

```
# Step 1: Import necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_

# Step 2: Load the Iris dataset
iris = load_iris()

# Create a DataFrame
df = pd.DataFrame(data=iris.data, columns=iris.feature_names)
df['species'] = iris.target
df['species_name'] = df['species'].apply(lambda x: iris.target_names[x])

# Show first 5 rows
df.head()
```



	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species	species_name
0	5.1	3.5	1.4	0.2	0	setosa
1	4.9	3.0	1.4	0.2	0	setosa
2	4.7	3.2	1.3	0.2	0	setosa
3	4.6	3.1	1.5	0.2	0	setosa



Next
steps:

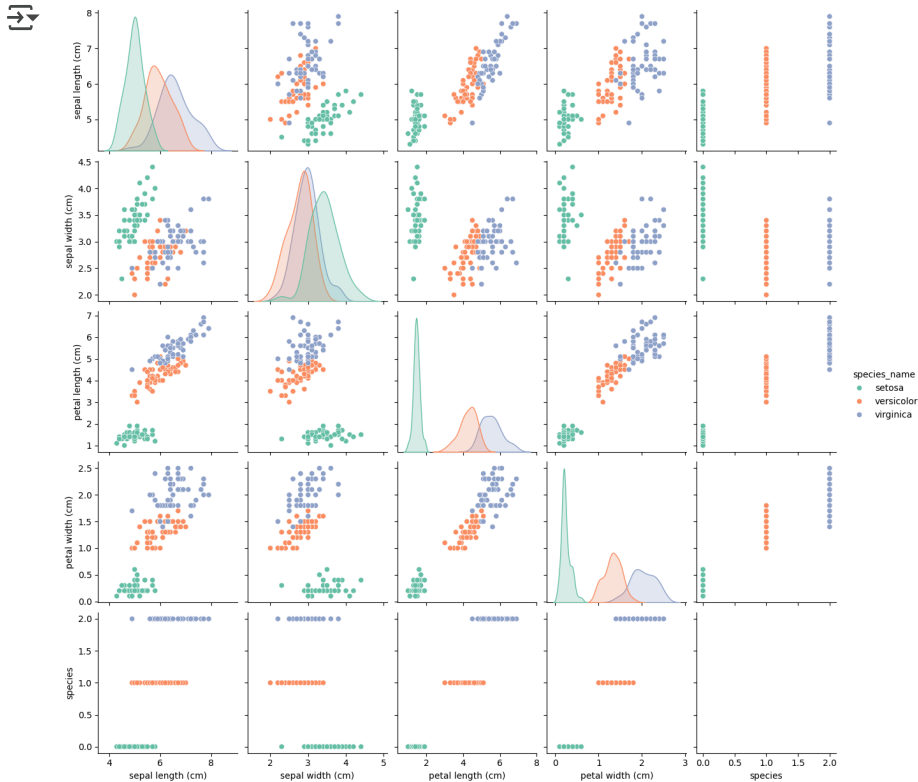
[Generate code with df](#)[View recommended plots](#)[New interactive sheet](#)

```
# Step 3: Visualize the dataset
sns.pairplot(df, hue='species_name', palette='Set2')
plt.show()
```



What can I help you build?





```
# Step 4: Prepare training and testing data
```

```
X = df[iris.feature_names]
```

```
y = df['species']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# Step 5: Train a classifier (Random Forest)
```

```
model = RandomForestClassifier(n_estimators=100, random_state=42)
```

```
model.fit(X_train, y_train)
```



▼ RandomForestClassifier



RandomForestClassifier(random_state=42)

```
# Step 6: Make predictions and evaluate
```

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

```
print("Accuracy:", accuracy_score(y_test, y_pred))
```

```
print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred))
```

```
print("\nClassification Report:\n", classification_report(y_test, y_pred, target_names=iris.target_names))
```



Accuracy: 1.0

Confusion Matrix:

```
[[10  0  0]
```

```
 [ 0  9  0]
```

```
 [ 0  0 11]]
```

Classification	Report:			
	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	1.00	1.00	1.00	9
virginica	1.00	1.00	1.00	11
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

```
# Step 7: Predict on new flower data
new_sample = [[5.1, 3.5, 1.4, 0.2]] # Example: Sepal Length, Sepal Width, Peta
prediction = model.predict(new_sample)
print(f"\nPredicted species: {iris.target_names[prediction[0]]}")
```



```
Predicted species: setosa
/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739:
warnings.warn(
```

[+ Code](#)[+ Text](#)