Inclass - Lab (Day 4)

Table of Content

Let us import the required libraries.

In [88]:
```
# import the libraries
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns

import plotly.express as px
```

Let's begin with some hands-on practice exercises

1. Plots using Library Matplotlib

a. Scatter Plots

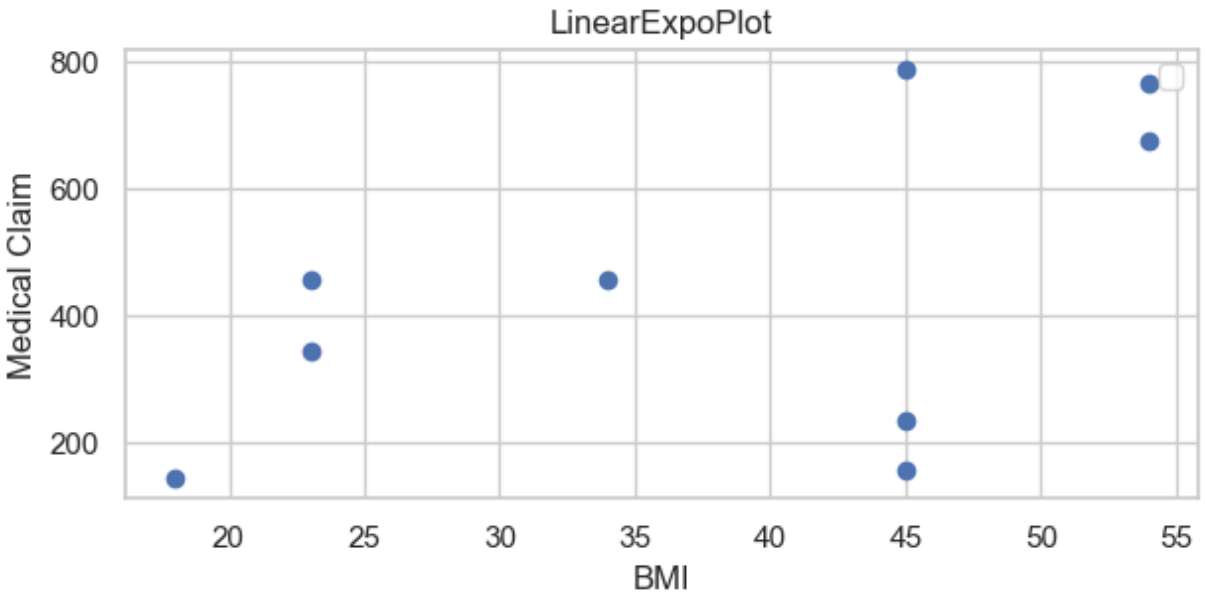
1. Plot a scatter plot for the following data. Also add title and axis names

```
bmi = (18, 34, 54, 45, 45, 45, 23, 23, 54)
medical_claim = (145, 456, 764, 234, 156, 786, 345, 455, 675)
```

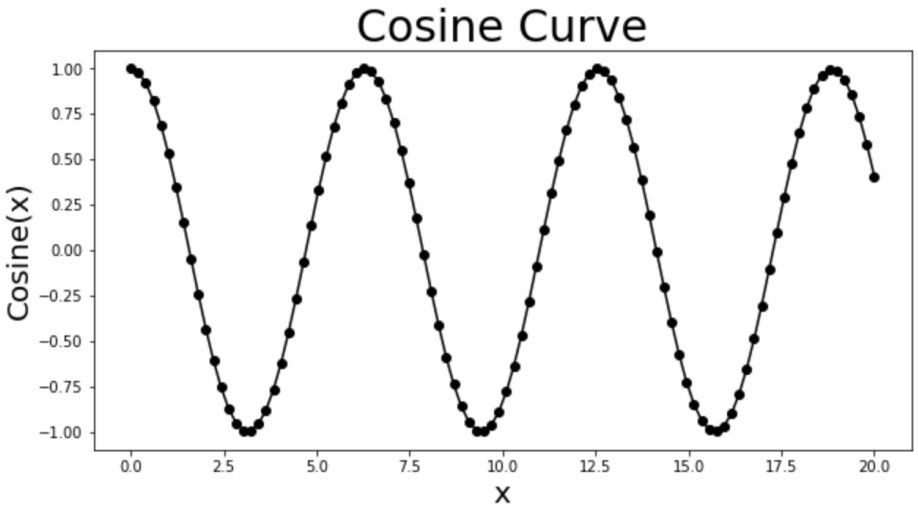
In [59]:
```
# type your code here
bmi = (18, 34, 54, 45, 45, 45, 23, 23, 54)
medical_claim = (145, 456, 764, 234, 156, 786, 345, 455, 675)
```

In [60]:
```
plt.figure(figsize=(7,3),dpi=100)
plt.scatter(bmi,medical_claim)
plt.xlabel('BMI')
plt.ylabel('Medical Claim')
plt.title('LinearExpoPlot')
plt.legend()
plt.show()
```

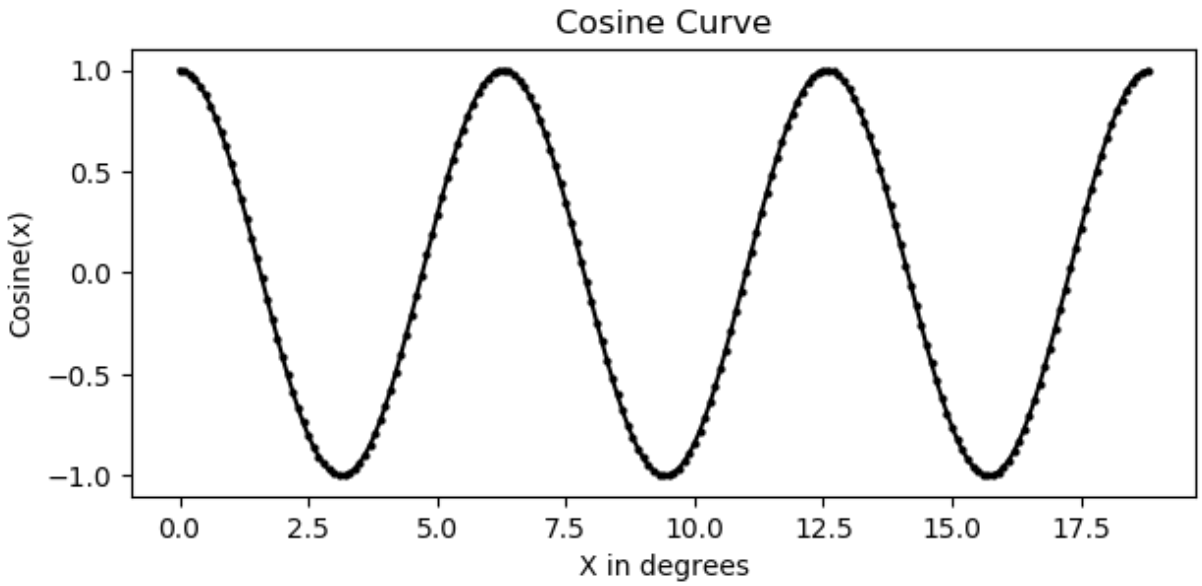
```
No handles with labels found to put in legend.
```



2. Write a code to get the following output



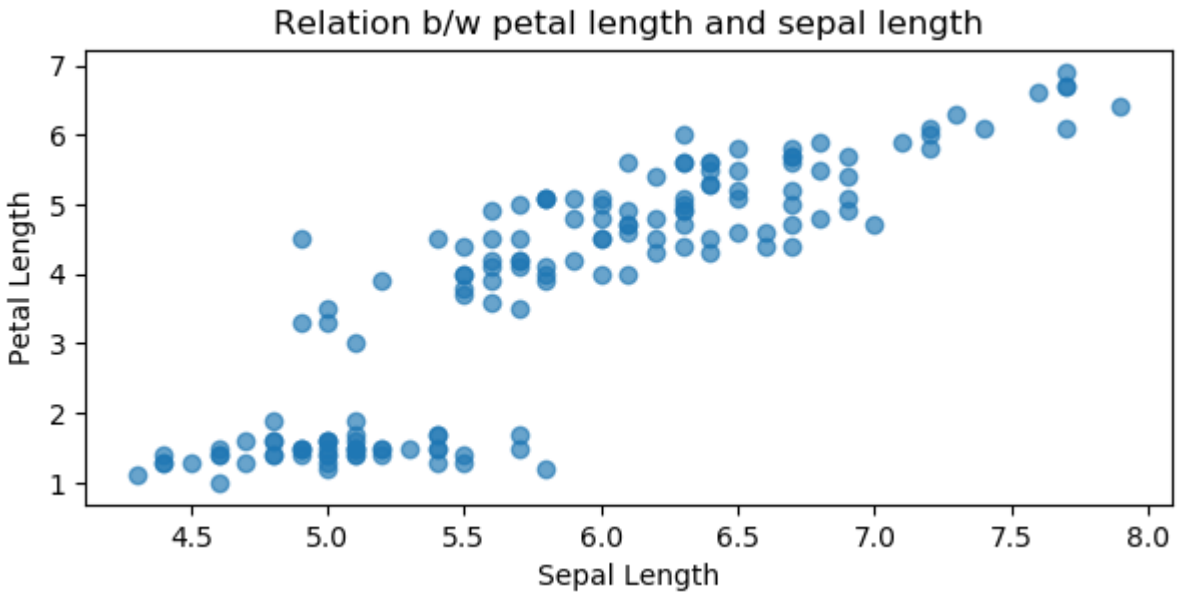
In [3]:
```
# type your code here
x = np.arange(0,np.pi*6,0.1)
y = np.cos(x)
plt.figure(figsize=(7,3),dpi=100)
plt.plot(x,y,marker='.',mew=0.02,color='black')
plt.xlabel("X in degrees")
plt.ylabel("Cosine(x)")
plt.title('Cosine Curve')
plt.show()
```



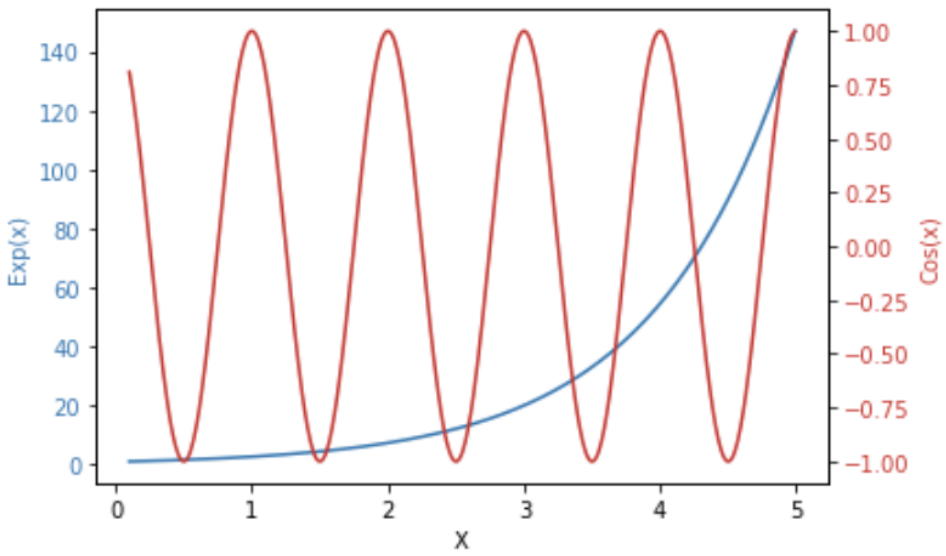
3. Create a bubble plot. Further use the parameter 'alpha' to adjust the transparency level. (Generate the data using random number)

Note: A bubble chart is a type of scatter plot that represents three dimensional data. The third variable is represented by the size of a point (marker).

In [4]:
```
# type your code here
data = px.data.iris()
plt.figure(figsize=(7,3),dpi=100)
plt.scatter(x=data.sepal_length,y=data.petal_length,alpha=0.67)
plt.xlabel('Sepal Length')
plt.ylabel('Petal Length')
plt.title('Relation b/w petal length and sepal length')
plt.show()
```



4. Plot cosine curve and exponential curve for natural numbers from 0.1 to 20, in the same plot (get the following output)



In [ ]:

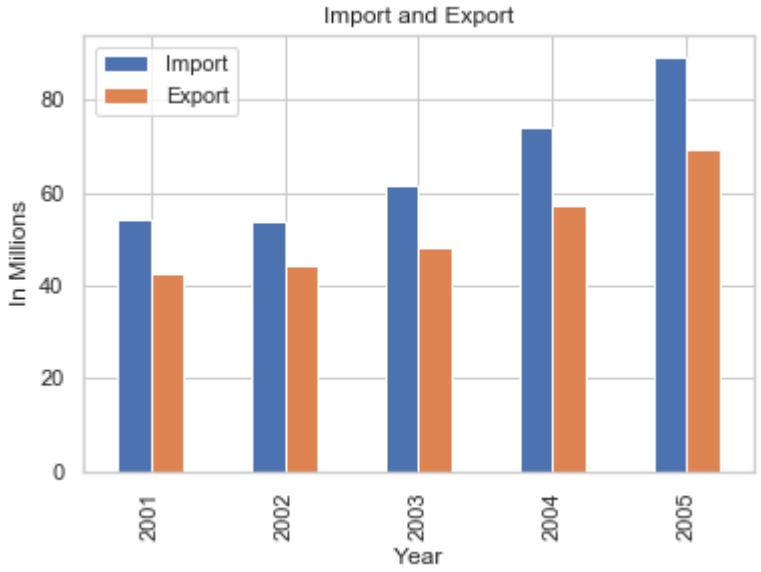
b. Barplots

5. The exports and imports (in billion dollars) is given for a country from 2001 to 2005. Draw a barplot for the data

Year	Import	Export
2001	54.4	42.5
2002	53.8	44.5
2003	61.6	48.3
2004	74.15	57.24
2005	89.33	69.18

In [63]:
```
# type your code here
y= [i for i in range(2001,2006)]
im= [54.4, 53.8,61.6,74.15,89.33]
iex= [42.5,44.5,48.3,57.24,69.18]
df= pd.DataFrame(zip(y,im,iex), columns= ['Year', 'Import', 'Export'])

plt.figure(figsize=(7,3),dpi=100)
df.plot(x= 'Year', y= ['Import', 'Export'], kind= 'bar', title= 'Import and Expor
plt.xlabel('Year')
plt.ylabel('In Millions')
plt.show()
```

```
<Figure size 700x300 with 0 Axes>
```

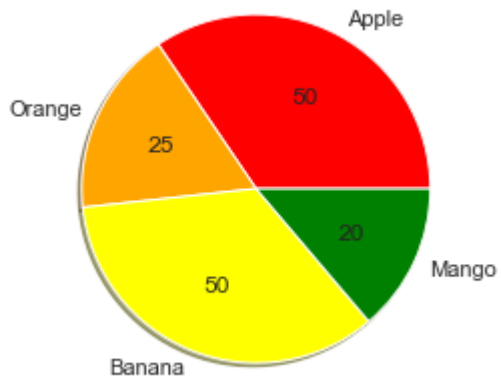


c. Pie plots

6. Plot a pie chart for the following data

```
prices = [50, 25, 50, 20]
labels = ['Apple', 'Orange','Banana', 'Mango']
```

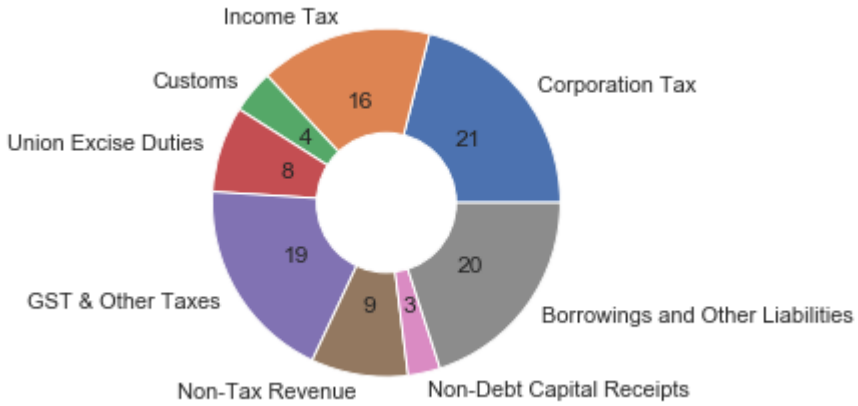
In [64]:
```
# type your code here
prices = [50, 25, 50, 20]
labels = ['Apple', 'Orange','Banana', 'Mango']

df= pd.Series(prices, labels)
plt.pie(df, labels=labels, autopct=(lambda x: '{:.0f}'.format((x/100)*sum(prices)
plt.show()
```



7. Following is the information release after the Indian budget is declared. Plot a donut chart for it

Income Form	Amount (in paise)
Corporation Tax	21
Income Tax	16
Customs	4
Union Excise Duties	8
GST & Other Taxes	19
Non-Tax Revenue	9
Non-Debt Capital Receipts	3
Borrowings and Other Liabilities	20

In [65]:
```
# type your code here
inc= ['Corporation Tax', 'Income Tax', 'Customs', 'Union Excise Duties', 'GST & C
am= [21,16,4,8,19,9,3,20]
d= pd.Series(am, inc)
d

plt.pie(d, labels=inc, autopct=(lambda x: '{:.0f}'.format((x/100)*sum(am))))
circle = plt.Circle(xy=(0,0), radius=0.4, color='white')
ax= plt.gca()
ax.add_artist(circle)
plt.show()
```

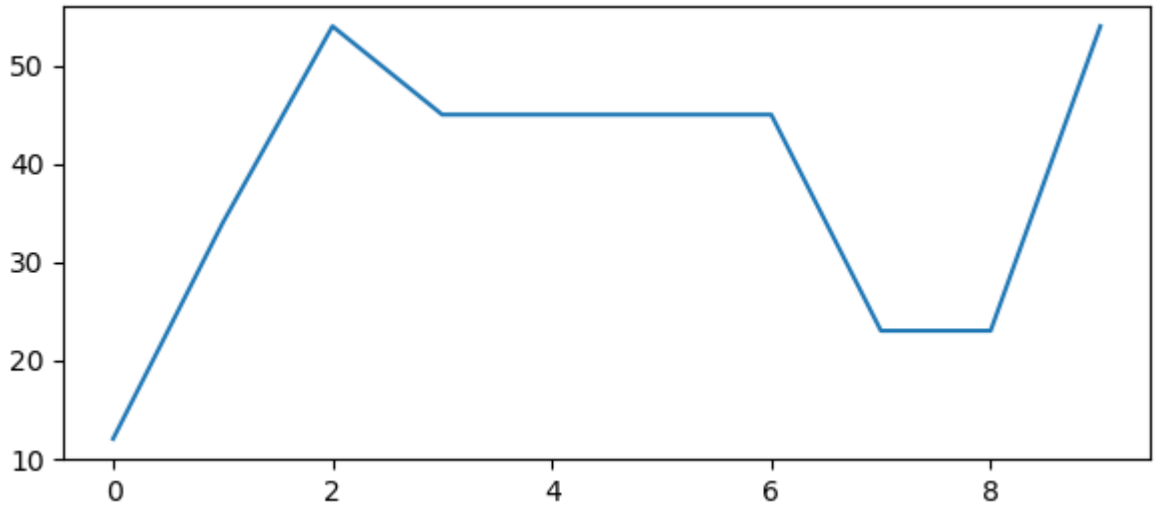


d. Line plots

8. Create a tuple of numbers 12, 34, 54, 45, 45, 45, 23, 23, 54 and plot a simple line plot

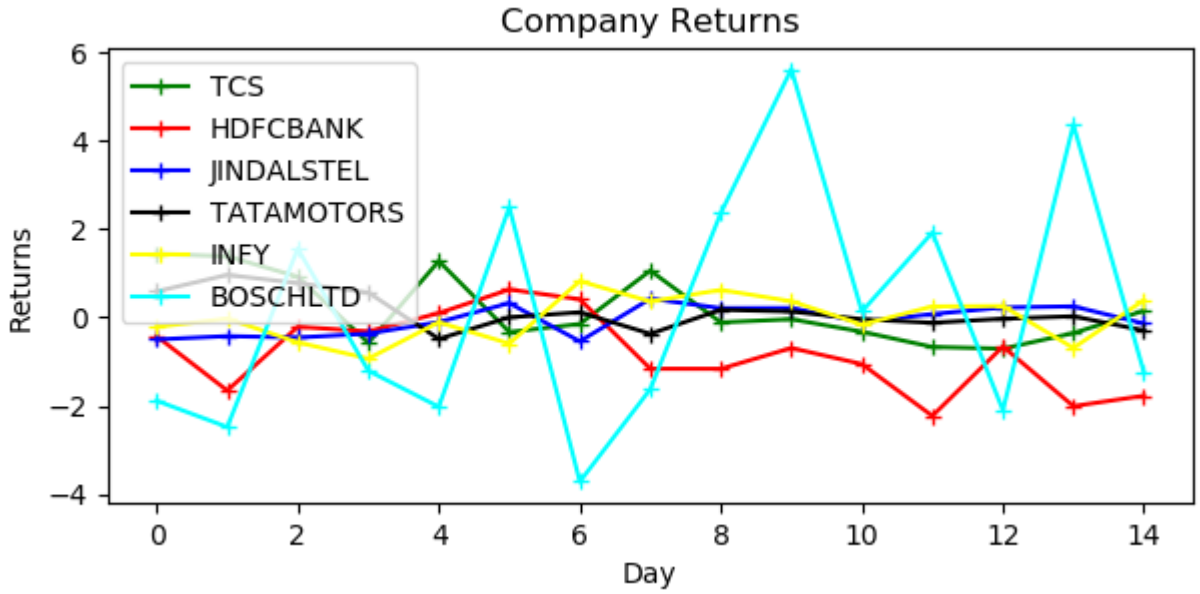
In [30]:
```
# type your code here
dat = (12,34,54,45,45,45,45,23,23,54)
x = (1,2,3,4,5,6,7,8,9)
print(dat)

plt.figure(figsize=(7,3),dpi=100)
plt.plot(dat)
plt.show()
```

```
(12, 34, 54, 45, 45, 45, 45, 23, 23, 54)
```



9. Import the 'Returns.csv'. Plot a multiple line plot for the returns of each company

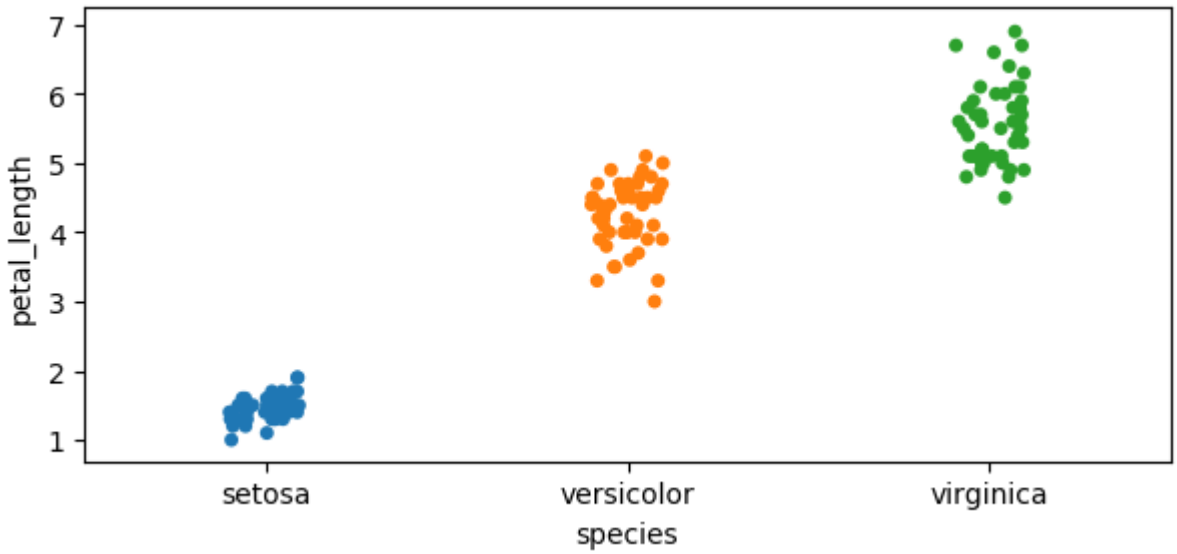
In [29]:
```
# type your code here
pd9 = pd.read_csv('Returns.csv')
plt.figure(figsize=(7,3),dpi=100)
pd9.TCS.plot(kind='line',color='green',marker='+',label='TCS')
pd9.HDFCBANK.plot(kind='line',color='red',marker='+',label='HDFCBANK')
pd9.JINDALSTEL.plot(kind='line',color='blue',marker='+',label='JINDALSTEL')
pd9.TATAMOTORS.plot(kind='line',color='black',marker='+',label='TATAMOTORS')
pd9.INFY.plot(kind='line',color='yellow',marker='+',label='INFY')
pd9.BOSCHLTD.plot(kind='line',color='cyan',marker='+',label='BOSCHLTD')
plt.xlabel('Day')
plt.ylabel('Returns')
plt.title('Company Returns')
plt.legend()
plt.show()
```



2. Plots using Library Seaborn

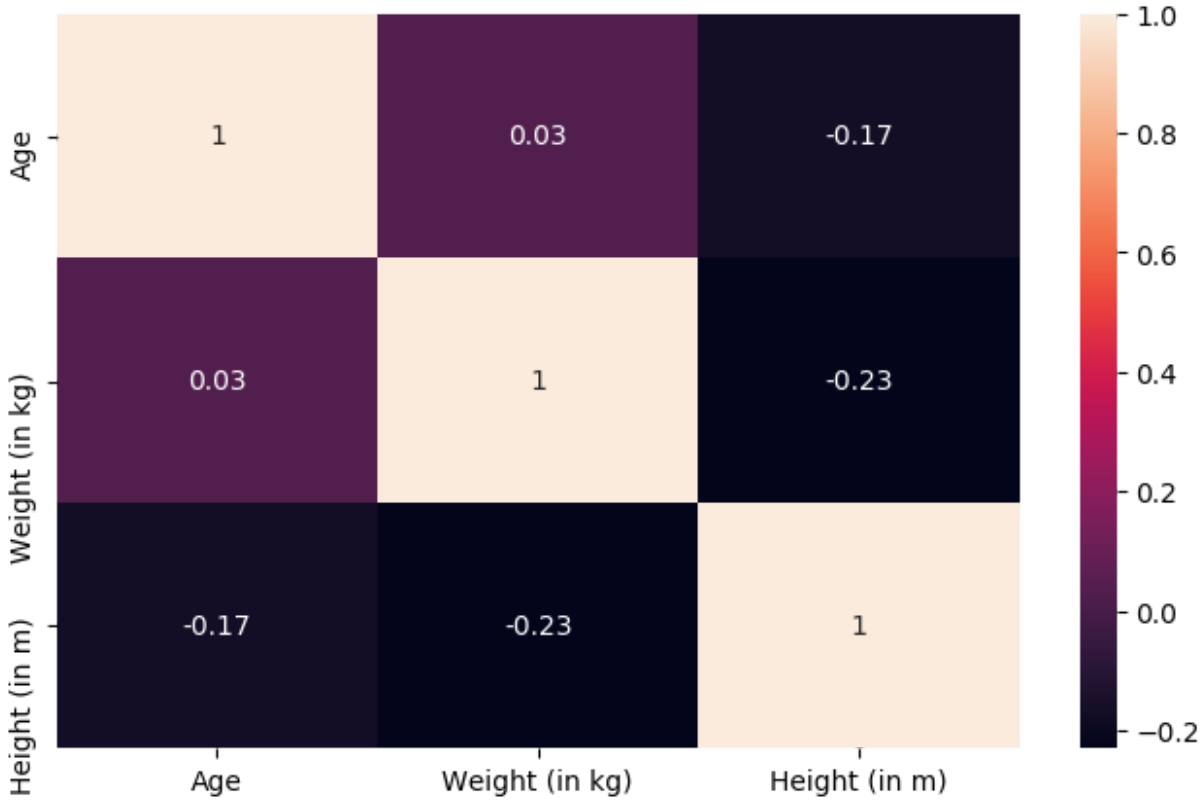
10. Plot a strip plot using inbuilt data-set 'iris' given in seaborn

In [41]:
```
# type your code here
data = sns.load_dataset('iris')
plt.figure(figsize=(7,3),dpi=100)
sns.stripplot(x=data.species,y=data.petal_length)
plt.show()
```



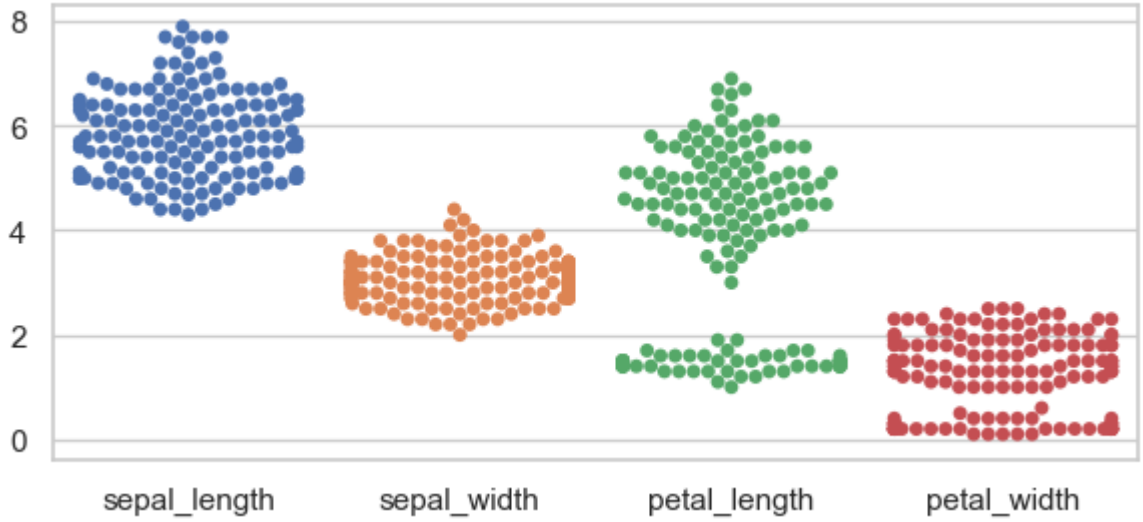
11. Import the dataset age_height.csv. Plot a heat map for the correlation matrix

In [47]:
```
# type your code here
pd11 = pd.read_csv('age_height.csv')
hm = pd11.corr()
plt.figure(figsize=(8,5),dpi=100)
sns.heatmap(hm,annot=True)
plt.show()
```



12. Plot a swarmplot using inbuilt data-set 'iris' given in seaborn

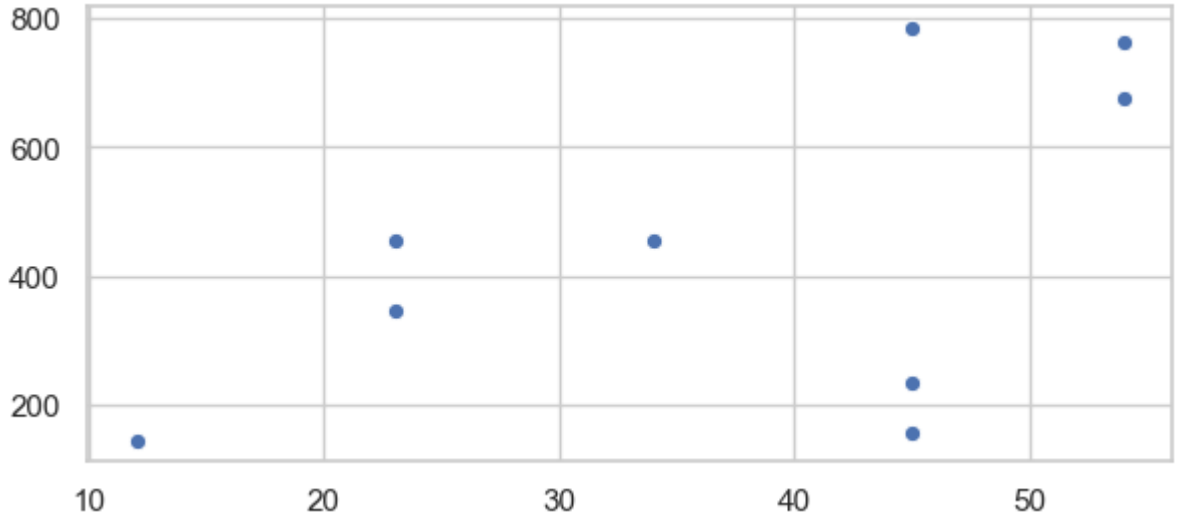
In [68]:
```
# type your code here
sns.set(style='whitegrid')
plt.figure(figsize=(7,3),dpi=100)
sns.swarmplot(data=data)
plt.show()
```



13a. Using the following data, plot a scatter plot with the function available in seaborn

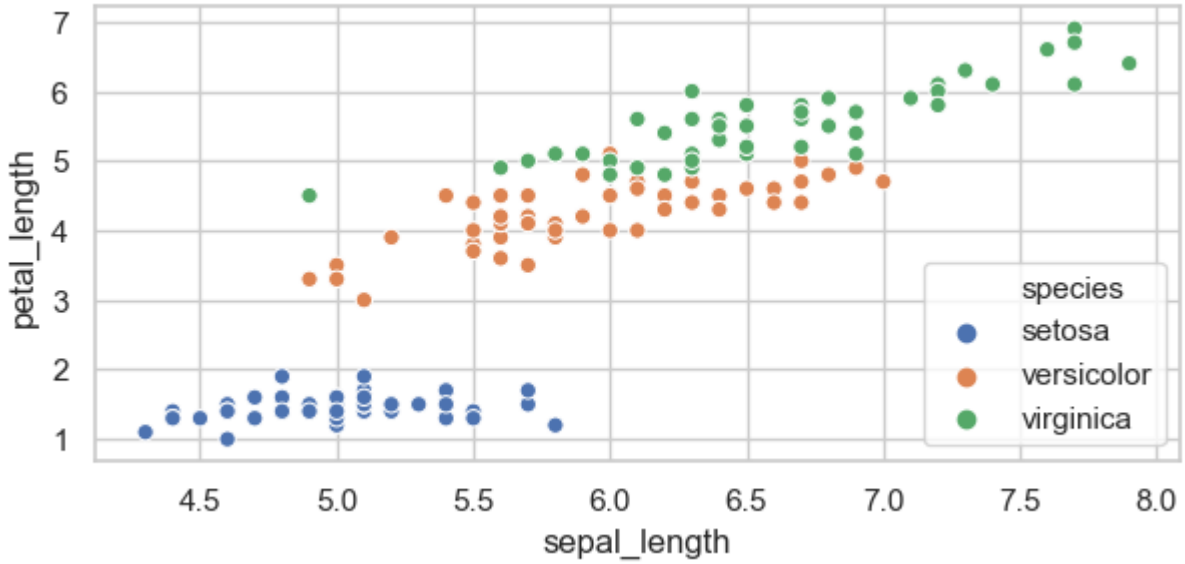
```
x = (12, 34, 54, 45, 45, 45, 23, 23, 54)
y = (145, 456, 764, 234, 156, 786, 345, 455, 675)
```

In [52]:
```
# type your code here
x1 = (12, 34, 54, 45, 45, 45, 23, 23, 54)
y1 = (145, 456, 764, 234, 156, 786, 345, 455, 675)

plt.figure(figsize=(7,3),dpi=100)
sns.set(style='whitegrid')
sns.scatterplot(x=x1,y=y1)
plt.show()
```



13b. Using the data generated in question 3, plot a scatter plot with the function available in seaborn

In [53]:
```
# type your code here
plt.figure(figsize=(7,3),dpi=100)
sns.scatterplot(x=data.sepal_length,y=data.petal_length,hue=data.species)
plt.show()
```

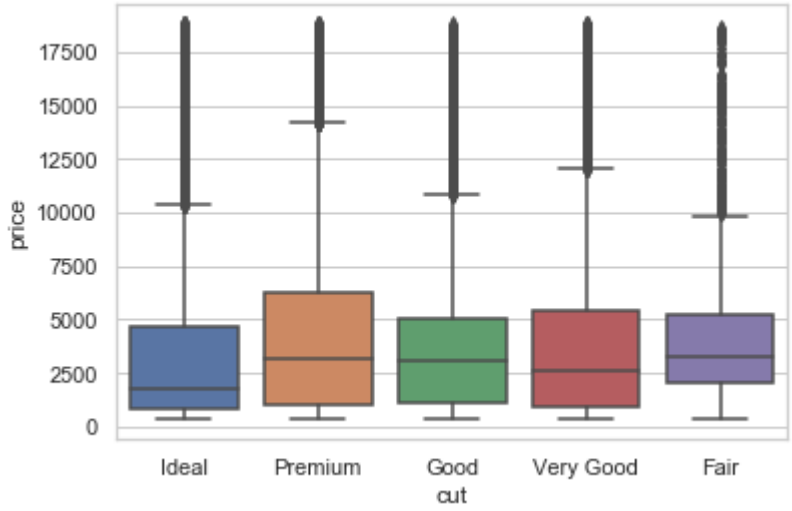


14. Using the 'diamonds' data available in the library seaborn, consider the price for the type of cut. Display the data using the following plots:

1. **Multiple boxplot**
2. **Violin Plot**
3. **Boxen Plot**

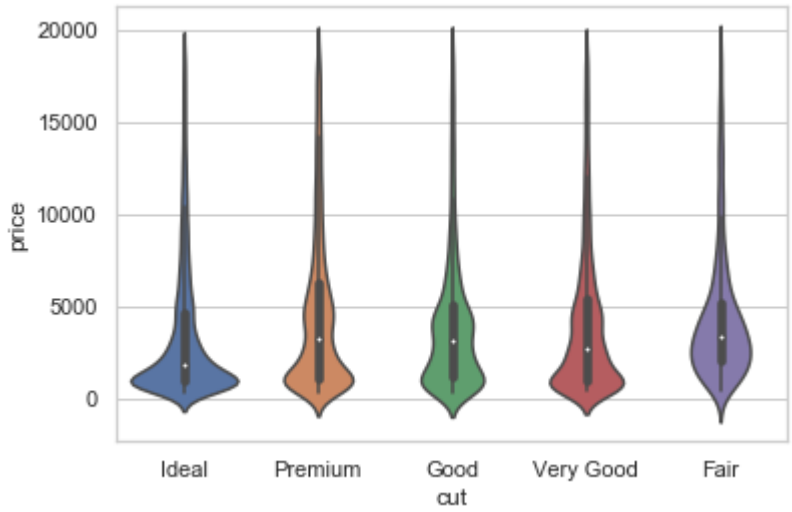
In [91]:
```
sns.boxplot(x='cut', y= 'price', data=dia)
```

Out[91]: <matplotlib.axes._subplots.AxesSubplot at 0x2180c99a388>



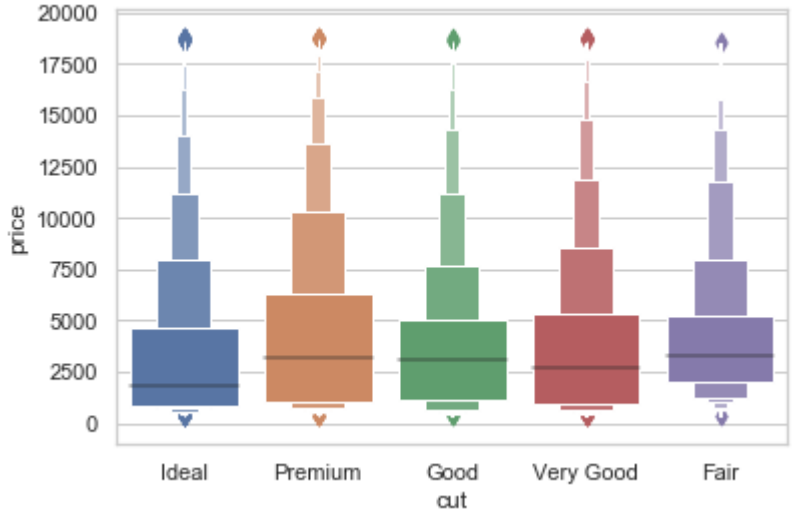
In [90]:
```
sns.violinplot(x='cut', y= 'price', data=dia)
```

Out[90]: <matplotlib.axes._subplots.AxesSubplot at 0x2180c1fe3c8>



In [89]:
```
# type your code here
sns.boxenplot(x='cut', y= 'price', data=dia)
```

Out[89]: <matplotlib.axes._subplots.AxesSubplot at 0x2180c8f58c8>



15. Generate random number form normal distribution, plot the following:
1. **Histogram**
2. **Histogram with frequency curve**

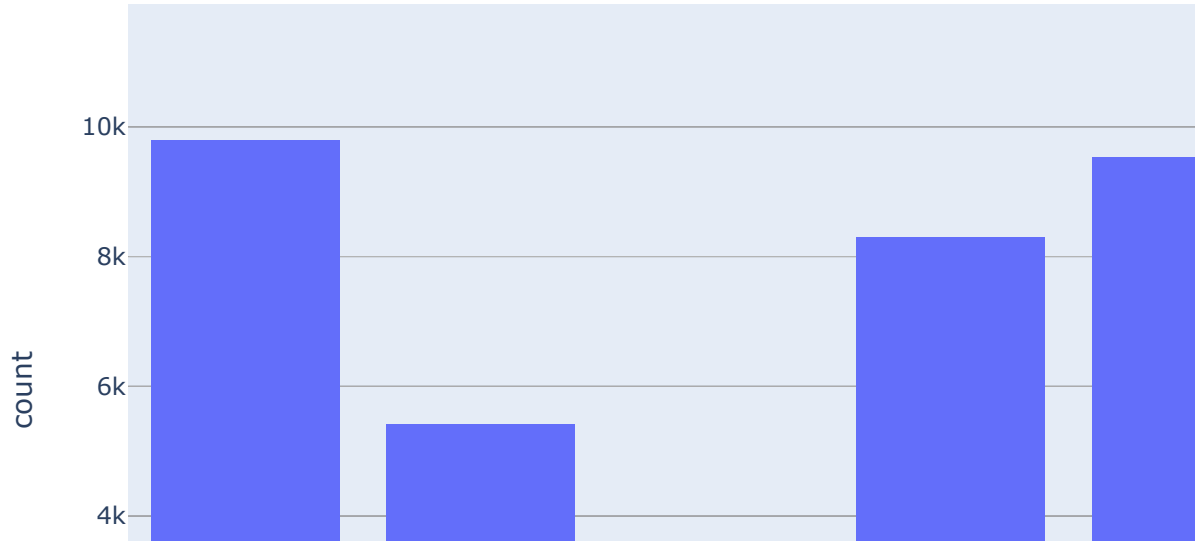
In [86]:
```
# type your code here
x = np.random.rand(100)
sns.histplot(data=x)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-86-7c8a8dd71724> in <module>
      1 # type your code here
      2 x = np.random.rand(100)
----> 3 sns.histplot(data=x)

AttributeError: module 'seaborn' has no attribute 'histplot'
```

3. Plots using Library Plotly

16. Use the 'diamonds' data set from seaborn to plot the histogram of price by colors

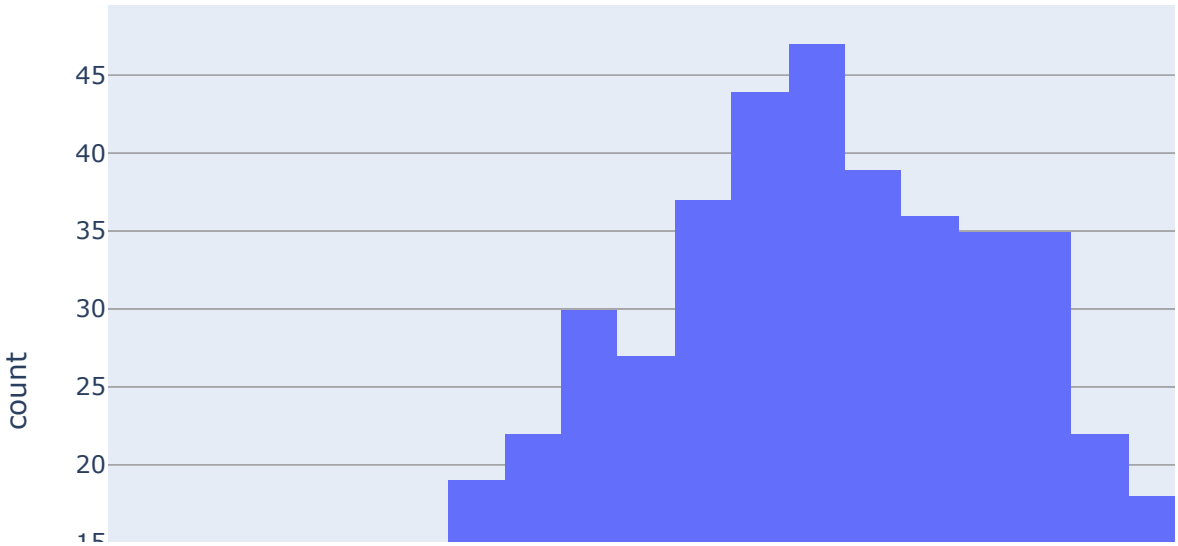
In [71]:
```
# type your code here
dia = sns.load_dataset('diamonds')
dia
#pt.hist_frame(dia)
px.hist_frame(dia, x='color')
```



17. Plot a histrogram for 500 random numbers which have mean 0 and variance 1

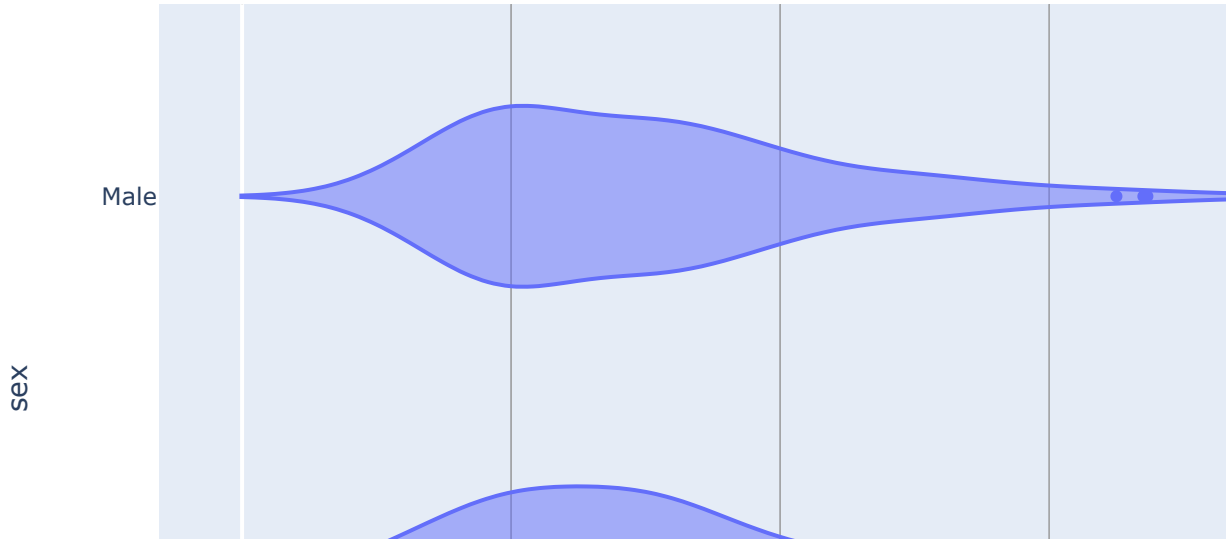
In [72]:
```
# type your code here
np.random.seed(23)
x = np.random.standard_normal(500)
print(np.mean(x))
print(np.var(x))
px.hist_frame(x)
```

```
-0.05407657171596391
0.9196498245435362
```



18. Using the 'tips' data from seaborn, plot a violin plot for 'tips' based on the 'sex'

In [78]:
```
# type your code here
df = sns.load_dataset('tips')
fig = px.violin(data_frame=df,y='sex',x='tip')
fig.show()
```



19. Using the tips dataset from seaborn, plot a donut plot with legend representing the percentage of the tip got on that day

In [81]:
```
# type your code here
```

20. The exports and imports (in billion dollars) is given for a country from 2001 to 2005. Draw a barplot for the data

Year	Import	Export
2001	54.4	42.5
2002	53.8	44.5
2003	61.6	48.3
2004	74.15	57.24

2005 89.33 69.18

In [85]:
```
# type your code here
df20 = pd.DataFrame({'Year':[2001,2002,2003,2004,2005],
                     'Import':[54.4,53.8,61.6,74.15,89.33],
                     'Export':[42.5,44.5,48.3,57.24,69.18]})
fig = px.bar(df20, x='Year', y='Export')
fig.show()
```