Data Visualization

In [ ]:

In [133…
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

$$y =2x$$

In [134…
```
x = np.linspace(0, 1, 100)
x
```

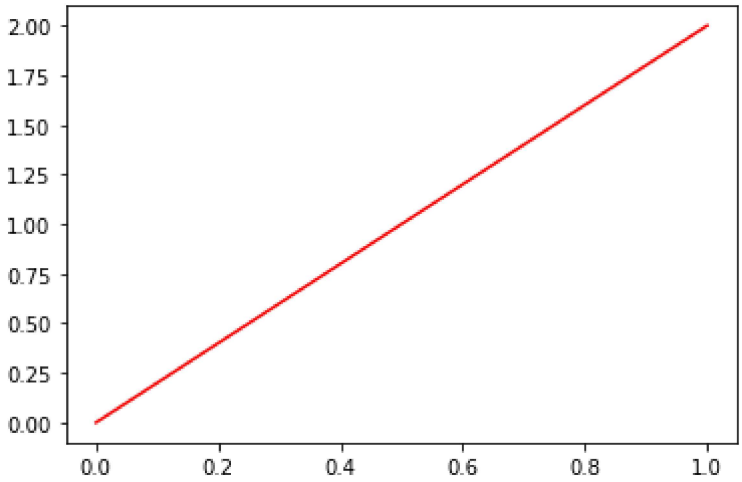
Out[134]:
```
array([0.        , 0.01010101, 0.02020202, 0.03030303, 0.04040404,
       0.05050505, 0.06060606, 0.07070707, 0.08080808, 0.09090909,
       0.1010101 , 0.11111111, 0.12121212, 0.13131313, 0.14141414,
       0.15151515, 0.16161616, 0.17171717, 0.18181818, 0.19191919,
       0.2020202 , 0.21212121, 0.22222222, 0.23232323, 0.24242424,
       0.25252525, 0.26262626, 0.27272727, 0.28282828, 0.29292929,
       0.3030303 , 0.31313131, 0.32323232, 0.33333333, 0.34343434,
       0.35353535, 0.36363636, 0.37373737, 0.38383838, 0.39393939,
       0.4040404 , 0.41414141, 0.42424242, 0.43434343, 0.44444444,
       0.45454545, 0.46464646, 0.47474747, 0.48484848, 0.49494949,
       0.50505051, 0.51515152, 0.52525253, 0.53535354, 0.54545455,
       0.55555556, 0.56565657, 0.57575758, 0.58585859, 0.5959596 ,
       0.60606061, 0.61616162, 0.62626263, 0.63636364, 0.64646465,
       0.65656566, 0.66666667, 0.67676768, 0.68686869, 0.6969697 ,
       0.70707071, 0.71717172, 0.72727273, 0.73737374, 0.74747475,
       0.75757576, 0.76767677, 0.77777778, 0.78787879, 0.7979798 ,
       0.80808081, 0.81818182, 0.82828283, 0.83838384, 0.84848485,
       0.85858586, 0.86868687, 0.87878788, 0.88888889, 0.8989899 ,
       0.90909091, 0.91919192, 0.92929293, 0.93939394, 0.94949495,
       0.95959596, 0.96969697, 0.97979798, 0.98989899, 1.        ])
```

In [135…
```
y =2*x
y
```

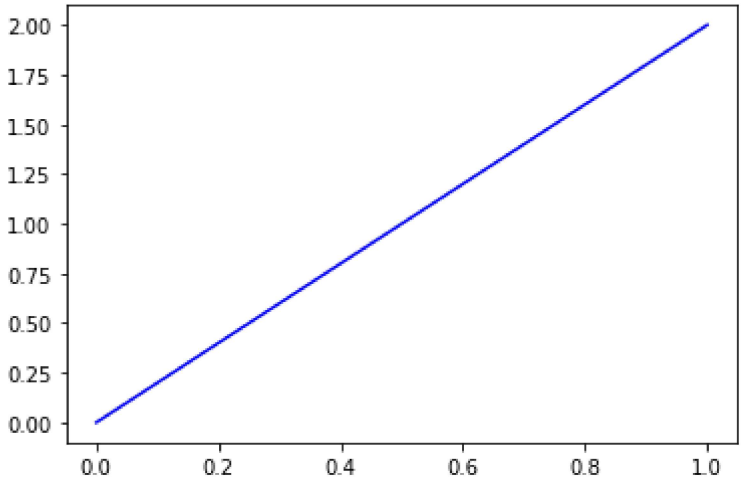
Out[135]:
```
array([0.        , 0.02020202, 0.04040404, 0.06060606, 0.08080808,
       0.1010101 , 0.12121212, 0.14141414, 0.16161616, 0.18181818,
       0.2020202 , 0.22222222, 0.24242424, 0.26262626, 0.28282828,
       0.3030303 , 0.32323232, 0.34343434, 0.36363636, 0.38383838,
       0.4040404 , 0.42424242, 0.44444444, 0.46464646, 0.48484848,
       0.50505051, 0.52525253, 0.54545455, 0.56565657, 0.58585859,
       0.60606061, 0.62626263, 0.64646465, 0.66666667, 0.68686869,
       0.70707071, 0.72727273, 0.74747475, 0.76767677, 0.78787879,
       0.80808081, 0.82828283, 0.84848485, 0.86868687, 0.88888889,
       0.90909091, 0.92929293, 0.94949495, 0.96969697, 0.98989899,
       1.01010101, 1.03030303, 1.05050505, 1.07070707, 1.09090909,
       1.11111111, 1.13131313, 1.15151515, 1.17171717, 1.19191919,
       1.21212121, 1.23232323, 1.25252525, 1.27272727, 1.29292929,
       1.31313131, 1.33333333, 1.35353535, 1.37373737, 1.39393939,
       1.41414141, 1.43434343, 1.45454545, 1.47474747, 1.49494949,
       1.51515152, 1.53535354, 1.55555556, 1.57575758, 1.5959596 ,
       1.61616162, 1.63636364, 1.65656566, 1.67676768, 1.6969697 ,
       1.71717172, 1.73737374, 1.75757576, 1.77777778, 1.7979798 ,
       1.81818182, 1.83838384, 1.85858586, 1.87878788, 1.8989899 ,
       1.91919192, 1.93939394, 1.95959596, 1.97979798, 2.        ])
```

In [136…
```
plt.plot(x, y, 'r')
# plt.show()
```

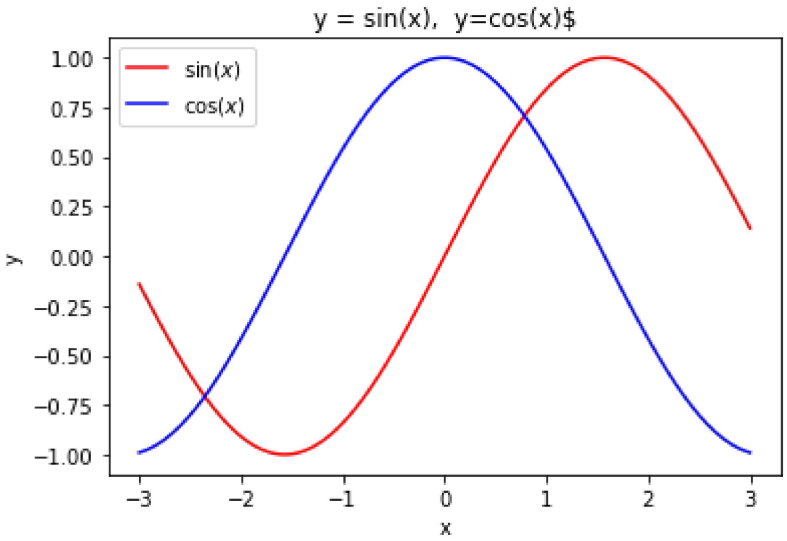
Out[136]:
```
[<matplotlib.lines.Line2D at 0x1de58af9250>]
```



In [137…
```
plt.plot(x, y, 'b')
plt.show()
```



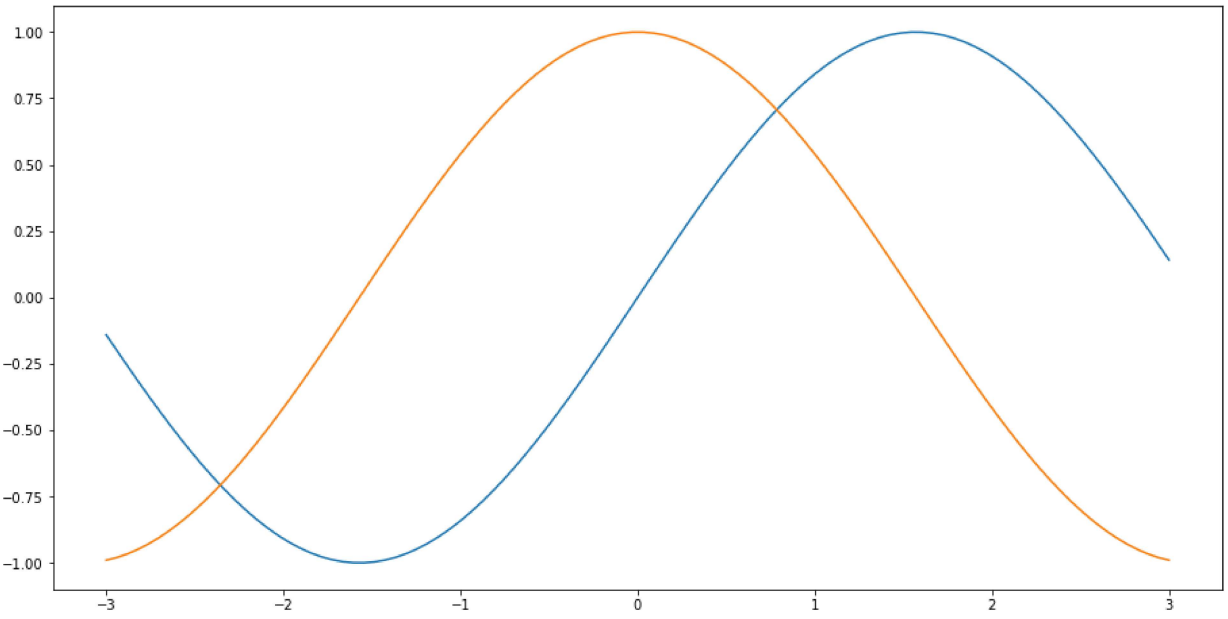
In [138…
```
x1 = np.linspace(-3, 3, 100)

y1 = np.sin(x1)
y2 = np.cos(x1)

plt.plot(x1, y1, 'r')
plt.plot(x1, y2, 'b')
plt.xlabel('x')
plt.ylabel('y')
plt.title("y = sin(x),  y=cos(x)$")
plt.legend(['$\sin(x)$', '$\cos(x)$'], loc=2)
# plt.ylim(-0.75, 0.75)
# plt.xlim(-2, 2)
plt.savefig('first_graph.png')
plt.show()
```



Creating Figures and Subplots

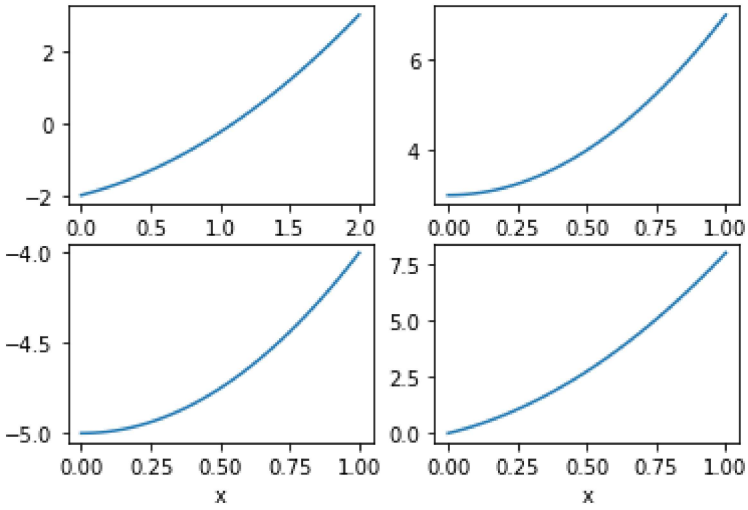
In [139…
```
fig =plt.figure(figsize=(12, 6))
ax = fig.add_axes([0, 0, 1, 1])

ax.plot(x1, y1)
ax.plot(x1, y2)
```

Out[139]:
```
[<matplotlib.lines.Line2D at 0x1de58c109a0>]
```



In [140…
```
yy1 = 3*x**2 +2*x -2
yy2 = 4*x**2 +3
yy3 = x**2 -5
yy4 = 5*x**2 +3*x

fig1, ax = plt.subplots(2, 2)

ax[0,0].plot(y, yy1)
ax[0,0].set_xlabel('x')
```

```
ax[0,1].plot(x, yy2)
ax[0,1].set_xlabel('x')

ax[1, 0].plot(x, yy3)
ax[1,0].set_xlabel('x')

ax[1, 1].plot(x, yy4)
ax[1,1].set_xlabel('x')

# fig.subplots_adjust(left=0.5, bottom=0.7, right=0.9, top=0.7, wspace=0.8, hspace=0.2
plt.show()
```



In [141…
```
import pandas as pd
```

In [142…
```
chdata = pd.read_csv('church_attendance.csv')
audata = pd.read_csv('autos_mpg.csv')
persondata = pd.read_csv('person.csv')
```

In [143…
```
audata.head()
```

Out[143]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino

In [144…
```
chdata
```

Out[144]:

	First Name	Last Name	Worker_Status	House Holde Number	Service attended
0	Daniel	Johnson	worker	5	15
1	Blessing	Madu	worker	3	20
2	Makanjuola	Adekunmi	worker	2	14
3	John	James	Non worker	6	21
4	Jeremiah	Akindele	worker	7	25

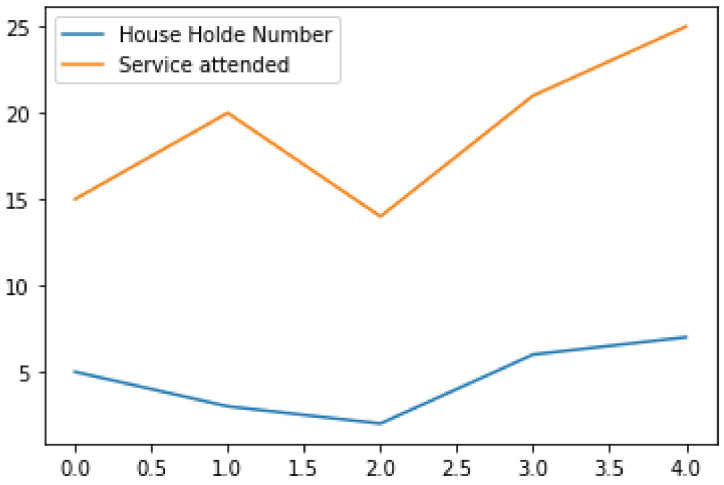
In [145…
```
persondata
```

Out[145]:

	Name	Age	Gender
0	John	34	male
1	sam	32	male
2	reena	24	female
3	romio	35	male
4	julie	23	female
5	dasie	26	female
6	bob	26	male
7	carl	28	male
8	Jessica	46	female
9	tom	65	male

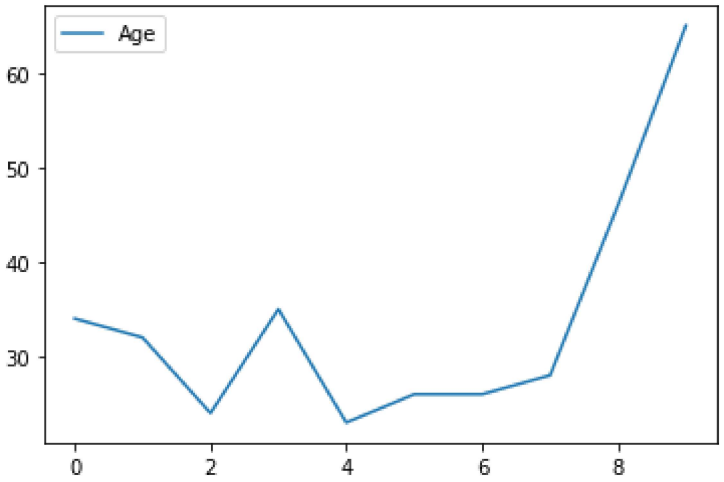
In [146…
```
##church attendance
chdata.plot()
```

Out[146]:
```
<AxesSubplot:>
```



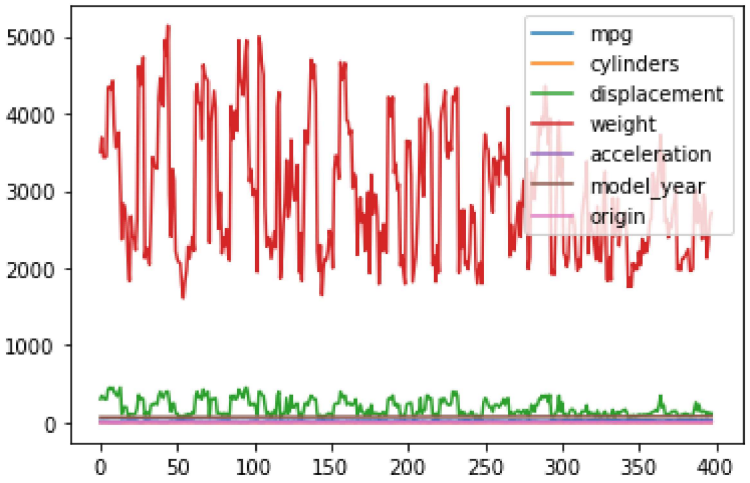
In [147…
```
persondata.plot()
```

Out[147]:
```
<AxesSubplot:>
```



In [148…
```
audata.plot()
```

Out[148]:
```
<AxesSubplot:>
```



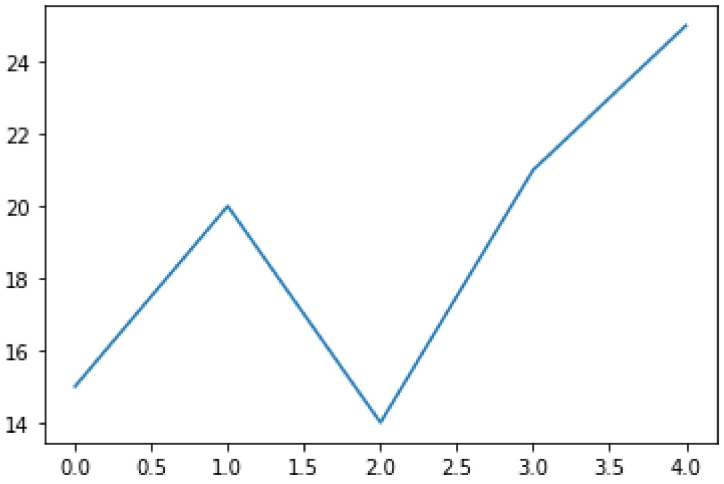
In [149…
```
#church data
chdata.describe()
```

Out[149]:

	House Holde Number	Service attended
count	5.000000	5.000000
mean	4.600000	19.000000
std	2.073644	4.527693
min	2.000000	14.000000
25%	3.000000	15.000000
50%	5.000000	20.000000
75%	6.000000	21.000000
max	7.000000	25.000000

