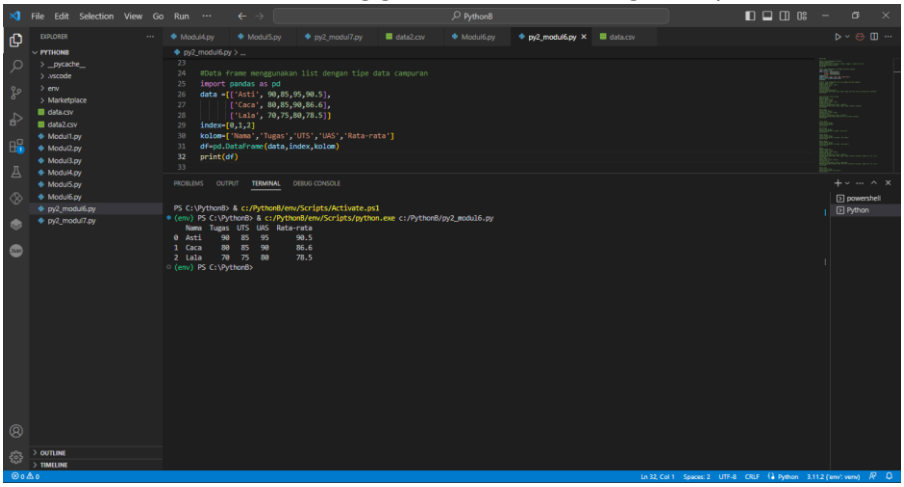
Merupakan array dua dimensi dengan baris dan kolom. DataFrame merupakan tabel/data tabular. Setiap kolom pada DataFrame merupakan objek dari Series, dan baris terdiri dari elemen yang ada pada Series. Contoh DataFrame Menggunakan List



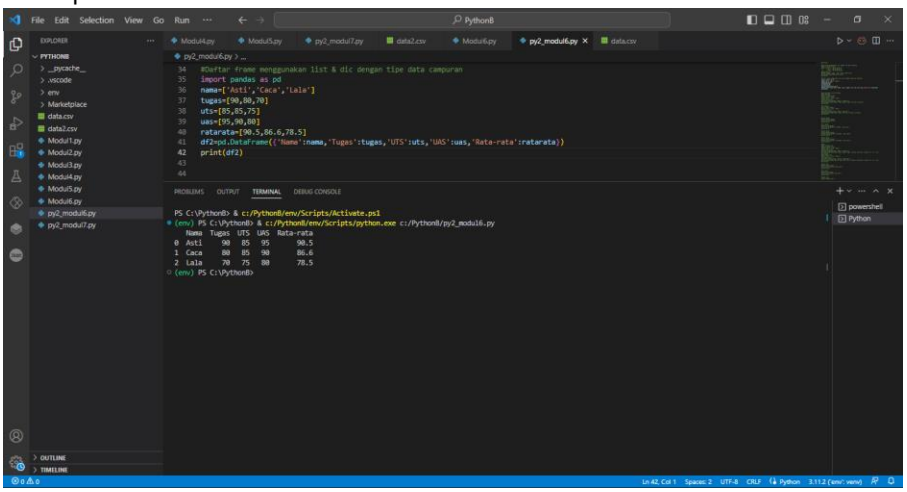
d) Contoh DataFrame Menggunakan Dictionary



e) Contoh DataFrame Menggunakan List dengan Tipe Data Campuran



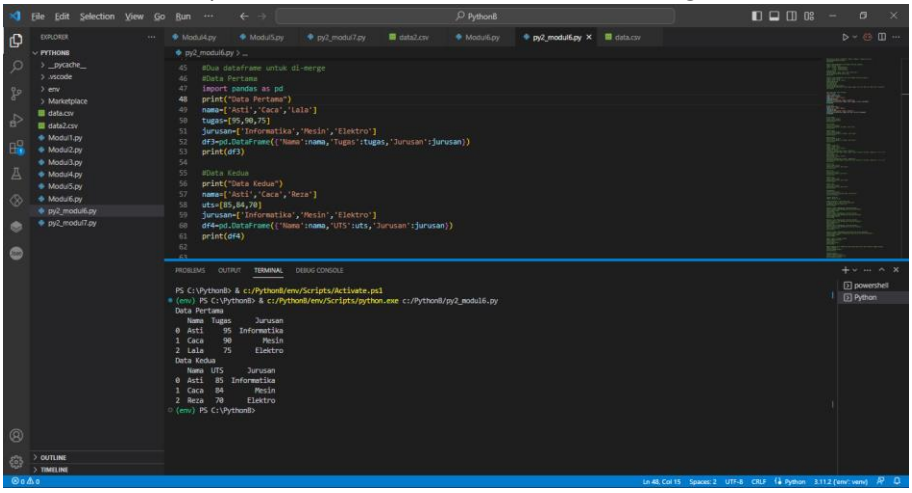
f) Contoh DataFrame Menggunakan List & Dictionary dengan Tipe Data Campuran



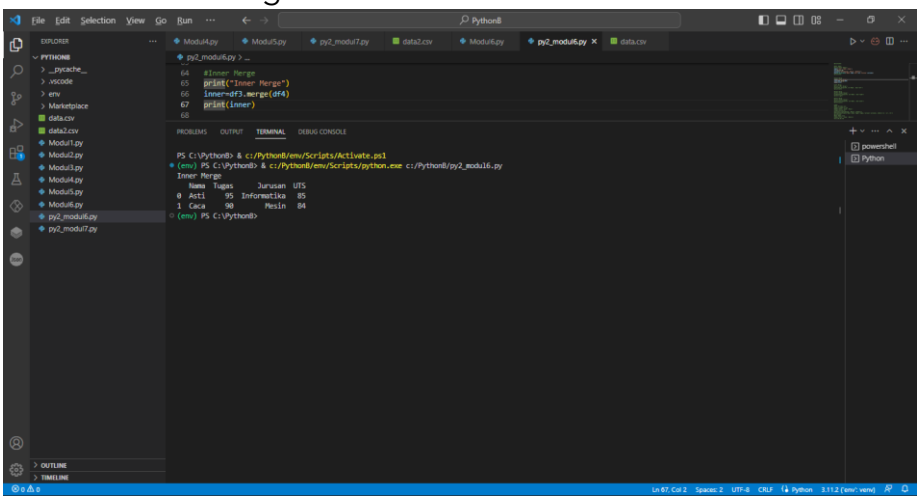
2. Merge, Join, & Concatenate DataFrame

Merupakan operasi penggabungan.

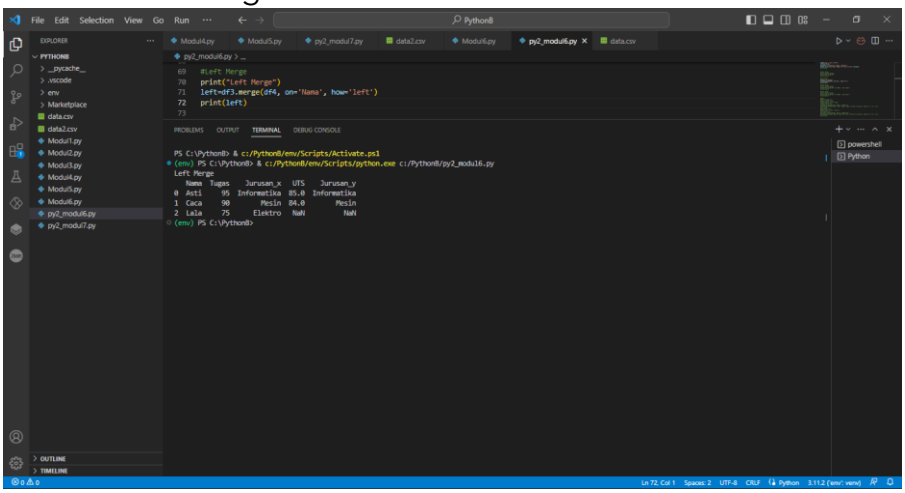
a) Merge adalah operasi penggabungan antara DataFrame
Contoh Persiapkan 2 Data Frame Untuk di Marge



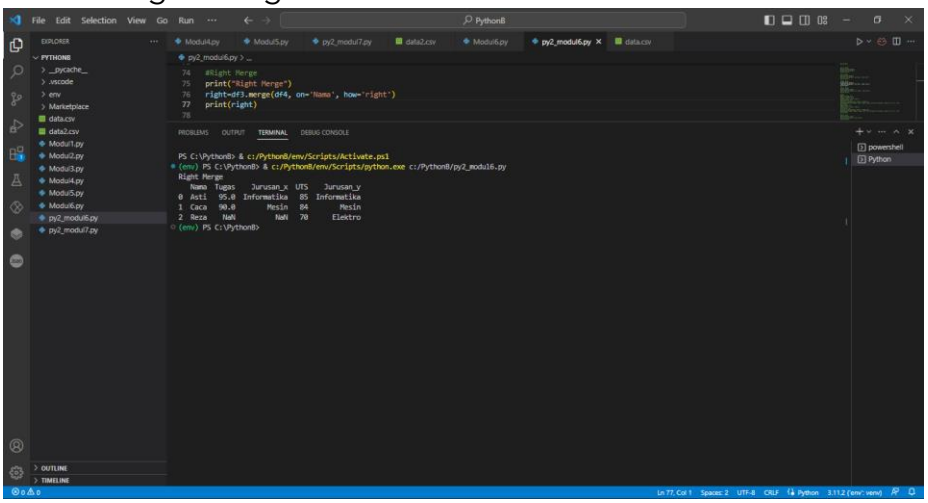
b) Contoh Inner Merge



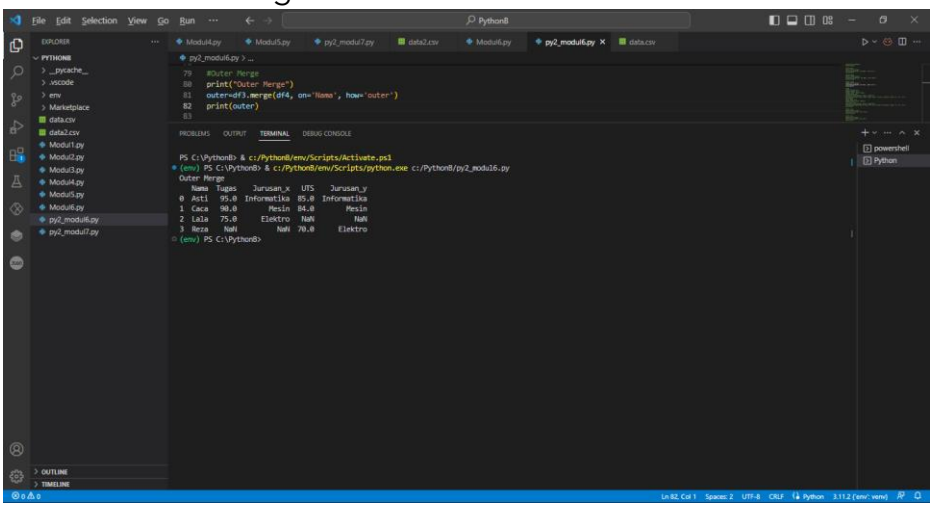
c) Contoh Left Merge



d) Contoh Right Merge



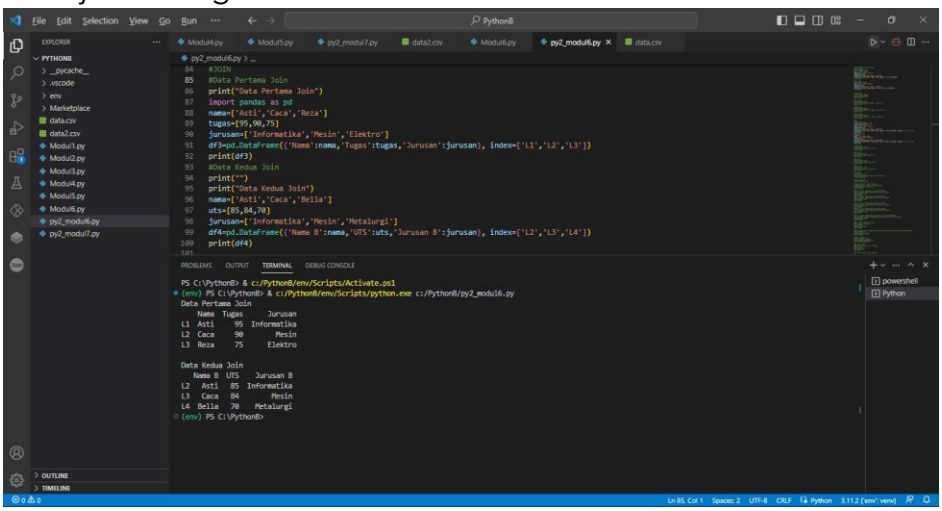
e) Contoh Outer Merge



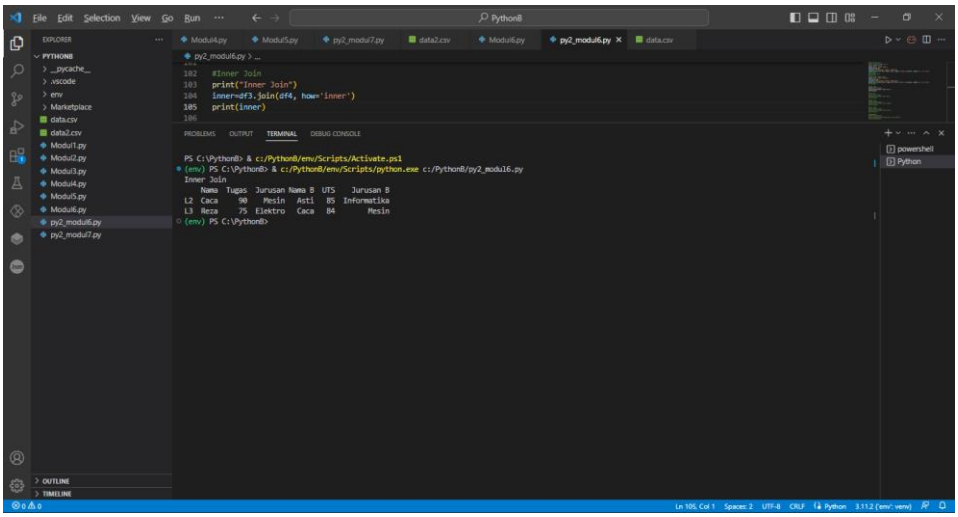
3. Join

Adalah operasi penggabungan dengan menggunakan index. Siapkan Dua DataFrame untuk di-Join .

