



Modul Praktikum **Kecerdasan Buatan**



Visualisasi Data

Pengertian Visualisasi Data

Visualisasi Data atau *data visualization* adalah tampilan berupa grafis atau visual dari informasi dan data. Dengan kata lain, visualisasi data mengubah kumpulan data menjadi hal lebih sederhana untuk ditampilkan. Dengan menggunakan beberapa elemen visual, orang-orang akan lebih mudah memahami tren, outliers, dan pola dalam suatu data.

Dalam bisnis dan entrepreneur, visualisasi data memungkinkan melihat data secara analitik yang disajikan secara visual dalam membuat sebuah keputusan. Dengan begitu, mereka dapat memahami konsep atau mengidentifikasi pola baru. Hal ini akan membuat pengambilan keputusan menjadi lebih mudah dan tepat.

Kebutuhan Visualisasi Data



Menurut penelitian yang dilakukan oleh Massachusetts Institute of Technology, 90% informasi yang diterima oleh otak seseorang berbentuk visual dan sekitar 80% orang akan mengingat apa yang dilihat dibanding apa yang didengar dan dibaca. Artinya, manusia adalah makhluk visual.

Hal ini didukung dengan hasil penelitian yang menyatakan bahwa otak manusia akan memproses gambar 60.000 kali lebih cepat daripada teks. Untuk itulah memvisualisasikan data menjadi penting agar kumpulan data besar yang dimiliki dapat diproses dan dipahami lebih cepat.

Visualisasi data mempermudah penyampaian informasi secara efektif. Para analis yang bekerja dengan data mentah dengan jumlah besar akan menggunakan



sistem visualisasi data. Bila prosesnya tepat, maka hal ini akan memudahkan pembuatan kesimpulan dan menyederhanakan proses pengambilan keputusan.

Untuk memudahkan proses visualisasi data. Terdapat beberapa tahapan yang dapat diikuti sebagai berikut.

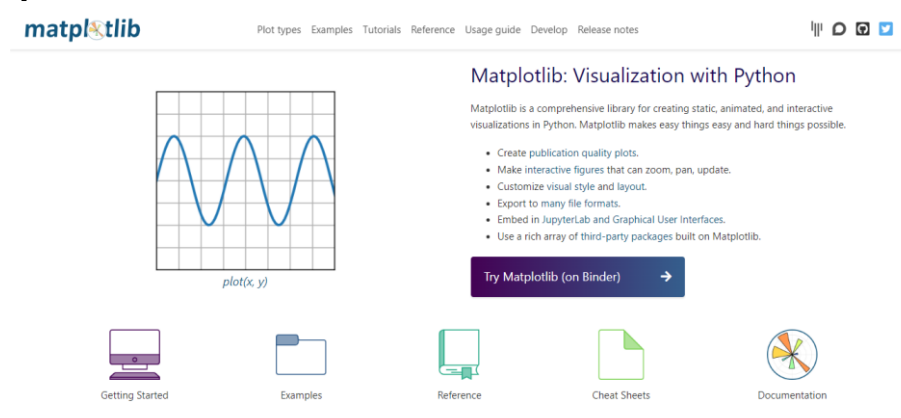
- Identifikasi tujuan mempresentasikan data
- Pilih visualisasi yang sesuai
- Hilangkan grafik yang tidak diperlukan
- Fokus pada insight yang ditekankan
- Ceritakan dengan baik

Library Python dalam Visualisasi Data

1. Pandas

Pandas adalah *open source python package* yang paling banyak digunakan dalam data science, data analysis, dan machine learning. Dimana Pandas dibangun di atas *python package* lain bernama Numpy, yang menyediakan dukungan untuk array multi-dimensi. Pandas digunakan untuk mengolah data dalam bentuk dataframe.

2. Matplotlib



Matplotlib adalah *cross-platform*, visualisasi data, dan *plotting library* grafis untuk Python dan ekstensi numerik NumPy. Pengembang juga dapat menggunakan API (Application Programming Interfaces) matplotlib untuk menyematkan plot dalam aplikasi GUI. Salah satu API dari matplotlib yang sering dipakai adalah Pyplot.

Cara menginstall matplotlib dalam command prompt adalah sebagai berikut.

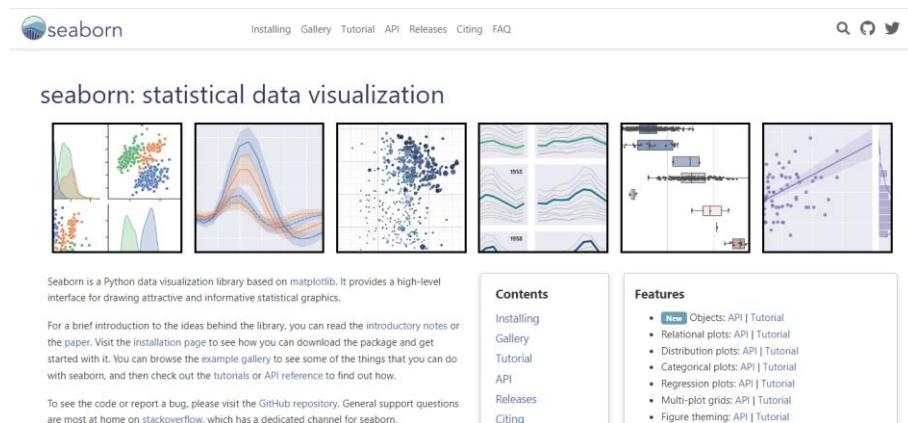


```
pip install matplotlib
```

```
Command Prompt
Microsoft Windows [Version 10.0.19044.2006]
(c) Microsoft Corporation. All rights reserved.

C:\Users\arifh>pip install matplotlib
Defaulting to user installation because normal site-packages is not writeable
Collecting matplotlib
  Downloading matplotlib-3.6.0-cp310-cp310-win_amd64.whl (7.2 MB)
    -----7.2/7.2 MB 1.4 MB/s eta 0:00:00
Requirement already satisfied: fonttools>=4.22.0 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (4.33.3)
Requirement already satisfied: numpy>=1.19 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (1.23.3)
Collecting contourpy>=1.0.1
  Downloading contourpy-1.0.5-cp310-cp310-win_amd64.whl (164 kB)
    -----164.1/164.1 kB 894.8 kB/s eta 0:00:00
Requirement already satisfied: packaging>=20.0 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (21.3)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: pillow>=6.2.0 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (9.2.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib) (0.11.0)
```

3. Seaborn



Seaborn adalah pustaka visualisasi data Python berdasarkan matplotlib. Package ini menyediakan antarmuka tingkat tinggi untuk menggambar grafik statistik yang menarik dan informatif. Cara menginstall seaborn pada command prompt adalah sebagai berikut

```
pip install seaborn
```

```
Command Prompt
C:\Users\arifh>pip install seaborn
Defaulting to user installation because normal site-packages is not writeable
Collecting seaborn
  Downloading seaborn-0.12.0-py3-none-any.whl (285 kB)
    -----285.1/285.1 kB 42.7 kB/s eta 0:00:00
Requirement already satisfied: numpy>=1.17 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from seaborn) (1.23.3)
Requirement already satisfied: matplotlib>=3.1 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from seaborn) (3.6.0)
Requirement already satisfied: pandas>=0.25 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from seaborn) (1.4.4)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib>=3.1->seaborn) (1.0.5)
Requirement already satisfied: packaging>=20.0 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib>=3.1->seaborn) (21.3)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib>=3.1->seaborn) (4.33.3)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib>=3.1->seaborn) (1.4.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib>=3.1->seaborn) (2.8.2)
Requirement already satisfied: cycler>=0.10 in c:\users\arifh\appdata\roaming\python\python310\site-packages (from matplotlib>=3.1->seaborn) (0.11.0)
```



Visualisasi Data dengan Matplotlib dan Seaborn

Sebelum kita memvisualisasi data dari dataset berikut [Kaggle/iris.csv](https://www.kaggle.com/irisdataset/iris), kita dapat mengimport package Pandas dengan matplotlib dan menampilkan dataset dengan menggunakan package Pandas pada Python.

```
import pandas as pd  
from matplotlib import pyplot as plt
```

```
df = pd.read_csv('Iris.csv')  
df
```

```
[5]: import pandas as pd  
      from matplotlib import pyplot as plt  
  
[6]: df = pd.read_csv('Iris.csv')  
      df  
  
[6]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

Kemudian, kita akan melihat informasi dari berapa jumlah spesies dengan menggunakan

```
df['Species'].value_counts()
```

```
[10]: df['Species'].value_counts()  
  
[10]: Iris-setosa      50  
      Iris-versicolor  50  
      Iris-virginica   50  
      Name: Species, dtype: int64
```



1. Grafik Batang

Grafik batang sangat cocok untuk memvisualisasi data kategorikal. Grafik ini biasa digunakan untuk membandingkan kategori. Kita dapat memunculkan visualisasi data dengan menggunakan syntax berikut.

```
x = list(df.iloc[:,5])  
y = list(df.iloc[:,2])
```

```
plt.bar(x, y, color='red')  
plt.title("Panjang Sepal")  
plt.xlabel("Jenis Iris")  
plt.ylabel("Panjang Sepal")
```



2. Grafik Plot dan Garis

Grafik Plot dan Garis sangat cocok digunakan di dalam data time series, karena dataset di atas bukan termasuk time series, anda dapat mengunduh data time series pada link berikut [Kaggle/TimeSeries](https://www.kaggle.com/datasets/timelseries) dan untuk visualisasi data dengan diagram garis sebagai berikut.

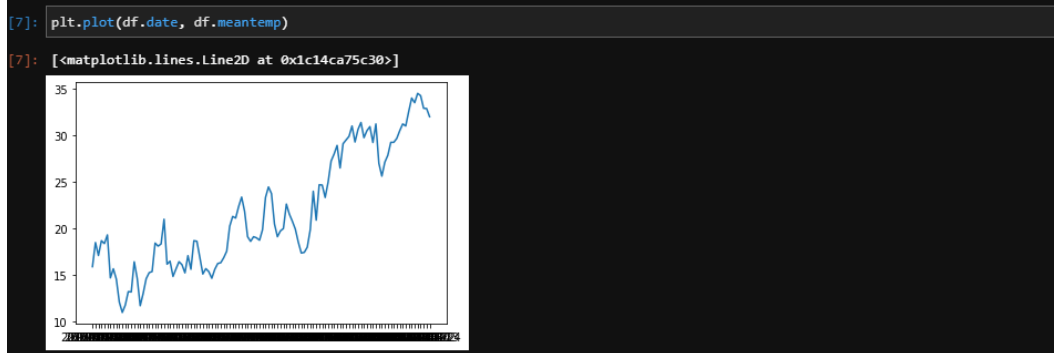
```
import pandas as pd  
from matplotlib import pyplot as plt
```



```
df = pd.read_csv('DailyDelhiClimateTest.csv')  
  
df.head()  
  
plt.plot(df.date, df.meantemp)
```

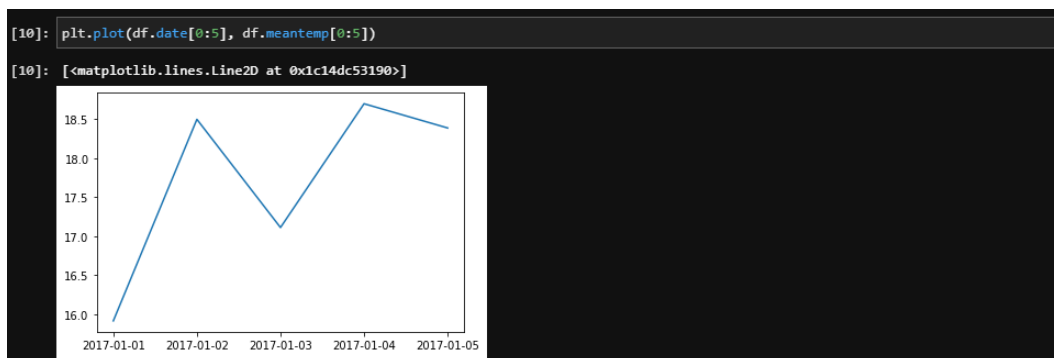
```
[8]: import pandas as pd  
from matplotlib import pyplot as plt  
  
[9]: df = pd.read_csv('DailyDelhiClimateTest.csv')  
df.head()  
  
[9]:
```

	date	meantemp	humidity	wind_speed	meanpressure
0	2017-01-01	15.913043	85.869565	2.743478	59.000000
1	2017-01-02	18.500000	77.222222	2.894444	1018.277778
2	2017-01-03	17.111111	81.888889	4.016667	1018.333333
3	2017-01-04	18.700000	70.050000	4.545000	1015.700000
4	2017-01-05	18.388889	74.944444	3.300000	1014.333333



Dengan menggunakan Index

```
plt.plot(df.date[0:5], df.meantemp[0:5])
```





3. Grafik Pie

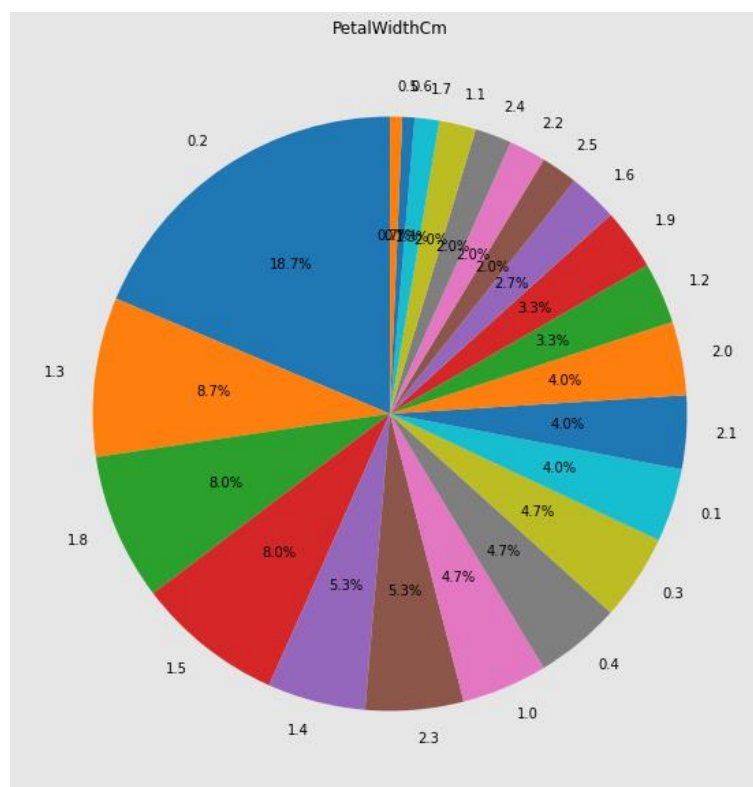
Sama dengan Grafik Batang, Grafik Pie cocok digunakan pada data categorical. Grafik ini biasa digunakan untuk melihat distribusi. Implementasi grafik pie dengan dataset iris.csv adalah sebagai berikut.

```
df = df.dropna(subset=['PetalWidthCm'])
```

```
data_pie =  
df['PetalWidthCm'].value_counts().rename_axis('PetalWidthCm').reset_index(n  
ame='Id_count')
```

```
plt.figure(figsize=(10,10))  
  
plt.pie(data_pie.Id_count, labels=data_pie.PetalWidthCm, startangle=90,  
autopct='%.1f%%')  
  
plt.title('PetalWidthCm')  
plt.show()
```

```
[60]: df = df.dropna(subset=['PetalWidthCm'])  
[68]: data_pie = df['PetalWidthCm'].value_counts().rename_axis('PetalWidthCm').reset_index(name='Id_count')  
[69]: plt.figure(figsize=(10,10))  
plt.pie(data_pie.Id_count, labels=data_pie.PetalWidthCm, startangle=90, autopct='%.1f%%')  
plt.title('PetalWidthCm')  
plt.show()
```





4. Scatter Plot

Scatter Plot atau yang memiliki nama lain plot pencar merupakan diagram grafis yang dibangun dari dua sumbu X dan Y (variabel X dan variabel Y). Scatter plot menunjukkan hubungan antara dua variabel. Scatter plot sering digunakan sebagai analisis tindak lanjut untuk menentukan apakah penyebab yang ada benar-benar memberikan dampak kepada karakteristik kualitas. Selain itu scatter plot juga digunakan untuk melihat kelompok data dalam clustering dan melihat outliers. Cara mengimplementasikan scatter plot adalah sebagai berikut.

```
import matplotlib.pyplot as plt  
import seaborn as sns  
import pandas as pd
```

```
df = pd.read_csv('Iris.csv')  
df.head()
```

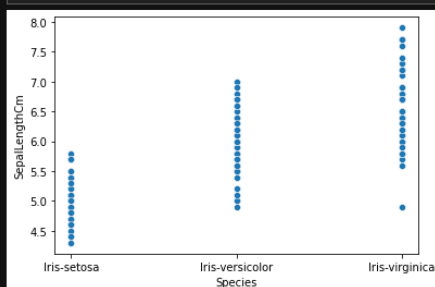
```
sns.scatterplot(data = df, x = "Species", y = "SepalLengthCm")  
plt.show()
```

```
[4]: import matplotlib.pyplot as plt  
import seaborn as sns  
import pandas as pd
```

```
[5]: df = pd.read_csv('Iris.csv')  
df.head()
```

[5]:		Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa	
1	2	4.9	3.0	1.4	0.2	Iris-setosa	
2	3	4.7	3.2	1.3	0.2	Iris-setosa	
3	4	4.6	3.1	1.5	0.2	Iris-setosa	
4	5	5.0	3.6	1.4	0.2	Iris-setosa	

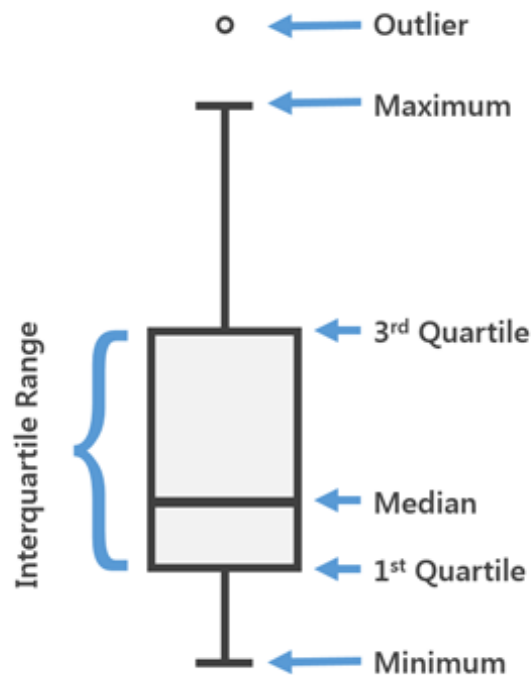
```
[8]: sns.scatterplot(data = df, x = "Species", y = "SepalLengthCm")  
plt.show()
```





5. Grafik Boxplot

Grafik Boxplot merupakan ringkasan distribusi sampel yang disajikan secara grafis yang bisa menggambarkan bentuk distribusi data (skewness), ukuran tendensi sentral dan ukuran penyebaran (keragaman) data pengamatan. Biasanya grafik ini digunakan untuk melihat persebaran data dan mencari outliers.



Implementasi grafik boxplot sebagai berikut

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
```

```
df = pd.read_csv('Iris.csv')
df.head()
```

```
df.isnull().values.any()
```

```
SpL = df['SepalLengthCm']
SpW = df['SepalWidthCm']
PtL = df['PetalLengthCm']
```



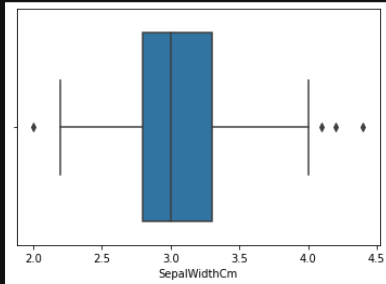
```
PtW = df['PetalWidthCm']
```

```
sns.boxplot(x=SpW)
```

```
plt.show()
```

```
[17]: SpL = df['SepallengthCm']  
      SpW = df['SepalWidthCm']  
      PEL = df['PetalLengthCm']  
      PtW = df['PetalWidthCm']
```

```
[23]: sns.boxplot(x=SpW)  
      plt.show()
```



```
[14]: import matplotlib.pyplot as plt  
      import seaborn as sns  
      import pandas as pd
```

```
[15]: df = pd.read_csv('Iris.csv')  
      df.head()
```

```
[15]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
[16]: df.isnull().values.any()
```

```
[16]: False
```

6. Grafik Heatmap

Heatmap disebut juga visualisasi atau pemetaan dengan menampilkan data dengan cara representasi warna yang berbeda-beda. Biasanya pada heatmap semakin tinggi angka suatu kelompok data maka warnanya akan semakin gelap dan umumnya disimbolkan dengan warna berwarna merah. Grafik ini biasa digunakan untuk menunjukkan korelasi antar variable. Implementasi heatmap dataset iris .csv adalah sebagai berikut.

```
import matplotlib.pyplot as plt  
  
import seaborn as sns  
  
import pandas as pd
```

```
df = pd.read_csv('Iris.csv')  
  
df.head()
```



```
fig, ax = plt.subplots(figsize=(10,6))  
  
sns.heatmap(df.corr(), center=0, cmap='Blues')  
  
ax.set_title('Korelasi antar atribut pada iris.csv')
```

```
[1]: import matplotlib.pyplot as plt  
import seaborn as sns  
import pandas as pd
```

```
[7]: df = pd.read_csv('Iris.csv')  
df.head()
```

```
[7]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
[8]: fig, ax = plt.subplots(figsize=(10,6))  
sns.heatmap(df.corr(), center=0, cmap='Blues')  
ax.set_title('Korelasi antar atribut pada iris.csv')
```

```
[8]: Text(0.5, 1.0, 'Korelasi antar atribut pada iris.csv')
```

