NO 1

Menulis deskripsi statistika (Descriptive Statistics) dari semua kolom pada data yang bersifat numerik, terdiri dari mean, median, modus, standar deviasi, variansi, range, nilai minimum, maksimum, kuartil, IQR, skewness dan kurtosis.

```
import pandas as pd
import numpy as np

col_names = ['id', 'pH', 'Hardness', 'Solids', 'Chloramines',
'Sulfate', 'Conductivity', 'OrganicCarbon', 'Trihalomethanes',
'Turbidity', 'Potability']
df = pd.read_csv('../data/water_potability.csv', names=col_names)
df = df.drop(["id", "Potability"], axis=1)

df.head()
```

```
          pH    Hardness        Solids  Chloramines     Sulfate
Conductivity  \
0   8.316766  214.373394  22018.417441     8.059332  356.886136
363.266516
1   9.092223  181.101509  17978.986339     6.546600  310.135738
398.410813
2   5.584087  188.313324  28748.687739     7.544869  326.678363
280.467916
3  10.223862  248.071735  28749.716544     7.513408  393.663396
283.651634
4   8.635849  203.361523  13672.091764     4.563009  303.309771
474.607645

   OrganicCarbon  Trihalomethanes  Turbidity
0      18.436524       100.341674   4.628771
1      11.558279        31.997993   4.075075
2       8.399735        54.917862   2.559708
3      13.789695        84.603556   2.672989
4      12.363817        62.798309   4.401425
```

Menampilkan banyaknya data, rata - rata, standar deviasi, nilai minimal, nilai ketiga quartil serta nilai maksimal dari setiap kolom

```
df.describe()
```

```
                pH     Hardness        Solids  Chloramines
Sulfate  \
count  2010.000000  2010.000000   2010.000000  2010.000000
2010.000000
mean      7.087193   195.969209  21904.673439     7.134322
333.211376
std       1.572803    32.643166   8625.397911     1.585214
41.211111
min       0.227499    73.492234    320.942611     1.390871
```

```
129.000000
25%       6.090785   176.740657  15614.412962     6.138326
307.626986
50%       7.029490   197.203525  20926.882155     7.142014
332.214113
75%       8.053006   216.447589  27170.534649     8.109933
359.268147
max      14.000000   317.338124  56488.672413    13.127000
481.030642

       Conductivity  OrganicCarbon  Trihalomethanes    Turbidity
count   2010.000000    2010.000000      2010.000000  2010.000000
mean     426.476708      14.357940        66.400717     3.969497
std       80.701872       3.325770        16.081109     0.780471
min      201.619737       2.200000         8.577013     1.450000
25%      366.619219      12.122530        55.949993     3.442882
50%      423.438372      14.323286        66.482041     3.967374
75%      482.209772      16.683562        77.294613     4.514663
max      753.342620      27.006707       124.000000     6.494749
```

Menampilkan statistik dari masing-masing kolom

```
from scipy.stats import iqr

def descriptive_statistics(df):
    print("Mean: \t\t\t", df.mean())
    print("Median: \t\t", df.median())
    if (len(df.mode()) == len(df)):
        print("Semua data unik sehingga modus yang dipilih adalah
nilai yang terkecil dari data dengan nilai " + str(df.mode()[0]) )
    else:
        print("Modus:\t\t", df.mode())
    print("Standar Deviasi:\t", df.std())
    print("Variansi:\t\t", df.var())
    print("Range:\t\t\t", df.max()-df.min())
    print("Nilai Minimum:\t\t", df.min())
    print("Nilai Maksimum:\t\t", df.max())
    print("Kuartil pertama:\t", df.quantile(0.25))
    print("Kuartil kedua:\t\t", df.quantile(0.5))
    print("Kuartil ketiga:\t\t", df.quantile(0.75))
    print("Interquartile Range:\t", iqr(df))
    print("Skewness:\t\t", df.skew())
    print("Kurtosis:\t\t", df.kurtosis())

list_of_column = df.columns.values.tolist()
for i in range(0, 9):
    print("Deskripsi statistik untuk kolom "+str(list_of_column[i]))
    descriptive_statistics(df[list_of_column[i]])
    print()
```

Deskripsi statistik untuk kolom pH
Mean: 7.0871927687138205
Median: 7.029490455474185
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil dari data dengan nilai 0.2274990502021987
Standar Deviasi: 1.5728029470456644
Variansi: 2.473709110235527
Range: 13.7725009497978
Nilai Minimum: 0.2274990502021987
Nilai Maksimum: 13.999999999999998
Kuartil pertama: 6.09078502142353
Kuartil kedua: 7.029490455474185
Kuartil ketiga: 8.053006240791538
Interquartile Range: 1.9622212193680078
Skewness: 0.04853451405270669
Kurtosis: 0.6269041256617065

Deskripsi statistik untuk kolom Hardness
Mean: 195.96920903783553
Median: 197.20352491941043
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil dari data dengan nilai 73.4922336890611
Standar Deviasi: 32.64316585942984
Variansi: 1065.5762773262459
Range: 243.84589036652147
Nilai Minimum: 73.4922336890611
Nilai Maksimum: 317.33812405558257
Kuartil pertama: 176.74065667669896
Kuartil kedua: 197.20352491941043
Kuartil ketiga: 216.44758866727156
Interquartile Range: 39.7069319905726
Skewness: -0.08532104172868622
Kurtosis: 0.5254804942991402

Deskripsi statistik untuk kolom Solids
Mean: 21904.67343905309
Median: 20926.88215534375
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil dari data dengan nilai 320.942611274359
Standar Deviasi: 8625.39791119058
Variansi: 74397489.12637082
Range: 56167.72980146483
Nilai Minimum: 320.942611274359
Nilai Maksimum: 56488.67241273919
Kuartil pertama: 15614.412961614333
Kuartil kedua: 20926.88215534375
Kuartil ketiga: 27170.534648603603
Interquartile Range: 11556.12168698927
Skewness: 0.5910113724580447
Kurtosis: 0.33732026745944976

Deskripsi statistik untuk kolom Chloramines
Mean: 7.134322344600092
Median: 7.1420143046226645
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil dari data dengan nilai 1.3908709048851806
Standar Deviasi: 1.5852140982642096
Variansi: 2.5129037373356113
Range: 11.736129095114823
Nilai Minimum: 1.3908709048851806
Nilai Maksimum: 13.127000000000002
Kuartil pertama: 6.138326387572855
Kuartil kedua: 7.1420143046226645
Kuartil ketiga: 8.109933216133502
Interquartile Range: 1.9716068285606472
Skewness: 0.013003497779569528
Kurtosis: 0.5497821097667472

Deskripsi statistik untuk kolom Sulfate
Mean: 333.21137641518925
Median: 332.2141128069568
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil dari data dengan nilai 129.00000000000003
Standar Deviasi: 41.21111102560977
Variansi: 1698.355671965135
Range: 352.03064230599716
Nilai Minimum: 129.00000000000003
Nilai Maksimum: 481.0306423059972
Kuartil pertama: 307.6269864860709
Kuartil kedua: 332.2141128069568
Kuartil ketiga: 359.26814739141554
Interquartile Range: 51.641160905344634
Skewness: -0.04572780443653543
Kurtosis: 0.7868544988131605

Deskripsi statistik untuk kolom Conductivity
Mean: 426.4767083525792
Median: 423.43837202443706
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil dari data dengan nilai 201.6197367551575
Standar Deviasi: 80.70187180729437
Variansi: 6512.792113200974
Range: 551.7228828031471
Nilai Minimum: 201.6197367551575
Nilai Maksimum: 753.3426195583046
Kuartil pertama: 366.61921929632433
Kuartil kedua: 423.43837202443706
Kuartil ketiga: 482.2097724598859
Interquartile Range: 115.5905531635616
Skewness: 0.26801233302645316

```
Kurtosis:        -0.23720600574806516

Deskripsi statistik untuk kolom OrganicCarbon
Mean:                  14.357939902048088
Median:          14.323285610653329
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil
dari data dengan nilai 2.1999999999999886
Standar Deviasi: 3.3257700016987193
Variansi:        11.060746104199099
Range:                 24.80670661116602
Nilai Minimum:         2.1999999999999886
Nilai Maksimum:        27.00670661116601
Kuartil pertama: 12.122530374047727
Kuartil kedua:         14.323285610653329
Kuartil ketiga:        16.683561746173808
Interquartile Range:   4.561031372126081
Skewness:        -0.02021975629181238
Kurtosis:        0.031018388192253

Deskripsi statistik untuk kolom Trihalomethanes
Mean:                  66.40071666307463
Median:          66.48204080309809
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil
dari data dengan nilai 8.577012932983806
Standar Deviasi: 16.08110898232513
Variansi:        258.602066101418
Range:                 115.4229870670162
Nilai Minimum:         8.577012932983806
Nilai Maksimum:        124.0
Kuartil pertama: 55.94999302803186
Kuartil kedua:         66.48204080309809
Kuartil ketiga:        77.2946128060674
Interquartile Range:   21.344619778035543
Skewness:        -0.05138268451619478
Kurtosis:        0.2230167810639787

Deskripsi statistik untuk kolom Turbidity
Mean:                  3.969496912630371
Median:          3.967373963531836
Semua data unik sehingga modus yang dipilih adalah nilai yang terkecil
dari data dengan nilai 1.45
Standar Deviasi: 0.7804710407083955
Variansi:        0.6091350453844459
Range:                 5.044748555990993
Nilai Minimum:         1.45
Nilai Maksimum:        6.494748555990993
Kuartil pertama: 3.442881623557439
Kuartil kedua:         3.967373963531836
Kuartil ketiga:        4.5146627202018825
Interquartile Range:   1.0717810966444437
```

```
Skewness:         -0.03226597968019271
Kurtosis:         -0.049830796949249745
```

Interquartile range adalah selisih dari nilai Q3 dan Q1 yang biasanya digunakan untuk mengidentifikasi nilai pencilan

Skewness dapat diartikan sebagai ukuran ketidaksimetrian suatu distribusi data. Skewness dapat bernilai positif, negatif, serta nol. Atribut yang memiliki nilai skewness positif artinya data berada lebih banyak disebelah kiri dibandingkan kanan karena ekor kurva berada pada sebelah kanan begitupula sebaliknya

Kurtosis digunakan untuk menunjukan derajat keruncingan, semakin besar maka semakin runcing kurva tersebut. Nilai nol menunjukan data normal nilai positif menunjukan data semakin homogen serta nilai negatif menunjukan data semakin menyebar dan tidak homogen

Kondisi ideal ketika nilai skewness bernilai nol serta nilai kurtosis bernilai 3. Kondisi ini menunjukan bahwa kurva terdistribusi secara normal. Dapat terlihat dari beberapa atribut yang ada pada bagian sebelumnya tidak ada kurva yang memiliki distribusi normal namun ada kurga yang memiliki nilai skewness yang nol artinya kurvanya normal

NO 2

Membuat Visualisasi plot distribusi, dalam bentuk histogram dan boxplot untuk setiap kolom numerik. Berikan uraian penjelasan kondisi setiap kolom berdasarkan kedua plot tersebut

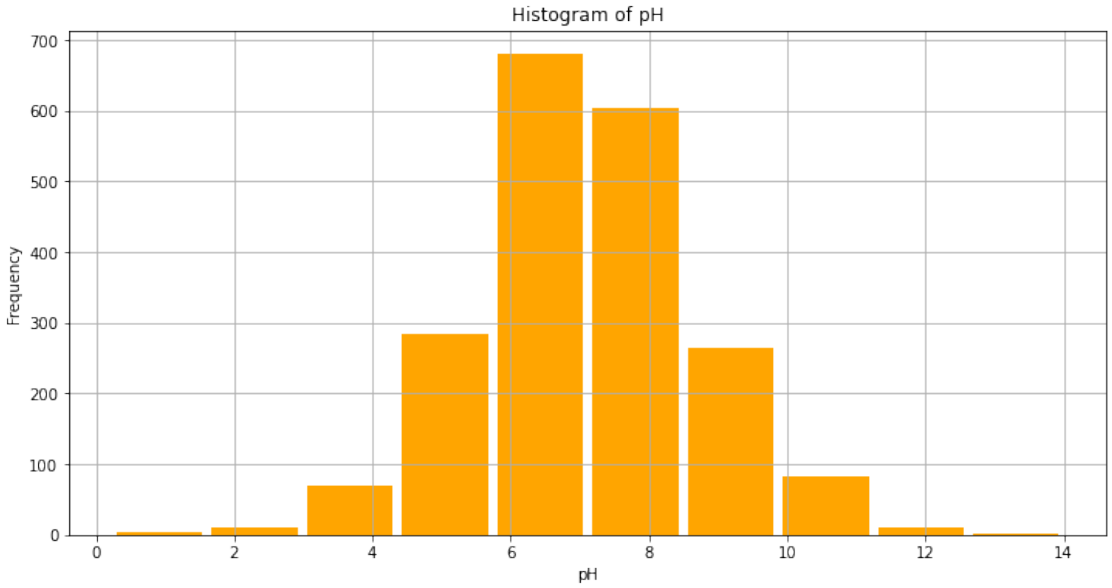
```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as st
import numpy as np

col_names = ['id', 'pH', 'Hardness', 'Solids', 'Chloramines',
'Sulfate', 'Conductivity', 'OrganicCarbon', 'Trihalomethanes',
'Turbidity', 'Potability']
df = pd.read_csv('../data/water_potability.csv', names=col_names)
df.drop(["id", "Potability"], axis=1)
```

```
             pH    Hardness        Solids  Chloramines     Sulfate  \
0      8.316766  214.373394  22018.417441     8.059332  356.886136
1      9.092223  181.101509  17978.986339     6.546600  310.135738
2      5.584087  188.313324  28748.687739     7.544869  326.678363
3     10.223862  248.071735  28749.716544     7.513408  393.663396
4      8.635849  203.361523  13672.091764     4.563009  303.309771
...         ...         ...           ...          ...         ...
2005   8.197353  203.105091  27701.794055     6.472914  328.886838
2006   8.989900  215.047358  15921.412018     6.297312  312.931022
2007   6.702547  207.321086  17246.920347     7.708117  304.510230
2008  11.491011   94.812545  37188.826022     9.263166  258.930600
2009   6.069616  186.659040  26138.780191     7.747547  345.700257

      Conductivity  OrganicCarbon  Trihalomethanes  Turbidity
0       363.266516      18.436524       100.341674   4.628771
1       398.410813      11.558279        31.997993   4.075075
2       280.467916       8.399735        54.917862   2.559708
3       283.651634      13.789695        84.603556   2.672989
4       474.607645      12.363817        62.798309   4.401425
...            ...            ...              ...        ...
2005    444.612724      14.250875        62.906205   3.361833
2006    390.410231       9.899115        55.069304   4.613843
2007    329.266002      16.217303        28.878601   3.442983
2008    439.893618      16.172755        41.558501   4.369264
2009    415.886955      12.067620        60.419921   3.669712

[2010 rows x 9 columns]
```

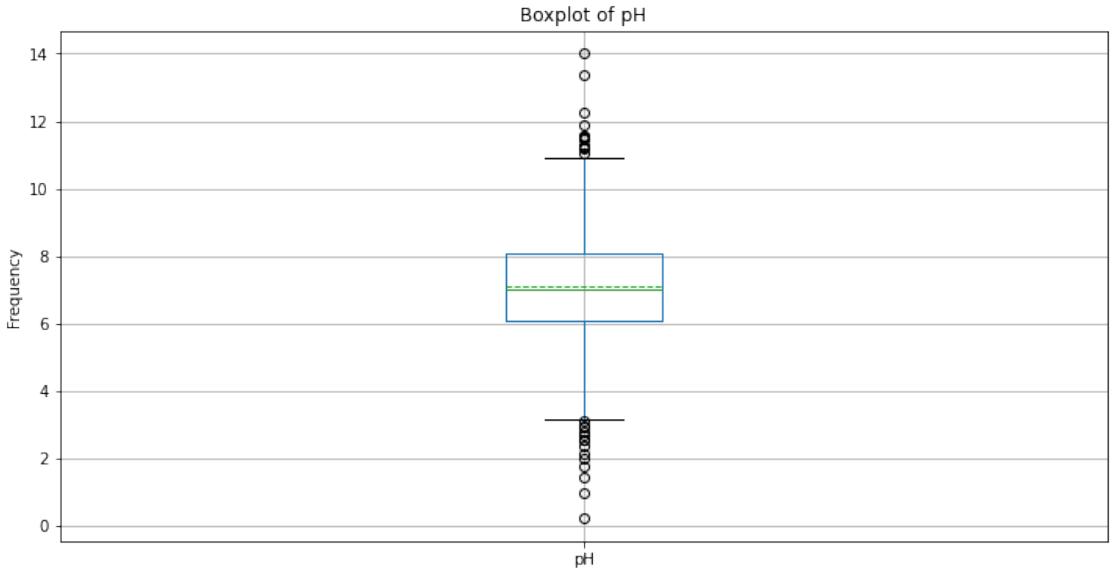
pH

```
df.hist(column = 'pH', figsize = (12,6), rwidth = 0.9, bins = 10,
color = 'orange')
# give title
plt.title('Histogram of pH')
```

```
# give xlabel
plt.xlabel('pH')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```




```
df.boxplot(column='pH', figsize = (12,6), meanline = True, showmeans =
True)
# give title
plt.title('Boxplot of pH')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```

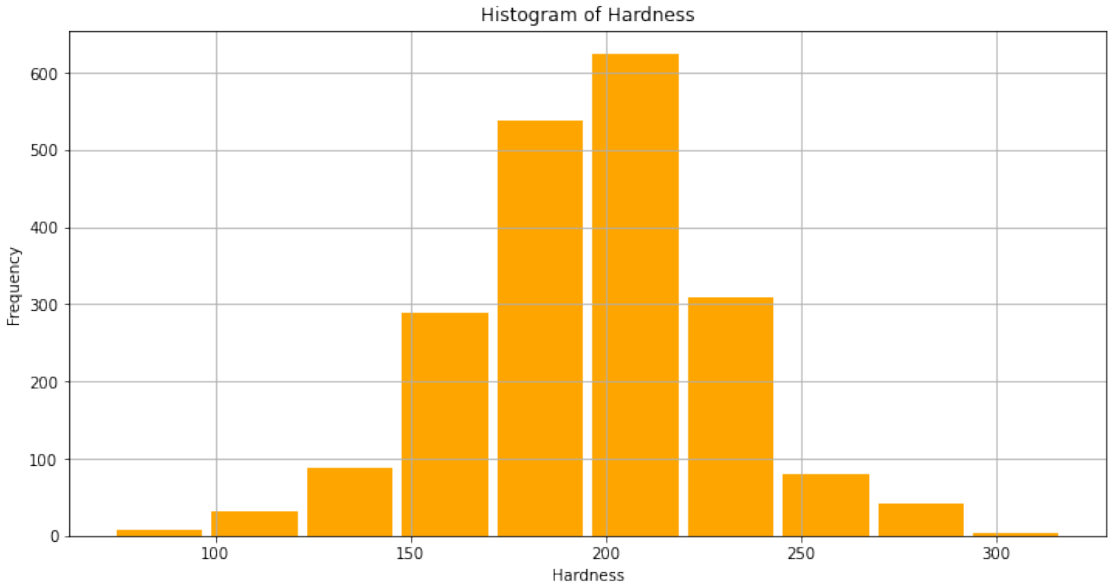


Berdasrkan histogram tersebut dapat terlihat bahwa distribusi pH terlihat condong ke arah kiri (negatively skewed) serta dari boxplot dapat terlihat bahwa terdapat beberapa data outlier ytang berada di rentang 0 - 3 dan 11 - 14, terlihat median berada di sekitar 7 dengan kuartil pertama di sekitar 6, kuartil ketiga disekitar 8, dengan nilai minimum di sekitar 0.2 dan nilai maksimum mendekati 14

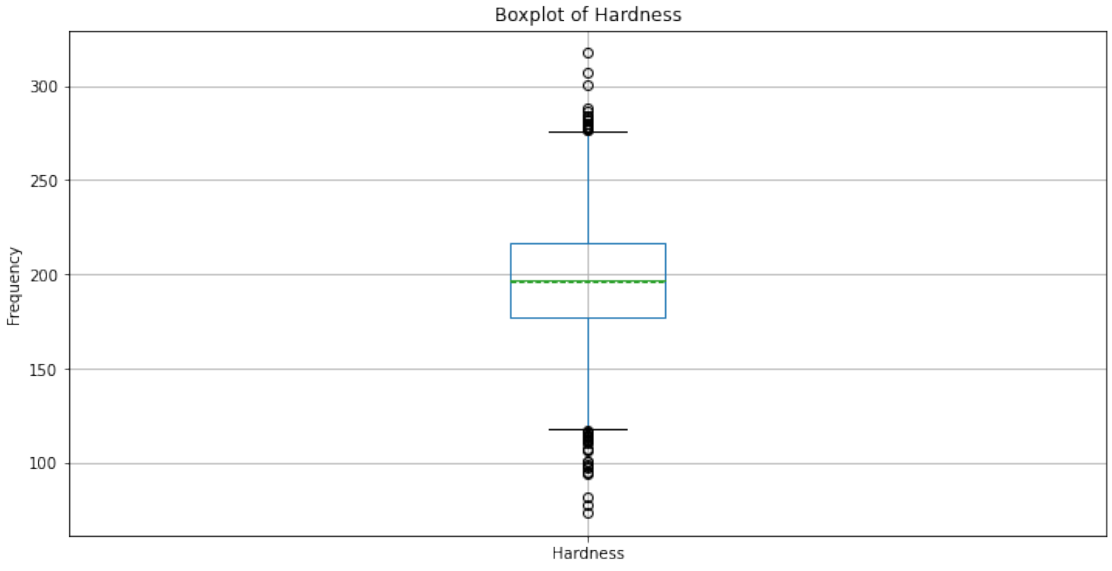
Hardness

```
df.hist(column = 'Hardness', figsize = (12,6), rwidth = 0.9, bins = 10, color = 'orange')
# give title
plt.title('Histogram of Hardness')
# give xlabel
plt.xlabel('Hardness')
# give ylabel
plt.ylabel('Frequency')
# show plot
```

```
Text(0, 0.5, 'Frequency')
```




```
df.boxplot(column='Hardness', figsize = (12,6), meanline = True,
showmeans = True)
# give title
plt.title('Boxplot of Hardness')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```

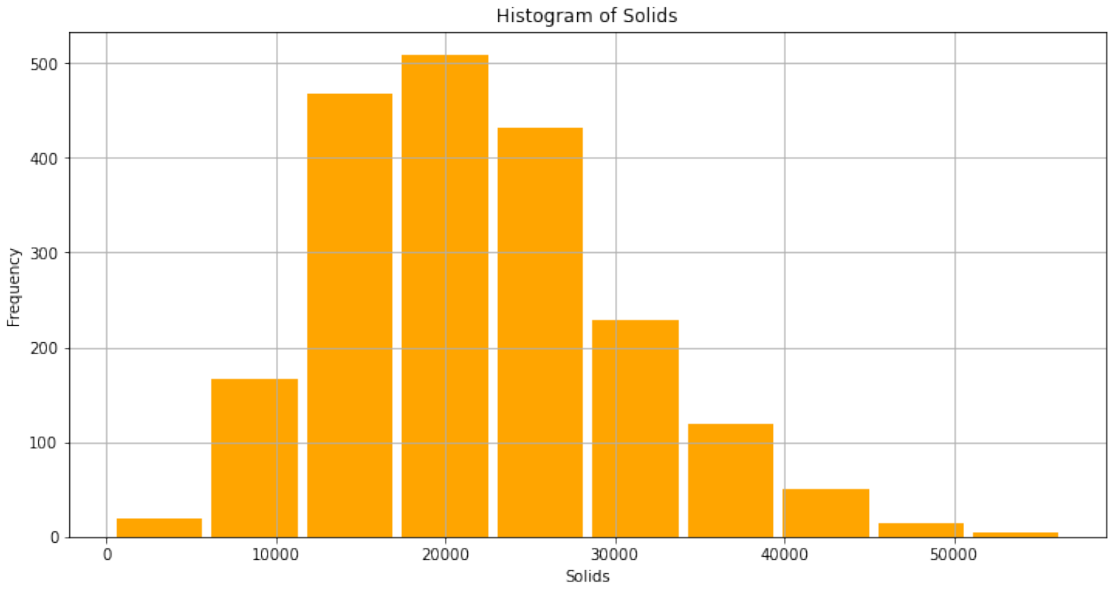


Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Hardness memiliki kecenderungan ke arah kiri (negatively skewed) serta jika dilihat dari boxplot terdapat beberapa outlier dengan nilai hardness 0 - ~110 dan ~270 - ~350, terlihat median berada di sekitar 190an dengan kuartil pertama di sekitar 170an, kuartil ketiga disekitar 210an, dengan nilai minimum di sekitar 70an dan nilai maksimum mendekati 300an

Solids

```
df.hist(column = 'Solids', figsize = (12,6), rwidth = 0.9, bins = 10,
color = 'orange')
# give title
plt.title('Histogram of Solids')
# give xlabel
plt.xlabel('Solids')
# give ylabel
plt.ylabel('Frequency')
# show plot
```

```
Text(0, 0.5, 'Frequency')
```



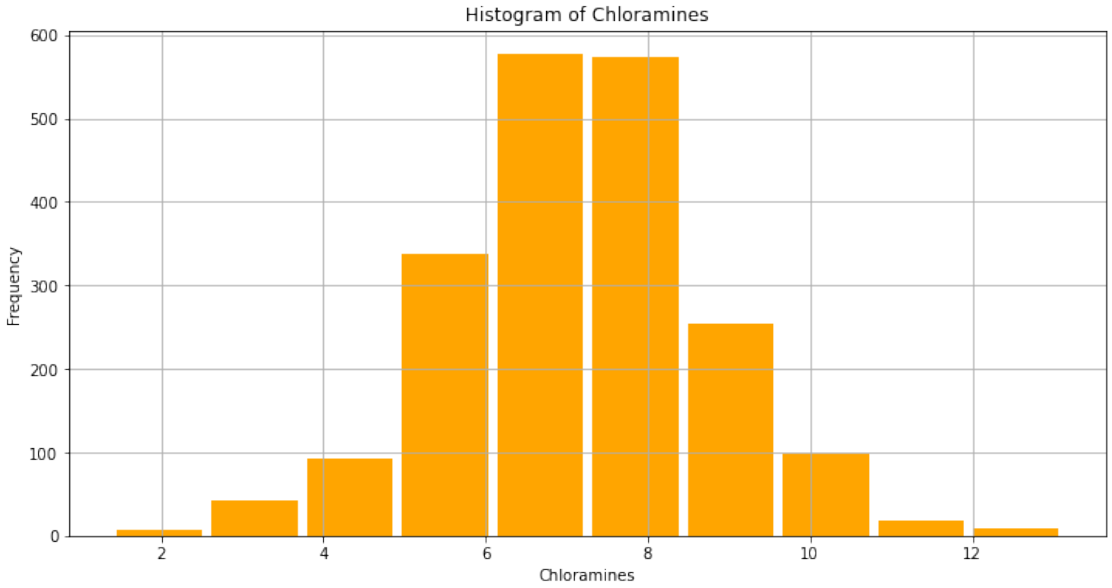

```
df.boxplot(column='Solids', figsize = (12,6), meanline = True,
showmeans = True)
# give title
plt.title('Boxplot of Solids')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```

Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Solids memiliki kecenderungan ke arah kanan (Positively skewed) serta jika dilihat dari boxplot terdapat outlier pada satu sisi saja yaitu yang lebih besar dariapda nilai maximum pada rentang ~45000 - ~55000, terlihat median berada di sekitar 20000an dengan kuartil pertama di sekitar 1500an, kuartil ketiga disekitar 2700an, dengan nilai minimum di sekitar 300an dan nilai maksimum mendekati 50000an

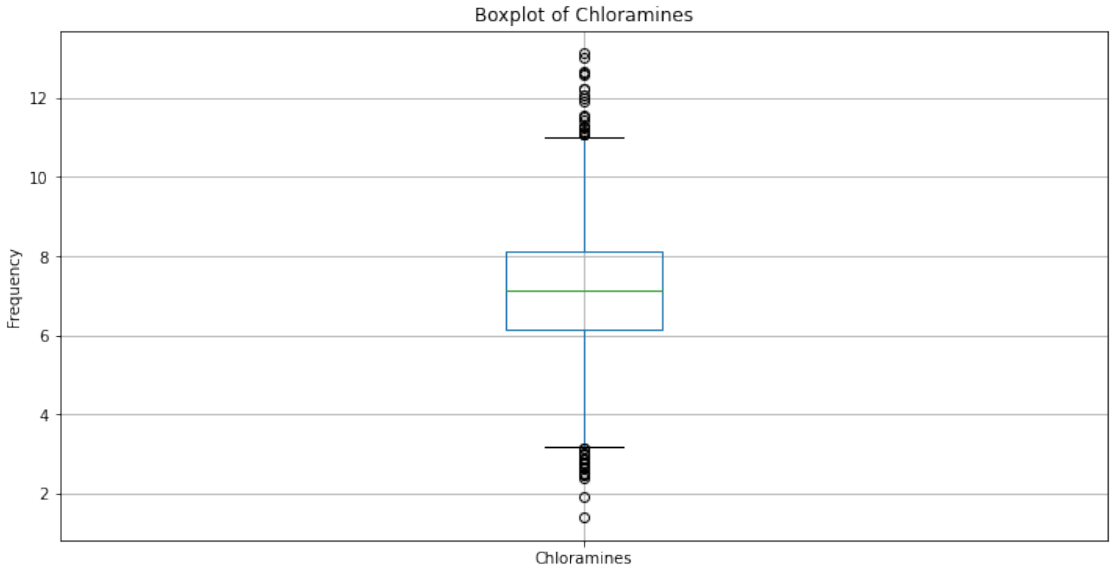
Chloramines

```
df.hist(column = 'Chloramines', figsize = (12,6), rwidth = 0.9, bins = 10, color = 'orange')
# give title
plt.title('Histogram of Chloramines')
# give xlabel
plt.xlabel('Chloramines')
# give ylabel
plt.ylabel('Frequency')
# show plot
```

```
Text(0, 0.5, 'Frequency')
```




```
df.boxplot(column='Chloramines', figsize = (12,6), meanline = True,
showmeans = True)
# give title
plt.title('Boxplot of Chloramines')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```



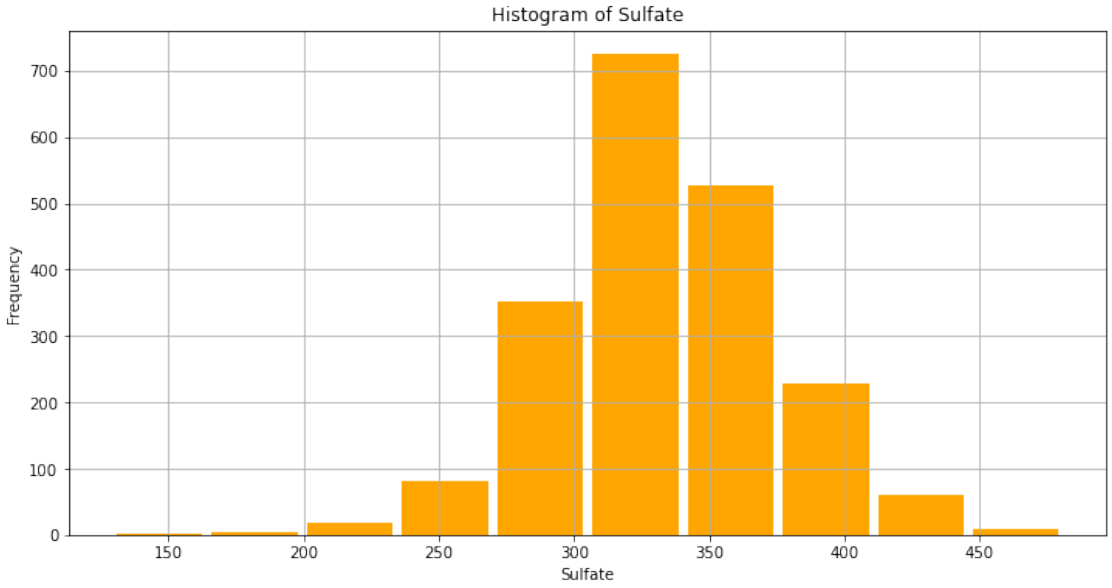
Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Chloramines memiliki distribusi secara normal (tidak ada kecenderungan ke arah kanan maupun kiri) serta jika dilihat dari boxplot terdapat beberapa outlier dengan nilai Chloramines ~1 - ~3 dan ~11 - ~13, terlihat median berada di sekitar 7an dengan kuartil pertama di sekitar 6an, kuartil

ketiga disekitar 8an, dengan nilai minimum di sekitar 1an dan nilai maksimum mendekati 13an

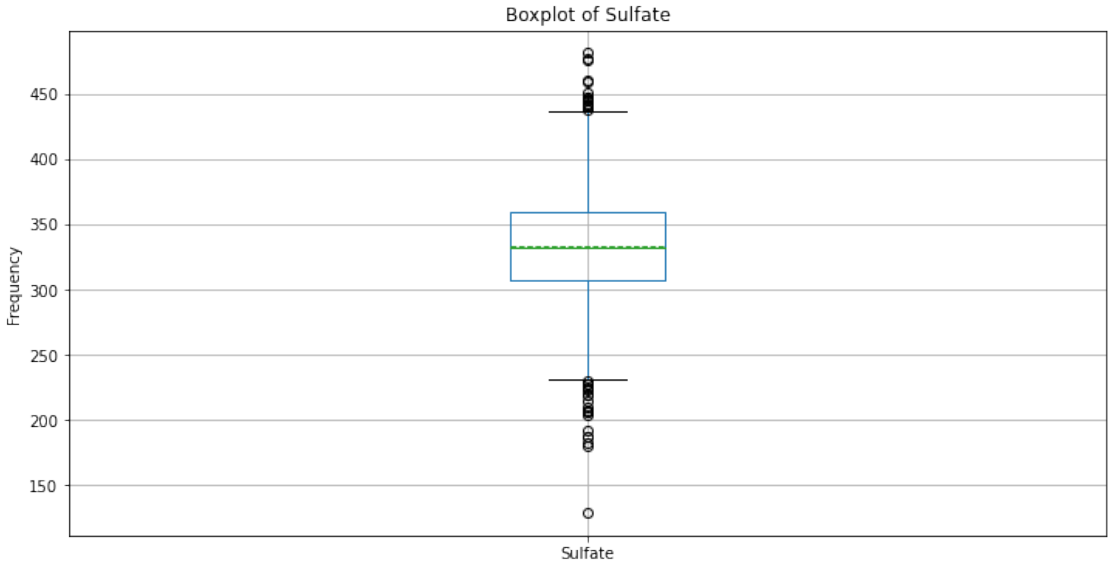
Sulfate

```
df.hist(column = 'Sulfate', figsize = (12,6), rwidth = 0.9, bins = 10,
color = 'orange')
# give title
plt.title('Histogram of Sulfate')
# give xlabel
plt.xlabel('Sulfate')
# give ylabel
plt.ylabel('Frequency')
# show plot
```

```
Text(0, 0.5, 'Frequency')
```



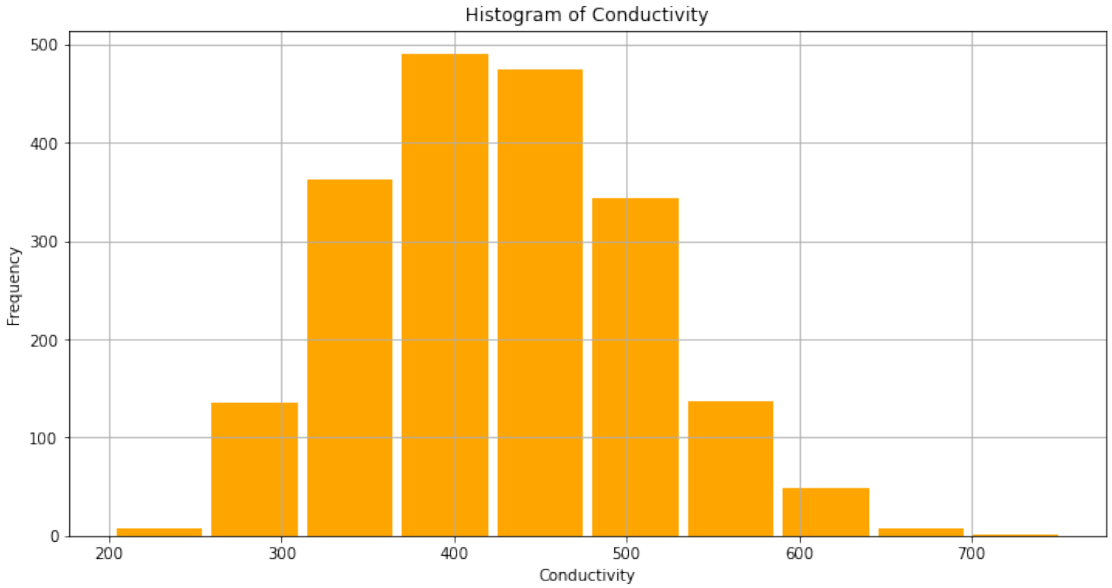

```
df.boxplot(column='Sulfate', figsize = (12,6), meanline = True,
showmeans = True)
# give title
plt.title('Boxplot of Sulfate')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```