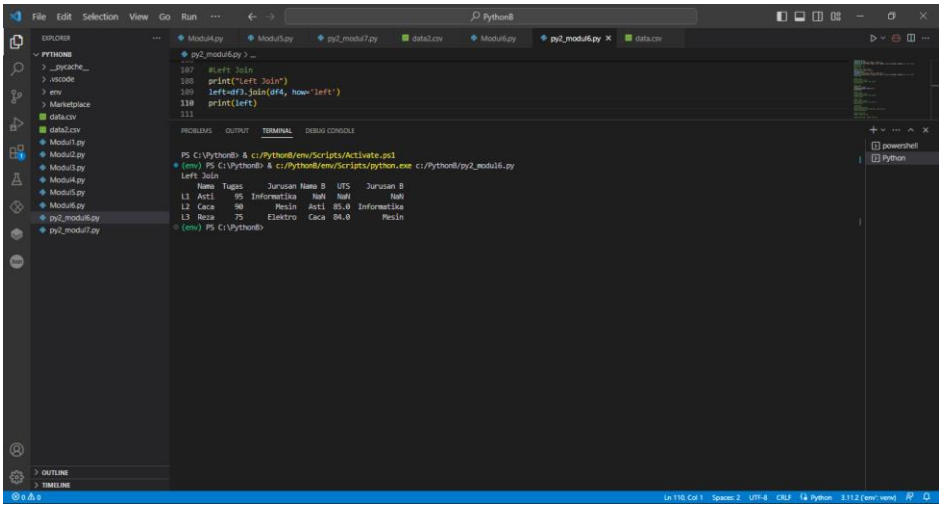
a) Data join sebagai berikut :



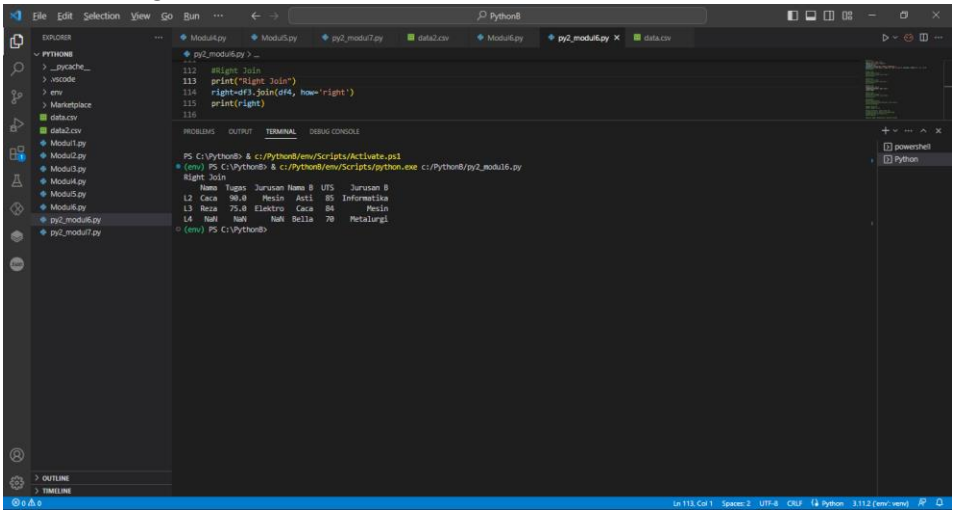
b) Contoh Inner Join



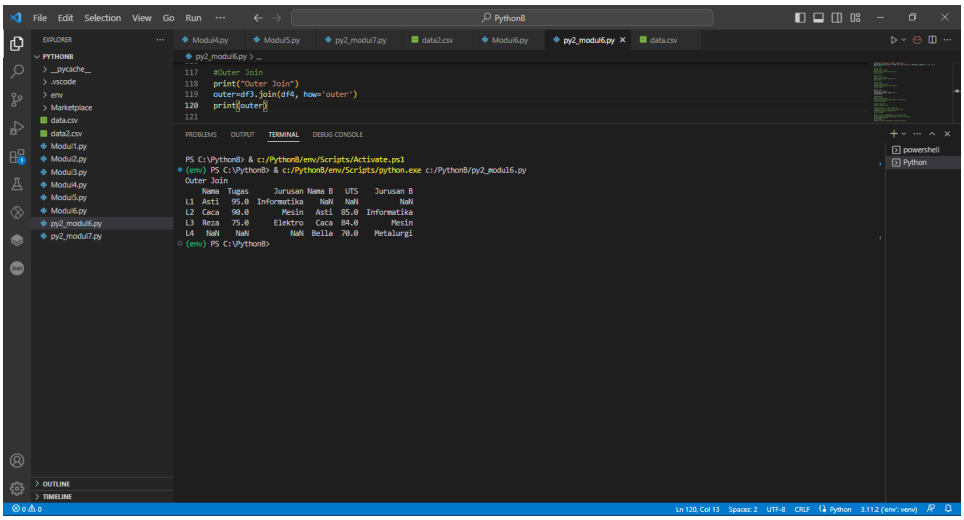
c) Contoh Left Join



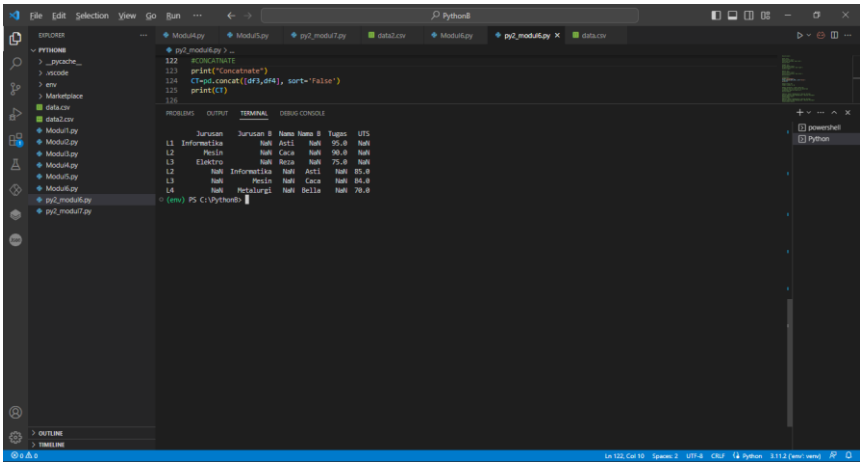
d) Contoh Right Join



e) Contoh Outer Join



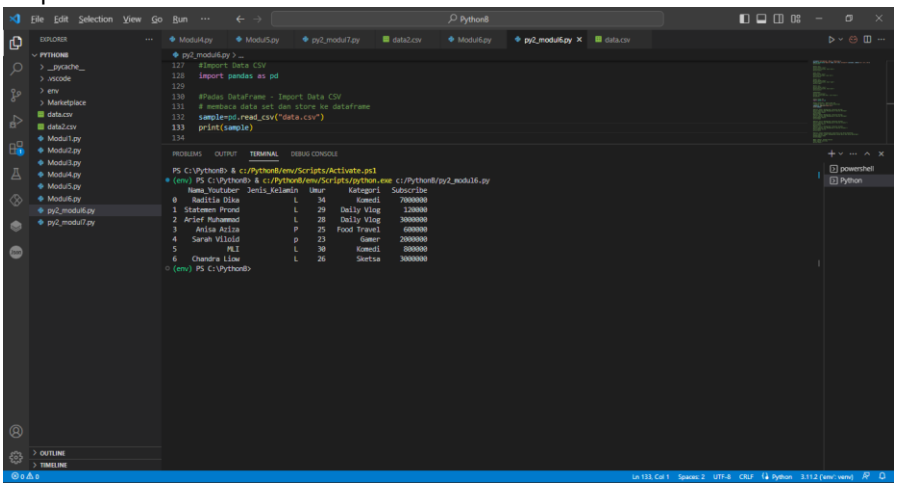
f) Concatenate adalah operasi penggabungan objek DataFrame secara vertical. Contoh



4. Pandas DataFrame - Import Data CSV

Pada tahap ini akan dibahas mengenai cara import data CSV ke dalam Panda DataFrame dan mengolah datanya.

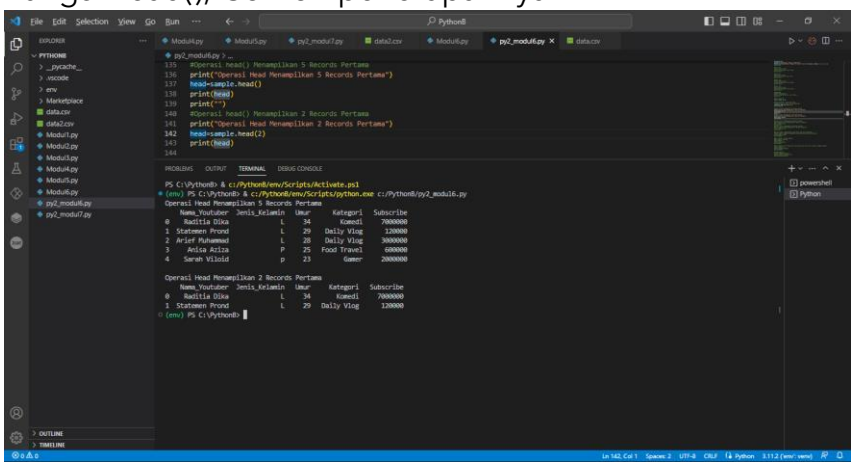
a) Import Data CSV



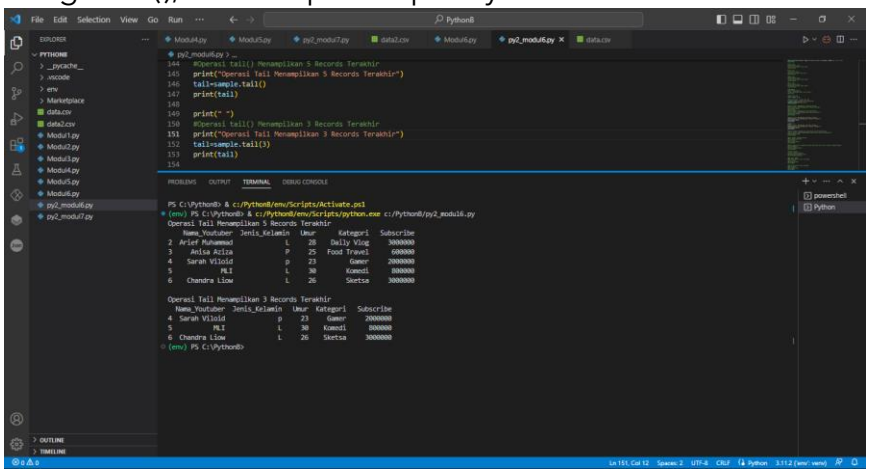
b) Macam - Macam Operasi

Fungsi-fungsi yang dapat dilakukan oleh Pandas DataFrame adalah sebagai berikut.

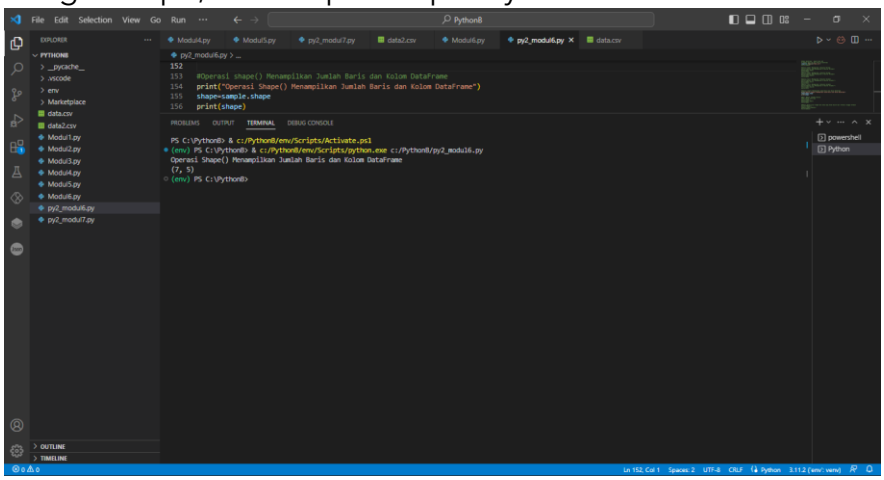
1) Fungsi head(), Contoh penerapannya



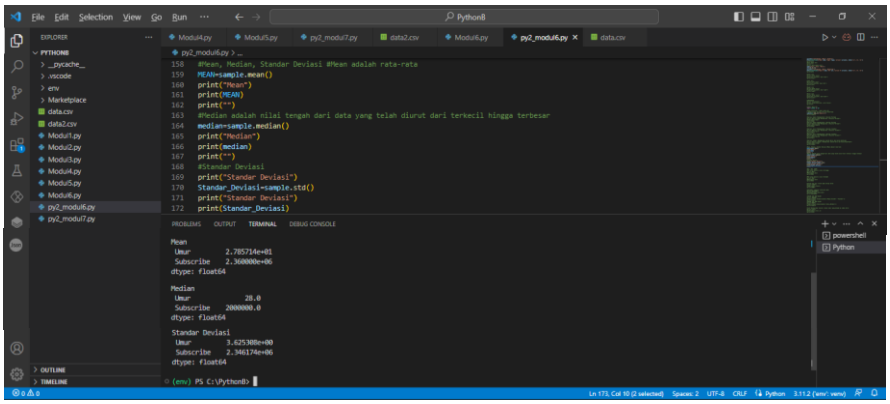
2) Fungsi tail(), Contoh penerapannya



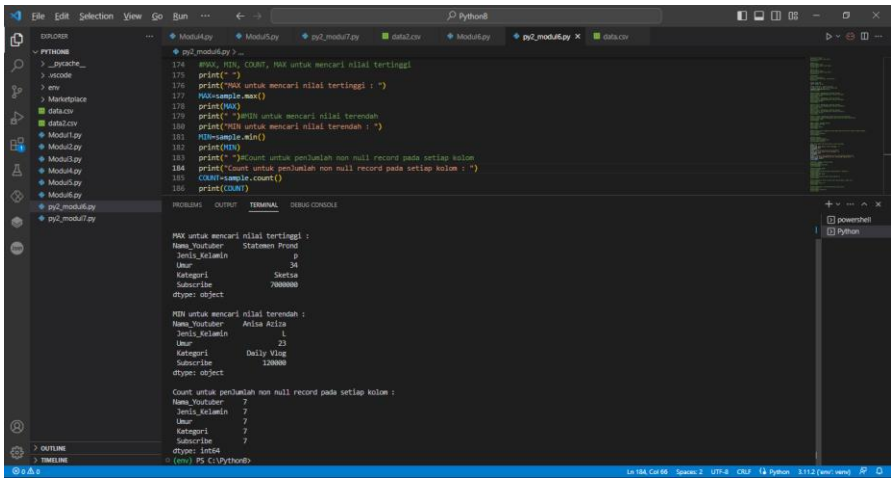
3) Fungsi shape, contoh penerapannya



4) Contoh Mean, Median, Standar Deviasi



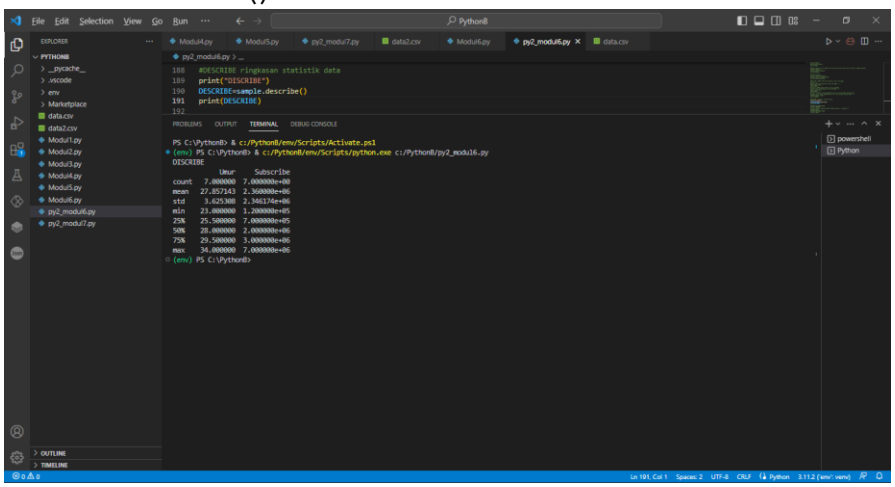
5) Contoh Max, Min, Count



```
#MAX, MIN, COUNT, MAX untuk mencari nilai tertinggi
print(" ")
print("MAX untuk mencari nilai tertinggi : ")
MAX=sample.max()
print(MAX)
print(" ")#MIN untuk mencari nilai terendah
print("MIN untuk mencari nilai terendah : ")
MIN=sample.min()
print(MIN)
print(" ")#Count untuk penJumlah non null record pada setiap kolom
print("Count untuk penJumlah non null record pada setiap kolom : ")
COUNT=sample.count()
print(COUNT)
```

```
MAX untuk mencari nilai tertinggi :
Nama_Youtuber     Statemen Prond
 Jenis_Kelamin                 p
 Umur                         34
 Kategori                 Sketsa
 Subscribe               7000000
dtype: object

MIN untuk mencari nilai terendah :
Nama_Youtuber     Anisa Aziza
 Jenis_Kelamin               L
 Umur                       23
 Kategori          Daily Vlog
 Subscribe             120000
dtype: object

Count untuk penJumlah non null record pada setiap kolom :
Nama_Youtuber    7
 Jenis_Kelamin   7
 Umur            7
 Kategori        7
 Subscribe       7
dtype: int64
(env) PS C:\Python8>
```

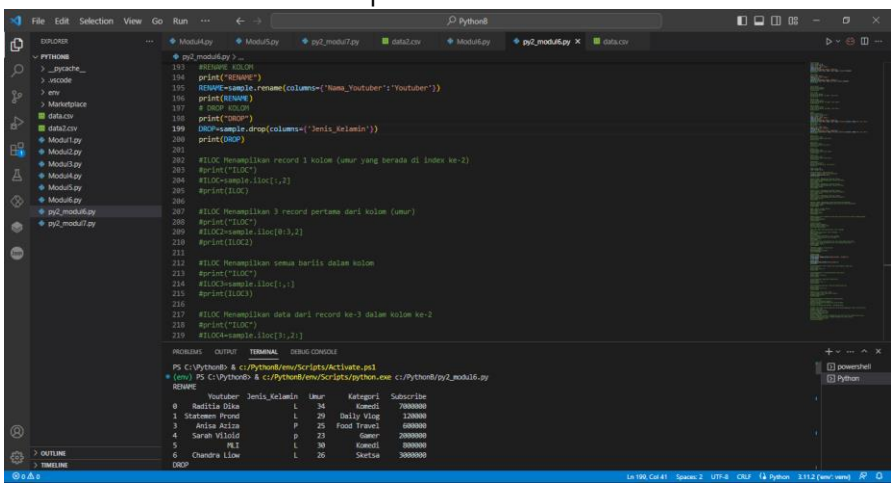
6) Contoh describe()



```
#DESCRIBE ringkasan statistik data
print("DISCRIBE")
DESCRIBE=sample.describe()
print(DESCRIBE)
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul6.py
DISCRIBE
              Umur     Subscribe
count   7.000000  7.000000e+00
mean   27.857143  2.360000e+06
std     3.625308  2.346174e+06
min    23.000000  1.200000e+05
25%    25.500000  7.000000e+05
50%    28.000000  2.000000e+06
75%    29.500000  3.000000e+06
max    34.000000  7.000000e+06
(env) PS C:\Python8>
```

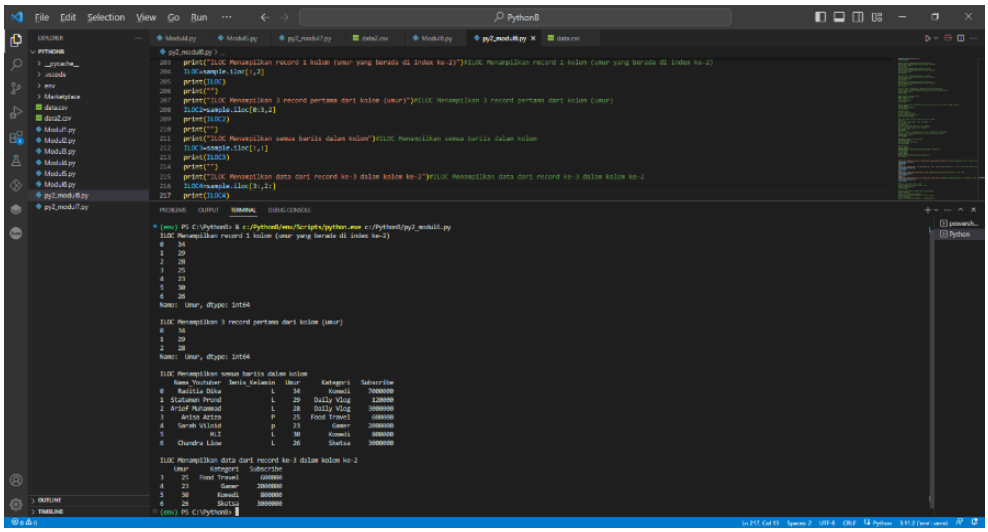
7) Contoh Rename dan Drop Kolom



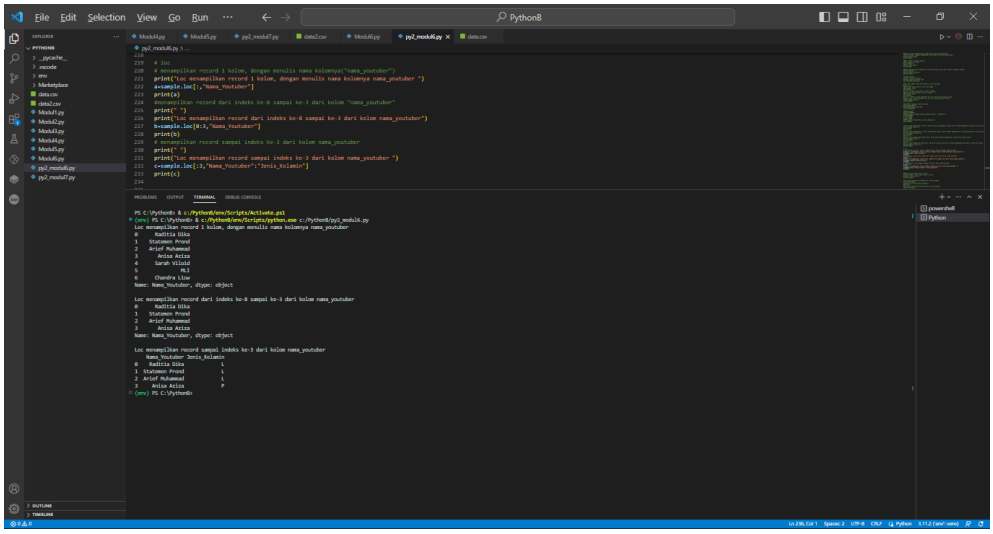
```
#RENAME KOLOM
print("RENAME")
RENAME=sample.rename(columns={'Nama_Youtuber':'Youtuber'})
print(RENAME)
# DROP KOLOM
print("DROP")
DROP=sample.drop(columns=['Jenis_Kelamin'])
print(DROP)

#ILOC Menampilkan record 1 kolom (umur yang berada di index ke-2)
#print("ILOC")
#ILOC=sample.iloc[:,2]
#print(ILOC)

#ILOC Menampilkan 3 record pertama dari kolom (umur)
#print("ILOC")
#ILOC2=sample.iloc[0:3,2]
#print(ILOC2)

#ILOC Menampilkan semua bariis dalam kolom
#print("ILOC")
#ILOC3=sample.iloc[:,:]
#print(ILOC3)

#ILOC Menampilkan data dari record ke-3 dalam kolom ke-2
#print("ILOC")
#ILOC4=sample.iloc[3:,2:]
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul6.py
RENAME
          Youtuber Jenis_Kelamin  Umur     Kategori  Subscribe
0    Radzila Olka             L    34       Komedi    7000000
1  Statemen Prond             L    29   Daily Vlog     120000
3     Anisa Aziza             P    25  Food Travel     680000
4   Sarah Vilold             p    23        Gamer    2000000
5             MLI             L    30       Komedi     800000
6   Chandra Liow              L    26       Sketsa    3000000
DROP
```

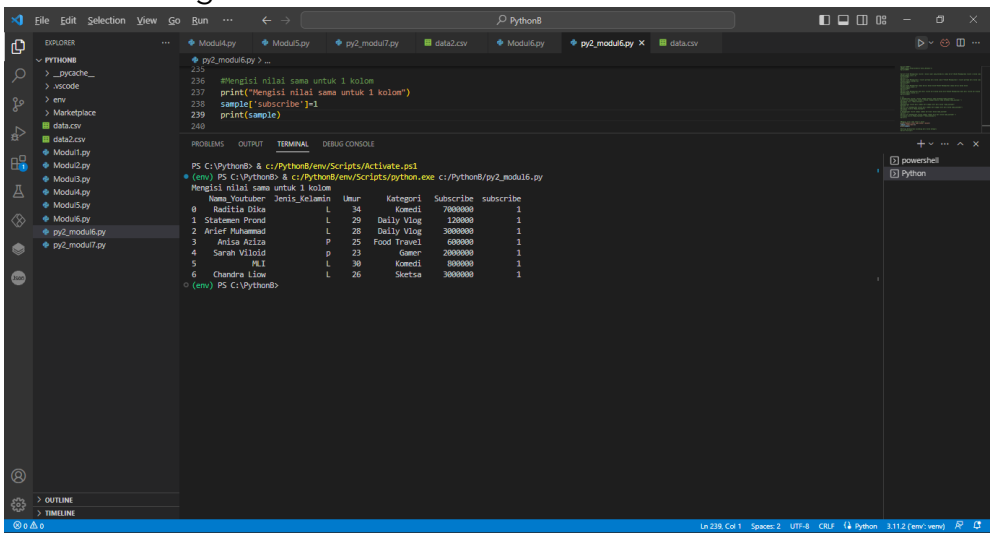
8) Contoh iloc



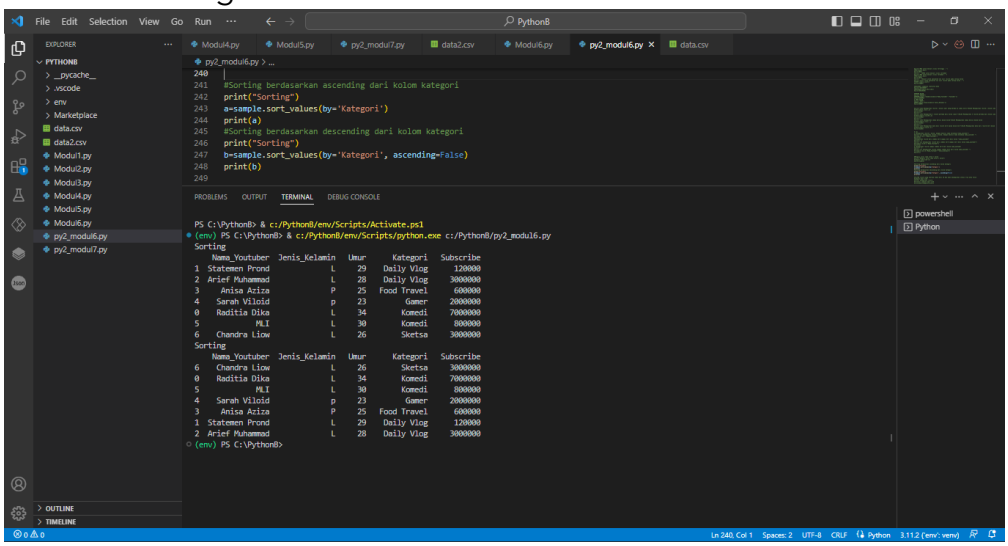
9) Contoh loc



10) Contoh Mengisi Nilai Sama Untuk 1 Kolom



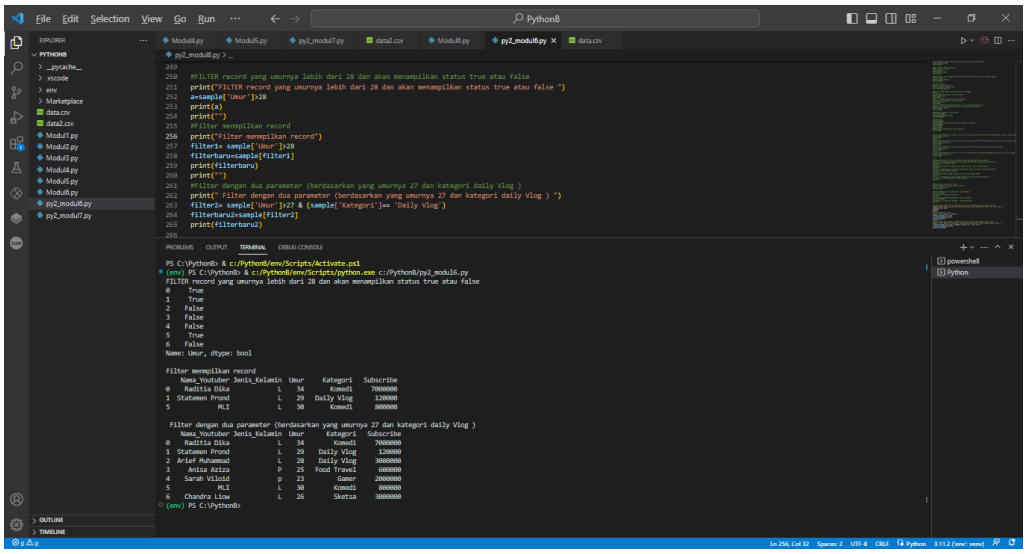
11) Contoh Sorting



```
#Sorting berdasarkan ascending dari kolom kategori
print("Sorting")
a=sample.sort_values(by='Kategori')
print(a)
#Sorting berdasarkan descending dari kolom kategori
print("Sorting")
b=sample.sort_values(by='Kategori', ascending=False)
print(b)
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul6.py
Sorting
    Nama_Youtuber  Jenis_Kelamin  Umur      Kategori  Subscribe
1  Statemen Prond              L    29    Daily Vlog     120000
2  Arief Muhammad              L    28    Daily Vlog    3000000
3     Anisa Aziza              P    25  Food Travel     600000
4   Sarah Viloid               p    23         Gamer   2000000
0   Raditia Dika               L    34        Komedi   7000000
5             MLI              L    30        Komedi    800000
6   Chandra Liow               L    26        Sketsa   3000000
Sorting
    Nama_Youtuber  Jenis_Kelamin  Umur      Kategori  Subscribe
6   Chandra Liow               L    26        Sketsa   3000000
0   Raditia Dika               L    34        Komedi   7000000
5             MLI              L    30        Komedi    800000
4   Sarah Viloid               p    23         Gamer   2000000
3     Anisa Aziza              P    25  Food Travel     600000
1  Statemen Prond              L    29    Daily Vlog     120000
2  Arief Muhammad              L    28    Daily Vlog    3000000
(env) PS C:\Python8>
```

12) Contoh Filter



```
#FILTER record yang umurnya lebih dari 28 dan akan menampilkan status true atau false
print("FILTER record yang umurnya lebih dari 28 dan akan menampilkan status true atau false ")
a=sample['Umur']>28
print(a)
print("")
#filter menmpilkan record
print("Filter menmpilkan record")
filter1= sample['Umur']>28
filterbaru=sample[filter1]
print(filterbaru)
print("")
#filter dengan dua parameter (berdasarkan yang umurnya 27 dan kategori daily Vlog )
print(" Filter dengan dua parameter (berdasarkan yang umurnya 27 dan kategori daily Vlog ) ")
filter2= sample['Umur']>27 & (sample['Kategori']== 'Daily Vlog')
filterbaru2=sample[filter2]
print(filterbaru2)
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul6.py
FILTER record yang umurnya lebih dari 28 dan akan menampilkan status true atau false
0    True
1    True
2   False
3   False
4   False
5    True
6   False
Name: Umur, dtype: bool

Filter menmpilkan record
    Nama_Youtuber Jenis_Kelamin  Umur      Kategori  Subscribe
0   Raditia Dika              L    34        Komedi   7000000
1  Statemen Prond             L    29    Daily Vlog     120000
5             MLI             L    30        Komedi    800000

 Filter dengan dua parameter (berdasarkan yang umurnya 27 dan kategori daily Vlog )
    Nama_Youtuber Jenis_Kelamin  Umur      Kategori  Subscribe
0   Raditia Dika              L    34        Komedi   7000000
1  Statemen Prond             L    29    Daily Vlog     120000
2  Arief Muhammad             L    28    Daily Vlog    3000000
3     Anisa Aziza             P    25  Food Travel     600000
4   Sarah Viloid              p    23         Gamer   2000000
5             MLI             L    30        Komedi    800000
6   Chandra Liow              L    26        Sketsa   3000000
(env) PS C:\Python8>
```

MODUL 7

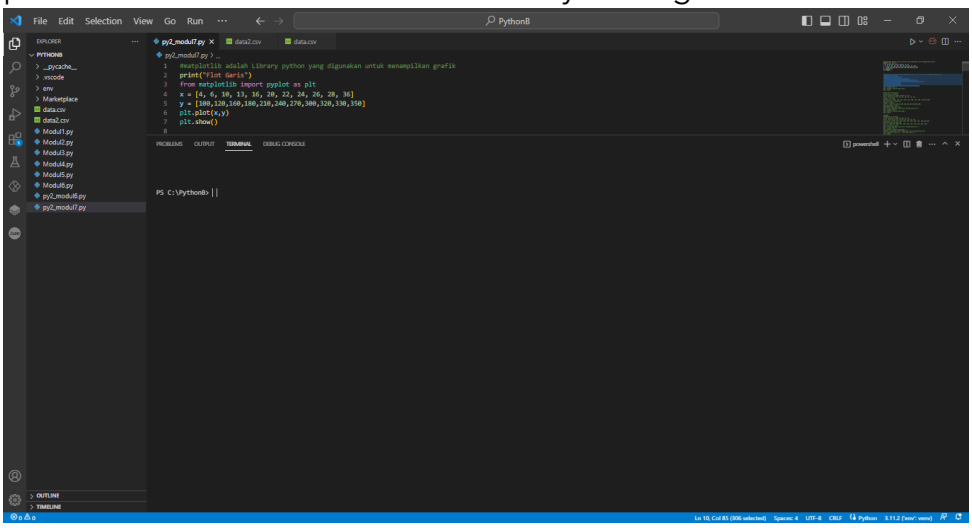
PYTHON (FOR DATA SCIENCE)

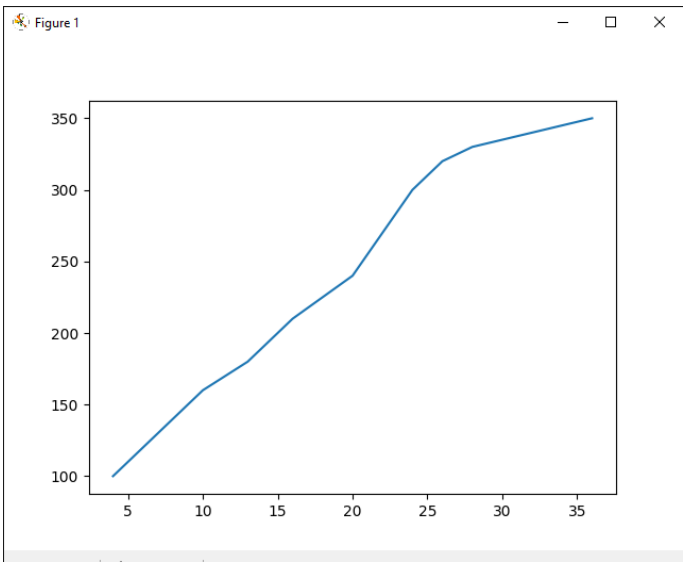
A. Matplotlib

Merupakan library Python 2D yang dapat menghasilkan plot dengan kualitas tinggi dalam berbagai format yang dapat digunakan di banyak platform. Matplotlib dapat digunakan sebagai pembuat grafik dalam berbagai platform, seperti Python dan Jupyter. Grafik yang dapat dibuat beragam, seperti grafik garis, batang, lingkaran, histogram. Terdapat berbagai macam jenis plot/grafik yang dapat dibuat oleh Matplotlib sebagai berikut :

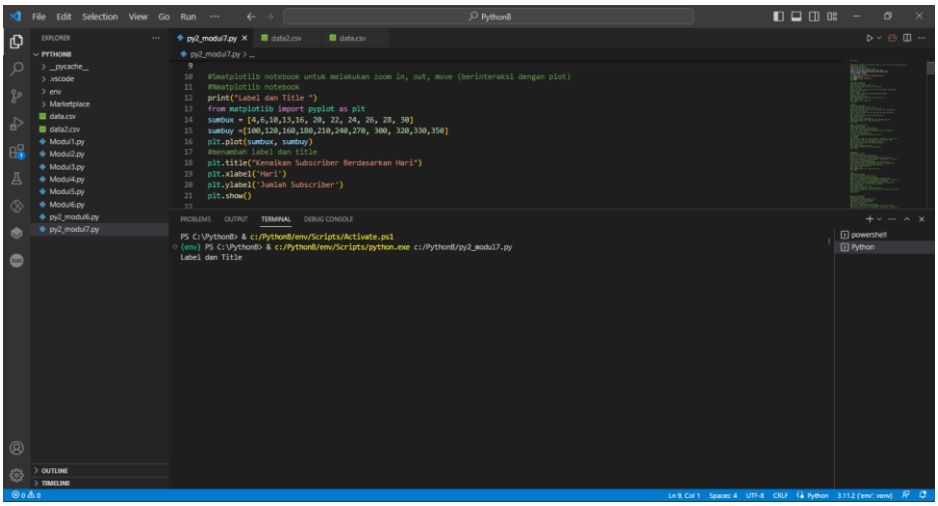
1. Jenis Plot Garis

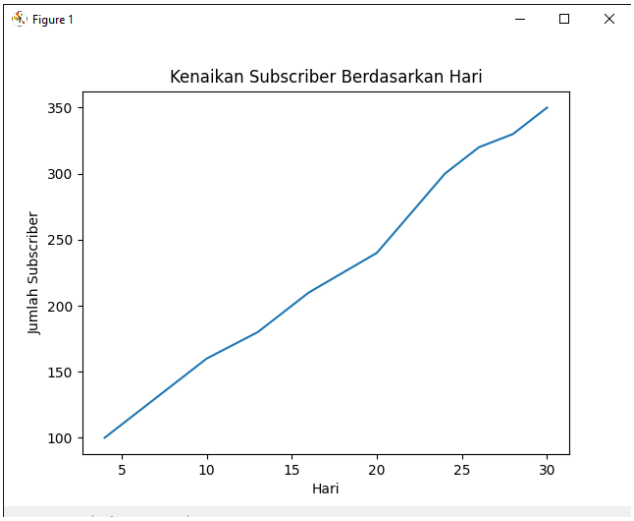
Merupakan representasi berupa garis yang menghubungkan antar posisi koordinat data. Cara membuatnya sebagai berikut.



