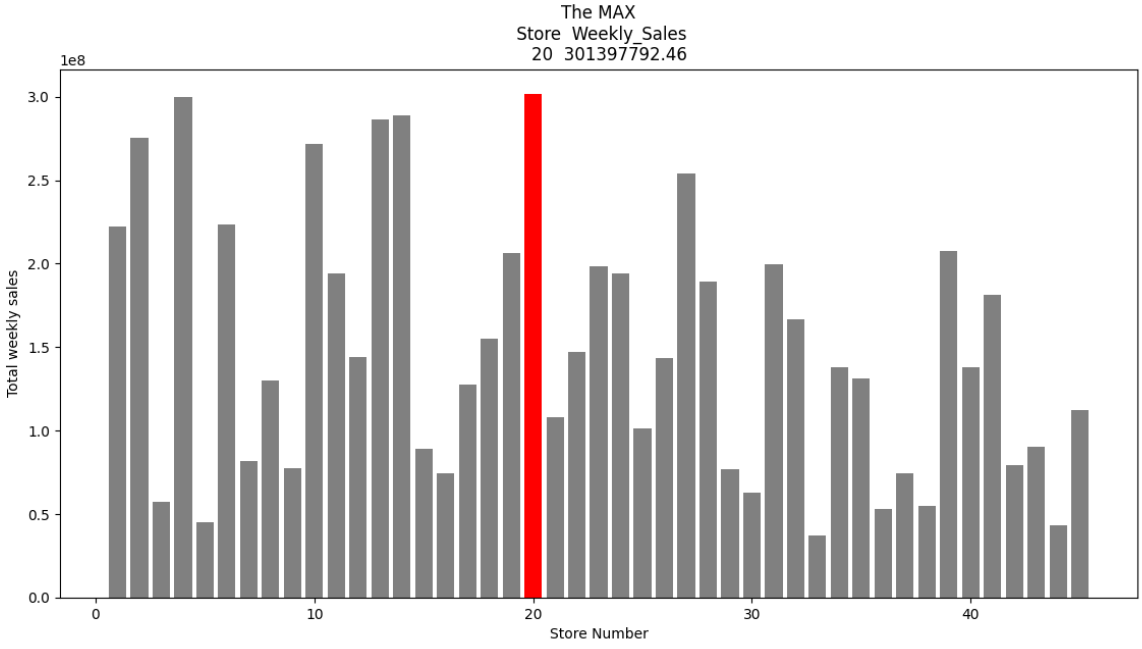
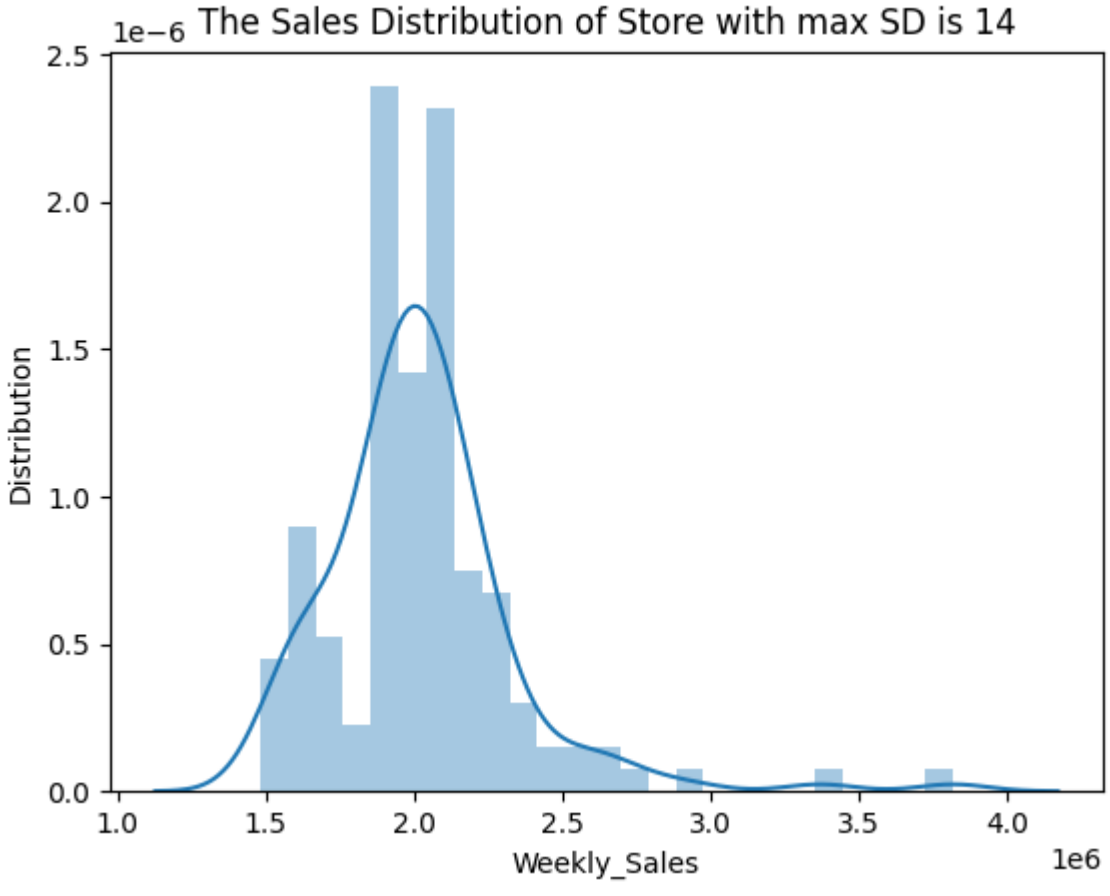
a) Which store has maximum sales?




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

data=pd.read_csv("C:\\Users\\Sherif Tarfa\\Desktop\\Data
project\\Walmart.csv")
dFrame=pd.DataFrame(data)
df=dFrame.groupby("Store")["Weekly_Sales"].sum().to_frame().reset_ind
ex()

colors = ['r' if (bar == max(df['Weekly_Sales'])) else 'grey' for bar
in df['Weekly_Sales']]

plt.bar(x=df["Store"],height=df["Weekly_Sales"],color=colors)

maxSale=df[df["Weekly_Sales"] == max(df["Weekly_Sales"])]
plt.xlabel("Store Number")
plt.ylabel("Total weekly sales")
plt.title("The MAX\n"+maxSale.to_string(index=False))

plt.show()
```

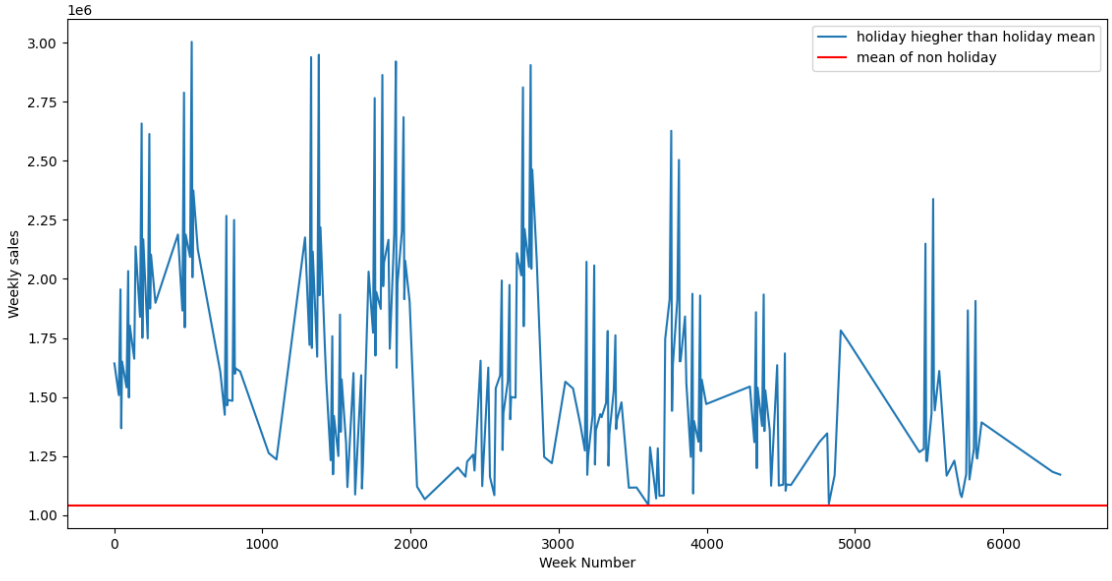
b) Which store has maximum standard deviation i.e., the sales vary a lot




```
The store has maximum standard deviation is 14 with 317569.9494755081
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

data=pd.read_csv("C:\\Users\\Sherif Tarfa\\Desktop\\Data project\\Walmart.csv")
data_std =
pd.DataFrame(data.groupby('Store')['Weekly_Sales'].std().sort_values(ascending=Fals
e))

print("The store has maximum standard deviation is
"+str(data_std.head(1).index[0])+" with
"+str(data_std.head(1).Weekly_Sales[data_std.head(1).index[0]]))

sns.distplot(data[data['Store'] == data_std.head(1).index[0]]['Weekly_Sales'])
plt.title('The Sales Distribution of Store with max SD is '+
str(data_std.head(1).index[0]))
plt.ylabel("Distribution")

plt.show()
```

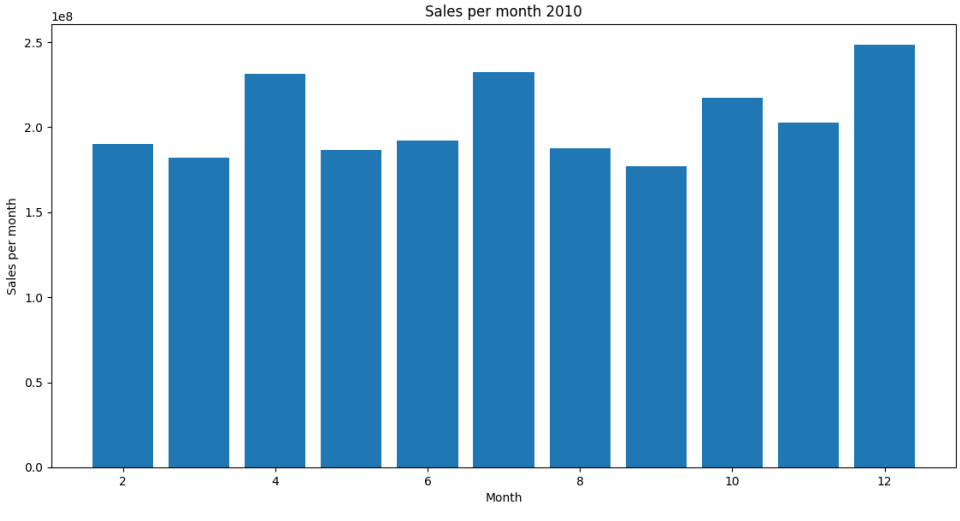
c) Some holidays have a negative impact on sales. Find out holidays that have higher sales than the mean sales in the non-holiday season for all stores together.



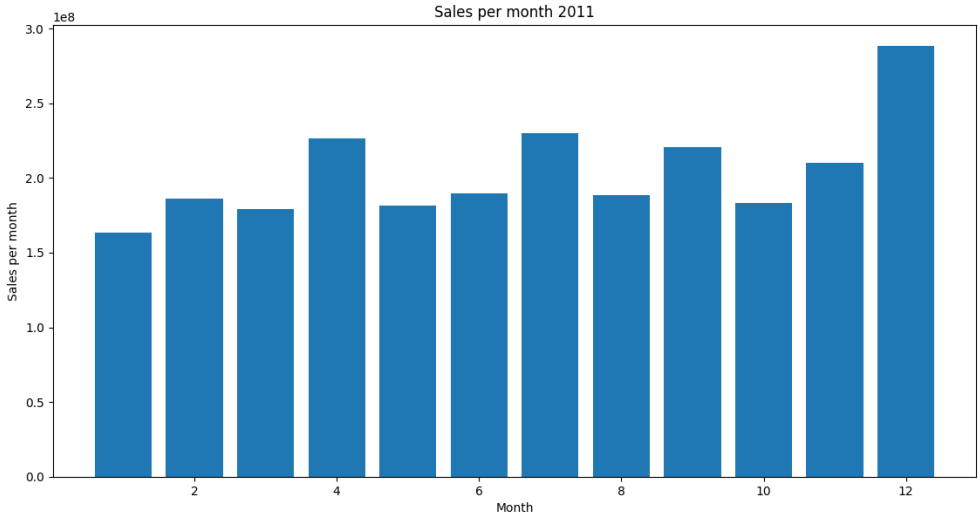
	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployment
1	1	12-02-2010	1641957.44	1	38.51	2.548	211.242170	8.106
31	1	10-09-2010	1507460.69	1	78.69	2.565	211.495190	7.787
42	1	26-11-2010	1955624.11	1	64.52	2.735	211.748433	7.838
47	1	31-12-2010	1367320.01	1	48.43	2.943	211.404932	7.838
53	1	11-02-2011	1649614.93	1	36.39	3.022	212.936705	7.742
83	1	09-09-2011	1540471.24	1	76.00	3.546	215.861056	7.962
94	1	25-11-2011	2033320.66	1	60.14	3.236	218.467621	7.866
99	1	30-12-2011	1497462.72	1	44.55	3.129	219.535990	7.866
105	1	10-02-2012	1802477.43	1	48.02	3.409	220.265178	7.348
135	1	07-09-2012	1661767.33	1	83.96	3.730	222.439015	6.908
144	2	12-02-2010	2137809.50	1	38.49	2.548	210.897994	8.324
174	2	10-09-2010	1839128.83	1	79.09	2.565	211.153210	8.099
185	2	26-11-2010	2658725.29	1	62.98	2.735	211.406287	8.163
190	2	31-12-2010	1750434.55	1	47.30	2.943	211.064774	8.163
196	2	11-02-2011	2168041.61	1	33.19	3.022	212.592862	8.028
226	2	09-09-2011	1748000.65	1	77.97	3.546	215.514829	7.852
237	2	25-11-2011	2614202.30	1	56.36	3.236	218.113027	7.441
242	2	30-12-2011	1874226.52	1	44.57	3.129	219.177306	7.441
248	2	10-02-2012	2103322.68	1	46.98	3.409	219.904907	7.057
278	2	07-09-2012	1898777.07	1	87.65	3.730	222.074763	6.565

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

data=pd.read_csv("C:\\Users\\Sherif Tarfa\\Desktop\\Data project\\Walmart.csv")
df=pd.DataFrame(data)

nonHoliDf=df[df.Holiday_Flag == 0]
mean_nonHoli = nonHoliDf["Weekly_Sales"].mean()

compa = df[(df.Holiday_Flag == 1) & (df.Weekly_Sales > mean_nonHoli)]

plt.plot(compa["Weekly_Sales"] , label="holiday hiegher than holiday mean")
plt.axhline(y = mean_nonHoli, color = 'r', linestyle = '-' , label="mean of non
holiday")
plt.legend()
plt.xlabel("Week Number")
plt.ylabel("Weekly sales")

print(compa.to_string())

plt.show()
```

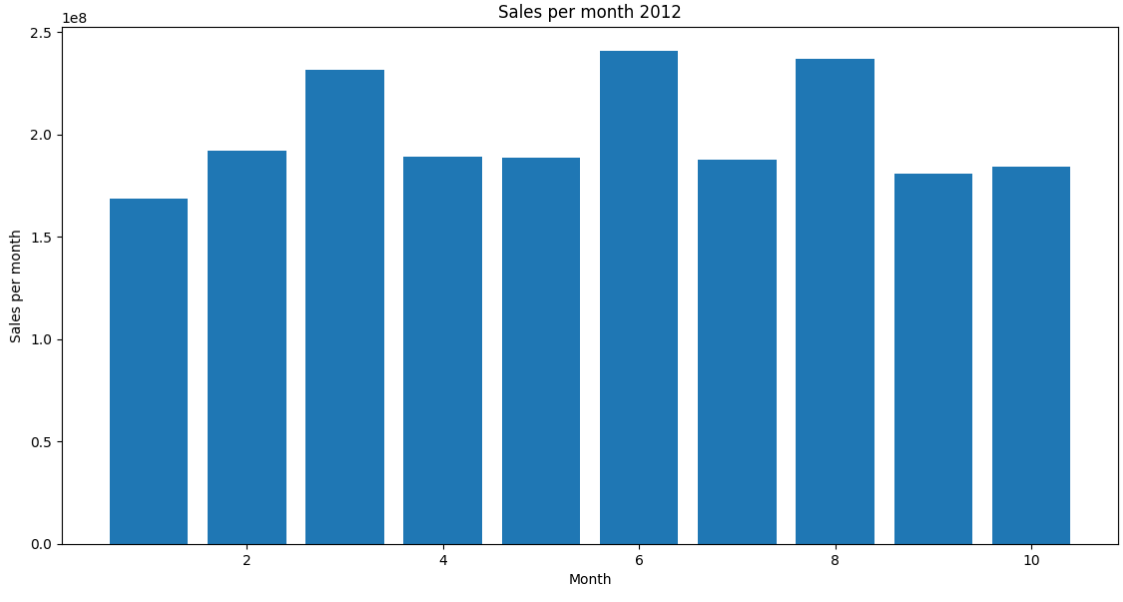
d) Provide a monthly and semester view of sales in units and give insights.



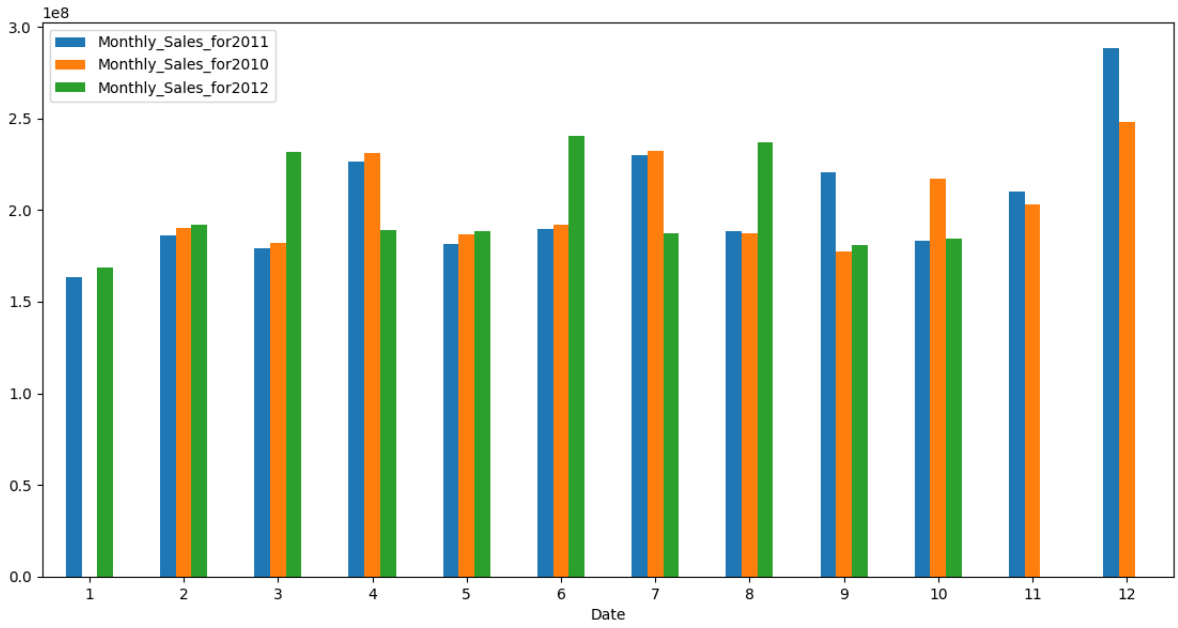

```
plt.bar(df2010ByMonth.index , height=df2010ByMonth["Monthly_Sales_for2010"] )
plt.xlabel("Month")
plt.ylabel("Sales per month")
plt.title("Sales per month 2010")
```




```
plt.bar(df2011ByMonth.index , height=df2012ByMonth["Monthly_Sales_for2011"] )
plt.xlabel("Month")
plt.ylabel("Sales per month")
plt.title("Sales per month 2011")
```

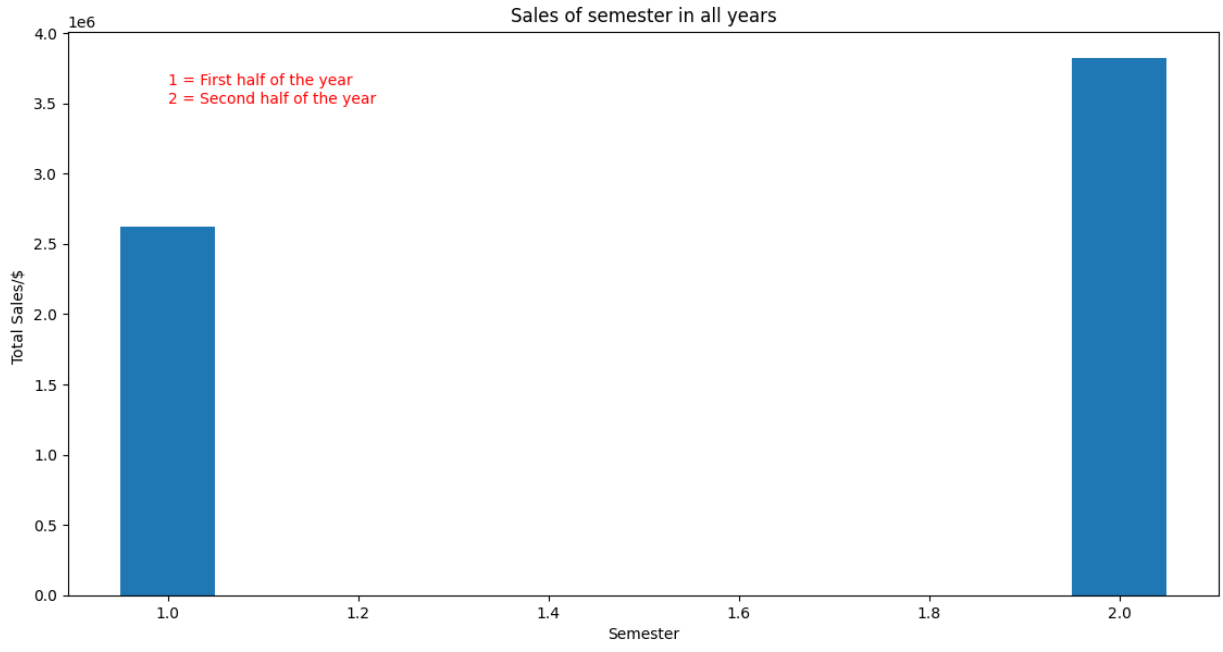



```
plt.bar(df2012ByMonth.index , height=df2012ByMonth["Monthly_Sales_for2012"] )
plt.xlabel("Month")
plt.ylabel("Sales per month")
plt.title("Sales per month 2012")
```



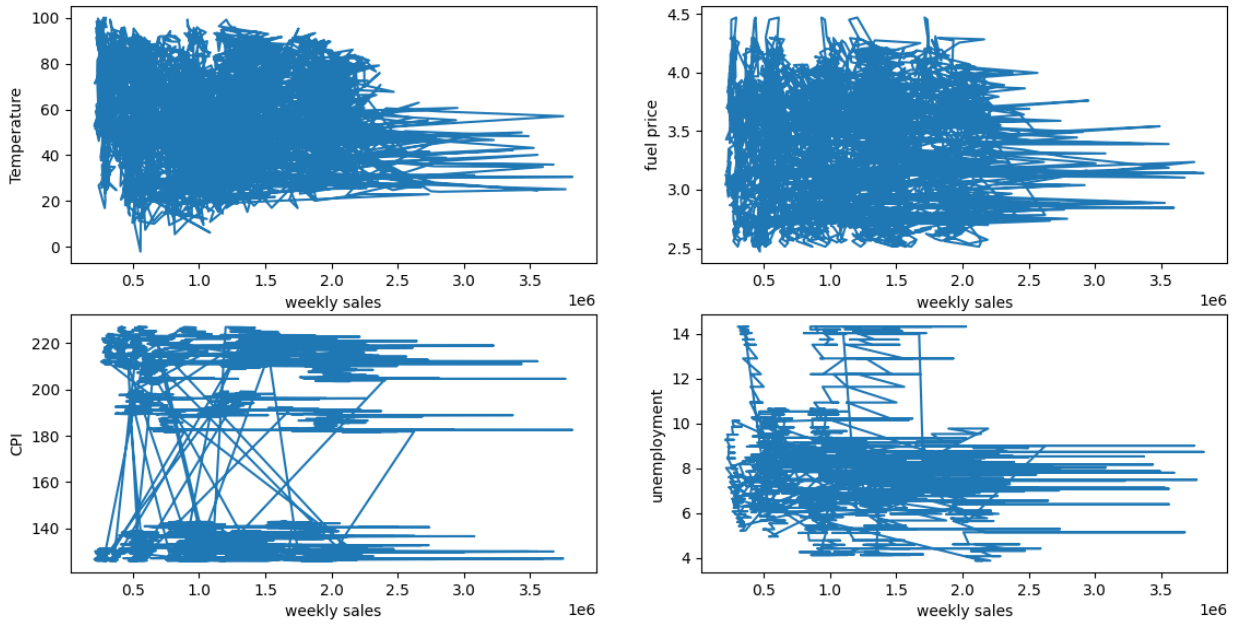

```
import pandas as pd
import matplotlib.pyplot as plt

data=pd.read_csv("C:\\Users\\Sherif Tarfa\\Desktop\\Data project\\Walmart.csv")
df=pd.DataFrame(data)

df['Date'] = pd.to_datetime(df['Date'], format='%d-%m-%Y')

df2010=df.loc[(df['Date'] >= '2010-01-01')
                    & (df['Date'] < '2010-12-31')]

df2011=df.loc[(df['Date'] >= '2011-01-01')
                    & (df['Date'] < '2011-12-31')]

df2012=df.loc[(df['Date'] >= '2012-01-01')
                    & (df['Date'] < '2012-12-31')]
############################################################
df2010ByMonth=df2010.groupby(df2010.Date.dt.month)['Weekly_Sales'].sum()
df2010ByMonth=pd.DataFrame(df2010ByMonth)
df2010ByMonth.columns=["Monthly_Sales_for2010"]
###
df2011ByMonth=df2011.groupby(df2011.Date.dt.month)['Weekly_Sales'].sum()
df2011ByMonth=pd.DataFrame(df2011ByMonth)
df2011ByMonth.columns=["Monthly_Sales_for2011"]
###
df2012ByMonth=df2012.groupby(df2012.Date.dt.month)['Weekly_Sales'].sum()
df2012ByMonth=pd.DataFrame(df2012ByMonth)
df2012ByMonth.columns=["Monthly_Sales_for2012"]
############################################################
#For Each month

#plt.bar(df2012ByMonth.index , height=df2012ByMonth["Monthly_Sales_for2012"] )
#plt.xlabel("Month")
#plt.ylabel("Sales per month")
#plt.title("Sales per month 2012")

#########################################################
#for all months togther

df2 = df2011ByMonth.join(df2010ByMonth)
df3 = df2.join(df2012ByMonth)

ax = df3.plot.bar(rot=0)
```

```
plt.show()
```




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

data=pd.read_csv("C:\\Users\\Sherif Tarfa\\Desktop\\Data project\\Walmart.csv")
df=pd.DataFrame(data)

df['Date'] = pd.to_datetime(df['Date'], format='%d-%m-%Y')

df["Date"]=pd.to_datetime(df["Date"])
df["quarter"]=df["Date"].dt.quarter
df["semster"]=np.where(df["quarter"].isin([1,2]),1,2)

plt.title("Sales of semester in all years")
plt.xlabel("Semester")
plt.ylabel("Total Sales/$ ")
plt.text(1,3500000,"1 = First half of the year \n2 = Second half of the
year",color="r")

plt.bar(df["semster"],height=df["Weekly_Sales"],width=0.1 , align='center')

plt.show()
```

e) Plot the relations between weekly sales vs. other numeric features and give insights.




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

data=pd.read_csv("C:\\Users\\Sherif Tarfa\\Desktop\\Data project\\Walmart.csv")
df=pd.DataFrame(data)

fig,axis = plt.subplots(nrows=2 , ncols=2 )

ax=df.plot("Weekly_Sales","Temperature",ax=axis[0,0])
#ax.set(title="weekly sales vs temp.")
ax.set_xlabel("weekly sales")
ax.set_ylabel("Temperature")
ax.get_legend().remove()

ax=df.plot("Weekly_Sales","Fuel_Price",ax=axis[0,1])
#ax.set_title("weekly sales vs fuel price")
ax.set_xlabel("weekly sales")
ax.set_ylabel("fuel price")
ax.get_legend().remove()

ax=df.plot("Weekly_Sales","CPI",ax=axis[1,0])
#ax.set_title("weekly sales vs cpi")
ax.set_xlabel("weekly sales")
ax.set_ylabel("CPI")
ax.get_legend().remove()

ax=df.plot("Weekly_Sales","Unemployment",ax=axis[1,1])
#ax.set_title("weekly sales vs unemployment")
ax.set_xlabel("weekly sales")
ax.set_ylabel("unemployment")
ax.get_legend().remove()

plt.show()
```