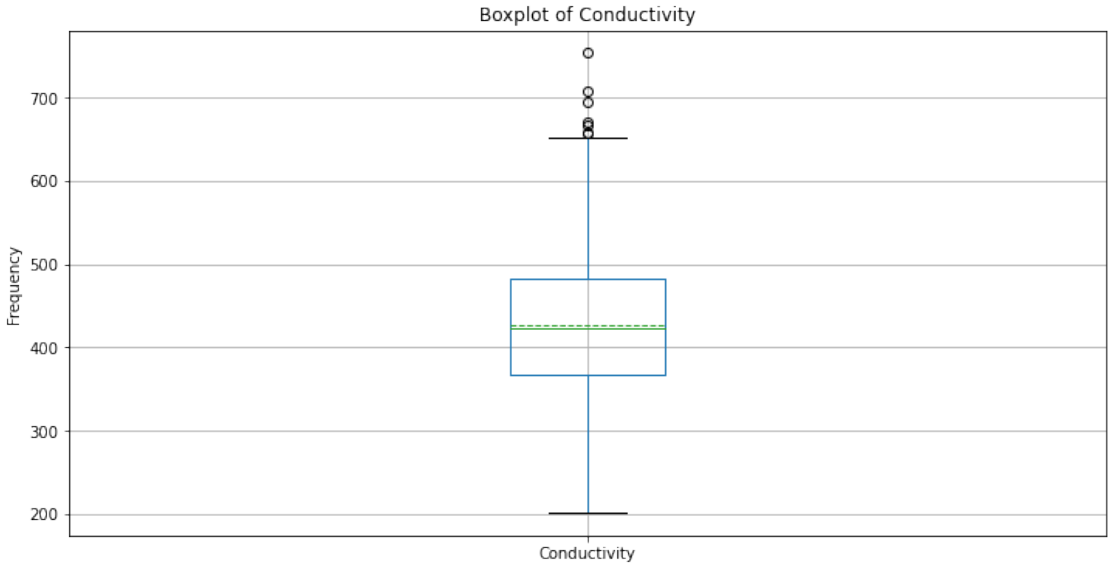
Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Sulfate memiliki kecenderungan ke arah kiri (negatively skewed) serta jika dilihat dari boxplot terdapat beberapa outlier dengan nilai Sulfate 0 - ~230 dan ~440 - ~500, terlihat median berada di sekitar 330an dengan kuartil pertama di sekitar 300an, kuartil ketiga disekitar 360an, dengan nilai minimum di sekitar 130an dan nilai maksimum mendekati 480an

Conductivity

```
df.hist(column = 'Conductivity', figsize = (12,6), rwidth = 0.9, bins
= 10, color = 'orange')
# give title
plt.title('Histogram of Conductivity')
# give xlabel
plt.xlabel('Conductivity')
# give ylabel
plt.ylabel('Frequency')
# show plot

Text(0, 0.5, 'Frequency')
```




```
df.boxplot(column='Conductivity', figsize = (12,6), meanline = True,
showmeans = True)
# give title
plt.title('Boxplot of Conductivity')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```



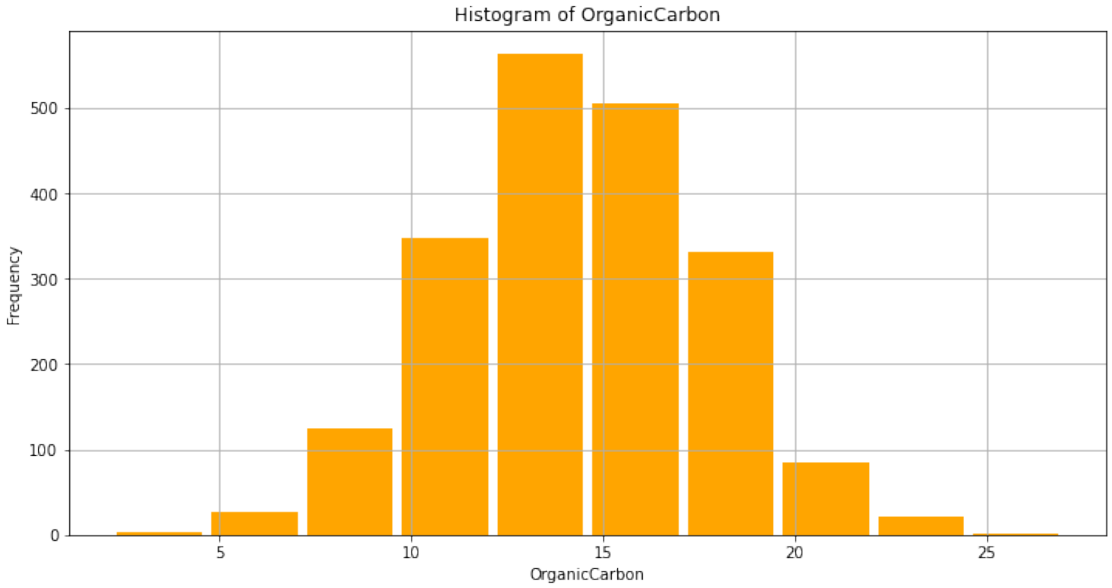
Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Conductivity memiliki kecenderungan ke arah kanan (Positively skewed) serta jika dilihat dari boxplot hanya terdapat outlier yang lebih besar daripada nilai maksimum di range sekitar 650 - 750 nilai conductivity, terlihat median berada di sekitar 420an dengan kuartil pertama di sekitar

360an, kuartil ketiga disekitar 480an, dengan nilai minimum di sekitar 200an dan nilai maksimum mendekati 750an

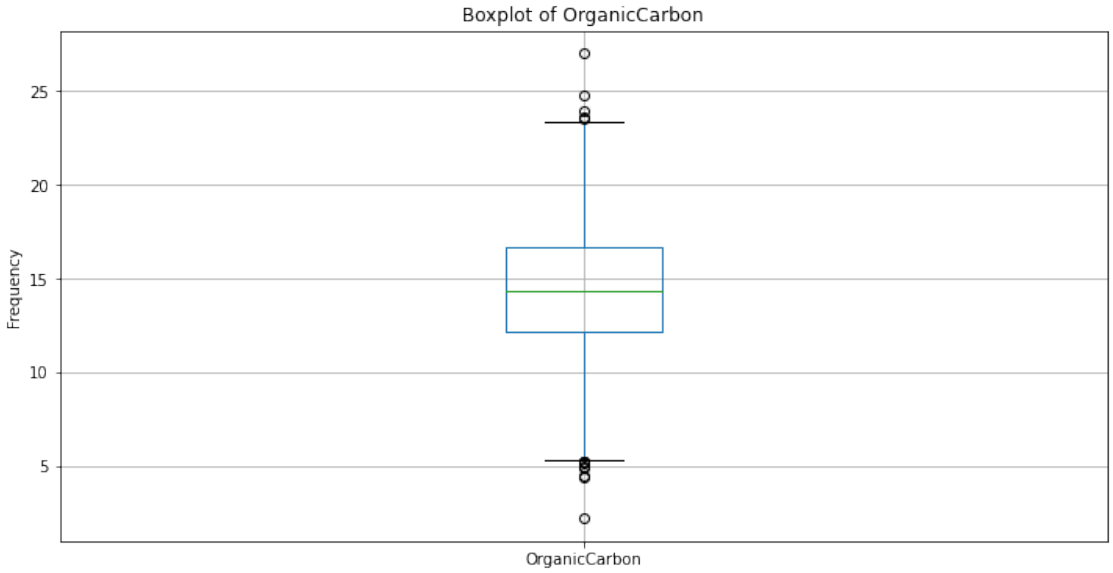
OrganicCarbon

```
df.hist(column = 'OrganicCarbon', figsize = (12,6), rwidth = 0.9, bins
= 10, color = 'orange')
# give title
plt.title('Histogram of OrganicCarbon')
# give xlabel
plt.xlabel('OrganicCarbon')
# give ylabel
plt.ylabel('Frequency')
# show plot
```

```
Text(0, 0.5, 'Frequency')
```




```
df.boxplot(column='OrganicCarbon', figsize = (12,6), meanline = True,
showmeans = True)
# give title
plt.title('Boxplot of OrganicCarbon')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```

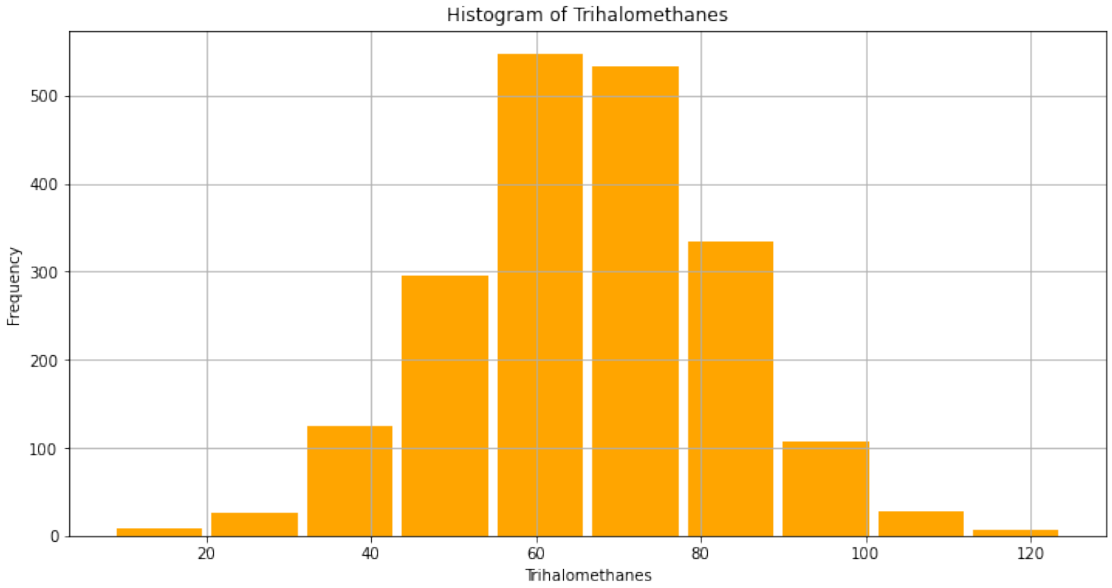


Berdasarkan histogram tersebut dapat terlihat bahwa distribusi OrganicCarbon memiliki kecenderungan ke arah kiri (negatively skewed) serta jika dilihat dari boxplot terdapat sedikit outlier dibagian atas dengan range 24 - 27 serta pada bagian bawah di range 0 - 5, terlihat median berada di sekitar 14an dengan kuartil pertama di sekitar 12an, kuartil ketiga disekitar 17an, dengan nilai minimum di sekitar 2an dan nilai maksimum mendekati 27an

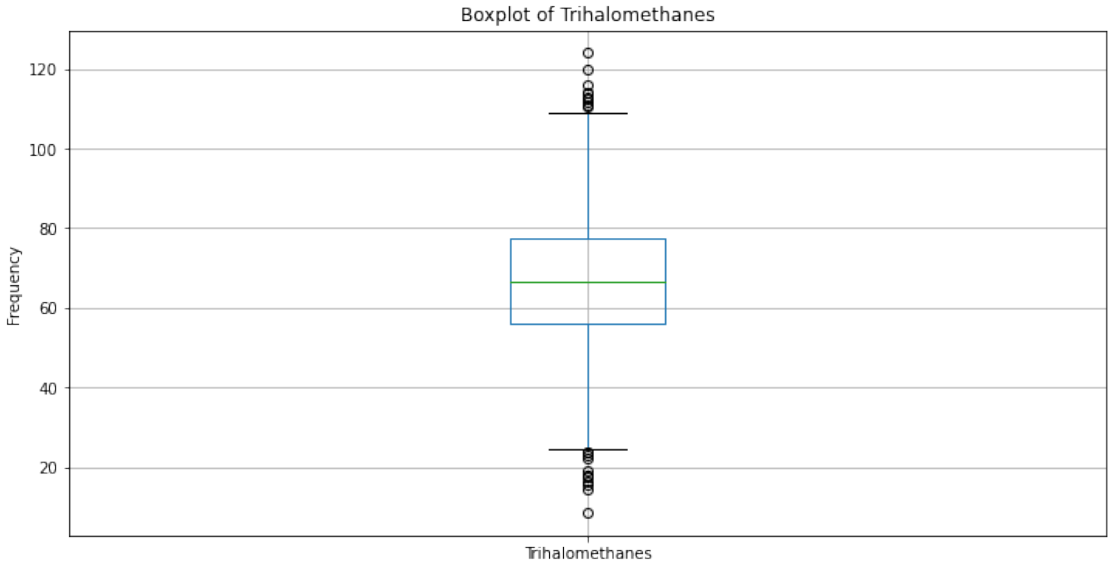
Trihalomethanes

```
df.hist(column = 'Trihalomethanes', figsize = (12,6), rwidth = 0.9,
bins = 10, color = 'orange')
# give title
plt.title('Histogram of Trihalomethanes')
# give xlabel
plt.xlabel('Trihalomethanes')
# give ylabel
plt.ylabel('Frequency')
```

```
Text(0, 0.5, 'Frequency')
```




```
df.boxplot(column='Trihalomethanes', figsize = (12,6), meanline =
True, showmeans = True)
# give title
plt.title('Boxplot of Trihalomethanes')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```



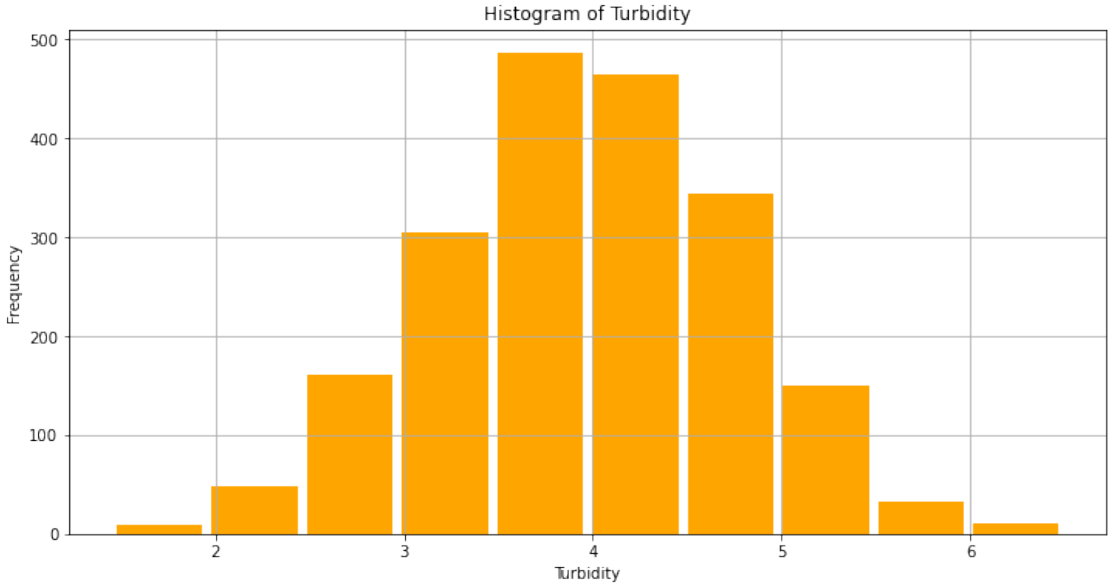
Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Trihalomethanes memiliki distribusi normal (tidak memiliki kecenderungan ke arah kiri maupun kanan) serta jika dilihat dari boxplot terdapat beberapa outlier pada bagian bawah dengan range 110 - 130 dan pada bagian bawah 0 - 22, terlihat median berada di sekitar 66an dengan kuartil

pertama di sekitar 55an, kuartil ketiga disekitar 77an, dengan nilai minimum di sekitar 8an dan nilai maksimum mendekati 125an

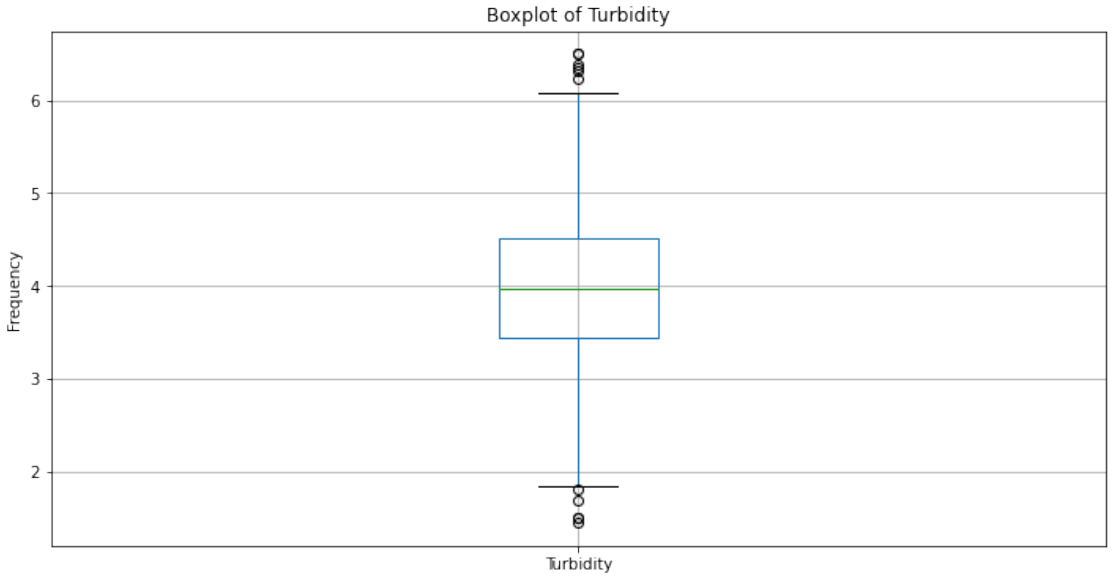
Turbidity

```
df.hist(column = 'Turbidity', figsize = (12,6), rwidth = 0.9, bins = 10, color = 'orange')
# give title
plt.title('Histogram of Turbidity')
# give xlabel
plt.xlabel('Turbidity')
# give ylabel
plt.ylabel('Frequency')
```

```
Text(0, 0.5, 'Frequency')
```




```
df.boxplot(column='Turbidity', figsize = (12,6), meanline = True, showmeans = True)
# give title
plt.title('Boxplot of Turbidity')
# give ylabel
plt.ylabel('Frequency')
# show plot
plt.show()
```



Berdasarkan histogram tersebut dapat terlihat bahwa distribusi Turbidity memiliki distribusi normal (tidak memiliki kecenderungan ke arah kiri maupun kanan) serta jika dilihat dari boxplot terdapat beberapa outlier pada bagian atas dengan nilai turbidty sekitar 6.2 serta dibawah minimum yaitu sekitar 1.8, terlihat median berada di sekitar 4 dengan kuartil pertama di sekitar 3.5, kuartil ketiga disekitar 4.5an, dengan nilai minimum di sekitar 1.5an dan nilai maksimum mendekati 6.5an

No 3

Menentukan setiap kolom numerik berdistribusi normal atau tidak

```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as st
import numpy as np
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

col_names = ['id', 'pH', 'Hardness', 'Solids', 'Chloramines',
'Sulfate', 'Conductivity', 'OrganicCarbon', 'Trihalomethanes',
'Turbidity', 'Potability']
df = pd.read_csv('../data/water_potability.csv', names=col_names)
df.drop(["id", "Potability"], axis=1)
```

```
             pH    Hardness        Solids  Chloramines     Sulfate  \
0      8.316766  214.373394  22018.417441     8.059332  356.886136
1      9.092223  181.101509  17978.986339     6.546600  310.135738
2      5.584087  188.313324  28748.687739     7.544869  326.678363
3     10.223862  248.071735  28749.716544     7.513408  393.663396
4      8.635849  203.361523  13672.091764     4.563009  303.309771
...         ...         ...           ...          ...         ...
2005   8.197353  203.105091  27701.794055     6.472914  328.886838
2006   8.989900  215.047358  15921.412018     6.297312  312.931022
2007   6.702547  207.321086  17246.920347     7.708117  304.510230
2008  11.491011   94.812545  37188.826022     9.263166  258.930600
2009   6.069616  186.659040  26138.780191     7.747547  345.700257

      Conductivity  OrganicCarbon  Trihalomethanes  Turbidity
0       363.266516      18.436524       100.341674   4.628771
1       398.410813      11.558279        31.997993   4.075075
2       280.467916       8.399735        54.917862   2.559708
3       283.651634      13.789695        84.603556   2.672989
4       474.607645      12.363817        62.798309   4.401425
...            ...            ...              ...        ...
2005    444.612724      14.250875        62.906205   3.361833
2006    390.410231       9.899115        55.069304   4.613843
2007    329.266002      16.217303        28.878601   3.442983
2008    439.893618      16.172755        41.558501   4.369264
2009    415.886955      12.067620        60.419921   3.669712

[2010 rows x 9 columns]
```

Menentukan setiap kolom numerik berdistribusi normal atau tidak. Gunakan normality test yang dikaitkan dengan histogram plot.

Normality test dilakukan dengan menggunakan fungsi `normaltest` dari library scipy. Implementasi normality test jenis ini didasarkan pada **D'Agostino-Pearson Test**.

Tes D'Agostino-Pearson, atau disebut juga Omnibus D'Agostino, dilakukan dengan menggabungkan hasil tes skewness dan kurtosis D'Agostino. Rumusnya diberikan sebagai berikut:

$$K^2 = Z_s^2 + Z_k^2$$

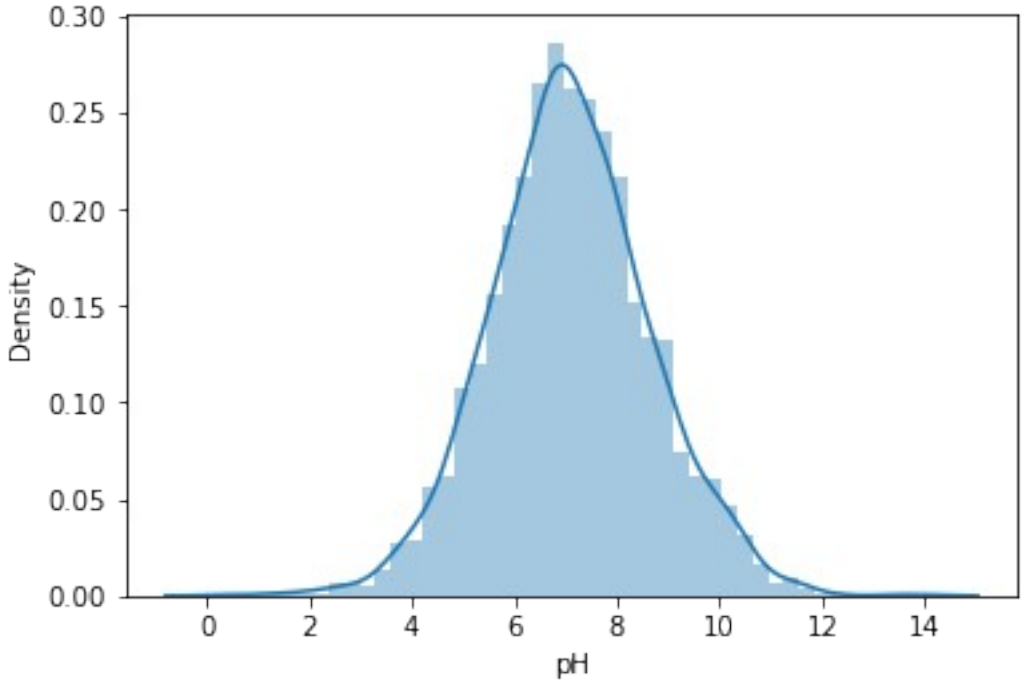
Z_s^2 adalah z-score dari tes skewness D'Agostino dan Z_k^2 adalah z-score dari tes kurtosis D'Agostino. Jika hipotesis null terbukti, K^2 diaproksimasi terdistribusi chi-squared dengan derajat kebebasan 2.

Dalam soal ini, diambil hipotesis null (H_0) yaitu data terdistribusi normal. H_0 diuji dengan membandingkan nilai α yang ditetapkan sebesar 0.05 dengan p-value yang didapat dari `normaltest`. H_0 akan diterima jika p-value lebih besar dari α dan akan ditolak jika p-value lebih kecil dari α.

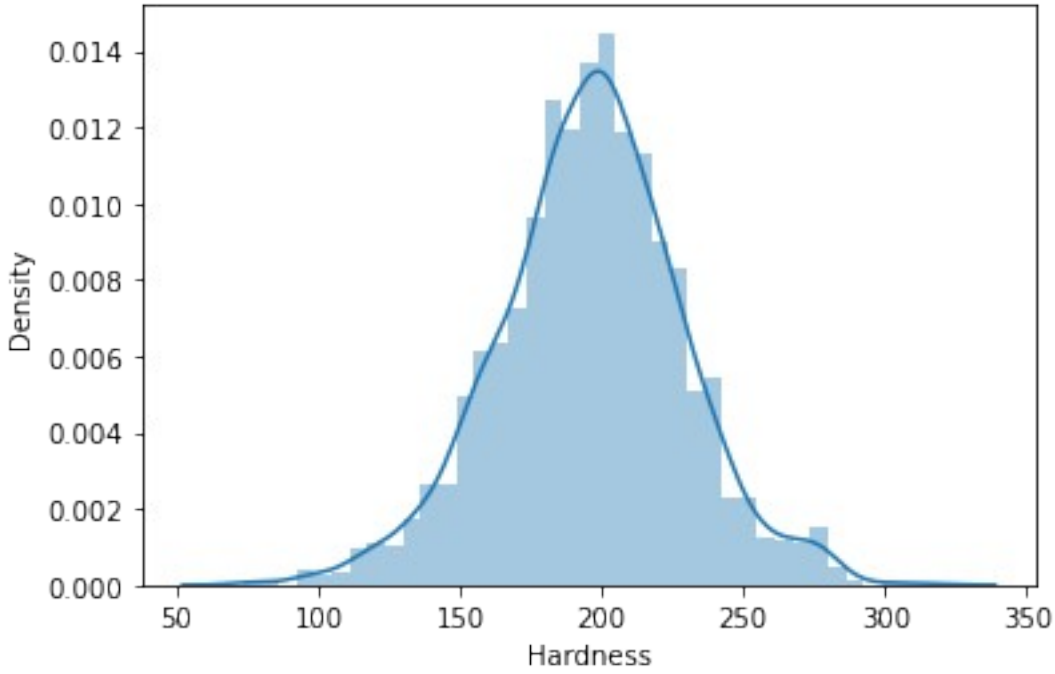
Untuk membantu pembuktian hasil normality test, ditampilkan pula histogram distribusi data dengan fungsi `distplot` dari library seaborn. Data yang terdistribusi normal akan menghasilkan histogram berbentuk kurva simetris (bell curve).

```
for i in range(1, 10):
    stat, p = st.normaltest(df[col_names[i]])
    alpha = 0.05

    # plot distribution)
    print(f"Nilai p ialah: {p}")
    if p < alpha:
        print('Hipotesis nol ditolak, data '+ col_names[i] + ' tidak
berdistribusi normal')
    else:
        print('Hipotesis nol diterima, data '+ col_names[i] + '
berdistribusi normal')
    sns.distplot(df[col_names[i]])
    plt.show()
```

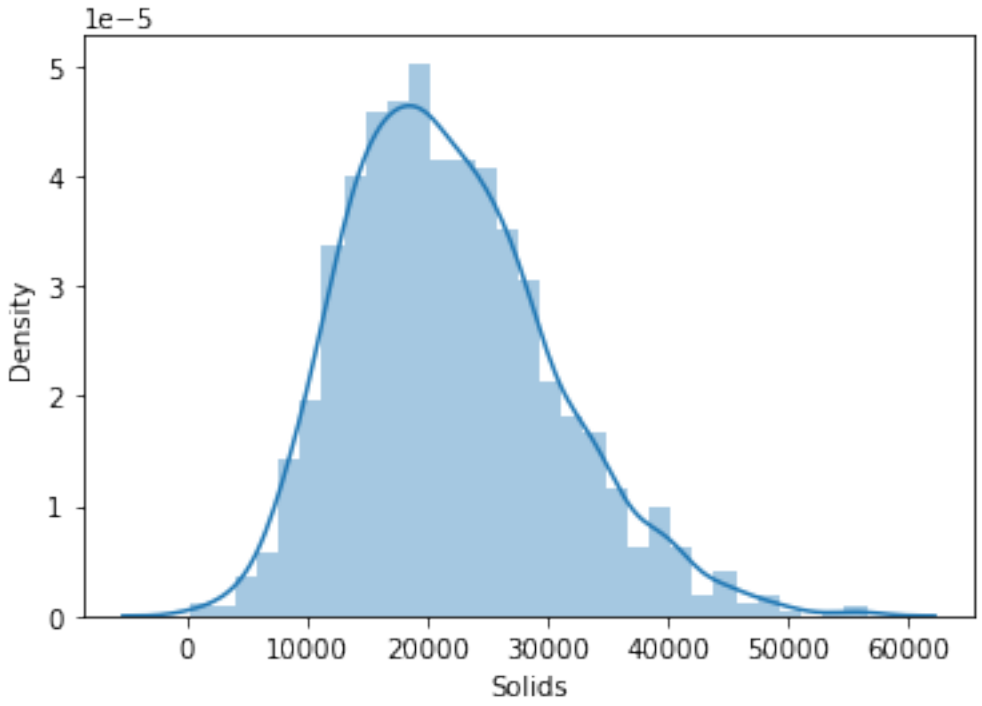
```
Nilai p ialah: 2.6514813346797777e-05
Hipotesis nol ditolak, data pH tidak berdistribusi normal
```



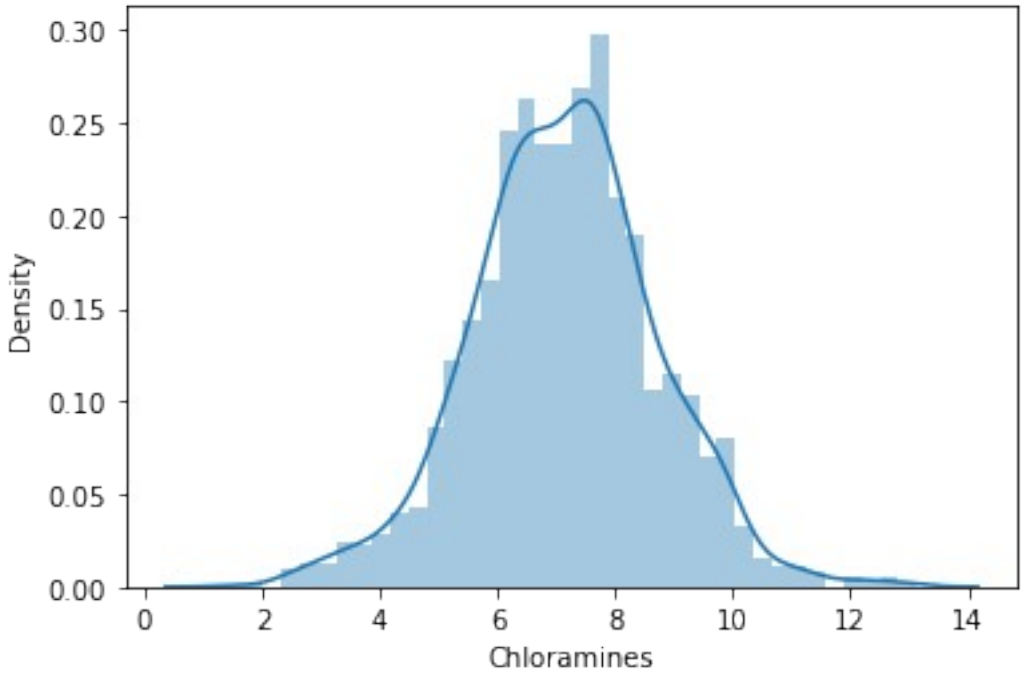
Nilai p ialah: 0.00013442428699593753
Hipotesis nol ditolak, data Hardness tidak berdistribusi normal



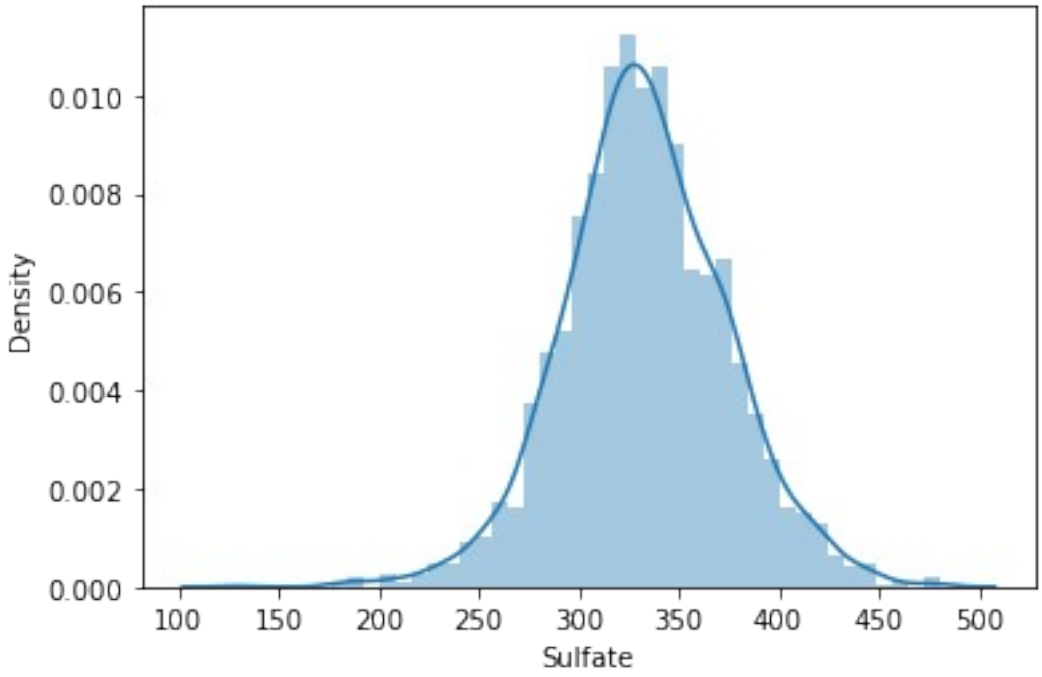
Nilai p ialah: 2.0796613688739523e-24
Hipotesis nol ditolak, data Solids tidak berdistribusi normal



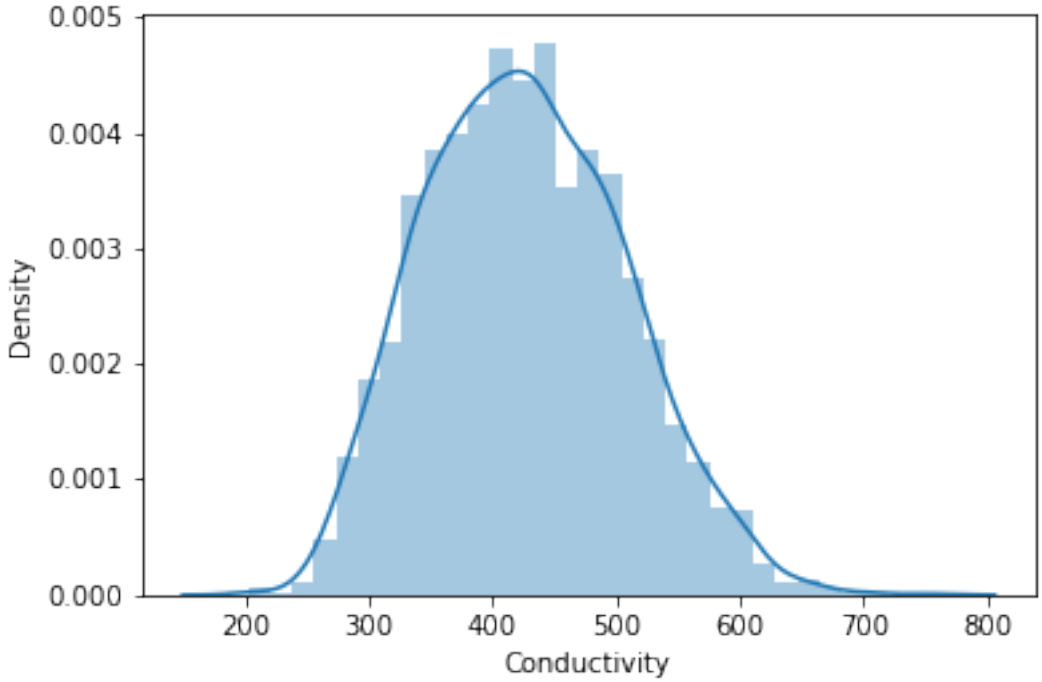
Nilai p ialah: 0.0002504831654753917
Hipotesis nol ditolak, data Chloramines tidak berdistribusi normal



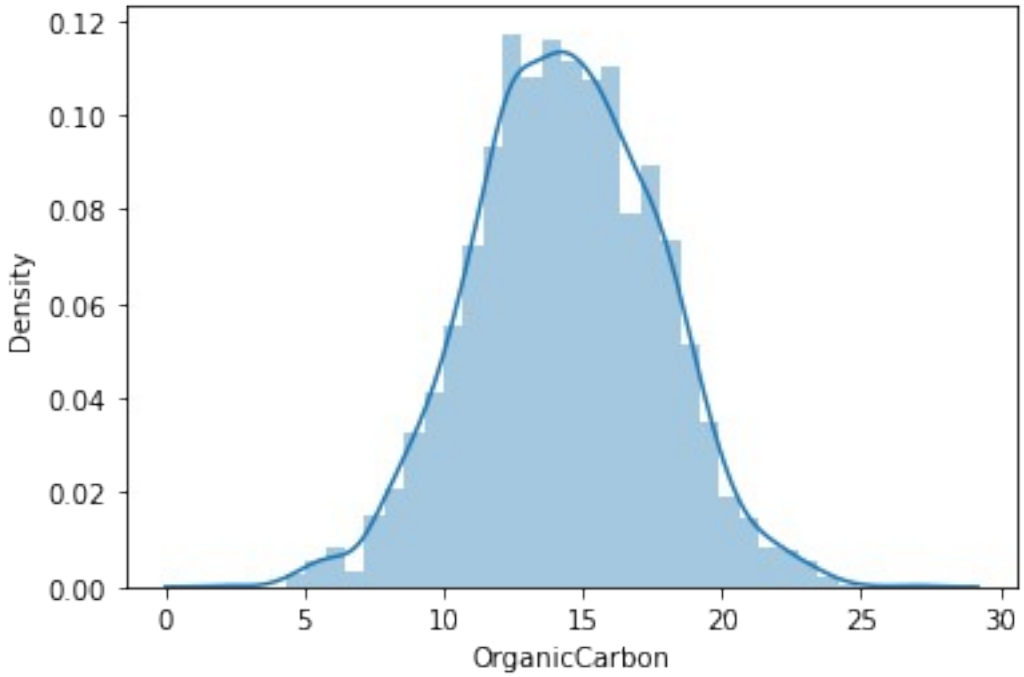
Nilai p ialah: 4.4255936678013136e-07
Hipotesis nol ditolak, data Sulfate tidak berdistribusi normal



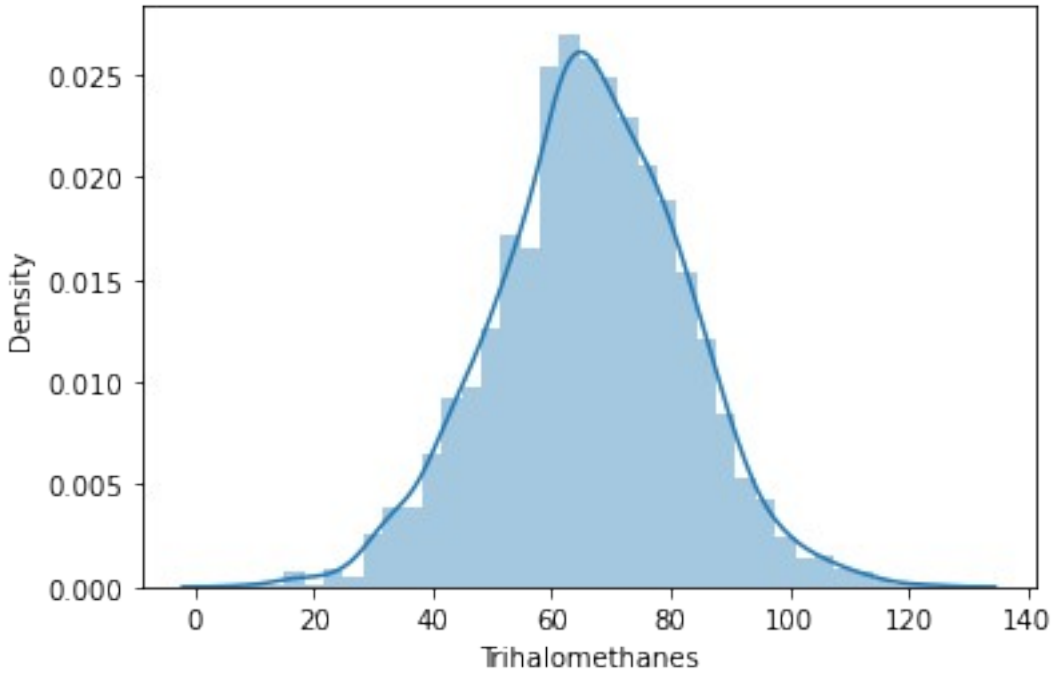
Nilai p ialah: 4.39018078287845e-07
Hipotesis nol ditolak, data Conductivity tidak berdistribusi normal



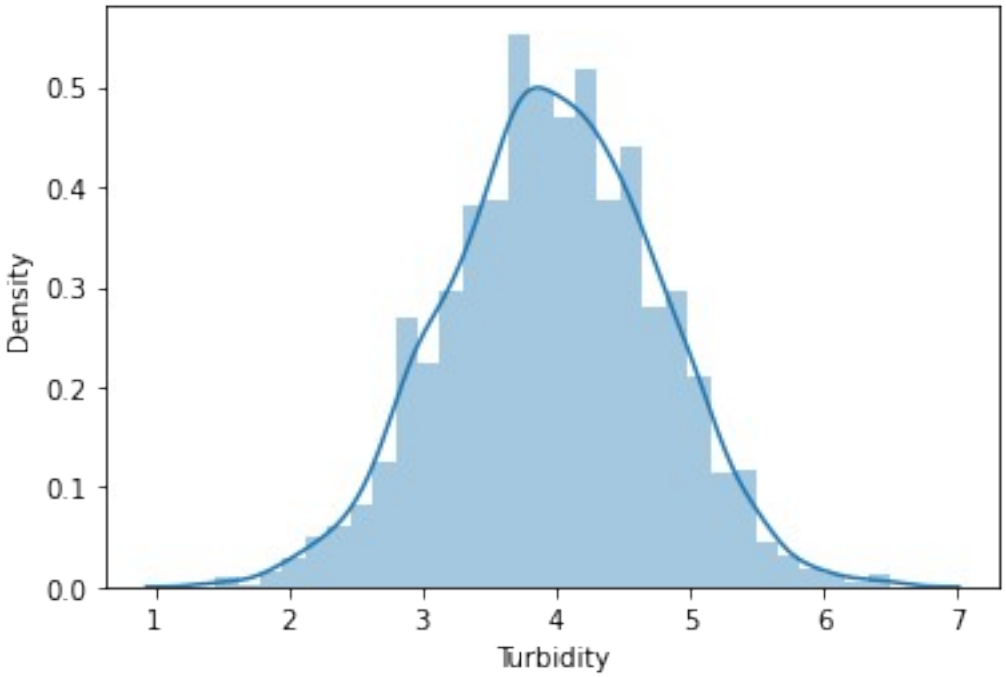
Nilai p ialah: 0.8825496581408284
Hipotesis nol diterima, data OrganicCarbon berdistribusi normal



Nilai p ialah: 0.1043598441875204
Hipotesis nol diterima, data Trihalomethanes berdistribusi normal



Nilai p ialah: 0.7694717369961169
Hipotesis nol diterima, data Turbidity berdistribusi normal


0.5
0.4
0.3
0.2
0.1
0.0
Density
1
2
3
4
5
6
7
Turbidity

NO 4

Melakukan test hipotesis 1 sampel

```
import scipy.stats as st
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.stats.weightstats import ztest
from statsmodels.stats.proportion import proportions_ztest

col_names = ['id', 'pH', 'Hardness', 'Solids', 'Chloramines',
'Sulfate',
             'Conductivity', 'OrganicCarbon', 'Trihalomethanes',
'Turbidity', 'Potability']
df = pd.read_csv('../data/water_potability.csv', names=col_names)

df.head()
```

```
   id         pH    Hardness        Solids  Chloramines     Sulfate  \
0   1   8.316766  214.373394  22018.417441     8.059332  356.886136
1   2   9.092223  181.101509  17978.986339     6.546600  310.135738
2   3   5.584087  188.313324  28748.687739     7.544869  326.678363
3   4  10.223862  248.071735  28749.716544     7.513408  393.663396
4   5   8.635849  203.361523  13672.091764     4.563009  303.309771

   Conductivity  OrganicCarbon  Trihalomethanes  Turbidity  Potability

0    363.266516      18.436524       100.341674   4.628771           0

1    398.410813      11.558279        31.997993   4.075075           0

2    280.467916       8.399735        54.917862   2.559708           0

3    283.651634      13.789695        84.603556   2.672989           0

4    474.607645      12.363817        62.798309   4.401425           0
```

a) Nilai rata - rata pH di atas 7?

H_0: Nilai rata rata pH sama dengan 7 ($\mu = 7$)
H_1: Nilai rata rata pH lebih dari 7 ($\mu > 7$)

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan one tailed test ke arah kanan (right tailed test) karena $\mu > 7$.
Ambil critical section $z > z_\alpha$

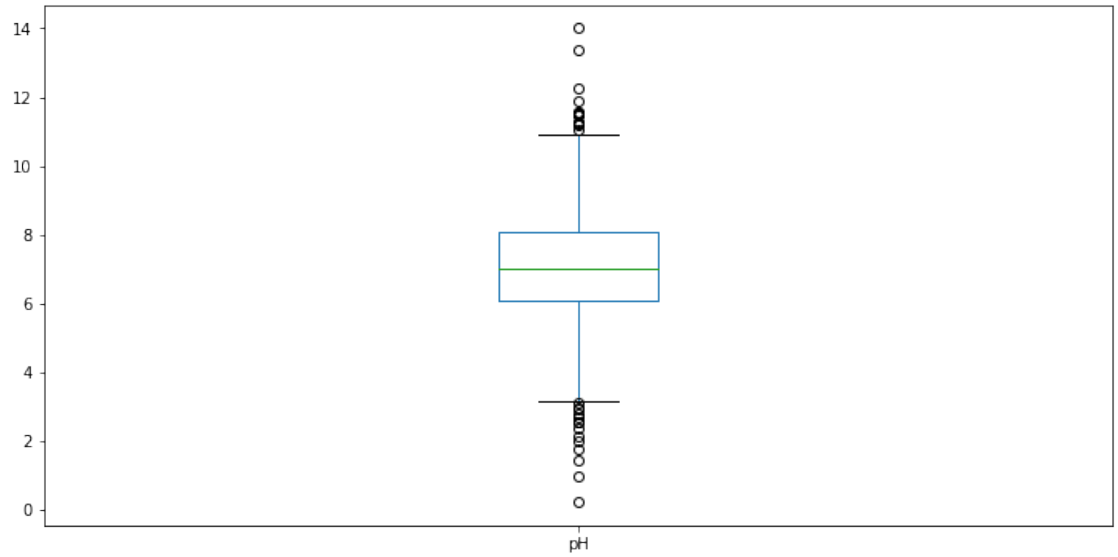
Hitung nilai z dengan rumus

$$z = \frac{\acute{x} - \mu_0}{\sigma / \sqrt{n}}$$

Tolak H_0 jika $z > z_\alpha$ dan $p < \alpha$
Terima H_0 jika $z \leq z_\alpha$ dan $p \geq \alpha$

```
miu = 7
alpha = 0.05

#calculate z and p using ztest module
z, p = ztest(df["pH"], value=miu)

#calculate z_alpha
z_a = st.norm.ppf(1-alpha)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
df["pH"].plot(kind="box", figsize=(12,6))
plt.title=("pH")
plt.show()
```

```
Nilai z: 2.4854
Nilai z_alpha: 1.6449
Nilai p: 0.0129
```



Nilai z lebih besar dibandingkan dengan z_α $(2.4854 > 1.6449)$
Nilai p lebih kecil dibandingkan α $(0.0129 < 0.05)$
Maka tolak H_0

Kesimpulan: rata-rata pH lebih dari 7

b) Nilai rata rata hardness tidak sama dengan 205?

H_0: Nilai rata rata Hardness sama dengan 205 ($\mu = 205$)
H_1: Nilai rata rata Hardness tidak sama dengan 205 ($\mu \neq 205$)

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan two tailed test karena harus di cek pada bagian kanan $\mu > 205$ dengan $z > z_{\alpha/2}$ dan pada bagian kiri $\mu < 205$ dengan $z < -z_{\alpha/2}$

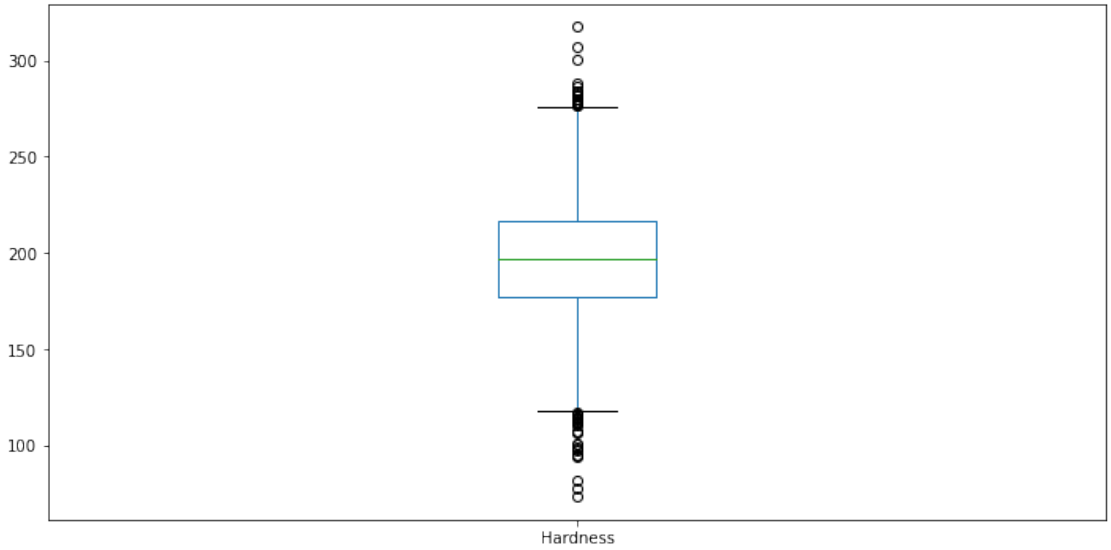
Hitung nilai z dengan rumus

$$z = \frac{\acute{x} - \mu_0}{\sigma / \sqrt{n}}$$

Tolak H_0 jika ¿ atau $z < -z_{\alpha/2}$ ¿ dan $p < \alpha$
Terima H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$ dan $p \geq \alpha$

```
miu = 205
alpha = 0.05

#calculate z and p using ztest module
z, p = ztest(df["Hardness"], value=miu)

#calculate z_alpha
z_a = st.norm.ppf(1-alpha/2)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
df["Hardness"].plot(kind="box", figsize=(12, 6))
plt.title=("Hardness")
plt.show()
```

```
Nilai z: -12.4031
Nilai z_alpha/2: 1.96
Nilai p: 0.0
```



Nilai z lebih kecil dibandingkan dengan $-z_{\alpha/2}\ (-12.4031<1.96)$
Nilai p lebih kecil dibandingkan $\alpha\ (0<0.05)$
Maka tolak H_0

Kesimpulan: rata-rata Hardness tidak sama dengan 205

c) Nilai Rata-rata 100 baris pertama kolom Solids bukan 21900?

H_0: Nilai rata rata 100 baris pertama kolom Solids sama dengan 21900 ($\mu=21900$)
H_1: Nilai rata rata 100 baris pertama kolom Solids tidak sama dengan 21900 ($\mu\neq21900$)

Tingkat Signifikan $\alpha=0.05$

Lakukan uji statistik dengan two tailed test karena harus di cek pada bagian kanan $\mu>21900$ dengan $z>z_{\alpha/2}$ dan pada bagian kiri $\mu<21900$ dengan $z<-z_{\alpha/2}$

Hitung nilai z dengan rumus

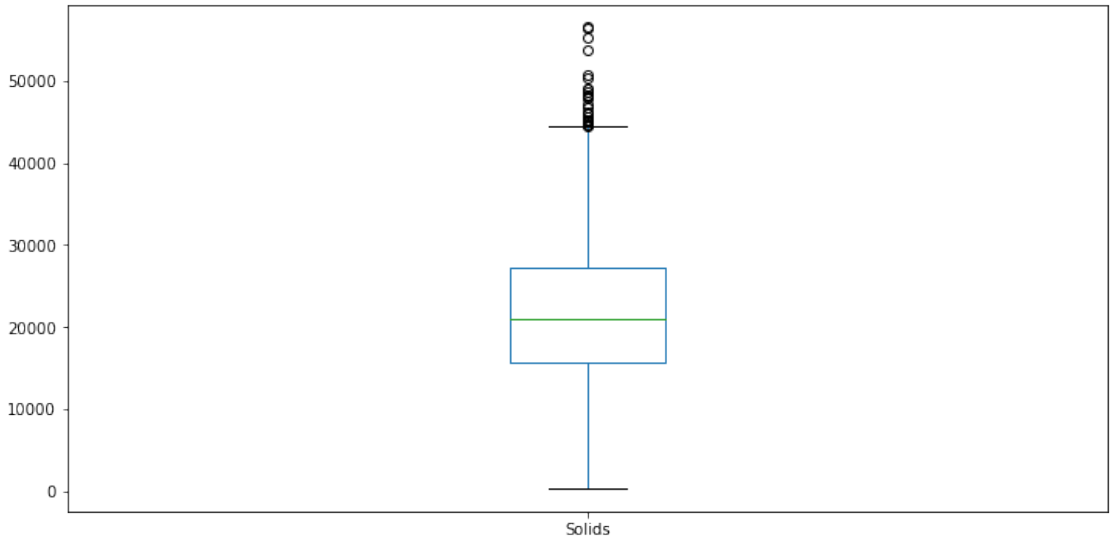
$$z=\frac{\acute{x}-\mu_0}{\sigma/\sqrt{n}}$$

Tolak H_0 jika ¿ atau $z<-z_{\alpha/2}$ ¿ dan $p<\alpha$
Terima H_0 jika $-z_{\alpha/2}\leq z\leq z_{\alpha/2}$ dan $p\geq\alpha$

```
miu = 21900
alpha = 0.05

#calculate z and p using ztest module
z, p = ztest(df["Solids"].head(100), value=miu)

#calculate z_alpha/2
z_a = st.norm.ppf(1-alpha/2)
```

```
print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
df["Solids"].plot(kind="box", figsize=(12, 6))
plt.title = ("Solids")
plt.show()
```

```
Nilai z: 0.5637
Nilai z_alpha/2: 1.96
Nilai p: 0.573
```



Nilai z berada pada rentang $-z_{\alpha/2} \le z \le z_{\alpha/2} \left(-1.96 \le 0.5637 \le 1.96\right)$
Nilai p lebih besar dibandingkan $\alpha \left(0.573 \ge 0.05\right)$
Maka terima H_0

Kesimpulan: rata-rata 100 baris pertama kolom solids sama dengan 21900

d) Proporsi nilai Conductivity yang lebih dari 450, adalah tidak sama dengan 10%?

H_0: Proporsi nilai Conductivity yang lebih dari 450 sama dengan 10% ($p = 0.1$)
H_1: Proporsi nilai Conductivity yang lebih dari 450 tidak sama dengan 10% ($p \neq 0.1$)

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan two tailed test karena harus di cek pada bagian kanan dengan $z > z_{\alpha/2}$ dan pada bagian kiri dengan $z < -z_{\alpha/2}$

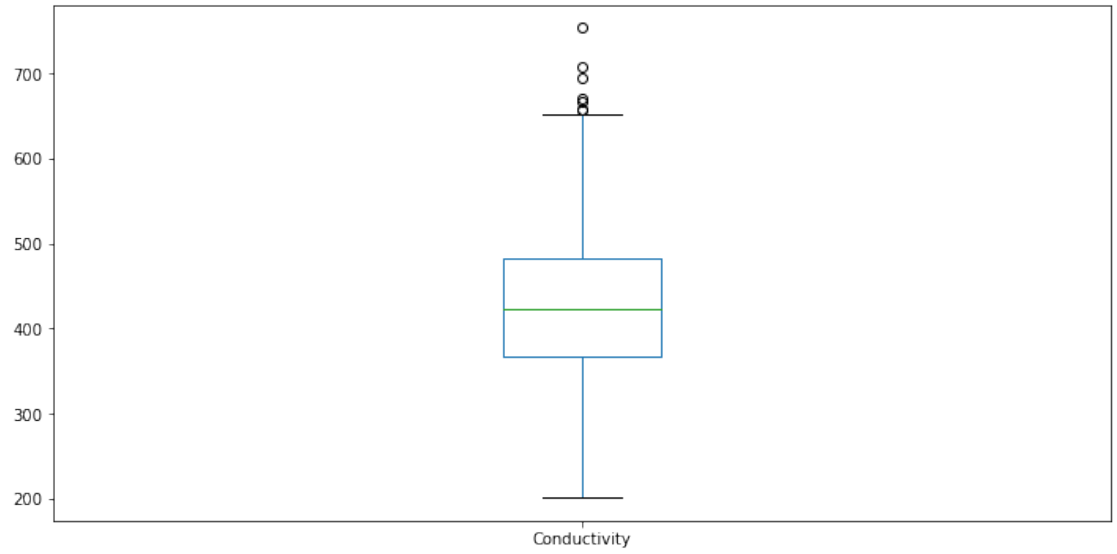
Hitung nilai z dengan rumus

$$z = \frac{\hat{p} - p_0}{\sqrt{p_0 q_0 / n}}$$

Tolak H_0 jika ¿ atau $z < -z_{\alpha/2}$ ¿ dan $p < \alpha$
Terima H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$ dan $p \geq \alpha$

```
p_0 = 0.10
alpha = 0.05

conductivity_over_450 = len(df[df["Conductivity"] > 450])

z, p = proportions_ztest(conductivity_over_450,
                         len(df), value=p_0, prop_var=p_0)
z_a = st.norm.ppf(1-alpha/2)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
df["Conductivity"].plot(kind="box", figsize=(12, 6))
plt.title = ("Conductivity")
plt.show()
```

```
Nilai z: 40.4464
Nilai z_alpha/2: 1.96
Nilai p: 0.0
```



Nilai z lebih dari $z_{\alpha/2}$ $(40.4464 > 1.96)$
Nilai p lebih kecil dibandingkan α $(0.0 < 0.05)$
Maka tolak H_0

Kesimpulan: Proporsi nilai Conductivity yang lebih dari 450 tidak sama dengan 10%

e) Proporsi nilai Trihalomethanes yang kurang dari 40, adalah kurang dari 5%

H_0: Proporsi nilai Trihalomethanes yang kurang dari 40 sama dengan 5% $(p = 0.05)$
H_1: Proporsi nilai Trihalomethanes yang kurang dari 40 kurang dari 5% $(p < 0.05)$

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan one tailed test karena harus di cek pada bagian kiri dengan $z < z_\alpha$

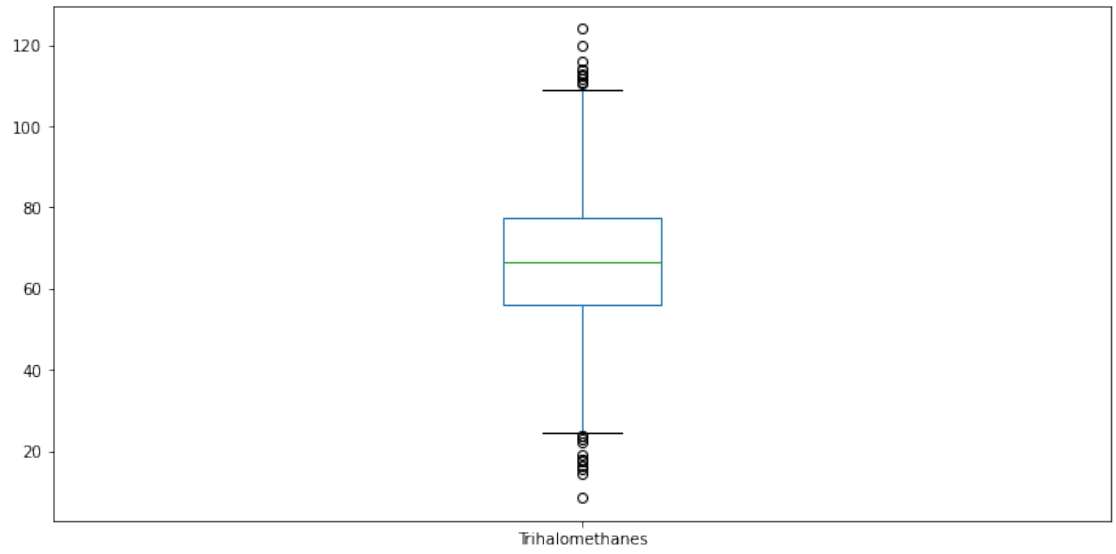
Hitung nilai z dengan rumus

$$z = \frac{\hat{p} - p_0}{\sqrt{p_0 q_0 / n}}$$

Tolak H_0 jika $z < z_\alpha$ dan $p < \alpha$
Terima H_0 jika $z \geq z_\alpha$ dan $p \geq \alpha$

```
p_0 = 0.05
alpha = 0.05

trihalomethanes_less_40 = len(df[df["Trihalomethanes"] < 40])

z, p = proportions_ztest(trihalomethanes_less_40,
                         len(df), value=p_0, prop_var=p_0)
z_a = st.norm.ppf(1-alpha)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
df["Trihalomethanes"].plot(kind="box", figsize=(12, 6))
plt.title = ("Trihalomethanes")
plt.show()
```

```
Nilai z: 0.5629
Nilai z_alpha: 1.6449
Nilai p: 0.5735
```



Nilai z lebih kecil dari z_{α} $(0.5629<1.6449)$
Nilai p lebih besar dibandingkan α $(0.5735>0.05)$
Maka terima H_0

Kesimpulan: Proporsi nilai Trihalomethanes yang kurang dari 40 kurang dari 5%

NO 5

Melakukan test hipotesis 2 sampel

```
import scipy.stats as st
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.stats.weightstats import ztest
from statsmodels.stats.proportion import proportions_ztest

col_names = ['id', 'pH', 'Hardness', 'Solids', 'Chloramines',
'Sulfate',
             'Conductivity', 'OrganicCarbon', 'Trihalomethanes',
'Turbidity', 'Potability']
df = pd.read_csv('../data/water_potability.csv', names=col_names)
df.head()
```

```
   id         pH    Hardness        Solids  Chloramines     Sulfate  \
0   1   8.316766  214.373394  22018.417441     8.059332  356.886136
1   2   9.092223  181.101509  17978.986339     6.546600  310.135738
2   3   5.584087  188.313324  28748.687739     7.544869  326.678363
3   4  10.223862  248.071735  28749.716544     7.513408  393.663396
4   5   8.635849  203.361523  13672.091764     4.563009  303.309771

   Conductivity  OrganicCarbon  Trihalomethanes  Turbidity  Potability

0    363.266516      18.436524       100.341674   4.628771           0

1    398.410813      11.558279        31.997993   4.075075           0

2    280.467916       8.399735        54.917862   2.559708           0

3    283.651634      13.789695        84.603556   2.672989           0

4    474.607645      12.363817        62.798309   4.401425           0
```

a) Data kolom Sulfate dibagi 2 sama rata: bagian awal dan bagian akhir kolom. Benarkah rata-rata kedua bagian tersebut sama?

H_0: Nilai rata-rata kolom awal sulfate sama dengan nilai rata-rata kolom akhir sulfate $(\mu_1 - \mu_2 = 0)$

H_1: Nilai rata-rata kolom awal sulfate tidak sama dengan nilai rata-rata kolom akhir sulfate $(\mu_1 - \mu_2 \neq 0)$

Tingkat signifikan $\alpha = 0.05$

Lakukan pengujian two tailed test karena akan dicek pada bagian kiri dengan $z < -z_{\alpha/2}$ serta bagian kanan dengan $z > z_{\alpha/2}$

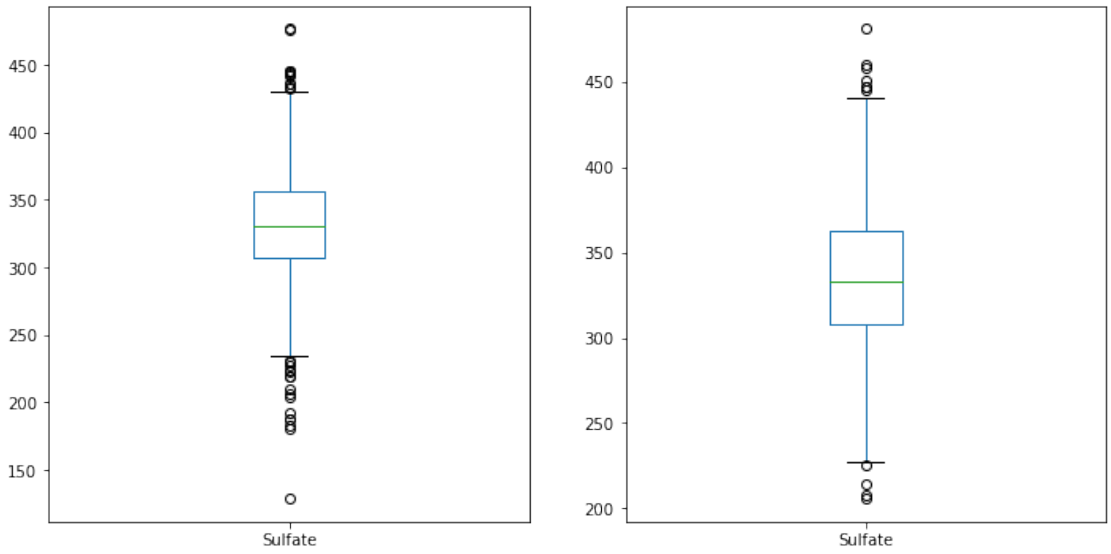
Hitung nilai z:

$$z=\frac{\left(\acute{x}_1-\acute{x}_2\right)-\left(\mu_1-\mu_2\right)}{\sqrt{\sigma_1^2/n_1+\sigma_2^2/n_2}}$$

Tolak H_0 jika ¿ atau $z<-z_{\alpha/2}$ ¿ dan $p<\alpha$
Terima H_0 jika $-z_{\alpha/2}\leq z\leq z_{\alpha/2}$ dan $p\geq\alpha$

```
delta = 0
alpha = 0.05

#calculate z and p using ztest module from beginning and end
len_data_per_2 = len(df) // 2
df_beg = df["Sulfate"][:len_data_per_2]
df_end = df["Sulfate"][len_data_per_2:]
z, p = ztest(df_beg, df_end, value=delta)

#calculate z_alpha/2
z_a = st.norm.ppf(1-alpha/2)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
plt.subplot(1,2,1)
df_beg.plot(kind="box", figsize=(12,6))
plt.title=("Sulfate lower")
plt.subplot(1,2,2)
df_end.plot(kind="box", figsize=(12,6))
plt.title=("Sulfate upper")
plt.show()
```

```
Nilai z: -2.0753
Nilai z_alpha/2: 1.96
Nilai p: 0.038
```



Nilai z lebih kecil dibandingkan $-z_{\alpha/2}\left(-2.0753<1.96\right)$
Nilai p lebih kecil dibandingkan $\alpha\ \left(0.038<0.05\right)$
Maka tolak H_0

Kesimpulan: rata-rata bagian awal dan akhir kolom sulfat ialah tidak sama

b) Data kolom OrganicCarbon dibagi sama rata: bagian awal dan bagian akhir kolom. Benarkah rata-rata bagian awal lebih besar daripada bagian akhir sebesar 0.15?

H_0: Nilai rata-rata kolom awal organic carbon sama dengan kolom akhir organic carbon sebesar ditambah 0.15 $\left(\mu_1-\mu_2=0.15\right)$

H_1: Nilai rata-rata kolom awal organic carbon tidak sama dengan nilai rata-rata kolom akhir organic carbon ditambah sebesar 0.15 $\left(\mu_1-\mu_2\neq0.15\right)$

Tingkat signifikan $\alpha=0.05$

Lakukan pengujian two tailed test karena akan dicek pada bagian kiri dengan $z<-z_{\alpha/2}$ serta bagian kanan dengan $z>z_{\alpha/2}$

Hitung nilai z:

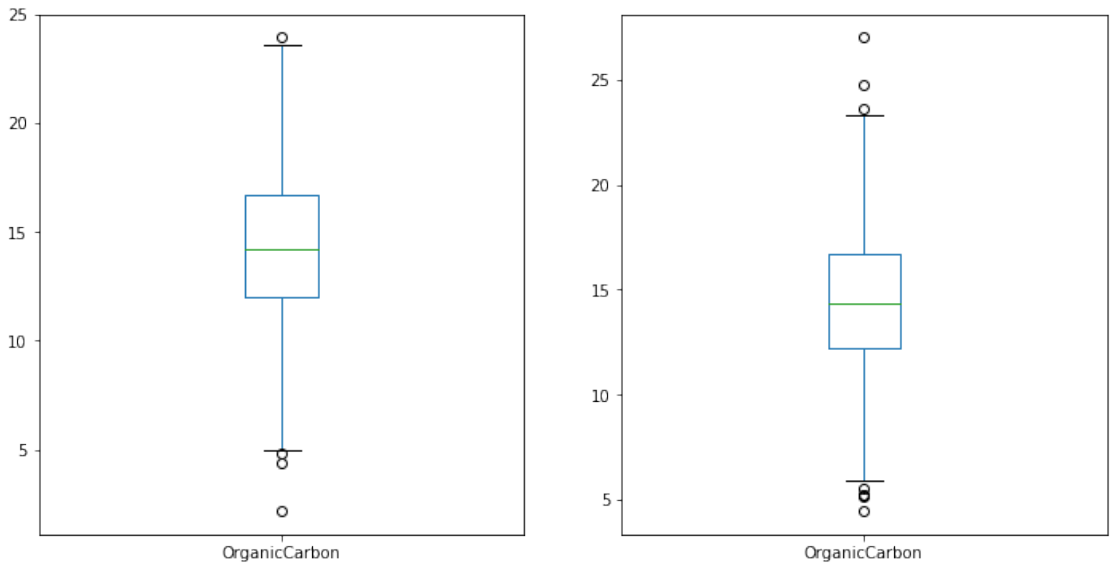
$$z=\frac{\left(\acute{x}_1-\acute{x}_2\right)-\left(\mu_1-\mu_2\right)}{\sqrt{\sigma_1^2/n_1+\sigma_2^2/n_2}}$$

Tolak H_0 jika ¿ atau $z<-z_{\alpha/2}$¿ dan $p<\alpha$
Terima H_0 jika $-z_{\alpha/2}\leq z\leq z_{\alpha/2}$ dan $p\geq\alpha$

```
delta = 0.15
alpha = 0.05

#calculate z and p using ztest module from beginning and end
```

```
len_data_per_2 = len(df) // 2
df_beg = df["OrganicCarbon"][:len_data_per_2]
df_end = df["OrganicCarbon"][len_data_per_2 :]
z, p = ztest(df_beg, df_end, value=delta)

#calculate z_alpha/2
z_a = st.norm.ppf(1-alpha/2)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
plt.subplot(1, 2, 1)
df_beg.plot(kind="box", figsize=(12, 6))
plt.subplot(1, 2, 2)
df_end.plot(kind="box", figsize=(12, 6))
plt.show()
```

```
Nilai z: -2.4131
Nilai z_alpha/2: 1.96
Nilai p: 0.0158
```



Nilai z lebih kecil dibandingkan $-z_{\alpha/2}\left(-2.4131<1.96\right)$
Nilai p lebih kecil dibandingkan $\alpha\ \left(0.0158<0.05\right)$
Maka tolak H_0

Kesimpulan: bagian awal dan bagian akhir kolom organic carbon memiliki rata - rata bagian awal lebih besar daripada bagian akhir sebesar 0.15

c) Rata-rata 100 baris pertama kolom Chloramines sama dengan 100 baris terakhirnya?

H_0: Nilai rata-rata 100 baris pertama kolom Chloramines sama dengan rata-rata 100 baris kolom akhir Chloramines $\left(\mu_1-\mu_2=0\right)$

H_1: Nilai rata-rata 100 baris pertama kolom Chloramines tidak sama dengan rata-rata 100 baris kolom akhir Chloramines $(\mu_1 - \mu_2 \neq 0)$

Tingkat signifikan $\alpha = 0.05$

Lakukan pengujian two tailed test karena akan dicek pada bagian kiri dengan $z < -z_{\alpha/2}$ serta bagian kanan dengan $z > z_{\alpha/2}$

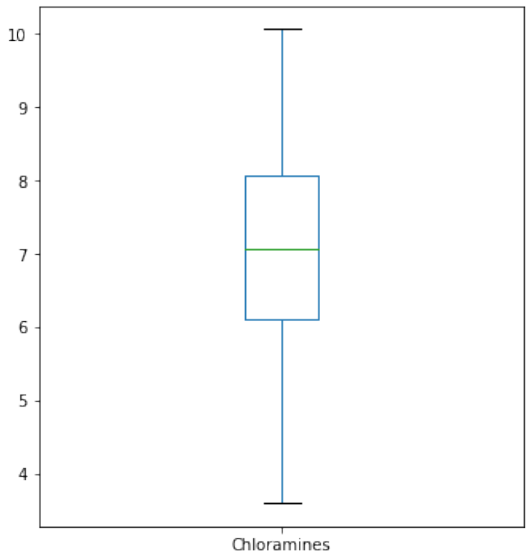
Hitung nilai z:

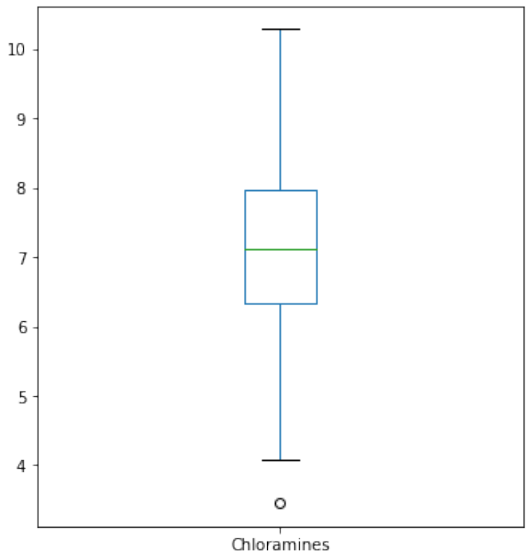
$$z = \frac{(\acute{x}_1 - \acute{x}_2) - (\mu_1 - \mu_2)}{\sqrt{\sigma_1^2/n_1 + \sigma_2^2/n_2}}$$

Tolak H_0 jika ¿ atau $z < -z_{\alpha/2}$ ¿ dan $p < \alpha$
Terima H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$ dan $p \geq \alpha$

```
delta = 0
alpha = 0.05

#calculate z and p using ztest module from beginning and end
df_beg = df["Chloramines"].head(100)
df_end = df["Chloramines"].tail(100)
z, p = ztest(df_beg,df_end, value=delta)

#calculate z_alpha/2
z_a = st.norm.ppf(1-alpha/2)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
plt.subplot(1, 2, 1)
df_beg.plot(kind="box", figsize=(12, 6))
plt.subplot(1, 2, 2)
df_end.plot(kind="box", figsize=(12, 6))
plt.show()
```

```
Nilai z: -0.7059
Nilai z_alpha/2: 1.96
Nilai p: 0.4802
```





Nilai z berada di rentang dibandingkan $-z_{\alpha/2} \le z \le z_{\alpha/2} \left(-1.96 \le -0.7059 \le 1.96\right)$
Nilai p lebih besar dibandingkan $\alpha \left(0.4802 > 0.05\right)$
Maka terima H_0

Kesimpulan: Rata-rata 100 baris pertama kolom Chloramines sama dengan 100 baris terakhirnya

d) Proporsi nilai bagian awal Turbidity yang lebih dari 4 adalah lebih besar daripada proporsi nilai yang sama pada di bagian akhir Turbidity?

H_0: Proporsi nilai bagian awal Turbidity yang lebih dari 4 sama dengan proporsi nilai bagian akhir turbidity $\left(p_1 - p_2 = 0\right)$

H_1: Proporsi nilai bagian awal Turbidity yang lebih dari 4 lebih dari proporsi nilai bagian akhir $\left(p_1 > p_2\right)$

Tingkat signifikan $\alpha = 0.05$

Lakukan pengujian one tailed test karena akan dicek pada bagian kanan dengan $z > z_\alpha$

Hitung nilai z:

$$z = \frac{\widehat{p_1} - \widehat{p_2}}{\sqrt{\hat{p}\,\hat{q}\left(1/n_1 + 1/n_2\right)}}$$

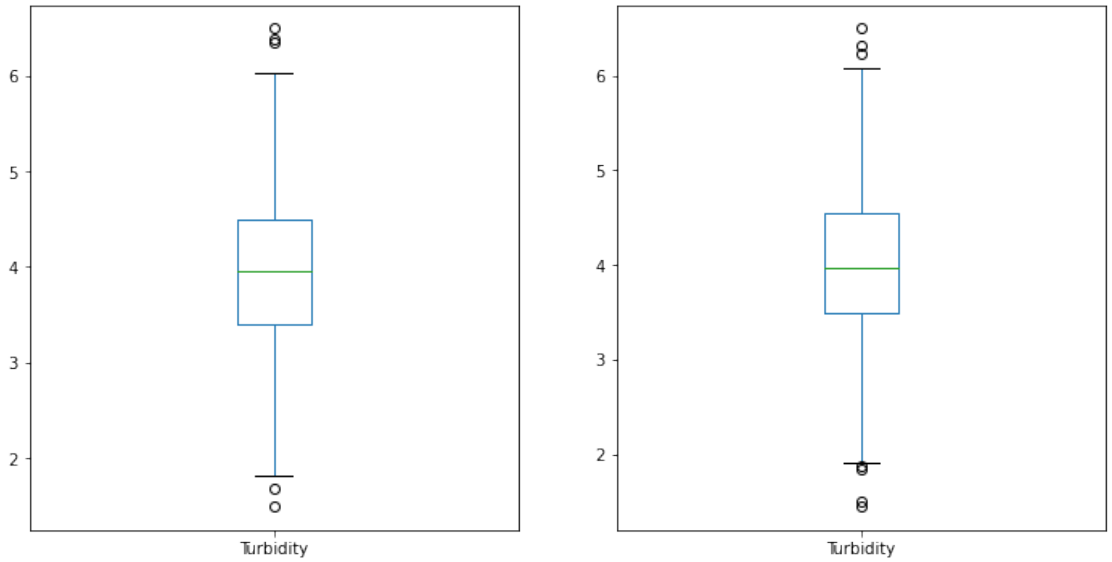
dengan nilai $\hat{p}$:

$$\hat{p} = \frac{x_1 + x_2}{n_1 + n_2}, \hat{q} = 1 - \hat{p}$$

Tolak H_0 jika $z > z_\alpha$ dan $p < \alpha$
Terima H_0 jika $z \le z_\alpha$ dan $p \ge \alpha$

```
delta = 0
alpha = 0.05

len_data_per_2 = len(df) // 2
df_beg = df[:len_data_per_2]
df_end = df[len_data_per_2:]

#calculate z and p using ztest module from beginning and end
z, p = proportions_ztest([len(df_beg[df_beg["Turbidity"] > 4]),
len(df_end[df_end["Turbidity"] > 4])], [len(df_beg), len(df_end)],
value=delta, prop_var=delta)

#calculate z_alpha/2
z_a = st.norm.ppf(1-alpha)

print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
plt.subplot(1, 2, 1)
df_beg["Turbidity"].plot(kind="box", figsize=(12, 6))
plt.subplot(1, 2, 2)
df_end["Turbidity"].plot(kind="box", figsize=(12, 6))
plt.show()
```

```
Nilai z: -0.1339
Nilai z_alpha: 1.6449
Nilai p: 0.8935
```



Nilai z kurang dari z_α $(-0.1339 \leq 1.6449)$
Nilai p lebih besar dibandingkan α $(0.8935 > 0.05)$
Maka terima H_0

Kesimpulan: Proporsi nilai bagian awal Turbidity yang lebih dari 4 sama dengan proporsi nilai bagian akhir turbidity

e) Bagian awal kolom Sulfate memiliki variansi yang sama pada bagian akhirnya?

H_0: Variansi bagian awal kolom Sulfate memiliki nilai yang sama dengan bagian akhir kolom Sulfate $\left(\sigma_1^2 = \sigma_2^2\right)$

H_1: Variansi bagian awal kolom Sulfate memiliki nilai yang berbeda dengan bagian akhir kolom Sulfate $\left(\sigma_1^2 \neq \sigma_2^2\right)$

Tingkat signifikan $\alpha = 0.05$

Lakukan pengujian two tailed f test karena akan dicek pada bagian kanan dengan $f > f_{\alpha/2}\left(v_1, v_2\right)$ dan bagian kiri pada $f < f_{1-\alpha/2}\left(v_1, v_2\right)$

Hitung nilai f:

$$f = \frac{s_1^2}{s_2^2}$$

dengan nilai v_1 dan v_2:

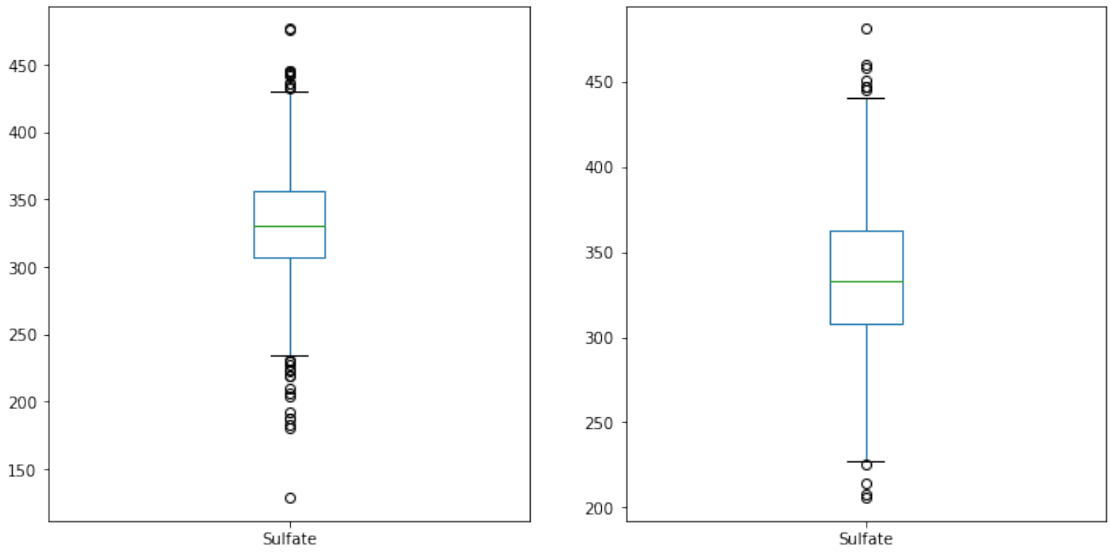
$$v_1 = n_1 - 1, v_2 = n_2 - 2$$

Tolak H_0 jika $f > f_{\alpha/2}\left(v_1, v_2\right)$ atau $f < f_{1-\alpha/2}\left(v_1, v_2\right)$ dan $p < \alpha$
Terima H_0 jika $f_{1-\alpha/2}\left(v_1, v_2\right) \leq f \leq f_{\alpha/2}\left(v_1, v_2\right)$ dan $p \geq \alpha$

```
alpha = 0.05
len_data_per_2 = len(df) // 2
df_beg = df["Sulfate"][:len_data_per_2]
df_end = df["Sulfate"][len_data_per_2:]
v1 = len(df_beg) - 1
v2 = len(df_end) - 1

#calculate f and p using scipy module from beginning and end
f = df_beg.var() / df_end.var()
f_up = st.f.ppf(1 - alpha/2, v1, v2)
f_down = st.f.ppf(alpha/2, v1, v2)

# p is the area from beginning to f
p = 1 - st.f.cdf(f, v1, v2)

print(f"Nilai f: {round(f, 4)}")
print(f"Nilai f_up: {round(f_up, 4)}")
print(f"Nilai f_down: {round(f_down, 4)}")
print(f"Nilai p: {round(p, 4)}")
plt.subplot(1, 2, 1)
df_beg.plot(kind="box", figsize=(12, 6))
```

```
plt.subplot(1, 2, 2)
df_end.plot(kind="box", figsize=(12, 6))
plt.show()
```

```
Nilai f: 1.0153
Nilai f_up: 1.1318
Nilai f_down: 0.8836
Nilai p: 0.4053
```



Nilai f berada pada range $f_{1-\alpha/2}(v_1, v_2) \leq f \leq f_{\alpha/2}(v_1, v_2)$ $(0.8836 < 1.0153 < 1.1318)$
Nilai p lebih besar dibandingkan α $(0.4053 > 0.05)$
Maka terima H_0

Kesimpulan: Variansi bagian awal kolom Sulfate memiliki nilai yang sama dengan bagian akhir kolom Sulfate

NO 6

Test korelasi: tentukan apakah setiap kolom non-target berkorelasi dengan kolom target, dengan menggambarkan juga scatter plot nya. Gunakan correlation test. Tes Korelasi yang digunakan adalah Pearson's Correlation Coefficient, mengukur hubungan linier antar variabel. Secara matematis jika (σXY) adalah kovarian antara X dan Y, dan (σX) adalah simpangan baku dari X, maka koefisien korelasi Pearson ρ adalah

```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as st
import numpy as np

col_names = ['id', 'pH', 'Hardness', 'Solids', 'Chloramines',
'Sulfate', 'Conductivity', 'OrganicCarbon', 'Trihalomethanes',
'Turbidity', 'Potability']
df = pd.read_csv('../data/water_potability.csv', names=col_names)

y = df['Potability']
```

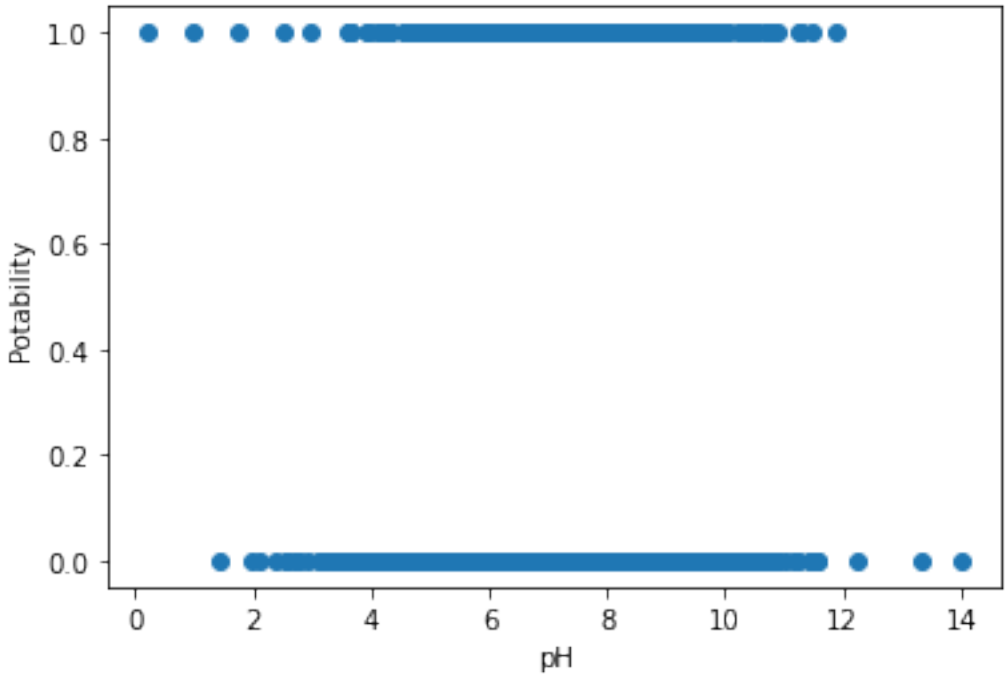
Kekuatan arah dan hubungan menurut Sarwono (2006). **Dengan kriteria:**

- 0 : **Tidak ada Korelasi**
- 0 - 2,5 : **Korelasi sangat lemah**
- 0,25 - 0,5 : **Korelasi cukup**
- 0,5 - 0,75 : **Korelasi kuat**
- 1 : **Korelasi Sempurna**

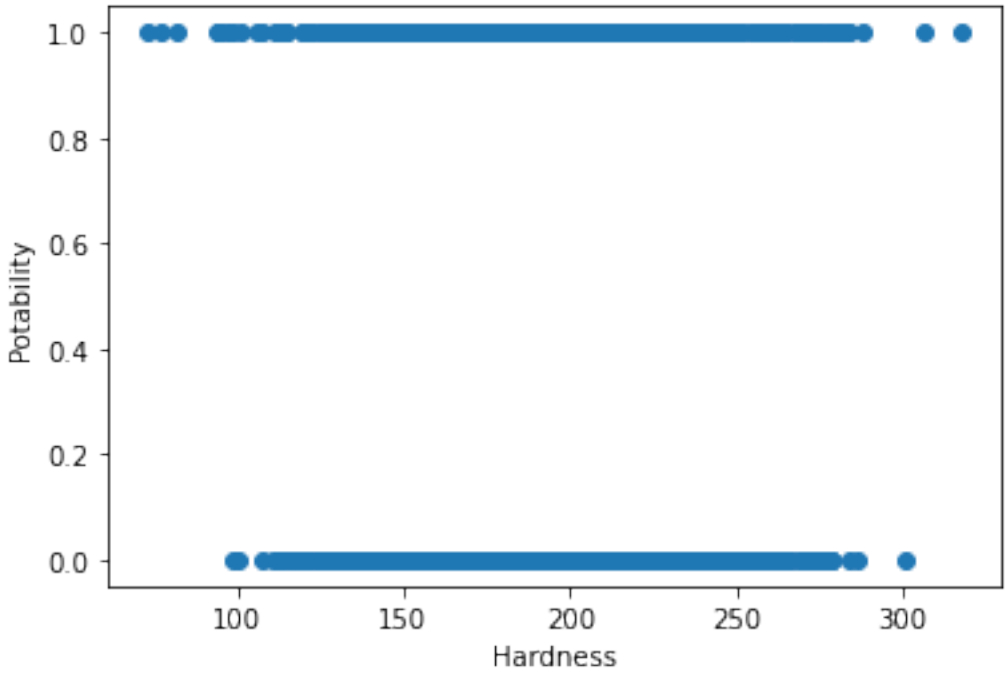
```
for i in range(1, 10):
    x1 = df[col_names[i]]
    corr = np.corrcoef(x1, y)
    print("Korelasi antara {} dan Potability: {}".format(col_names[i],
corr[0, 1]))
    if (corr[0, 1] == 0):
        print("Tidak terkorelasi")
    elif(corr[0, 1] > 0 and corr[0, 1] < 0.25):
        print("Korelasi sangat lemah")
    elif(corr[0, 1] >= 0.25 and corr[0, 1] < 0.5):
        print("Korelasi cukup")
    elif(corr[0, 1] >= 0.5 and corr[0, 1] < 0.75):
        print("Korelasi kuat")
    elif(corr[0, 1] >= 0.75 and corr[0, 1] < 1):
        print("Korelasi sangat kuat")
    elif(corr[0, 1] == 1):
        print ("Korelasi Sempurna")
    elif(corr[0, 1] < 0):
        print("Hubungan keduanya berbanding terbalik")
    plt.scatter(x1, y)
    plt.xlabel(col_names[i])
```

```
    plt.ylabel('Potability')
    plt.show()
```

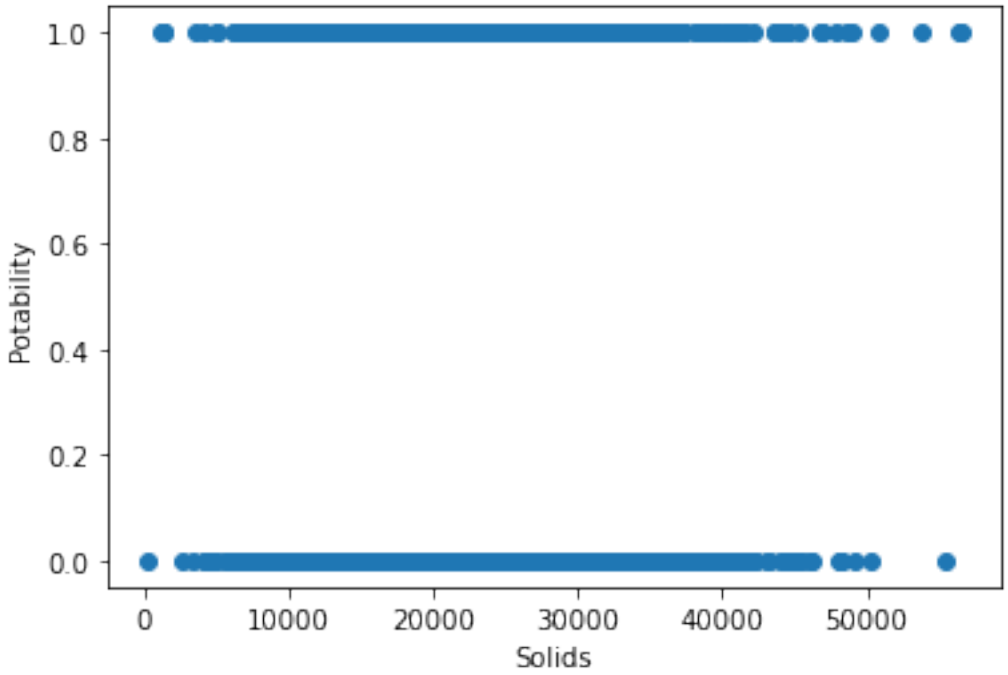
```
Korelasi antara pH dan Potability: 0.015475094408433481
Korelasi sangat lemah
```



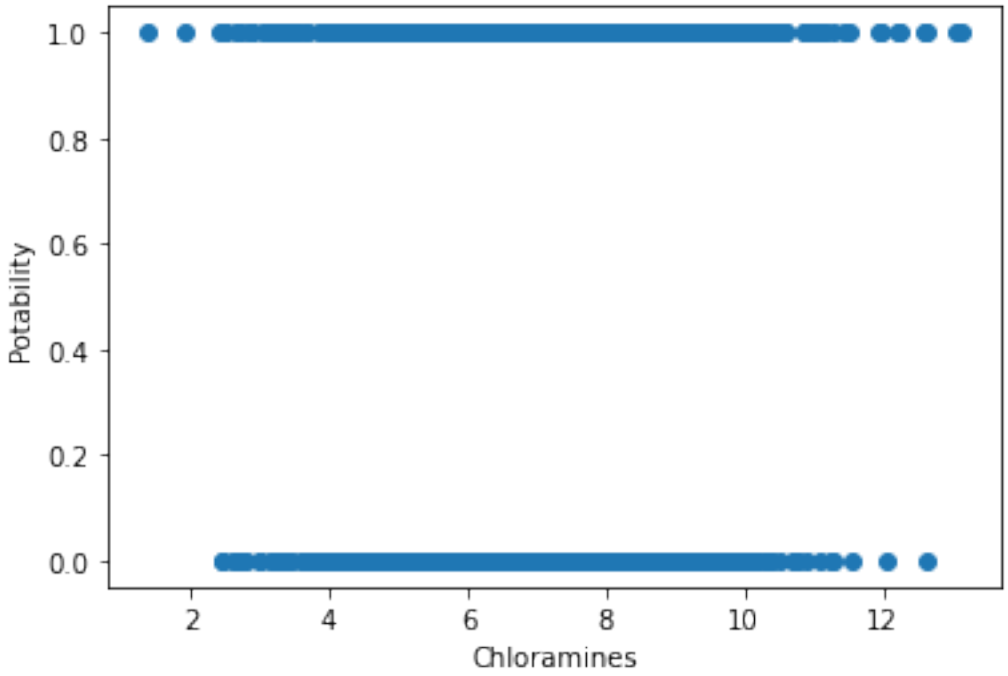

```
Korelasi antara Hardness dan Potability: -0.0014631528959479485
Hubungan keduanya berbanding terbalik
```



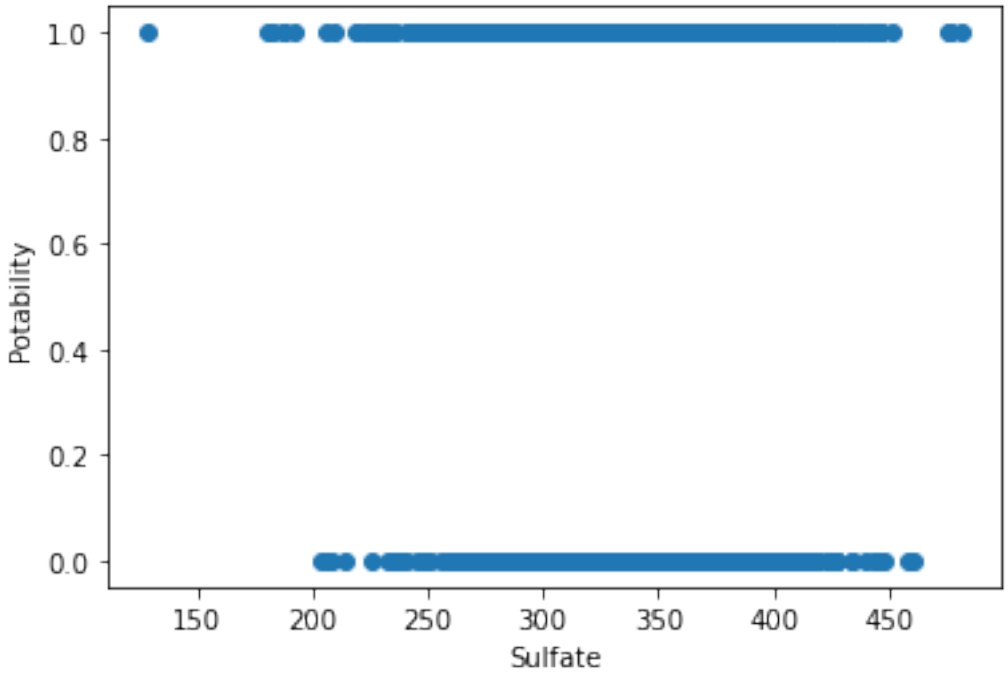
Korelasi antara Solids dan Potability: 0.038976578181734715
Korelasi sangat lemah



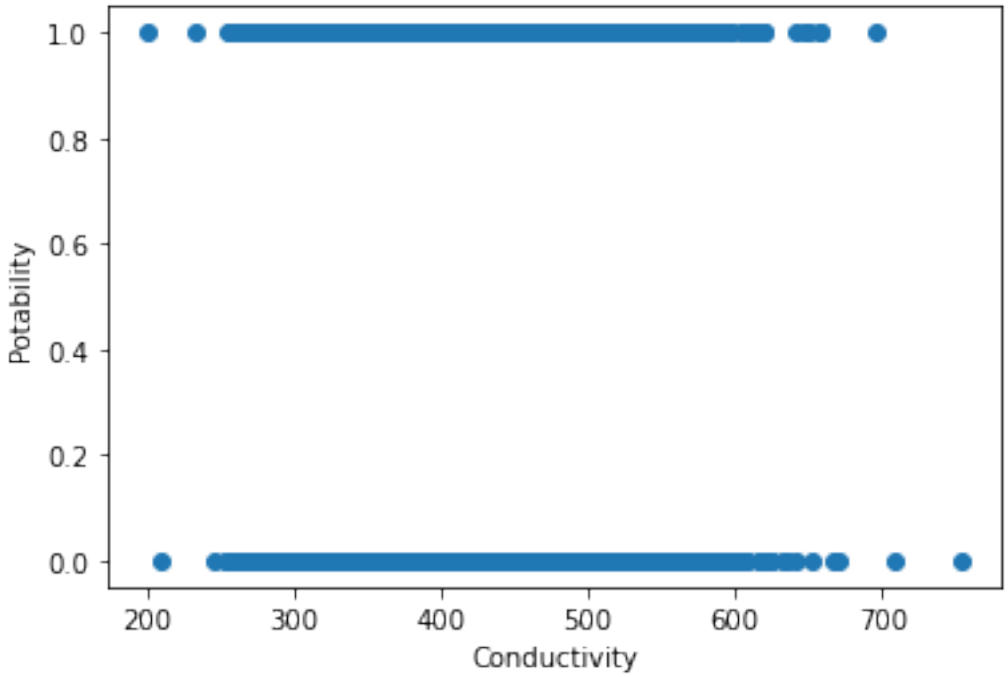
Korelasi antara Chloramines dan Potability: 0.020778921840524118
Korelasi sangat lemah



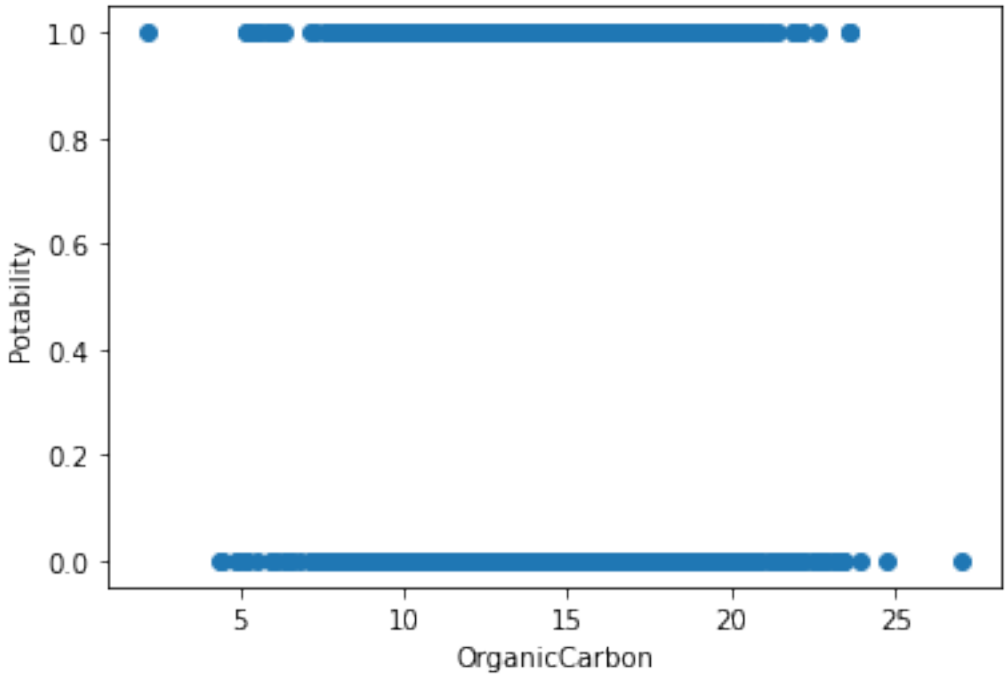
Korelasi antara Sulfate dan Potability: -0.015703164419273795
Hubungan keduanya berbanding terbalik



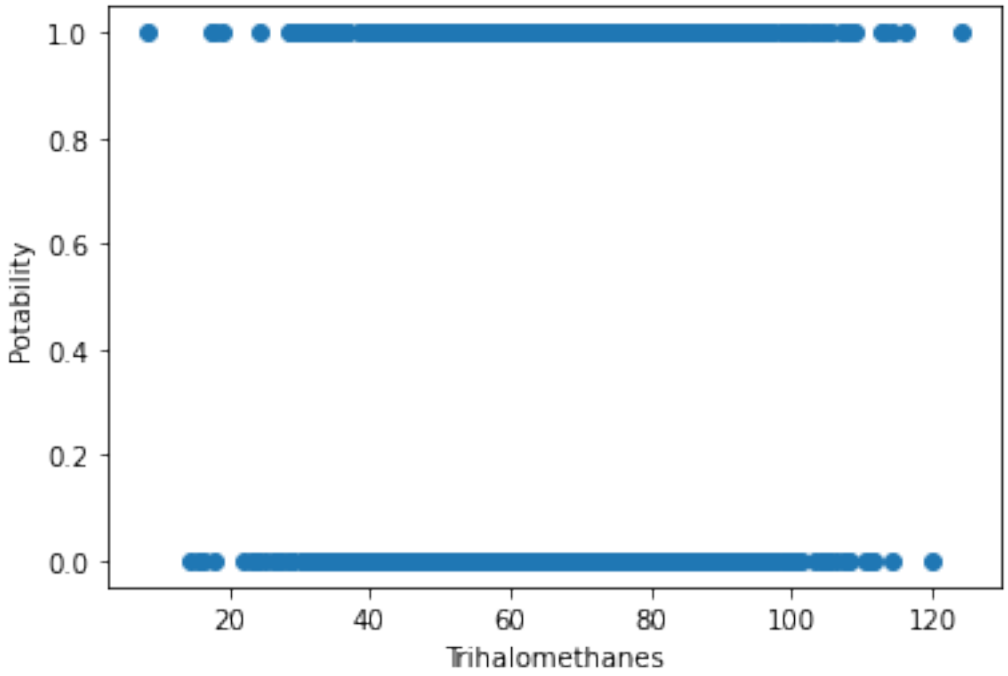
Korelasi antara Conductivity dan Potability: -0.016257120111377085
Hubungan keduanya berbanding terbalik



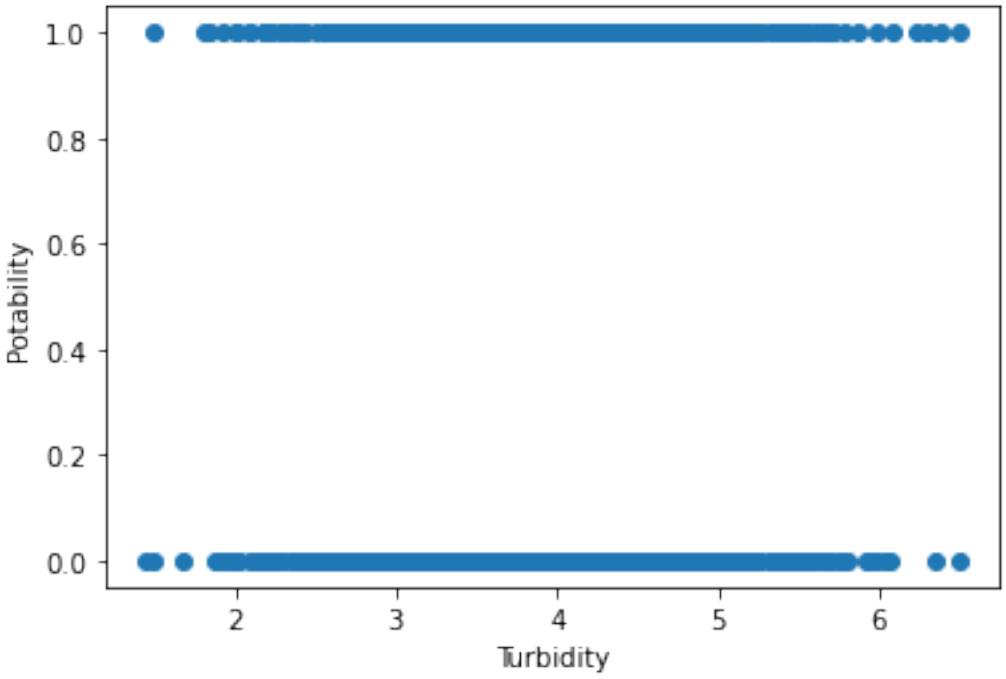
Korelasi antara OrganicCarbon dan Potability: -0.01548846191074729
Hubungan keduanya berbanding terbalik




```
Korelasi antara Trihalomethanes dan Potability: 0.009236711064713028
Korelasi sangat lemah
```




```
Korelasi antara Turbidity dan Potability: 0.022331042640622693
Korelasi sangat lemah
```


1.0
0.8
0.6
0.4
0.2
0.0
Potability
2
3
4
5
6
Turbidity