

```
In [16]: import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
```

```
In [17]: data = pd.read_csv("Iris.csv")
data
```

```
Out[17]:
```

	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 [18]: data.columns = ['id', 'sepal length', 'sepal width', 'petal length', 'petal width', 'species']
```

```
In [29]: # Split the dataset into training and testing sets
X = data.drop('species', axis=1)
y = data['species']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.9, random_state=70)
```

```
In [30]: # Train a random forest classifier on the training data
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)
```

```
Out[30]: RandomForestClassifier(random_state=42)
```

```
In [31]: # Make predictions on the test data
y_pred = clf.predict(X_test)
```

```
In [32]: # Evaluate the model's accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

Accuracy: 0.9925925925925926

```
In [ ]:
```