

```
In [248]: # Run this cell, but please don't change it.

# These lines import the Numpy and Datascience modules.
import numpy as np
from datascience import *

# These lines do some fancy plotting magic
import matplotlib
%matplotlib inline
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
```

Part 1: Data Visualizations

```
In [249]: months=('April','May','June','July')
adults=(315,400,1000,1500)
children=(780,1050,3056,5025)
waterParkVisitors=Table().with_columns('Months',months,'Adults',adults,
waterParkVisitors)
```

```
Out[249]:
```

Months	Adults	Children
April	315	780
May	400	1050
June	1000	3056
July	1500	5025

```

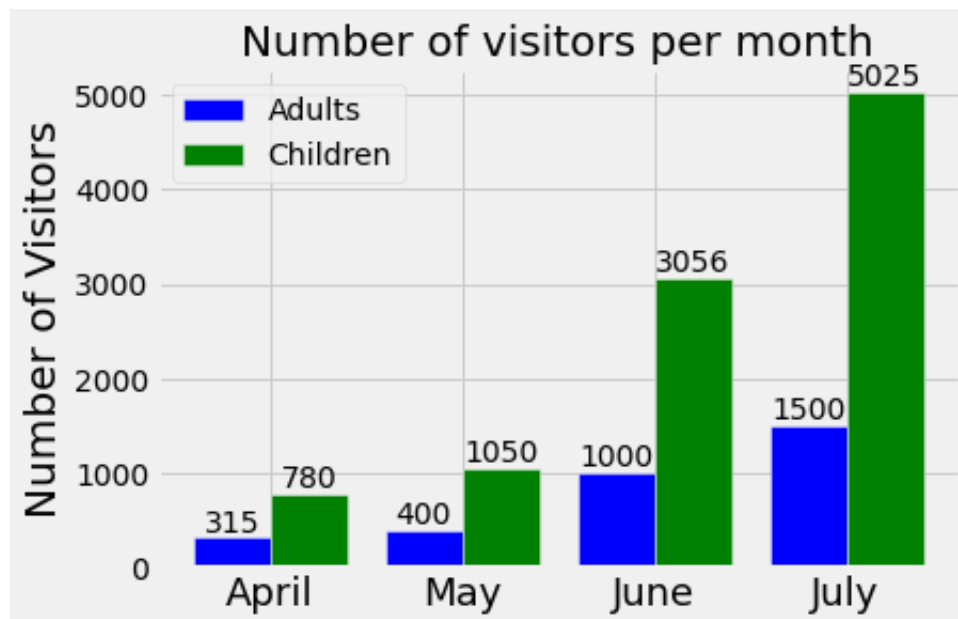
In [250]: index=np.arange(4)
width=0.40
fig, ax = plt.subplots()
bar1=ax.bar(index,adults,width,color='blue')
bar2=ax.bar(index+width,children,width,color='green')

ax.set_ylabel('Number of Visitors',fontsize=20)
ax.set_title('Number of visitors per month',fontsize=20)
ax.set_xticks(index + width / 2)
ax.set_xticklabels(('April', 'May', 'June', 'July'),fontsize=18)

ax.legend((bar1[0], bar2[0]), ('Adults', 'Children'))

def labelbars(bars):
    for bar in bars:
        height = bar.get_height()
        ax.text(bar.get_x() + bar.get_width()/2., height,
                '%d' % int(height),
                ha='center', va='bottom')
labelbars(bar1)
labelbars(bar2)
plt.show()

```



```

In [251]: Width2= [i + (width/50) for i in index]

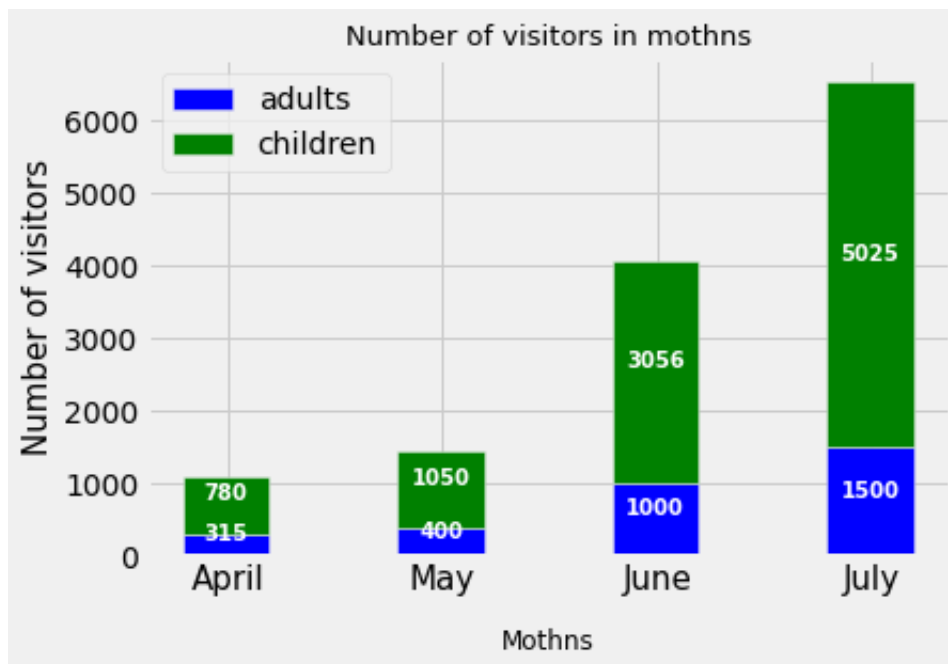
b1= plt.bar(index,adults, width, color='blue',align='center')
b2 = plt.bar(index,children,width,bottom=adults,color='green',align='c

plt.ylabel('Number of visitors',fontsize=15)
plt.xlabel('Mothns',fontsize=12,labelpad=15)
plt.title('Number of visitors in mothns',fontsize=13)
plt.xticks(Width2,('April','May','June','July'), fontsize=15)
plt.legend((b1[0], b2[0]), ('adults', 'children'))

for b1, b2 in zip(b1, b2):
    h1 = b1.get_height()
    h2 = b2.get_height()
    plt.text(b1.get_x() + b1.get_width() / 2., h1 / 2., "%d" % h1, ha=
    plt.text(b2.get_x() + b2.get_width() / 2., h1 + h2 / 2., "%d" % h2

plt.show()

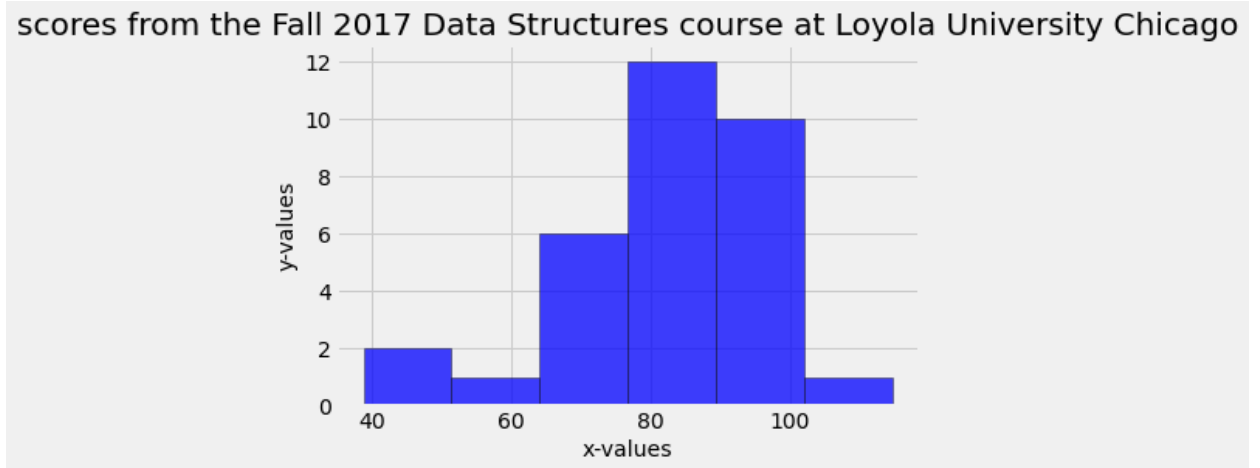
```



```
In [252]: scores=(114.8, 98.8, 97.3, 96, 94.1, 93.1, 93.1, 91.6, 91.5, 91.3, 90.
num_bins =6
n, bins, patches = plt.hist(scores, num_bins, facecolor='blue', alpha=

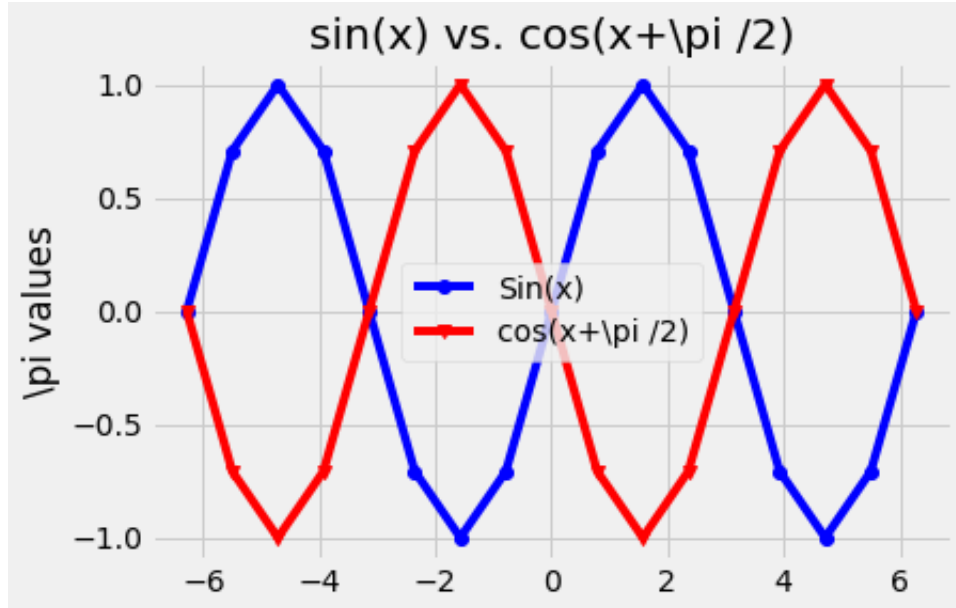
plt.title('scores from the Fall 2017 Data Structures course at Loyola
plt.xlabel('x-values',fontsize=14)
plt.ylabel('y-values',fontsize=14)

plt.show()
```



```
In [253]: x = (-6.283, -5.498, -4.712, -3.927, -3.142, -2.356, -1.571, -.7854, 0,
sin0Fx =( 0, .70711, 1, .70711, 0, -.70711, -1, -.70711, 0, .70711, 1,
cos0FxPlusnby2=( 0, -.70711, -1, -.70711, 0, .70711, 1, .70711, 0, -.7
plt.plot(x,sin0Fx,marker='o',color='b')
plt.plot(x,cos0FxPlusnby2,marker='v',color='r')
plt.title("sin(x) vs. cos(x+\pi /2)")
plt.ylabel("\pi values")
plt.legend(['Sin(x)', 'cos(x+\pi /2)'],loc='best')

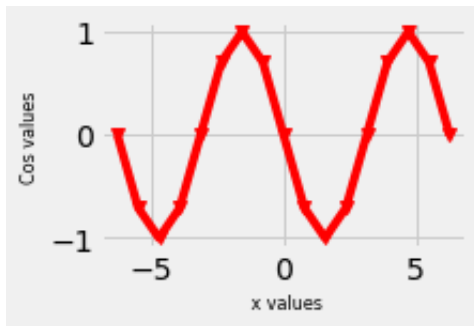
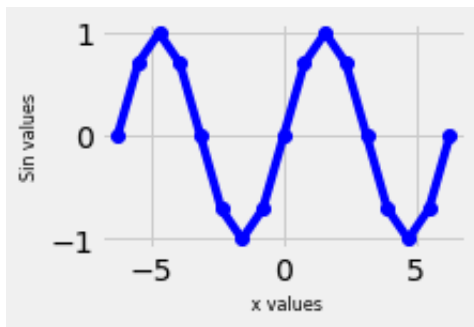
plt.show()
```



```
In [254]: plt.subplot(2,2,1)
plt.plot(x,sinOfx,marker='o',color='b')
plt.xlabel(' x values',fontsize=8)
plt.ylabel('Sin values',fontsize=8)
plt.show()

plt.subplot(2,2,2)
plt.plot(x,cosOfxPlusnby2,marker='v',color='r')
plt.xlabel(' x values',fontsize=8)
plt.ylabel('Cos values',fontsize=8)
plt.show()

plt.tight_layout()
plt.show
```

