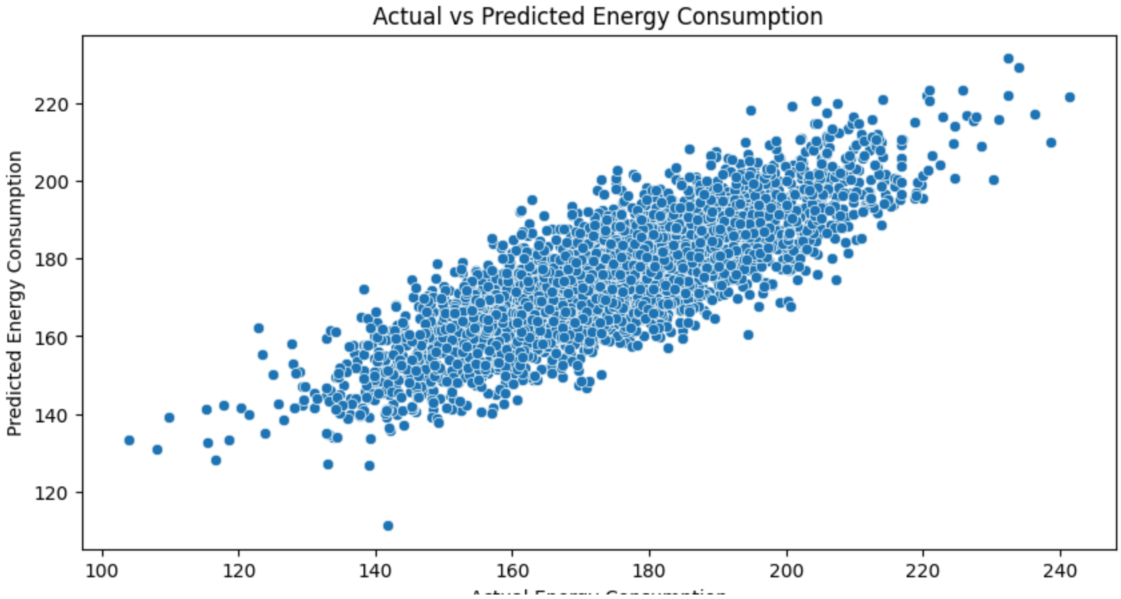
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score

# 1. Generate synthetic dataset
np.random.seed(42)
date_range = pd.date_range(start='2023-01-01', end='2024-12-31', freq='H')
n = len(date_range)

data = pd.DataFrame({
    'datetime': date_range,
    'temperature': np.random.normal(loc=25, scale=5, size=n),  # Celsius
    'humidity': np.random.normal(loc=60, scale=10, size=n),    # Percent
    'occupancy': np.random.poisson(lam=50, size=n),            # People count
})

# Simulate energy consumption with noise
data['energy_consumption'] = (
    20 + 1.5 * data['occupancy'] +
    2.0 * data['temperature'] +
    0.5 * data['humidity'] +
    np.random.normal(scale=10, size=n)
)

# 2. Feature engineering
data['hour'] = data['datetime'].dt.hour
data['day_of_week'] = data['datetime'].dt.dayofweek

# 3. Model building
features = ['temperature', 'humidity', 'occupancy', 'hour', 'day_of_week']
X = data[features]
y = data['energy_consumption']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)

# 4. Evaluation
y_pred = model.predict(X_test)

print("Model Coefficients:", dict(zip(features, model.coef_)))
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
print("R^2 Score:", r2_score(y_test, y_pred))

# 5. Visualization
plt.figure(figsize=(10, 5))
sns.scatterplot(x=y_test, y=y_pred)
plt.xlabel("Actual Energy Consumption")
plt.ylabel("Predicted Energy Consumption")
plt.title("Actual vs Predicted Energy Consumption")
plt.show()
```

```
<ipython-input-2-d6cd0876ea40>:11: FutureWarning: 'H' is deprecated and will be removed in a future version, please use 'h' instead
  date_range = pd.date_range(start='2023-01-01', end='2024-12-31', freq='H')
Model Coefficients: {'temperature': np.float64(2.0113888817739545), 'humidity': np.float64(0.5026685314455257), 'occupancy': np.floa
Mean Squared Error: 106.27370007038388
R^2 Score: 0.6902323526391327
```


Actual vs Predicted Energy Consumption
Predicted Energy Consumption
Actual Energy Consumption