Iris Dataset Linear Regression - Code and Output

Model 1: Sepal.Width to predict Sepal.Length

Code:

```
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import sklearn
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns

# Load dataset
iris = pd.read_csv(r"C:\Users\mukun\OneDrive\Documents\iris.csv")

# Split data
y = iris[['Sepal.Length']]
x = iris[['Sepal.Width']]
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size=0.3)

# Fit the linear regression model
lr = LinearRegression()
lr.fit(x_train, y_train)

# Make predictions
y_pred = lr.predict(x_test)

# Calculate mean squared error
from sklearn.metrics import mean_squared_error
mse_model_1 = mean_squared_error(y_test, y_pred)
print("Mean Squared Error (Model 1):", mse_model_1)
```

Output:

```
Mean Squared Error (Model 1): 0.0887759826152683
```

Model 2: Sepal.Width, Petal.Length, Petal.Width to predict Sepal.Length

Code:

```
# Split data for Model 2
y = iris[['Sepal.Length']]
x = iris[['Sepal.Width', 'Petal.Length', 'Petal.Width']]
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size=0.3)

# Fit the second linear regression model
lr2 = LinearRegression()
lr2.fit(x_train, y_train)

# Make predictions
y_pred_2 = lr2.predict(x_test)

# Calculate mean squared error
mse_model_2 = mean_squared_error(y_test, y_pred_2)
print("Mean Squared Error (Model 2):", mse_model_2)
```

Output:

```
Mean Squared Error (Model 2): 0.0817969124133592
```