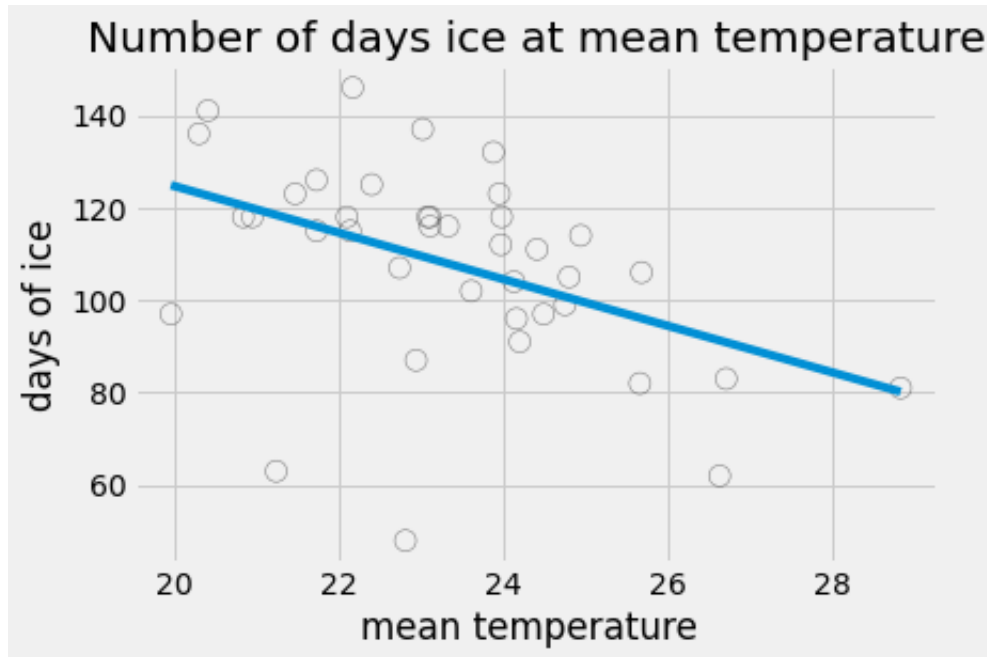


```
Out[254]: <function matplotlib.pyplot.show(*args, **kw)>
<Figure size 432x288 with 0 Axes>
```

```
In [255]: meanTemperature=(22.94, 23.02, 25.68, 19.96, 24.80, 23.98, 22.10, 20.3
daysOfIce=(87, 137, 106, 97, 105, 118, 118, 136, 91, 107, 96, 114, 125

plt.scatter(meanTemperature,daysOfIce,s=100, facecolors ='none', edgec
plt.plot(np.unique(meanTemperature),np.poly1d(np.polyfit(meanTemperatu
plt.xlabel('mean temperature '))
plt.ylabel('days of ice')
plt.title('Number of days ice at mean temperature')

plt.show()
```



Part 2: Basic Data Structure

```
In [256]: myList=['one', 'two', 'three', 'four', 'five']

for i in myList:
    print(i)
```

```
one
two
three
four
five
```

```
In [257]: myList.remove('three')
```

```
In [258]: myList
```

```
Out[258]: ['one', 'two', 'four', 'five']
```

```
In [259]: 'four' in myList
```

```
Out[259]: True
```

```
In [260]: myList.append('six')
```

```
In [261]: for i in myList:
           print(f'{i} is {len(i)}')
```

```
one is 3
two is 3
four is 4
five is 4
six is 3
```

```
In [262]: def strings(List):
           for i in List:
               print(f'{len(i)}')
           strings(myList)
```

```
3
3
4
4
3
```

Dictionaries

```
In [263]: Dict={'apple':'Apfel','apples':'Äpfel','I':'Ich','and':'und','like':'m
Dict
```

```
Out[263]: {'apple': 'Apfel',
           'apples': 'Äpfel',
           'I': 'Ich',
           'and': 'und',
           'like': 'mag',
           'strawberries': 'Erdbeeren'}
```

```
In [264]: Dict['apple']
```

```
Out[264]: 'Apfel'
```



```
In [265]: Dict['like']
```

```
Out[265]: 'mag'
```

```
In [266]: var='I like apples and strawberries'
```

```
In [267]: Lst=var.split(' ')
Lst
```

```
Out[267]: ['I', 'like', 'apples', 'and', 'strawberries']
```

```
In [268]: for i in Lst:
            if i in Dict.keys():
                Lst[Lst.index(i)]=Dict[i]
            print(Lst)

['Ich', 'mag', 'Äpfel', 'und', 'Erdbeeren']
```

```
In [269]: " ".join(Lst)
```

```
Out[269]: 'Ich mag Äpfel und Erdbeeren'
```

Arrays

```
In [270]: array=np.zeros((8,8),dtype=int)
array
```

```
Out[270]: array([[0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0],
                 [0, 0, 0, 0, 0, 0, 0, 0]])
```

```
In [271]: array.dtype
```

```
Out[271]: dtype('int64')
```

```
In [272]: n=1
          for i in range(len(array)):
              for j in range(len(array[i])):
                  array[i,j]=n
                  n+=1
          print(array)
```

```
[[ 1  2  3  4  5  6  7  8]
 [ 9 10 11 12 13 14 15 16]
 [17 18 19 20 21 22 23 24]
 [25 26 27 28 29 30 31 32]
 [33 34 35 36 37 38 39 40]
 [41 42 43 44 45 46 47 48]
 [49 50 51 52 53 54 55 56]
 [57 58 59 60 61 62 63 64]]
```

```
In [273]: print(array.transpose())
```

```
[[ 1  9 17 25 33 41 49 57]
 [ 2 10 18 26 34 42 50 58]
 [ 3 11 19 27 35 43 51 59]
 [ 4 12 20 28 36 44 52 60]
 [ 5 13 21 29 37 45 53 61]
 [ 6 14 22 30 38 46 54 62]
 [ 7 15 23 31 39 47 55 63]
 [ 8 16 24 32 40 48 56 64]]
```

```
In [274]: print(array[:4,:4])
```

```
[[ 1  2  3  4]
 [ 9 10 11 12]
 [17 18 19 20]
 [25 26 27 28]]
```

```
In [275]: array.flatten()
```

```
Out[275]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
 17,
 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
 34,
 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
 51,
 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64])
```

In [276]: `array.reshape(8,8)`

Out[276]: `array([[1, 2, 3, 4, 5, 6, 7, 8],
[9, 10, 11, 12, 13, 14, 15, 16],
[17, 18, 19, 20, 21, 22, 23, 24],
[25, 26, 27, 28, 29, 30, 31, 32],
[33, 34, 35, 36, 37, 38, 39, 40],
[41, 42, 43, 44, 45, 46, 47, 48],
[49, 50, 51, 52, 53, 54, 55, 56],
[57, 58, 59, 60, 61, 62, 63, 64]])`

Part 3: Data Frames

In [277]: `import pandas as pd
TitanicDf={'Sex':['Children','Children','Children','Men','Men','Men','
 'Class':['First','Second','Third','First','Second','Third'],
 'Survived':[6,24,27,57,14,75,192,140,80,76,20],
 'Died':[0,0,52,118,154,387,693,4,13,89,3]}
TitanicDf=pd.DataFrame(TitanicDf)
TitanicDf`

Out[277]:

	Sex	Class	Survived	Died
0	Children	First	6	0
1	Children	Second	24	0
2	Children	Third	27	52
3	Men	First	57	118
4	Men	Second	14	154
5	Men	Third	75	387
6	Men	Crew	192	693
7	Women	First	140	4
8	Women	Second	80	13
9	Women	Third	76	89
10	Women	Crew	20	3

```
In [278]: TitanicDf[TitanicDf['Class']=='First']
```

```
Out[278]:
```

	Sex	Class	Survived	Died
0	Children	First	6	0
3	Men	First	57	118
7	Women	First	140	4

```
In [279]: ind=TitanicDf[TitanicDf['Class']=='Crew'].index
ind
```

```
Out[279]: Int64Index([6, 10], dtype='int64')
```

```
In [280]: TitanicDf=TitanicDf.drop(ind,axis=0)
```

```
In [281]: TitanicDf
```

```
Out[281]:
```

	Sex	Class	Survived	Died
0	Children	First	6	0
1	Children	Second	24	0
2	Children	Third	27	52
3	Men	First	57	118
4	Men	Second	14	154
5	Men	Third	75	387
7	Women	First	140	4
8	Women	Second	80	13
9	Women	Third	76	89

```
In [282]: TitanicDf['Total members']=TitanicDf['Survived']+TitanicDf['Died']
```

```
In [283]: TitanicDf['Servived_Percentage']=TitanicDf['Survived']/TitanicDf['Total members']
```

In [284]: TitanicDf

Out[284]:

	Sex	Class	Survived	Died	Total members	Servived_Percentage
0	Children	First	6	0	6	100.000000
1	Children	Second	24	0	24	100.000000
2	Children	Third	27	52	79	34.177215
3	Men	First	57	118	175	32.571429
4	Men	Second	14	154	168	8.333333
5	Men	Third	75	387	462	16.233766
7	Women	First	140	4	144	97.222222
8	Women	Second	80	13	93	86.021505
9	Women	Third	76	89	165	46.060606

In [285]: TitanicDf.drop('Total members',axis=1,inplace=True)

In [286]: TitanicDf

Out[286]:

	Sex	Class	Survived	Died	Servived_Percentage
0	Children	First	6	0	100.000000
1	Children	Second	24	0	100.000000
2	Children	Third	27	52	34.177215
3	Men	First	57	118	32.571429
4	Men	Second	14	154	8.333333
5	Men	Third	75	387	16.233766
7	Women	First	140	4	97.222222
8	Women	Second	80	13	86.021505
9	Women	Third	76	89	46.060606

In [287]: TitanicDf[TitanicDf['Survived_Percentage']>80]

Out[287]:

	Sex	Class	Survived	Died	Survived_Percentage
0	Children	First	6	0	100.000000
1	Children	Second	24	0	100.000000
7	Women	First	140	4	97.222222
8	Women	Second	80	13	86.021505

In [288]: TitanicDf[TitanicDf['Survived_Percentage']<40]

Out[288]:

	Sex	Class	Survived	Died	Survived_Percentage
2	Children	Third	27	52	34.177215
3	Men	First	57	118	32.571429
4	Men	Second	14	154	8.333333
5	Men	Third	75	387	16.233766

In [289]: TitanicDfGroupbyClass=TitanicDf.groupby('Class').sum()

In [290]: TitanicDfGroupbyClass['Total']=TitanicDfGroupbyClass['Survived']+TitanicDfGroupbyClass['Died']

In [291]: TitanicDfGroupbyClass.drop('Survived_Percentage',inplace=True,axis=1)

In [292]: TitanicDfGroupbyClass['Survived_Percentage']=TitanicDfGroupbyClass['Survived']/TitanicDfGroupbyClass['Total']
TitanicDfGroupbyClass['Died_Percentage']=TitanicDfGroupbyClass['Died']/TitanicDfGroupbyClass['Total']

In [293]: TitanicDfGroupbyClass

Out[293]:

	Survived	Died	Total	Survived Percentage	Died Percentage
Class					
First	203	122	325	62.461538	37.538462
Second	118	167	285	41.403509	58.596491
Third	178	528	706	25.212465	74.787535

In [294]: TitanicDfGroupbyClass=TitanicDf.groupby(['Sex','Class']).sum()

In [295]: TitanicDfGroupbyClass

Out[295]:

		Survived	Died	Servived_Percentage
Sex	Class			
Children	First	6	0	100.000000
	Second	24	0	100.000000
	Third	27	52	34.177215
Men	First	57	118	32.571429
	Second	14	154	8.333333
	Third	75	387	16.233766
Women	First	140	4	97.222222
	Second	80	13	86.021505
	Third	76	89	46.060606

In [296]: TitanicDfGroupbyClass.to_csv('Titanic.csv',header=True)

I have Duplicated the CSV file on my computer since i will be editing the copied version . I have opened the new CSV file in a text editor. Now, in the text editor,i have added new row 'Crew' .

In [299]: titanicDf2=pd.read_csv('Titanic2.csv')

In [300]: titanicDf2

Out[300]:

	Sex	Class	Survived	Died	Servived_Percentage
0	Children	First	6	0	100.000000
1	Children	Second	24	0	100.000000
2	Children	Third	27	52	34.177215
3	Men	First	57	118	32.571429
4	Men	Second	14	154	8.333333
5	Men	Third	75	387	16.233766
6	Women	First	140	4	97.222222
7	Women	Second	80	13	86.021505
8	Women	Third	76	89	46.060606
9	women	Crew	20	3	86.960000
10	Men	Crew	192	693	21.690000