In [1]:

```
# Render our plots inline

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

Reading Data

Data: Weight1.csv (https://github.com/LOVENISHGAUR/Python/tree/master/Pandas/Data)

In [2]:

```
#Reading csv file
Raw = pd.read_csv('Weight1.csv')

# Look at the first 10 rows
Raw[:10]
```

Out[2]:

	Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_
0	01/01/20	26	18	13	145	98	
1	02/01/20	27	12	19	124	100	
2	03/01/20	22	15	22	111	99	
3	04/01/20	22	11	11	104	100	
4	05/01/20	30	16	20	114	97	
5	06/01/20	27	14	13	106	97	
6	07/01/20	22	17	14	106	99	
7	08/01/20	27	15	10	127	99	
8	09/01/20	21	17	25	122	99	
9	10/01/20	28	17	24	134	97	

In [3]:

```
data = pd.read_csv('Weight1.csv', dayfirst=True, index_col=0) # Reading CSV file
and setting the index
data [:10]# Checking the firest 10 columns
```

Out[3]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
01/01/20	26	18	13	145	98	
02/01/20	27	12	19	124	100	
03/01/20	22	15	22	111	99	
04/01/20	22	11	11	104	100	
05/01/20	30	16	20	114	97	
06/01/20	27	14	13	106	97	
07/01/20	22	17	14	106	99	
08/01/20	27	15	10	127	99	
09/01/20	21	17	25	122	99	
10/01/20	28	17	24	134	97	

Describing Data

In [4]:

```
data.head() # Getting the view of data by looking at first 5 rows
```

Out[4]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
01/01/20	26	18	13	145	98	
02/01/20	27	12	19	124	100	
03/01/20	22	15	22	111	99	
04/01/20	22	11	11	104	100	
05/01/20	30	16	20	114	97	

In [5]:

```
data.tail()# Getting the view of data by looking at last 5 rows
```

Out[5]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
27/02/20	30	16	17	145	97	
28/02/20	21	18	11	133	100	
29/02/20	29	15	16	110	99	
01/03/20	21	15	20	124	100	
02/03/20	22	17	13	145	98	

In [6]:

```
data.dtypes# Checking the data types
```

Out[6]:

```
Run(Minutes)              int64
Jog(Minutes)              int64
Exercise(Minutes)         int64
Hearbeat                  int64
Body_Temp(F)              int64
Running_Distance(Km)      int64
Jogging_Distance(Km)      int64
Calories_Burnt(Kcal)      int64
Weight_72                object
dtype: object
```

In [7]:

```
data.shape # checking the structure of the data frame
```

Out[7]:

```
(62, 9)
```

In [8]:

```
data.ndim # Checking the dimension of the data frame
```

Out[8]:

```
2
```

In [9]:

```
data.describe() # Deascribing the data
```

Out[9]:

	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dist
count	62.000000	62.000000	62.000000	62.000000	62.000000	6
mean	24.854839	15.258065	17.451613	124.387097	98.629032	
std	3.401286	3.130681	4.678789	13.807048	1.119629	
min	20.000000	10.000000	10.000000	100.000000	97.000000	
25%	22.000000	12.000000	13.250000	111.500000	98.000000	
50%	25.000000	16.000000	18.000000	124.500000	99.000000	
75%	27.750000	17.750000	21.000000	134.750000	100.000000	
max	30.000000	20.000000	25.000000	149.000000	100.000000	

In [10]:

```
data['Hearbeat'].describe() # Describing specific column 'Heartbeats'
```

Out[10]:

```
count     62.000000
mean     124.387097
std       13.807048
min      100.000000
25%      111.500000
50%      124.500000
75%      134.750000
max      149.000000
Name: Hearbeat, dtype: float64
```

In [11]:

```
data['Running_Distance(Km)'].describe() # Describing specific column 'Running_Di
stance'
```

Out[11]:

```
count    62.000000
mean      4.467742
std       1.112046
min       3.000000
25%       3.250000
50%       4.500000
75%       5.000000
max       6.000000
Name: Running_Distance(Km), dtype: float64
```

In [12]:

```
data['Jogging_Distance(Km)'].describe()# Describing specific column 'Jogging_Dis
tance'
```

Out[12]:

```
count    62.000000
mean      1.983871
std       0.757302
min       1.000000
25%       1.000000
50%       2.000000
75%       3.000000
max       3.000000
Name: Jogging_Distance(Km), dtype: float64
```

In [13]:

```
data.isnull() # Checking for Null values
```

Out[13]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
01/01/20	False	False	False	False	False	
02/01/20	False	False	False	False	False	
03/01/20	False	False	False	False	False	
04/01/20	False	False	False	False	False	
05/01/20	False	False	False	False	False	
...	...	...	...	...	...	
27/02/20	False	False	False	False	False	
28/02/20	False	False	False	False	False	
29/02/20	False	False	False	False	False	
01/03/20	False	False	False	False	False	
02/03/20	False	False	False	False	False	

62 rows × 9 columns

In [14]:

```
data.mean() # Finding the mean of the data set
```

Out[14]:

```
Run(Minutes)              24.854839
Jog(Minutes)              15.258065
Exercise(Minutes)         17.451613
Hearbeat                 124.387097
Body_Temp(F)              98.629032
Running_Distance(Km)       4.467742
Jogging_Distance(Km)       1.983871
Calories_Burnt(Kcal)     277.096774
dtype: float64
```

Selecting a column

In [15]:

```
data.columns # Checking al the columns of the data set
```

Out[15]:

```
Index(['Run(Minutes)', 'Jog(Minutes)', 'Exercise(Minutes)', 'Hearbea
t',
       'Body_Temp(F)', 'Running_Distance(Km)', 'Jogging_Distance(K
m)',
       'Calories_Burnt(Kcal)', 'Weight_72'],
      dtype='object')
```

In [16]:

```
data['Run(Minutes)']
```

Out[16]:

```
Date
01/01/20    26
02/01/20    27
03/01/20    22
04/01/20    22
05/01/20    30
            ..
27/02/20    30
28/02/20    21
29/02/20    29
01/03/20    21
02/03/20    22
Name: Run(Minutes), Length: 62, dtype: int64
```

In [17]:

```
data[['Run(Minutes)','Jog(Minutes)','Exercise(Minutes)']]
```

Out[17]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)
01/01/20	26	18	13
02/01/20	27	12	19
03/01/20	22	15	22
04/01/20	22	11	11
05/01/20	30	16	20
...	...	...	...
27/02/20	30	16	17
28/02/20	21	18	11
29/02/20	29	15	16
01/03/20	21	15	20
02/03/20	22	17	13

62 rows × 3 columns

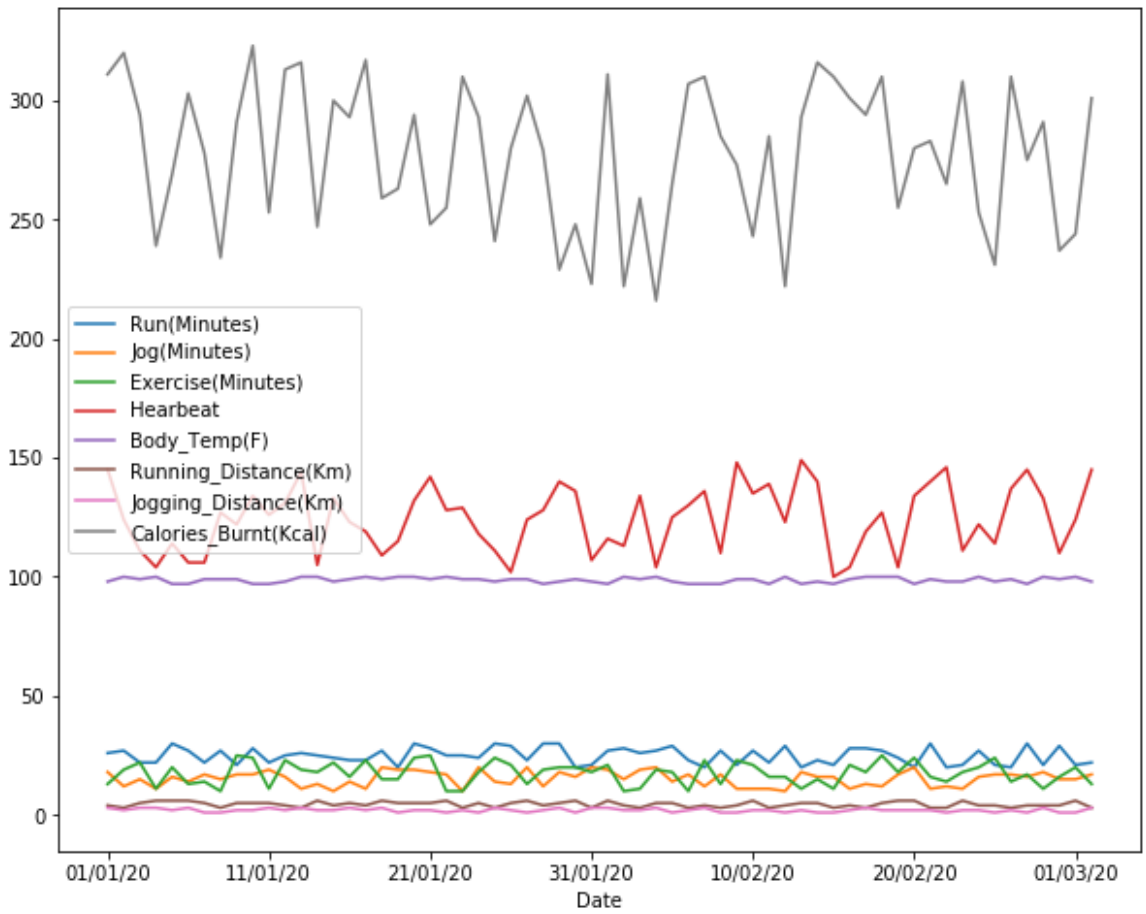
Plotting a column

In [18]:

```
data.plot(figsize= (10,8))
```

Out[18]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11f2a0f50>
```

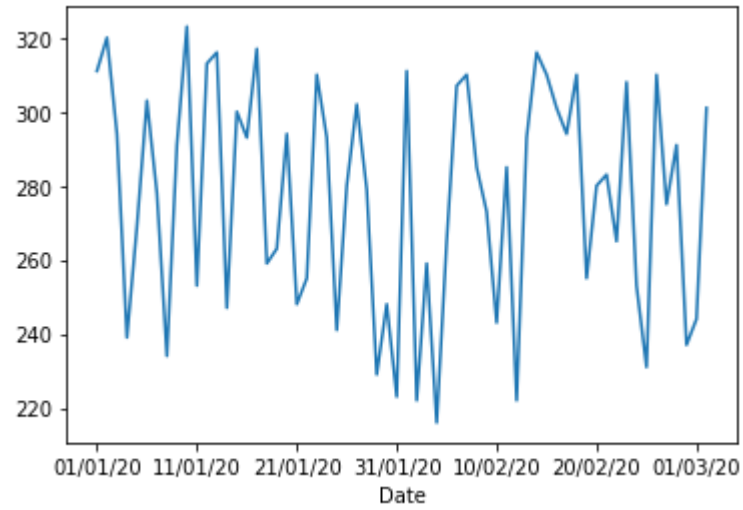


In [19]:

```
data['Calories_Burnt(Kcal)'].plot()
```

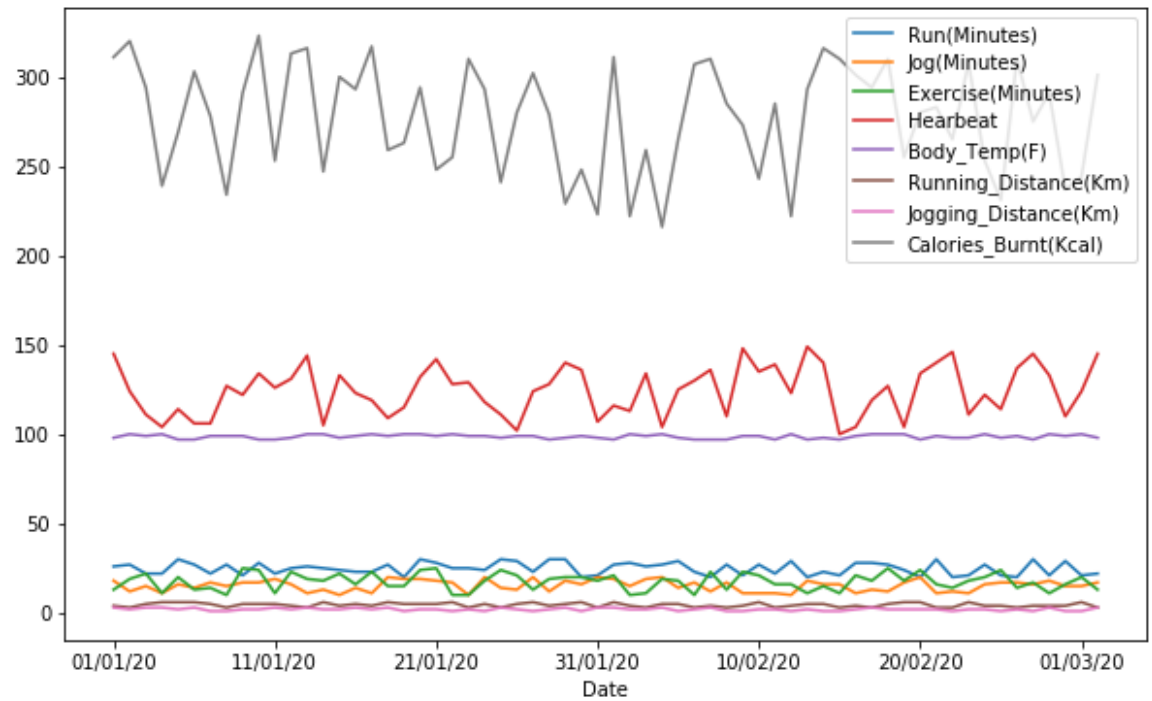
Out[19]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11fc14590>
```



In [20]:

```
data.plot(figsize=(10, 6))
plt.show()
```



Finding NA values in data

In [21]:

```
data.isnull()
```

Out[21]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
01/01/20	False	False	False	False	False	
02/01/20	False	False	False	False	False	
03/01/20	False	False	False	False	False	
04/01/20	False	False	False	False	False	
05/01/20	False	False	False	False	False	
...	...	...	...	...	...	
27/02/20	False	False	False	False	False	
28/02/20	False	False	False	False	False	
29/02/20	False	False	False	False	False	
01/03/20	False	False	False	False	False	
02/03/20	False	False	False	False	False	

62 rows × 9 columns

In [22]:

```
data.apply(len)
```

Out[22]:

```
Run(Minutes)            62
Jog(Minutes)            62
Exercise(Minutes)       62
Hearbeat                62
Body_Temp(F)            62
Running_Distance(Km)    62
Jogging_Distance(Km)    62
Calories_Burnt(Kcal)    62
Weight_72               62
dtype: int64
```

Filtering Data

In [23]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 62 entries, 01/01/20 to 02/03/20
Data columns (total 9 columns):
 #   Column                Non-Null Count  Dtype 
---  ------                --------------  ----- 
 0   Run(Minutes)          62 non-null     int64 
 1   Jog(Minutes)          62 non-null     int64 
 2   Exercise(Minutes)     62 non-null     int64 
 3   Hearbeat              62 non-null     int64 
 4   Body_Temp(F)          62 non-null     int64 
 5   Running_Distance(Km)  62 non-null     int64 
 6   Jogging_Distance(Km)  62 non-null     int64 
 7   Calories_Burnt(Kcal)  62 non-null     int64 
 8   Weight_72             62 non-null     object
dtypes: int64(8), object(1)
memory usage: 7.3+ KB
```

In [24]:

```
data.loc[:,'Weight_72']
```

Out[24]:

```
Date
01/01/20    Yes
02/01/20     No
03/01/20    Yes
04/01/20    Yes
05/01/20     No
           ...
27/02/20    Yes
28/02/20    Yes
29/02/20    Yes
01/03/20    Yes
02/03/20    Yes
Name: Weight_72, Length: 62, dtype: object
```

In [25]:

```
# Finding running distance more than 2 Km
run_2 = data['Running_Distance(Km)'] >=2
run_2
```

Out[25]:

```
Date
01/01/20    True
02/01/20    True
03/01/20    True
04/01/20    True
05/01/20    True
            ...
27/02/20    True
28/02/20    True
29/02/20    True
01/03/20    True
02/03/20    True
Name: Running_Distance(Km), Length: 62, dtype: bool
```

In [26]:

```
data[run_2]
```

Out[26]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
01/01/20	26	18	13	145	98	
02/01/20	27	12	19	124	100	
03/01/20	22	15	22	111	99	
04/01/20	22	11	11	104	100	
05/01/20	30	16	20	114	97	
...	...	...	...	...	...	
27/02/20	30	16	17	145	97	
28/02/20	21	18	11	133	100	
29/02/20	29	15	16	110	99	
01/03/20	21	15	20	124	100	
02/03/20	22	17	13	145	98	

62 rows × 9 columns

In [27]:

```
data.loc[run_2,'Body_Temp(F)']
```

Out[27]:

```
Date
01/01/20     98
02/01/20    100
03/01/20     99
04/01/20    100
05/01/20     97
           ...
27/02/20     97
28/02/20    100
29/02/20     99
01/03/20    100
02/03/20     98
Name: Body_Temp(F), Length: 62, dtype: int64
```

In [28]:

```
Calories = [230,231,232,233,234,235,236,237,238,239,240]
Filt = data['Calories_Burnt(Kcal)'].isin(Calories)
```

In [29]:

```
data[Filt]
```

Out[29]:

Date	Run(Minutes)	Jog(Minutes)	Exercise(Minutes)	Hearbeat	Body_Temp(F)	Running_Dis
04/01/20	22	11	11	104	100	
08/01/20	27	15	10	127	99	
25/02/20	21	17	24	114	98	
29/02/20	29	15	16	110	99	

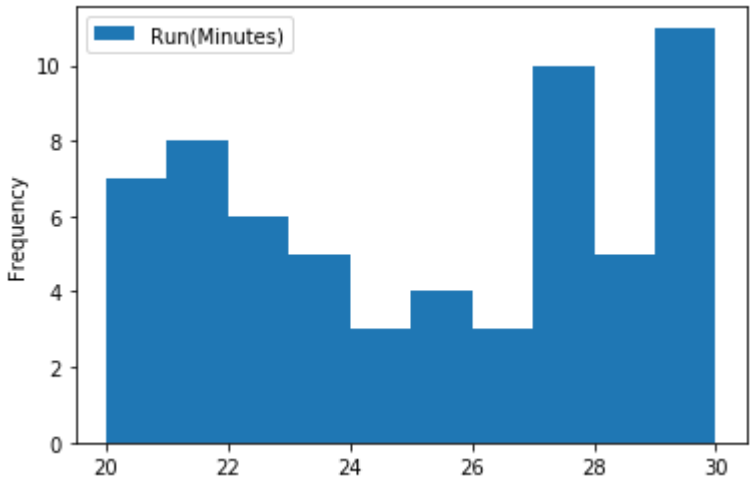
Plotting Data

In [30]:

```
data.plot.hist(x= 'Calories_Burnt(Kcal)', y='Run(Minutes)')
```

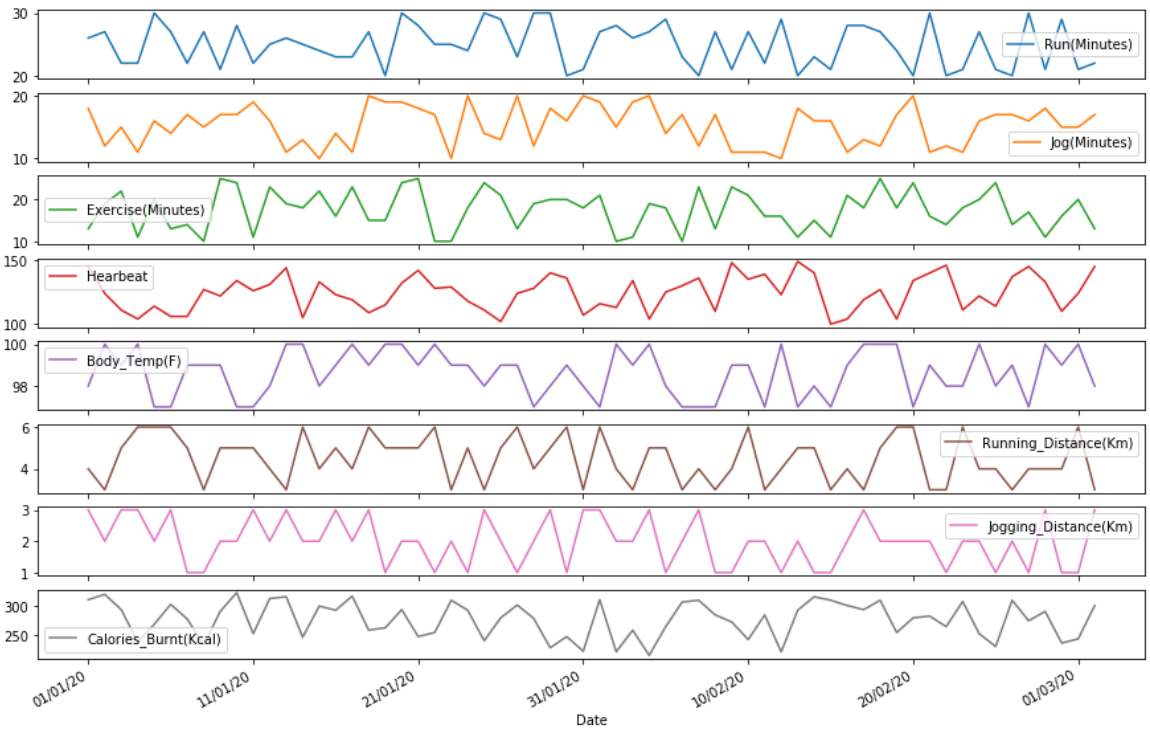
Out[30]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11fe7a850>
```



In [31]:

```
data.plot.line(figsize=(15,10), subplots=True)
plt.show()
```



In [32]:

```
import matplotlib
from matplotlib import pyplot as plt
matplotlib.use('macosx')
```

In [33]:

```
plt.bar('Running_Distance(Km)','Calories_Burnt(Kcal)',data=data,color = 'b',alig
n='center')
plt.xlabel('Calories Burnt')
plt.ylabel("Running Distance ")
plt.title('Distance Run vs Calories Burnt')
plt.show()
```

In [37]:

```
data.columns
```

Out[37]:

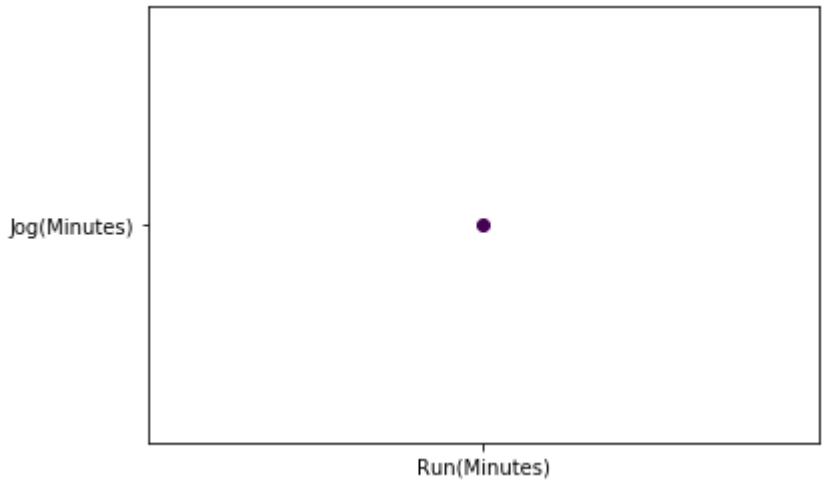
```
Index(['Run(Minutes)', 'Jog(Minutes)', 'Exercise(Minutes)', 'Hearbea
t',
       'Body_Temp(F)', 'Running_Distance(Km)', 'Jogging_Distance(K
m)',
       'Calories_Burnt(Kcal)', 'Weight_72'],
      dtype='object')
```

In [45]:

```
color = [1]
plt.scatter('Run(Minutes)','Jog(Minutes)',c = color)
```

Out[45]:

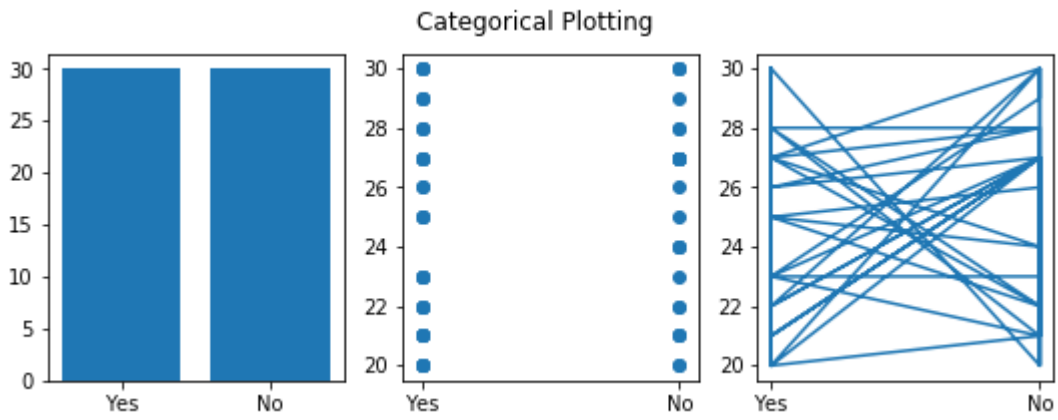
```
<matplotlib.collections.PathCollection at 0x123fd1710>
```



Plotting with categorical variables

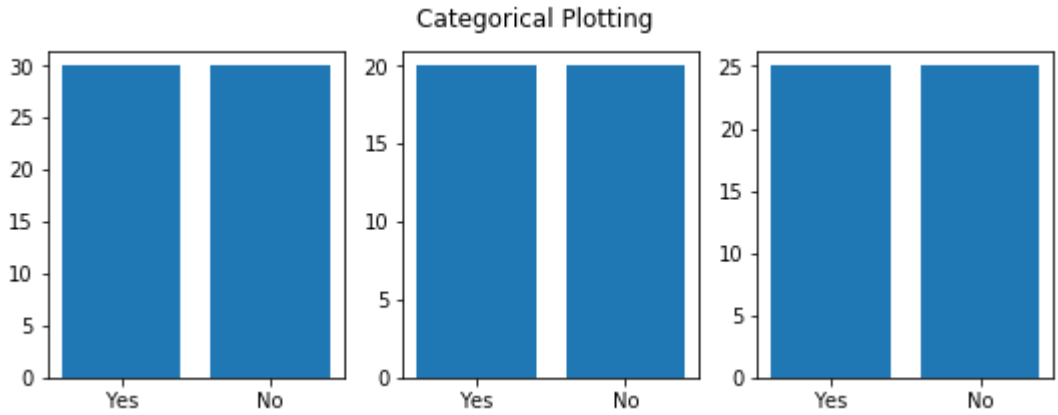
In [57]:

```
plt.figure(figsize=(9, 3))

plt.subplot(131)
plt.bar('Weight_72','Run(Minutes)' , data=data)
plt.subplot(132)
plt.scatter('Weight_72','Run(Minutes)' , data=data)
plt.subplot(133)
plt.plot('Weight_72','Run(Minutes)' , data=data)
plt.suptitle('Categorical Plotting')
plt.show()
```



In [67]:

```
plt.figure(figsize=(9, 3))

plt.subplot(131)
plt.bar('Weight_72','Run(Minutes)' , data=data)
plt.subplot(132)
plt.bar('Weight_72','Jog(Minutes)' , data=data)
plt.subplot(133)
plt.bar('Weight_72','Exercise(Minutes)' , data=data)
plt.suptitle('Categorical Plotting')
plt.show()
```



In [ ]:

In [ ]: