Import Libaries and Load Data

In [1]:

```
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import LSTM, Dense
import matplotlib.pyplot as plt
import warnings

# Suppress the warning messages
warnings.filterwarnings("ignore")

# Read the CSV file into a pandas DataFrame
file_path = "london_weather[1].csv"
df = pd.read_csv(file_path)

# Prepare the data
df['date'] = pd.to_datetime(df['date'], format='%Y%m%d')
df.set_index('date', inplace=True)
mean_temp_data = df['mean_temp'].dropna()
```

Normalize Data and Split into Train-Test Sets

In [2]:

```
# Normalize the data
scaler = MinMaxScaler()
normalized_data = scaler.fit_transform(mean_temp_data.values.reshape(-1, 1))

# Split data into training and testing sets
train_size = int(0.8 * len(normalized_data))
train_data, test_data = normalized_data[:train_size], normalized_data[train_size:]
```

Create Sequences for LSTM

In [3]:

```
# Create sequences for LSTM
def create_sequences(data, seq_length):
    sequences = []
    for i in range(len(data) - seq_length):
        sequence = data[i:i+seq_length]
        target = data[i+seq_length]
        sequences.append((sequence, target))
    return sequences

seq_length = 10  # You can adjust this sequence length
train_sequences = create_sequences(train_data, seq_length)
test_sequences = create_sequences(test_data, seq_length)

# Convert sequences to numpy arrays
X_train = np.array([seq for seq, _ in train_sequences])
y_train = np.array([target for _, target in train_sequences])
X_test = np.array([seq for seq, _ in test_sequences])
y_test = np.array([target for _, target in test_sequences])
```

Build and Train the LSTM Model

In [4]:

```
# Build the LSTM model
model = Sequential()
model.add(LSTM(50, activation='relu', input_shape=(seq_length, 1)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mean_squared_error')

# Train the model
model.fit(X_train, y_train, epochs=50, batch_size=32, verbose=1)
```

```
Epoch 1/50
383/383 [==============================] - 7s 9ms/step - loss: 0.0124
Epoch 2/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0044
Epoch 3/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0036
Epoch 4/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0030
Epoch 5/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0029
Epoch 6/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0028
Epoch 7/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0028
Epoch 8/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 9/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 10/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0028
Epoch 11/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0028
Epoch 12/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 13/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 14/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 15/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 16/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 17/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 18/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 19/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 20/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 21/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 22/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 23/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 24/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 25/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 26/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 27/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 28/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 29/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 30/50
383/383 [==============================] - 4s 9ms/step - loss: 0.0027
Epoch 31/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 32/50
383/383 [==============================] - 4s 9ms/step - loss: 0.0027
```

```
Epoch 33/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 34/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 35/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 36/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 37/50
383/383 [==============================] - 3s 8ms/step - loss: 0.0027
Epoch 38/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 39/50
383/383 [==============================] - 3s 9ms/step - loss: 0.0027
Epoch 40/50
383/383 [==============================] - 4s 10ms/step - loss: 0.0027
Epoch 41/50
383/383 [==============================] - 4s 10ms/step - loss: 0.0026
Epoch 42/50
383/383 [==============================] - 4s 10ms/step - loss: 0.0027
Epoch 43/50
383/383 [==============================] - 4s 10ms/step - loss: 0.0027
Epoch 44/50
383/383 [==============================] - 4s 11ms/step - loss: 0.0027
Epoch 45/50
383/383 [==============================] - 4s 11ms/step - loss: 0.0027
Epoch 46/50
383/383 [==============================] - 5s 12ms/step - loss: 0.0026
Epoch 47/50
383/383 [==============================] - 4s 11ms/step - loss: 0.0027
Epoch 48/50
383/383 [==============================] - 4s 10ms/step - loss: 0.0026
Epoch 49/50
383/383 [==============================] - 4s 11ms/step - loss: 0.0026
Epoch 50/50
383/383 [==============================] - 4s 9ms/step - loss: 0.0027
```

Out[4]:

```
<keras.src.callbacks.History at 0x1fde9343730>
```

Make Predictions and Plot Results

In [6]:

```python
# Make predictions
train_predictions = model.predict(X_train)
test_predictions = model.predict(X_test)

# Inverse transform the predictions to the original scale
train_predictions = scaler.inverse_transform(train_predictions)
test_predictions = scaler.inverse_transform(test_predictions)

test_predictions
```

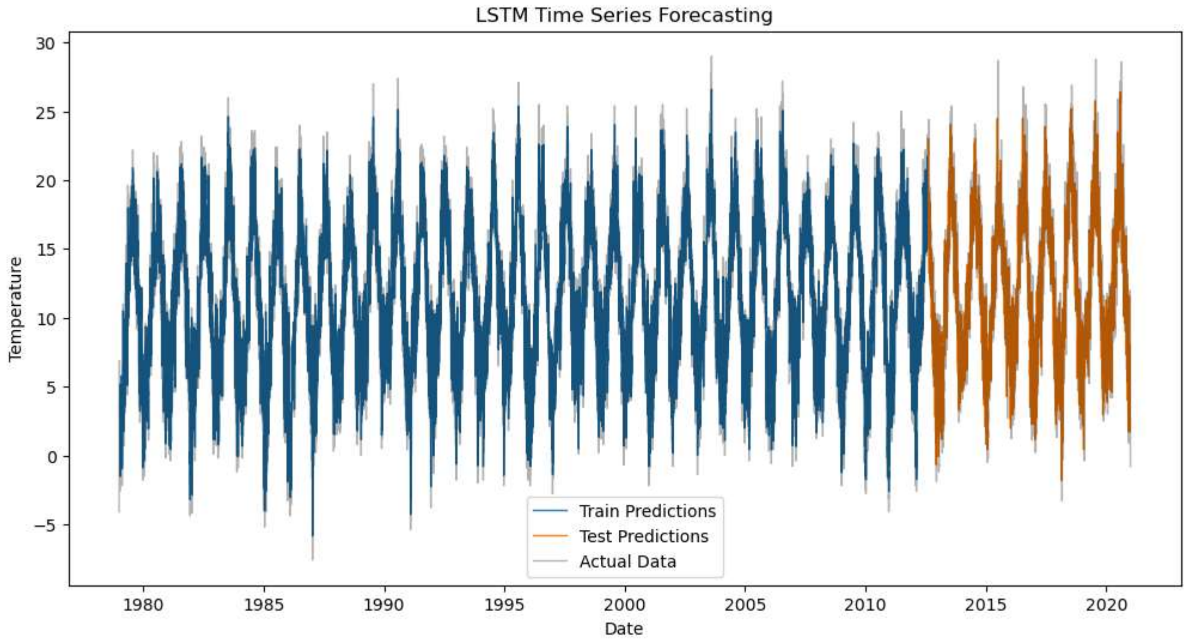
```
383/383 [==============================] - 2s 5ms/step
96/96 [==============================] - 1s 5ms/step
```

Out[6]:

```
array([[21.888954 ],
       [23.019127 ],
       [22.759136 ],
       ...,
       [ 1.6819607],
       [ 4.0202446],
       [ 3.6736486]], dtype=float32)
```

In [8]:

```python
plt.figure(figsize=(12, 6))
plt.plot(mean_temp_data.index[seq_length:len(train_predictions)+seq_length], train_
plt.plot(mean_temp_data.index[len(train_predictions)+seq_length:len(train_predicti
plt.plot(mean_temp_data.index, mean_temp_data.values, label='Actual Data', color='
plt.legend()
plt.xlabel('Date')
plt.ylabel('Temperature')
plt.title('LSTM Time Series Forecasting')
plt.show()
```


LSTM Time Series Forecasting
Temperature
Train Predictions
Test Predictions
Actual Data
Date