Anova

In [1]:

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

In [2]:

```
import warnings
warnings.filterwarnings('ignore')
```

In [3]:

```
df = pd.read_csv("C:\\Users\\arnak\\Downloads\\Iris.csv")
```

In [4]:

```
df
```

Out[4]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [5]:

```
df['Species'].unique()
```

Out[5]:

```
array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

In [6]:

```
group1 = df[df['Species'] == 'Iris-setosa']['SepalWidthCm']
```

```
In [7]: group2 = df[df['Species'] == 'Iris-versicolor']['SepalWidthCm']
```

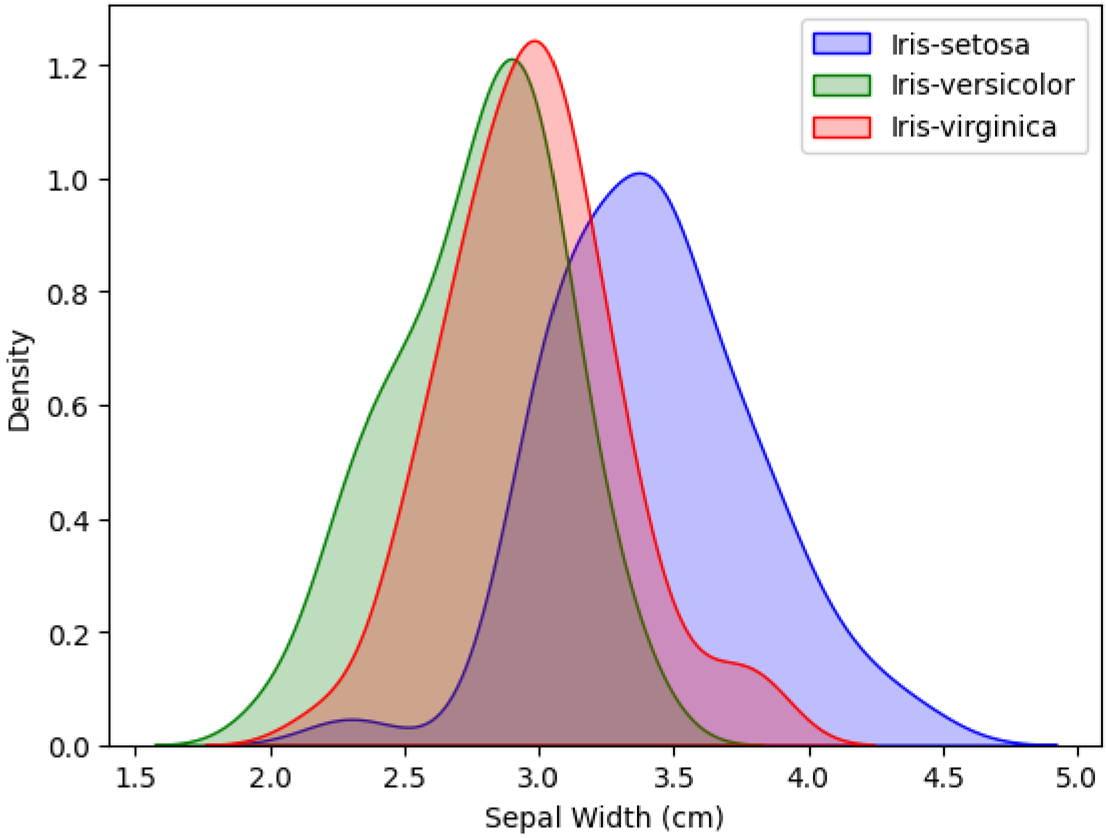
```
In [8]: group3 = df[df['Species'] == 'Iris-virginica']['SepalWidthCm']
```

```
In [9]: group1.mean(),group2.mean(),group3.mean()
```

```
Out[9]: (3.418, 2.7700000000000005, 2.974)
```

```
In [10]: sns.kdeplot(group1, color='blue', label='Iris-setosa', shade=True)
         sns.kdeplot(group2, color='green', label='Iris-versicolor', shade=True)
         sns.kdeplot(group3, color='red', label='Iris-virginica', shade=True)
         plt.legend()
         plt.xlabel('Sepal Width (cm)')
         plt.ylabel('Density')
```

```
Out[10]: Text(0, 0.5, 'Density')
```



```
In [11]: from scipy import stats
```

```
In [12]: stats.f_oneway(group1,group2,group3)
```

```
Out[12]: F_onewayResult(statistic=47.36446140299382, pvalue=1.3279165184572242e-16)
```

```
In [13]: df.describe()
```

Out[13]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

In [14]:

```
df.head()
```

Out[14]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

In [15]:

```
stats.f_oneway(df.iloc[:,1],df.iloc[:,2],df.iloc[:,3],df.iloc[:,4])
```

Out[15]:

```
F_onewayResult(statistic=483.57128302425997, pvalue=3.4996987081933735e-159)
```

Chi-Square Test

In [17]:

```
from scipy.stats import chi2_contingency
```

In [22]:

```
data=[[14,4],[0,10]]
```

In [23]:

```
data
```

Out[23]:

```
[[14, 4], [0, 10]]
```

In [24]:

```
df=pd.DataFrame(data,index=['Athlete','non-Alt'],columns=['non-Smoker','Smoker']
```

In [25]:

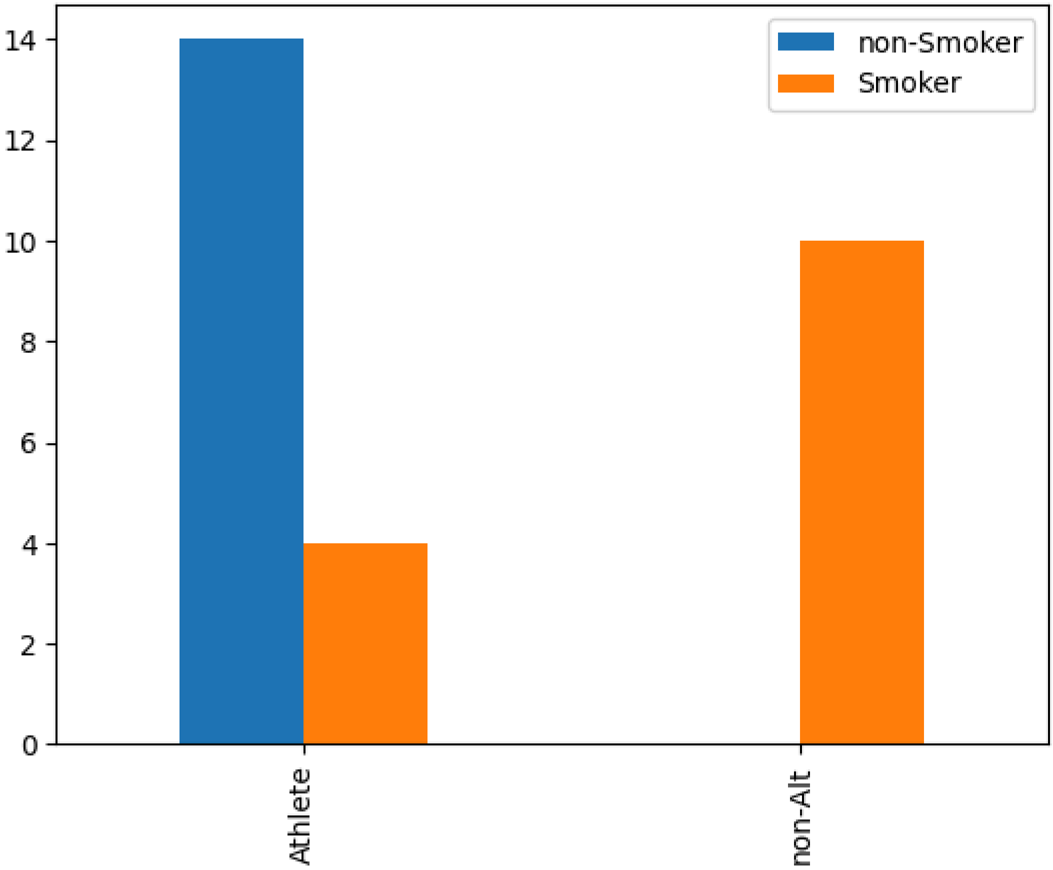
```
df
```

Out[25]:

	non-Smoker	Smoker
Athlete	14	4
non-Alt	0	10

```
In [29]: df.plot(kind='bar', stacked=False)
```

```
Out[29]: <Axes: >
```



```
In [31]: sta,pval,dof,freq=chi2_contingency(df)
```

```
In [32]: sta,pval,dof,freq
```

```
Out[32]: (12.600000000000001,
          0.00038574675568207l,
          1,
          array([[9., 9.],
                 [5., 5.]]))
```

```
In [33]: chi2_contingency(df)
```

```
Out[33]: Chi2ContingencyResult(statistic=12.600000000000001, pvalue=0.000385746755682007
         1, dof=1, expected_freq=array([[9., 9.],
                [5., 5.]]))
```

```
In [ ]:
```