**Introduction:**

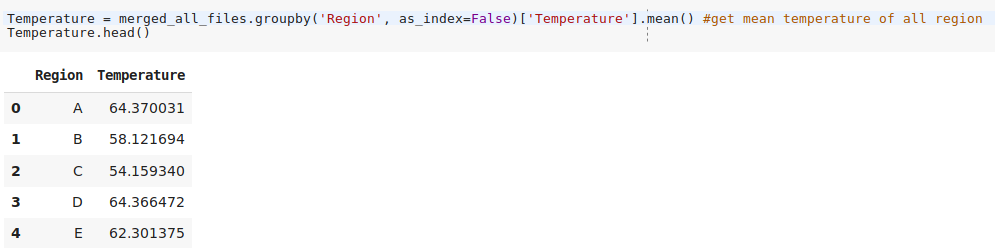
Matplotlib is use for visualization. And import all files using pandas and merge all three files using

.merge method



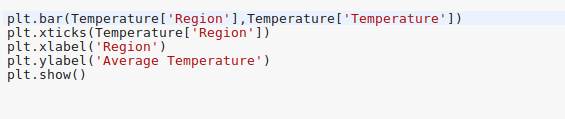
# **1) compare average temperature of five different region's:**

In this task we compute and plot average temperature of each region.



In above code we use groupby method with mean() which will calculate mean of temperature where Region is same I.e. for A ,B,C,D,C regions

Now we will plot this data using followig code:

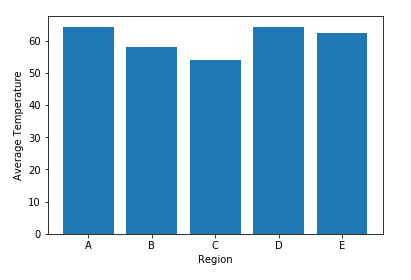


In above code we plot our data using plt.bar() method. In argument we pass Temperature[‘Region’]

As for x-axis and Temperature[‘Temperature’] for y-axis. For x-ticks we use methon plt.xticks()

And pass all region. We pass xlable and ylable using plt.xlable() and plt.ylabel() metho respectively.

Above code will produce following output.

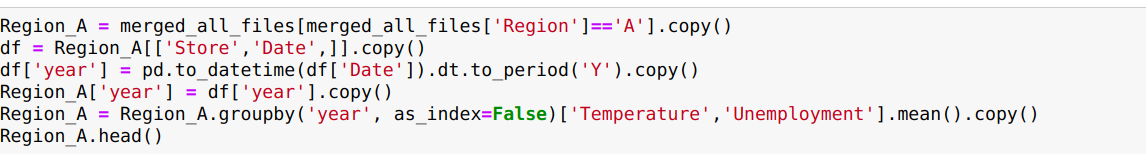


In this graph x-axis shows Region and y-axis shows Average temperature . From this graph we can say that region A and region D has almost same temperature which is highest temperatre among all.

# 

# 2) analyze yearly average temperature and unemployment for region A

In this task we calculate average temperature and Unemployment rate for region A for each year.



In above code we first extract all data from merged\_all\_files where Region = ‘A’. And store it into new

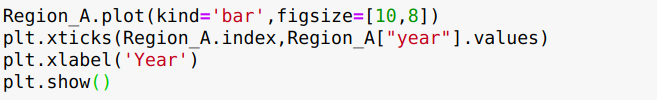
Dataframe Region\_A. Now we need to extract year from Date column for getting average of temperature and Unemployment from dataset. For that we use following syntax:

df['year'] ==pd.to\_datetime(df['Date']).dt.to\_period('Y')

In which to\_datetime method will convert ‘Date’ column into DateandTime format. Then .dt can be used to access the values of the series as datetimelike and return several properties. PandasSeries.dt.to\_period(‘Y’)will return year from datetime.

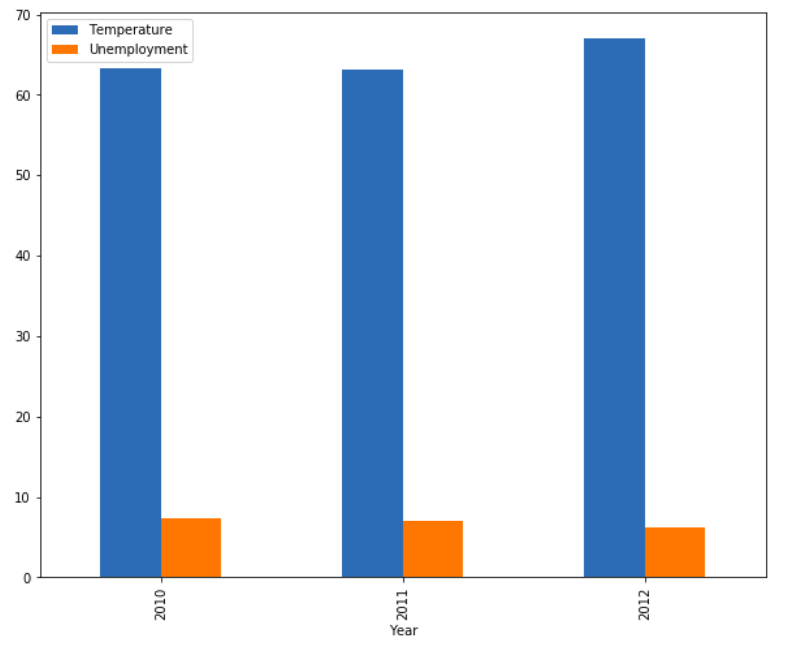
Now we store that year in df[‘year’] and that that column will store in our Region\_A data frame. Now we take average of Temperature and Unemployment in Region\_A using groupby as we use in task1.

Now we will plot this data using followig code:



In this code we use dataframe’s plot methos to plot data. Here plot method takes two argument kind which indicates which typw of bar we need and figsize which is size of our figure for chart.

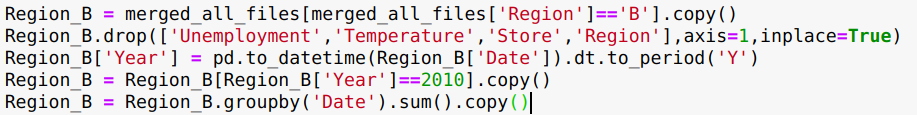
This code will produce following output.



From above graph we can say that the average temperature is high in 2012 and Unemployment rate is lowest in 2012.

# 3) Weekly sales of Region B for 2010

In this task we will compute weekly sales of region B and plot it.



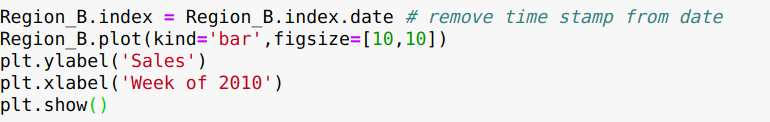
In above code we first extract all data from merged\_all\_files where Region = ‘B’. And store it into new Dataframe Region\_B. Now we drop unnecessary columns like ‘Unemployment’ ,’Temperature’, ’Store’, ’Region’ using drop method.

Now we will create new column year using pd.to\_Datetime(Region\_B[‘Date’]).dt.to\_period(‘Y’) as we create in task 2.

Now extract data where year = 2010

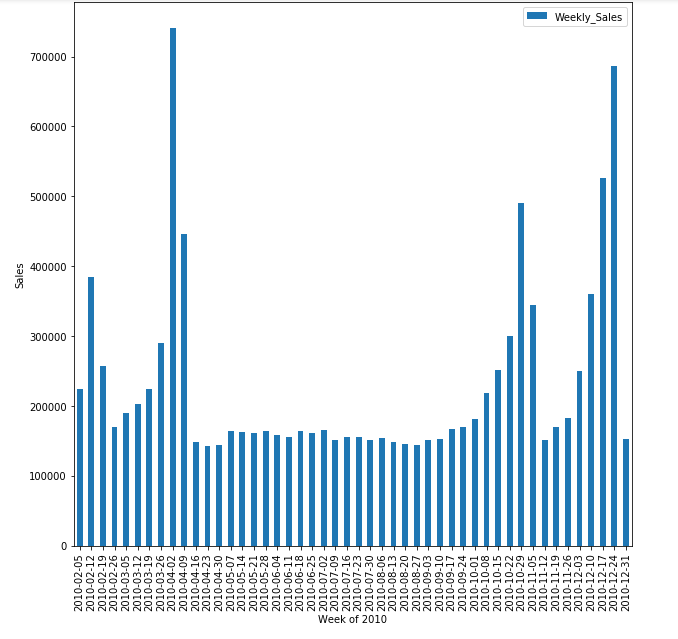
Now we calculate sum where date is same in Region\_B using groupby method.

Following code will plot this Region\_B



In Region\_B date is index. And Region\_B.index = Region\_B.index.date line will remove time stamp from index of Region\_B so we can plot index without time stamp.

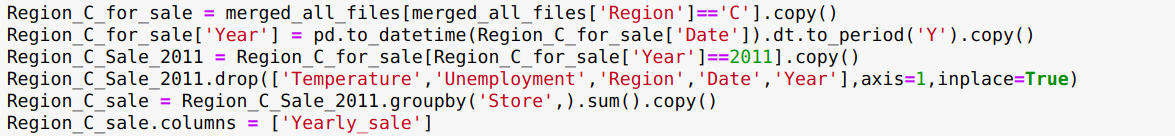
We use Region\_B.plot() method which will plot dataframe and takes two argument as kind which indicated type of chart which is ‘bar’ and other is figsize which will set size of figure of chart.



From above chart we came to know that week 2010-04-02 has high sale and 2010-04-23 has lowest sale.

# 4) sale of store for region C in 2011

In this task we calculate sales of store in region C for year 2011.



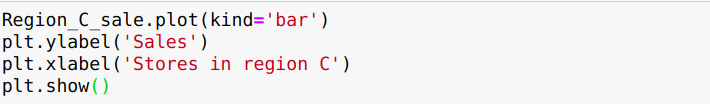
In above code we first extract all data from merged\_all\_files where Region = ‘B’. And store it into new Dataframe Region\_B.

Now we extract Year from Date column using folowing syntax:

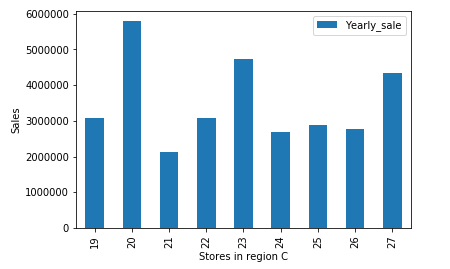
Region\_C\_for\_sale['Year'] = pd.to\_datetime(Region\_C\_for\_sale['Date']).dt.to\_period('Y')

As we done in earlier tasks. now we extract data where year is equal to 2011. And then we drop all unnecessary columns such as ‘Temperature’, ’Unemployment’, ’Region’,’Date’,’Year’ using drop method. Now we calculate sum of all sales based on store value that means where store numbers are equal all sales will be added. We use groupby method with sum(). We store this data into Region\_C\_sale dataframe and change that column name as ‘Yearly\_Sale’.

Following code will plot this data



We use .plot method to plot this data . Here this method takes one argument kind which states type of our chart which is bar. We set ylable and xlabel using plt.xlabel and plt.ylabel method respectively.



From above chart we came to know that store number 20 sales most pizza in 2011.and store number 21 has lowest sale in 2011.