

```

1 # Import necessary libraries
2 import pandas as pd
3 import numpy as np
4 import matplotlib.pyplot as plt
5 from statsmodels.tsa.holtwinters import SimpleExpSmoothing
6 from sklearn.metrics import mean_squared_error

1 # Load your dataset
2 df = pd.read_csv(r'/content/drive/MyDrive/PRN23039142546/SES_data.csv', index_col=0, parse_dates=True)

1 # Use only the 'Price' column
2 series = df['Price']
3
4 # Train-test split (80% train, 20% test)
5 train_size = int(len(series) * 0.8)
6 train, test = series[:train_size], series[train_size:]

1 # Fit SES model with fixed alpha
2 model = SimpleExpSmoothing(train)
3 model_fit = model.fit(smoothing_level=0.9, optimized=False) # Try alpha from 0.3 to 0.9 if needed

/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)

1 # Forecast for test period
2 preds = model_fit.forecast(len(test))

/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Predic
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In t
return get_prediction_index(

1 # Calculate RMSE
2 rmse = np.sqrt(mean_squared_error(test, preds))
3 print(f"SES RMSE: {rmse:.2f}")

SES RMSE: 65.20

1 # Plot the results
2 plt.figure(figsize=(12, 5))
3 plt.plot(train.index, train, label='Train', color='blue')
4 plt.plot(test.index, test, label='Actual', color='orange')
5 plt.plot(test.index, preds, label='Forecast', color='green')
6 plt.title(f"SES Forecast | RMSE: {rmse:.2f}")
7 plt.xlabel("Date")
8 plt.ylabel("Price")
9 plt.legend()
10 plt.grid(True)
11 plt.tight_layout()
12 plt.show()

```