In [150…
```
plt.plot(chdata['Service attended'])
```

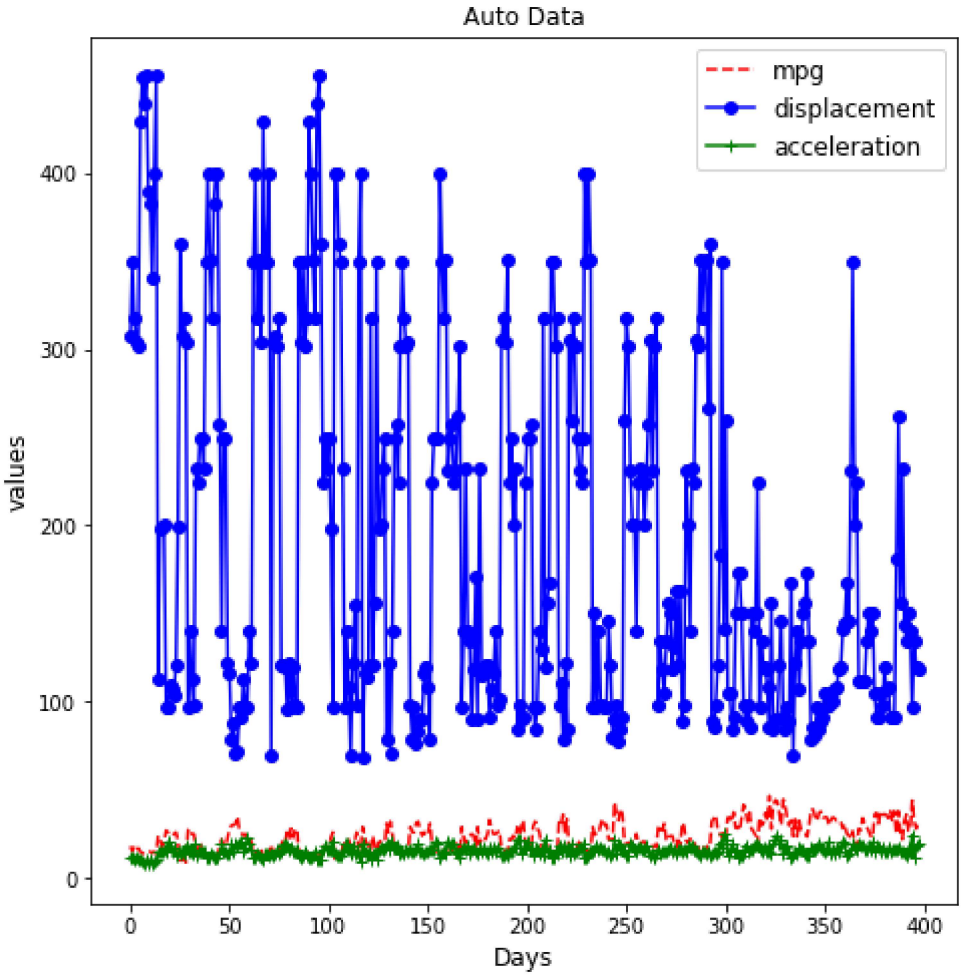
Out[150]:
```
[<matplotlib.lines.Line2D at 0x1de4b60f940>]
```



In [151…
```
audata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64  
 2   displacement  398 non-null    float64
 3   horsepower    398 non-null    object 
 4   weight        398 non-null    int64  
 5   acceleration  398 non-null    float64
 6   model_year    398 non-null    int64  
 7   origin        398 non-null    int64  
 8   car_name      398 non-null    object 
dtypes: float64(3), int64(4), object(2)
memory usage: 28.1+ KB
```

In [152…
```
font =12
fig = plt.figure(figsize=(6, 6))
axx= fig.add_axes([0, 0, 1, 1])
axx.plot(audata['mpg'], '--r')
axx.plot(audata['displacement'], '-ob')
axx.plot(audata['acceleration'], '-+g')
axx.legend(['mpg', 'displacement', 'acceleration'], fontsize = font)
axx.set_xlabel('Days', fontsize= font)
axx.set_ylabel('values', fontsize= font)
axx.set_title('Auto Data', fontsize= font)
fig.savefig("auto.png")
plt.show()
```



In [153…
```
## Assignment: Add height, weight  to the person data and then try to visualize Age, h
```

In [154…
```
persondata

#add height, weight
```

Out[154]:

	Name	Age	Gender
0	John	34	male
1	sam	32	male
2	reena	24	female
3	romio	35	male
4	julie	23	female
5	dasie	26	female
6	bob	26	male
7	carl	28	male
8	Jessica	46	female
9	tom	65	male

In [155…
```
arr1= np.array([5.7, 5.6, 6.5, 6.1, 4.5, 5.9, 5.8, 5.8, 5.6, 4.9]).reshape((-1,1))
```

```
arr2 =np.array([50, 55, 56, 65, 75, 60, 80, 51, 52, 53]).reshape((-1,1))
arrdata =np.hstack((arr1, arr2))
```

In [156…
```
twodata = pd.DataFrame(arrdata, columns = ["Height", "Weight"])
```

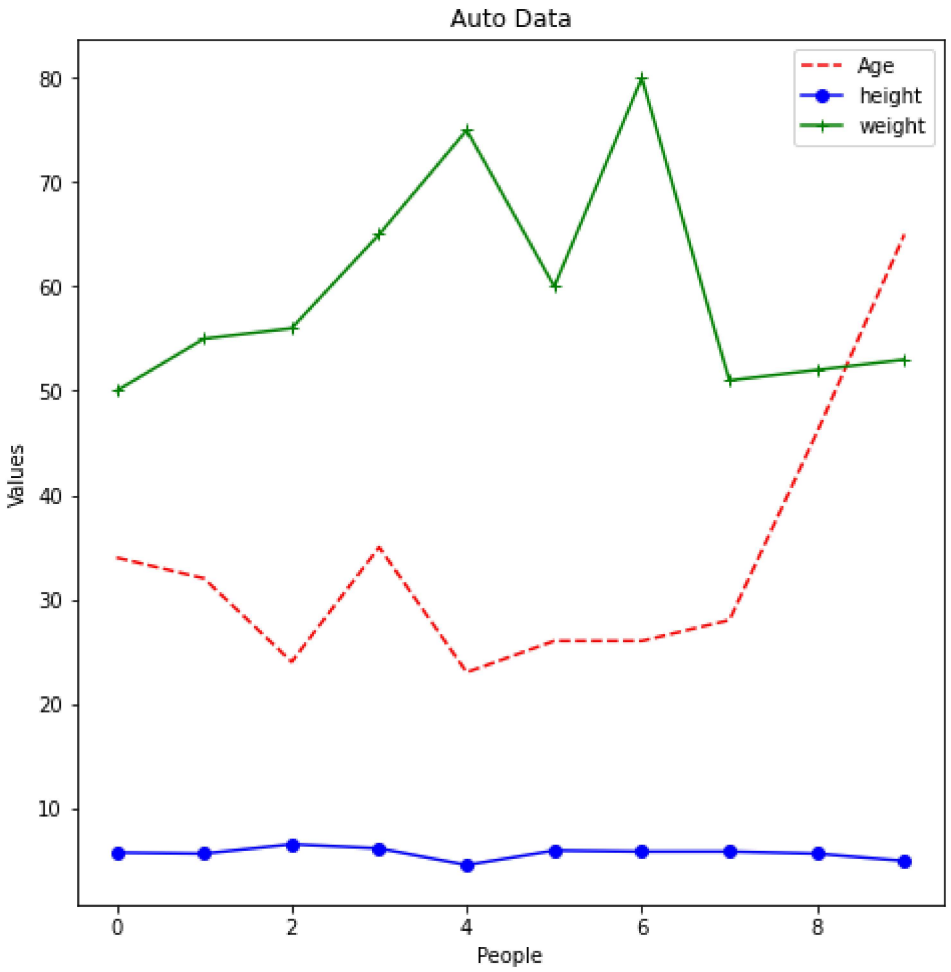
In [157…
```
twodata
```

Out[157]:

	Height	Weight
0	5.7	50.0
1	5.6	55.0
2	6.5	56.0
3	6.1	65.0
4	4.5	75.0
5	5.9	60.0
6	5.8	80.0
7	5.8	51.0
8	5.6	52.0
9	4.9	53.0

In [158…
```
#merge the data
persondata['height'] = twodata['Height']
persondata['weight'] = twodata['Weight']
persondata.to_csv('person_update.csv')
```

In [159…
```
fig = plt.figure(figsize=(6, 6))
axx= fig.add_axes([0, 0, 1, 1])
axx.plot(persondata['Age'], '--r')
axx.plot(persondata['height'], '-ob')
axx.plot(persondata['weight'], '-+g')
axx.legend(['Age', 'height', 'weight'])
axx.set_xlabel('People')
axx.set_ylabel('Values')
axx.set_title('Auto Data')
plt.savefig("person.png")
plt.show()
```



Seaborn

In [160…
```
import seaborn as sns
```

In [161…
```
onengghdata = pd.read_excel("oneNGGHdata.xlsx")
```

In [162…
```
onengghdata
```

Out[162]:

	dept	level	training	years	produce	salary
0	ACC	some college	2	7	100	5000
1	HR	associate	3	3	200	7000
2	IT	high school	3	5	180	6500
3	RD	masters	5	3	170	6100
4	MKT	high school	3	4	175	6150

In [163…
```
onengghdata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5 entries, 0 to 4
Data columns (total 6 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   dept      5 non-null      object
 1   level     5 non-null      object
 2   training  5 non-null      int64
 3   years     5 non-null      int64
 4   produce   5 non-null      int64
 5   salary    5 non-null      int64
dtypes: int64(4), object(2)
memory usage: 368.0+ bytes
```

In [164…
```
onengghdata.describe()
```

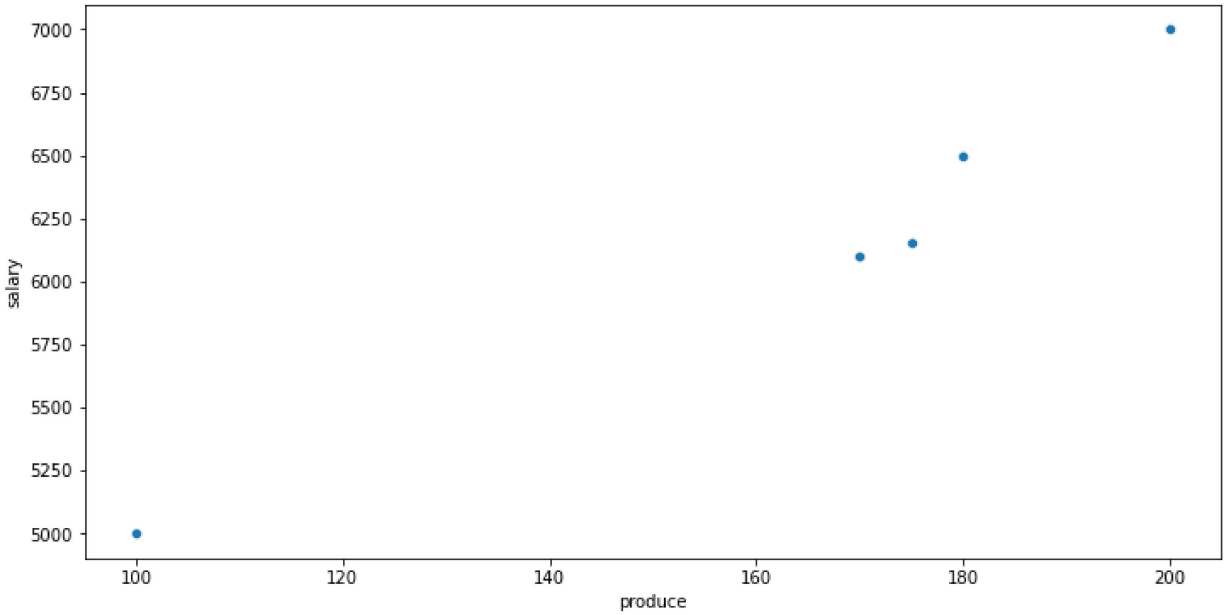
Out[164]:

	training	years	produce	salary
count	5.000000	5.00000	5.000000	5.000000
mean	3.200000	4.40000	165.000000	6150.000000
std	1.095445	1.67332	38.078866	736.545993
min	2.000000	3.00000	100.000000	5000.000000
25%	3.000000	3.00000	170.000000	6100.000000
50%	3.000000	4.00000	175.000000	6150.000000
75%	3.000000	5.00000	180.000000	6500.000000
max	5.000000	7.00000	200.000000	7000.000000

Quantitative Data

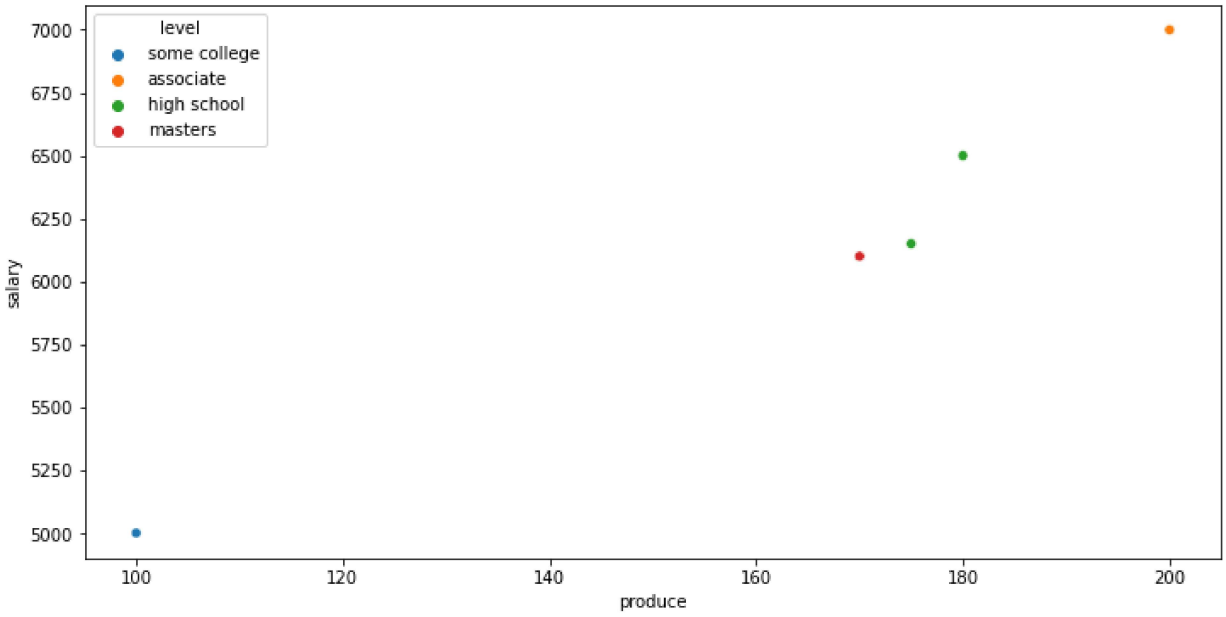
In [165…
```
plt.figure(figsize=(12,6))
sns.scatterplot(x='produce', y = 'salary', data=onengghdata)
```

Out[165]:
```
<AxesSubplot:xlabel='produce', ylabel='salary'>
```



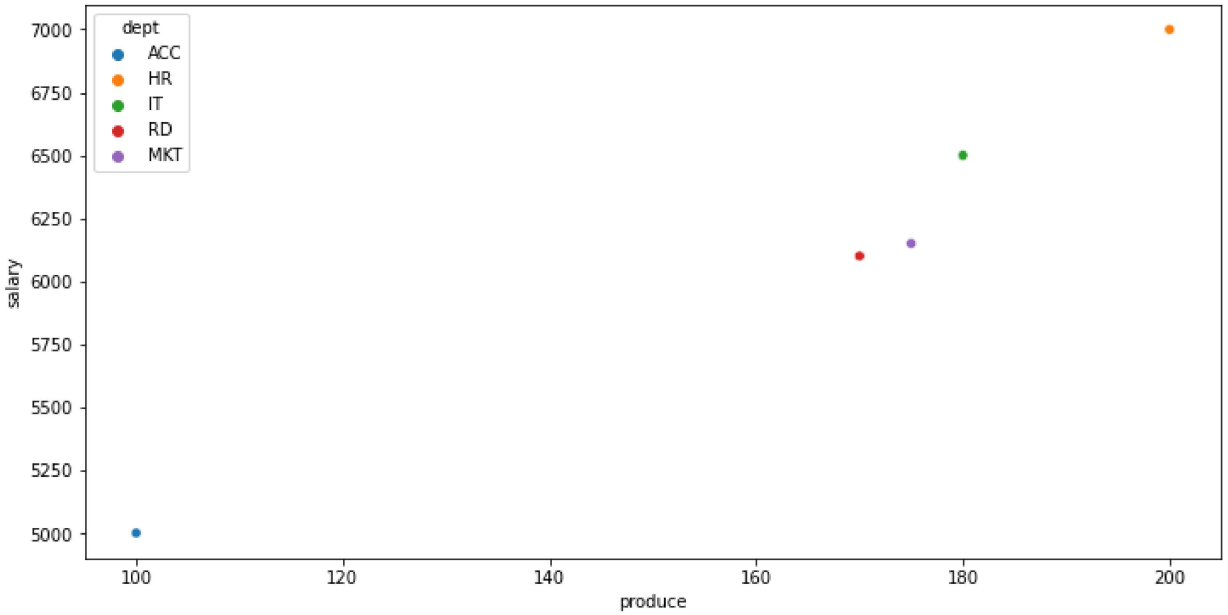
In [166…
```
plt.figure(figsize=(12,6))
sns.scatterplot(x='produce', y = 'salary', data=onengghdata, hue='level')
```

Out[166]:
```
<AxesSubplot:xlabel='produce', ylabel='salary'>
```



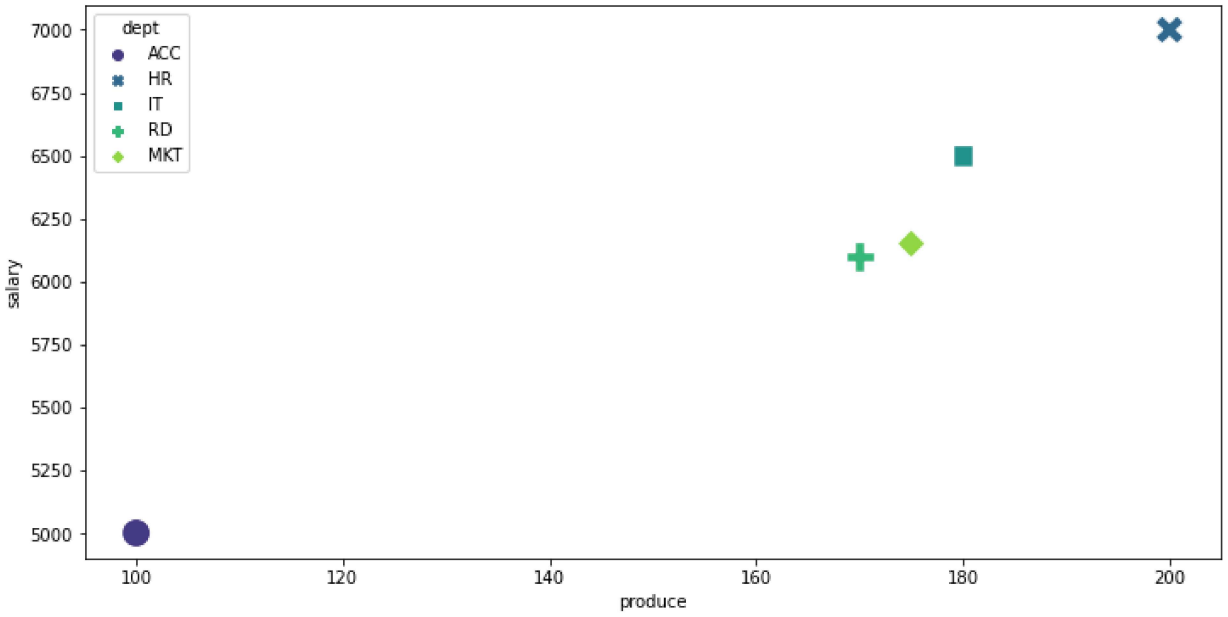
In [167…
```
plt.figure(figsize=(12,6))
sns.scatterplot(x='produce', y = 'salary', data=onengghdata, hue='dept')
```

Out[167]:
```
<AxesSubplot:xlabel='produce', ylabel='salary'>
```



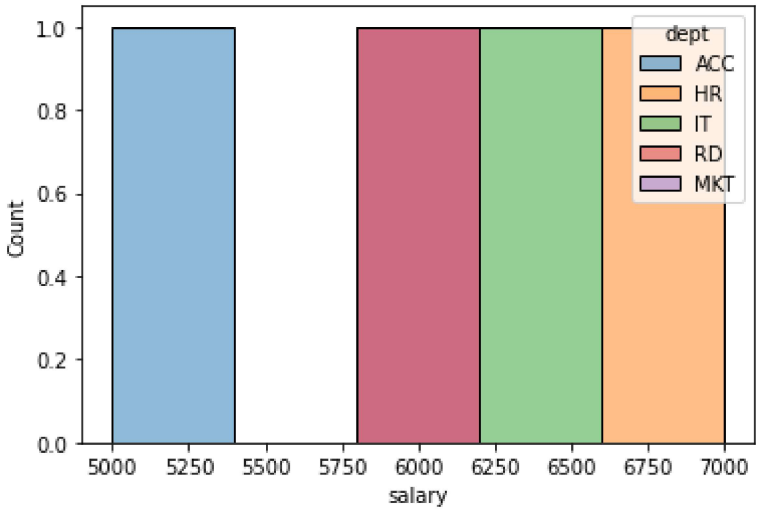
In [168…
```
plt.figure(figsize=(12,6))
sns.scatterplot(x='produce', y = 'salary', data=onengghdata, hue='dept', palette='viri
```

Out[168]:
```
<AxesSubplot:xlabel='produce', ylabel='salary'>
```



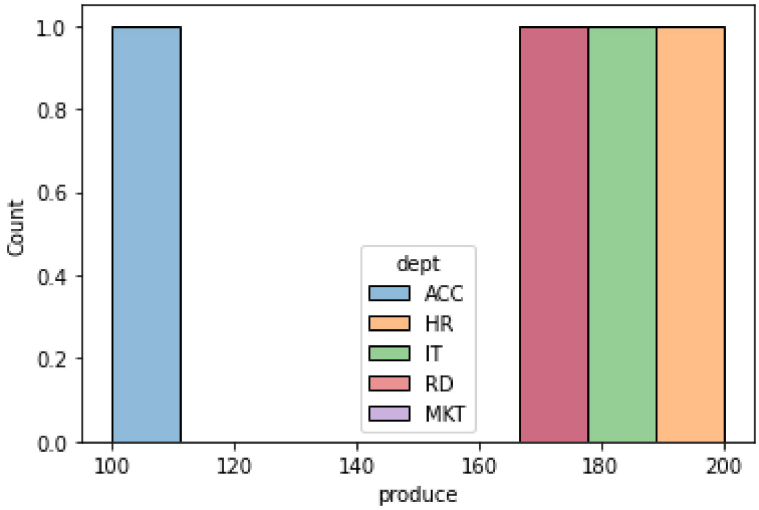
In [169…
```
sns.histplot(x='salary', data=onengghdata, hue= 'dept')
```

Out[169]:
```
<AxesSubplot:xlabel='salary', ylabel='Count'>
```



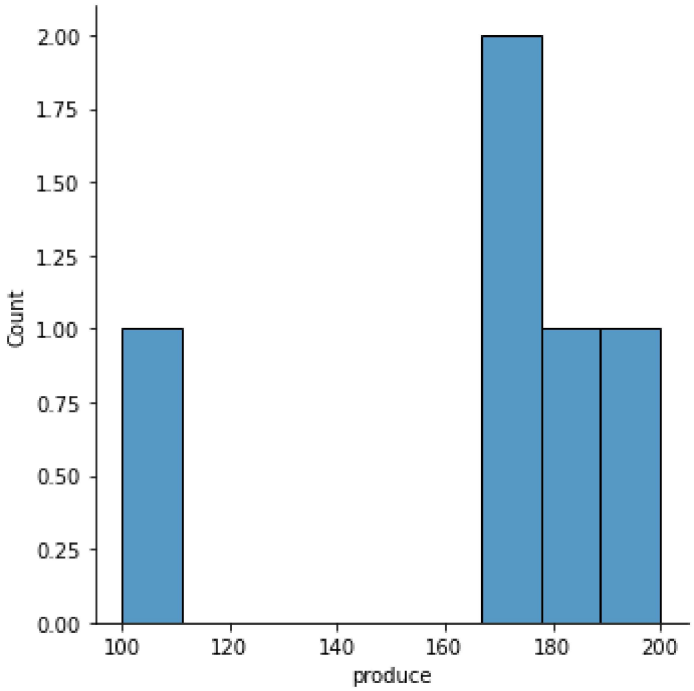
In [170…
```
sns.histplot(x='produce', data=onengghdata, hue = 'dept')
```

Out[170]:
```
<AxesSubplot:xlabel='produce', ylabel='Count'>
```



In [171…
```
sns.displot(x='produce', data=onengghdata)
```

Out[171]:
```
<seaborn.axisgrid.FacetGrid at 0x1de5a53d280>
```

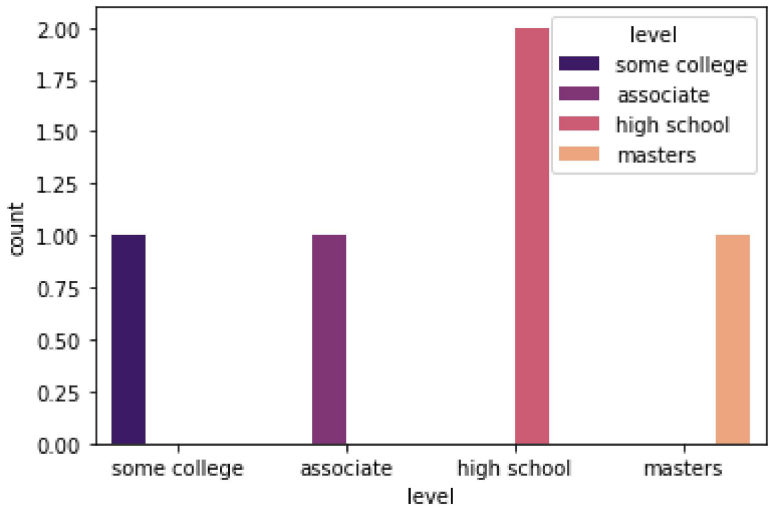


In [ ]:

Categorical data

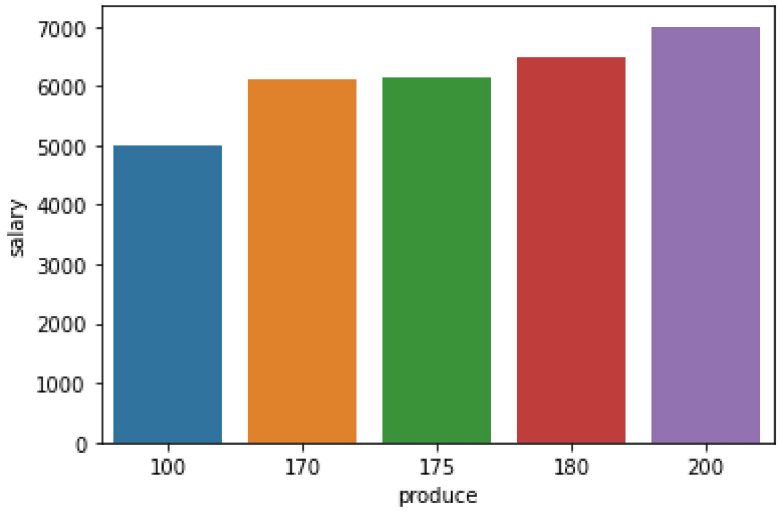
In [172…
```
sns.countplot(x='level', data=onengghdata, hue = 'level', palette = 'magma')
```

Out[172]:
```
<AxesSubplot:xlabel='level', ylabel='count'>
```



In [173…
```
sns.barplot(x='produce', y='salary', data=onengghdata)
```

Out[173]:
```
<AxesSubplot:xlabel='produce', ylabel='salary'>
```



In [174…
```
##Instructions
##Use scatter plot to show the relationship between height and weight
##Use barplot to show age against gender
```

In [179…
```
persondata2 = pd.read_csv('person_update.csv')
```

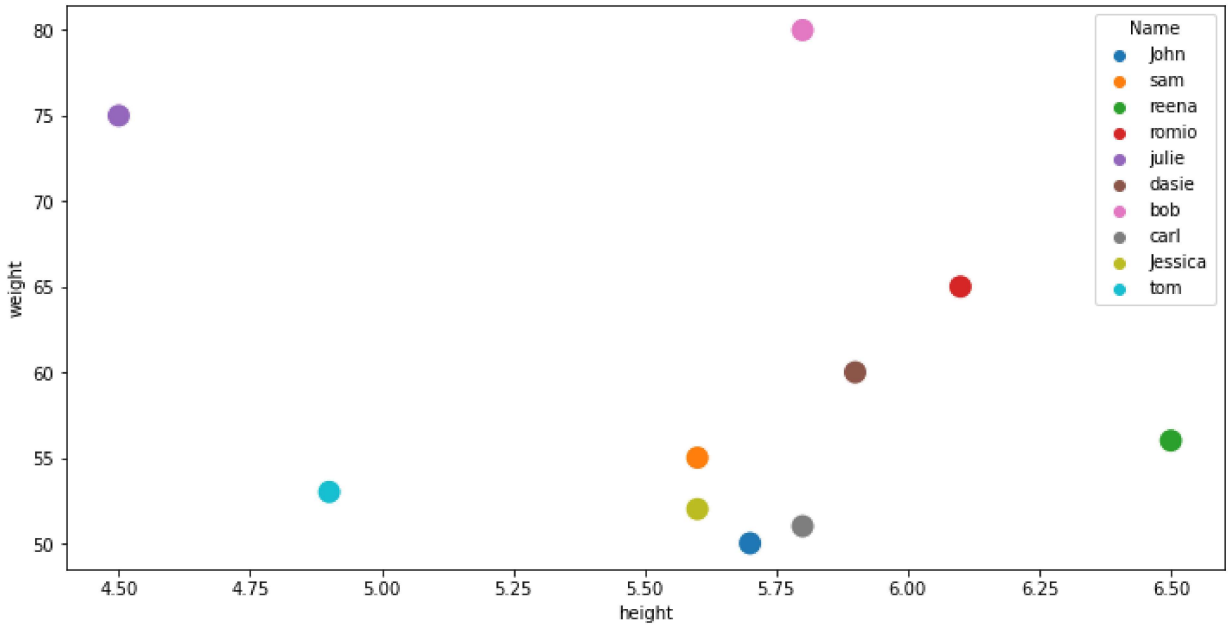
In [180…
```
persondata2
```

Out[180]:

	Unnamed: 0	Name	Age	Gender	height	weight
0	0	John	34	male	5.7	50.0
1	1	sam	32	male	5.6	55.0
2	2	reena	24	female	6.5	56.0
3	3	romio	35	male	6.1	65.0
4	4	julie	23	female	4.5	75.0
5	5	dasie	26	female	5.9	60.0
6	6	bob	26	male	5.8	80.0
7	7	carl	28	male	5.8	51.0
8	8	Jessica	46	female	5.6	52.0
9	9	tom	65	male	4.9	53.0

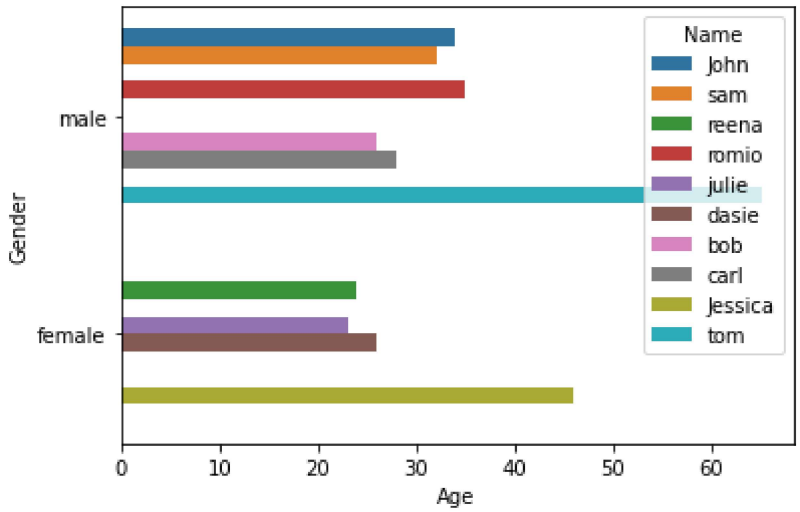
In [184…
```
plt.figure(figsize=(12,6))
sns.scatterplot(x='height', y = 'weight', data=persondata2, hue='Name', s=200)
```

Out[184]:
```
<AxesSubplot:xlabel='height', ylabel='weight'>
```



In [182…
```
sns.barplot(x='Age', y='Gender', data=persondata2, hue = 'Name')
```

Out[182]:
```
<AxesSubplot:xlabel='Age', ylabel='Gender'>
```



In [ ]: