week2 combined

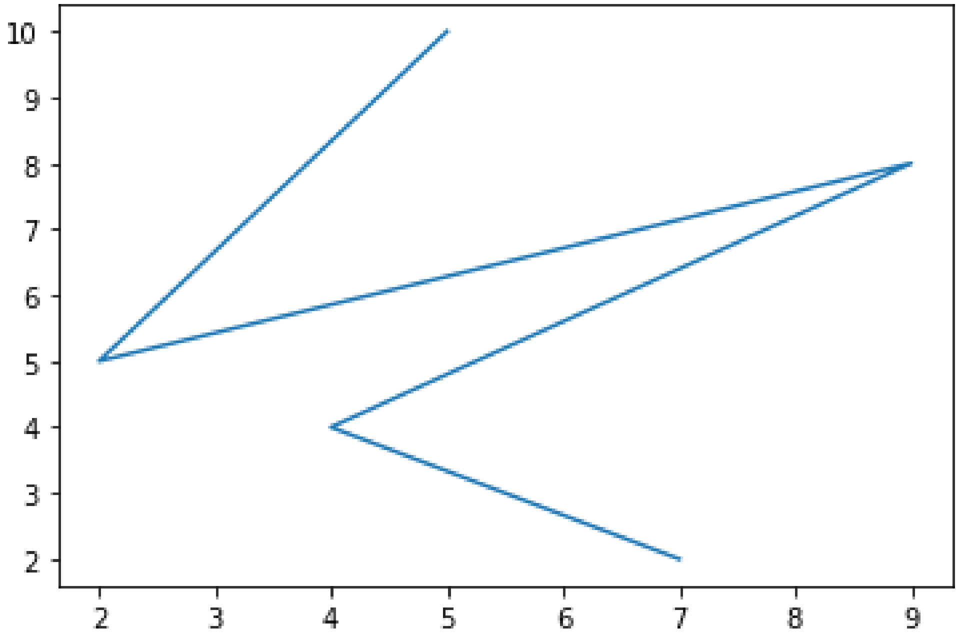
August 9, 2023

[1]:

```
import matplotlib.pyplot as plt
```

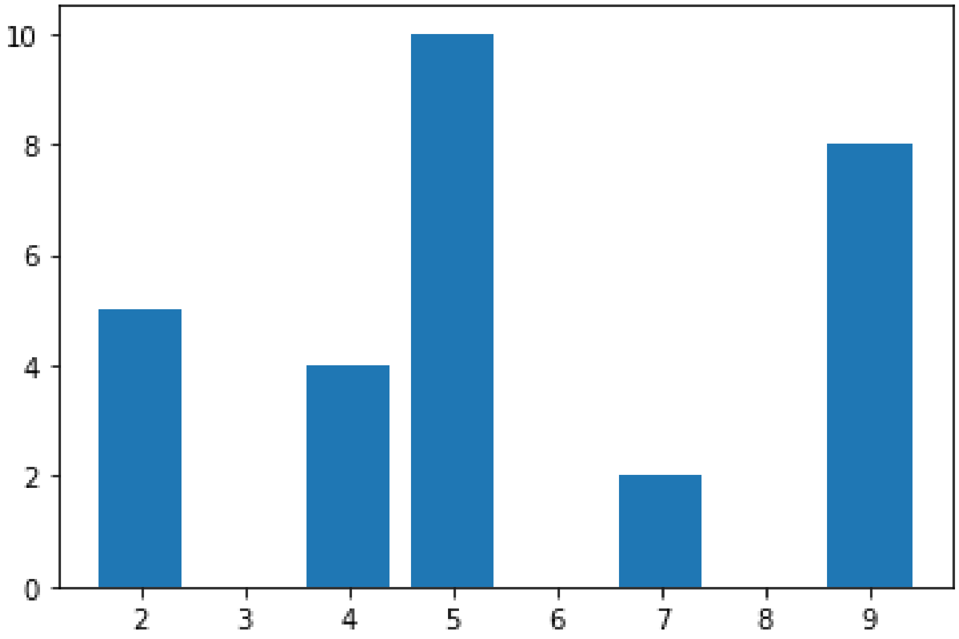
[2]:

```
# Line plot
x = [5, 2, 9, 4, 7]
y = [10, 5, 8, 4, 2]
plt.plot(x,y)
plt.show()
```



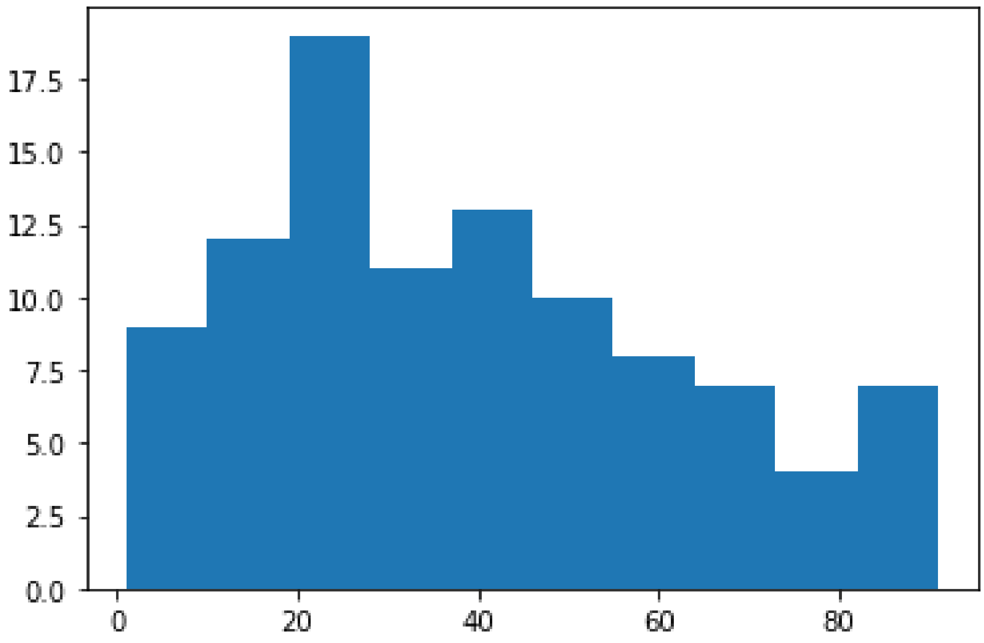
[3]:

```
# Bar Plot
x = [5, 2, 9, 4, 7]
y = [10, 5, 8, 4, 2]
plt.bar(x,y)
plt.show()
```



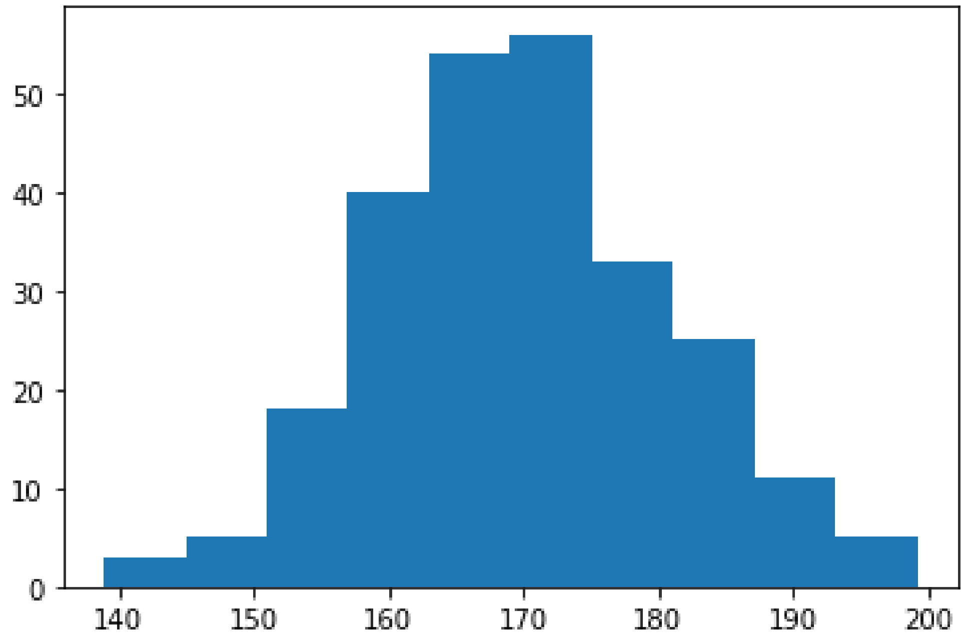
[4]:

```
# histogram
x = [1, 1, 2, 3, 3, 5, 7, 8, 9, 10,
     10, 11, 11, 13, 13, 15, 16, 17, 18, 18,
     18, 19, 20, 21, 21, 23, 24, 24, 25, 25,
     25, 25, 26, 26, 26, 27, 27, 27, 27, 27,
     29, 30, 30, 31, 33, 34, 34, 34, 35, 36,
     36, 37, 37, 38, 38, 39, 40, 41, 41, 42,
     43, 44, 45, 45, 46, 47, 48, 48, 49, 50,
     51, 52, 53, 54, 55, 55, 56, 57, 58, 60,
     61, 63, 64, 65, 66, 68, 70, 71, 72, 74,
     75, 77, 81, 83, 84, 87, 89, 90, 90, 91
     ]
plt.hist(x, bins=10)
plt.show()
```



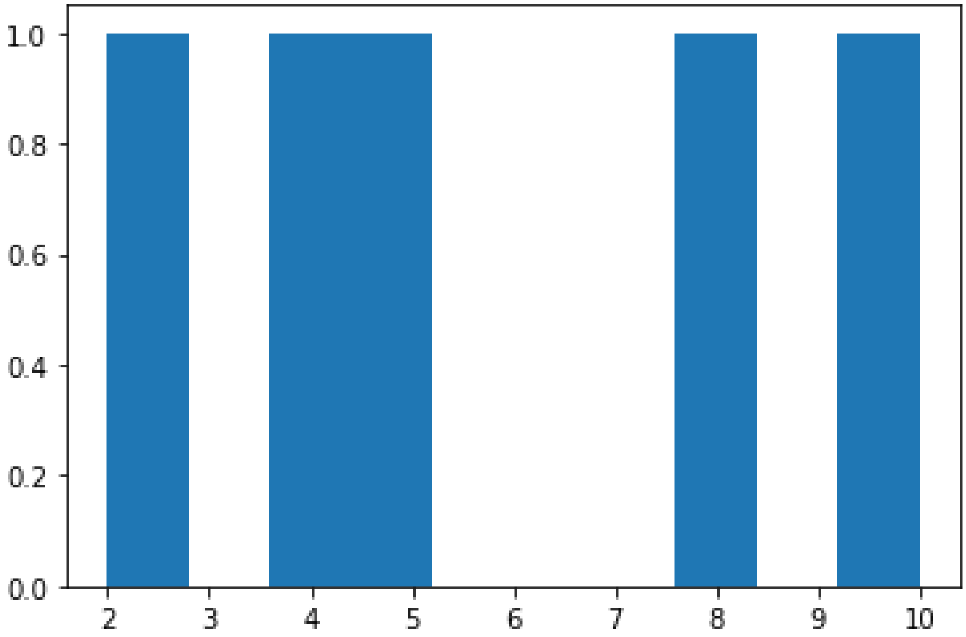
[5]:

```
#numPy to randomly generate an array with 250 values,
#where the values will concentrate around 170, and the standard deviation is 10.
 ↪
import numpy as np
x = np.random.normal(170,10,250)
plt.hist(x)
plt.show()
```



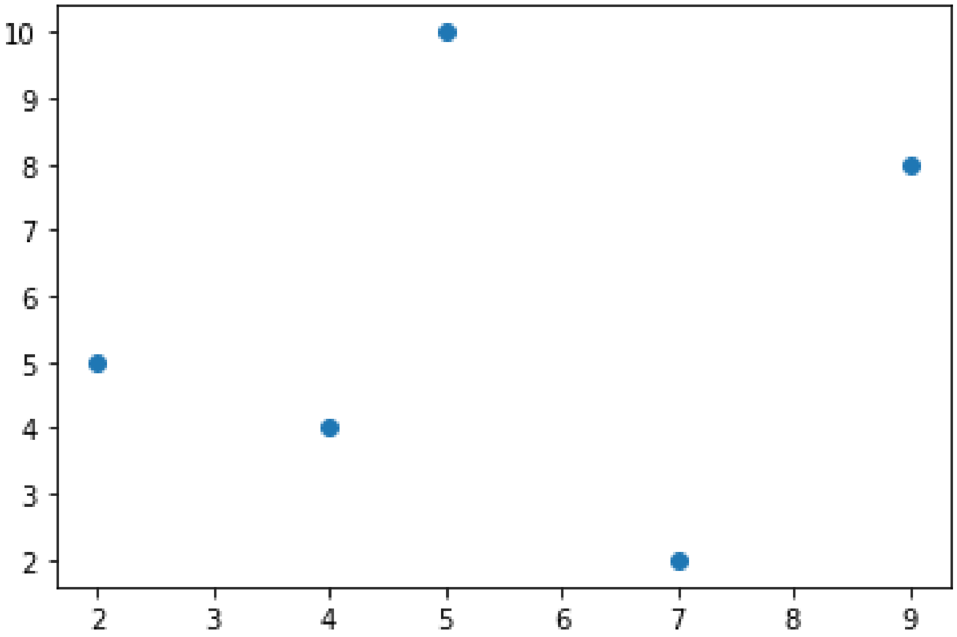
[6]:

```
y = [10, 5, 8, 4, 2]
plt.hist(y)
plt.show()
```



[7]:

```
# scatter plot
x = [5, 2, 9, 4, 7]
y = [10, 5, 8, 4, 2]
plt.scatter(x,y)
plt.show()
```



[8]:

```
# constants
from scipy import constants
print(constants.peta)
print(constants.mega)
print(constants.deka)
print(constants.deci)
print(constants.pico)
```

```
1000000000000000.0
1000000.0
10.0
0.1
1e-12
```

[9]:

```
# CSR (compressed sparse row)
import numpy as np
from scipy.sparse import csr_matrix
arr = np.array([0,0,0,0,0,1,1,0,2])
print(csr_matrix(arr))
```

```
  (0, 5)        1
  (0, 6)        1
  (0, 8)        2
```

[10]:

```
import numpy as np
from scipy.sparse import csr_matrix
```

```
arr = np.array([[0,0,0],[0,0,1],[1,0,2]])
print(csr_matrix(arr))
```

```
  (1, 2)        1
  (2, 0)        1
  (2, 2)        2
```

[11]:

```
import numpy as np
from scipy.sparse import csr_matrix
arr = np.array([[0,0,0],[0,0,1],[1,0,2]])
newarr = csr_matrix(arr).tocsc()
print(newarr)
```

```
  (2, 0)        1
  (1, 2)        1
  (2, 2)        2
```

[12]:

```
import pandas as pd
mydataset = {'cars':['BMW','Audi','Merc'],
            'passings':[3,7,2]}
myvar = pd.DataFrame(mydataset)
print(myvar)
```

```
   cars  passings
0   BMW         3
1  Audi         7
2  Merc         2
```

[13]:

```
# series
new_series = pd.Series([7,4,8,9,2])
print(new_series)
print('-------------')
print(new_series[2])
```

```
0    7
1    4
2    8
3    9
4    2
dtype: int64
-------------
8
```

[14]:

```
# more on series
new_series = pd.Series([10,20,30,40,50],index=['a','b','c','d','e'])
print(new_series)
print('-------------')
print(new_series.values)
print('-------------')
print(new_series['b'])
print('-------------')
print(new_series[['a','b','d']])
print('-------------')
print(new_series[new_series>25])
```

```
a    10
b    20
c    30
d    40
e    50
dtype: int64
-------------
[10 20 30 40 50]
-------------
20
-------------
a    10
b    20
d    40
dtype: int64
-------------
c    30
d    40
e    50
dtype: int64
```

[15]:

```
new_series2 = new_series[new_series>35]*2
print(new_series2)
```

```
d     80
e    100
dtype: int64
```

[16]:

```
# Data Frame
```

[17]:

```
import pandas as pd
data = {'calories':[100,250,900],
       'duration':[10,15,80]}
df = pd.DataFrame(data)
print(df)
```

```
   calories  duration
0       100        10
1       250        15
2       900        80
```

[18]:
```
# loc to locate row
print(df.loc[1])
```

```
calories    250
duration     15
Name: 1, dtype: int64
```

[19]:
```
# index
df = pd.DataFrame(data,index=['day 1','day 2','day 3'])
print(df)
```

```
       calories  duration
day 1       100        10
day 2       250        15
day 3       900        80
```

[20]:
```
# dataframe using list
list2 = [[0,1,2],[3,4,5],[6,7,8]]
df = pd.DataFrame(list2)
print(df)
print('--------')
df.columns = ['C1','C2','C3']
print(df)
```

```
   0  1  2
0  0  1  2
1  3  4  5
2  6  7  8
--------
   C1  C2  C3
0   0   1   2
1   3   4   5
2   6   7   8
```

[21]:
```
import pandas as pd
df=pd.DataFrame({
      'Country': ['Kazakhstan','Russia','Belarus','Ukraine'],
      'Population': [17.04,143.5,9.5,45.5],
      'Square':[2724902, 17125191,207600,603628],
       }, index= ['az','by','cv','dw']
     )
print(df)
```

```
       Country  Population    Square
az  Kazakhstan       17.04   2724902
by      Russia      143.50  17125191
cv     Belarus        9.50    207600
dw     Ukraine       45.50    603628
```

[22]:

```
print(df.columns)
print(df.index)
df.index.name = 'Country code'
print(df)
```

```
Index(['Country', 'Population', 'Square'], dtype='object')
Index(['az', 'by', 'cv', 'dw'], dtype='object')
                 Country  Population    Square
Country code
az            Kazakhstan       17.04   2724902
by                Russia      143.50  17125191
cv               Belarus        9.50    207600
dw               Ukraine       45.50    603628
```

[23]:

```
print(df.iloc[0])
```

```
Country       Kazakhstan
Population         17.04
Square           2724902
Name: az, dtype: object
```

[24]:

```
print(df.loc[['az','by'],'Population'])
```

```
Country code
az     17.04
by    143.50
Name: Population, dtype: float64
```

[25]:

```
import pandas as pd
import numpy as np

df = pd.DataFrame(np.arange(20).reshape(5,4), columns = ['A','B','C','D'])
print(df)
```

```
    A   B   C   D
0   0   1   2   3
1   4   5   6   7
2   8   9  10  11
3  12  13  14  15
4  16  17  18  19
```

[26]:

```
# slicing rows and columns
```

[27]:

```
df.loc[:,'B':'D']
```

[27]:

```
    B   C   D
0   1   2   3
1   5   6   7
```

```
2   9  10  11
3  13  14  15
4  17  18  19
```

[28]:

```
df.iloc[0:2,:]
```

[28]:

```
   A  B  C  D
0  0  1  2  3
1  4  5  6  7
```

[29]:

```
df=pd.DataFrame({
      'Country': ['Kazakhstan','Russia','Belarus','Ukraine'],
      'Population': [17.04,143.5,9.5,45.5],
      'Square':[2724902, 17125191,207600,603628],
       })
print(df)
```

```
      Country  Population    Square
0  Kazakhstan       17.04   2724902
1      Russia      143.50  17125191
2     Belarus        9.50    207600
3     Ukraine       45.50    603628
```

[30]:

```
# filtering
print([[df.Population > 10],'Country','Square'])
```

```
[[0     True
1     True
2    False
3     True
Name: Population, dtype: bool], 'Country', 'Square']
```

[31]:

```
# drop
df.drop(['Population'],axis='columns')
```

[31]:

```
      Country    Square
0  Kazakhstan   2724902
1      Russia  17125191
2     Belarus    207600
3     Ukraine    603628
```

[32]:

```
df.drop([2], axis= 'index')
```

[32]:

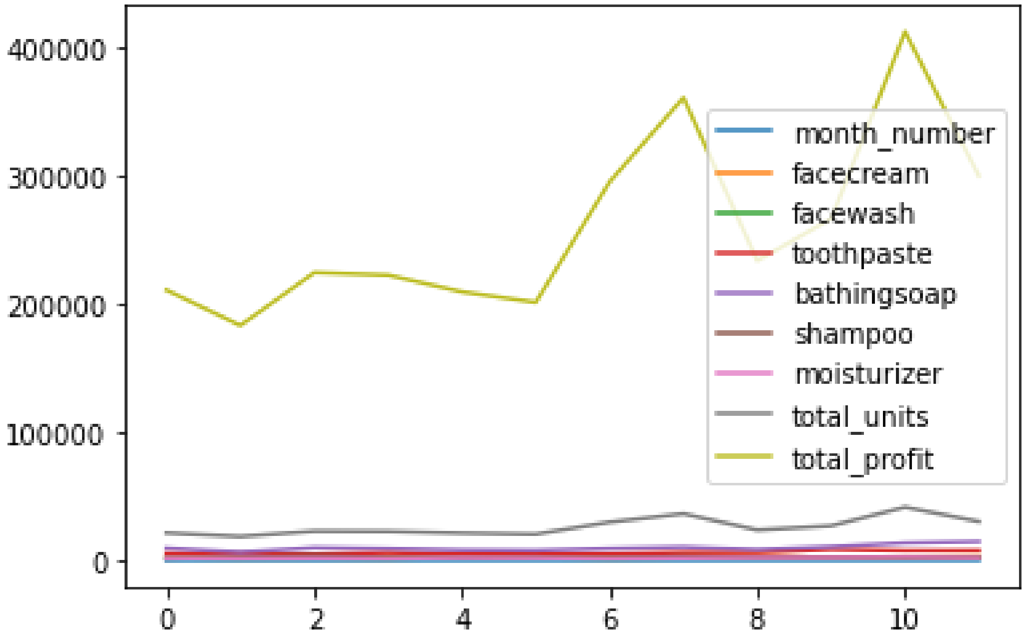
```
      Country  Population    Square
0  Kazakhstan       17.04   2724902
1      Russia      143.50  17125191
3     Ukraine       45.50    603628
```

```
[33]: # using pandas to read CSV
```

```
[34]: import pandas as pd
      df = pd.read_csv('company_sales_data.csv')
      print(df.to_string())
```

```
    month_number  facecream  facewash  toothpaste  bathingsoap  shampoo
moisturizer  total_units  total_profit
0              1       2500      1500        5200         9200     1200
1500        21100        211000
1              2       2630      1200        5100         6100     2100
1200        18330        183300
2              3       2140      1340        4550         9550     3550
1340        22470        224700
3              4       3400      1130        5870         8870     1870
1130        22270        222700
4              5       3600      1740        4560         7760     1560
1740        20960        209600
5              6       2760      1555        4890         7490     1890
1555        20140        201400
6              7       2980      1120        4780         8980     1780
1120        29550        295500
7              8       3700      1400        5860         9960     2860
1400        36140        361400
8              9       3540      1780        6100         8100     2100
1780        23400        234000
9             10       1990      1890        8300        10300     2300
1890        26670        266700
10            11       2340      2100        7300        13300     2400
2100        41280        412800
11            12       2900      1760        7400        14400     1800
1760        30020        300200
```

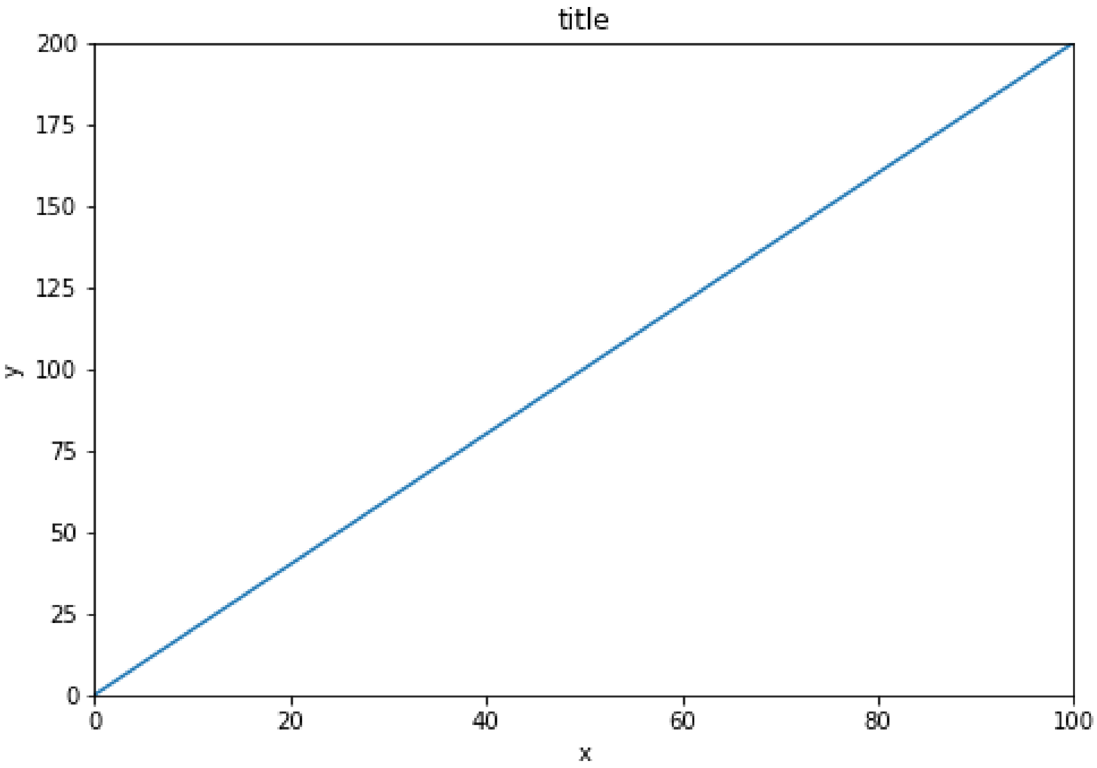
```
[35]: import matplotlib.pyplot as plt
      df.plot()
      plt.show()
```



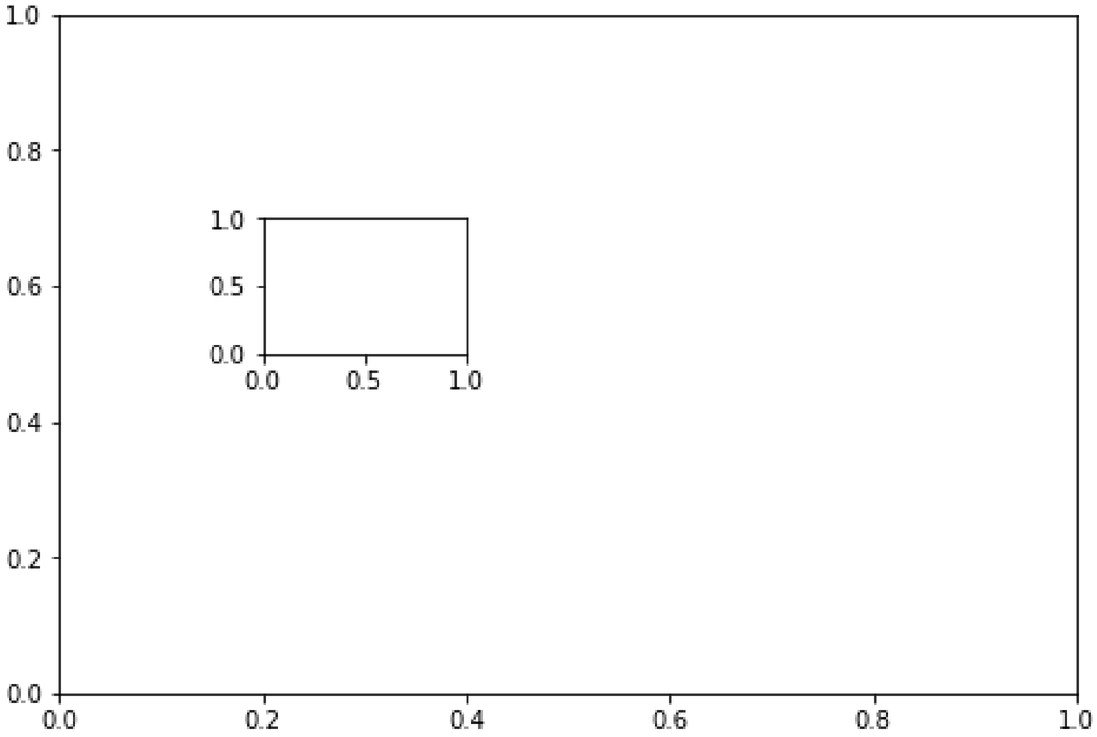
```
[36]: import matplotlib.pyplot as plt
      import numpy as np
```

```
[37]: #1. Follow along with these steps:
      #   a) Create a figure object called fig using plt.figure()
      #   b) Use add_axes to add an axis to the figure canvas at [0,0,1,1]. Call this␣
       ↪new axis ax.
      #   c) Plot (x,y) on that axes and set the labels and titles to match the plot␣
       ↪below:

      fig = plt.figure()
      ax = fig.add_axes([0,0,1,1])
      x = np.linspace(0,100,100)
      y = x*2

      ax.set_xlim([min(x),max(x)])
      ax.set_ylim([min(y),max(y)])
      ax.plot(x,y)
      ax.set_xlabel('x')
      ax.set_ylabel('y')
      ax.set_title('title')
```

```
[37]: Text(0.5, 1.0, 'title')
```



[38]:
```
# 2. Create a figure object and put two axes on it, ax1 and ax2.
#Located at [0,0,1,1] and [0.2,0.5,.2,.2] respectively. Now plot (x,y) on both␣
↪axes. And call your figure object to show it.
fig = plt.figure()
ax1 = fig.add_axes([0,0,1,1])
ax2 = fig.add_axes([0.2,0.5,.2,.2])
```



[39]:
```
# 3) Use the company sales dataset csv file, read Total profit of all months␣
↪and show it using a line plot
```

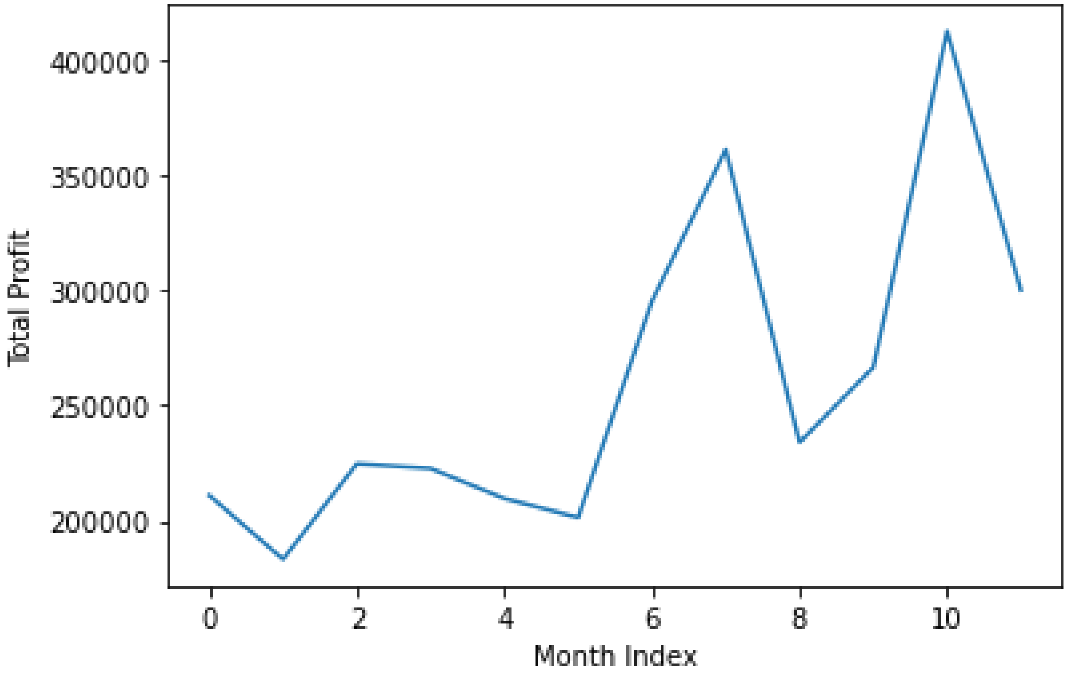
[40]:
```
import pandas as pd

df = pd.read_csv('company_sales_data.csv')
print(df.to_string())
```

```
    month_number  facecream  facewash  toothpaste  bathingsoap  shampoo
moisturizer  total_units  total_profit
0              1       2500      1500        5200         9200     1200
1500        21100        211000
1              2       2630      1200        5100         6100     2100
1200        18330        183300
2              3       2140      1340        4550         9550     3550
1340        22470        224700
3              4       3400      1130        5870         8870     1870
1130        22270        222700
4              5       3600      1740        4560         7760     1560
1740        20960        209600
5              6       2760      1555        4890         7490     1890
1555        20140        201400
6              7       2980      1120        4780         8980     1780
```

```
1120          29550        295500
7               8        3700        1400          5860          9960       2860
1400          36140        361400
8               9        3540        1780          6100          8100       2100
1780          23400        234000
9              10        1990        1890          8300         10300       2300
1890          26670        266700
10             11        2340        2100          7300         13300       2400
2100          41280        412800
11             12        2900        1760          7400         14400       1800
1760          30020        300200
```

[41]:

```
y = df['total_profit'] # store total profit in an array
x = df.index.values
plt.xlabel('Month Index')
plt.ylabel('Total Profit')
plt.plot(x,y)
plt.show()
```

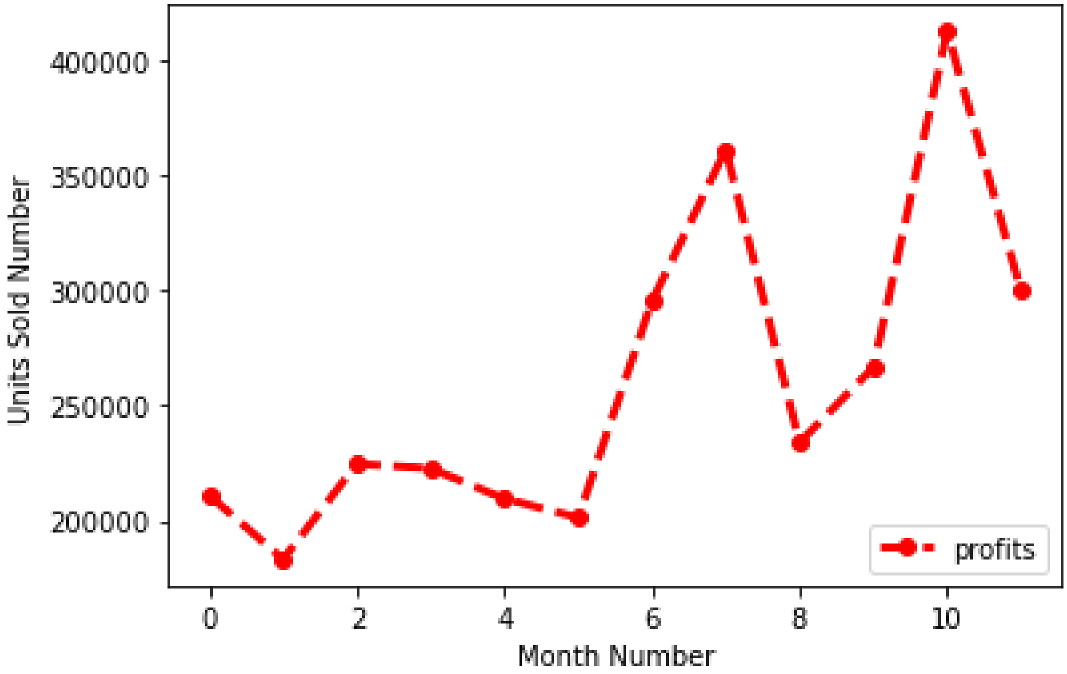


[42]:

```
# 4) Use the company sales dataset csv file, get total profit of all months
#and show line plot with the following Style properties.
#    a. Line Style dotted and Line-color should be red
#    b. Show legend at the lower right location.
#    c. X label name = Month Number
#    d. Y label name = Sold units number
#     e. Add a circle marker.
#     f. Line marker color as read
#     g. Line width should be 3
```

[43]:

```
# Line Style dotted and Line-color should be red
plt.plot(x,y,linewidth =3, linestyle='dashed',color='red', marker='o')
plt.xlabel('Month Number')
plt.ylabel('Units Sold Number')
plt.legend(["profits"], loc ="lower right")
plt.show()
```



[ ]: