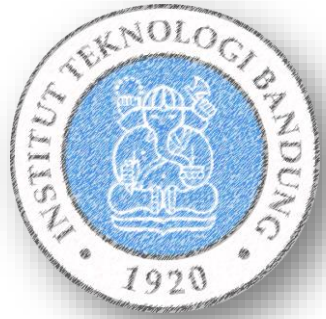


Visualisasi Data (tool: Python)

Tim Penyusun Materi Pengenalan Komputasi
Institut Teknologi Bandung © 2019





Objektif

- Mahasiswa memahami latar belakang dan pentingnya visualisasi data
- Mahasiswa memahami prinsip-prinsip dasar visualisasi data, dan mampu mengembangkan visualisasi data yang tepat
- Mahasiswa dapat menggunakan grafik yang disediakan Python library untuk keperluan visualisasi data

Pentingnya *Insight*

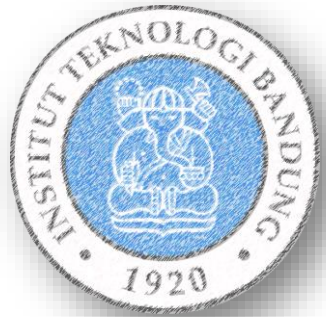


Richard Hamming (1962):
“The purpose of computing is
insight not numbers”

Insight: wawasan/pengertian/pengetahuan yang mendalam

Chris North:
Data-transfer via Vision vs Ears = 100MB/s : (<100b/s)

Statistik tidak cukup?



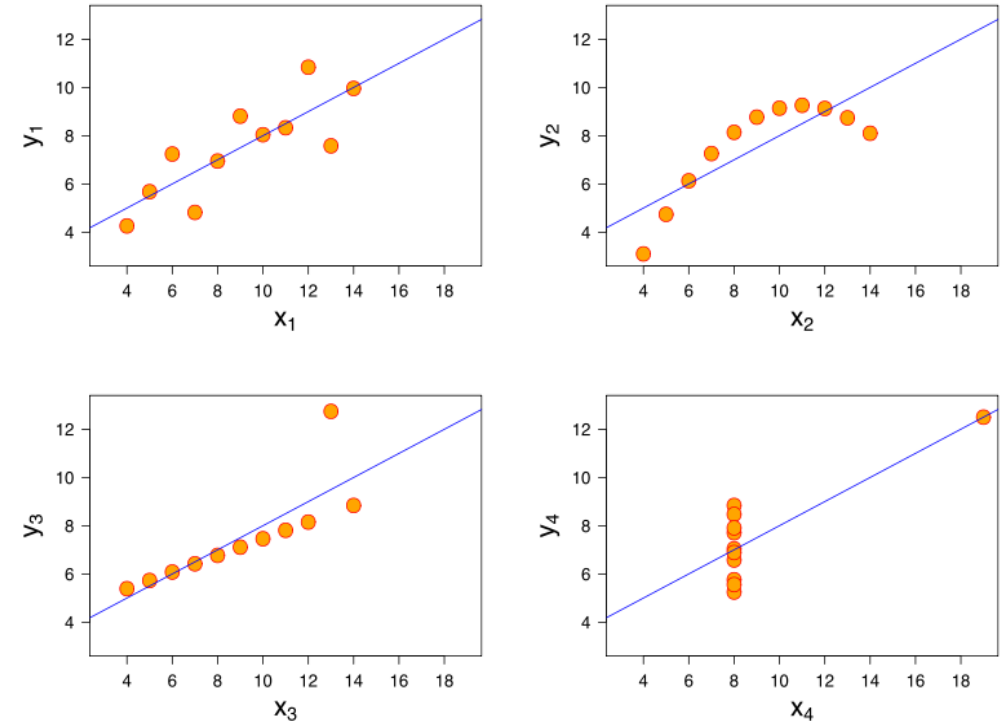
Anscombe's Quartet: Raw Data

	1		2		3		4	
	X	Y	X	Y	X	Y	X	Y
	10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
	8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
	13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
	9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
	11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
	14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
	6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
	4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
	12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
	7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
	5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89
Mean	9.0	7.5	9.0	7.5	9.0	7.5	9.0	7.5
Variance	10.0	3.75	10.0	3.75	10.0	3.75	10.0	3.75
Correlation	0.816		0.816		0.816		0.816	

Statistik tidak cukup? Visualisasi perlu?

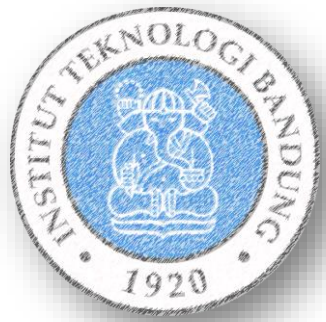
- *Summary statistics* mungkin kehilangan *trend* penting
- Visualisasi data memudahkan data analysis
- Visualisasi data dapat menampilkan kerumitan data menjadi sederhana dan menampilkan berbagai sudut pandang dari data

Anscombe's Quartet



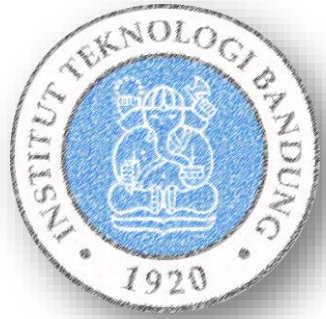
http://en.wikipedia.org/wiki/Anscombe%27s_quartet

(c) Angela Zoss (angela.zoss@duke.edu)



Kelebihan Visualisasi Data

- Memberikan pengertian komprehensif atas data yang banyak
- Memungkinkan persepsi secara cepat terhadap property atau karakteristik penting dari data.
- Dapat memperlihatkan secara cepat persoalan yang ada pada data, misalnya ada nilai data yang tidak masuk akal, outlier, dll
- Memfasilitasi pemahaman terhadap fitur data, baik secara *large-scale* maupun *small-scale*



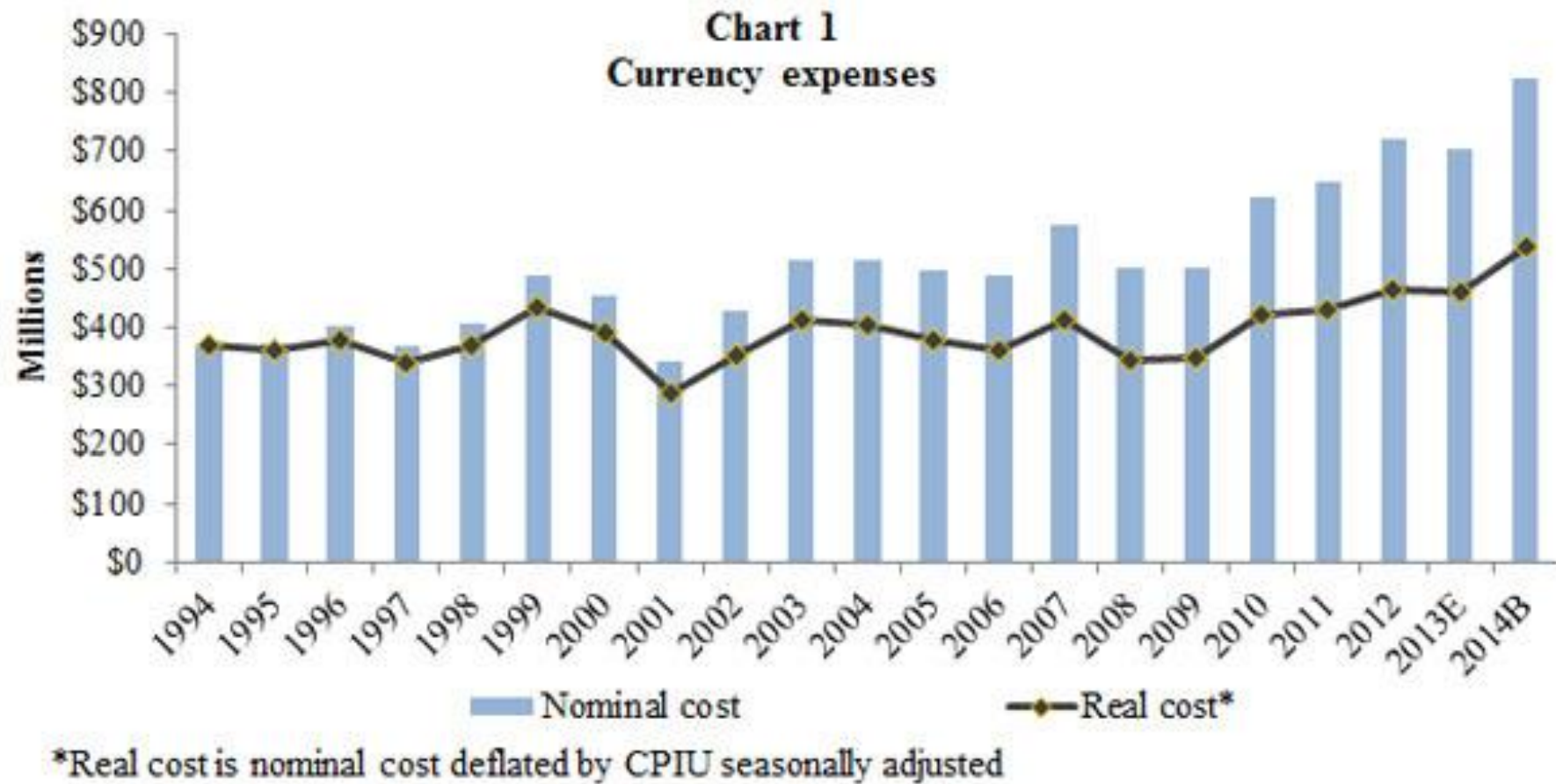
Beberapa Definisi

- Visualisasi Data :
 - *an umbrella term*, mengkonversi sumber data ke dalam sebuah representasi visual
- Visualisasi Saintifik:
 - Visualisasi data saintifik yang berhubungan erat dengan **objek dunia-nyata** yang memiliki property **spasial**
 - Contoh: visualisasi data gempa, visualisasi arah angin
- Visualisasi Information
 - Visualisasi dalam bentuk bagan (chart), grafik, metafora spasial/visual yang digunakan untuk merepresentasikan dataset yang tidak memiliki komponen spasial.
 - Contoh: visualisasi harga saham, visualisasi perbandingan jumlah mahasiswa ITB

Contoh Visualisasi Saintifik

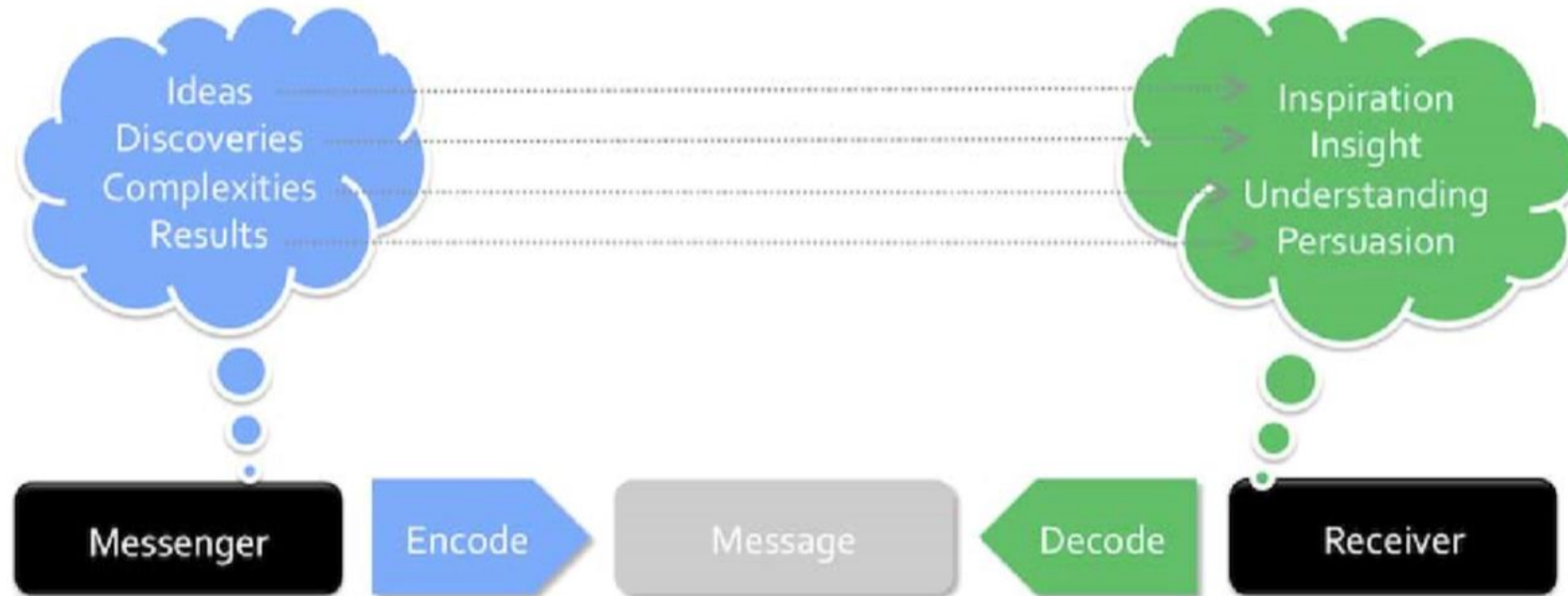


Contoh Visualisasi Informasi



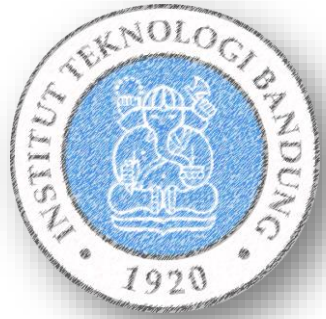
Sumber: <https://www.federalreserve.gov/foia/2014currency.htm>

Data Visualization



Data Visualization:
the **representaton** and **presentation** of data
that exploits our **visual perception abilities**
in order to **amplify cognition**

(c) Andy Kirk, 2012

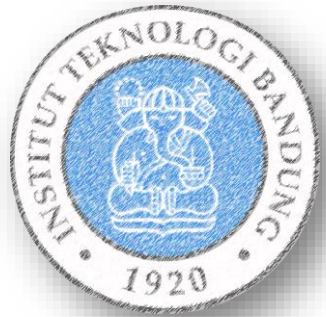
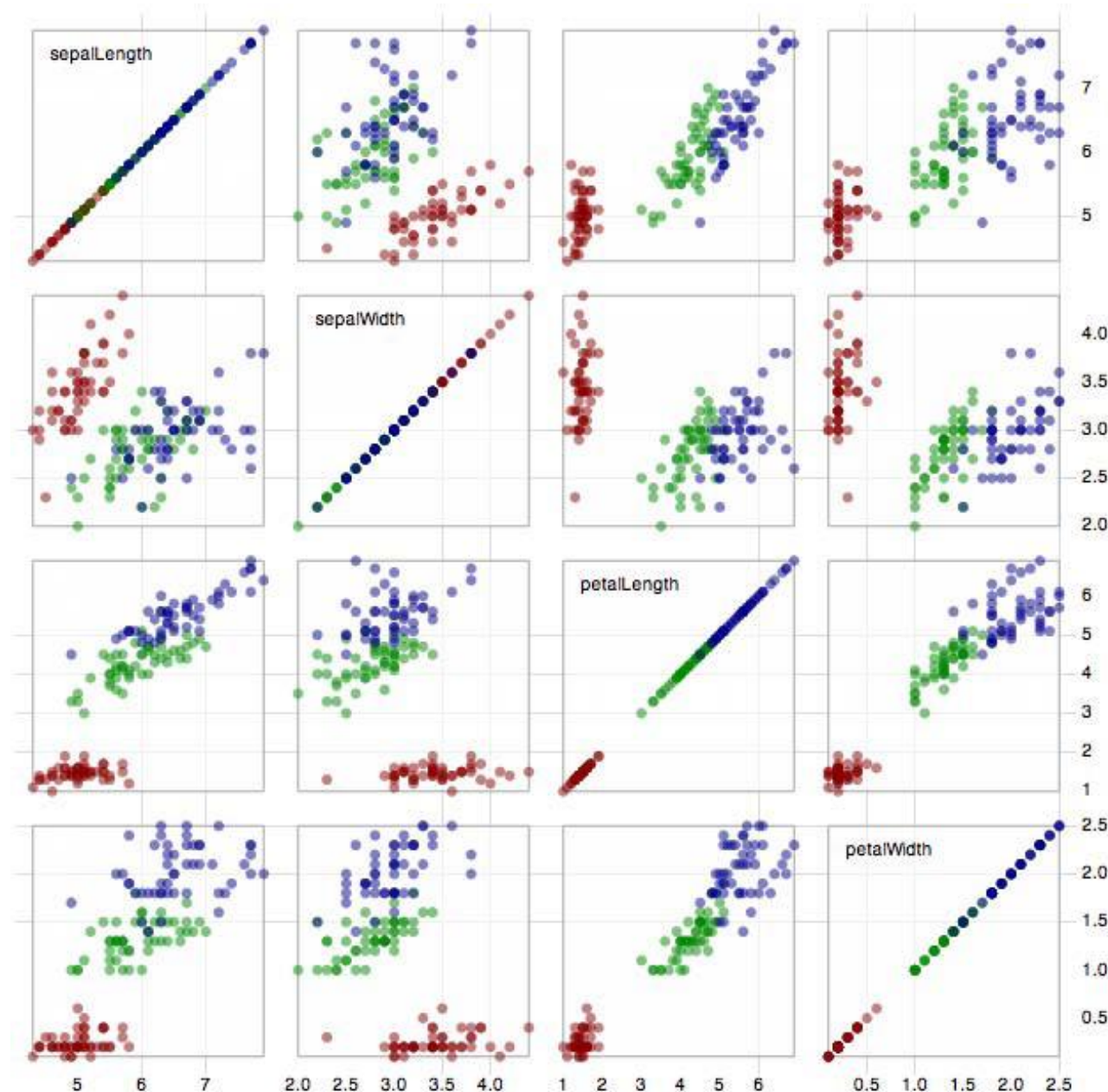


Tujuan Visualisasi:

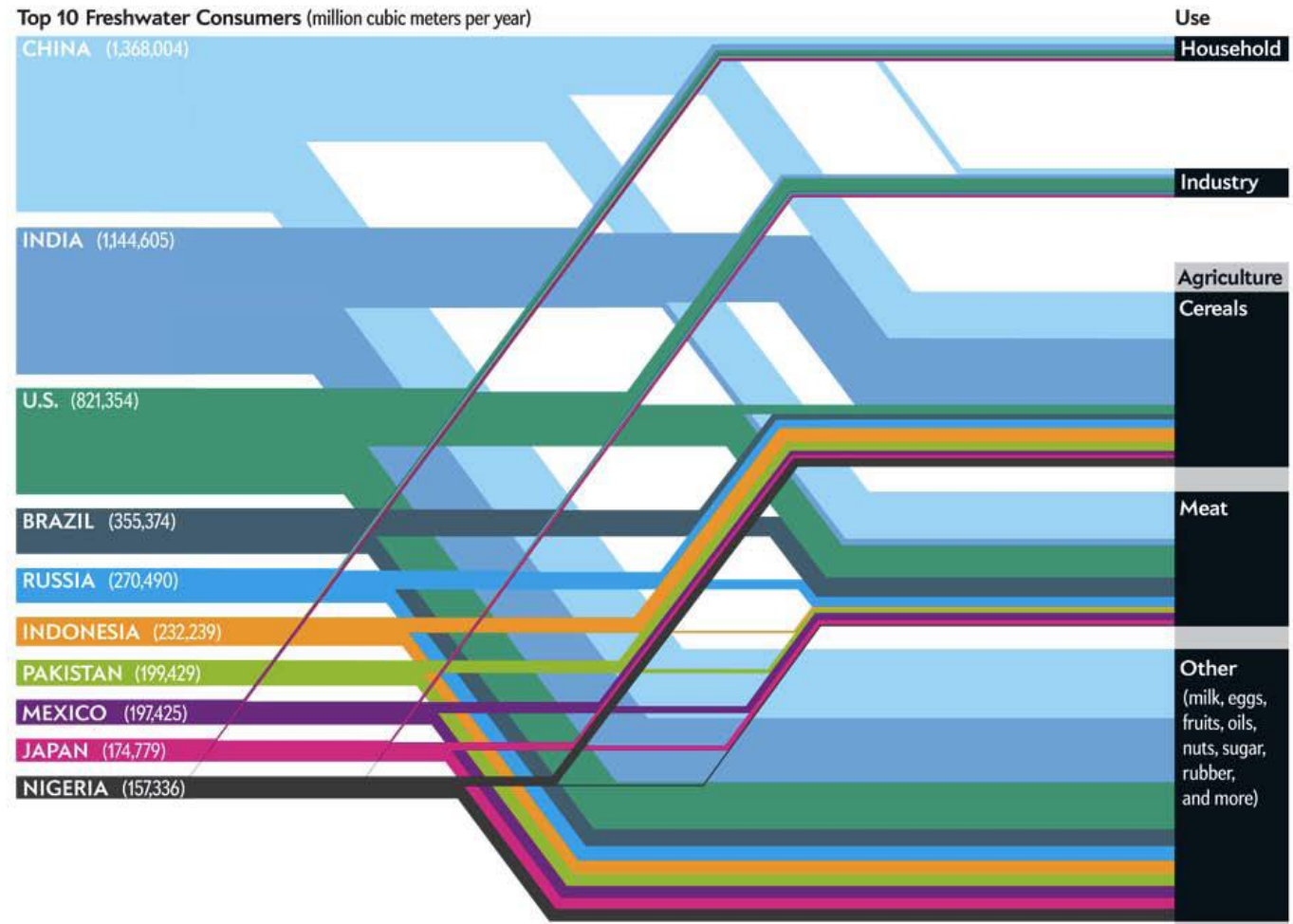
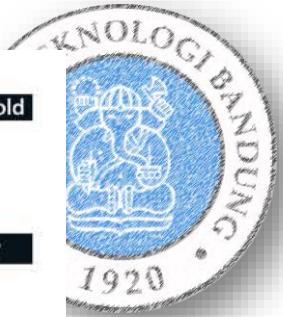
- Visualisasi untuk **analisis** (a.k.a. “**visual analytics**”)
 - Eksploitasi kekuatan persepsi visual untuk mengeksplor atau menganalisis hubungan antar data
 - Biasanya ditampilkan dalam berbagai bentuk atau sudut pandang visualisasi untuk menemukan hubungan yang diinginkan
- Visualisasi untuk **komunikasi**
 - Memilih view tertentu dari data untuk di-share
 - Biasanya dilakukan dengan konstruksi visualisasi dengan tujuan menjelaskan kepada audiens

(c) Angela Zoss (angela.zoss@duke.edu)

Visualisasi untuk Tujuan Analisis (Exploratory)



Visualisasi untuk Tujuan Komunikasi (Explanatory)



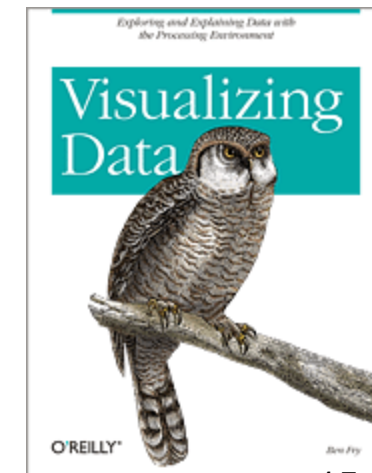
Tahapan Visualisasi Data

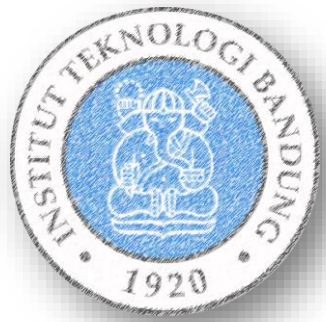
1. **Acquire**: Obtain **the** data...
2. **Parse**: Provide some structure for the data's **meaning**, and order it into categories.
3. **Filter**: Remove all but the data of **interest**.
4. **Mine**: **Apply** methods from statistics or data mining as a way to discern patterns or place the data in mathematical context.
5. **Represent**: **Choose** a basic visual model, such as a bar graph, list, or tree.
6. **Refine**: Improve the basic representation to make it **clearer** and more visually engaging.
7. **Interact**: Add methods for **manipulating** the data or controlling what features are visible.

Note: stages are often iterative and may have a flexible order or even be omitted in some projects.

Fry, B. (2008). [*Visualizing data*](#). Sebastopol, CA: O'Reilly Media, Inc.

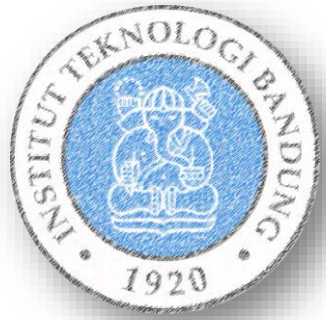
(c) Angela Zoss (angela.zoss@duke.edu)





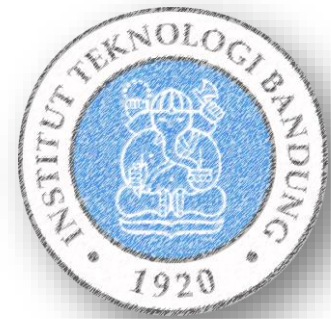
Tipe Data

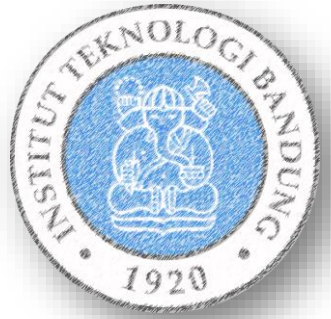
- Categorical-Nominal
 - Nama negara, warna kulit, nama program studi, dll
- Categorical-Ordinal
 - Likert scale (“sangat setuju” s.d. “sangat tidak setuju”)
 - Indeks nilai A, B, C, D, E
- Categorical-Binary
 - Jenis kelamin, status mahasiswa (aktif, tidak aktif), dll
- Quantitative-Discrete
 - Banyaknya anak, banyaknya mahasiswa, banyaknya sks lulus
- Quantitative-Continues
 - Usia, berat badan, tinggi, suhu



Klasifikasi Visualisasi Data

- Perbandingan Kategori (*Comparing Categories*)
- Penampilan Perubahan Terhadap Waktu (*Showing over Times*)
- Penampilan Hirarki dan Hubungan Keseluruhan-Bagian (*Whole-part relationship*)
- *Plotting relationships*
- Pemetaan Data Geospatial (**tidak dibahas di kuliah ini**)



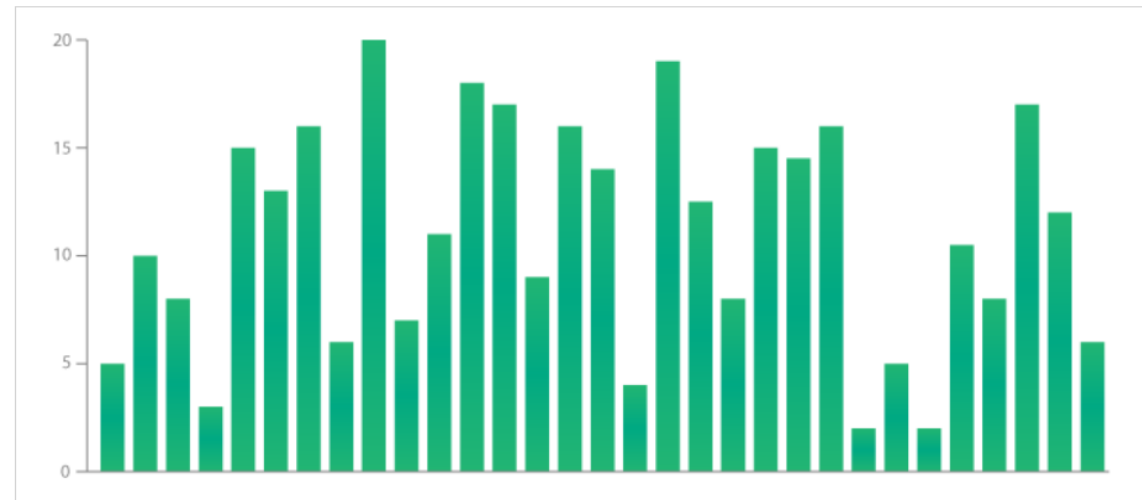
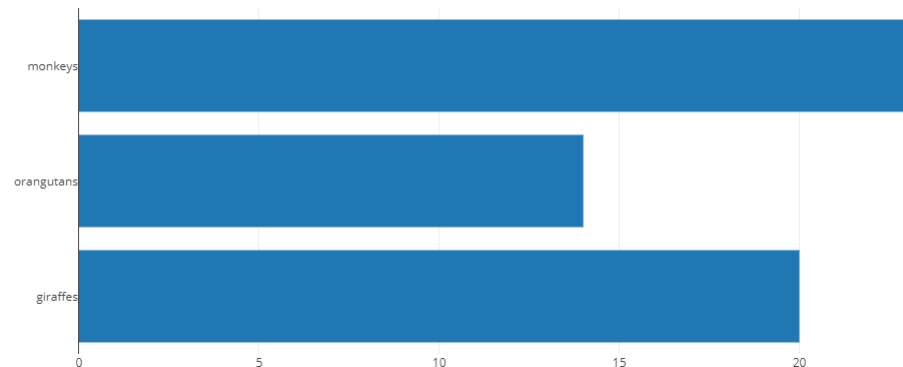


Comparing Categories

- Membandingkan nilai-nilai dari suatu variabel bersifat kategorikal
 - Contoh: membandingkan jumlah mahasiswa untuk beberapa prodi, pendapatan per kapita negara-negara di dunia,
- Grafik yang dapat digunakan:
 - [Vertical/Horizontal] bar chart/column chart
 - Histogram
 - Radial chart
 - Dot plot
 - Dll.

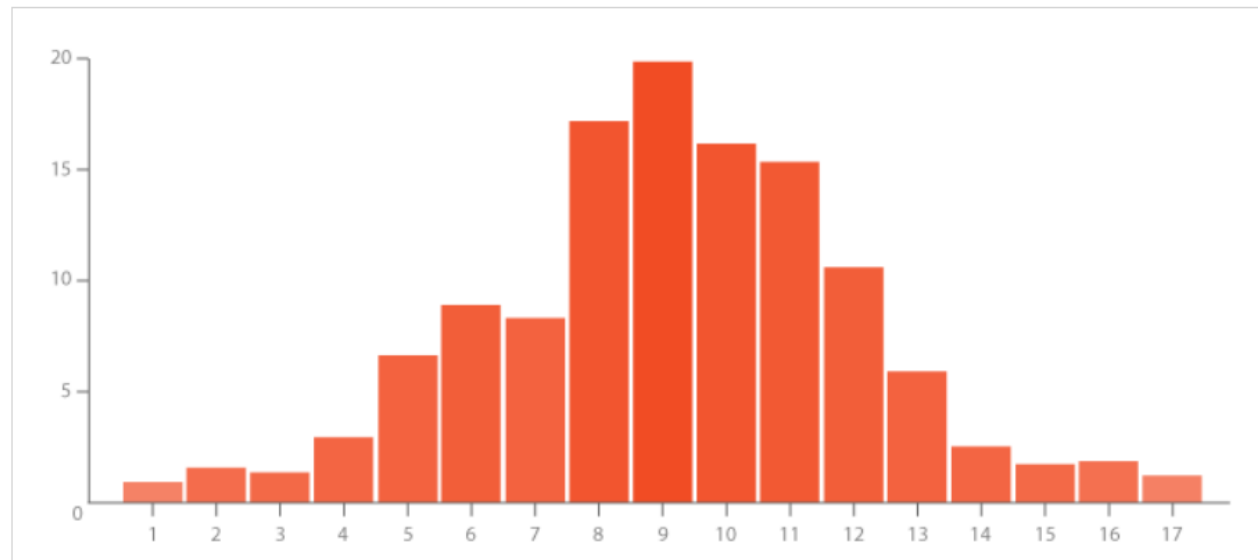
Comparing categories: Bar chart

- **Bar chart/grafik garis:** Menggunakan batang vertikal atau horizontal untuk menunjukkan **perbandingan** nilai-nilai numerik pada kategori-kategori tertentu



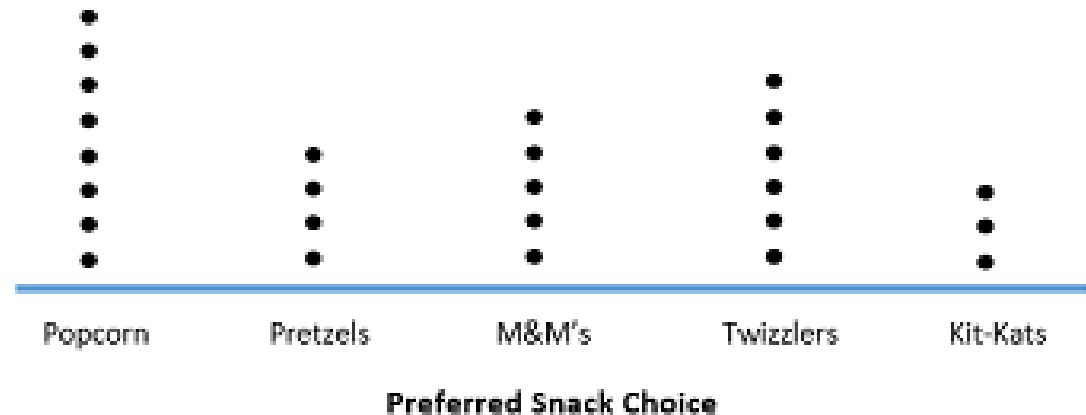
Comparing categories: Histogram chart

- **Histogram:** Memvisualisasikan distribusi data untuk interval-interval nilai atau periode waktu. Setiap batang pada histogram merepresentasikan frekuensi data untuk tiap interval.



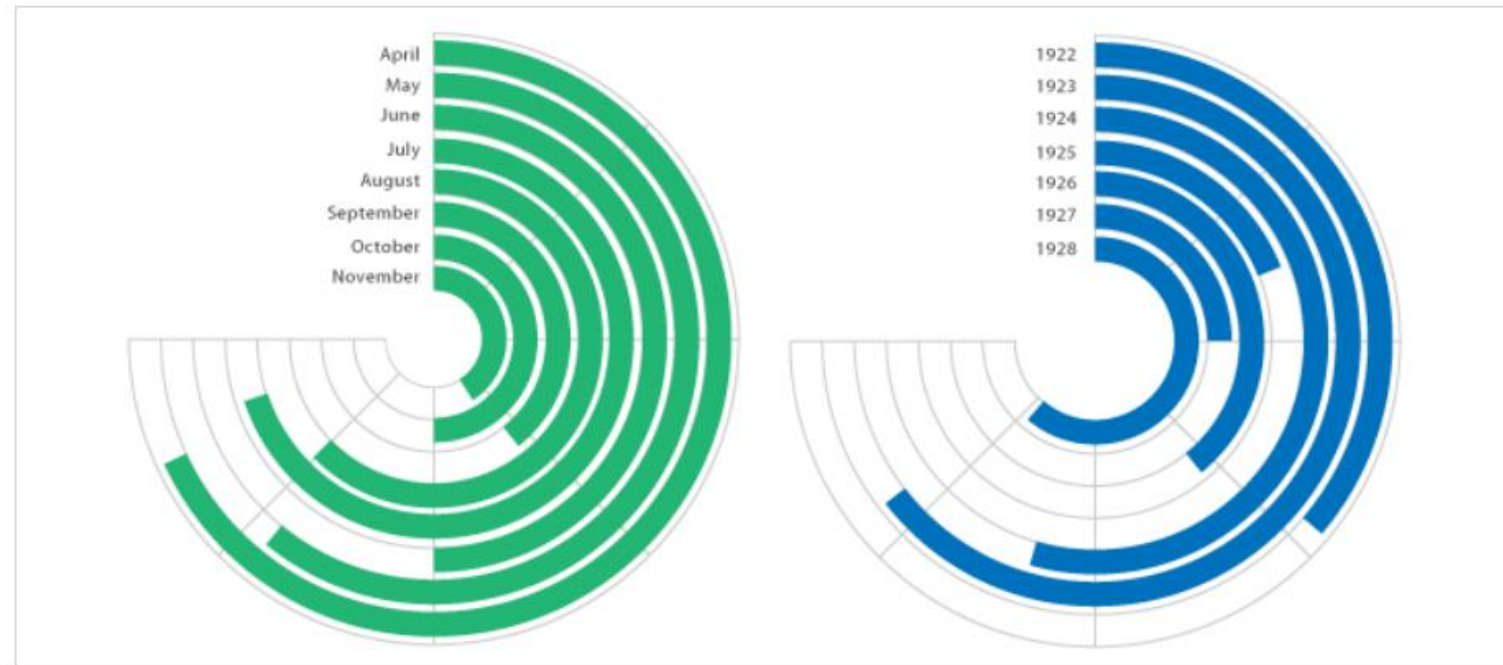
Comparing categories: Dot plot

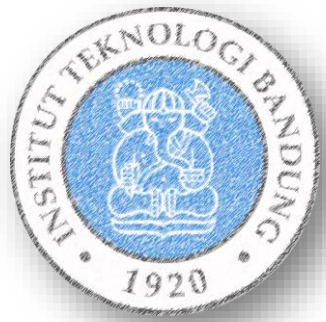
- **Dot chart/dot plot** : adalah chart di mana titik-titik data di-plotkan ke atas skala sederhana,
- Dot plot merupakan alternative dari bar chart untuk merepresentasikan nilai-nilai kuantitatif berasosiasi dengan variable kategorikal



Comparing categories: Radial [bar] chart

- Radial [bar] chart: bar chart yang diplot di atas sistem koordinat polar
 - Problem: panjang batang dapat salah diinterpretasi



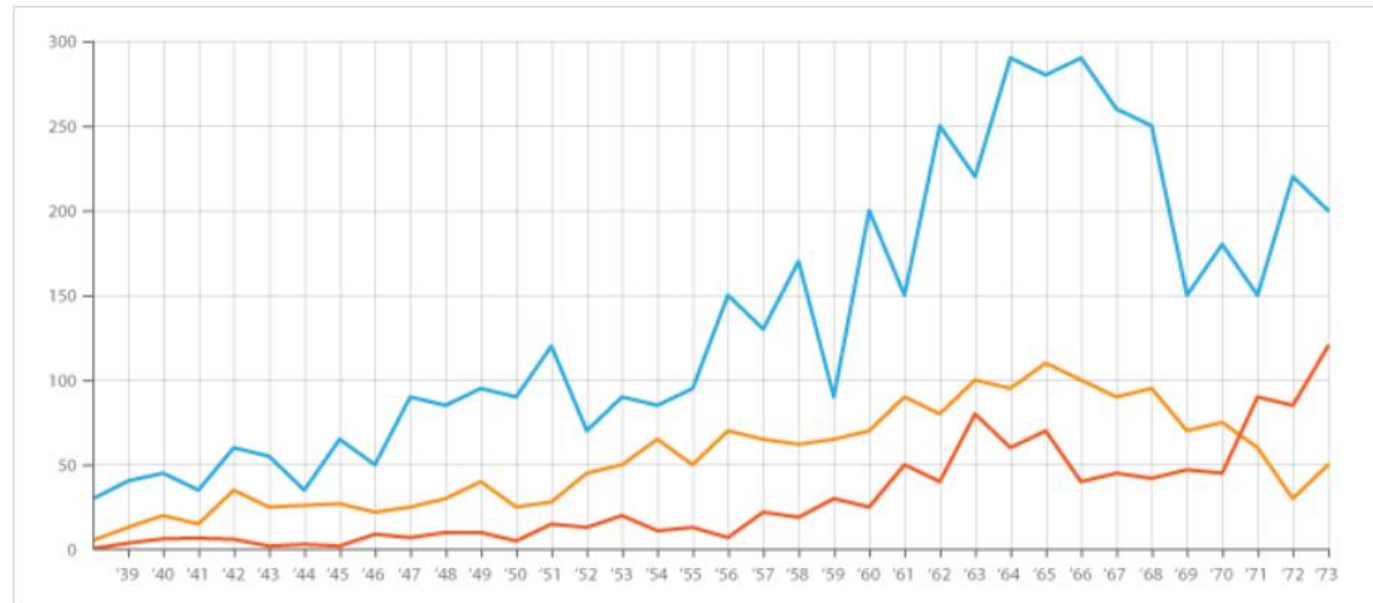


Penampilan Perubahan Terhadap Waktu (*Showing over Times*)

- Visualisasi menampilkan data dalam suatu periode waktu untuk menemukan trend atau perubahan terkait waktu
- Grafik/chart yang dapat digunakan:
 - Line chart
 - Area chart + stacked area chart
 - Histogram
 - Dll.

Showing over times: Line chart

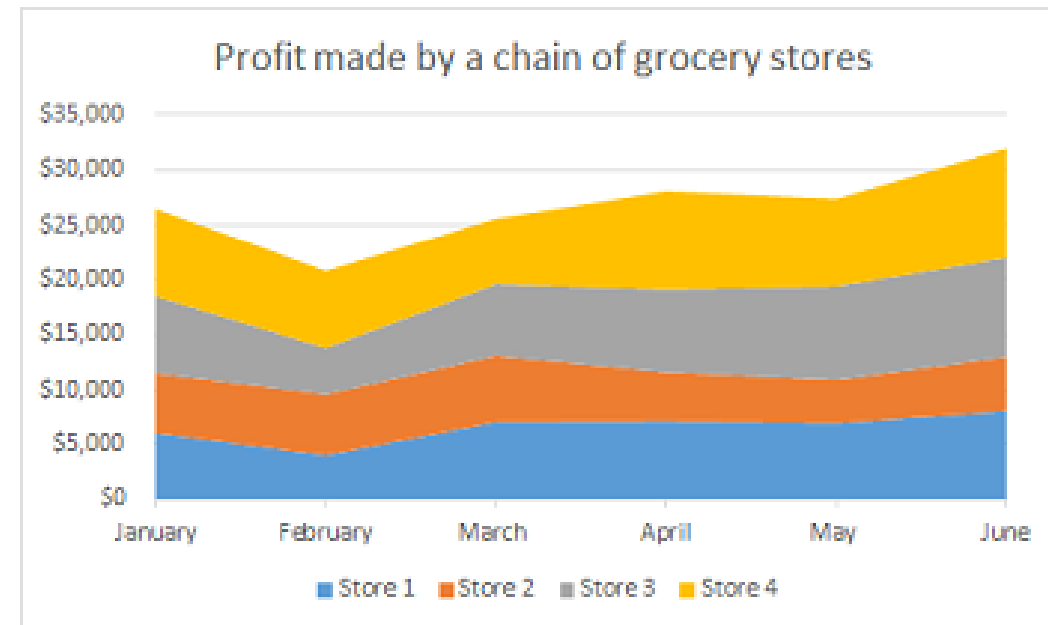
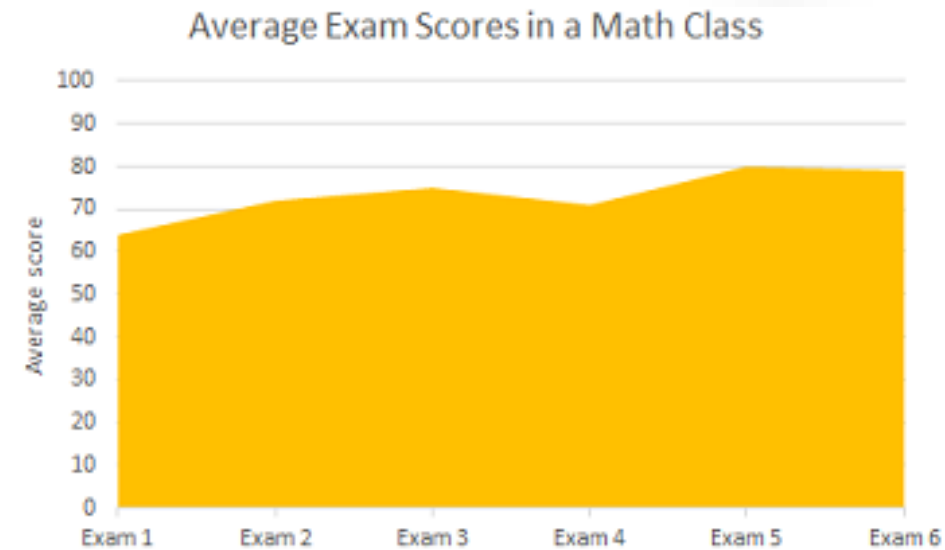
- **Line chart:** menggunakan garis yang menghubungkan titik-titik data untuk menunjukkan perubahan terkait waktu atau interval nilai tertentu

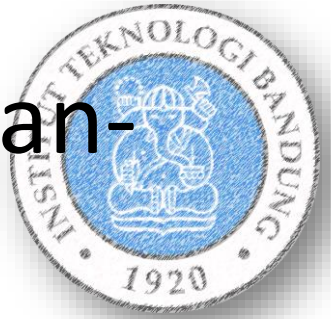


Showing over times:

Area Chart + Stacked Area Chart

- **Area chart** menggunakan wilayah di bawah garis (seperti pada line chart) untuk menyatakan data kuantitatif terkait waktu
- **Stacked Area Chart** terdiri atas beberapa area chart yang bertumpuk satu di atas yang lain yang menandakan kategori yang berbeda





Penampilan Hierarki dan Hubungan Keseluruhan-Bagian (*Whole-part relationship*)

- **Hierarki:** Menampilkan bagaimana ranking atau urutan data atau objek dirangking
- ***Whole-part relationship*:** menunjukkan proporsi bagian-bagian pada suatu variabel dari keseluruhannya
- Grafik yang dapat digunakan:
 - Pie chart
 - Stacked bar chart
 - Treemap
 - Dll.

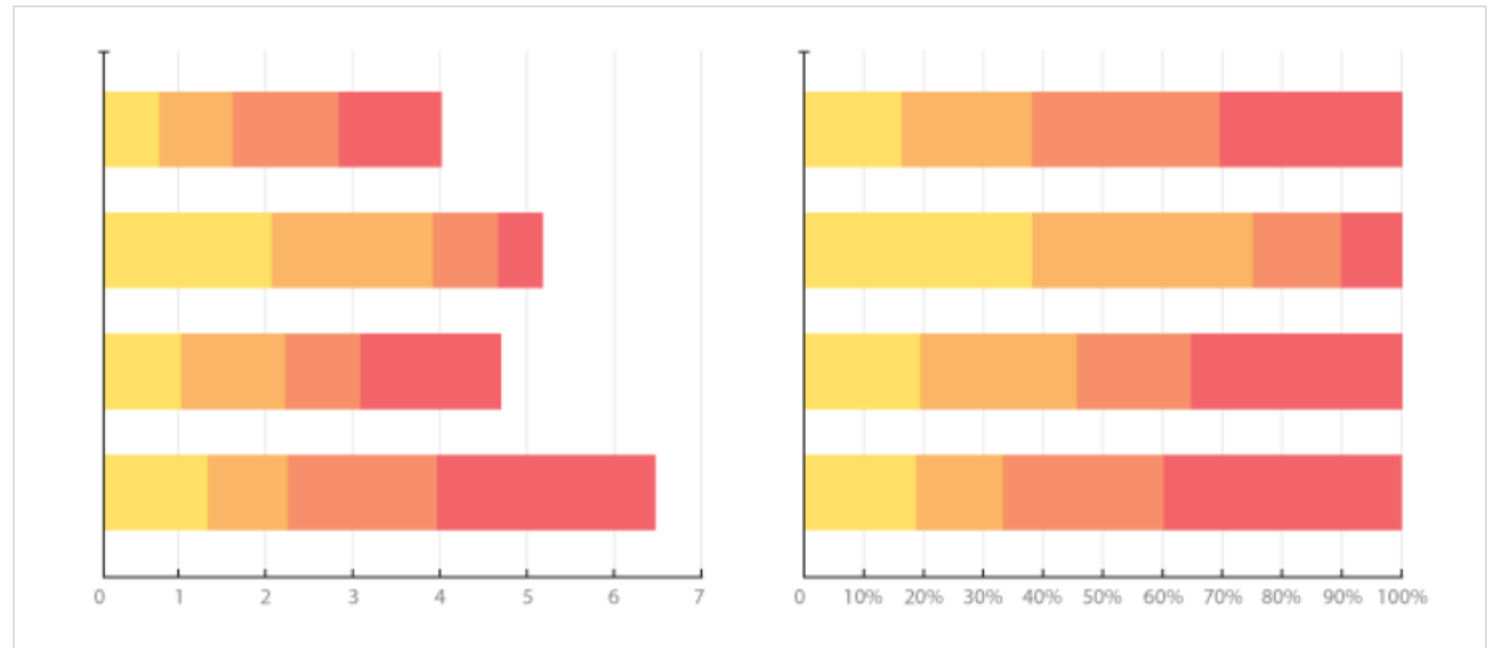
Hierarchy + Whole Part Relationship: Pie Chart

- **Pie chart:** menunjukkan proporsi/persentase dari kategori-kategori dalam suatu variable



Hierarchy + Whole Part Relationship: Stacked Bar Chart

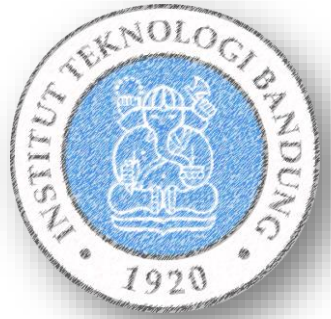
- **Stacked bar chart:** menampilkan bagian-bagian dari total nilai untuk suatu kategori dalam satu bar/batang



Hierarchy + Whole Part Relationship: TreeMap

- **Treemap:** menunjukkan struktur hierarki dari nilai kuantitatif melalui ukuran area



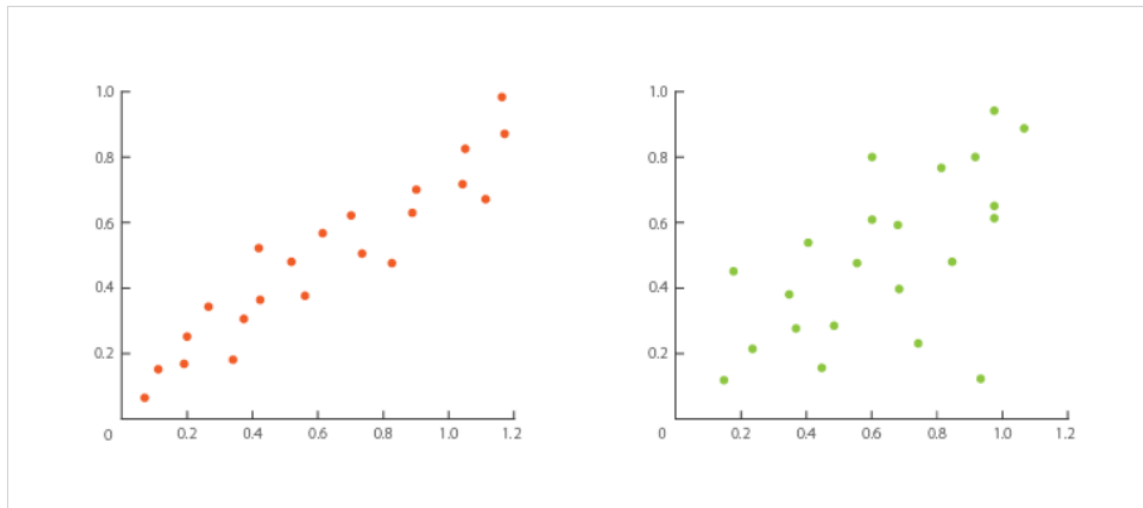


Plotting relationships

- Menunjukkan hubungan-hubungan antar data atau korelasi antara 2 variable atau lebih
- Grafik yang dapat digunakan:
 - Scatter plot
 - Bubble plot
 - Heatmap
 - Dll.

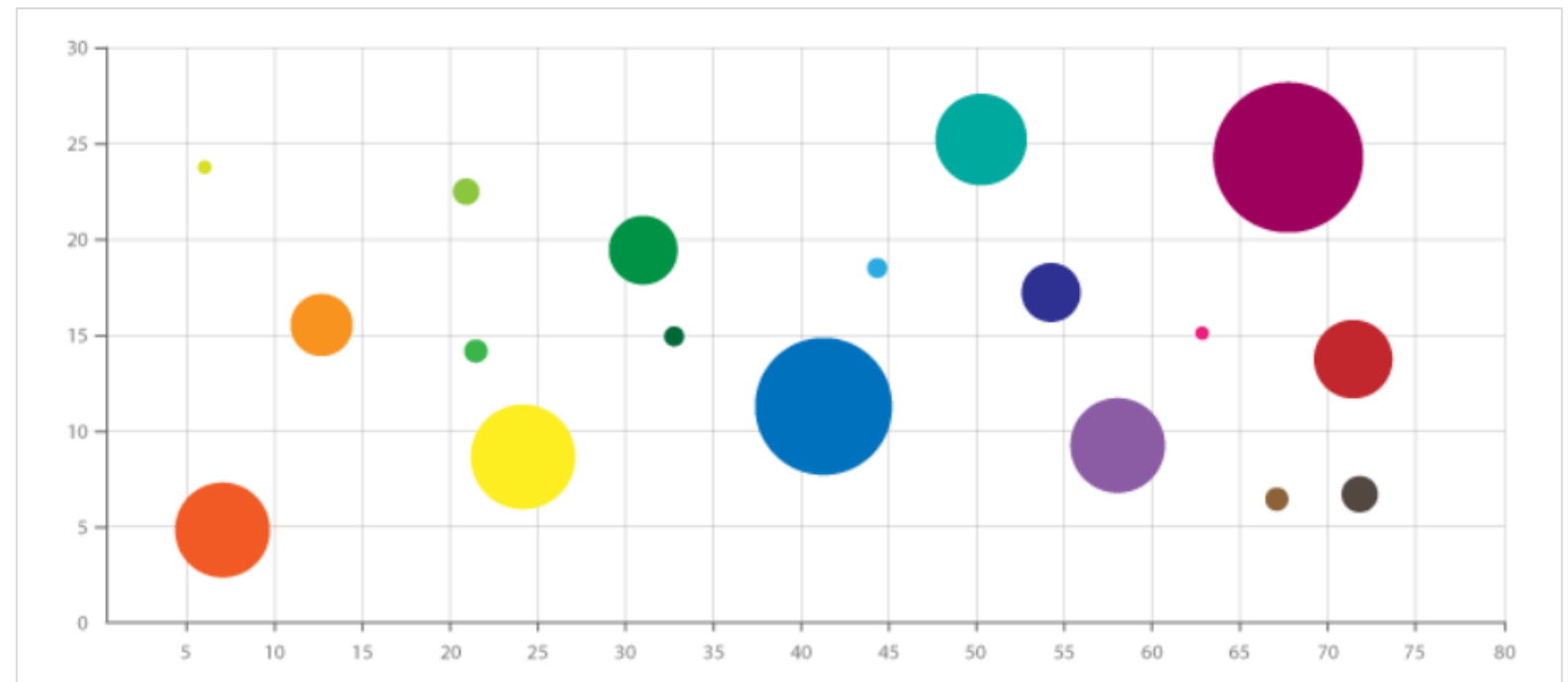
Plotting Relationship: Scatter Plot

- **Scatter plot** terdiri atas titik-titik nilai yang dipetakan di atas koordinat x dan y yang merepresentasikan nilai dari 2 variable
- Dapat digunakan untuk menunjukkan korelasi antara kedua variabel



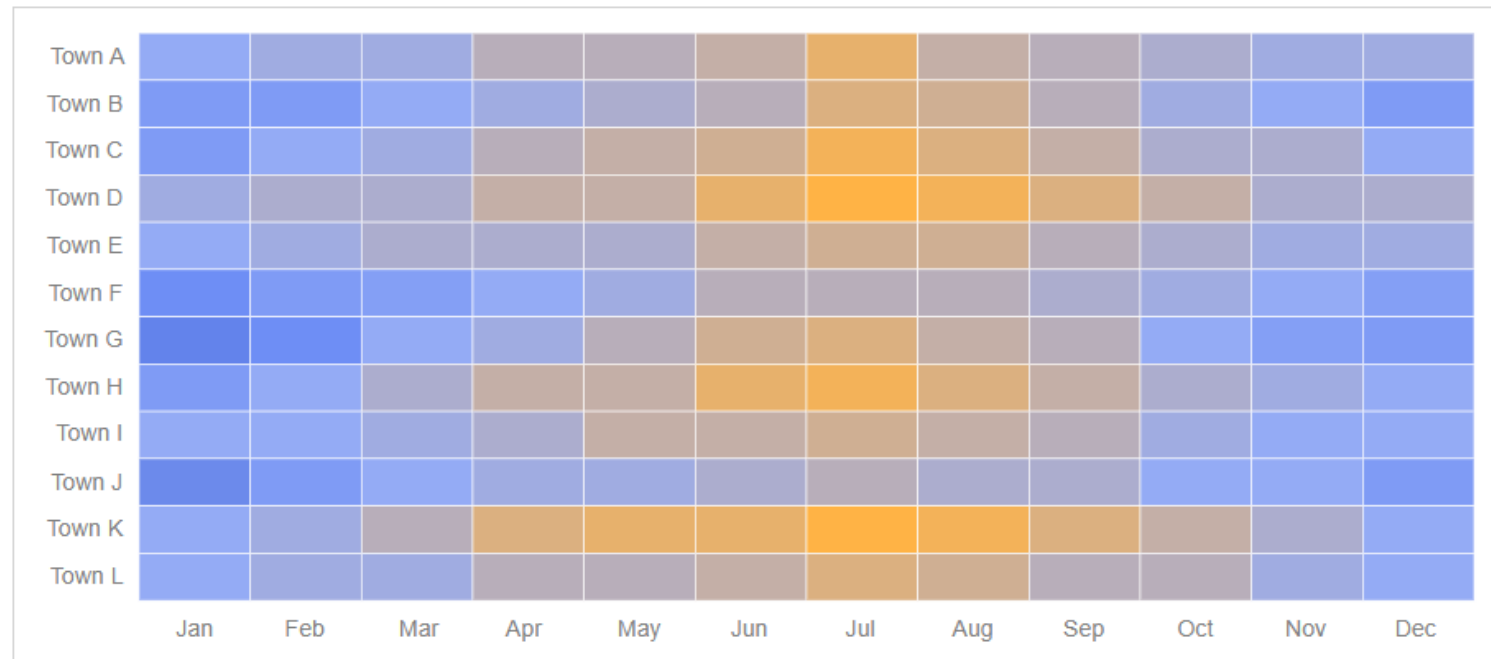
Plotting Relationship: Bubble plot

- **Bubble plot** adalah grafik multi-variable yang memplot nilai-nilai kuantitatif dalam bentuk lingkaran yang berbeda luasnya.



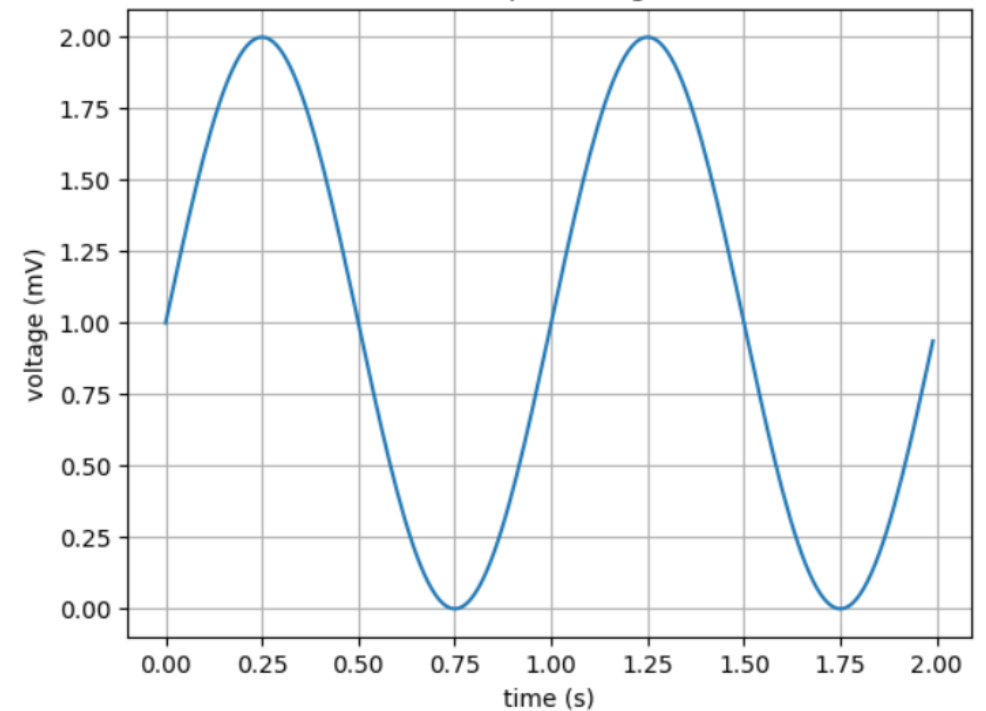
Plotting Relationship: Heatmap

- **Heatmap** digunakan untuk memvisualisasikan data melalui berbagai variasi warna. Dalam bentuk tabular, heatmap dapat digunakan untuk memetakan data dari 2 variable atau lebih.



Matplotlib

- **Matplotlib** adalah library di Python yang digunakan untuk plotting grafik 2 dimensi dalam berbagai format
 - **pyplot** adalah modul untuk plotting sederhana dengan interface yang mirip seperti MATLAB
- Dikembangkan oleh John Hunter (1968-2012) dan rekan-rekan, merupakan salah satu library yang banyak dimanfaatkan untuk visualisasi data saintifik



Contoh data-1

- data.csv

- Load pandas dan dataframe dari file data.csv

```
import pandas as pd  
df = pd.read_csv("D:/data.csv")
```

- Load library matplotlib.pyplot

```
import matplotlib.pyplot as plt
```

	name	age	gender	state	num_children	num_pets
0	john	23	M	CA	2	5
1	marry	78	F	DC	0	1
2	peter	22	M	CA	0	0
3	jeff	19	M	DC	3	5
4	bill	45	M	CA	2	2
5	lisa	33	F	TX	1	2
6	jose	20	M	TX	4	3

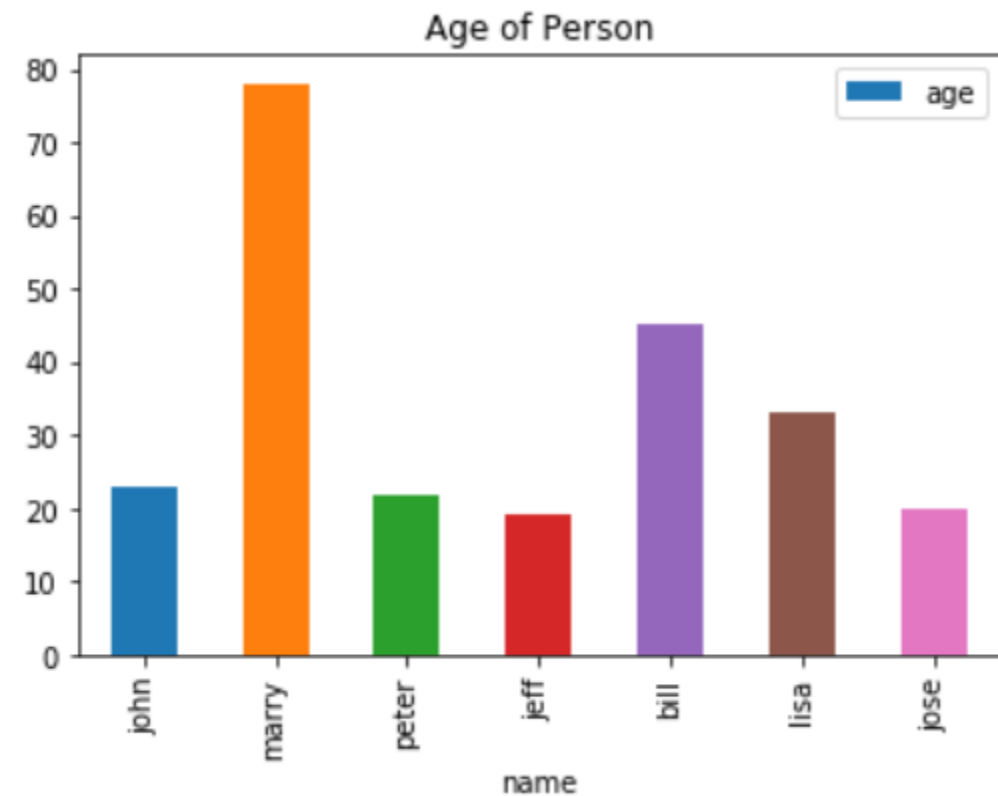
Lebih lanjut fungsi plot pada dataframe:
<https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.plot.html>

Bar chart (1)

- Buat vertical bar chart untuk menampilkan umur dari setiap orang

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

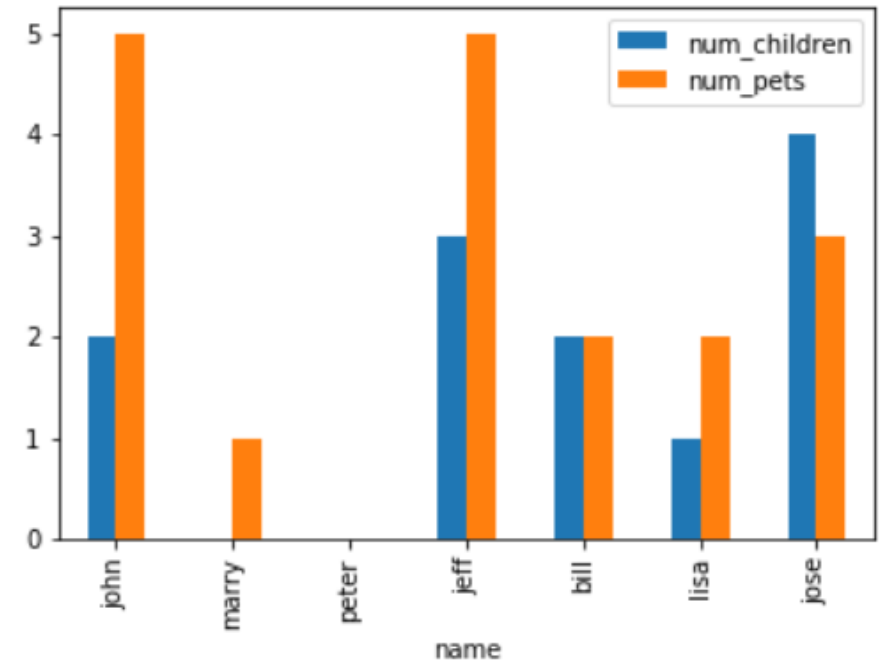
```
df = pd.read_csv("D:/data.csv")
df.plot(kind="bar", x="name", y="age", title="Age of Person")
plt.show()
```

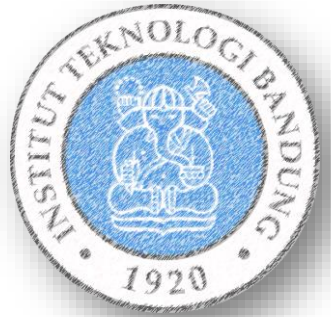


Bar Chart (2)

- Tampilkan banyaknya anak (num_children) dan banyaknya piaraan (num_pets) dalam 1 grafik vertical bar chart

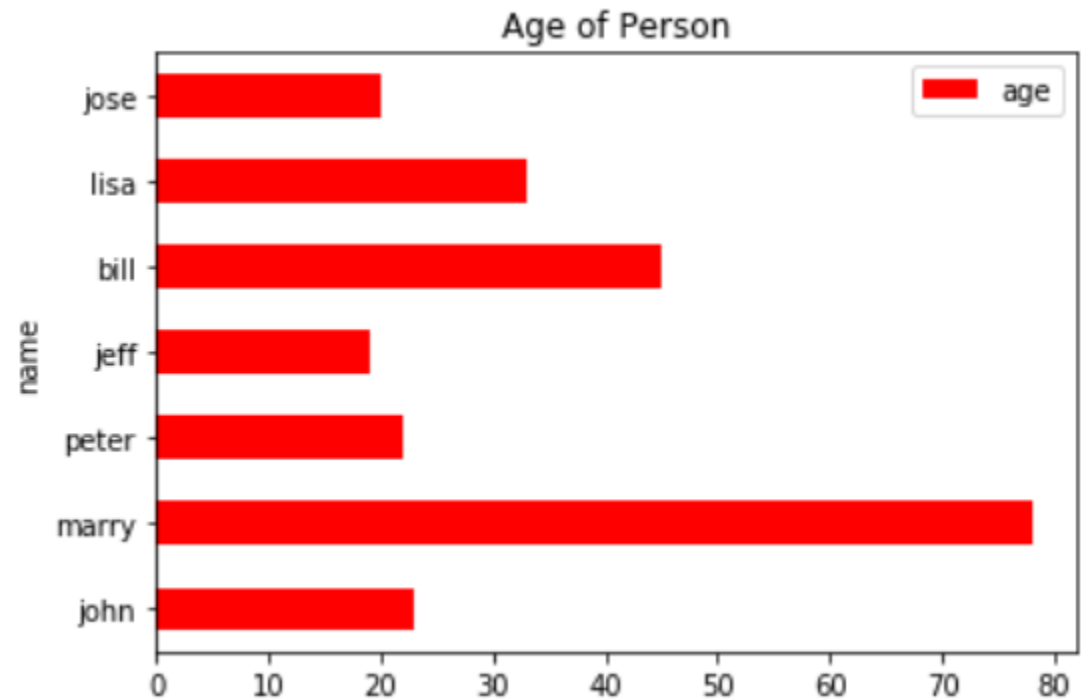
```
df.plot(kind="bar", x="name", y=["num_children", "num_pets"])  
plt.show()
```





Horizontal Bar Chart

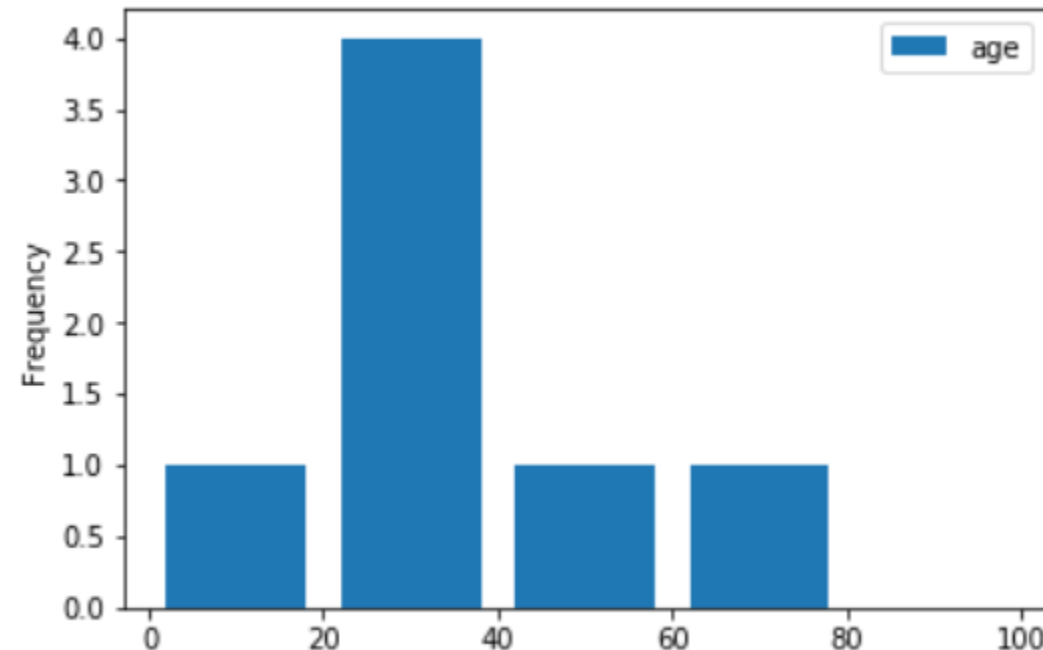
```
df.plot(kind="barh", x="name", y="age", title="Age of Person",  
color="red")  
plt.show()
```



Histogram

- Tampilkan distribusi orang berdasarkan kelompok umur: 0-20; 21-40; 41-60; 61-80; 81-100

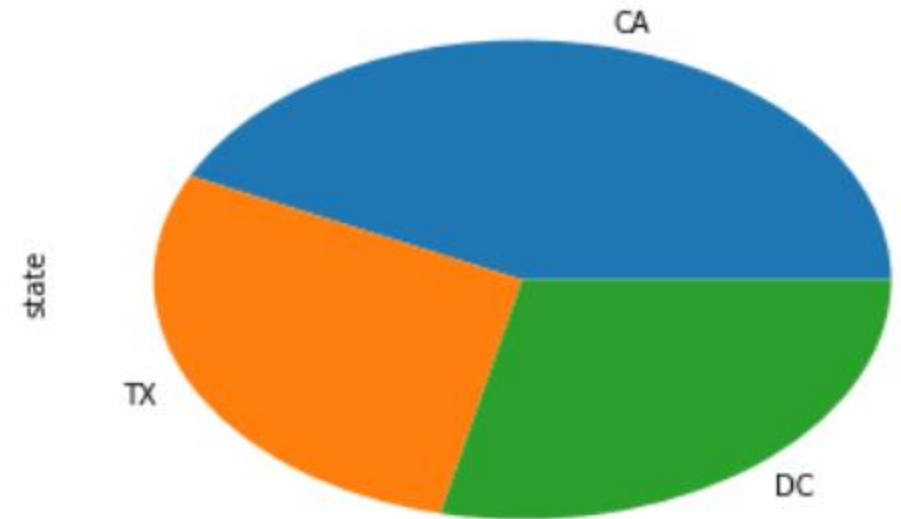
```
df[["age"]].plot(kind="hist",bins=[0,20,40,60,80,100],rwidth=0.8)  
plt.show()
```

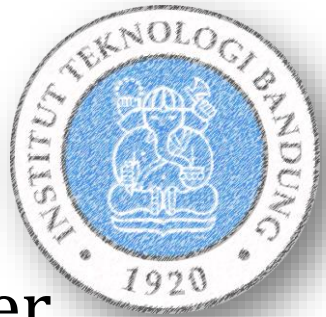


Pie Chart

- Tampilkan komposisi banyaknya orang berdasarkan negara
- Langkah:
 - Hitung distribusi frekuensi (lihat bahan kuliah sebelumnya)
 - Plot ke pie chart

```
df2 = df["state"].value_counts()  
df2.plot(kind = "pie")  
plt.show()
```





Stacked Bar Chart (1)

- Tampilkan data banyaknya data per jenis kelamin (gender) per negara bagian (state)
- Langkah:
 1. Membuat **tabel pivot**: menggunakan perintah group by, kelompokkan data terlebih dahulu
Misalnya: group by berdasarkan kolom gender, state, lalu count banyaknya data (misal berdasarkan kolom name)

```
df3 = df.groupby(["gender", "state"])["name"].size().unstack()
```

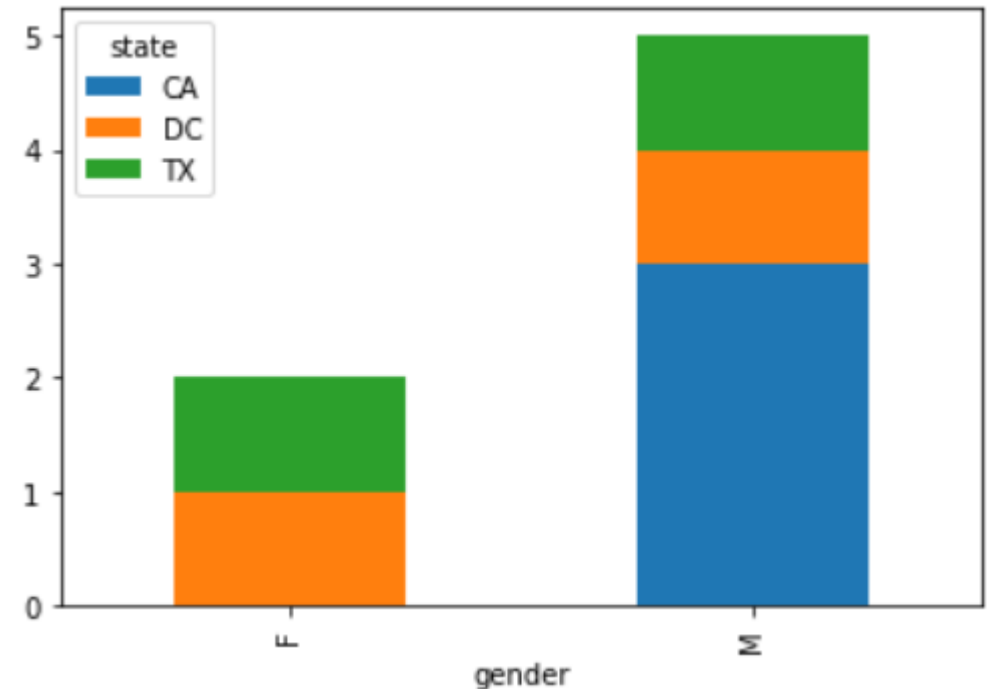
```
df3
```

state	CA	DC	TX
gender			
F	NaN	1.0	1.0
M	3.0	1.0	1.0

Stacked Bar Chart (2)

- Langkah:
2. Plot df3 ke bar chart dan stacked = True

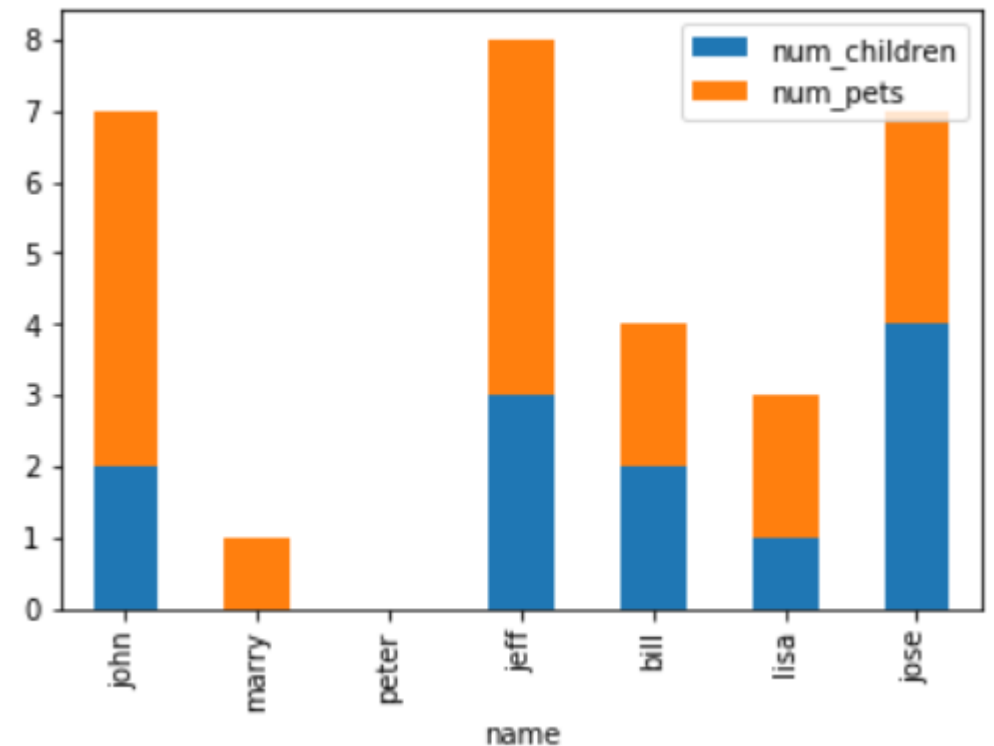
```
df3.plot(kind="bar", stacked=True)  
plt.show()
```

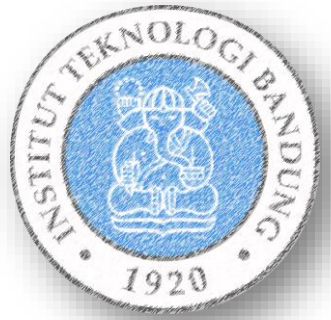


Stacked Bar Chart (3)

- Tampilkan banyaknya anak (num_children) dan banyaknya piaraan (num_pets) dalam 1 grafik stacked bar chart

```
df.plot(kind = "bar", x = "name",  
y=["num_children", "num_pets"],  
stacked = True)  
plt.show()
```





Contoh Data-2: Time-series

- animal.csv
- Load data:

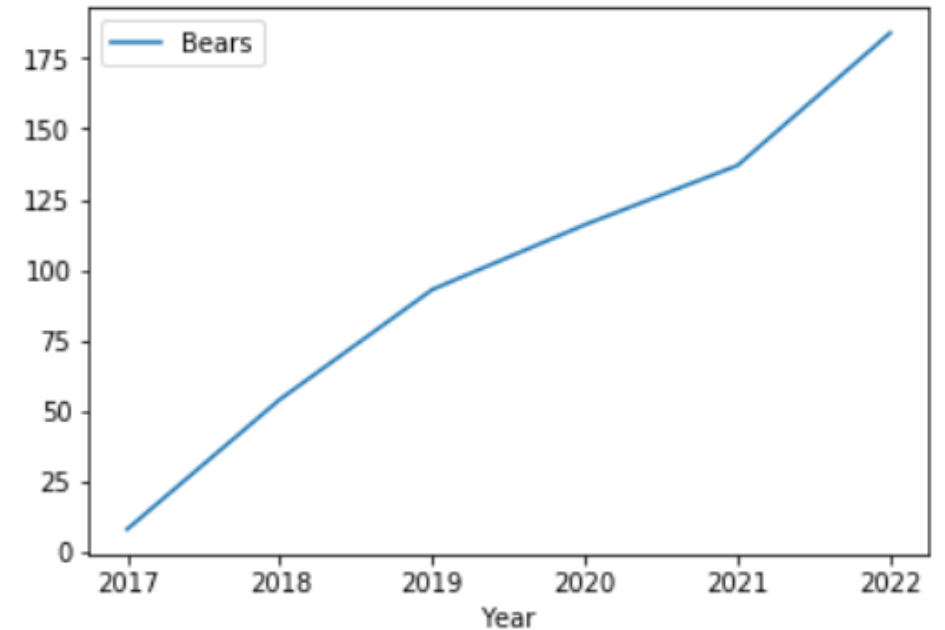
```
df1 = pd.read_csv("D:/animal.csv")
```

	Year	Bears	Dolphins	Whales
0	2017	8	150	80
1	2018	54	77	54
2	2019	93	32	100
3	2020	116	11	76
4	2021	137	6	93
5	2022	184	1	72

Line Chart (1)

- Tampilkan pertumbuhan populasi beruang (Bears) dari tahun ke tahun dalam line chart

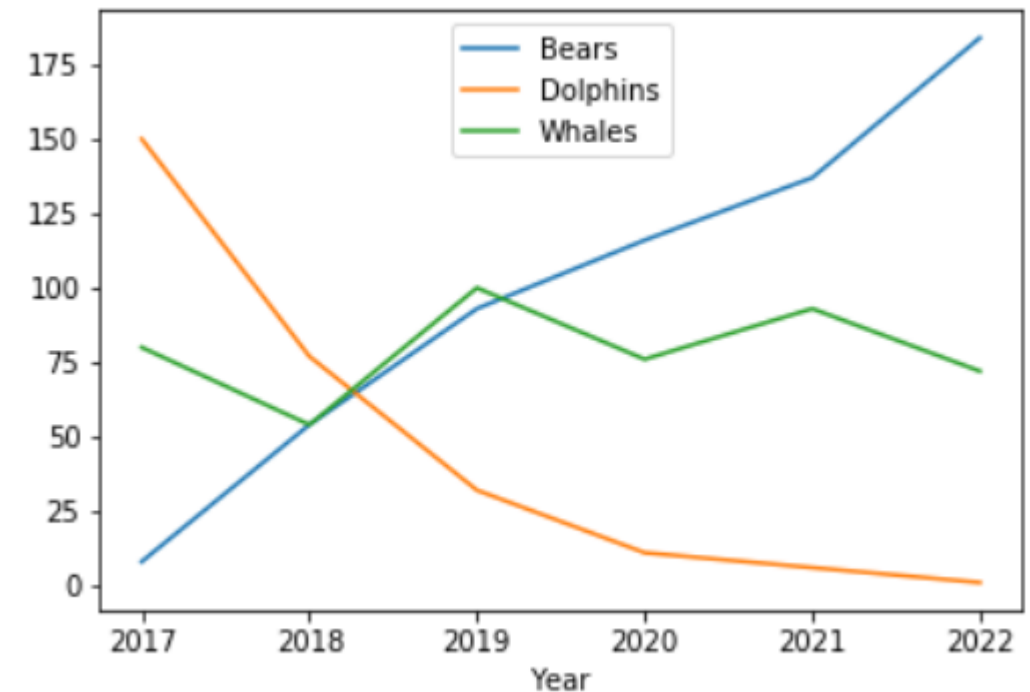
```
df1.plot(kind="line", x="Year", y="Bears")  
plt.show()
```



Line Chart (2)

- Tampilkan pertumbuhan populasi beruang (Bears), lumba-lumba (Dolphins), dan ikan paus (Whales) dari tahun ke tahun dalam 1 line chart

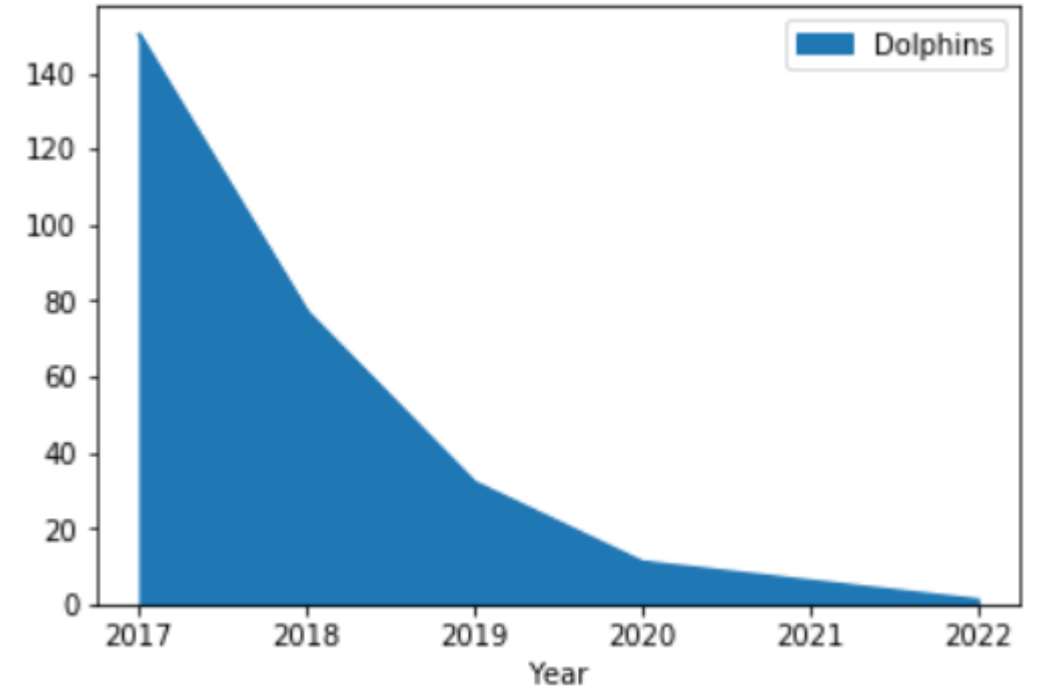
```
df1.plot(kind="line", x="Year",  
y=["Bears", "Dolphins", "Whales"])  
plt.show()
```



Area Chart

- Tampilkan pertumbuhan populasi lumba-lumba (Dolphins) dari tahun ke tahun dalam area chart

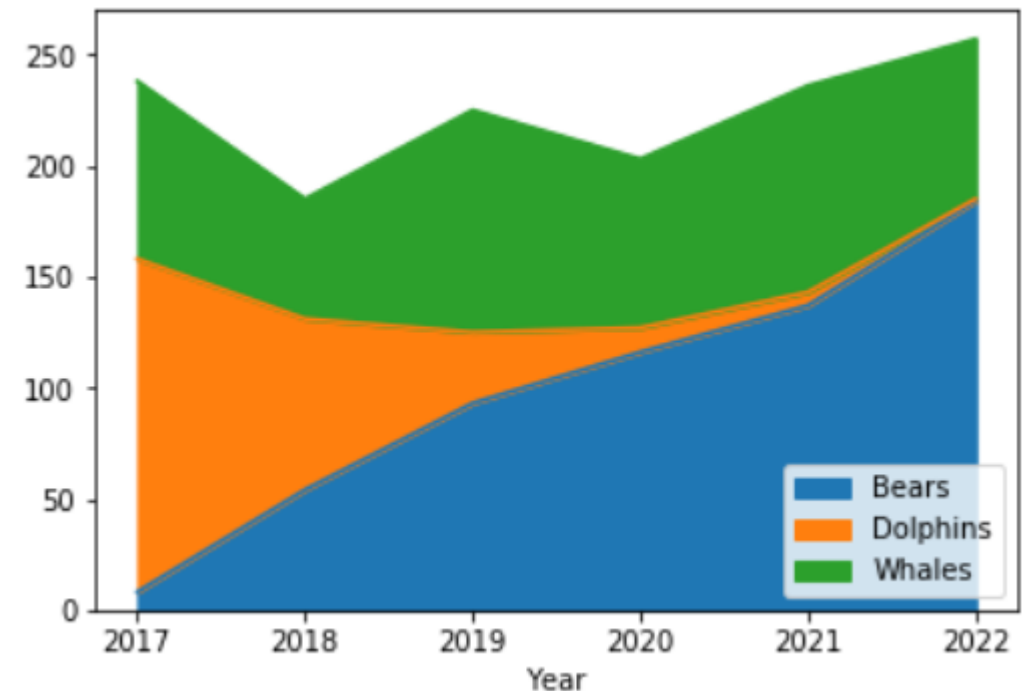
```
df1.plot(kind="area", x="Year",  
y="Dolphins")  
plt.show()
```

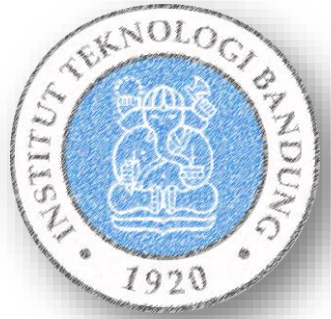


Stacked Area Chart

- Tampilkan pertumbuhan populasi beruang (Bears), lumba-lumba (Dolphins), dan ikan paus (Whales) dari tahun ke tahun dalam stacked area chart

```
df1.plot(kind="area", x="Year",  
y=["Bears", "Dolphins", "Whales"])  
plt.show()
```





Contoh Data-3: medali Asian Games

- Kembali ke contoh pada materi kuliah sebelumnya: file medali.csv
- Load data:

```
df4 = pd.read_csv("D:/medali.csv")
```

Scatter Plot

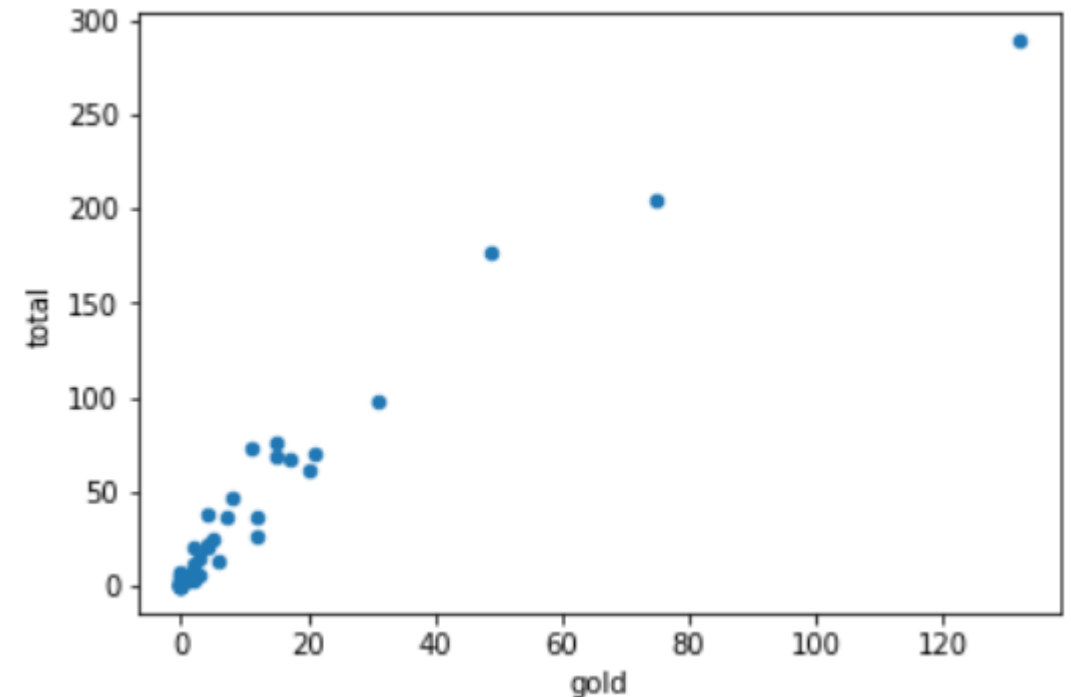
- Tampilkan relationship antara variable gold dan total dalam grafik scatter plot dan tunjukkan adanya korelasi positif

```
df4.plot(kind="scatter",  
x="gold", y="total")
```

Alternatif:

```
df4.plot.scatter(x="gold",  
y="total")
```

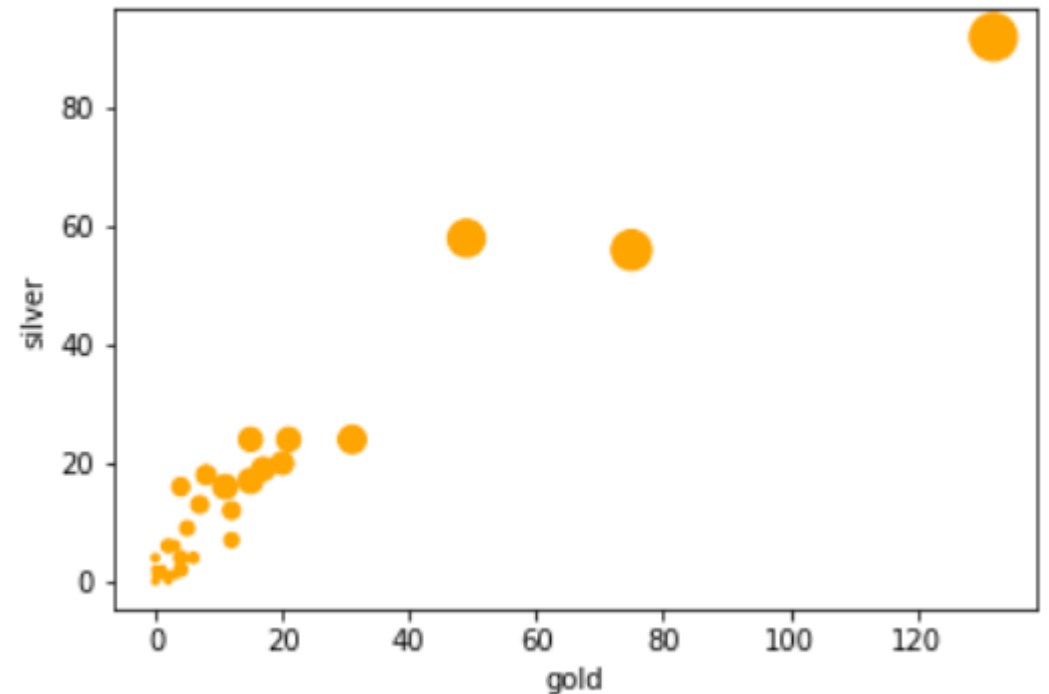
```
plt.show
```

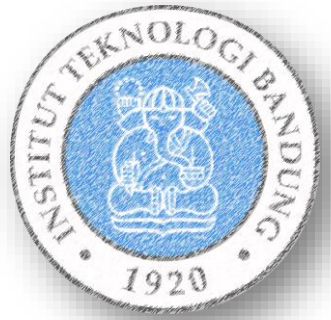


Bubble Plot

- Tampilkan banyaknya total medali dikaitkan dengan perolehan nilai medali emas (gold) pada sumbu x dan perolehan medali perak (silver) pada sumbu y dalam grafik bubble plot
- Bubble plot di Python dibuat berdasarkan scatter plot

```
df4.plot(kind="scatter", x="gold",  
y="silver", sizes=df4["total"],  
color="orange")  
plt.show()
```





Menyimpan grafik ke file

```
df.plot(kind="bar", x="name", y="age", title="Age of Person")  
plt.savefig("D:/agebarchart.png")
```