Project 5 - Retail Analysis with Walmart Data

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:
```
data=pd.read_csv("Walmart_Store_sales.csv")
data.head() #Fetches default first 5 rows
```

Out[2]:

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployment
0	1	05-02-2010	1643690.90	0	42.31	2.572	211.096358	8.106
1	1	12-02-2010	1641957.44	1	38.51	2.548	211.242170	8.106
2	1	19-02-2010	1611968.17	0	39.93	2.514	211.289143	8.106
3	1	26-02-2010	1409727.59	0	46.63	2.561	211.319643	8.106
4	1	05-03-2010	1554806.68	0	46.50	2.625	211.350143	8.106

In [3]:
```
#basic Functions
data.isnull().sum() #Null value check
```

Out[3]:
```
Store           0
Date            0
Weekly_Sales    0
Holiday_Flag    0
Temperature     0
Fuel_Price      0
CPI             0
Unemployment    0
dtype: int64
```

In [4]:
```
data.shape
```

Out[4]:
```
(6435, 8)
```

In [5]:
```
data.describe()
```

Out[5]:

	Store	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unempl
count	6435.000000	6.435000e+03	6435.000000	6435.000000	6435.000000	6435.000000	6435
mean	23.000000	1.046965e+06	0.069930	60.663782	3.358607	171.578394	7
std	12.988182	5.643666e+05	0.255049	18.444933	0.459020	39.356712	1
min	1.000000	2.099862e+05	0.000000	-2.060000	2.472000	126.064000	3
25%	12.000000	5.533501e+05	0.000000	47.460000	2.933000	131.735000	6
50%	23.000000	9.607460e+05	0.000000	62.670000	3.445000	182.616521	7
75%	34.000000	1.420159e+06	0.000000	74.940000	3.735000	212.743293	8
max	45.000000	3.818686e+06	1.000000	100.140000	4.468000	227.232807	14

In [6]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6435 entries, 0 to 6434
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Store         6435 non-null   int64  
 1   Date          6435 non-null   object 
 2   Weekly_Sales  6435 non-null   float64
 3   Holiday_Flag  6435 non-null   int64  
 4   Temperature   6435 non-null   float64
 5   Fuel_Price    6435 non-null   float64
 6   CPI           6435 non-null   float64
 7   Unemployment  6435 non-null   float64
dtypes: float64(5), int64(2), object(1)
memory usage: 402.3+ KB
```

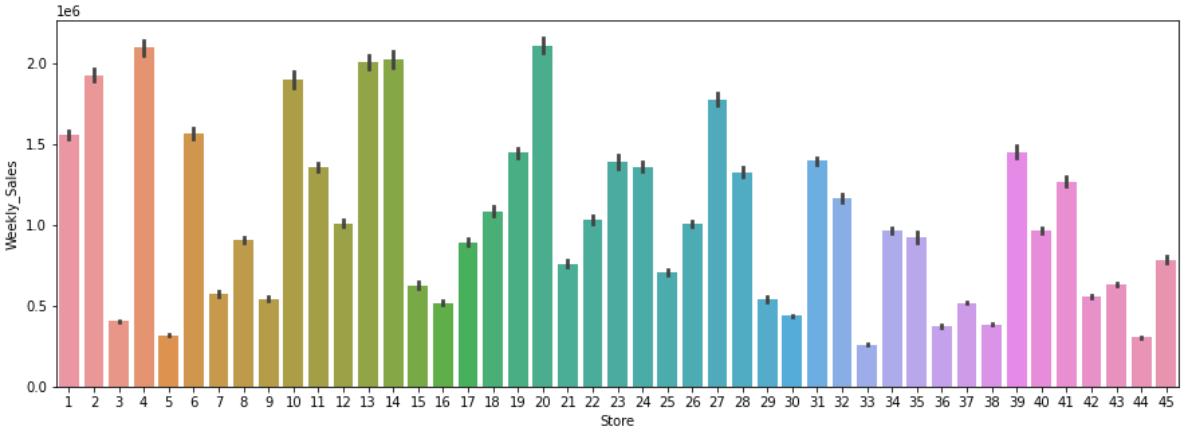
1. Which store has maximum sales ?

In [7]:
```
max_sales = data.groupby('Store')['Weekly_Sales'].sum()
max_sales.idxmax()
```

Out[7]:
```
20
```

In [8]:
```
#plotting the max sales in the Bar chart
plt.figure(figsize=(15,5))
sns.barplot(x=data.Store, y = data.Weekly_Sales)
```

Out[8]:
```
<AxesSubplot:xlabel='Store', ylabel='Weekly_Sales'>
```



Store 20 has maximum Sales

1. Which store has maximum standard deviation i.e., the sales vary a lot. Also, find out the coefficient of mean to standard deviation.

In [9]:
```
# maximum Standard deviation
max_std = data.groupby('Store')['Weekly_Sales'].std()
max_std.idxmax()
```

Out[9]:
```
14
```

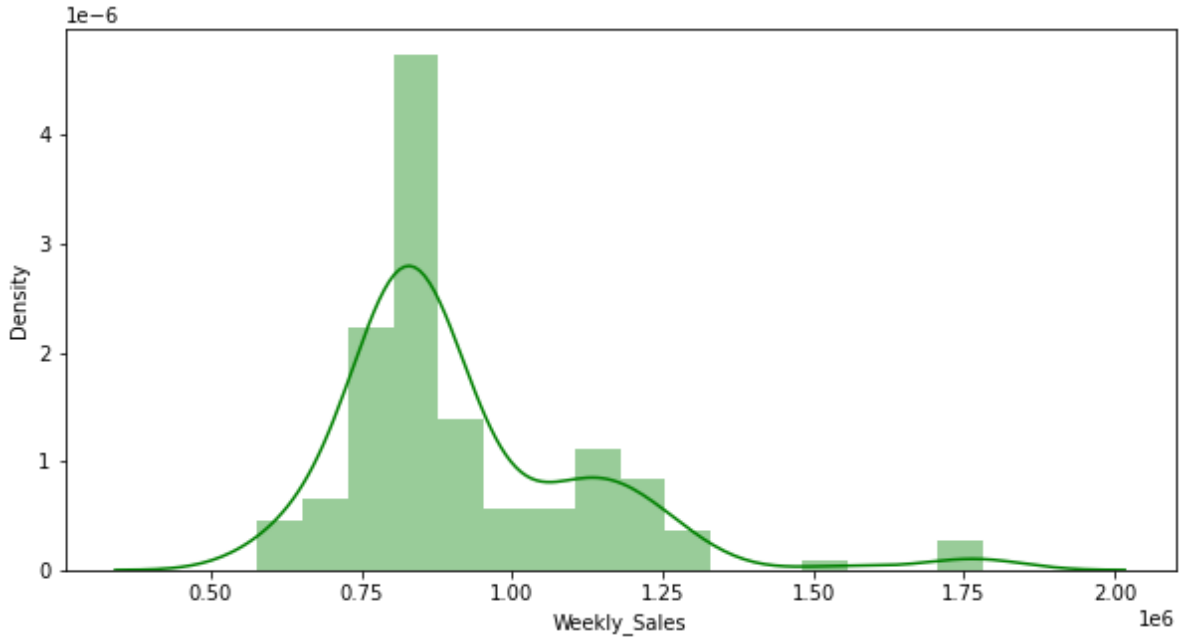
In [10]:
```
# maximum coefficient of variation
max_cov = ((data.groupby('Store')['Weekly_Sales'].std())/(data.groupby('Store')['W
max_cov.idxmax()
```

Out[10]:
```
35
```

In [11]:
```
#plotting the max sales in the Bar chart
stores = data.groupby('Store')
store_35 = stores.get_group(35)
plt.figure(figsize=(10,5))
sns.distplot(store_35.Weekly_Sales, color='green', label='Weekly Sales for Store 3
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a future
version. Please adapt your code to use either `displot` (a figure-level function w
ith similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

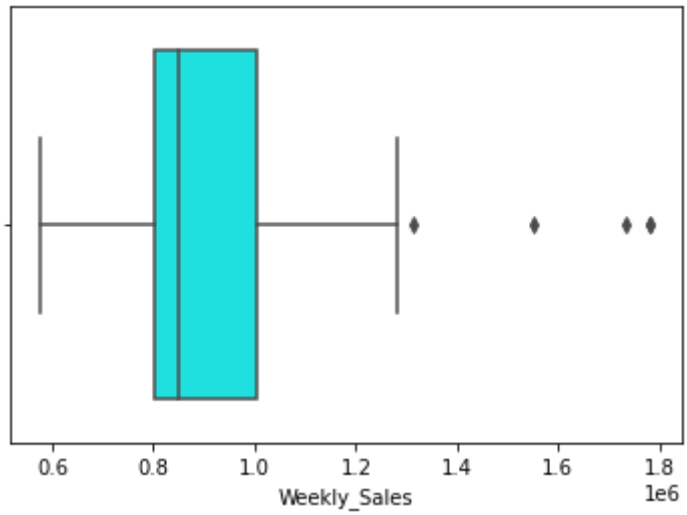
Out[11]:
```
<AxesSubplot:xlabel='Weekly_Sales', ylabel='Density'>
```



In [12]:
```
# Identify Outliers in weekly_sales for store 35
sns.boxplot(store_35.Weekly_Sales, color='cyan') #less outliers
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/seaborn/_decorators.py:36: Futu
reWarning: Pass the following variable as a keyword arg: x. From version 0.12, the
only valid positional argument will be `data`, and passing other arguments without
an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[12]: <AxesSubplot:xlabel='Weekly_Sales'>



1. Which store/s has good quarterly growth rate in Q3'2012 ?

In [13]:
```
# Grouping data by year and month
growth = data.copy()
growth['Date'] = pd.to_datetime(growth.Date,format='%d-%m-%Y')
growth['Year'] = growth['Date'].dt.year
growth['Month'] = growth['Date'].dt.month
growth
```

Out[13]:

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemploym
0	1	2010-02-05	1643690.90	0	42.31	2.572	211.096358	8
1	1	2010-02-12	1641957.44	1	38.51	2.548	211.242170	8
2	1	2010-02-19	1611968.17	0	39.93	2.514	211.289143	8
3	1	2010-02-26	1409727.59	0	46.63	2.561	211.319643	8
4	1	2010-03-05	1554806.68	0	46.50	2.625	211.350143	8
...	...	...	...	...	...	...	...	
6430	45	2012-09-28	713173.95	0	64.88	3.997	192.013558	8
6431	45	2012-10-05	733455.07	0	64.89	3.985	192.170412	8
6432	45	2012-10-12	734464.36	0	54.47	4.000	192.327265	8
6433	45	2012-10-19	718125.53	0	56.47	3.969	192.330854	8
6434	45	2012-10-26	760281.43	0	58.85	3.882	192.308899	8

6435 rows × 10 columns

In [14]:
```
# Group data with year = 2012
growth_rate = growth.groupby('Year')
growth_rate_2012 = growth_rate.get_group(2012)
growth_rate_2012.head()
```

Out[14]:

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployme
100	1	2012-01-06	1550369.92	0	49.01	3.157	219.714258	7.3
101	1	2012-01-13	1459601.17	0	48.53	3.261	219.892526	7.3
102	1	2012-01-20	1394393.84	0	54.11	3.268	219.985689	7.3
103	1	2012-01-27	1319325.59	0	54.26	3.290	220.078852	7.3
104	1	2012-02-03	1636339.65	0	56.55	3.360	220.172015	7.3

In [15]:
```
# Getting data for 4 quaters for year 2012

growth_rate_2012_Quaters = growth_rate_2012.groupby('Month')
growth_rate_2012_Q1_1 = growth_rate_2012_Quaters.get_group(1)
growth_rate_2012_Q1_2 = growth_rate_2012_Quaters.get_group(2)
growth_rate_2012_Q1_3 = growth_rate_2012_Quaters.get_group(3)

Quater_1 = growth_rate_2012_Q1_1.append(growth_rate_2012_Q1_2)
Quater_1 = Quater_1.append(growth_rate_2012_Q1_3) #Q1 data of 2012
display(Quater_1.head())

growth_rate_2012_Q2_4 = growth_rate_2012_Quaters.get_group(4)
growth_rate_2012_Q2_5 = growth_rate_2012_Quaters.get_group(5)
growth_rate_2012_Q2_6 = growth_rate_2012_Quaters.get_group(6)

Quater_2 = growth_rate_2012_Q2_4.append(growth_rate_2012_Q2_5)
Quater_2 = Quater_2.append(growth_rate_2012_Q2_6)  #Q2 data of 2012
display(Quater_2.head())

growth_rate_2012_Q3_7 = growth_rate_2012_Quaters.get_group(7)
growth_rate_2012_Q3_8 = growth_rate_2012_Quaters.get_group(8)
growth_rate_2012_Q3_9 = growth_rate_2012_Quaters.get_group(9)
Quater_3 = growth_rate_2012_Q3_7.append(growth_rate_2012_Q3_8)
Quater_3 = Quater_3.append(growth_rate_2012_Q3_9)  #Q3 data of 2012
display(Quater_3.head())

# Q4 data of 2012
growth_rate_2012_Q4_10 = growth_rate_2012_Quaters.get_group(10)
Quater_4 = growth_rate_2012_Q4_10
display(Quater_4.head())
```

```
/tmp/ipykernel_38219/3985971326.py:8: FutureWarning: The frame.append method is de
precated and will be removed from pandas in a future version. Use pandas.concat in
stead.
  Quater_1 = growth_rate_2012_Q1_1.append(growth_rate_2012_Q1_2)
/tmp/ipykernel_38219/3985971326.py:9: FutureWarning: The frame.append method is de
precated and will be removed from pandas in a future version. Use pandas.concat in
stead.
  Quater_1 = Quater_1.append(growth_rate_2012_Q1_3) #Q1 data of 2012
```

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployme
100	1	2012-01-06	1550369.92	0	49.01	3.157	219.714258	7.3
101	1	2012-01-13	1459601.17	0	48.53	3.261	219.892526	7.3
102	1	2012-01-20	1394393.84	0	54.11	3.268	219.985689	7.3
103	1	2012-01-27	1319325.59	0	54.26	3.290	220.078852	7.3
243	2	2012-01-06	1799520.14	0	46.75	3.157	219.355063	7.0

```
/tmp/ipykernel_38219/3985971326.py:16: FutureWarning: The frame.append method is d
eprecated and will be removed from pandas in a future version. Use pandas.concat i
nstead.
  Quater_2 = growth_rate_2012_Q2_4.append(growth_rate_2012_Q2_5)
/tmp/ipykernel_38219/3985971326.py:17: FutureWarning: The frame.append method is d
eprecated and will be removed from pandas in a future version. Use pandas.concat i
nstead.
  Quater_2 = Quater_2.append(growth_rate_2012_Q2_6)  #Q2 data of 2012
```

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployme
113	1	2012-04-06	1899676.88	0	70.43	3.891	221.435611	7.1
114	1	2012-04-13	1621031.70	0	69.07	3.891	221.510210	7.1
115	1	2012-04-20	1521577.87	0	66.76	3.877	221.564074	7.1
116	1	2012-04-27	1468928.37	0	67.23	3.814	221.617937	7.1
256	2	2012-04-06	2129035.91	0	68.43	3.891	221.073764	6.8

```
/tmp/ipykernel_38219/3985971326.py:23: FutureWarning: The frame.append method is d
eprecated and will be removed from pandas in a future version. Use pandas.concat i
nstead.
  Quater_3 = growth_rate_2012_Q3_7.append(growth_rate_2012_Q3_8)
/tmp/ipykernel_38219/3985971326.py:24: FutureWarning: The frame.append method is d
eprecated and will be removed from pandas in a future version. Use pandas.concat i
nstead.
  Quater_3 = Quater_3.append(growth_rate_2012_Q3_9)  #Q3 data of 2012
```

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployme
126	1	2012-07-06	1769854.16	0	81.57	3.227	221.883779	6.9
127	1	2012-07-13	1527014.04	0	77.12	3.256	221.924158	6.9
128	1	2012-07-20	1497954.76	0	80.42	3.311	221.932727	6.9
129	1	2012-07-27	1439123.71	0	82.66	3.407	221.941295	6.9
269	2	2012-07-06	2041507.40	0	84.20	3.227	221.521506	6.5

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployme
139	1	2012-10-05	1670785.97	0	68.55	3.617	223.181477	6.5
140	1	2012-10-12	1573072.81	0	62.99	3.601	223.381296	6.5
141	1	2012-10-19	1508068.77	0	67.97	3.594	223.425723	6.5
142	1	2012-10-26	1493659.74	0	69.16	3.506	223.444251	6.5
282	2	2012-10-05	1998321.04	0	70.27	3.617	222.815930	6.1

In [17]:
```
# Grouping the data "Store" wise each Quarter

df2 = pd.DataFrame(Quater_1.groupby('Store')['Weekly_Sales'].sum())

df2["Quater1_Sales"] = pd.DataFrame(Quater_1.groupby('Store')['Weekly_Sales'].sum()
df2["Quater2_Sales"] = pd.DataFrame(Quater_2.groupby('Store')['Weekly_Sales'].sum()
df2["Quater3_Sales"] = pd.DataFrame(Quater_3.groupby('Store')['Weekly_Sales'].sum()
df2["Quater4_Sales"] = pd.DataFrame(Quater_4.groupby('Store')['Weekly_Sales'].sum()
df2.drop('Weekly_Sales', axis = 1, inplace = True)
df2
```

Out[17]:

Store	Quater1_Sales	Quater2_Sales	Quater3_Sales	Quater4_Sales
1	20723762.83	20978760.12	20253947.78	6245587.29
2	24528220.70	25083604.88	24303354.86	7581514.93
3	5421809.72	5620316.49	5298005.47	1684307.82
4	27930310.30	28454363.67	27796792.46	8589722.81
5	4237380.83	4466363.69	4163790.99	1301302.62
6	19467939.96	20833909.92	20167312.24	5845884.88
7	7792647.21	7290859.27	8262787.39	2021262.60
8	11869407.28	11919630.95	11748952.70	3695929.20
9	7209983.86	7484935.11	7022149.56	2256961.05
10	24488944.65	23750369.17	23037258.76	6952044.36
11	17713050.31	17787371.95	17516081.44	5167561.98
12	13585746.35	13362388.58	12536324.37	3850386.42
13	25182790.59	27009207.14	26421259.30	8094197.99
14	24476492.09	25155535.41	21187560.65	6621810.11
15	7020781.97	7955243.07	7612081.03	2239424.64
16	6400479.72	6564335.98	7121541.64	2016067.98
17	11460215.01	12592400.93	12459453.05	3773309.64
18	13190110.09	13896194.65	13489765.27	4342506.79
19	17237532.28	18367300.24	18203554.85	5404045.91
20	26971607.79	27524197.32	26891526.98	8440377.29
21	9308307.68	9294596.35	9027599.32	2621383.36
22	12236244.62	13487894.06	12845139.71	4086377.84
23	16049454.58	18488882.82	18641489.15	5588152.20
24	16130180.70	17684218.91	17976377.72	5395618.84
25	8287084.85	9323012.09	9109081.84	2771326.93
26	12071075.04	13155335.57	13675691.91	4074341.77
27	20293011.81	22744012.75	22307711.41	6575320.15
28	17715246.14	16506893.13	16080704.97	5026062.83
29	6361590.26	7125307.50	6671234.14	2086249.72
30	5700327.43	5742314.29	5594701.86	1758306.50
31	18327012.80	18267238.50	17806714.45	5483441.47
32	14596588.10	15489271.05	15396528.95	4798727.90
33	3387560.76	3549000.39	3433620.36	1065369.52
34	12561536.03	12853618.02	12485995.94	3838014.16
35	9642858.59	10838313.00	11322421.12	3434129.81

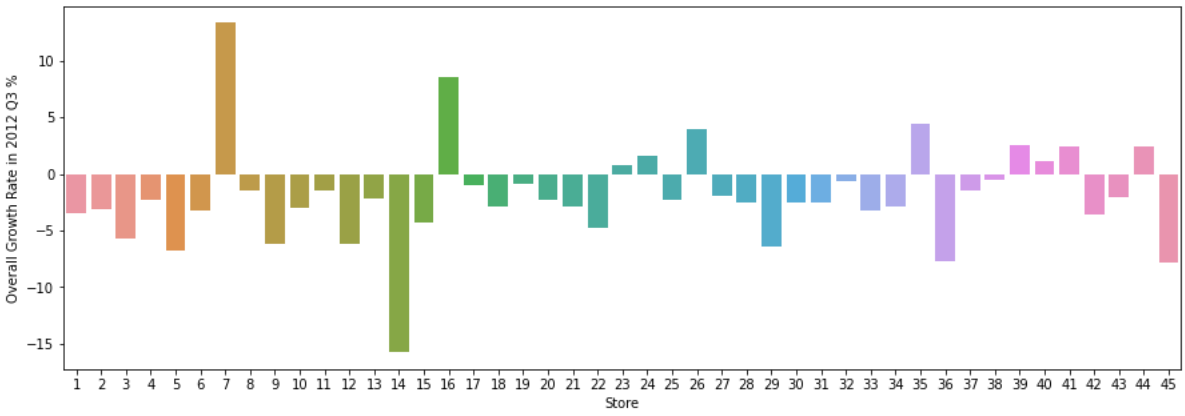
Store	Quater1_Sales	Quater2_Sales	Quater3_Sales	Quater4_Sales
36	4165563.14	4151991.58	3831691.64	1137224.17
37	6905516.43	6824549.37	6728068.24	2154640.65
38	5645775.74	5637918.82	5605482.38	1741896.51
39	18740604.09	20214128.46	20715116.23	6215814.07
40	11680404.16	12727737.53	12873195.37	3891070.28
41	15681607.44	17659942.73	18093844.01	5452445.75
42	7823655.75	7568239.27	7296759.34	2261705.49
43	8332318.07	8168836.35	8000572.16	2473507.39
44	4109696.37	4306405.78	4411251.16	1360020.41
45	9805267.57	10390767.83	9581268.38	2946326.39

In [18]:
```
# Growth rate formula- ((Present value – Past value )/Past value )*100

df2['Q3 - Q2'] = df2['Quater3_Sales'] - df2['Quater2_Sales']
df2['Overall Growth Rate in 2012 Q3 %'] = (df2['Q3 - Q2']/df2['Quater2_Sales'])*10

df2['Overall Growth Rate in 2012 Q3 %'].idxmax() # Store which has good growth in
```

Out[18]:
```
7
```

In [19]:
```
# Plotting the data in Bar chart
plt.figure(figsize=(15,5))
sns.barplot(x=df2.index, y = 'Overall Growth Rate in 2012 Q3 %', data = df2)
```

Out[19]:
```
<AxesSubplot:xlabel='Store', ylabel='Overall Growth Rate in 2012 Q3 %'>
```



Store 7 has good growth in Q3-2012

1. Some holidays have a negative impact on sales. Find out holidays which have higher sales than the mean sales in non-holiday season for all stores together.

In [20]:
```
#finding the mean sales of non holiday and holiday
data.groupby('Holiday_Flag')['Weekly_Sales'].mean()
```

Out[20]:
```
Holiday_Flag
0    1.041256e+06
1    1.122888e+06
Name: Weekly_Sales, dtype: float64
```

In [21]:
```
# Marking the holiday dates
data['Date'] = pd.to_datetime(data['Date'])

Christmas1 = pd.Timestamp(2010,12,31)
Christmas2 = pd.Timestamp(2011,12,30)
Christmas3 = pd.Timestamp(2012,12,28)
Christmas4 = pd.Timestamp(2013,12,27)

Thanksgiving1=pd.Timestamp(2010,11,26)
Thanksgiving2=pd.Timestamp(2011,11,25)
Thanksgiving3=pd.Timestamp(2012,11,23)
Thanksgiving4=pd.Timestamp(2013,11,29)

LabourDay1=pd.Timestamp(2010,9,10)
LabourDay2=pd.Timestamp(2011,9,9)
LabourDay3=pd.Timestamp(2012,9,7)
LabourDay4=pd.Timestamp(2013,9,6)

SuperBowl1=pd.Timestamp(2010,2,12)
SuperBowl2=pd.Timestamp(2011,2,11)
SuperBowl3=pd.Timestamp(2012,2,10)
SuperBowl4=pd.Timestamp(2013,2,8)

#Calculating the mean sales during the holidays
Christmas_mean_sales=data[(data['Date'] == Christmas1) | (data['Date'] == Christmas
Thanksgiving_mean_sales=data[(data['Date'] == Thanksgiving1) | (data['Date'] == Tha
LabourDay_mean_sales=data[(data['Date'] == LabourDay1) | (data['Date'] == LabourDay
SuperBowl_mean_sales=data[(data['Date'] == SuperBowl1) | (data['Date'] == SuperBowl
Christmas_mean_sales

list_of_mean_sales = {'Christmas_mean_sales' : round(Christmas_mean_sales['Weekly_S
'Thanksgiving_mean_sales': round(Thanksgiving_mean_sales['Weekly_Sales'].mean(),2),
'LabourDay_mean_sales' : round(LabourDay_mean_sales['Weekly_Sales'].mean(),2),
'SuperBowl_mean_sales':round(SuperBowl_mean_sales['Weekly_Sales'].mean(),2),
'Non holiday weekly sales' : round(data[data['Holiday_Flag'] == 0 ]['Weekly_Sales'
list_of_mean_sales
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '19-02-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '26-02-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '19-03-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '26-03-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '16-04-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '23-04-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '30-04-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '14-05-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '21-05-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '28-05-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '18-06-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '25-06-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '16-07-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '23-07-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '30-07-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '13-08-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '20-08-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '27-08-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '17-09-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '24-09-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '15-10-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '22-10-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '29-10-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '19-11-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '26-11-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '17-12-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '24-12-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '31-12-2010' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '14-01-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '21-01-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '28-01-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '18-02-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '25-02-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '18-03-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '25-03-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '15-04-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '22-04-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '29-04-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '13-05-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '20-05-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '27-05-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '17-06-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '24-06-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '15-07-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '22-07-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '29-07-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '19-08-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '26-08-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '16-09-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '23-09-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '30-09-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '14-10-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '21-10-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '28-10-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '18-11-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '25-11-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '16-12-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '23-12-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '30-12-2011' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '13-01-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '20-01-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '27-01-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '17-02-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '24-02-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '16-03-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '23-03-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '30-03-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '13-04-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '20-04-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '27-04-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '18-05-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '25-05-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '15-06-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '22-06-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '29-06-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '13-07-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '20-07-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '27-07-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '17-08-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '24-08-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '31-08-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '14-09-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '21-09-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '28-09-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '19-10-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
/home/spx072/anaconda3/lib/python3.9/site-packages/pandas/core/tools/datetimes.py:
1047: UserWarning: Parsing '26-10-2012' in DD/MM/YYYY format. Provide format or sp
ecify infer_datetime_format=True for consistent parsing.
  cache_array = _maybe_cache(arg, format, cache, convert_listlike)
```

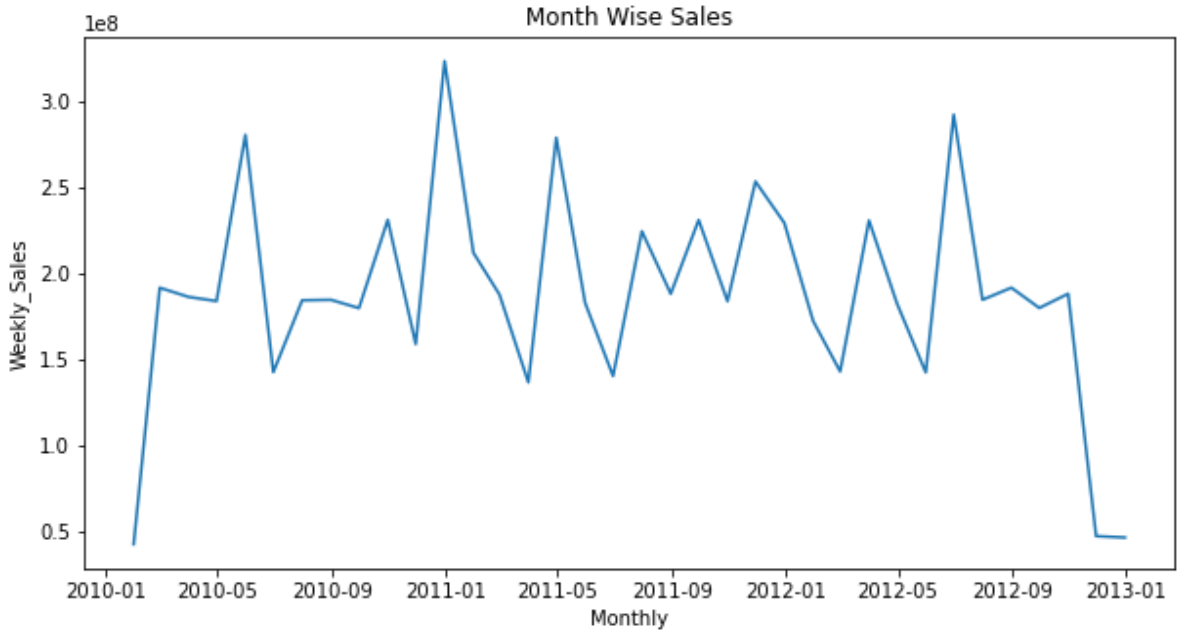
Out[21]:
```
{'Christmas_mean_sales': 960833.11,
 'Thanksgiving_mean_sales': 1471273.43,
 'LabourDay_mean_sales': 1039182.83,
 'SuperBowl_mean_sales': nan,
 'Non holiday weekly sales': 1041256.38}
```

"Thanksgiving Day" has much high sale than mean sales in Non-Holiday season.

1. Provide a monthly and semester view of sales in units and give insights

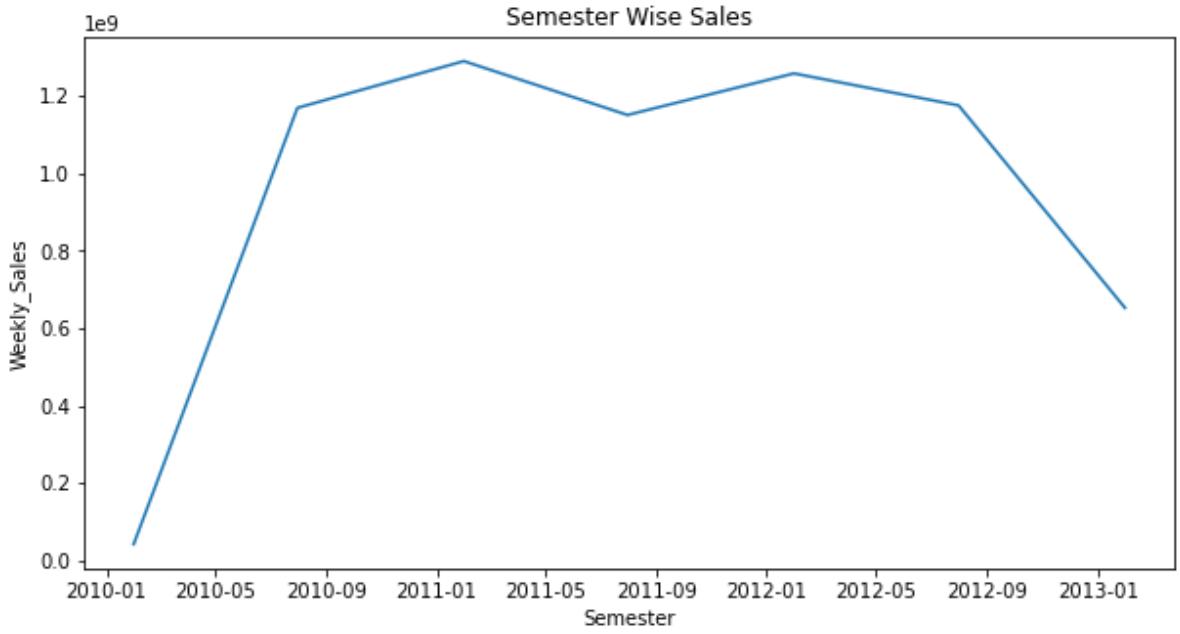
In [22]:
```
#Monthly sales
monthly = data.groupby(pd.Grouper(key='Date', freq='1M')).sum() # groupby each 1 m
monthly=monthly.reset_index()
fig, ax = plt.subplots(figsize=(10,5))
X = monthly['Date']
Y = monthly['Weekly_Sales']
plt.plot(X,Y)
plt.title('Month Wise Sales')
plt.xlabel('Monthly')
plt.ylabel('Weekly_Sales')

# Analysis- highest sum of sales is recorded in between jan-2011 to march-2011.
```

Out[22]:
```
Text(0, 0.5, 'Weekly_Sales')
```



In [23]:
```
#Semester Sales
Semester = data.groupby(pd.Grouper(key='Date', freq='6M')).sum()
Semester = Semester.reset_index()
fig, ax = plt.subplots(figsize=(10,5))
X = Semester['Date']
Y = Semester['Weekly_Sales']
plt.plot(X,Y)
plt.title('Semester Wise Sales')
plt.xlabel('Semester')
plt.ylabel('Weekly_Sales')

# ANalysis- sales are lowest in beginning of 1st sem of 2010 and 1st sem of 2013
```

Out[23]:
```
Text(0, 0.5, 'Weekly_Sales')
```



For Store 1 – Build prediction models to forecast demand Linear Regression – Utilize variables like date and restructure dates as 1 for 5 Feb 2010 (starting from the earliest date in order). Hypothesize if CPI, unemployment, and fuel price have any impact on sales.

In [24]:
```
hypothesis = growth.groupby('Store')[['Fuel_Price','Unemployment', 'CPI','Weekly_Sa
factors  = hypothesis.get_group(1) #Filter by Store 1
```

```
day_arr = [1]
for i in range (1,len(factors)):
    day_arr.append(i*7)