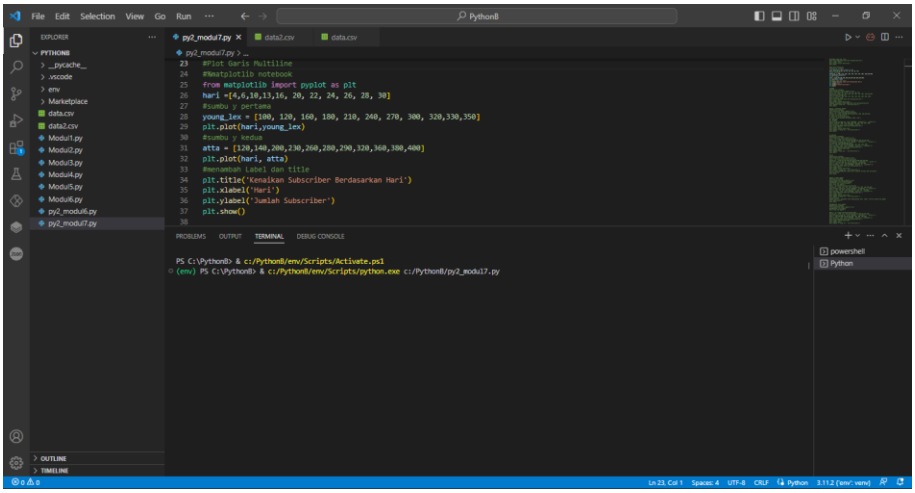


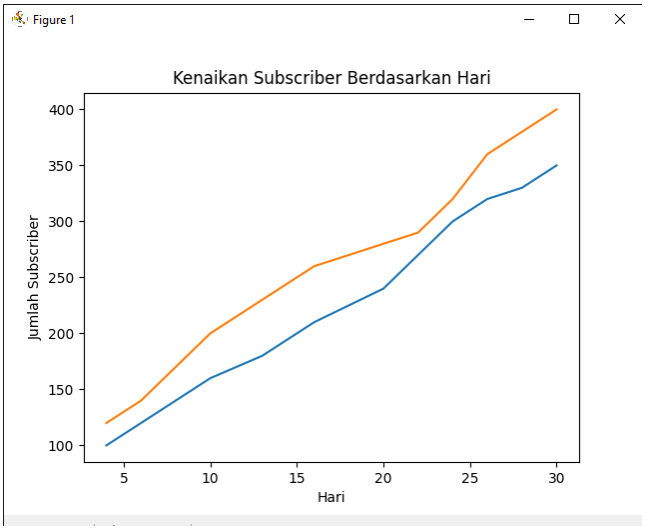
2. Contoh Menambah Label dan Title



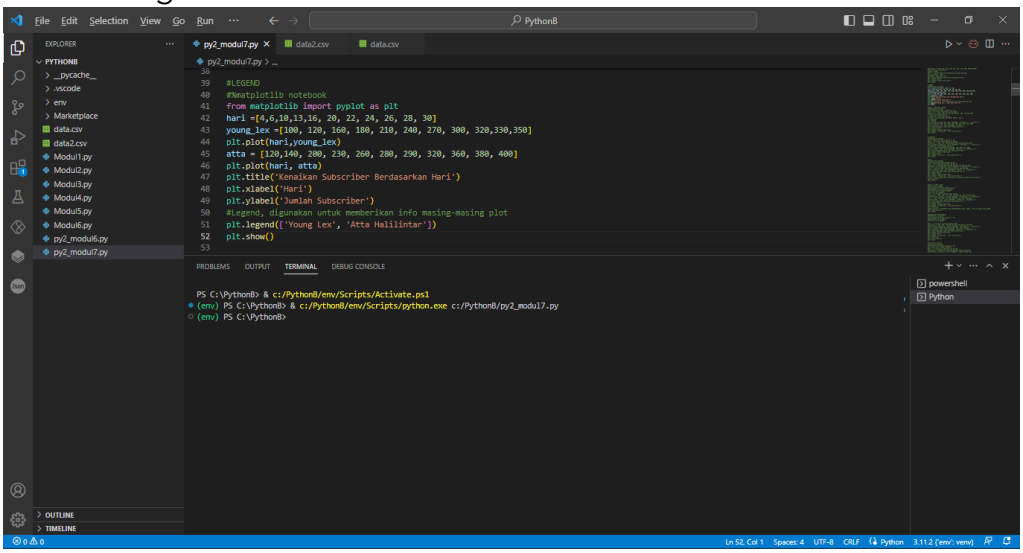


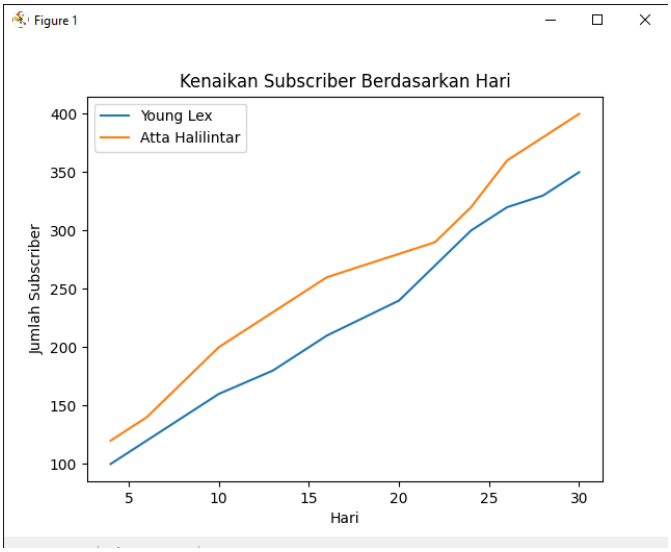
3. Contoh Plot Garis Multiline



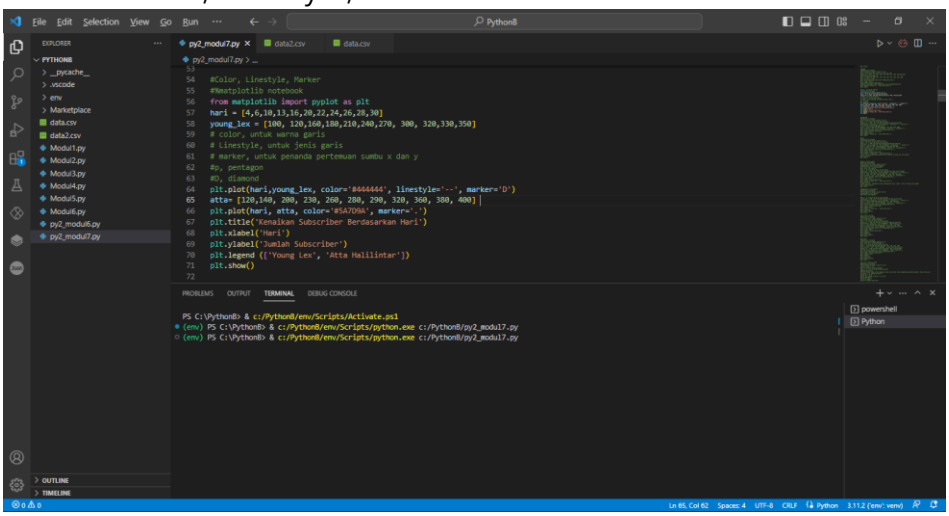


4. Contoh Legend



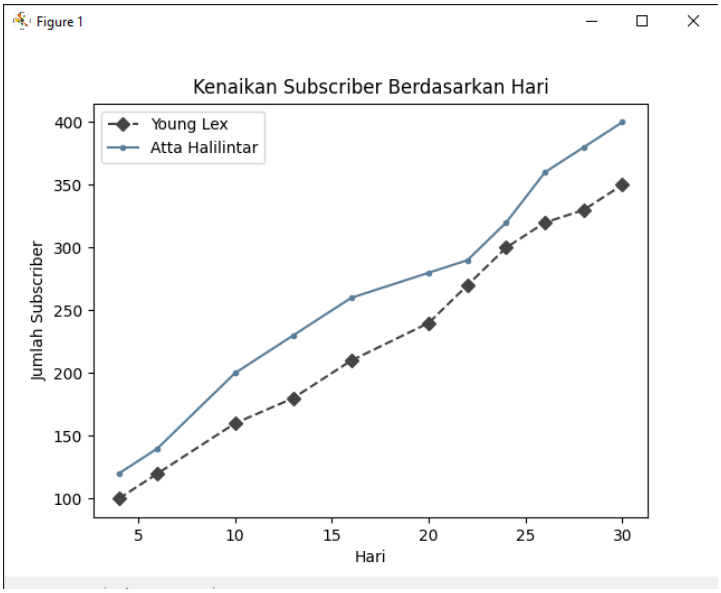


5. Contoh Color, Linestyle, Marker

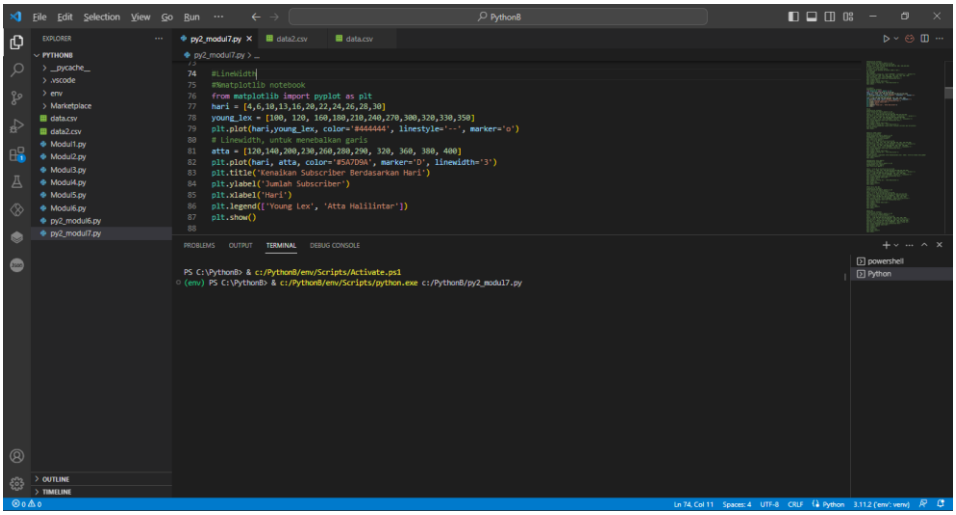


```
#Color, Linestyle, Marker
#%matplotlib notebook
from matplotlib import pyplot as plt
hari = [4,6,10,13,16,20,22,24,26,28,30]
young_lex = [100, 120,160,180,210,240,270, 300, 320,330,350]
# color, untuk warna garis
# Linestyle, untuk jenis garis
# marker, untuk penanda pertemuan sumbu x dan y
#p, pentagon
#D, diamond
plt.plot(hari,young_lex, color='#444444', linestyle='--', marker='D')
atta= [120,140, 200, 230, 260, 280, 290, 320, 360, 380, 400]
plt.plot(hari, atta, color='#5A7D9A', marker='.')
plt.title('Kenaikan Subscriber Berdasarkan Hari')
plt.xlabel('Hari')
plt.ylabel('Jumlah Subscriber')
plt.legend (['Young Lex', 'Atta Halilintar'])
plt.show()
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul7.py
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul7.py
```

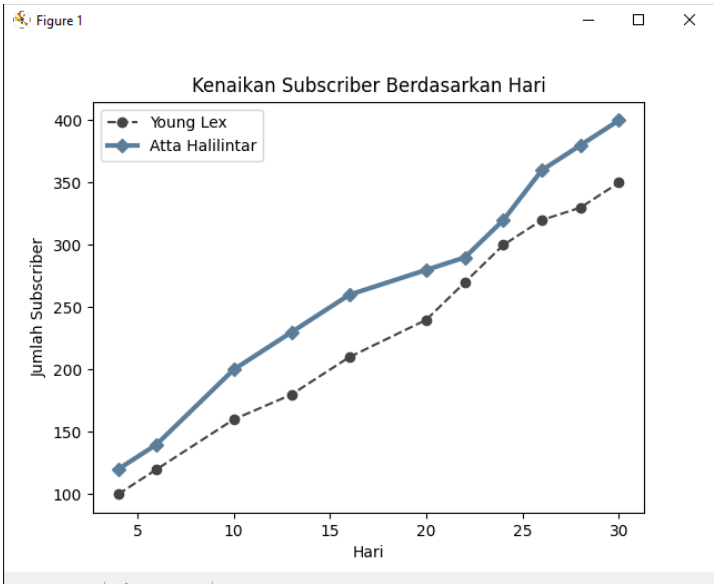


6. Contoh Linewidth

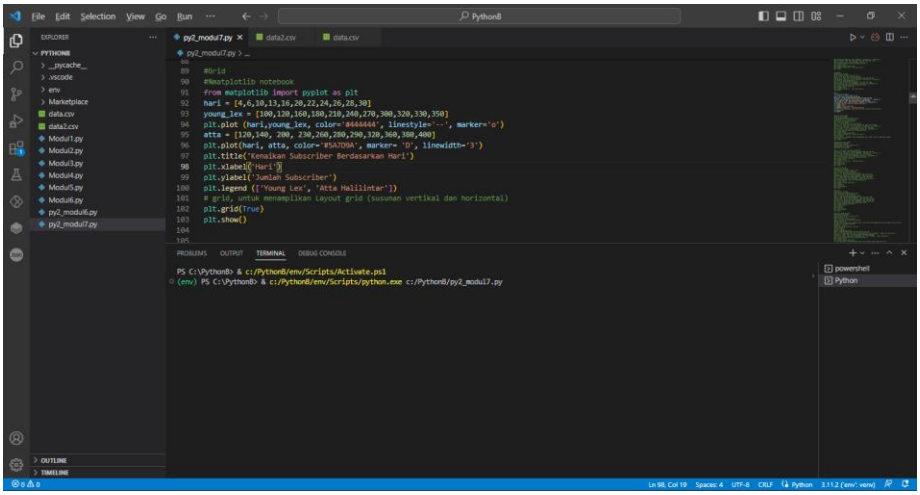


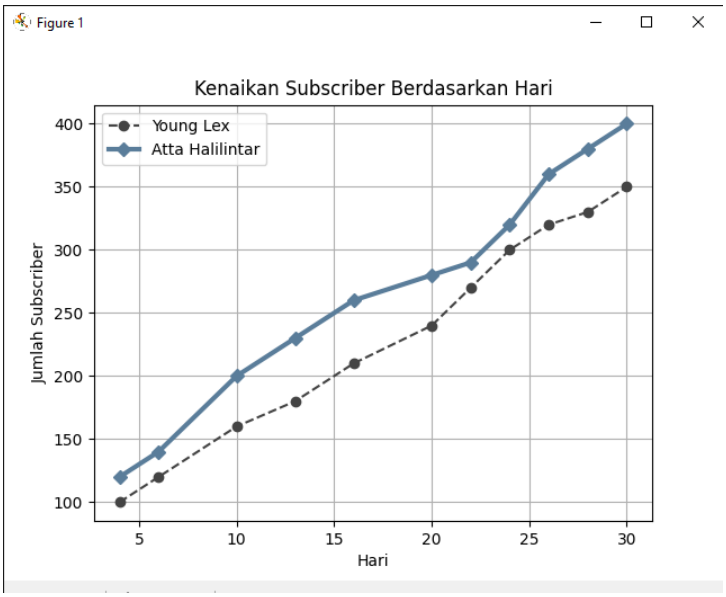
```
#Linewidth
#%matplotlib notebook
from matplotlib import pyplot as plt
hari = [4,6,10,13,16,20,22,24,26,28,30]
young_lex = [100, 120, 160,180,210,240,270,300,320,330,350]
plt.plot(hari,young_lex, color='#444444', linestyle='--', marker='o')
# Linewidth, untuk menebalkan garis
atta = [120,140,200,230,260,280,290, 320, 360, 380, 400]
plt.plot(hari, atta, color='#5A7D9A', marker='D', linewidth='3')
plt.title('Kenaikan Subscriber Berdasarkan Hari')
plt.ylabel('Jumlah Subscriber')
plt.xlabel('Hari')
plt.legend(['Young Lex', 'Atta Halilintar'])
plt.show()
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul7.py
```

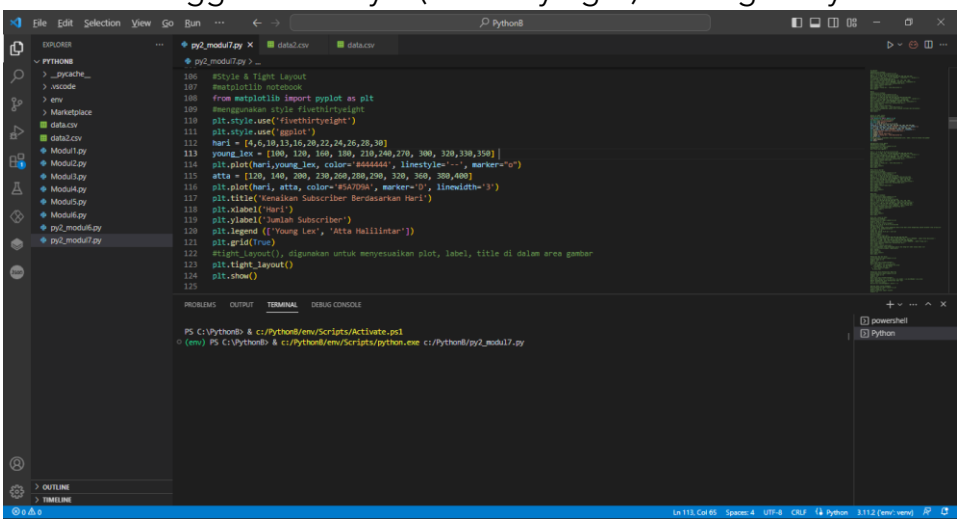


7. Contoh Grid



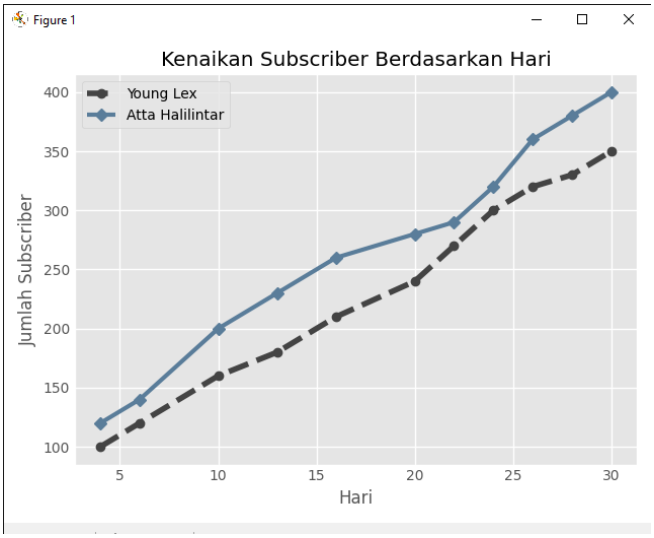


8. Contoh Menggunakan Style (fivethirtyeight) dan Tight Layout

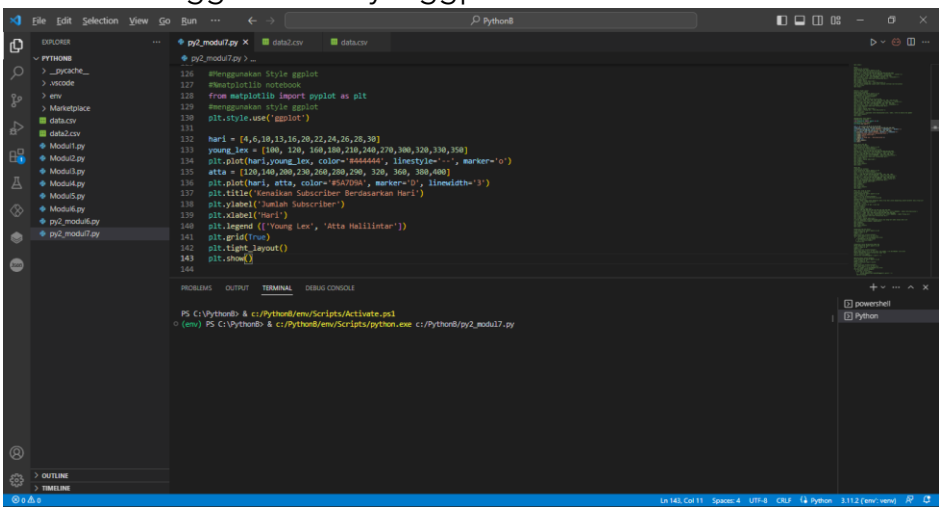


```
#Style & Tight Layout
#matplotlib notebook
from matplotlib import pyplot as plt
#menggunakan style fivethirtyeight
plt.style.use('fivethirtyeight')
plt.style.use('ggplot')
hari = [4,6,10,13,16,20,22,24,26,28,30]
young_lex = [100, 120, 160, 180, 210,240,270, 300, 320,330,350]
plt.plot(hari,young_lex, color='#444444', linestyle='--', marker="o")
atta = [120, 140, 200, 230,260,280,290, 320, 360, 380,400]
plt.plot(hari, atta, color='#5A7D9A', marker='D', linewidth='3')
plt.title('Kenaikan Subscriber Berdasarkan Hari')
plt.xlabel('Hari')
plt.ylabel('Jumlah Subscriber')
plt.legend (['Young Lex', 'Atta Halilintar'])
plt.grid(True)
#tight_layout(), digunakan untuk menyesuaikan plot, label, title di dalam area gambar
plt.tight_layout()
plt.show()
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul7.py
```

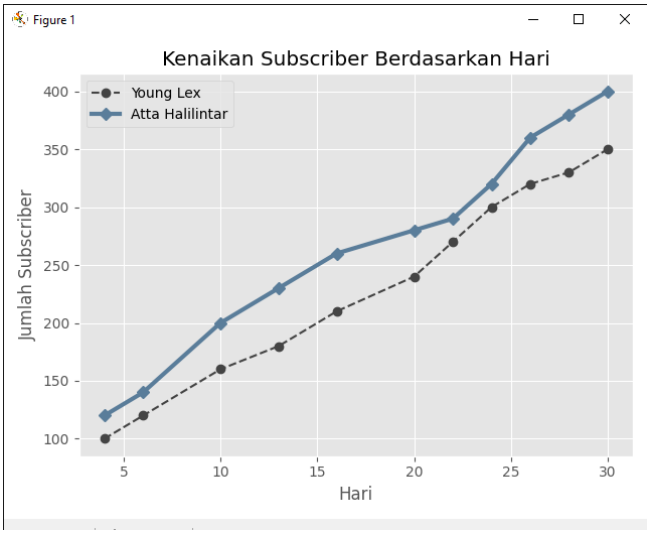


9. Contoh Menggunakan Style ggplot

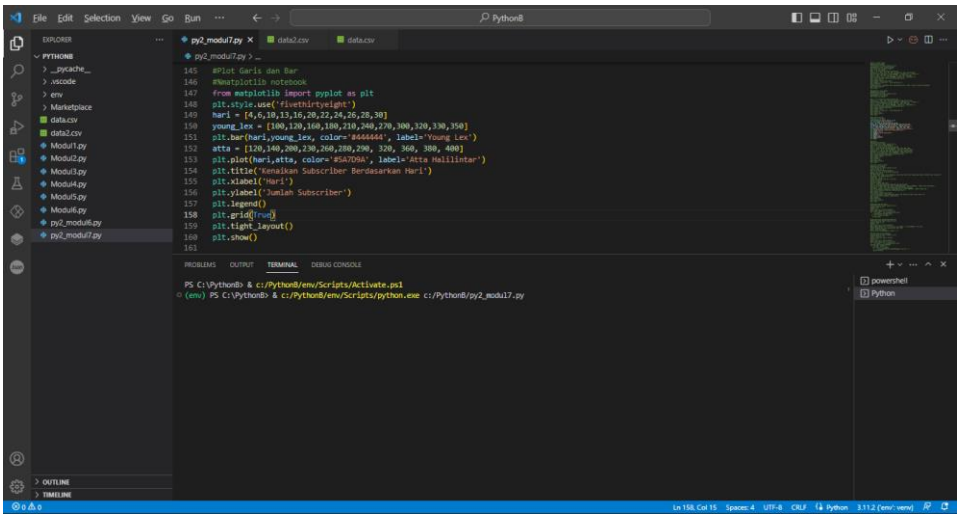


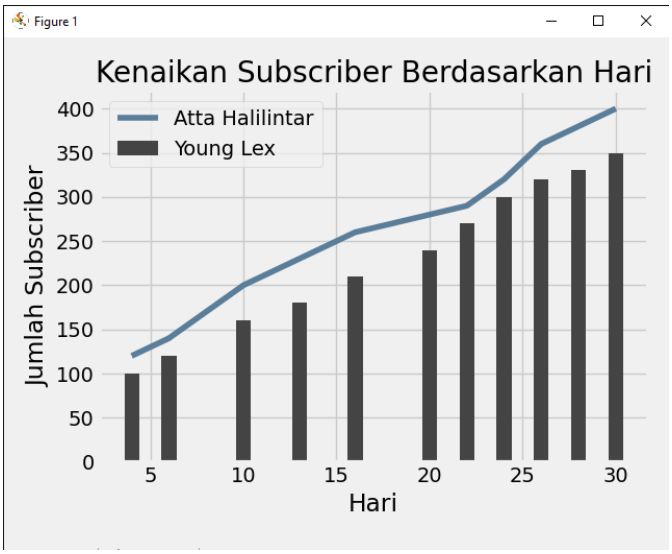
```
#Menggunakan Style ggplot
#Matplotlib notebook
from matplotlib import pyplot as plt
#menggunakan style ggplot
plt.style.use('ggplot')

hari = [4,6,10,13,16,20,22,24,26,28,30]
young_lex = [100, 120, 160,180,210,240,270,300,320,330,350]
plt.plot(hari,young_lex, color='#444444', linestyle='--', marker='o')
atta = [120,140,200,230,260,280,290, 320, 360, 380,400]
plt.plot(hari, atta, color='#5A7D9A', marker='D', linewidth='3')
plt.title('Kenaikan Subscriber Berdasarkan Hari')
plt.ylabel('Jumlah Subscriber')
plt.xlabel('Hari')
plt.legend (['Young Lex', 'Atta Halilintar'])
plt.grid(True)
plt.tight_layout()
plt.show()
```

```
PS C:\Python8> & c:/Python8/env/Scripts/Activate.ps1
(env) PS C:\Python8> & c:/Python8/env/Scripts/python.exe c:/Python8/py2_modul7.py
```

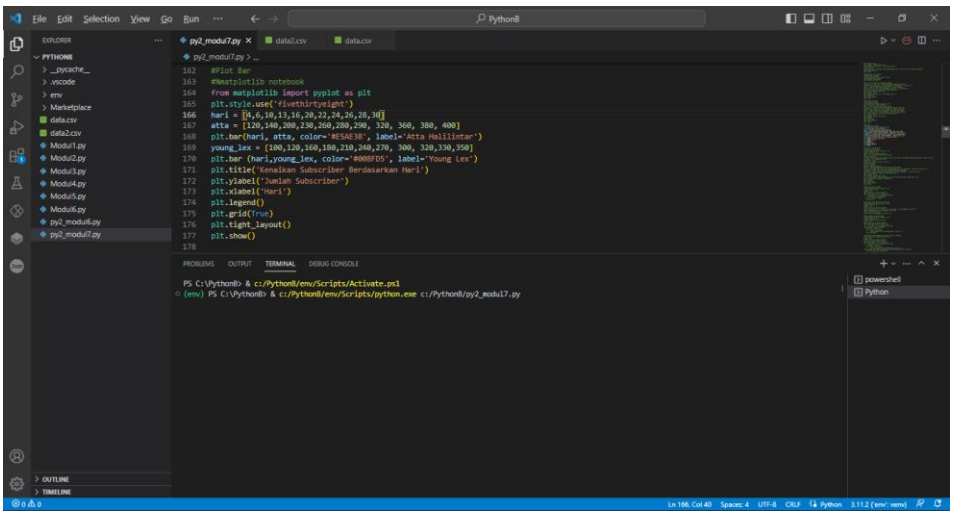


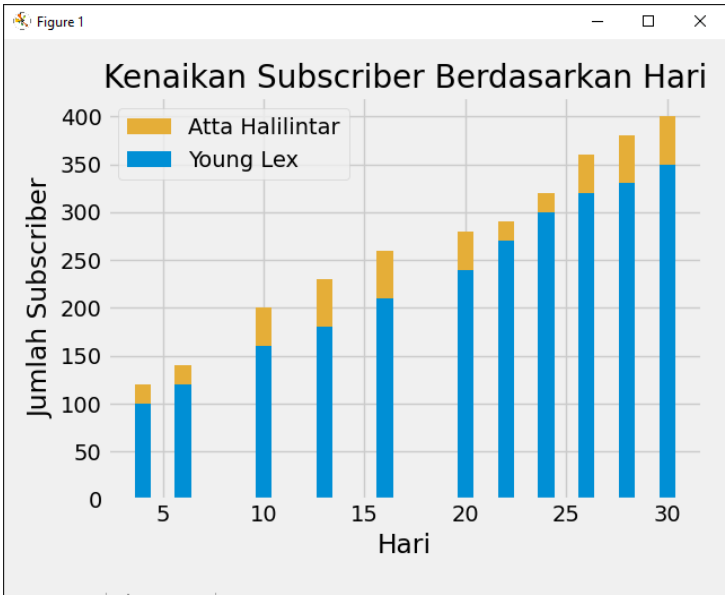
10. Contoh Plot Garis dan Bar



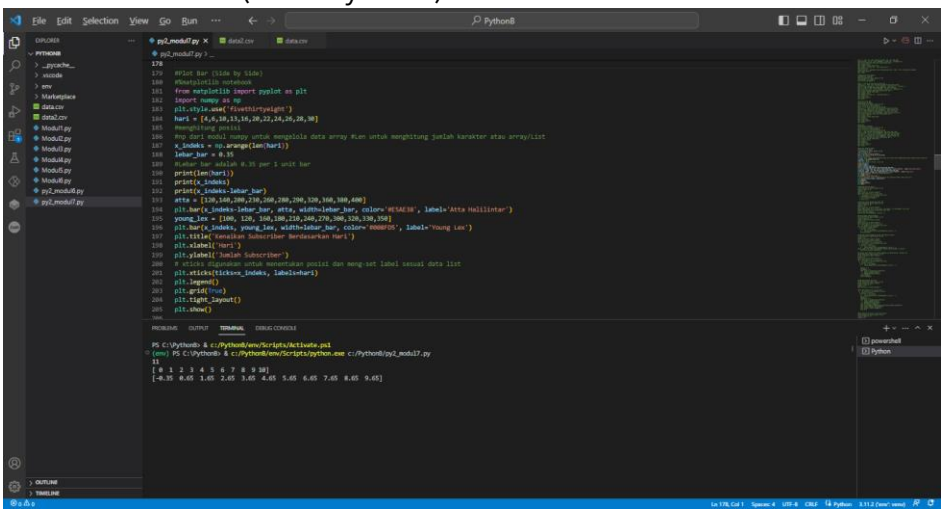


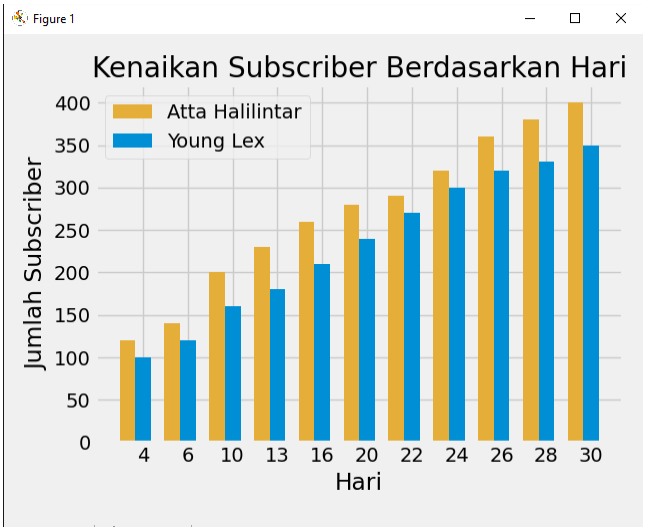
11. Contoh Plot Bar





12. Contoh Plot Bar (Side by Side)

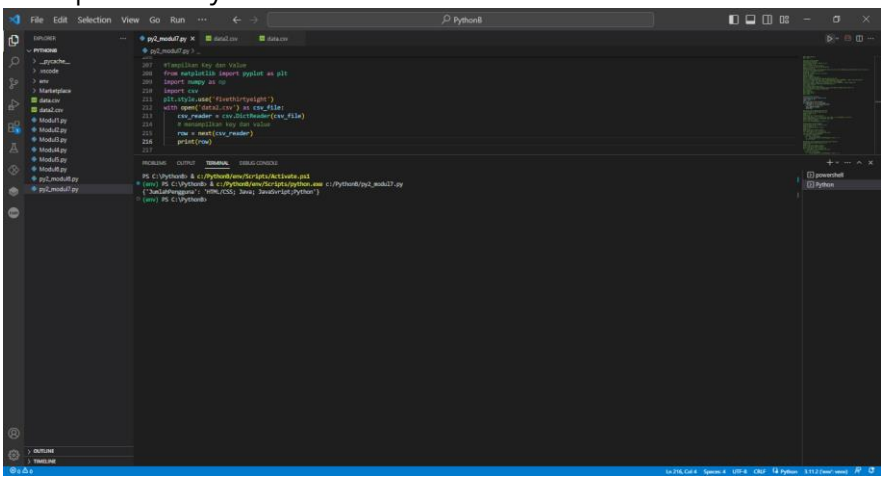




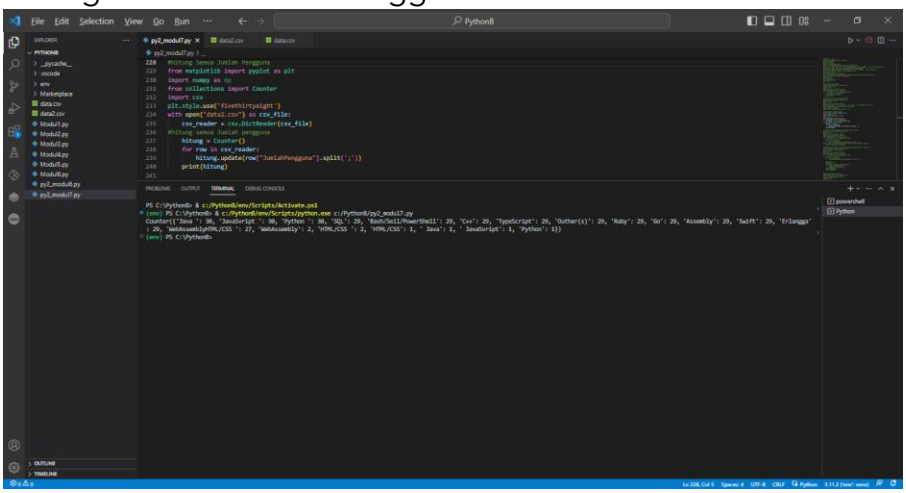
B. Matplotlib - Import Data CSV

Berikut praktikum cara menampilkan plot dengan menggunakan file CSV. Dengan Plot Bar Ikuti langkah-langkah dibawah untuk menampilkan data dari CSV menjadi plot bar.

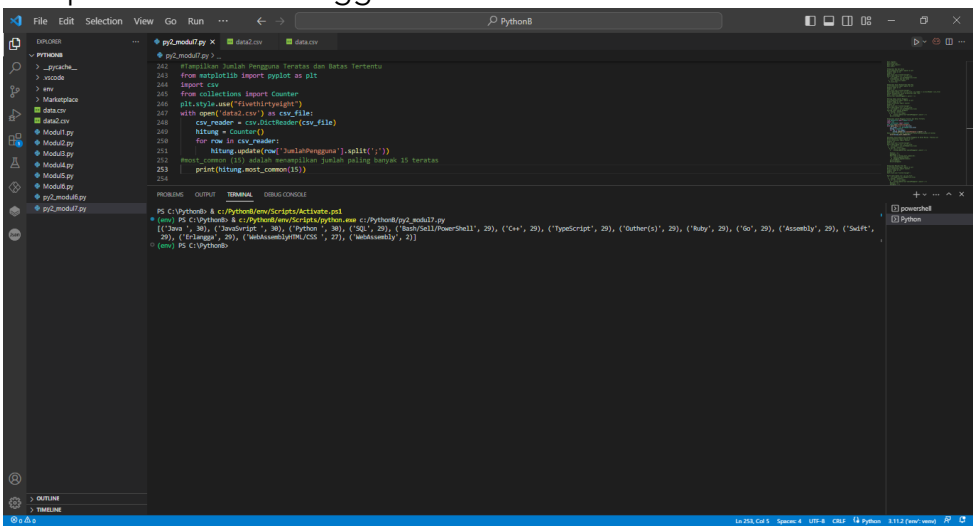
1. Tampilkan Key dan Value



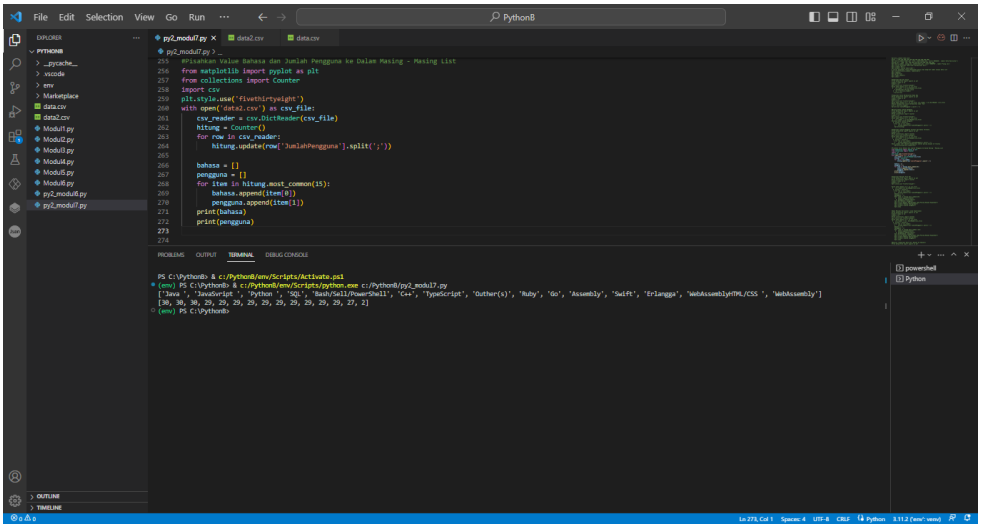
2. Hitung Semua Jumlah Pengguna



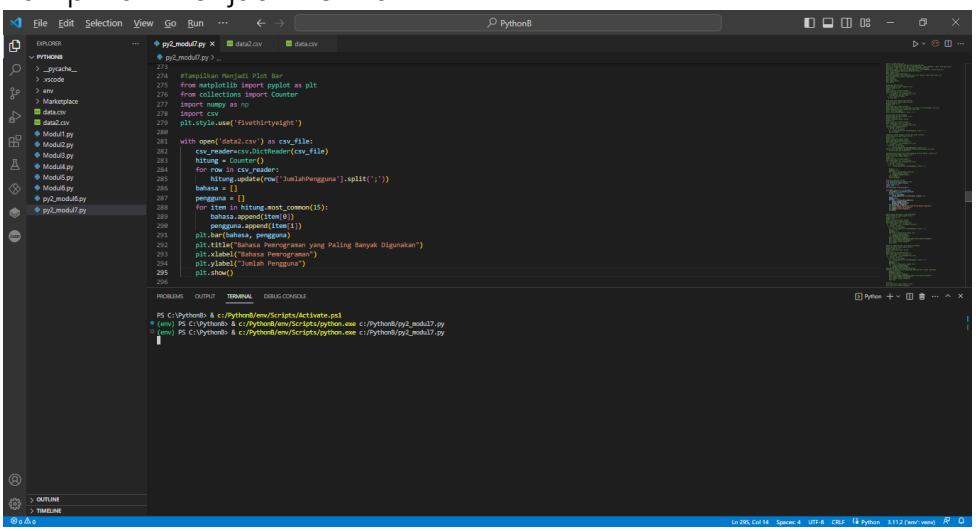
3. Tampilkan Jumlah Pengguna Teratas dan Batas Tertentu

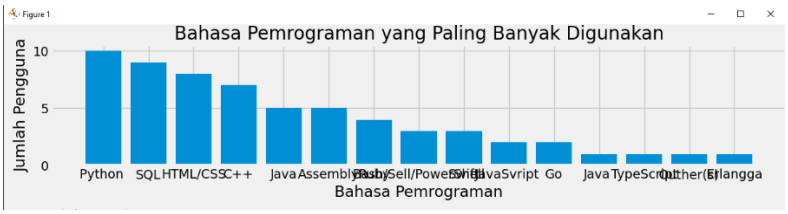


4. Pisahkan Value Bahasa dan Jumlah Pengguna ke Dalam Masing - Masing List

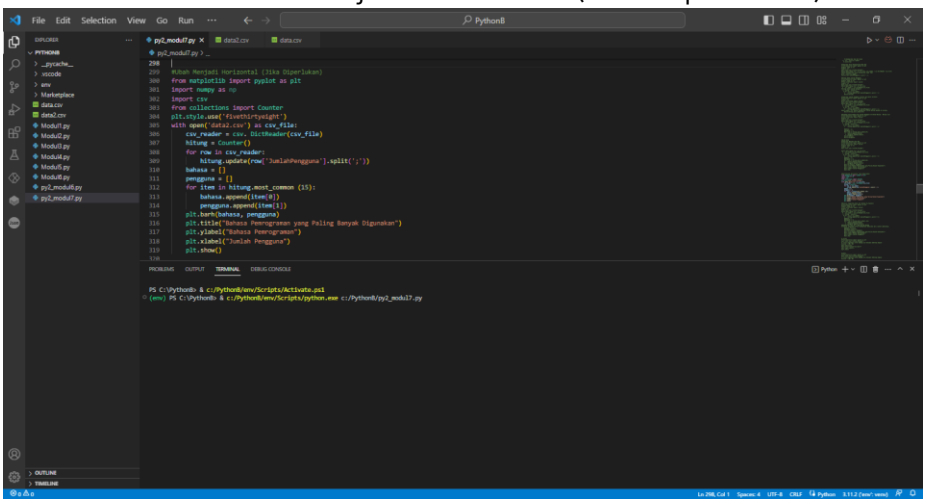


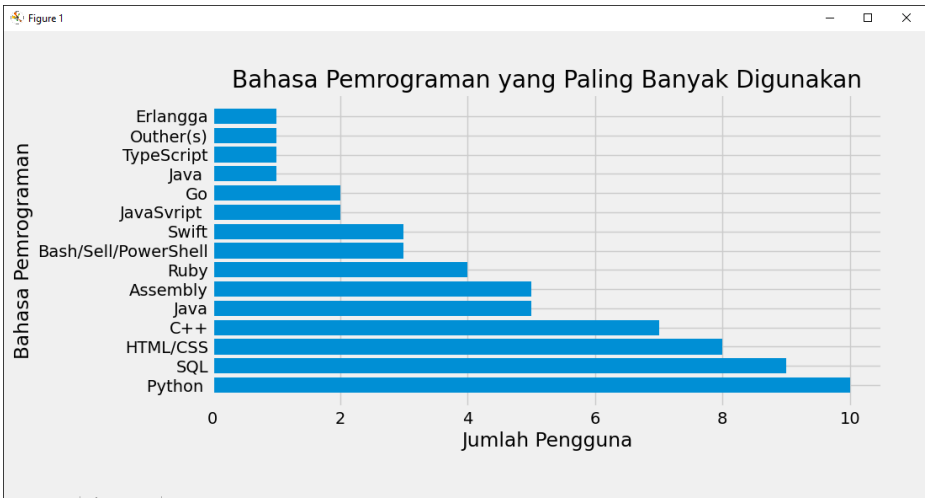
5. Tampilkan Menjadi Plot Bar



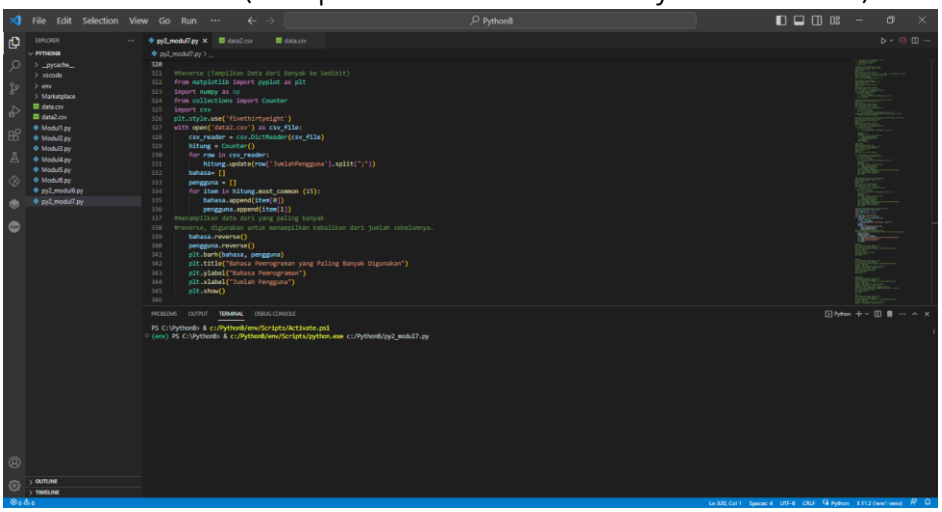


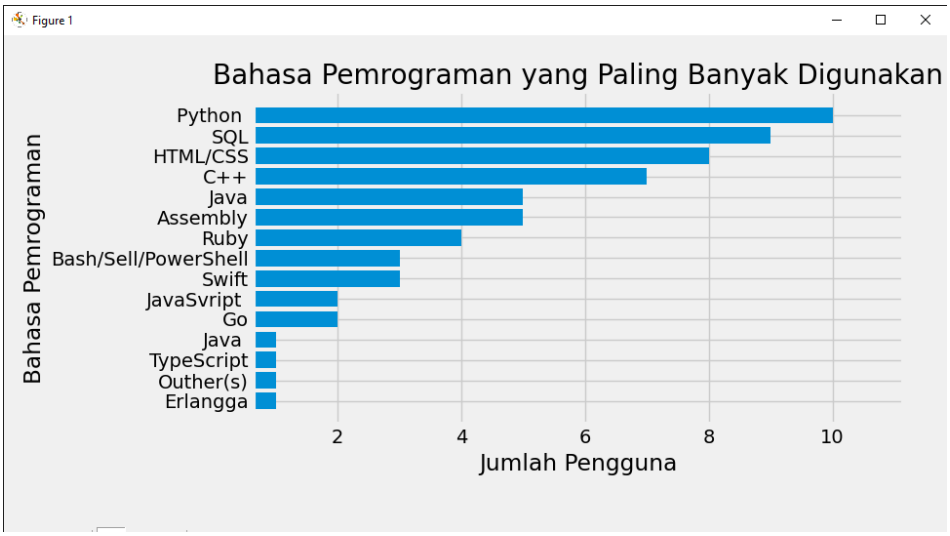
6. Contoh Ubah Data Menjadi Horizontal (Jika Diperlukan)





7. Contoh Reverse (Tampilkan Data Dari terbanyan Ke Sedikit)

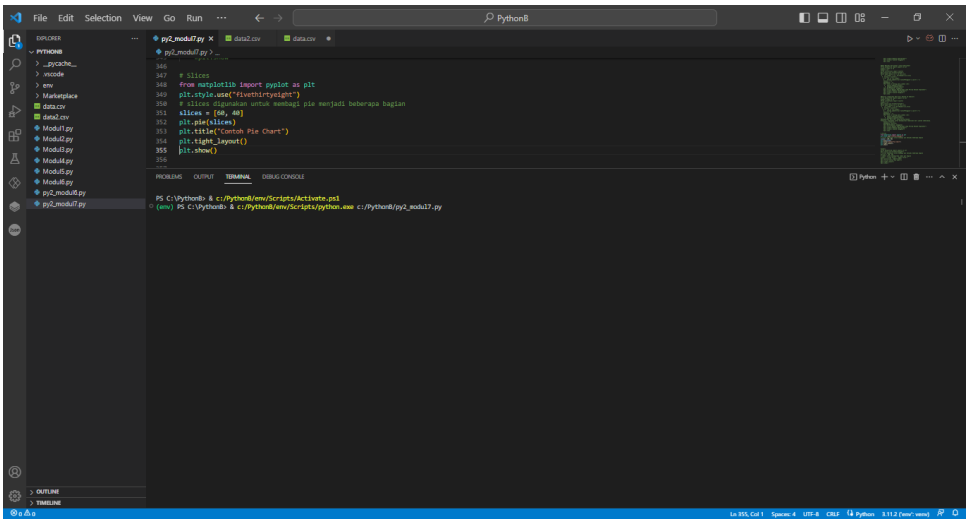


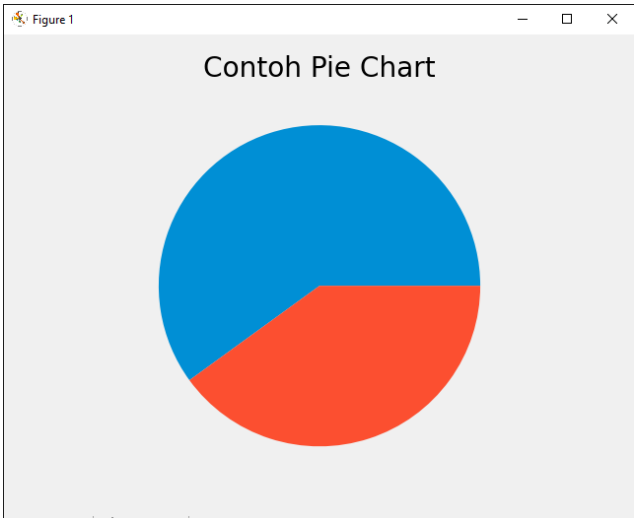


C. Plot Pie Chart

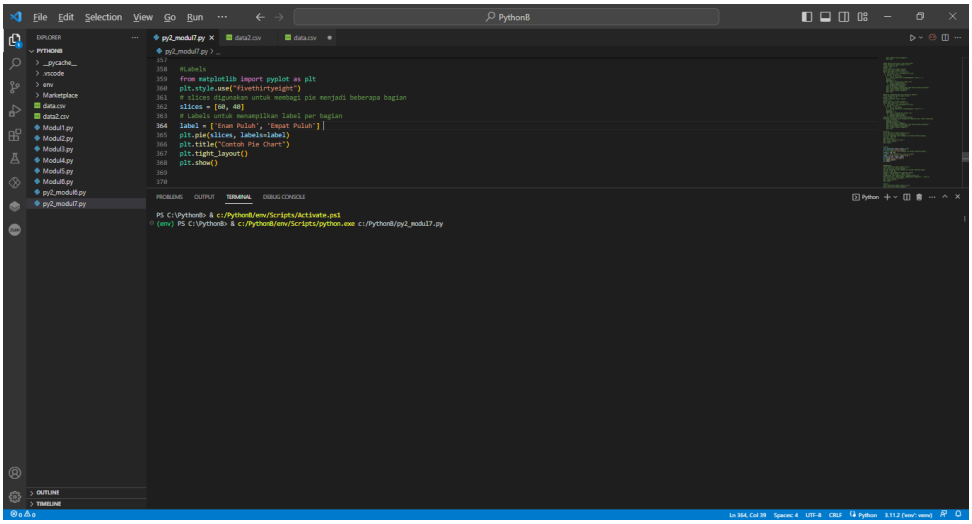
Berikut cara menampilkan data menjadi Pie Chart.

