

MODUL 6

Matplotlib

6.1. Tujuan Praktikum:

Adapun Tujuan pada pratikum modul 6 yaitu:

1. Mampu memahami konsep dari python Matplotlib Line, Labels, Grid, Subplot, Scatter, Bars, Histograms, Pie Charts.
2. Dapat mempraktikkan penerapan dari python Matplotlib Line, Labels, Grid, Subplot, Scatter, Bars, Histograms, Pie Charts.

6.2. Alat dan Bahan Praktikum

1. Laptop
2. Mouse
3. Software *Visual Studio Code*
4. Python

6.3. Dasar Teori

6.3.1. Matplotlib

Matplotlib merupakan pustaka open source dan gratis untuk visualisasi data yang dapat dijalankan berbagai platform, dibuat dengan fondasi array NumPy, dan terinspirasi dari fungsionalitas MATLAB dalam penggunaannya di Python. Matplotlib memungkinkan kita untuk memvisualisasi data, baik dalam 2 dimensi maupun 3 dimensi, menggunakan bahasa pemrograman Python. Hasil visualisasinya pun berkualitas tinggi dan bisa disimpan dalam berbagai format.

6.3.2. Pyplot

Pyplot adalah submodul dari Matplotlib yang difokuskan pada pembuatan visualisasi data sederhana, dilengkapi dengan berbagai fungsi untuk mengkustomisasi tampilan grafik. Pyplot dapat di import menggunakan `import matplotlib.pyplot`.

6.3.3. Matplotlib Plotting

Fungsi `plot()` dalam Matplotlib digunakan untuk membuat garis dengan menghubungkan titik-titik data yang telah ditentukan sebelumnya. Fungsi ini mengambil parameter yang berisi informasi tentang koordinat titik-titik tersebut, sehingga kita dapat mengontrol bentuk dan tampilan garis yang dihasilkan.

```

import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Pyplot

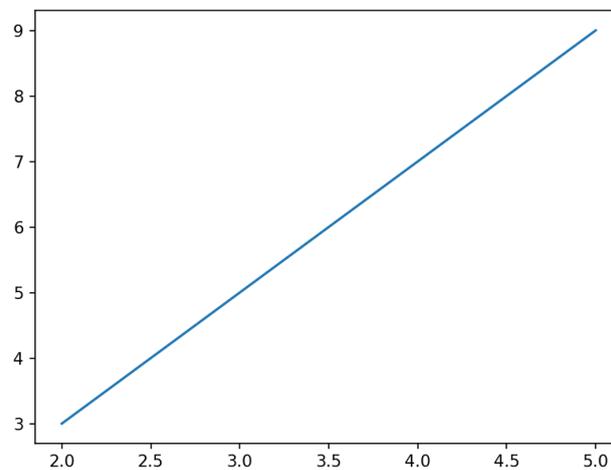
titik_x = np.array([2, 5])
titik_y = np.array([3, 9])

plt.plot(titik_x, titik_y)
plt.show()

```

Parameter 1 adalah array yang berisi titik-titik pada sumbu x.

Parameter 2 adalah array yang berisi titik-titik pada sumbu y.



6.3.4. Matplotlib Markers

Fungsi marker pada Matplotlib digunakan untuk menandai titik-titik data tertentu dengan simbol simbol yang telah ditentukan.

```

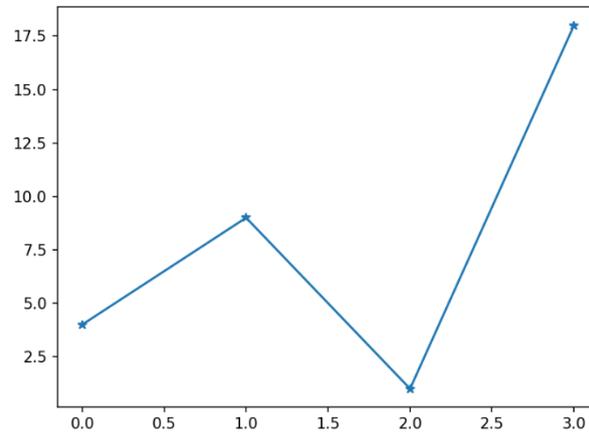
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Marker

titik_y = np.array([4, 9, 1, 18])

plt.plot(titik_y, marker = '*')
plt.show()

```



6.3.5. Python Matplotlib Line

Jenis visualisasi data ini merupakan representasi grafis yang paling umum digunakan, menampilkan data numerik sebagai serangkaian titik yang dihubungkan oleh garis lurus.

```
import matplotlib

import matplotlib.pyplot as plt
import numpy as np

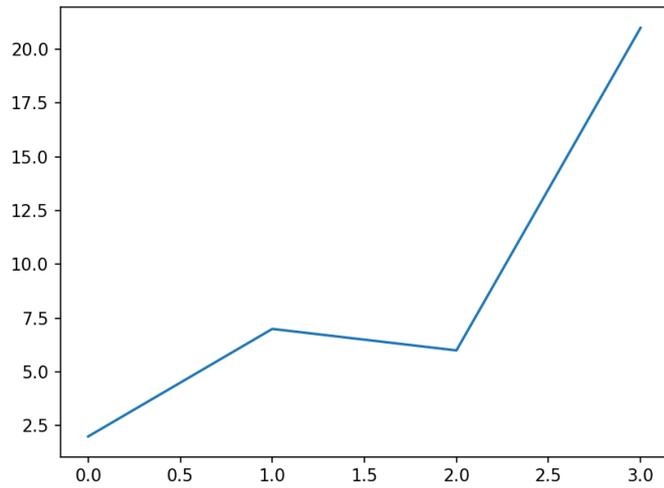
#Matplotlib Marker

titik_y = np.array([2, 7, 6, 21])

plt.plot(titik_y, linestyle = 'dashed')
plt.show()
```

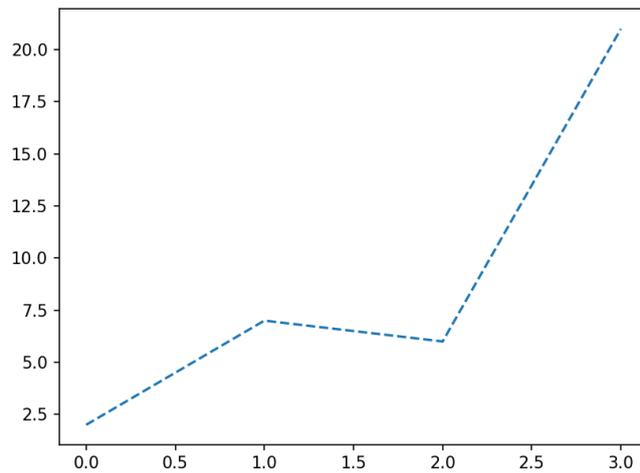
a. Garis Solid:

linestyle='-' atau linestyle='solid': Gaya garis ini adalah pengaturan awal yang menghasilkan garis dengan tipe 'solid', yaitu garis yang tidak memiliki celah atau putus-putus.



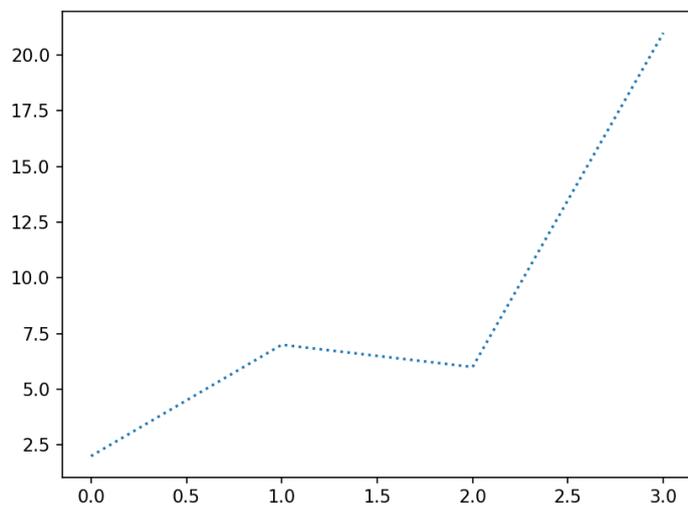
b. Garis Putus-putus:

`linestyle='--'` atau `linestyle='dashed'`: Gaya garis ini menghasilkan garis putus-putus.



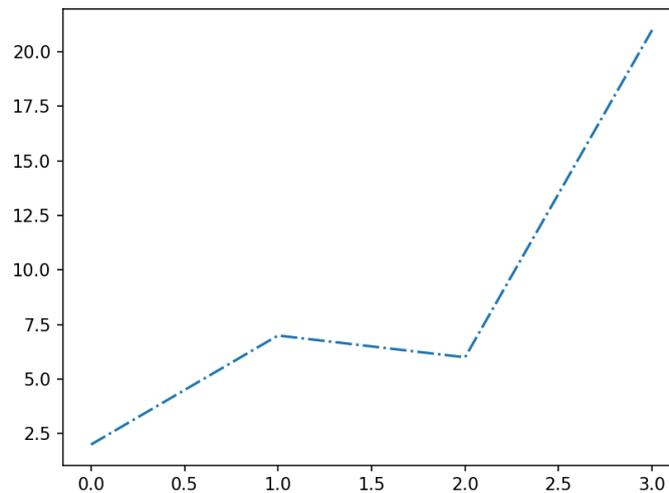
c. Garis Titik-titik:

`linestyle='.'` atau `linestyle='dotted'`: Gaya garis ini menghasilkan garis berupa titik-titik.



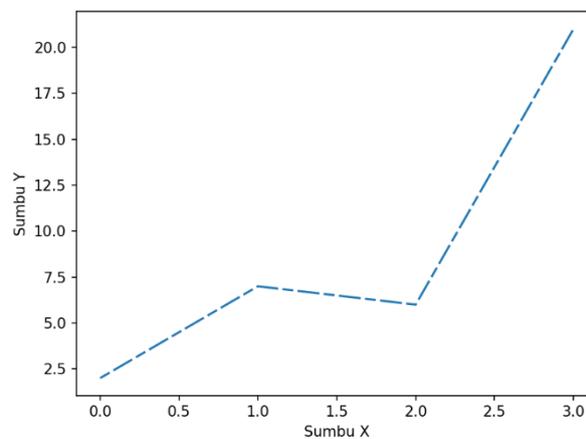
d. Garis Putus-ditambah:

`linestyle='-.'` atau `linestyle='dashdot'`: Gaya garis ini menghasilkan garis putus-ditambah.



e. Pola Garis Kustom:

Anda dapat menentukan pola garis kustom menggunakan parameter `dashes`. Misalnya, `dashes=[5, 2, 10, 2]` menghasilkan garis dengan pola ulang 5 titik hidup, 2 titik mati, 10 titik hidup, dan 2 titik mati.



6.3.6. Python Matplotlib Labels

Untuk memberi label pada sumbu horizontal (x) dan vertikal (y) pada grafik yang dibuat dengan Pyplot, gunakanlah fungsi `xlabel()` dan `ylabel()`.

```

import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Labels

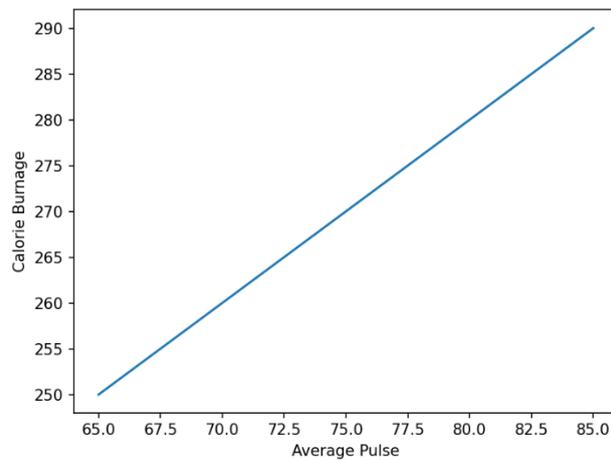
x = np.array([65, 70, 75, 80, 85])
y = np.array([250, 260, 270, 280, 290])

plt.plot(x, y)

plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.show()

