Nama : Rizqi Rohmatul Huda
Kelas : 2G-TI
Nomor Absen : 26
NIM : 2141720264

1. Tentukan deskripsi dari data
   a. Berdasarkan data yang diberikan, kolom mana yang dapat digunakan untuk permodelan regresi dan tidak

      Jawaban :

      - Dapat digunakan permodelan regresi
        - Weekly_Sales
        - Holiday_Flag
        - Temperature
        - Fuel_Price
        - CPI
        - Unemployment
      - Tidak dapat digunakan permodelan regresi
        - Store
        - Date

   b. Carilah nilai-nilai deskriptif mean, median, simpangan baku, varians pada kolom yang dapat digunakan untuk regresi sesuai nomor 1a hanya **untuk store**

      Jawaban :

```
print('1.b')
store_id = 4
df_filtered = df[df['Store'] == store_id]

weekly_sale_stats = df_filtered['Weekly_Sales'].describe()
holiday_flag_stats = df_filtered['Holiday_Flag'].describe()
temperature_stats = df_filtered['Temperature'].describe()
fuel_price_stats = df_filtered['Fuel_Price'].describe()
customer_price_index_stats = df_filtered['CPI'].describe()
unemployment_stats = df_filtered['Unemployment'].describe()

print("Statistics for Store ID = 4:")
print("Weekly Sale:")
print(weekly_sale_stats)

print("Holiday Flag:")
print(holiday_flag_stats)

print("Temperature:")
print(temperature_stats)

print("Fuel Price:")
print(fuel_price_stats)

print("Customer Price Index:")
print(customer_price_index_stats)

print("Unemployment:")
print(unemployment_stats)
```

```
PS E:\Local Disc A\semester4\statistika & komput
statistika & komputasi/uas_statistik/uas.py"
Nomor 1

1.b
Statistics for Store ID = 4:
Weekly Sale:
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
Holiday Flag:
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
Temperature:
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
Fuel Price:
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
Customer Price Index:
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
Unemployment:
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. Tentukan nilai dari QQ1,Q2,Q3 dan IQR untuk ‘Fuel_Price’, ‘CPI’, dan ‘Unemployment’ **HANYA UNTUK STORE**

Jawaban :

```
print('1.c')
fuel_price_q1 = df_filtered['Fuel_Price'].quantile(0.25)
fuel_price_q2 = df_filtered['Fuel_Price'].quantile(0.50)
fuel_price_q3 = df_filtered['Fuel_Price'].quantile(0.75)
fuel_price_iqr = fuel_price_q3 - fuel_price_q1

cpi_q1 = df_filtered['CPI'].quantile(0.25)
cpi_q2 = df_filtered['CPI'].quantile(0.50)
cpi_q3 = df_filtered['CPI'].quantile(0.75)
cpi_iqr = cpi_q3 - cpi_q1

unemployment_q1 = df_filtered['Unemployment'].quantile(0.25)
unemployment_q2 = df_filtered['Unemployment'].quantile(0.50)
unemployment_q3 = df_filtered['Unemployment'].quantile(0.75)
unemployment_iqr = unemployment_q3 - unemployment_q1

print("Statistics for Store ID = 4:")
print("Fuel Price:")
print("Q1:", fuel_price_q1)
print("Q2:", fuel_price_q2)
print("Q3:", fuel_price_q3)
print("IQR:", fuel_price_iqr)

print("Customer Price Index:")
print("Q1:", cpi_q1)
print("Q2:", cpi_q2)
print("Q3:", cpi_q3)
print("IQR:", cpi_iqr)

print("Unemployment:")
print("Q1:", unemployment_q1)
print("Q2:", unemployment_q2)
print("Q3:", unemployment_q3)
print("IQR:", unemployment_iqr)
```

```
1.c
Statistics for Store ID = 4:
Fuel Price:
Q1: 2.7645
Q2: 3.29
Q3: 3.5865
IQR: 0.8220000000000001
Customer Price Index:
Q1: 126.59009445
Q2: 129.0756774
Q3: 130.5024138
IQR: 3.912319350000004
Unemployment:
Q1: 4.607
Q2: 5.946
Q3: 7.127
IQR: 2.5199999999999996
```

d. Variansi dari 1-holiday week dan 0-non holiday week

Jawaban :

```
print('1d')
grouped_data = df.groupby('Holiday_Flag')['Weekly_Sales'].var()
print("Variance Description:")
for flag, variance in grouped_data.items():
    if flag == 1:
        print("Holiday Week:")
    else:
        print("Non-Holiday Week:")
    print("Variance:", variance)
    print()
```

```
1d
Variance Description:
Non-Holiday Week:
Variance: 312433415424.3816

Holiday Week:
Variance: 393988373714.1925
```

e. Apakah rata-rata weekly sales di setiap toko(store) sama?
Jawaban: Tidak, rata-rata Weekly Sales untuk setiap storenya adalah berbeda. Berikut adalah datanya

```
print('1e')
average_sales_by_store = df.groupby('Store')['Weekly_Sales'].mean()
is_average_sales_equal = average_sales_by_store.nunique() == 1
if is_average_sales_equal:
    print("Rata-rata Weekly Sales di setiap toko sama.")
else:
    print("Rata-rata Weekly Sales di setiap toko berbeda.")
```

```
1e
Rata-rata Weekly Sales di setiap toko adalah berbeda.
```

f. Dari setiap toko(store), CPI mana yang lebih tinggi
Jawaban :

```
print('1f')
max_cpi_by_store = df.groupby('Store')['CPI'].max()
higher_cpi_by_store = max_cpi_by_store.idxmax()
higher_cpi_value = max_cpi_by_store.max()

print("CPI yang lebih tinggi di setiap toko:")
for store_id in max_cpi_by_store.index:
    cpi_value = max_cpi_by_store.loc[store_id]
    print("Store ID:", store_id)
    print("CPI:", cpi_value)
    print()
```

```
1f
CPI yang lebih tinggi di setiap toko:
Store ID: 1
CPI: 223.4442513
```

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

```
print('1g')
average_cpi_holiday = df[df['Holiday_Flag'] == 1]['CPI'].mean()
average_cpi_non_holiday = df[df['Holiday_Flag'] == 0]['CPI'].mean()
if average_cpi_holiday > average_cpi_non_holiday:
    print("Rata-rata CPI pada holiday week lebih tinggi.")
elif average_cpi_holiday < average_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.")
```

```
1g
Rata-rata CPI pada non-holiday week lebih tinggi.
```

2. Berdasarkan data, tentukan
   a. Lakukan uji normalitas pada ‘Weekly_Sales’ dan ‘Fuel_Price’ menggunakan uji KS dengan @=0.05 . Apakah kedua nilai tersebut berdasar dari distribusi normal? Kemukakan jawaban Anda dengan bukti hasil analisis

      Jawaban :

```
class Nomor2() :
    weekly_sales = df['Weekly_Sales']
    fuel_price = df['Fuel_Price']
    alpha = 0.05

    print('Nomor 2\n')

    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. Berdasarkan soal 1a, tentukan variable independent dan variable dependen
      - Variabel Independen:
        - Holiday Flag: suatu variabel yang digunakan untuk menunjukkan apakah saat ini adalah hari libur atau tidak
        - Temperature: variable untuk mengukur suhu.
        - Fuel Price: Variabel ini digunakan untuk mencatat atau memperoleh informasi terkini tentang harga bahan bakar yang dapat digunakan dalam perhitungan, pemantauan, atau tindakan lain yang berkaitan dengan harga bahan bakar.

- Customer Price Index: Variabel yang mengukur indeks harga konsumen.
- Unemployment: Variabel yang mengukur tingkat pengangguran.

➢ Variabel Dependen:
- Weekly Sale: Variabel target yang ingin diprediksi atau dijelaskan menggunakan variabel-variabel independen di atas

3. Berdasarkan data, tentukan
   a. Lakukan uji korelasi masing-masing variable indepent dengan variable dependen. Berapa nilai korelasinya? (Contoh pasangan korelasi:Weekly_Sales-CPI
   b. Apakah ada pasangan variable independent dan dependen yang memiliki korelasi negative? Jika ya, sebutkan pasangan variabelnya

Jawaban :

```
class Nomor3() :
    print()
    print('Nomer 3')
    print('3a')
    correlation = df[['Holiday_Flag', 'Temperature', 'Fuel_Price', 'CPI', 'Unemployment', 'Weekly_Sales']].corr()
    print("Nilai korelasi antara variabel independen dan variabel dependen:")
    print(correlation['Weekly_Sales'])

    print('3b')
    correlation = df[['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)
```

3a.

```
Nomer 3
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
```

3b.

```
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 variable independent ‘Fuel_Price’. Nyatakan model Anda dalam bentuk y=a+bx

Jawaban :

```
class Nomor4() :
    print()
    print('Nomer 4')
    data = df[['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 = df[['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()
```

```
Nomer 4
Model regresi: y = 1007884.9200801749 + 11635.764282979584x
```