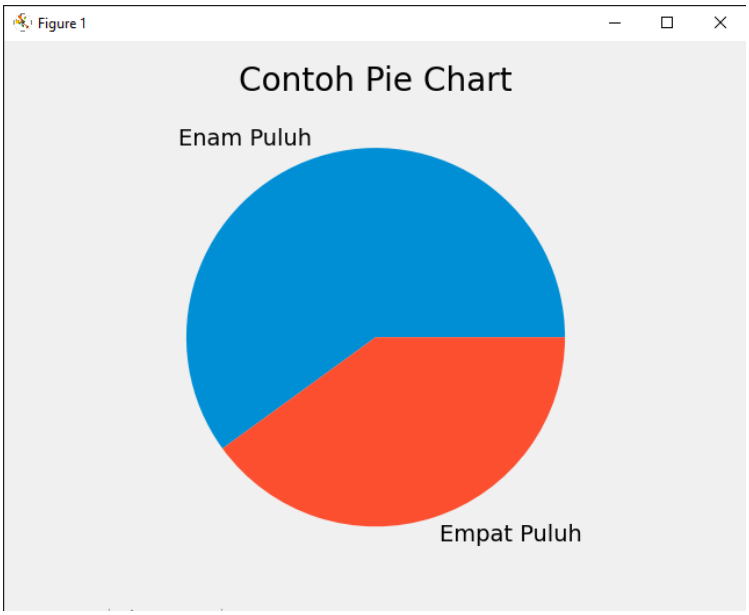
1. Contoh Slices



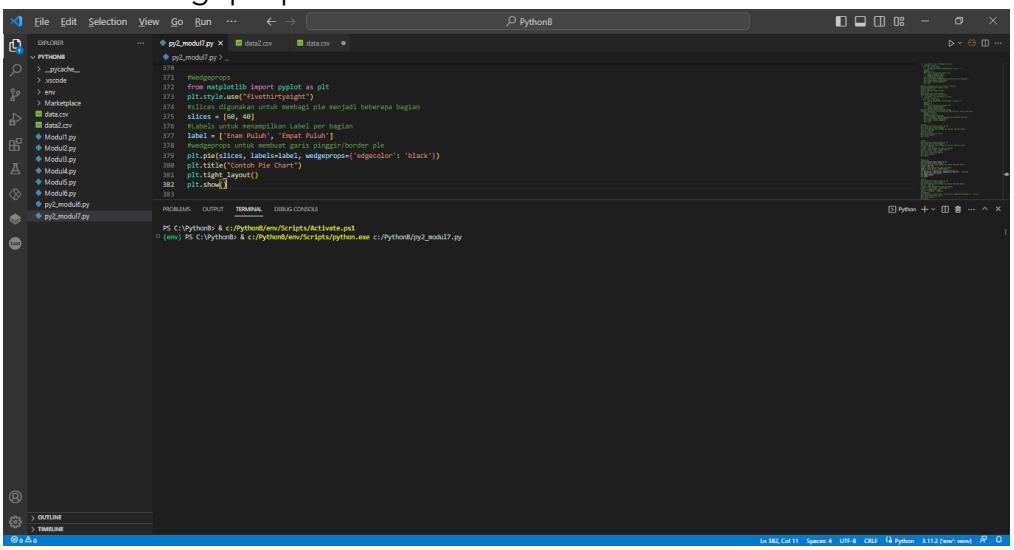


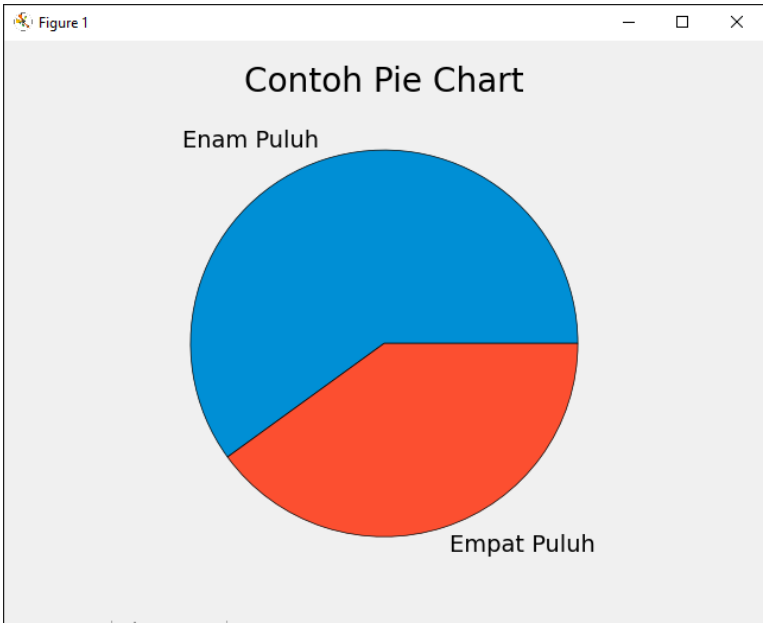
2. Contoh Labels



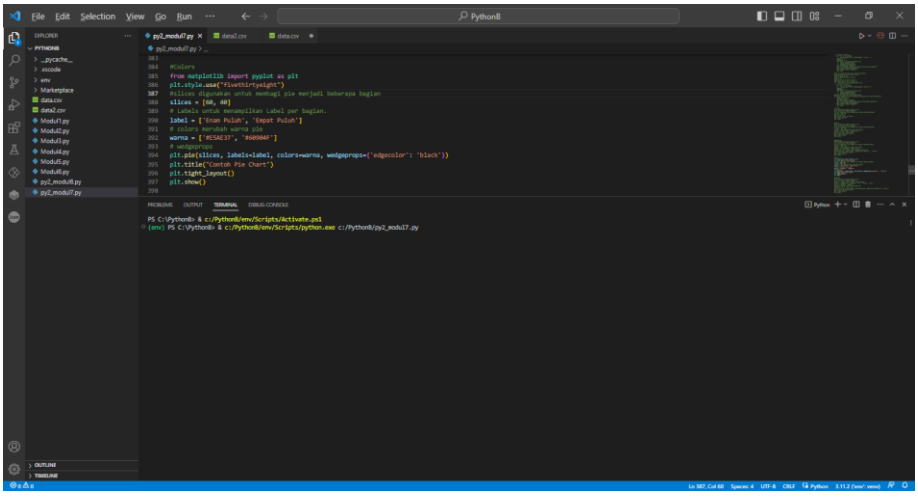


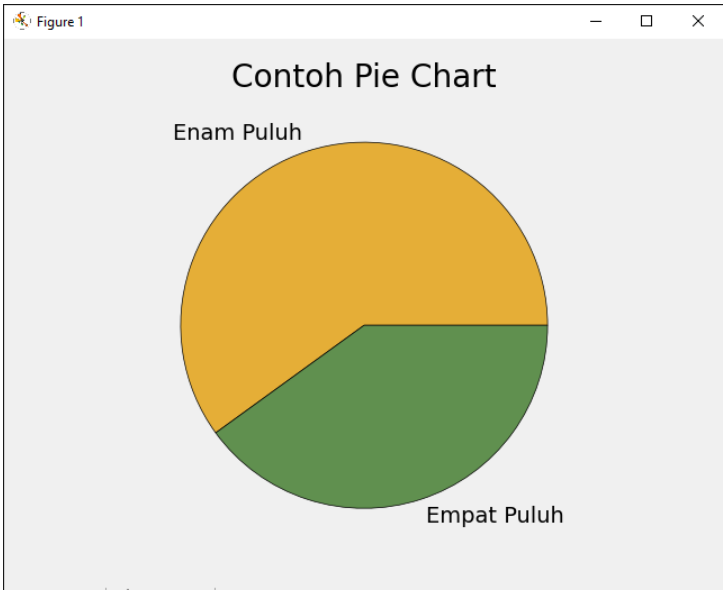
3. Contoh Wedgeprops



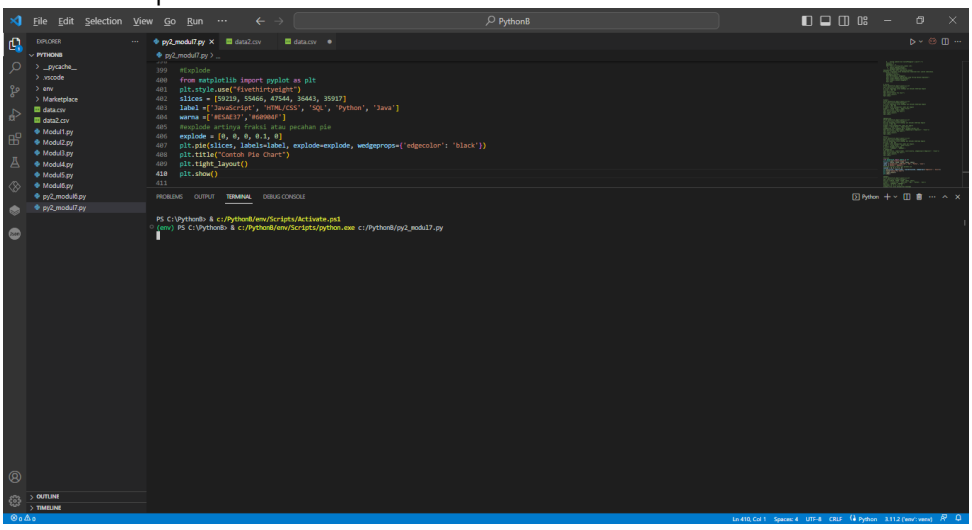


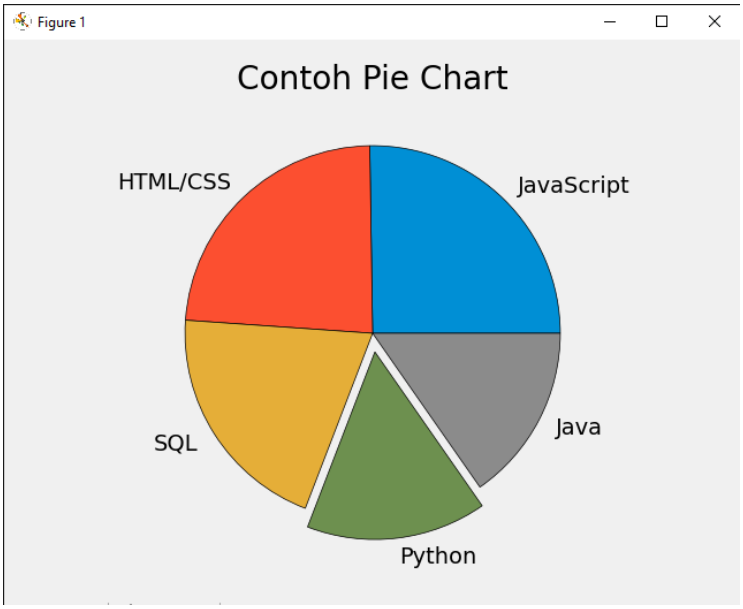
4. Contoh Colors



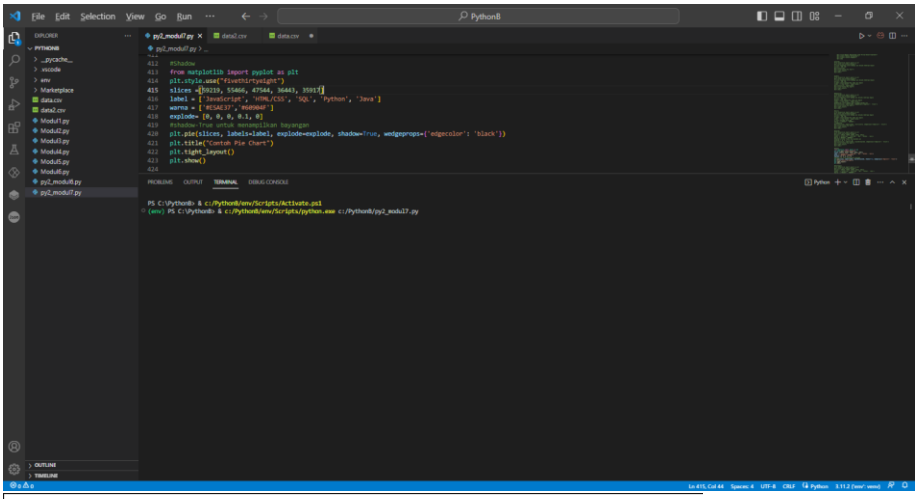


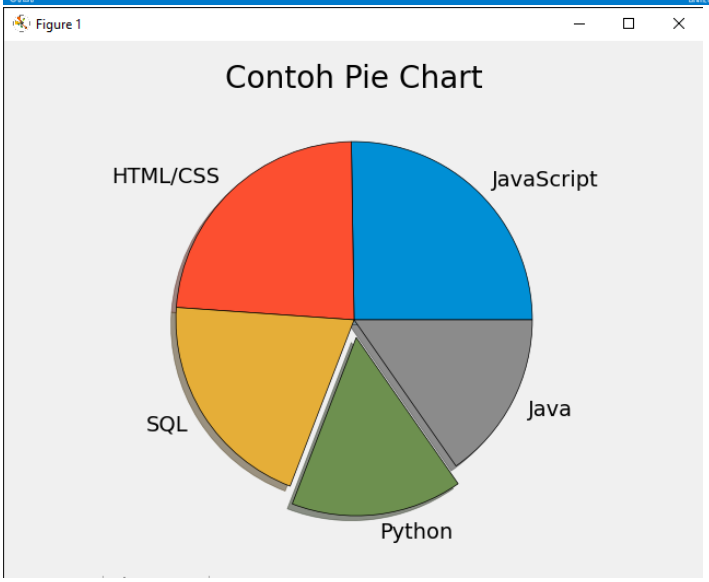
5. Contoh Explode



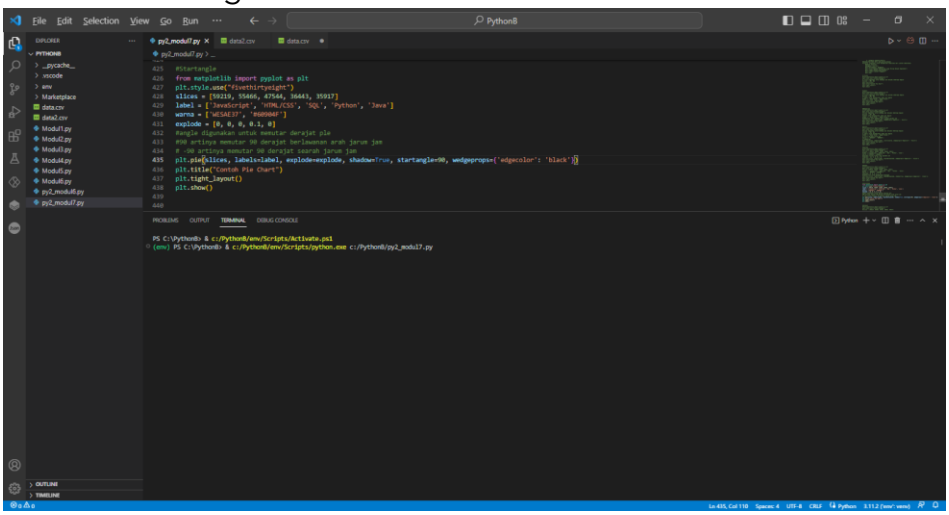


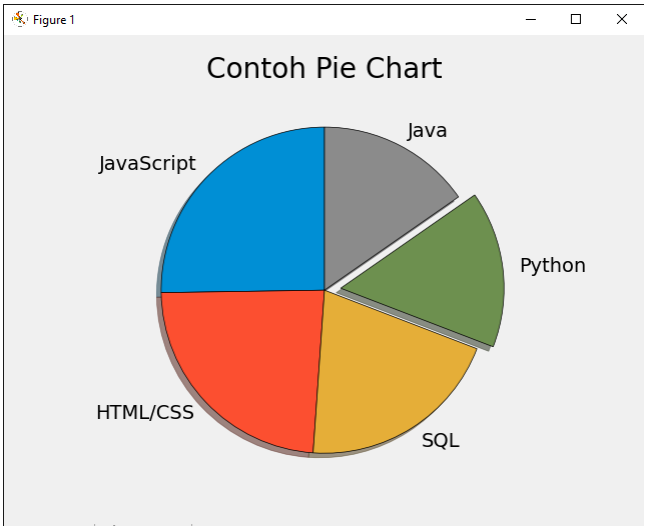
6. Contoh Shadow





7. Contoh Startangle





8. Contoh Autopct