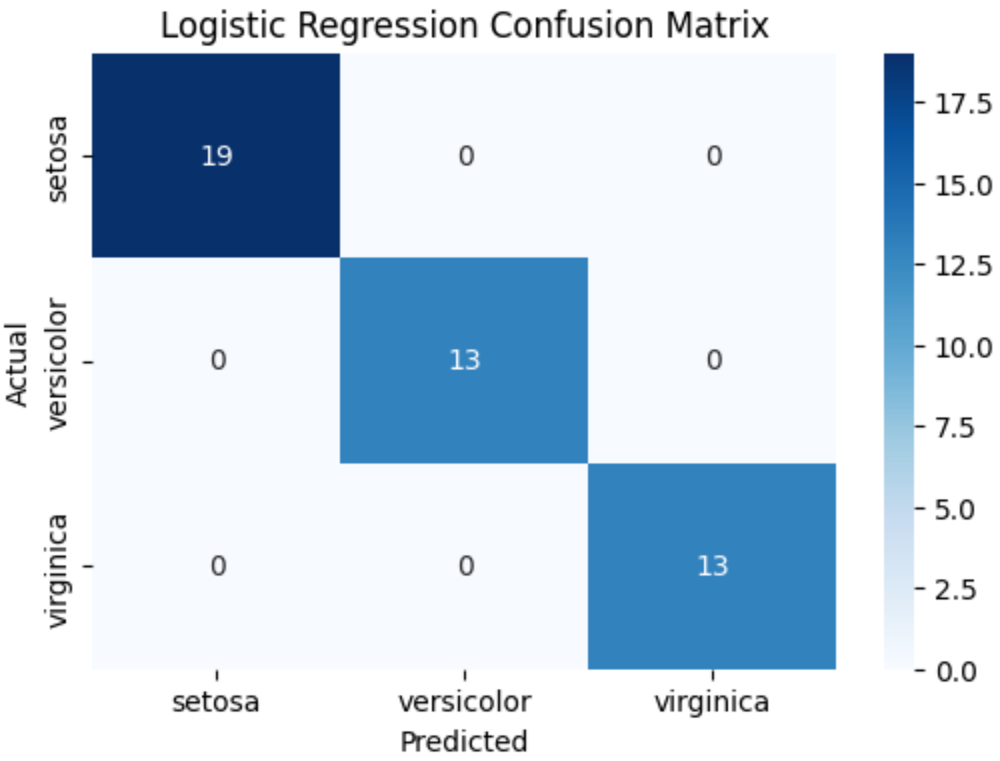
In [1]:
```
!pip install numpy pandas matplotlib seaborn scikit-learn
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: numpy in /home/jupyter-ra2311047010059/.local/lib/pyt
hon3.10/site-packages (2.0.2)
Requirement already satisfied: pandas in /home/jupyter-ra2311047010059/.local/lib/py
thon3.10/site-packages (2.2.3)
Requirement already satisfied: matplotlib in /home/jupyter-ra2311047010059/.local/li
b/python3.10/site-packages (3.10.3)
Requirement already satisfied: seaborn in /home/jupyter-ra2311047010059/.local/lib/p
ython3.10/site-packages (0.13.2)
Requirement already satisfied: scikit-learn in /home/jupyter-ra2311047010059/.local/
lib/python3.10/site-packages (1.6.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/tljh/user/lib/python
3.10/site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /home/jupyter-ra2311047010059/.local/
lib/python3.10/site-packages (from pandas) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /home/jupyter-ra2311047010059/.loca
l/lib/python3.10/site-packages (from pandas) (2025.1)
Requirement already satisfied: contourpy>=1.0.1 in /home/jupyter-ra2311047010059/.lo
cal/lib/python3.10/site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in /home/jupyter-ra2311047010059/.local/
lib/python3.10/site-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /home/jupyter-ra2311047010059/.l
ocal/lib/python3.10/site-packages (from matplotlib) (4.59.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /home/jupyter-ra2311047010059/.l
ocal/lib/python3.10/site-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /opt/tljh/user/lib/python3.10/sit
e-packages (from matplotlib) (24.0)
Requirement already satisfied: pillow>=8 in /home/jupyter-ra2311047010059/.local/li
b/python3.10/site-packages (from matplotlib) (11.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in /home/jupyter-ra2311047010059/.lo
cal/lib/python3.10/site-packages (from matplotlib) (3.2.1)
Requirement already satisfied: scipy>=1.6.0 in /home/jupyter-ra2311047010059/.local/
lib/python3.10/site-packages (from scikit-learn) (1.15.2)
Requirement already satisfied: joblib>=1.2.0 in /home/jupyter-ra2311047010059/.loca
l/lib/python3.10/site-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in /home/jupyter-ra231104701005
9/.local/lib/python3.10/site-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: six>=1.5 in /opt/tljh/user/lib/python3.10/site-packag
es (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

In [17]:
```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,classification_report,confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns


iris = load_iris()
X, y = iris.data, iris.target
target_names = iris.target_names

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_sta
model = LogisticRegression(max_iter=200)
model.fit(X_train, y_train)


y_pred = model.predict(X_test)
accuracy= accuracy_score(y_test,y_pred)
report = classification_report(y_test,y_pred)
# Evaluation
print("Accuracy:", accuracy_score(y_test, y_pred))
print("confusion matrix:\n", confusion_matrix(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))

plt.figure(figsize=(6,4))
sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', xticklabels
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Logistic Regression Confusion Matrix")
plt.show()
```

```
Accuracy: 1.0
confusion matrix:
 [[19  0  0]
 [ 0 13  0]
 [ 0  0 13]]
Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00        19
           1       1.00      1.00      1.00        13
           2       1.00      1.00      1.00        13

    accuracy                           1.00        45
   macro avg       1.00      1.00      1.00        45
weighted avg       1.00      1.00      1.00        45
```



In [ ]:

In [ ]:

```
1. Import required libraries:
   - Dataset loader
   - Train-test split function
   - Logistic Regression model
   - Evaluation metrics

2. Load the Iris dataset
   - Store features in variable X
   - Store labels in variable y

3. Split the dataset into training and testing sets
   - Use 70% data for training
   - Use 30% data for testing

4. Create Logistic Regression model
   - Set maximum iterations if needed

5. Train the model using training data (X_train, y_train)

6. Predict labels for test data (X_test)

7. Evaluate the model
   - Calculate accuracy
   - Generate classification report

8. Display results

End
```