LAPORAN UJIAN AKHIR SEMESTER

Statistika Komputasi

Oleh:

Aldin Ariel Pramudya

2141720233

02

2G/D4 Teknik Informatika

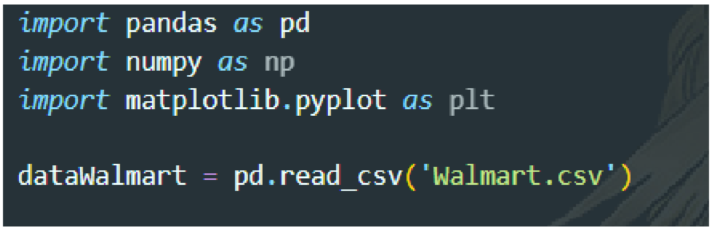
PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

0. Awal Ujian

   a. Melakukan import file .csv kedalam file python



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

dataWalmart = pd.read_csv('Walmart.csv')
```

1. Tentukan Deskripsi Data

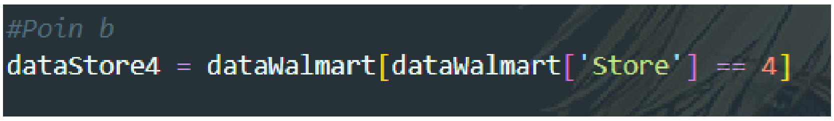
   a. Kolom yang dapat digunakan untuk permodelan regresi ?

   Kolom yang dapat dijadikan kolom regresi adalah :

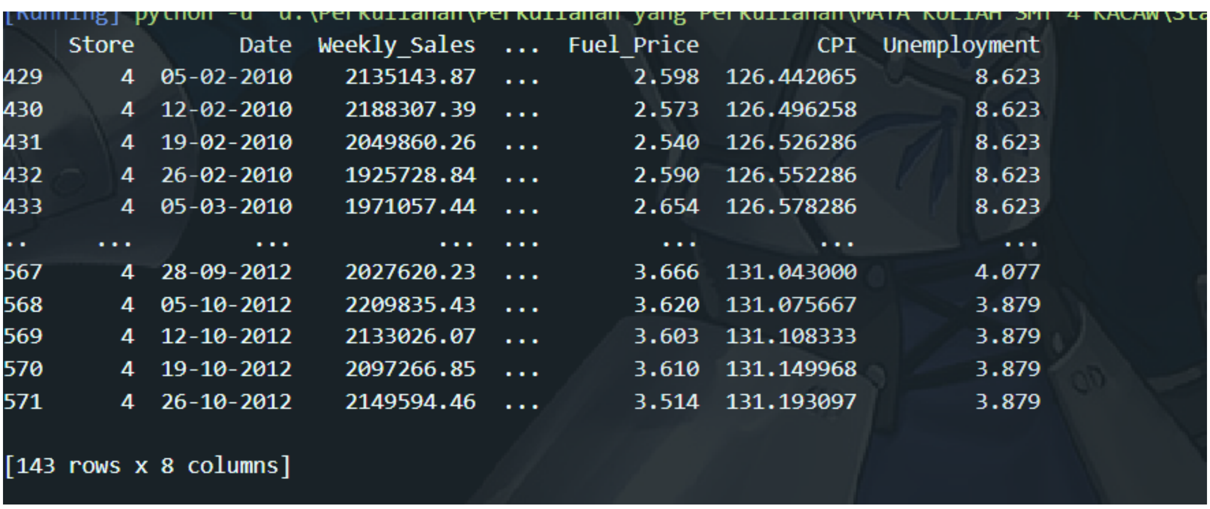
   - Weekly_Sales
   - Customer Price Index
   - Temperature
   - Fuel_Price
   - Unemployment
   - Holiday_Flag

   b. Nilai Mean, Median, Simpangan Baku, dan Varians dapat digunakan untuk regresi untuk store 4

   - Melakukan Seleksi Data untuk Store ke 4 saja



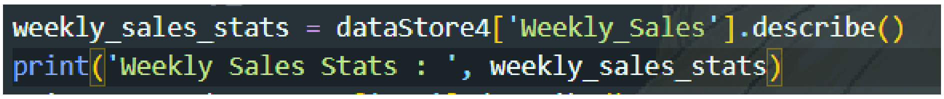
```
#Poin b
dataStore4 = dataWalmart[dataWalmart['Store'] == 4]
```



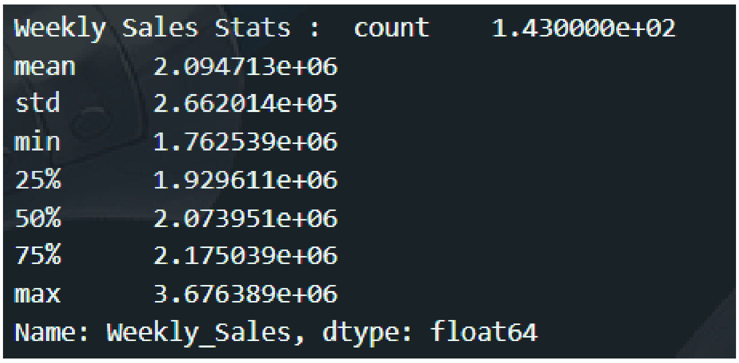
```
[Running] python -u "d:\Perkuliahan\Perkuliahan yang Perkuliahan\MATA KULIAH SMT 4 KACAW\Sta
     Store        Date  Weekly_Sales  ...  Fuel_Price         CPI  Unemployment
429      4  05-02-2010    2135143.87  ...       2.598  126.442065         8.623
430      4  12-02-2010    2188307.39  ...       2.573  126.496258         8.623
431      4  19-02-2010    2049860.26  ...       2.540  126.526286         8.623
432      4  26-02-2010    1925728.84  ...       2.590  126.552286         8.623
433      4  05-03-2010    1971057.44  ...       2.654  126.578286         8.623
..     ...         ...           ...  ...         ...         ...           ...
567      4  28-09-2012    2027620.23  ...       3.666  131.043000         4.077
568      4  05-10-2012    2209835.43  ...       3.620  131.075667         3.879
569      4  12-10-2012    2133026.07  ...       3.603  131.108333         3.879
570      4  19-10-2012    2097266.85  ...       3.610  131.149968         3.879
571      4  26-10-2012    2149594.46  ...       3.514  131.193097         3.879

[143 rows x 8 columns]
```

   - Nilai Mean, Median, Simpangan Baku, Varians Untuk Weekly_Sales

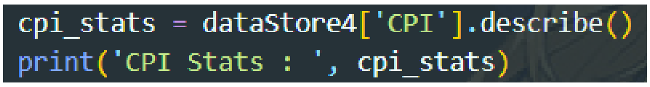


```
weekly_sales_stats = dataStore4['Weekly_Sales'].describe()
print('Weekly Sales Stats : ', weekly_sales_stats)
```

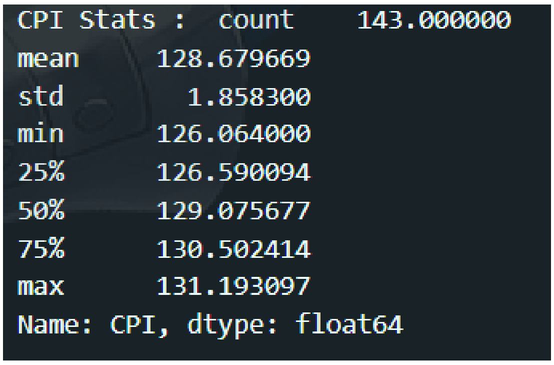


```
Weekly Sales Stats :  count    1.430000e+02
mean     2.094713e+06
std      2.662014e+05
min      1.762539e+06
25%      1.929611e+06
50%      2.073951e+06
75%      2.175039e+06
max      3.676389e+06
Name: Weekly_Sales, dtype: float64
```

- Nilai Mean, Median, Simpangan Baku, Varians untuk CPI

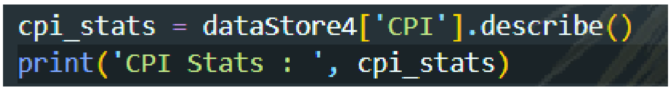


```
cpi_stats = dataStore4['CPI'].describe()
print('CPI Stats : ', cpi_stats)
```

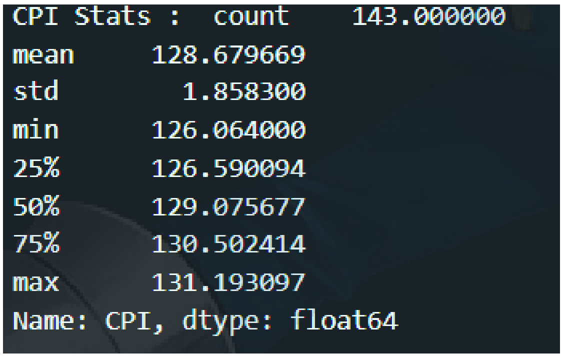


```
CPI Stats :  count    143.000000
mean     128.679669
std        1.858300
min      126.064000
25%      126.590094
50%      129.075677
75%      130.502414
max      131.193097
Name: CPI, dtype: float64
```

- Nilai Mean, Median, Simpangan Baku, Varians untuk CPI

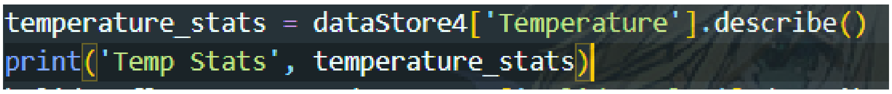


```
cpi_stats = dataStore4['CPI'].describe()
print('CPI Stats : ', cpi_stats)
```

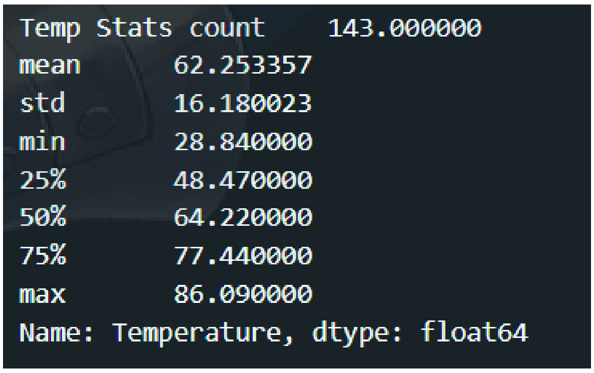


```
CPI Stats :  count    143.000000
mean     128.679669
std        1.858300
min      126.064000
25%      126.590094
50%      129.075677
75%      130.502414
max      131.193097
Name: CPI, dtype: float64
```

- Nilai Mean, Median, Simpangan Baku, Varians untuk Temperature

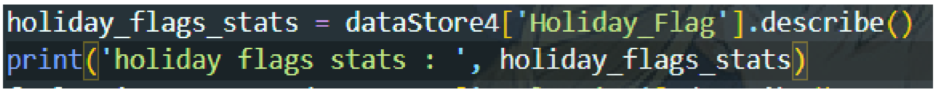


```
temperature_stats = dataStore4['Temperature'].describe()
print('Temp Stats', temperature_stats)
```

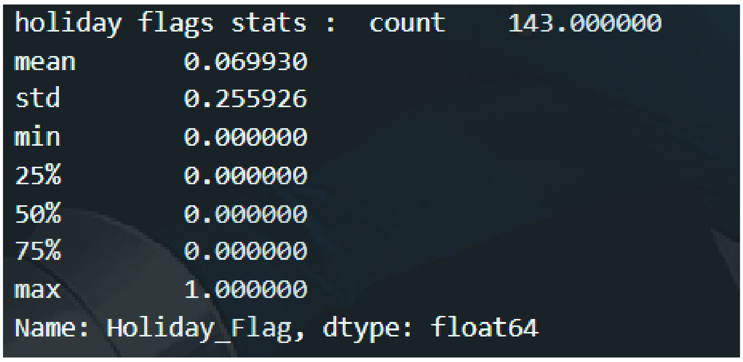


```
Temp Stats count    143.000000
mean         62.253357
std          16.180023
min          28.840000
25%          48.470000
50%          64.220000
75%          77.440000
max          86.090000
Name: Temperature, dtype: float64
```

- Nilai Mean, Median, Simpangan Baku, Varians untuk Holiday Flags

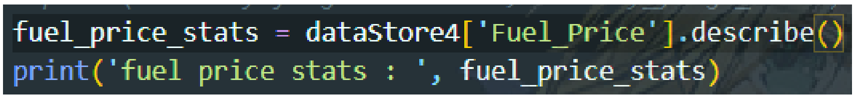


```
holiday_flags_stats = dataStore4['Holiday_Flag'].describe()
print('holiday flags stats : ', holiday_flags_stats)
```

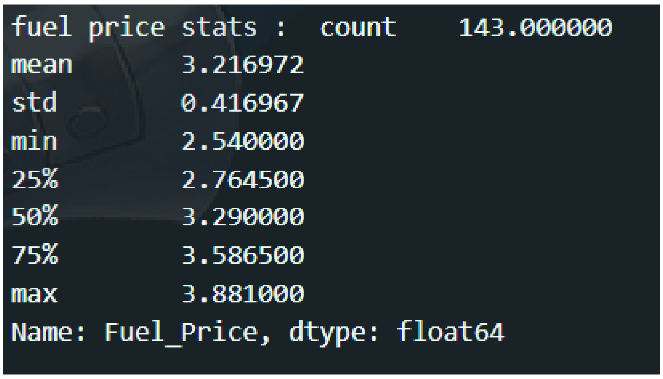


```
holiday flags stats :  count    143.000000
mean          0.069930
std           0.255926
min           0.000000
25%           0.000000
50%           0.000000
75%           0.000000
max           1.000000
Name: Holiday_Flag, dtype: float64
```

- Nilai Mean, Median, Simpangan Baku, Varians untuk Fuel Price

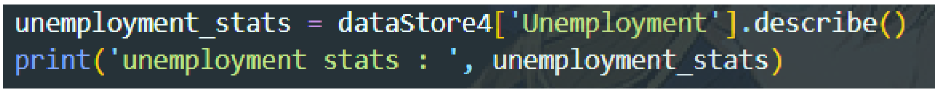


```
fuel_price_stats = dataStore4['Fuel_Price'].describe()
print('fuel price stats : ', fuel_price_stats)
```

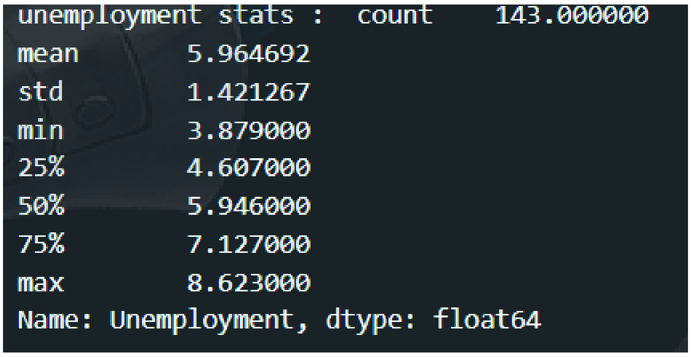


```
fuel price stats :  count    143.000000
mean          3.216972
std           0.416967
min           2.540000
25%           2.764500
50%           3.290000
75%           3.586500
max           3.881000
Name: Fuel_Price, dtype: float64
```

- Nilai Mean, Median, Simpangan Baku, Varians untuk Unemployment



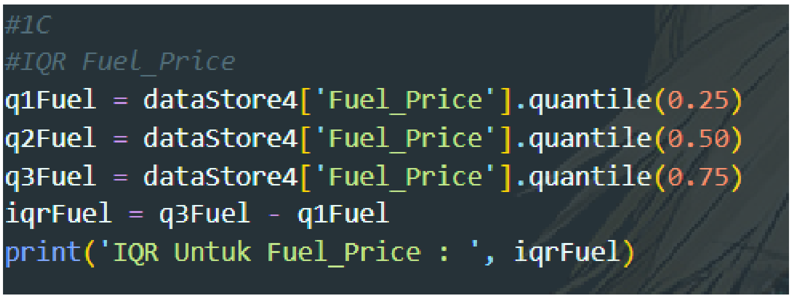
```
unemployment_stats = dataStore4['Unemployment'].describe()
print('unemployment stats : ', unemployment_stats)
```



```
unemployment stats :  count     143.000000
mean          5.964692
std           1.421267
min           3.879000
25%           4.607000
50%           5.946000
75%           7.127000
max           8.623000
Name: Unemployment, dtype: float64
```

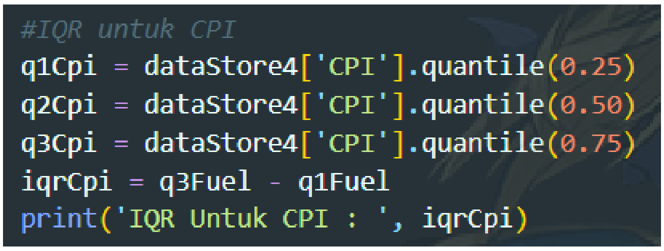
c. Menghitung IQR untuk Fuel_Price, CPI, Unemployment

- IQR Fuel Price



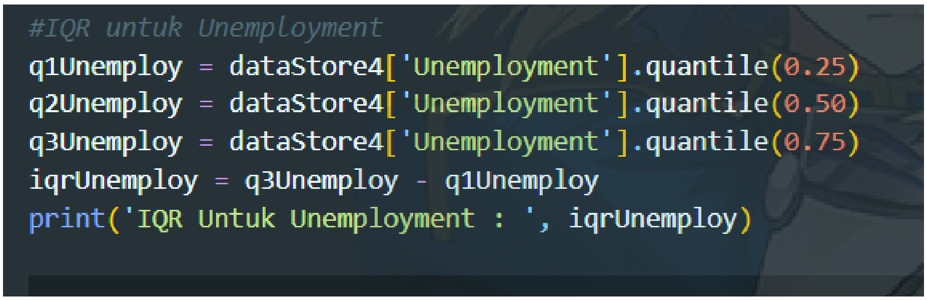
```
#1C
#IQR Fuel_Price
q1Fuel = dataStore4['Fuel_Price'].quantile(0.25)
q2Fuel = dataStore4['Fuel_Price'].quantile(0.50)
q3Fuel = dataStore4['Fuel_Price'].quantile(0.75)
iqrFuel = q3Fuel - q1Fuel
print('IQR Untuk Fuel_Price : ', iqrFuel)
```

- IQR CPI



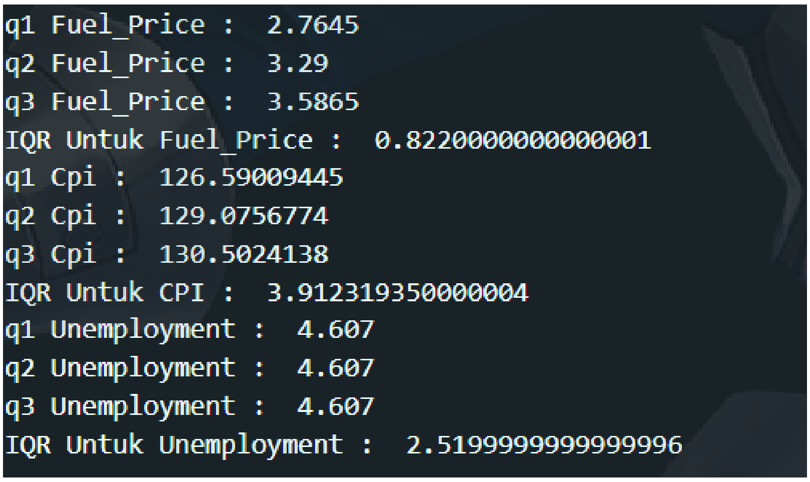
```
#IQR untuk CPI
q1Cpi = dataStore4['CPI'].quantile(0.25)
q2Cpi = dataStore4['CPI'].quantile(0.50)
q3Cpi = dataStore4['CPI'].quantile(0.75)
iqrCpi = q3Fuel - q1Fuel
print('IQR Untuk CPI : ', iqrCpi)
```

- IQR Unemployment



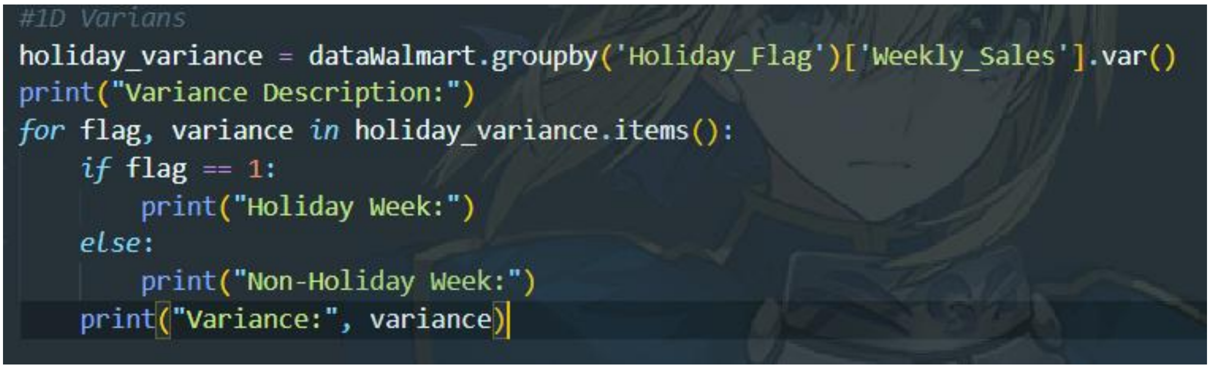
```
#IQR untuk Unemployment
q1Unemploy = dataStore4['Unemployment'].quantile(0.25)
q2Unemploy = dataStore4['Unemployment'].quantile(0.50)
q3Unemploy = dataStore4['Unemployment'].quantile(0.75)
iqrUnemploy = q3Unemploy - q1Unemploy
print('IQR Untuk Unemployment : ', iqrUnemploy)
```

- Hasil Penghitungan IQR,Q1,Q2,Q3

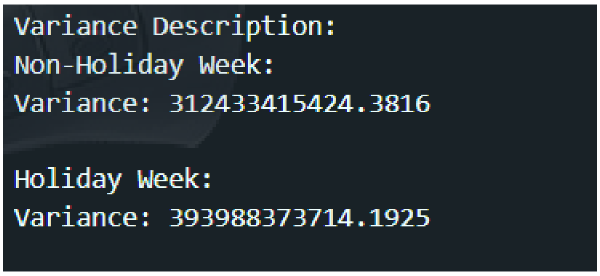


```
q1 Fuel_Price :  2.7645
q2 Fuel_Price :  3.29
q3 Fuel_Price :  3.5865
IQR Untuk Fuel_Price :  0.8220000000000001
q1 Cpi :  126.59009445
q2 Cpi :  129.0756774
q3 Cpi :  130.5024138
IQR Untuk CPI :  3.912319350000004
q1 Unemployment :  4.607
q2 Unemployment :  4.607
q3 Unemployment :  4.607
IQR Untuk Unemployment :  2.5199999999999996
```

d. Varians dari 1-Holiday Week dan 0-Non Holiady Week



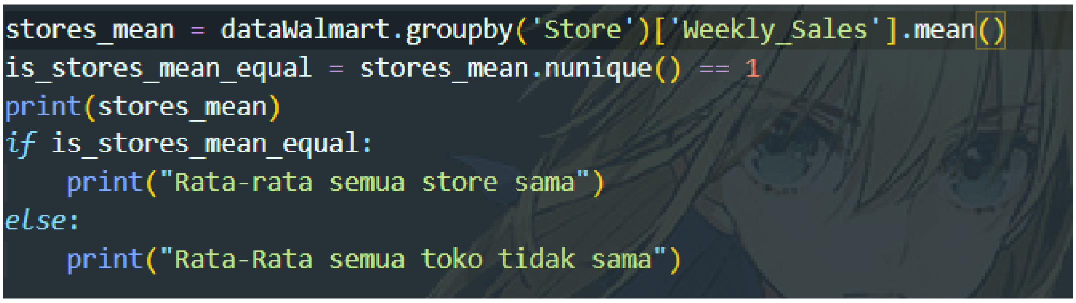
```
#1D Varians
holiday_variance = dataWalmart.groupby('Holiday_Flag')['Weekly_Sales'].var()
print("Variance Description:")
for flag, variance in holiday_variance.items():
    if flag == 1:
        print("Holiday Week:")
    else:
        print("Non-Holiday Week:")
    print("Variance:", variance)
```



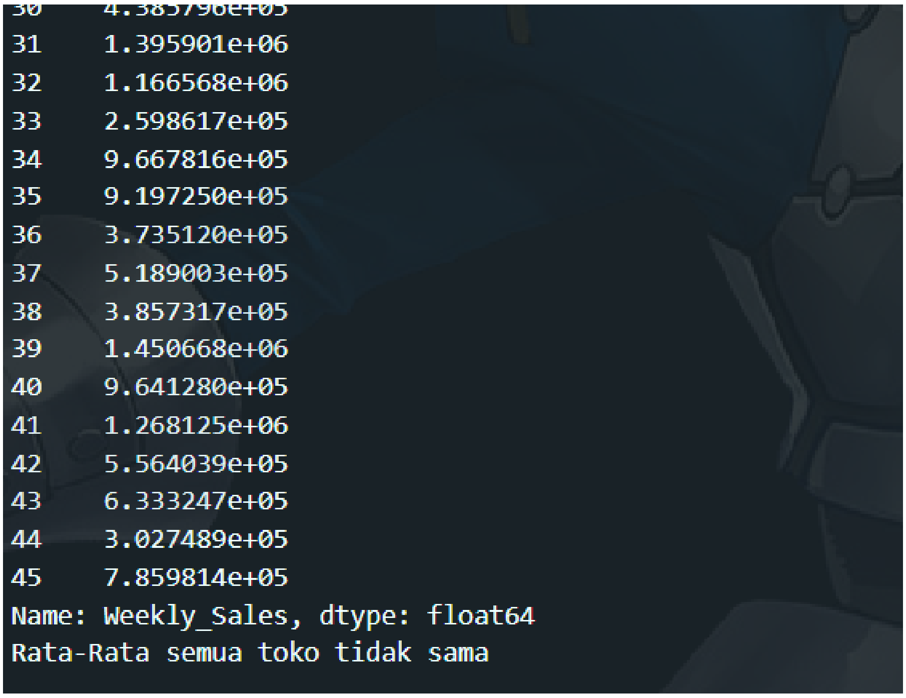
```
Variance Description:
Non-Holiday Week:
Variance: 312433415424.3816

Holiday Week:
Variance: 393988373714.1925
```

e. Apakah rata-rata weekly per store sama ?

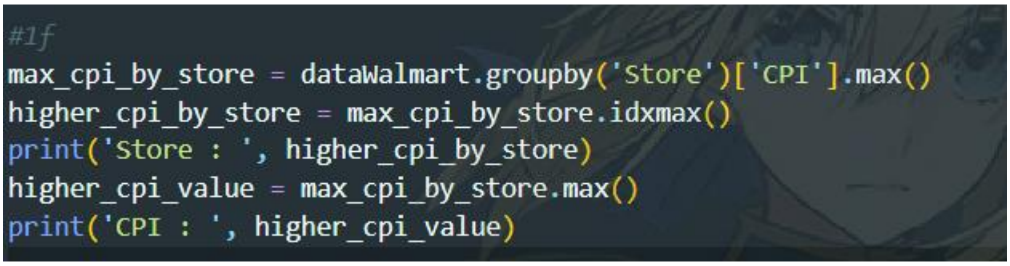


```
stores_mean = dataWalmart.groupby('Store')['Weekly_Sales'].mean()
is_stores_mean_equal = stores_mean.nunique() == 1
print(stores_mean)
if is_stores_mean_equal:
    print("Rata-rata semua store sama")
else:
    print("Rata-Rata semua toko tidak sama")
```

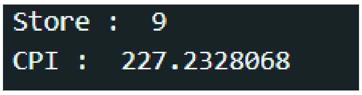


```
30      4.385796e+05
31      1.395901e+06
32      1.166568e+06
33      2.598617e+05
34      9.667816e+05
35      9.197250e+05
36      3.735120e+05
37      5.189003e+05
38      3.857317e+05
39      1.450668e+06
40      9.641280e+05
41      1.268125e+06
42      5.564039e+05
43      6.333247e+05
44      3.027489e+05
45      7.859814e+05
Name: Weekly_Sales, dtype: float64
Rata-Rata semua toko tidak sama
```

f. Dari setiap store manakah CPI yang paling tinggi

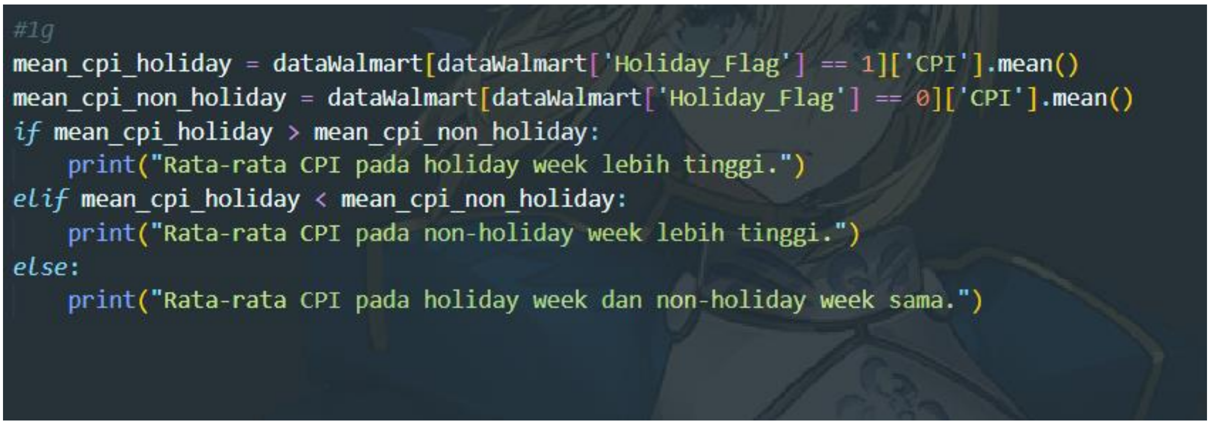


```
#1f
max_cpi_by_store = dataWalmart.groupby('Store')['CPI'].max()
higher_cpi_by_store = max_cpi_by_store.idxmax()
print('Store : ', higher_cpi_by_store)
higher_cpi_value = max_cpi_by_store.max()
print('CPI : ', higher_cpi_value)
```

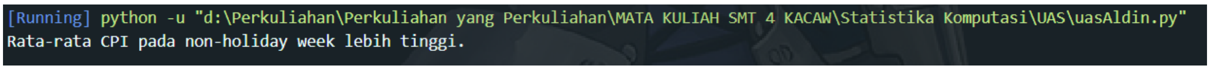


```
Store :  9
CPI :  227.2328068
```

g. CPI mana yang lebih tinggi, holiday week atau non holiday week



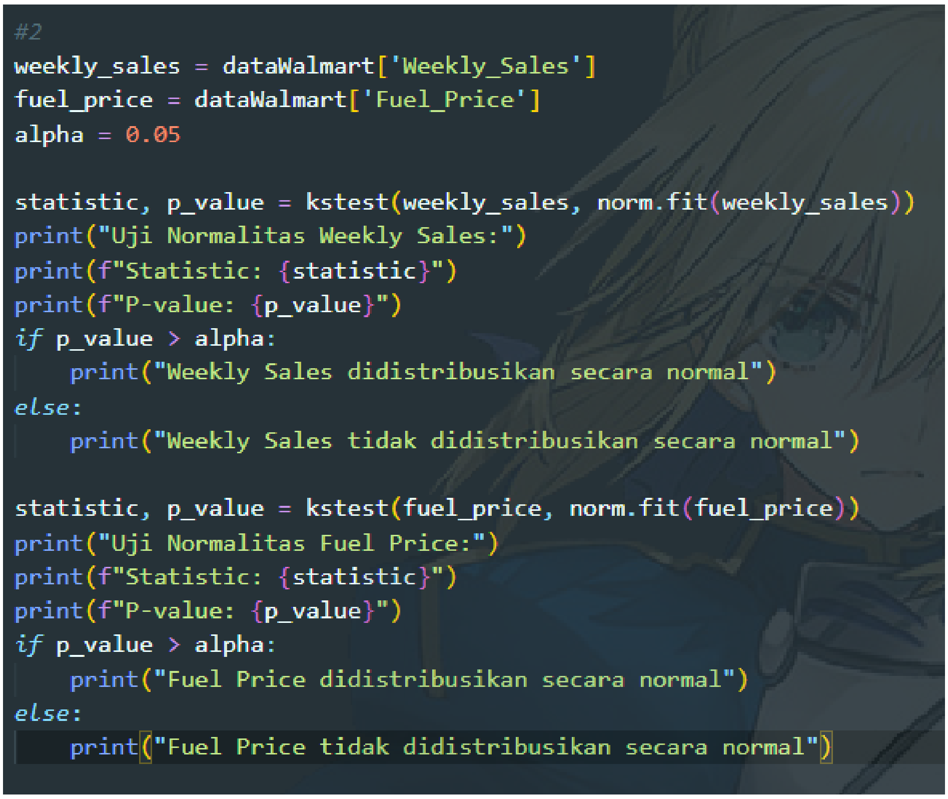
```
#1g
mean_cpi_holiday = dataWalmart[dataWalmart['Holiday_Flag'] == 1]['CPI'].mean()
mean_cpi_non_holiday = dataWalmart[dataWalmart['Holiday_Flag'] == 0]['CPI'].mean()
if mean_cpi_holiday > mean_cpi_non_holiday:
    print("Rata-rata CPI pada holiday week lebih tinggi.")
elif mean_cpi_holiday < mean_cpi_non_holiday:
    print("Rata-rata CPI pada non-holiday week lebih tinggi.")
else:
    print("Rata-rata CPI pada holiday week dan non-holiday week sama.")
```



```
[Running] python -u "d:\Perkuliahan\Perkuliahan yang Perkuliahan\MATA KULIAH SMT 4 KACAW\Statistika Komputasi\UAS\uasAldin.py"
Rata-rata CPI pada non-holiday week lebih tinggi.
```

2. Berdasarkan data, tentukan

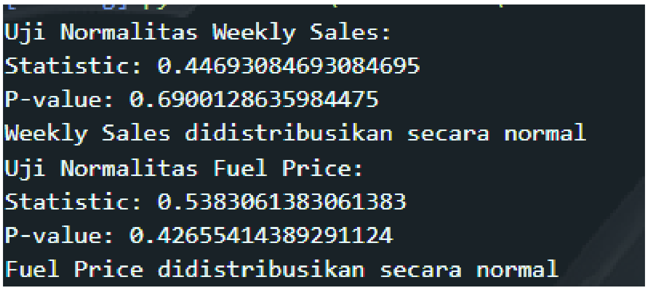
   a. Uji Normalitas dengan alpha = 0.05 pada Fuel Price dan Weekly Sales



```
#2
weekly_sales = dataWalmart['Weekly_Sales']
fuel_price = dataWalmart['Fuel_Price']
alpha = 0.05

statistic, p_value = kstest(weekly_sales, norm.fit(weekly_sales))
print("Uji Normalitas Weekly Sales:")
print(f"Statistic: {statistic}")
print(f"P-value: {p_value}")
if p_value > alpha:
    print("Weekly Sales didistribusikan secara normal")
else:
    print("Weekly Sales tidak didistribusikan secara normal")

statistic, p_value = kstest(fuel_price, norm.fit(fuel_price))
print("Uji Normalitas Fuel Price:")
print(f"Statistic: {statistic}")
print(f"P-value: {p_value}")
if p_value > alpha:
    print("Fuel Price didistribusikan secara normal")
else:
    print("Fuel Price tidak didistribusikan secara normal")
```



```
Uji Normalitas Weekly Sales:
Statistic: 0.44693084693084695
P-value: 0.6900128635984475
Weekly Sales didistribusikan secara normal
Uji Normalitas Fuel Price:
Statistic: 0.5383061383061383
P-value: 0.42655414389291124
Fuel Price didistribusikan secara normal
```

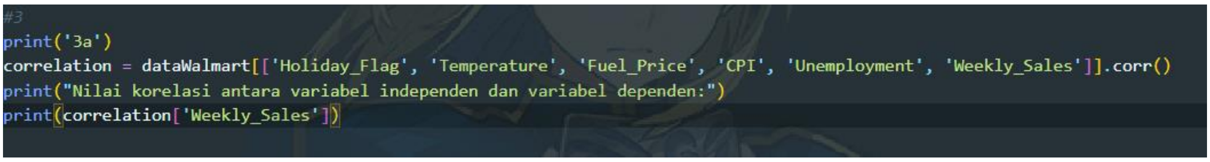
   b. Variabel Dependen adalah : Weekly_Sales

   Variabebel Independen adalah :

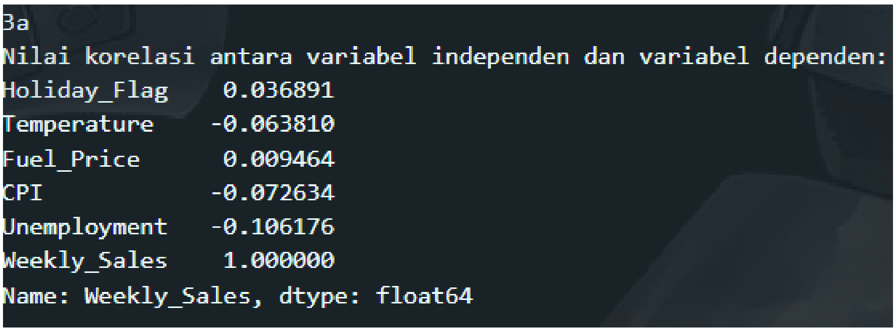
   - CPI
   - Temperature
   - Fuel_Price
   - Unemployment
   - Holiday_Flag

3. Berdasarkan Data, Tentukan :

   a. Lakukan Uji Korelasi masing masing variabel dependen dengna independent

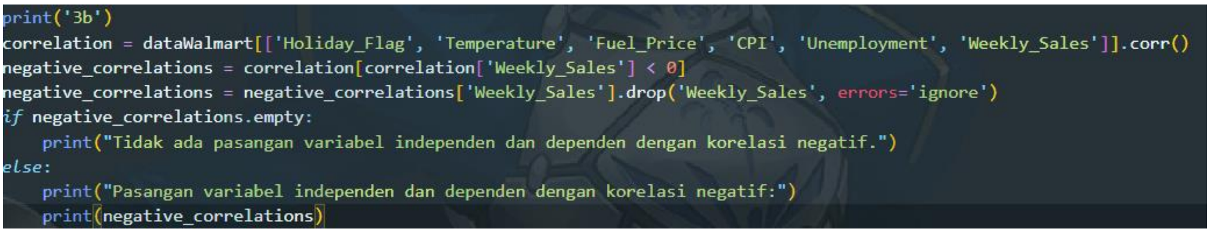


```
#3
print('3a')
correlation = dataWalmart[['Holiday_Flag', 'Temperature', 'Fuel_Price', 'CPI', 'Unemployment', 'Weekly_Sales']].corr()
print("Nilai korelasi antara variabel independen dan variabel dependen:")
print(correlation['Weekly_Sales'])
```

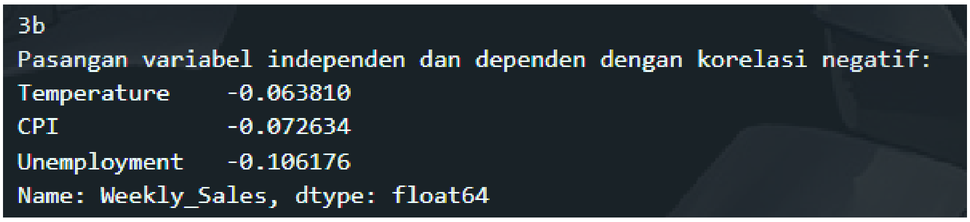


```
3a
Nilai korelasi antara variabel independen dan variabel dependen:
Holiday_Flag     0.036891
Temperature     -0.063810
Fuel_Price       0.009464
CPI             -0.072634
Unemployment    -0.106176
Weekly_Sales     1.000000
Name: Weekly_Sales, dtype: float64
```

b. Pasangan variabel independent dan dependen yang memiliki korelasi negative

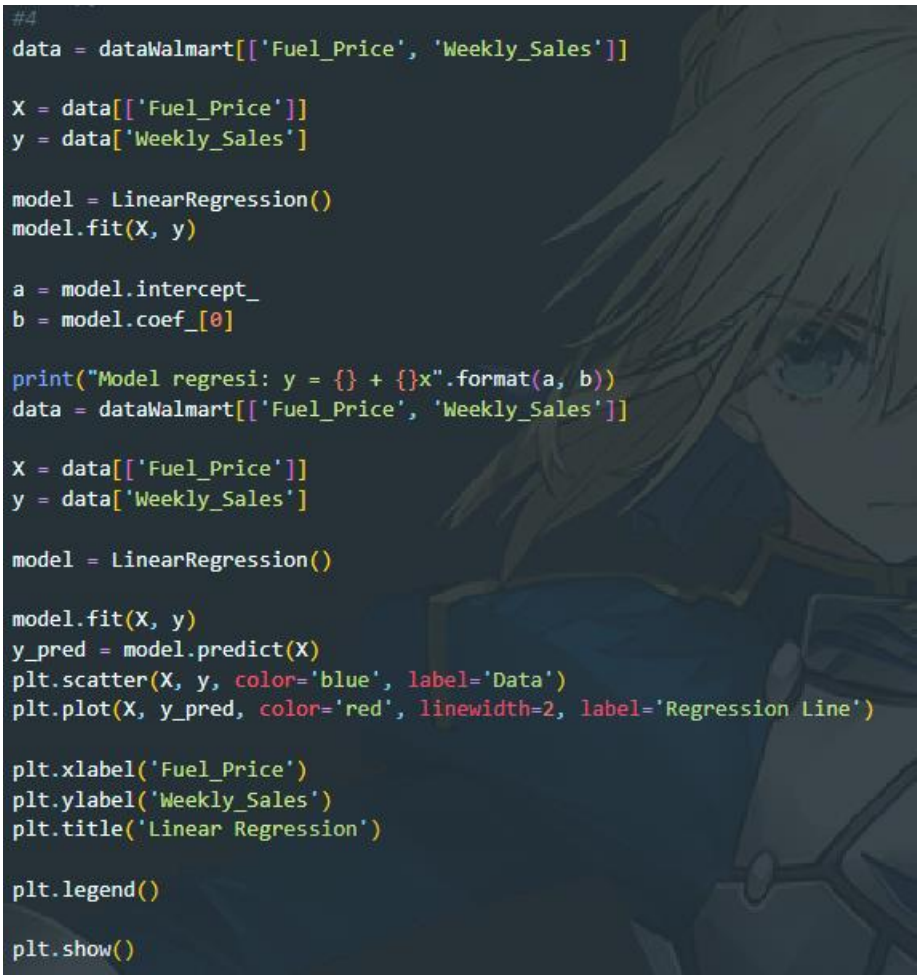


```
print('3b')
correlation = dataWalmart[['Holiday_Flag', 'Temperature', 'Fuel_Price', 'CPI', 'Unemployment', 'Weekly_Sales']].corr()
negative_correlations = correlation[correlation['Weekly_Sales'] < 0]
negative_correlations = negative_correlations['Weekly_Sales'].drop('Weekly_Sales', errors='ignore')
if negative_correlations.empty:
    print("Tidak ada pasangan variabel independen dan dependen dengan korelasi negatif.")
else:
    print("Pasangan variabel independen dan dependen dengan korelasi negatif:")
    print(negative_correlations)
```



```
3b
Pasangan variabel independen dan dependen dengan korelasi negatif:
Temperature     -0.063810
CPI             -0.072634
Unemployment    -0.106176
Name: Weekly_Sales, dtype: float64
```

4. Buatlah model regresi berdasarkan variabel independent ‘Fuel_Price’



```
#4
data = dataWalmart[['Fuel_Price', 'Weekly_Sales']]

X = data[['Fuel_Price']]
y = data['Weekly_Sales']

model = LinearRegression()
model.fit(X, y)

a = model.intercept_
b = model.coef_[0]

print("Model regresi: y = {} + {}x".format(a, b))
data = dataWalmart[['Fuel_Price', 'Weekly_Sales']]

X = data[['Fuel_Price']]
y = data['Weekly_Sales']

model = LinearRegression()

model.fit(X, y)
y_pred = model.predict(X)
plt.scatter(X, y, color='blue', label='Data')
plt.plot(X, y_pred, color='red', linewidth=2, label='Regression Line')

plt.xlabel('Fuel_Price')
plt.ylabel('Weekly_Sales')
plt.title('Linear Regression')

plt.legend()

plt.show()
```

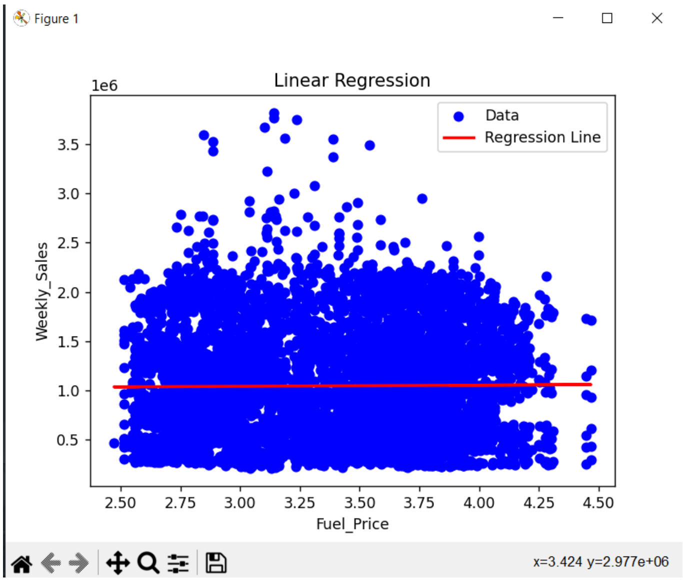


Figure 1
Linear Regression
1e6
Weekly_Sales
Fuel_Price
Data
Regression Line
x=3.424 y=2.977e+06