# Import necessary libraries

from sklearn import datasets

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.svm import SVC

from sklearn.metrics import classification\_report, accuracy\_score, confusion\_matrix

import matplotlib.pyplot as plt

import seaborn as sns

# Load the Iris dataset

iris = datasets.load\_iris()

X = iris.data

y = iris.target

# Split into training and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42, stratify=y)

# Standardize features (important for SVMs)

scaler = StandardScaler()

X\_train\_scaled = scaler.fit\_transform(X\_train)

X\_test\_scaled = scaler.transform(X\_test)

# Train an SVM classifier (RBF kernel by default)

model = SVC(kernel='rbf', C=1.0, gamma='scale', random\_state=42)

model.fit(X\_train\_scaled, y\_train)

# Predict on the test set

y\_pred = model.predict(X\_test\_scaled)

# Evaluate the model

print("Accuracy:", accuracy\_score(y\_test, y\_pred))

print("\nClassification Report:\n", classification\_report(y\_test, y\_pred))

# Confusion matrix

conf\_matrix = confusion\_matrix(y\_test, y\_pred)

sns.heatmap(conf\_matrix, annot=True, cmap='Blues', xticklabels=iris.target\_names, yticklabels=iris.target\_names)

plt.xlabel('Predicted')

plt.ylabel('True')

plt.title('Confusion Matrix')

plt.show()