In [ ]:
```
#load sample dataset
import pandas as pd
import numpy as np
import seaborn as sns
df = sns.load_dataset('iris')
df.head()
```

Out[ ]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

In [ ]:
```
x = df.iloc[:, :-1]
y = df.iloc[:, -1:]
```

In [ ]:
```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=100)
model.fit(x, y)
model.predict([[10,4,2,6]])
```

```
C:\Users\Dell\AppData\Local\Temp\ipykernel_18292\255824680.py:3: DataConversionWarnin
g: A column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  model.fit(x, y)
c:\Users\Dell\miniconda3\envs\pandas_env\lib\site-packages\sklearn\base.py:450: UserW
arning: X does not have valid feature names, but RandomForestClassifier was fitted wi
th feature names
  warnings.warn(
```

Out[ ]:
```
array(['setosa'], dtype=object)
```

In [ ]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2)
predictions = model.predict(X_test)
predictions
```

Out[ ]:
```
array(['setosa', 'virginica', 'virginica', 'virginica', 'virginica',
       'versicolor', 'virginica', 'setosa', 'virginica', 'versicolor',
       'setosa', 'virginica', 'versicolor', 'setosa', 'versicolor',
       'setosa', 'versicolor', 'virginica', 'virginica', 'versicolor',
       'versicolor', 'setosa', 'setosa', 'versicolor', 'setosa', 'setosa',
       'virginica', 'setosa', 'virginica', 'virginica'], dtype=object)
```

In [ ]:
```
# Accuracy test

score = model.score(X_test, y_test)
print("The accuracy of the model is: ", score)
```

```
The accuracy of the model is:  1.0
```

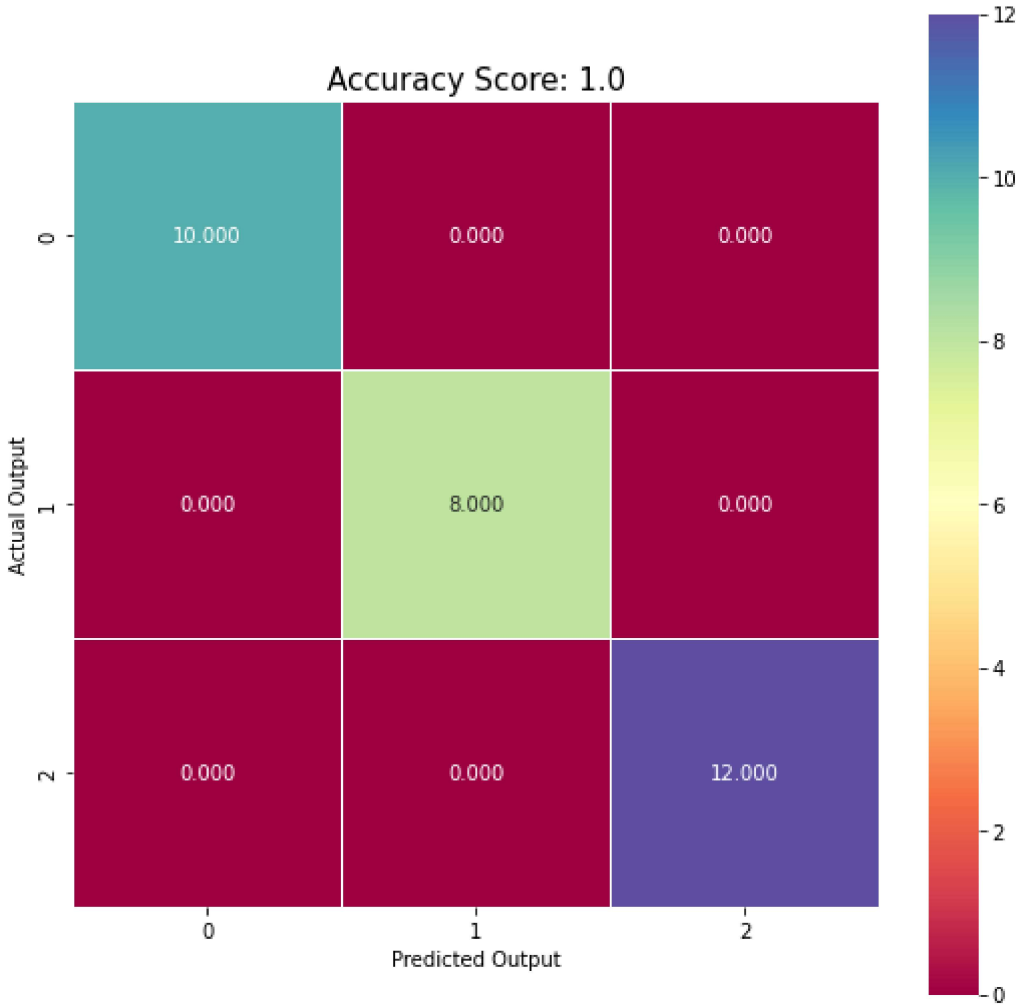
In [ ]:
```
from sklearn import metrics
print("Accuracy: ", metrics.accuracy_score(y_test, predictions))
```

```
Accuracy:  1.0
```

In [ ]:
```
from sklearn import metrics
cm = metrics.confusion_matrix(y_test, predictions)
cm
```

Out[ ]:
```
array([[10,  0,  0],
       [ 0,  8,  0],
       [ 0,  0, 12]], dtype=int64)
```

In [ ]:
```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(9,9))
sns.heatmap(cm, annot=True, fmt=".3f", linewidths=.5, square = True, cmap = 'Spectral'
plt.ylabel('Actual Output');
plt.xlabel('Predicted Output');
all_sample_title = 'Accuracy Score: {0}'.format(score)
plt.title(all_sample_title, size = 15);
```



In [ ]:
```
```