```
clc;
clear all;
close all;

% === Load and Normalize Data ===
solar = readtable('Corrected_House_1_Solar_Production.xlsx');
consumption = readtable('House_1_Consumption_Reshaped.xlsx');

% Combine data: [Solar, Consumption]
data = [solar.Power consumption.Power];

% === Min-Max Normalization ===
minVals = min(data);
maxVals = max(data);
dataNorm = (data - minVals) ./ (maxVals - minVals);

% === Sequence Settings ===
sequenceLength = 30;
numSamples = size(dataNorm, 1) - sequenceLength;

% === Prepare Sequences ===
XTrain = cell(numSamples, 1);
YTrain = zeros(numSamples, 1);

for i = 1:numSamples
    XTrain{i} = dataNorm(i:i+sequenceLength-1, :)';  % [features x timesteps]
    YTrain(i) = dataNorm(i+sequenceLength, 2);       % Target: next consumption
end

% === Define LSTM Network ===
layers = [
    sequenceInputLayer(2)
    lstmLayer(50, 'OutputMode', 'last')
    fullyConnectedLayer(1)
    reluLayer                       % Ensures output ≥ 0
    regressionLayer
];

options = trainingOptions('adam', ...
    'MaxEpochs', 150, ...
    'GradientThreshold', 1, ...
    'InitialLearnRate', 0.005, ...
    'Verbose', 0, ...
    'Plots', 'training-progress');

% === Train the Network ===
net = trainNetwork(XTrain, YTrain, layers, options);

% === Predict Next-Day Consumption ===
lastSequence = dataNorm(end-sequenceLength+1:end, :)';
predictedNorm = predict(net, {lastSequence});

% === Denormalize Prediction ===
minCons = minVals(2);
maxCons = maxVals(2);
predictedKWh = predictedNorm * (maxCons - minCons) + minCons;

% === Clamp Just in Case ===
predictedKWh = max(0, predictedKWh);
```

```
% === Output ===
fprintf("✅ Predicted next day consumption (normalized): %.4f\n", predictedNorm);
fprintf("✅ Actual predicted consumption (kWh): %.4f\n", predictedKWh);

% === Create timeseries for Simulink ===
tsTime = [0 8760]';  % Duration of simulation in hours
tsValues = [predictedKWh predictedKWh]';  % Constant value throughout

predictedTS = timeseries(tsValues, tsTime);
predictedTS.Name = 'PredictedConsumption';
assignin('base', 'predictedTS', predictedTS);  % Ready for From Workspace
```