factors['Day'] = day_arr.copy()
factors
```

```
/tmp/ipykernel_38219/469880652.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  factors['Day'] = day_arr.copy()
```

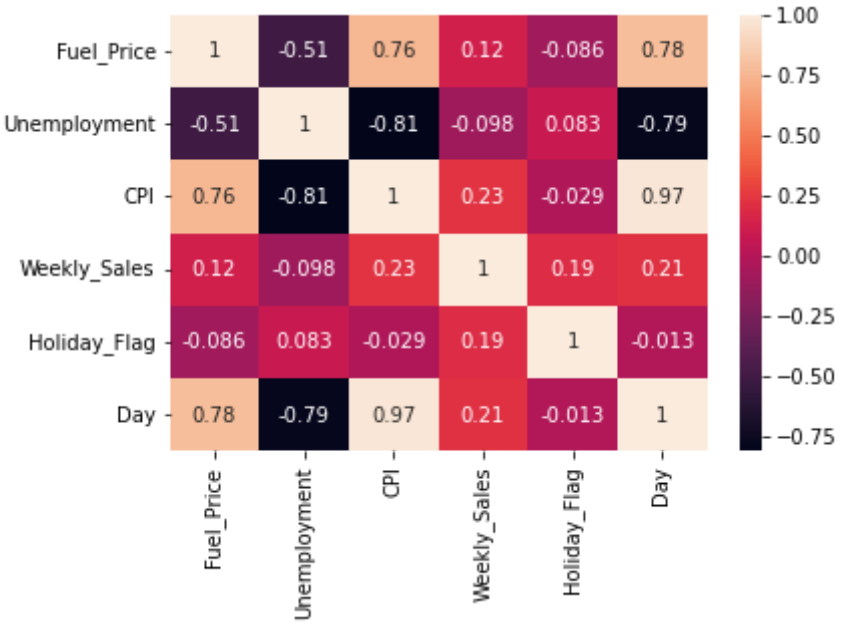
Out[24]:

	Fuel_Price	Unemployment	CPI	Weekly_Sales	Holiday_Flag	Day
0	2.572	8.106	211.096358	1643690.90	0	1
1	2.548	8.106	211.242170	1641957.44	1	7
2	2.514	8.106	211.289143	1611968.17	0	14
3	2.561	8.106	211.319643	1409727.59	0	21
4	2.625	8.106	211.350143	1554806.68	0	28
...	...	...	...	...	...	...
138	3.666	6.908	222.981658	1437059.26	0	966
139	3.617	6.573	223.181477	1670785.97	0	973
140	3.601	6.573	223.381296	1573072.81	0	980
141	3.594	6.573	223.425723	1508068.77	0	987
142	3.506	6.573	223.444251	1493659.74	0	994

143 rows × 6 columns

In [25]:
```
sns.heatmap(factors.corr(), annot = True)
```

Out[25]:
```
<AxesSubplot:>
```

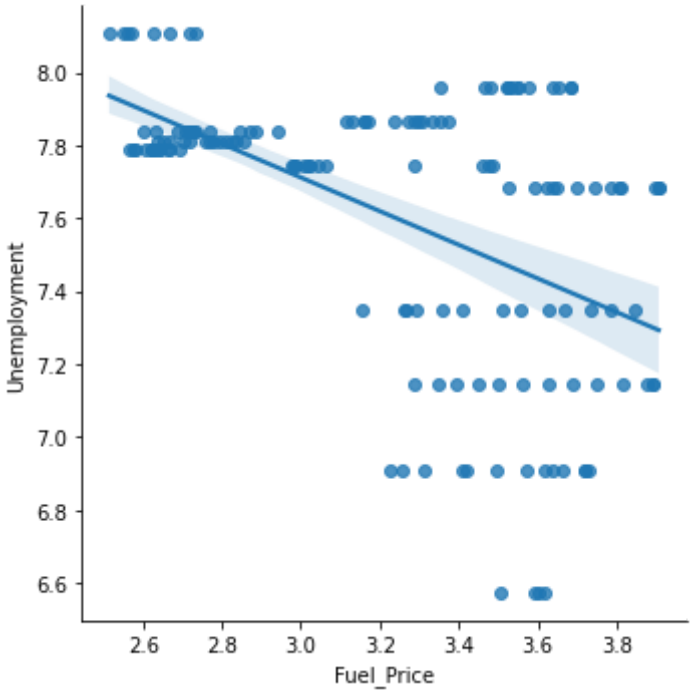


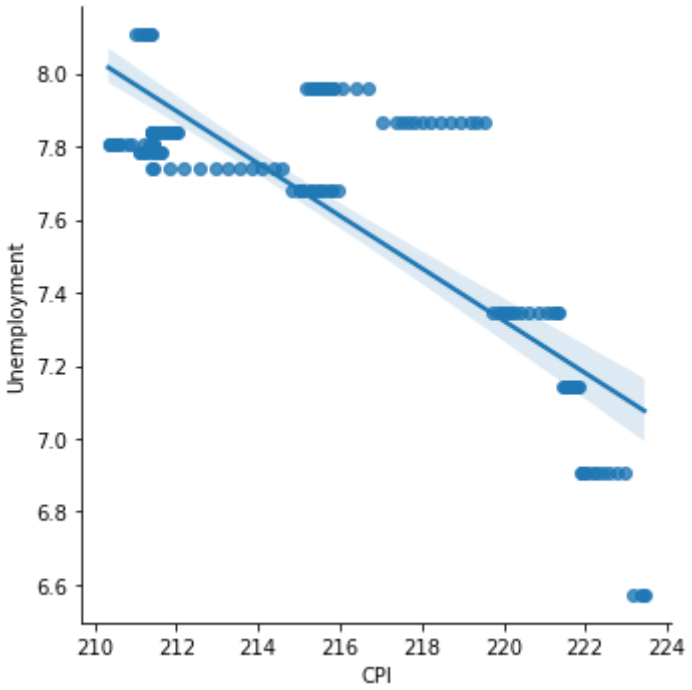
Few variables which are positive and have value greater than zero are correlated with

Weekly_Sales. We can also see CPI and Holiday_Flag is fairly strongly correlated to Weekly_Sales. Holiday_Flag = 1 means it's holiday_week we have sales more than the non_holiday_weeks.

In [26]:
```
sns.lmplot(x='Fuel_Price', y = 'Unemployment', data = factors)
#plt.figure()
sns.lmplot(x='CPI', y = 'Unemployment', data = factors)
```

Out[26]:
```
<seaborn.axisgrid.FacetGrid at 0x7fe98ddadd30>
```





As the Fuel_price and Cpi goes high, rate of Unemployment Fairly Decreases (shown above in Line Regression plot).

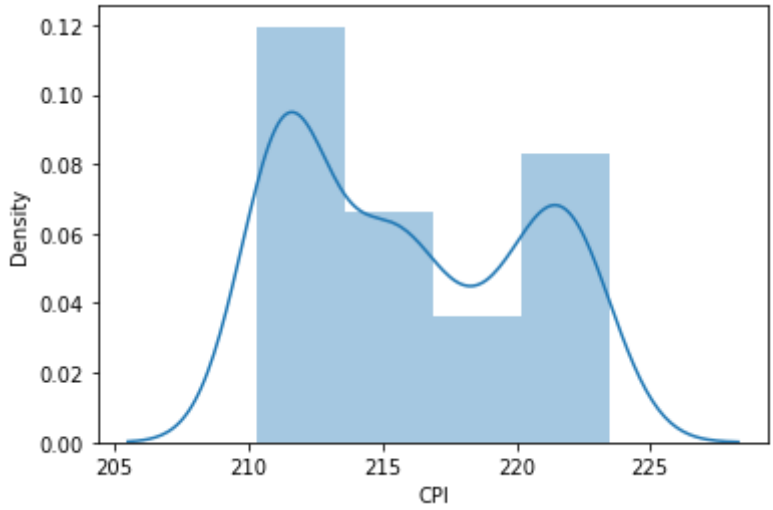
Hypothesis Testing - CPI

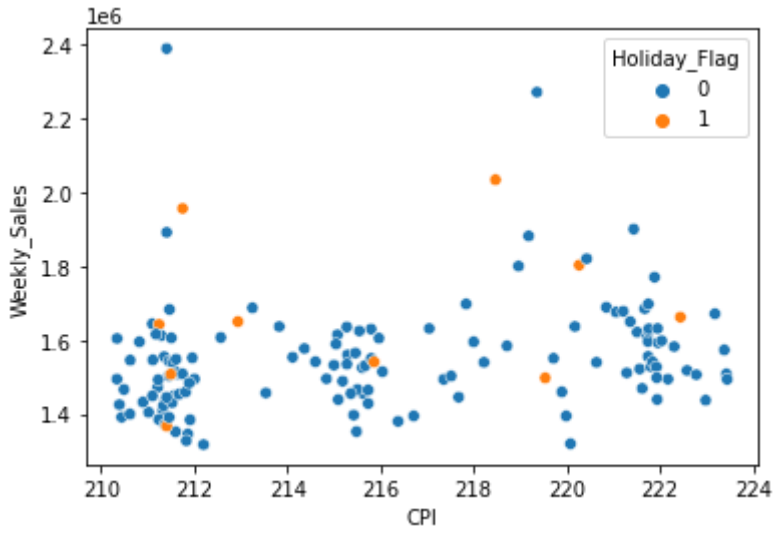
In [28]:
```
from scipy import stats
ttest,pval = stats.ttest_rel(factors['Weekly_Sales'],factors['CPI'])
```

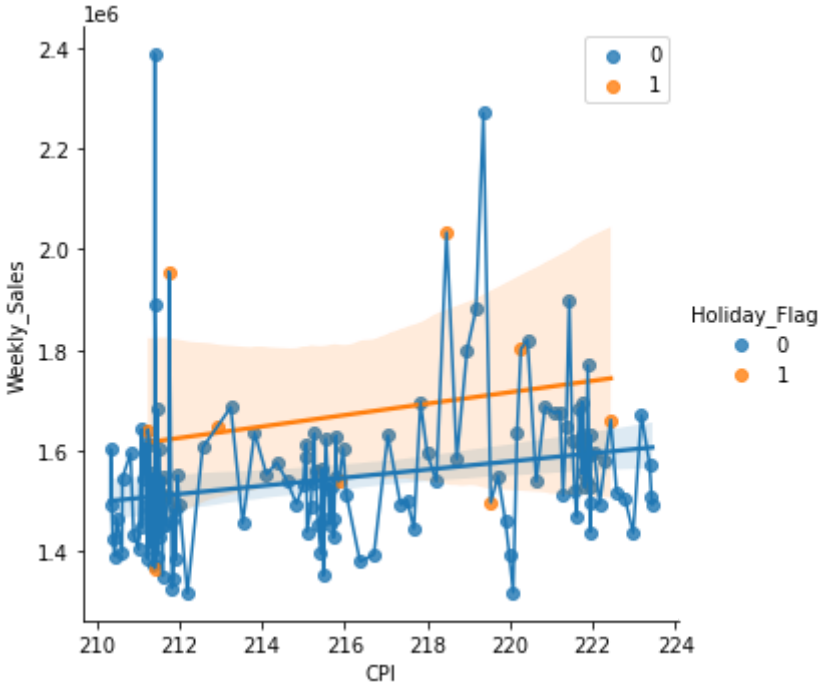
```
sns.distplot(factors.CPI)
plt.figure()
print(pval)
if pval<0.05:
    print("reject null hypothesis")
else:
    print("accept null hypothesis")

sns.scatterplot(x='CPI', y = 'Weekly_Sales', data = factors, hue = 'Holiday_Flag')
#plt.figure()
sns.lmplot(x='CPI', y = 'Weekly_Sales', data = factors, hue = 'Holiday_Flag')
#plt.figure()
sns.lineplot(x='CPI', y = 'Weekly_Sales', data = factors)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a future
version. Please adapt your code to use either `displot` (a figure-level function w
ith similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
3.106725927640744e-144
reject null hypothesis
```

Out[28]: <AxesSubplot:xlabel='CPI', ylabel='Weekly_Sales'>







1) Earlier, we rejected the null hypothesis saying that ther is no relationship between Weekly_sales and CPI. But we found there is a positive corrlation between CPI and Weekly_sales as shown in the above graphs.

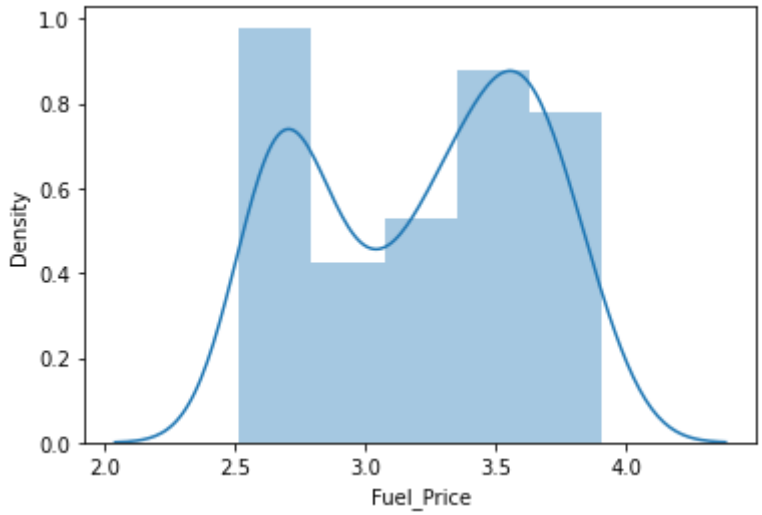
2) The CPI is not normally distributed and line regression plot is showing how CPI is varying with Weekly_Sales on days of Holidays and non holiday weeks.

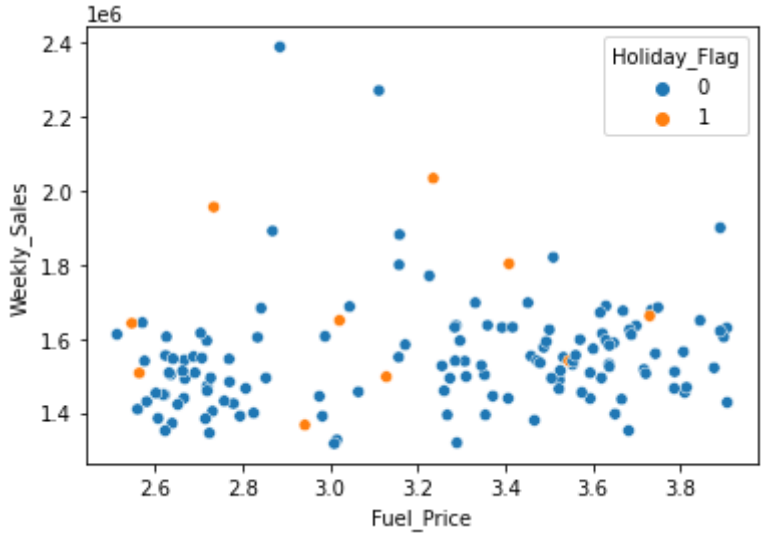
Hypothesis Testing - Fuel_Price

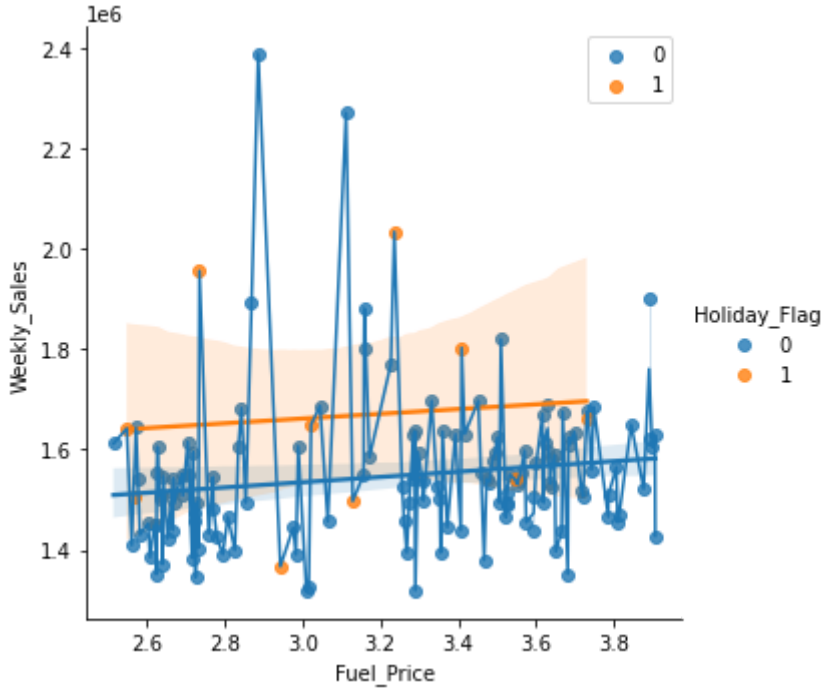
In [29]:
```
from scipy import stats
ttest,pval = stats.ttest_rel(factors['Weekly_Sales'],factors['Fuel_Price'])
sns.distplot(factors.Fuel_Price)
plt.figure()
print(pval)
if pval<0.05:
    print("reject null hypothesis")
else:
    print("accept null hypothesis")

sns.scatterplot(x='Fuel_Price', y = 'Weekly_Sales', data = factors, hue = 'Holiday_
#plt.figure()
sns.lmplot(x='Fuel_Price', y = 'Weekly_Sales', data = factors, hue = 'Holiday_Flag
#plt.figure()
sns.lineplot(x='Fuel_Price', y = 'Weekly_Sales', data = factors)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a future
version. Please adapt your code to use either `displot` (a figure-level function w
ith similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
3.050079726743709e-144
reject null hypothesis
```

Out[29]:
```
<AxesSubplot:xlabel='Fuel_Price', ylabel='Weekly_Sales'>
```







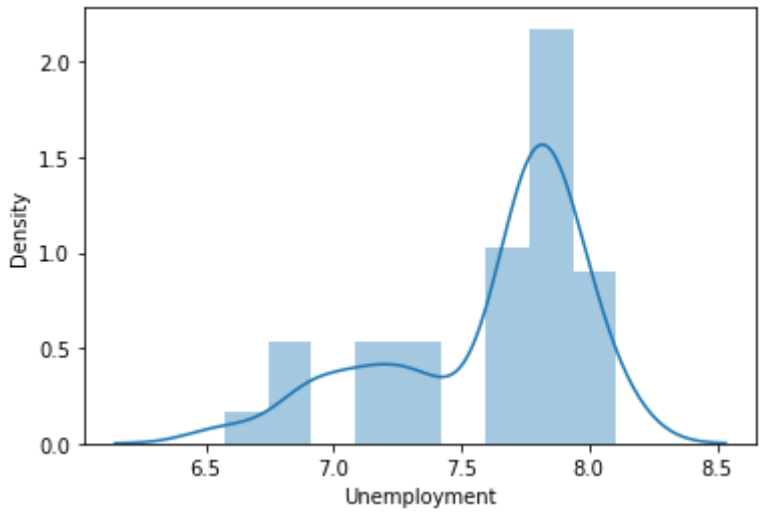
There are more number of Sales when the Fuel_Price are higher and also we can see more Sales during Holiday_Weeks when fuel_prices were fairly low. So its not clear to say on what factors Fuel_price has a direct dependency on Sales.

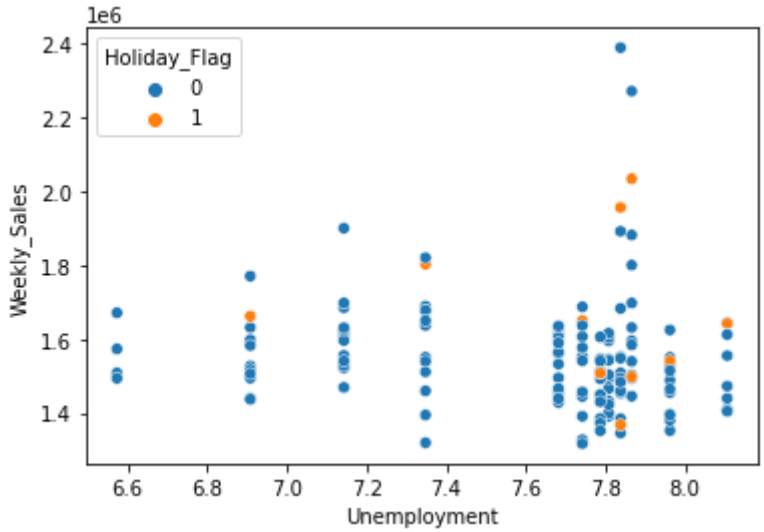
Hypothesis Testing - Uneployment

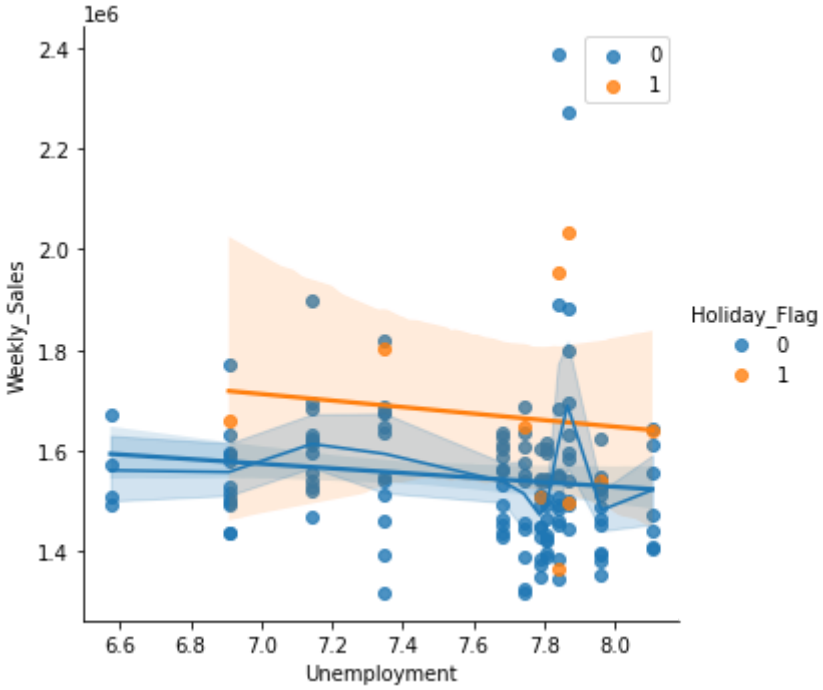
In [30]:
```
from scipy import stats
ttest,pval = stats.ttest_rel(factors['Weekly_Sales'],factors['Unemployment'])
sns.distplot(factors.Unemployment)
plt.figure()
print(pval)
if pval<0.05:
    print("reject null hypothesis")
else:
    print("accept null hypothesis")

sns.scatterplot(x='Unemployment', y = 'Weekly_Sales', data = factors, hue = 'Holida
#plt.figure()
sns.lmplot(x='Unemployment', y = 'Weekly_Sales', data = factors, hue = 'Holiday_Fla
#plt.figure()
sns.lineplot(x='Unemployment', y = 'Weekly_Sales', data = factors)
```

```
/home/spx072/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a future
version. Please adapt your code to use either `displot` (a figure-level function w
ith similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
3.0515405336011733e-144
reject null hypothesis
<AxesSubplot:xlabel='Unemployment', ylabel='Weekly_Sales'>
```

Out[30]:





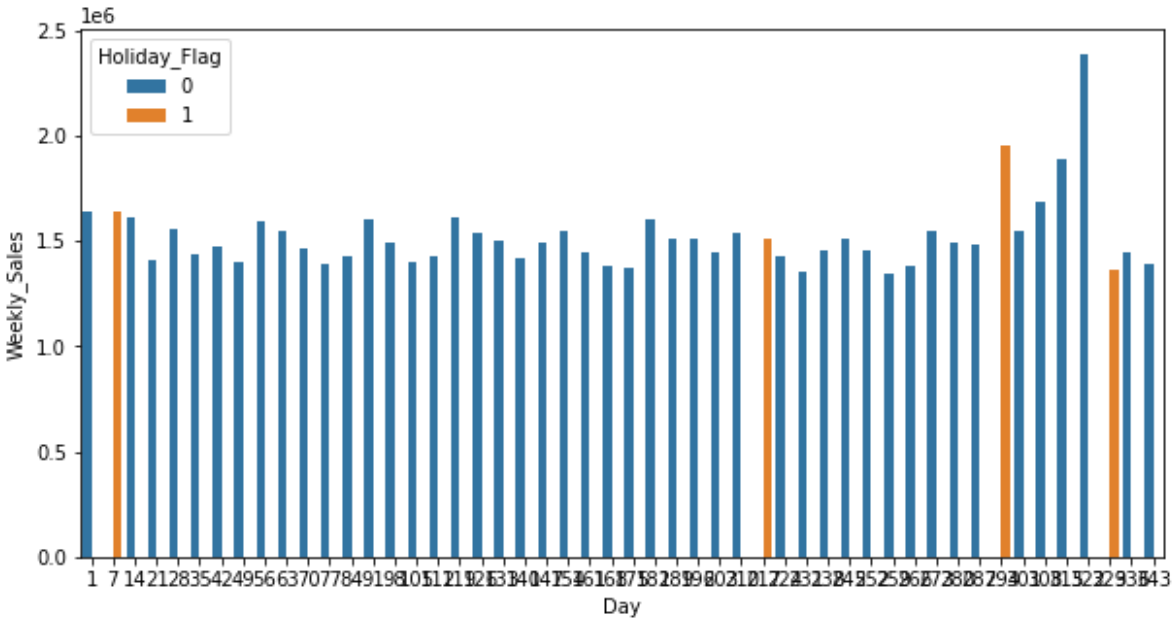


We can see as the rate of unemployment increases, people only buy during holiday seasons, as there are only few outliers present for weekly_sales and which are on the day of Holiday. Speaking of which people only buy necessary products and try to save more. Hence rejecting the null hypothesis was appropriate.

Plotting the Weekly_sales for store 1 (Day wise)

In [31]:
```
plt.figure(figsize=(10,5))
sns.barplot(x='Day', y = 'Weekly_Sales', data = factors.head(50), hue = 'Holiday_F
```

Out[31]:
```
<AxesSubplot:xlabel='Day', ylabel='Weekly_Sales'>
```



In [ ]: