



Kelompok 8

Tugas Besar Algoritma Pemrograman





Kelompok 8

Perkenalan Anggota Kelompok

- Christian Arvianus Nathanael Biran : 121450112
- Kanaya Dea Thalita Akhmad : 121450001
- Sella Dianka Fitri : 121450000
- Ayu Erlinawati : 121450025
- Ibnu Farhan Al-Ghfari : 121450121.



Latar Belakang

Data seiring dengan perkembangan ilmu sains data, pengolahan dataset dapat dilakukan dengan lebih cepat dan juga efisien. Pengolahan data dapat dilakukan dengan berbagai macam bahasa pemrograman seperti Python dan juga Studio R, selain itu kita juga dapat memanfaatkan banyak library yang ada untuk lebih memaksimalkan pengolahan data yang kita lakukan. Dengan latar belakang ini maka kami melakukan praktik pengolahan data yang mencakup data explore, data cleaning, data sorting dan lain-lain, dengan menggunakan bantuan dari library Pandas, Numpy, Seaborn dan juga Matplotlib..

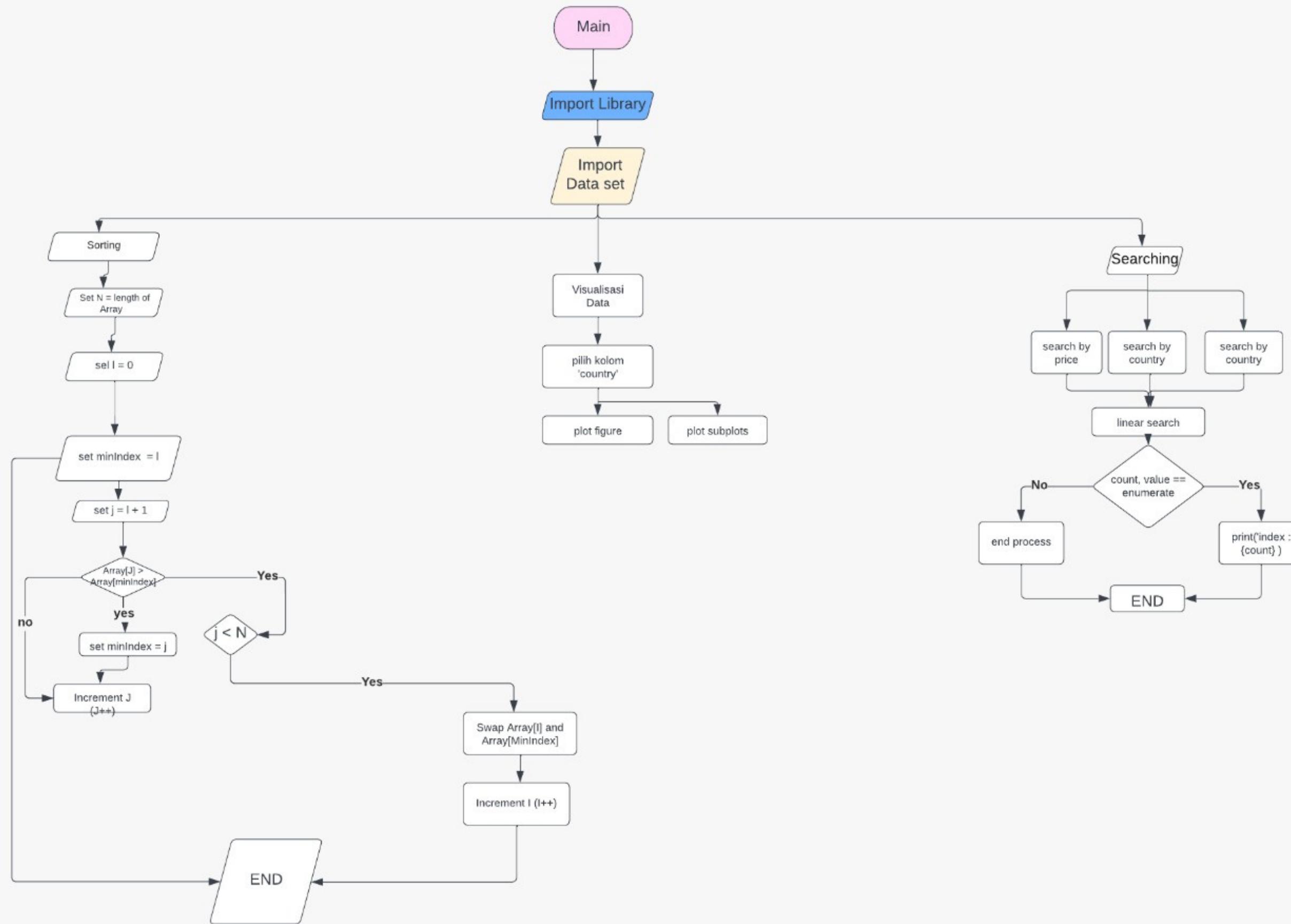
Deskripsi Data

winemag-data-130k-v2

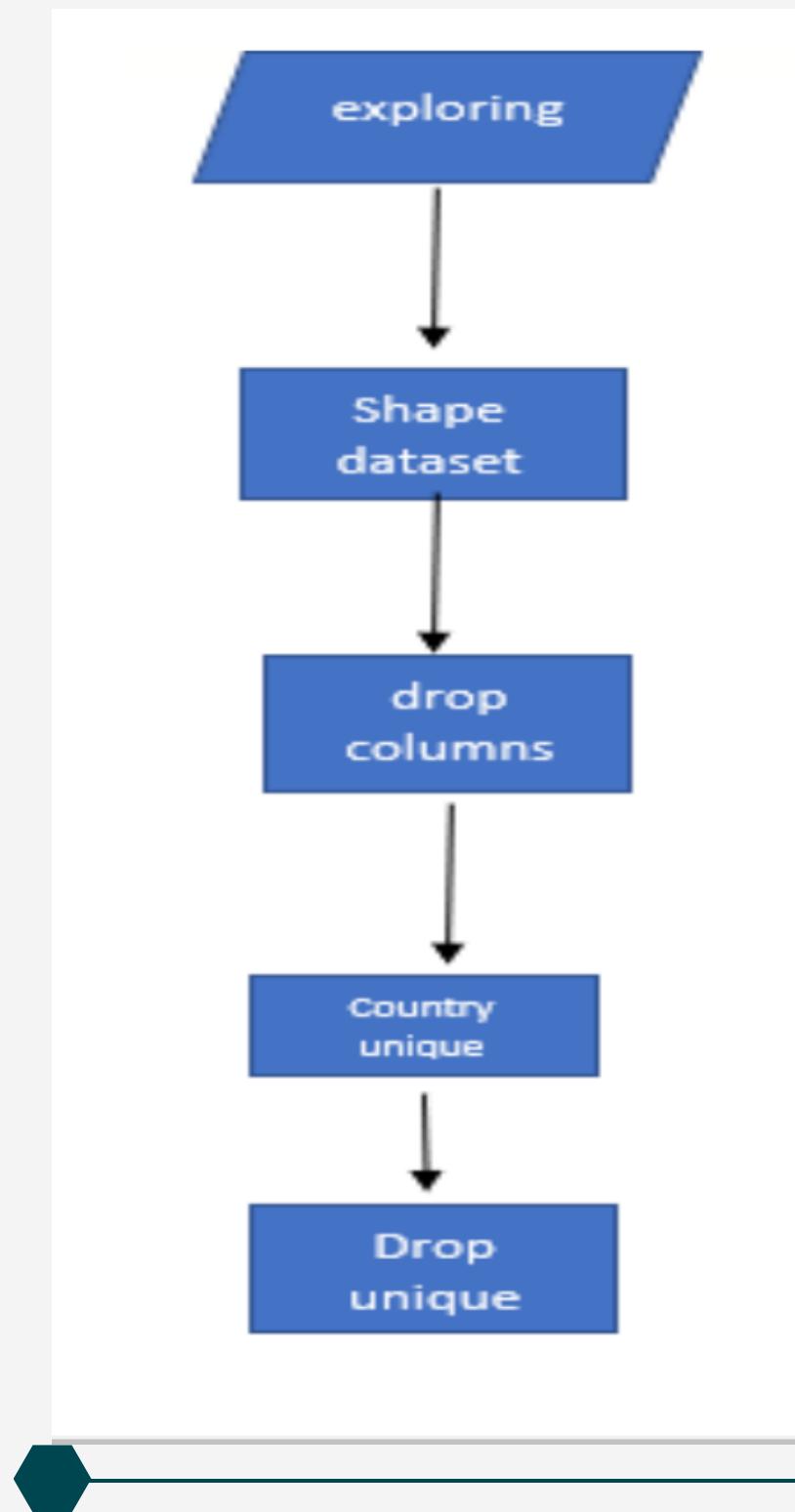
Dataset yang kami gunakan merupakan kumpulan data-data yang berbentuk file csv, dataset ini terdiri atas tipe data string, float dan juga integer yang terdiri dari 129971 baris dan juga 13 kolom. Data-data ini terdiri dari kolom-kolom country, description, designation, points, price, province, region_1, region_2, taster_name, taster_twitter_handle, title, variety, winery.



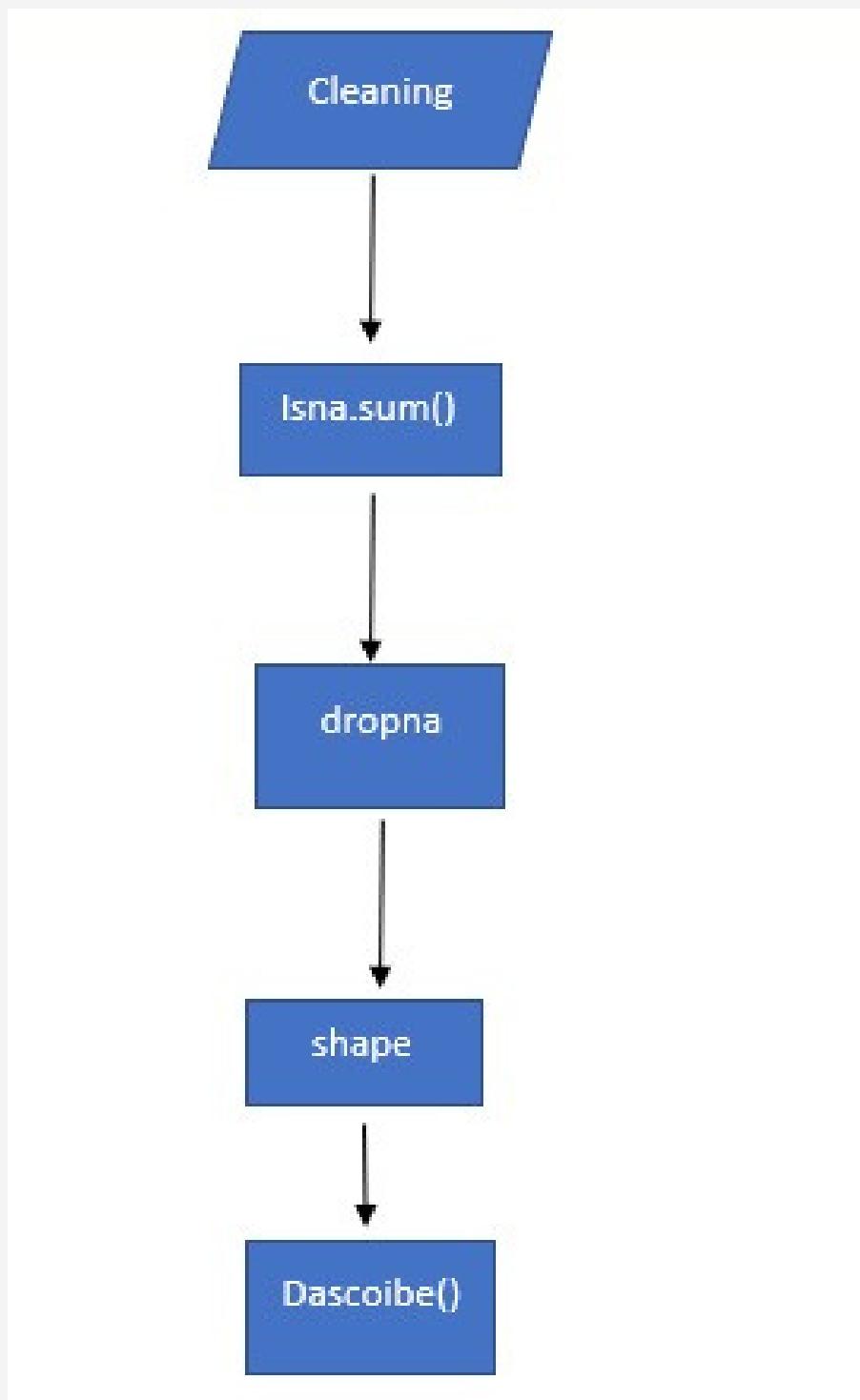
Flowchart Sistem



Eksploring



cleaning



Dependency / Library yang digunakan

```
[ ] import pandas as pd # data processing  
import numpy as np # linear algebra  
import seaborn as sns # visualisasi data  
import matplotlib.pyplot as plt
```

Pandas

Library pandas ini digunakan sebagai pengolah data terlebih di dalam data frame.

Numpy

Numpy berfungsi sebagai pengolah data yang bersifat aljabar linear, juga untuk mengembalikan dari fungsi array.

Matplotlib

Matplotlib digunakan sebagai library yang akan menampilkan grafik plot dalam program ini.



Daftar Fungsionalitas

Read CSV

```
[ ] wine_df = pd.read_csv('winemag-data-130k-v2.csv.zip', index_col = 0)
#wine_df.head()
```



Eksploring Dataset

Dataset yang digunakan dalam penelitian ini adalah dataset yang diambil dari platform kaggle dengan judul ‘winemag data’

- Data Frame Shape

Menampilkan ukuran dari data set, panjang kolom dan baris akan diukur menggunakan data frame.shape.

Dalam program ini ukuran dari dataset ‘winemag’ adalah 129971 baris dan 13 kolom

```
[ ] wine_df.shape  
(129971, 13)
```

- Data Frame Columns

Menampilkan index kolom, nama nama dari index kolom dataset akan dikembalikan dengan fungsi ini.

```
[ ] wine_df.columns  
Index(['country', 'description', 'designation', 'points', 'price', 'province',  
       'region_1', 'region_2', 'taster_name', 'taster_twitter_handle', 'title',  
       'variety', 'winery'],  
      dtype='object')
```

Eksploring Dataset

- Unique ()

Digunakan untuk mencari data yang unique, atau bisa disebut data yang sama. Data yang sama akan memberikan hasil analisis yang tidak murni. Oleh karena itu penggunaan unique () untuk menyeleksi data ~~yang duplikat~~

```
[ ] wine_df.country.unique()

array(['Italy', 'Portugal', 'US', 'Spain', 'France', 'Germany',
       'Argentina', 'Chile', 'Australia', 'Austria', 'South Africa',
       'New Zealand', 'Israel', 'Hungary', 'Greece', 'Romania', 'Mexico',
       'Canada', nan, 'Turkey', 'Czech Republic', 'Slovenia',
       'Luxembourg', 'Croatia', 'Georgia', 'Uruguay', 'England',
       'Lebanon', 'Serbia', 'Brazil', 'Moldova', 'Morocco', 'Peru',
       'India', 'Bulgaria', 'Cyprus', 'Armenia', 'Switzerland',
       'Bosnia and Herzegovina', 'Ukraine', 'Slovakia', 'Macedonia',
       'China', 'Egypt'], dtype=object)
```

- Drop Duplicates

Untuk menghapus data yang duplikat setelah diseleksi sebelumnya pada kolom 'country'.

```
[ ] wine_df.dropna(inplace = True)
```

Cleaning Dataset

Digunakan untuk membersihkan beberapa bagian dari kolom ataupun baris yang tidak terpakai dalam program ini. Sehingga analisis data dapat terfokus pada beberapa kolom saja dalam dataset.

- Isna

Mengecek apakah ada nilai NaN pada dataframe. Function ini akan mengembalikan nilai Boolean, True atau False, untuk tiap elemen di dataframe

```
[ ] wine_df.isna().sum()
```

```
country      45
points        0
price       192
dtype: int64
```

- describe()

digunakan untuk menampilkan deskriptif statistik data. Hanya kolom yang bertipe numerik yang akan ditampilkan statistiknya.

```
[ ] wine_df['price'].describe()
```

```
count    10392.000000
mean     69.246536
std      106.380201
min      4.000000
25%     22.000000
50%     41.000000
75%     78.000000
max     3300.000000
```

[Ke Halaman Agenda](#)

Searching

- Algoritma pencarian untuk data yang terurut. Pencarian dilakukan dengan cara menebak apakah data yang dicari berada ditengah-tengah data, kemudian membandingkan data yang dicari dengan data yang ada ditengah. Bila data yang ditengah sama dengan data yang dicari, berarti data ditemukan.

```
[ ] def search_by_country(name):
    result = wine_df.loc[wine_df['country'] == name]
    return result.head()

def search_by_price(num):
    result = wine_df['price'] == int(num)
    return wine_df[result].head()

def search_by_points(num):
    result = wine_df['points'] == int(num)
    return wine_df[result].head()
```

```
[ ] def LinearSearch(lys, element):
    values = np.array(lys)
    for count, value in enumerate(values):
        if value == element:
            return f'index : {count}'
```

Sorting

- Linear Search

teknik mencari data dengan cara memplot data satu per satu. Jika kecocokan ditemukan, program akan mengembalikan hasilnya, jika tidak, pencarian akan dilanjutkan hingga akhir array

```
[ ] reduced = np.array(wine_df['price'][:20])
reduced

array([15., 14., 13., 65., 15., 16., 24., 12., 27., 19., 30., 34., 12.,
       24., 30., 13., 28., 32., 23., 20.])
```

```
[ ] def sorting_values_asc(col):
    result = wine_df.sort_values(col, ascending = True)
    return result.head()

def sorting_values_des(col):
    result = wine_df.sort_values(col, ascending = False)
    return result.head()
```

```
[ ] arr = np.array(wine_df['price'])

[ ] def selectionSort(array, size):
    #Traverse through 1 to len(array)
    for ind in range(size):
        min_index = ind
        for j in range(ind + 1, size):
            if array[j] < array[min_index]:
                min_index = j
        # swapping the elements to sort the array
        (array[ind], array[min_index]) = (array[min_index], array[ind])
```

```
size = len(arr)
selectionSort(arr, size)
print('The array after sorting in Ascending Order by selection sort is:')
print(arr)
```

```
The array after sorting in Ascending Order by selection sort is:
[ 4.  4.  4. ... 2013. 2500. 3300.]
```

Inserting dan Deleting

- Inserting dibuat untuk menambahkan kolom dari dataset
- Deleting untuk menghapus kolom dari data set

```
[ ] def inserting(country, points, price):
    wine_df = wine_df.append({'country' : country, 'points' : int(points), 'price' : int(price)})
    wine_df.loc[len(wine_df)] = [country, points, price]
    return wine_df.tail()
```

```
[ ] inserting('indo', 90, 980)
```

	country	points	price
129818	France	89	115.0
129886	Argentina	91	88.0
129931	France	91	107.0
129948	Argentina	90	43.0
10392	indo	90	980.0

```
[ ] def deleting_row(i):
    wine_df.drop(wine_df.index[int(i)], inplace = True)
    return wine_df.tail()
```

```
[ ] deleting_row(10392)
```

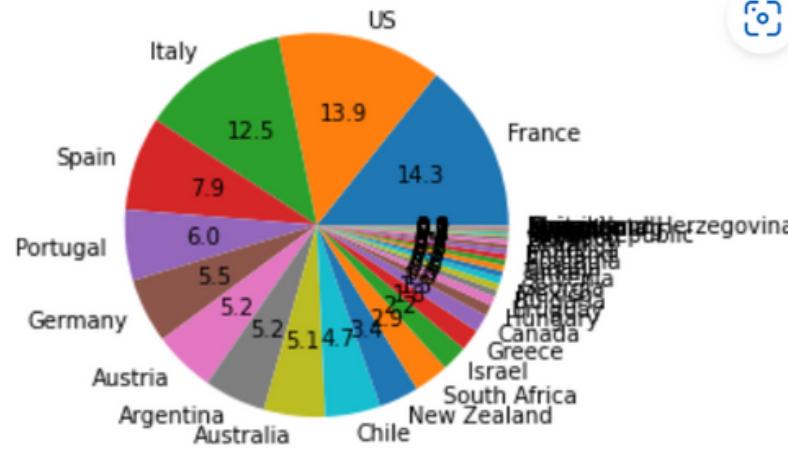
	country	points	price
129777	Argentina	89	70.0
129818	France	89	115.0
129886	Argentina	91	88.0
129931	France	91	107.0
129948	Argentina	90	43.0

```
[ ] wine_df.shape
```

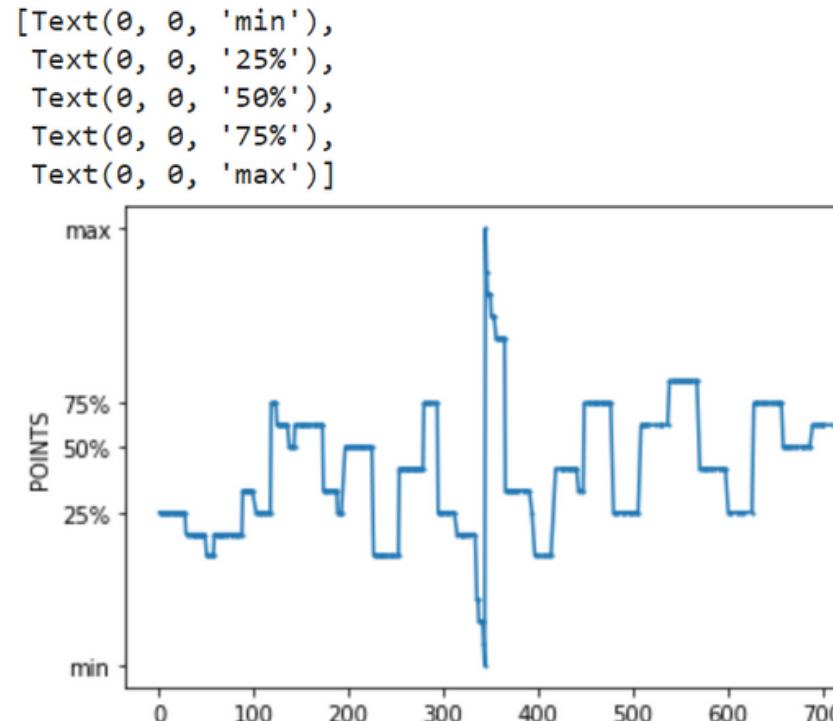
```
(10392, 3)
```

Visualization

```
[ ] fig = plt.figure()  
#plt.pie?  
plt.pie(wine_df['country'].value_counts(), labels = wine_df['country'].value_counts().keys(), autopct = '%.1f')  
plt.pie?
```

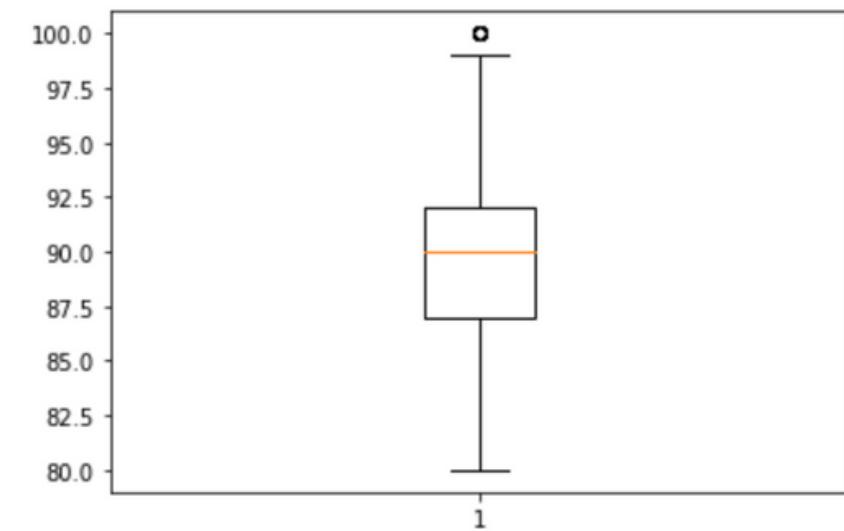


```
[ ] fig, ax = plt.subplots()  
ax.plot(wine_df['points'][:500], marker = 'o', markersize = 1)  
ax.set_ylabel('POINTS')  
ax.set_xlabel('INDEX')  
ax.set_yticks([80, 87, 90, 92, 100])  
ax.set_yticklabels(['min', '25%', '50%', '75%', 'max'])
```



```
[ ] fig = plt.figure()  
plt.boxplot(wine_df['points'])
```

```
{'whiskers': [<matplotlib.lines.Line2D at 0x7fdc178c3280>,  
               <matplotlib.lines.Line2D at 0x7fdc178c35e0>],  
 'caps': [<matplotlib.lines.Line2D at 0x7fdc178c3940>,  
          <matplotlib.lines.Line2D at 0x7fdc178c3ca0>],  
 'boxes': [<matplotlib.lines.Line2D at 0x7fdc178cdee0>],  
 'medians': [<matplotlib.lines.Line2D at 0x7fdc178bb040>],  
 'fliers': [<matplotlib.lines.Line2D at 0x7fdc178bb340>],  
 'means': []}
```



Kesimpulan

Algoritma dan pemrograman adalah urutan atau langkah-langkah yang dilakukan untuk menyelesaikan suatu permasalahan pada program. Pada tugas besar ini kami membuat program dimana kami mengambil dataset public dari Kaggle yaitu [winemag-data-130k-v2](#) dataset yang berbentuk csv. Dalam program tersebut kami menggunakan empat library yang sudah dipelajari yaitu Numpy, Pandas, Seaborn, dan Matplotlib. Library berfungsi untuk membuat pemrograman python menjadi lebih sederhana dan nyaman bagi programmer karena tidak perlu menulis kode yang sama berulang kali untuk program yang berbeda.

Pada visualisasi data yang kami lakukan pandas digunakan untuk membaca dan menampilkan data, Numpy untuk mengolah data yang bersifat aljabar linear, Seaborn sebagai penampil grafik dari dataset dan matplotlib untuk membuat grafik plot yang mempresentasikan kolom yang ada di dataset tersebut.



Terimakasih