```



6.3.7. Python Matplotlib Grid

Fungsi grid() pada objek sumbu menyediakan kontrol penuh terhadap tampilan grid pada grafik. Parameter-parameter yang dapat diatur meliputi visibilitas grid, visibilitas tanda centang mayor dan minor, serta atribut garis grid seperti warna, gaya, dan lebar.

```

import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Grid

x = np.array([65, 70, 75, 80, 85])
y = np.array([250, 260, 270, 280, 290])

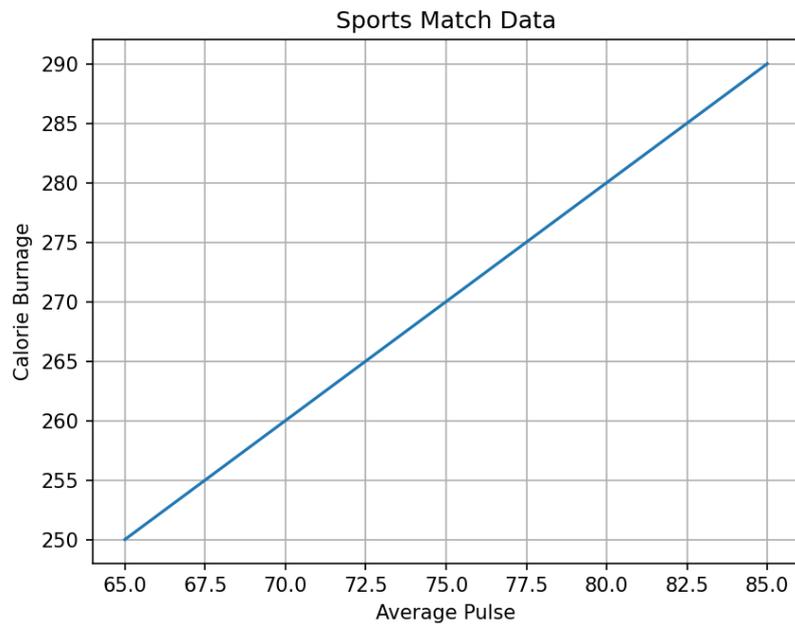
plt.title("Sports Match Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)

plt.grid()

plt.show()

```



6.3.8. Python Matplotlib Subplot

Fungsi subplot() digunakan untuk membagi area gambar menjadi beberapa bagian, sehingga kita dapat membuat beberapa plot yang berbeda dalam satu gambar.

```
import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Subplot

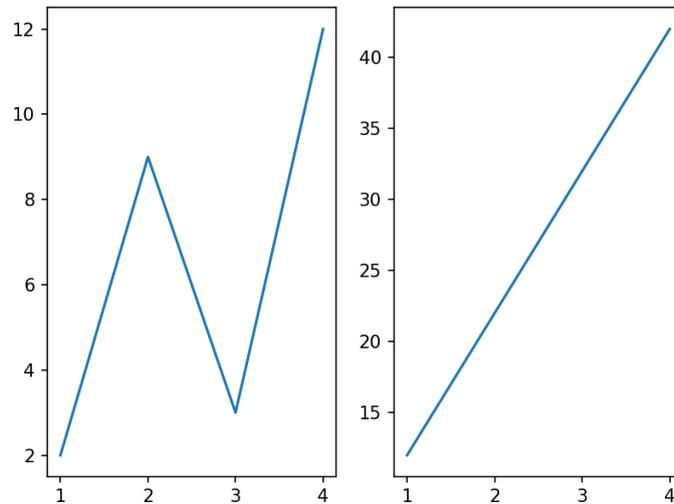
#plot 1
x = np.array([1, 2, 3, 4])
y = np.array([2, 9, 3, 12])

plt.subplot(1, 2, 1)
plt.plot(x, y)

#plot 2
x = np.array([1, 2, 3, 4])
y = np.array([12, 22, 32, 42])

plt.subplot(1, 2, 2)
plt.plot(x, y)

plt.show()
```



6.3.9. Python Matplotlib Scatter

Fungsi `scatter()` di Pyplot memungkinkan kita membuat grafik hamburan dengan memplot setiap data sebagai titik tunggal. Untuk menggunakannya, kita perlu memberikan dua kumpulan data yang berisi nilai-nilai pada sumbu x dan y.

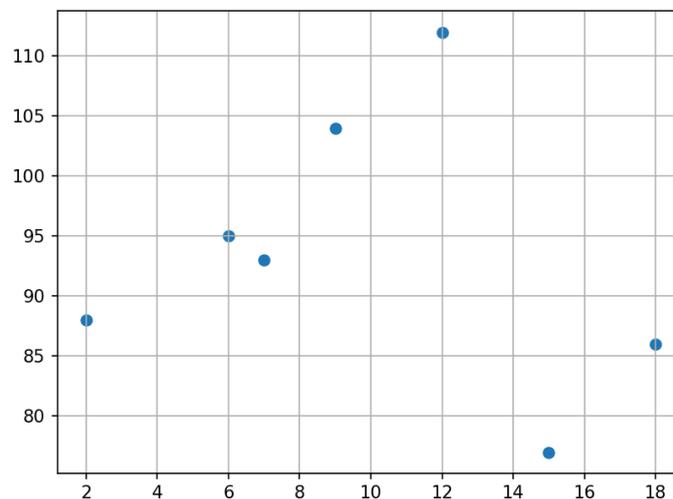
```
import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Scatter

x = np.array([7, 12, 2, 18, 9, 15, 6])
y = np.array([93, 112, 88, 86, 104, 77, 95])

plt.grid()

plt.scatter(x, y)
plt.show()
```



6.3.10. Python Matplotlib Bars

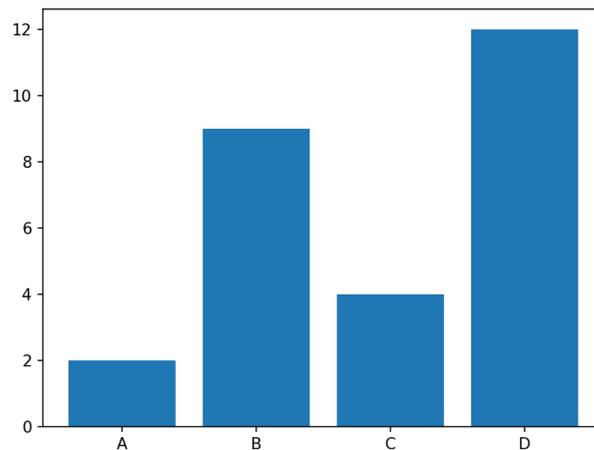
Bar Plot atau Bar Chart adalah jenis plot yang sangat berguna untuk visualisasi data kategorikal. Setiap kategori akan direpresentasikan oleh sebuah batang, dengan panjang batang yang proporsional dengan nilai data.

```
import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Bar

x = np.array(["A", "B", "C", "D"])
y = np.array([2, 9, 4, 12])

plt.bar(x, y)
plt.show()
```



6.3.11. Python Matplotlib Histograms

Untuk membuat histogram di Matplotlib, kita menggunakan fungsi hist(). Fungsi ini menerima sebuah array berisi data numerik sebagai input, lalu akan mengelompokkan data tersebut ke dalam beberapa interval dan menampilkannya dalam bentuk grafik batang.

```

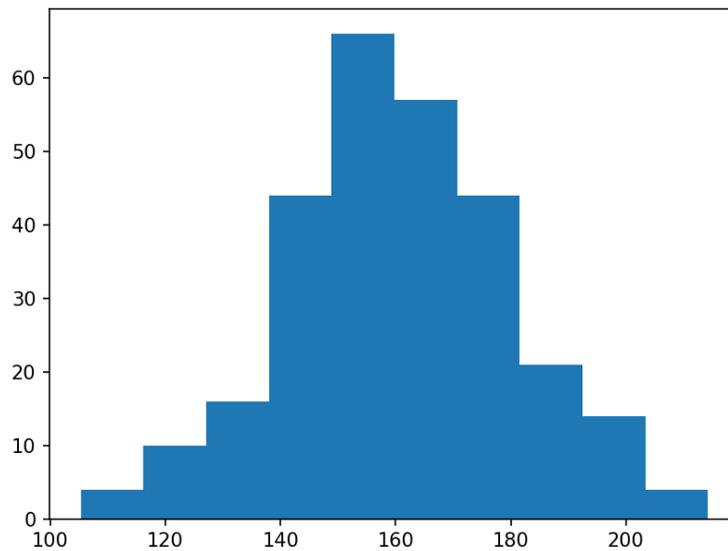
import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Histograms

x = np.random.normal(160, 20, 280)

plt.hist(x)
plt.show()

```



6.3.12. Python Matplotlib Pie Charts

Fungsi `pie()` dalam modul `pyplot` berfungsi untuk membuat visualisasi data dalam bentuk diagram lingkaran.

```

import matplotlib.pyplot as plt
import numpy as np

#Matplotlib Pie Charts

x = np.array([25, 25, 35, 15])
mylabels = ["Jeruk", "Pisang", "Apel", "Duku"]

plt.pie(x, labels = mylabels)
plt.show()

```

