In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [4]:

```
df = pd.read_csv('Iris.csv')
```

In [5]:

```
df
```

Out[5]:

	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 [6]:

```
df.head()
```

Out[6]:

	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 [7]:

```
df.tail()
```

Out[7]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
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

In [8]:

```
df.describe()
```

Out[8]:

	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 [10]:

```
df.dtypes
```

Out[10]:

```
Id               int64
SepalLengthCm    float64
SepalWidthCm     float64
PetalLengthCm    float64
PetalWidthCm     float64
Species           object
dtype: object
```

In [11]:

```
df.isna().sum()
```

Out[11]:

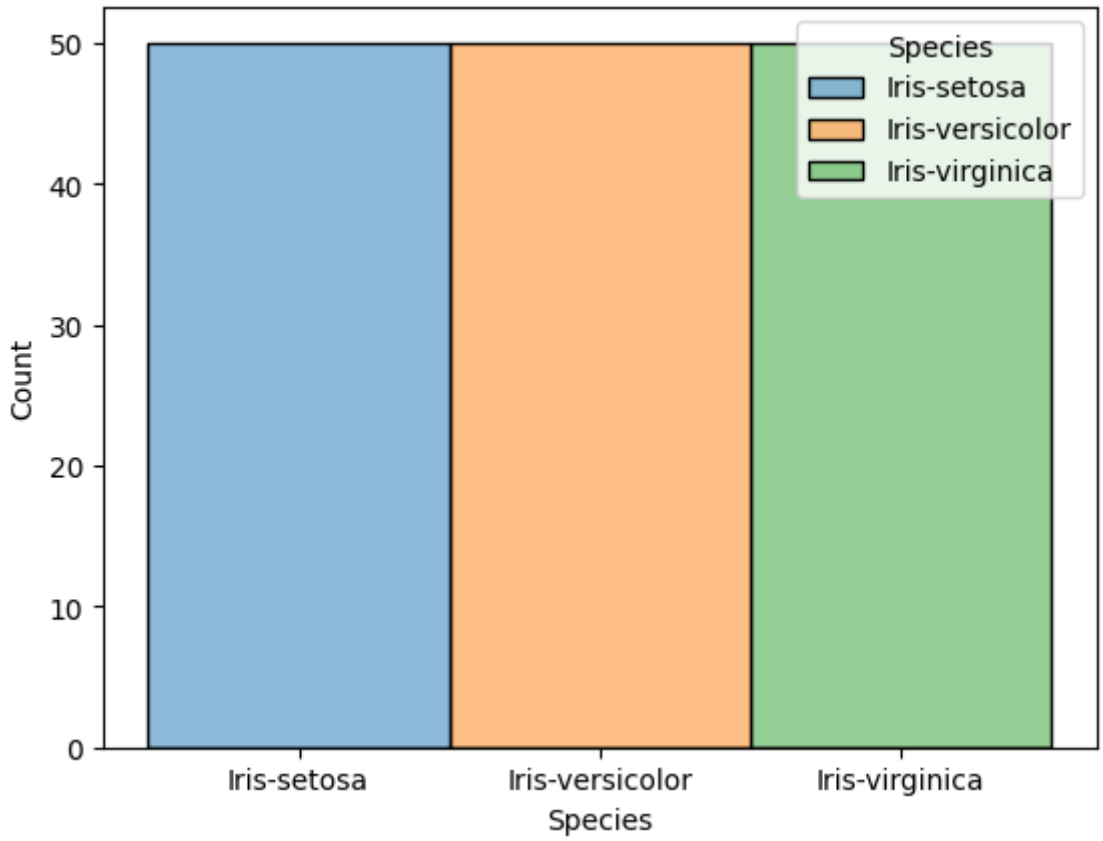
```
Id               0
SepalLengthCm    0
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0
dtype: int64
```

In [79]:

```
sns.histplot(data=df,x=df['Species'],hue='Species')
```

Out[79]:

```
<Axes: xlabel='Species', ylabel='Count'>
```

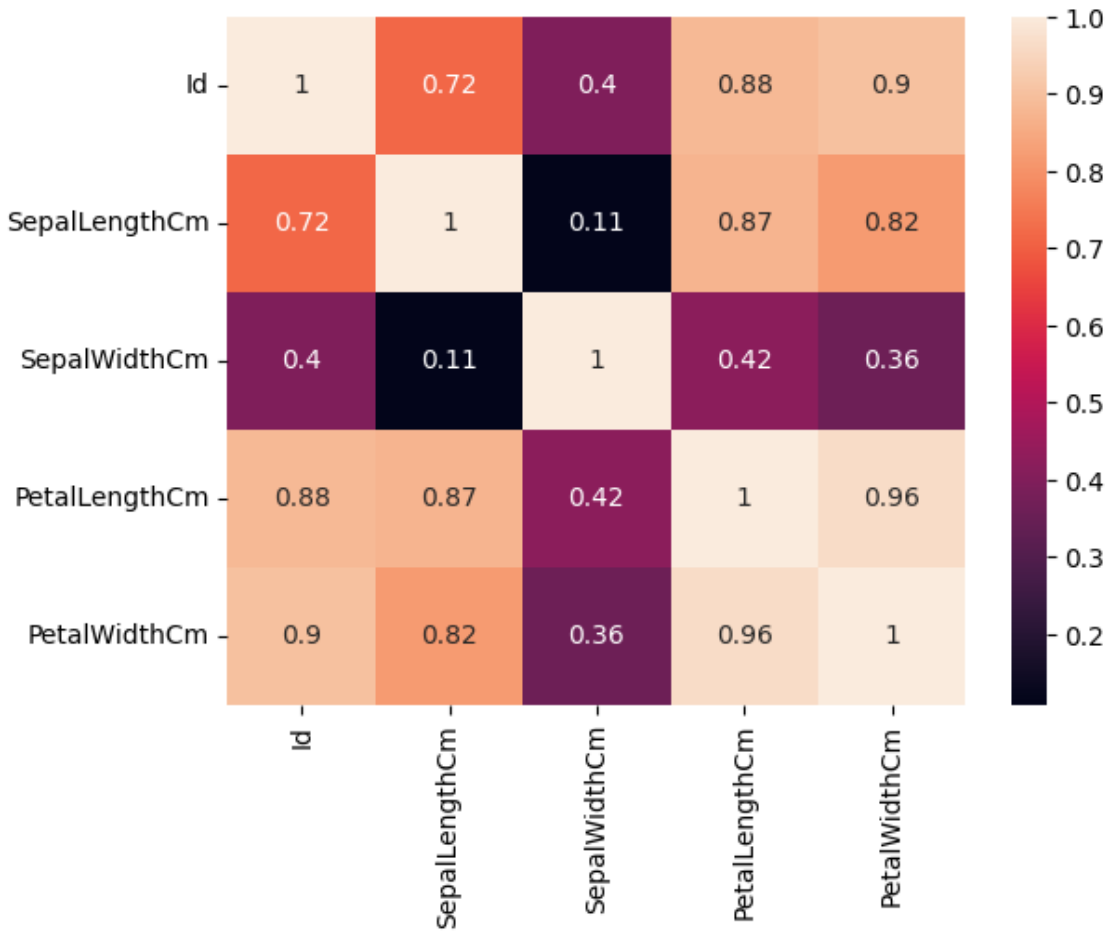


In [15]:

```
sns.heatmap(df.corr().abs(),annot = True)
```

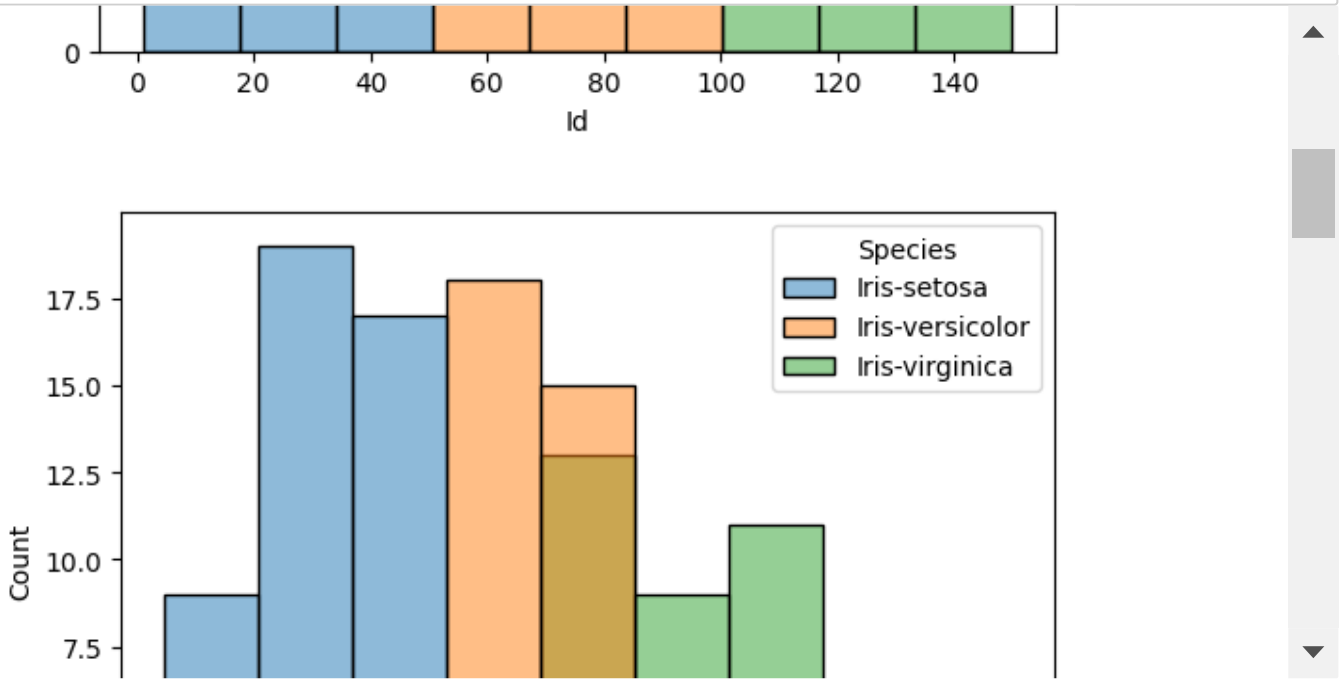
Out[15]:

<Axes: >



In [78]:

```
import matplotlib.pyplot as plt
#fig,axs = plt.subplots(ncols =4,nrows = 1,figsize =(40,20))
for column in df:
    if df[column].dtype!='object':
        sns.histplot(data=df, x=df[column],hue="Species")
        plt.show()
```



In [16]:

```
X = df.iloc[:,[1,4]]
```

In [17]:

```
Y = df.iloc[:,[-1]]
```

In [40]:

```
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size = 0.4,random_state = 45)
```

In [41]:

```
print(X_train.shape)
```

```
(90, 2)
```

In [42]:

```
from sklearn.naive_bayes import GaussianNB
model = GaussianNB()
model.fit(X_train,Y_train)
```

Out[42]:

```
GaussianNB()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [43]:

```
Y_pred = model.predict(X_test)
```

In [45]:

```
Y_pred
```

Out[45]:

```
array(['Iris-setosa', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-setosa',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-versicolor', 'Iris-setosa', 'Iris-virginica',
       'Iris-virginica', 'Iris-setosa', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-virginica',
       'Iris-versicolor', 'Iris-setosa', 'Iris-virginica',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-virginica', 'Iris-setosa',
       'Iris-virginica', 'Iris-setosa', 'Iris-setosa', 'Iris-versicolor',
       'Iris-setosa', 'Iris-setosa', 'Iris-versicolor', 'Iris-setosa',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-setosa', 'Iris-setosa', 'Iris-versicolor', 'Iris-virginica',
       'Iris-setosa', 'Iris-virginica', 'Iris-setosa', 'Iris-setosa',
       'Iris-versicolor', 'Iris-setosa', 'Iris-versicolor'], dtype='<U1
5')
```

In [48]:

```
Y_test
```

Out[48]:

	Species
0	Iris-setosa
43	Iris-setosa
129	Iris-virginica
3	Iris-setosa
34	Iris-setosa
44	Iris-setosa
38	Iris-setosa
105	Iris-virginica
123	Iris-virginica
140	Iris-virginica
28	Iris-setosa
125	Iris-virginica
113	Iris-virginica
103	Iris-virginica
133	Iris-virginica
35	Iris-setosa
145	Iris-virginica
142	Iris-virginica
40	Iris-setosa
87	Iris-versicolor
84	Iris-versicolor
85	Iris-versicolor
115	Iris-virginica
51	Iris-versicolor
4	Iris-setosa
112	Iris-virginica
92	Iris-versicolor
64	Iris-versicolor
10	Iris-setosa
91	Iris-versicolor
76	Iris-versicolor
96	Iris-versicolor
119	Iris-virginica
101	Iris-virginica
25	Iris-setosa
137	Iris-virginica
13	Iris-setosa

	Species
26	Iris-setosa
74	Iris-versicolor
30	Iris-setosa
33	Iris-setosa
82	Iris-versicolor
1	Iris-setosa
52	Iris-versicolor
93	Iris-versicolor
21	Iris-setosa
111	Iris-virginica
62	Iris-versicolor
117	Iris-virginica
37	Iris-setosa
45	Iris-setosa
66	Iris-versicolor
128	Iris-virginica
48	Iris-setosa
144	Iris-virginica
19	Iris-setosa
29	Iris-setosa
94	Iris-versicolor
47	Iris-setosa
90	Iris-versicolor

In [81]:

```
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report
accuracy = accuracy_score(Y_test,Y_pred)
accuracy
```

Out[81]:

```
0.9666666666666667
```

In [83]:

```
conf = confusion_matrix(Y_test,Y_pred)
print(conf)
```

```
[[24  0  0]
 [ 0 17  0]
 [ 0  2 17]]
```

In [85]:

```
print('Iris-Setosa')
tp = conf[0][0]
tn = conf[1][1]+conf[1][2]+conf[2][1]+conf[2][2]
fp = conf[1][0]+conf[2][0]
fn = conf[0][1]+conf[0][2]
total = tp+tn+fp+fn
error_rate = (fp+fn)/total
print('error = ',error_rate)
print('accuracy = ',1-error_rate)
```

```
Iris-Setosa
error =  0.0
accuracy =  1.0
```

In [86]:

```
print('Iris-versicolor')
tp = conf[1][1]
tn = conf[0][0]+conf[2][0]+conf[0][2]+conf[2][2]
fp = conf[0][1]+conf[0][2]
fn = conf[1][0]+conf[1][2]
total = tp+tn+fp+fn
error_rate = (fp+fn)/total
print('error = ',error_rate)
print('accuracy = ',1-error_rate)
```

```
Iris-versicolor
error =  0.0
accuracy =  1.0
```

In [87]:

```
print('Iris-virginica')
tp = conf[2][2]
tn = conf[0][0]+conf[1][1]+conf[0][1]+conf[1][0]
fp = conf[1][2]+conf[0][2]
fn = conf[2][1]+conf[2][0]
total = tp+tn+fp+fn
error_rate = (fp+fn)/total
print('error = ',error_rate)
print('accuracy = ',1-error_rate)
```

```
Iris-virginica
error =  0.03333333333333333
accuracy =  0.9666666666666667
```

In [89]:

```
print(classification_report(Y_test,Y_pred))
```

```
                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        24
Iris-versicolor       0.89      1.00      0.94        17
 Iris-virginica       1.00      0.89      0.94        19

       accuracy                           0.97        60
      macro avg       0.96      0.96      0.96        60
   weighted avg       0.97      0.97      0.97        60
```

In [ ]: