In [21]:

```
import numpy as np
import matplotlib.pyplot as plt

import seaborn as sns

from sklearn import datasets
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

In [25]:

```
df = pd.read_csv("penguin.csv")

# Drop rows with missing values
df.dropna(inplace=True)
df.head()
```

Out[25]:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE
5	Adelie	Torgersen	39.3	20.6	190.0	3650.0	MALE

In [26]:

```
# Convert categorical variables to numerical using pandas
df['species'] = pd.Categorical(df['species']).codes
df['island'] = pd.Categorical(df['island']).codes
df['sex'] = pd.Categorical(df['sex']).codes

df.head()
```

Out[26]:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	0	2	39.1	18.7	181.0	3750.0	2
1	0	2	39.5	17.4	186.0	3800.0	1
2	0	2	40.3	18.0	195.0	3250.0	1
4	0	2	36.7	19.3	193.0	3450.0	1
5	0	2	39.3	20.6	190.0	3650.0	2

In [87]:

```
# Split the dataset into features and target variable
X = df[['species', 'island', 'culmen_length_mm', 'culmen_depth_mm', 'flipper_length_mm', 'body_mass_g']].values
y = df['sex'].values
```

In [88]:

```
# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)
```

In [89]:

```
# Train the Support Vector Machine (SVM) model
svm_model = SVC(kernel='linear')
svm_model.fit(X_train, y_train)
```

Out[89]:

▾ SVC

```
SVC(kernel='linear')
```

In [90]:

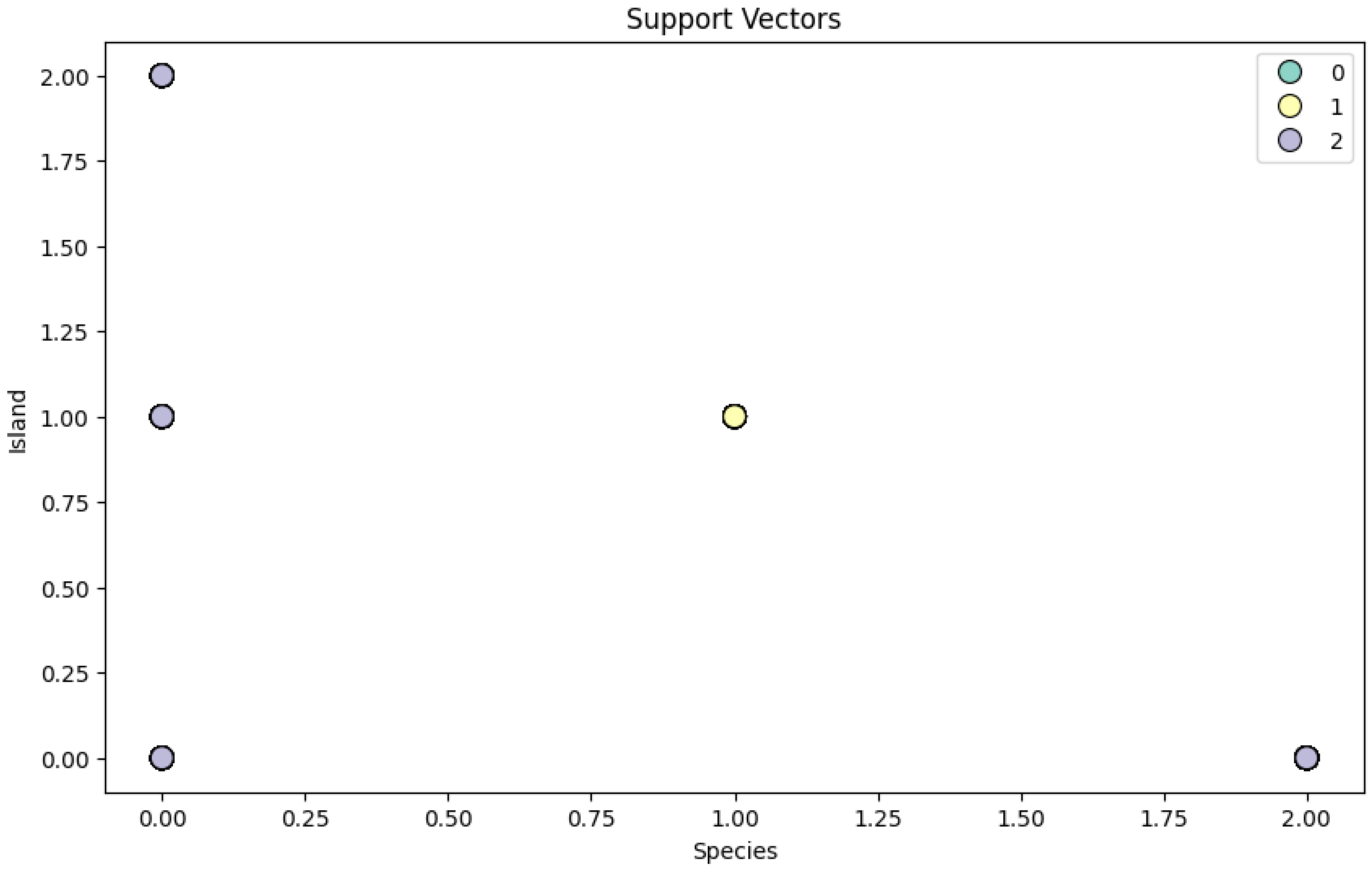
```
# Make predictions on the test set
y_pred = svm_model.predict(X_test)
```

In [98]:

```
# Plot the Support Vectors
plt.figure(figsize=(10, 6))
sns.scatterplot(x=X[:, 0], y=X[:, 1], hue=y, palette='Set3', edgecolor='k', s=100)
plt.xlabel('Species')
plt.ylabel('Island')
plt.title('Support Vectors')
```

Out[98]:

```
Text(0.5, 1.0, 'Support Vectors')
```



In [95]:

```
# Calculate accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

```
Accuracy: 0.9306930693069307
```

Model Interpretation

1. Why linear-SVM?
    - The dataset is a small but complex dataset that is linearly seperable.
    - Accuracy of our model is 93.07% which is a pretty good model for predecting the penguins.