410-Asif-Sayyed-ML-Case-Study-Question-1

In [3]:
```
# importing libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

%matplotlib inline
sns.set_style('whitegrid')
plt.rcParams['figure.dpi']=200
```

Loading and exploring the data

In [5]:
```
# reading the data from the CSV file.
df = pd.read_csv("D:\College\Academics\SEM 4\Machine Learning\Datasets\hvac.csv")
```

In [6]:
```
# checking the dimensions of the dataset.
df.shape
```

Out[6]: (45, 8)

The dataset has **45** records and **8** fields

In [7]:
```
# looking at the top 5, bottom 5 and random 5 records in the dataset.
display(df.head(),df.tail(),df.sample(5))
```

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Building Type	HVAC System	Average Temperature in last 24 hours (C)	Energy Consumption (kWh)
0	270	15	30	2.0	Residential	Central AC	28	380
1	273	20	28	1.5	Commercial	Split AC	26	420
2	276	18	26	1.8	Residential	Window AC	24	390
3	276	12	32	2.5	Residential	Central AC	32	320
4	276	22	20	1.2	Commercial	Split AC	18	480

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Building Type	HVAC System	Average Temperature in last 24 hours (C)	Energy Consumption (kWh)
40	274	20	28	1.5	Commercial	Split AC	28	420
41	277	18	22	1.8	Residential	Window AC	22	410
42	273	12	32	2.5	Residential	Central AC	32	320
43	277	22	20	1.2	Residential	Split AC	20	480
44	275	16	18	1.9	Residential	Window AC	18	400

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Building Type	HVAC System	Average Temperature in last 24 hours (C)	Energy Consumption (kWh)
22	276	20	28	1.5	Residential	Split AC	28	420
38	273	17	25	2.0	Residential	Window AC	25	380
21	273	14	15	2.2	Residential	Central AC	15	360
9	273	23	10	1.1	Residential	Central AC	10	510
39	275	15	30	2.0	Residential	Central AC	30	380

In [8]:
```
# looking at the dataframe information
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45 entries, 0 to 44
Data columns (total 8 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   Room Area (sq. ft.)                     45 non-null     int64
 1   Number of Appliances                    45 non-null     int64
 2   Outside Temperature (C)                 45 non-null     int64
 3   Insulation Thickness (inches)           45 non-null     float64
 4   Building Type                           45 non-null     object
 5   HVAC System                             45 non-null     object
 6   Average Temperature in last 24 hours (C)  45 non-null     int64
 7   Energy Consumption (kWh)                45 non-null     int64
dtypes: float64(1), int64(5), object(2)
memory usage: 2.9+ KB
```

Inference:

- There are no null values in the dataset
- In this dataset there are total 8 columns
- From the 8, six of the columns are numeric and two are categorical
- Categorical nominal columns: Building Type, HVAC System
- Numerical continous column: Insulation Thickness, Room Area, Outside Temperature, Energy Consumption, Average Temperature in last 24 hours
- Numerical discrete columns: Number of Application

Note: Usually temperature, room area can be continous as well but based on the granularity of the data I've concluded that it is discrete

In [9]:
```
# looking at the summary statistics
df.describe()
```

Out[9]:

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Average Temperature in last 24 hours (C)	Energy Consumption (kWh)
count	45.000000	45.000000	45.000000	45.000000	45.000000	45.000000
mean	274.822222	17.577778	23.044444	1.782222	22.844444	407.111111
std	1.556349	3.026316	6.201010	0.376158	6.226613	46.837350
min	270.000000	12.000000	10.000000	1.100000	10.000000	320.000000
25%	274.000000	16.000000	18.000000	1.500000	18.000000	380.000000
50%	275.000000	18.000000	23.000000	1.800000	23.000000	400.000000
75%	276.000000	20.000000	28.000000	2.000000	28.000000	430.000000
max	277.000000	23.000000	33.000000	2.500000	33.000000	510.000000

Inference:

Room Area (sq. ft.):

- The buildings in the dataset have room areas ranging from 270 sq. ft. to 277 sq. ft.
- The mean room area is approximately 274.82 sq. ft.

Number of Appliances:

- The number of appliances in the buildings ranges from 12 to 23.
- On average, there are approximately 17.58 appliances per building.

Outside Temperature (°C):

- The outside temperatures at the building locations range from 10°C to 33°C.
- The average outside temperature is around 23.04°C.

Insulation Thickness (inches):

- The insulation thickness in the buildings varies from 1.1 inches to 2.5 inches.
- The mean insulation thickness is approximately 1.78 inches.

Average Temperature in the last 24 hours (°C):

- The average temperatures in the last 24 hours range from 10°C to 33°C.
- The mean average temperature is about 22.84°C.

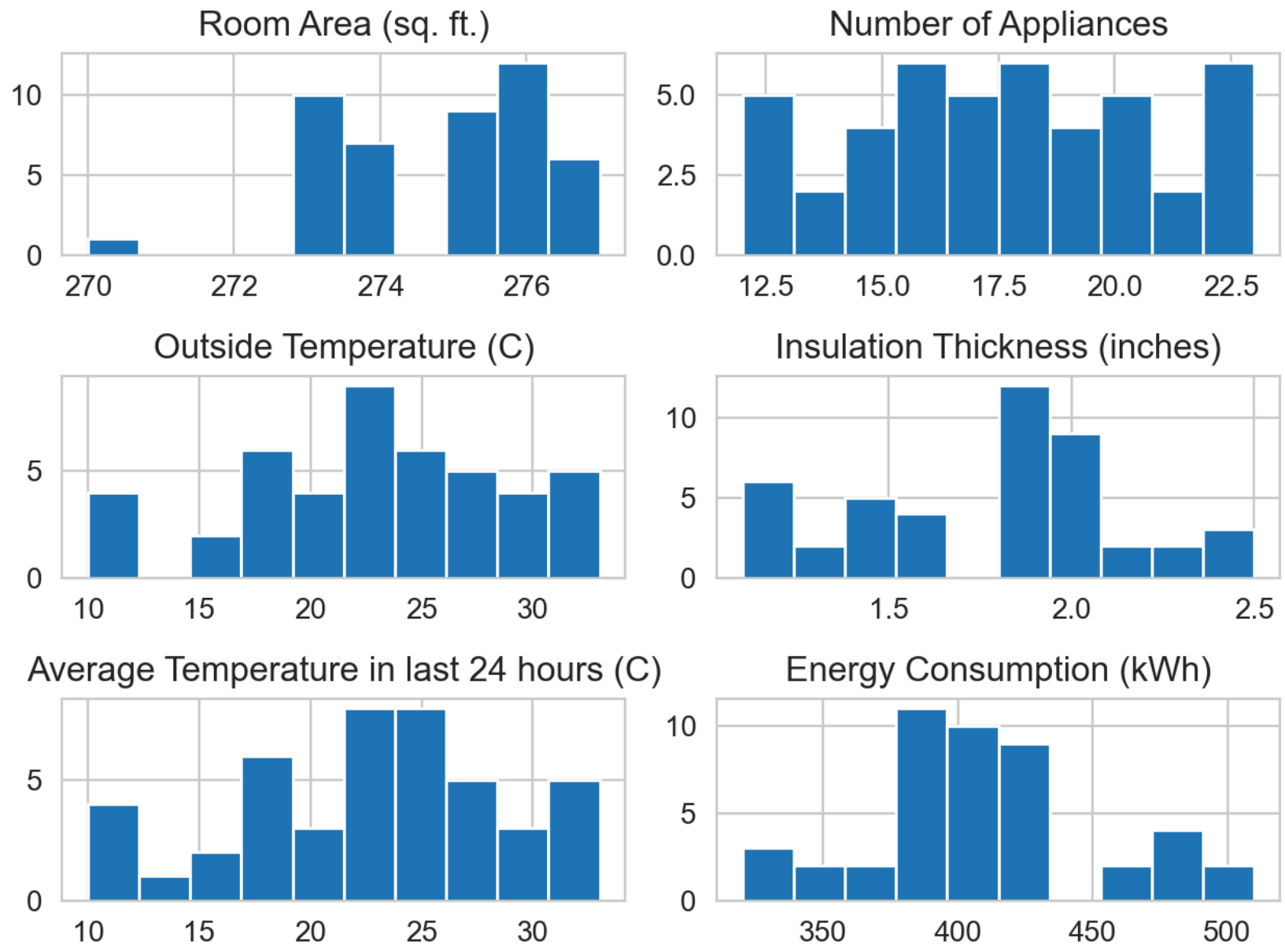
Energy Consumption (kWh):

- The energy consumption of buildings varies from 320 kWh to 510 kWh.
- The average energy consumption is approximately 407.11 kWh.

Creating histograms for each column

In [10]:
```
# fetching the names of the columns from the dataframe
cols = df.columns.to_list()

# making a histogram
df[cols].hist()
# adjusting the padding around the histogram
plt.tight_layout()
plt.show()
```



Inference:

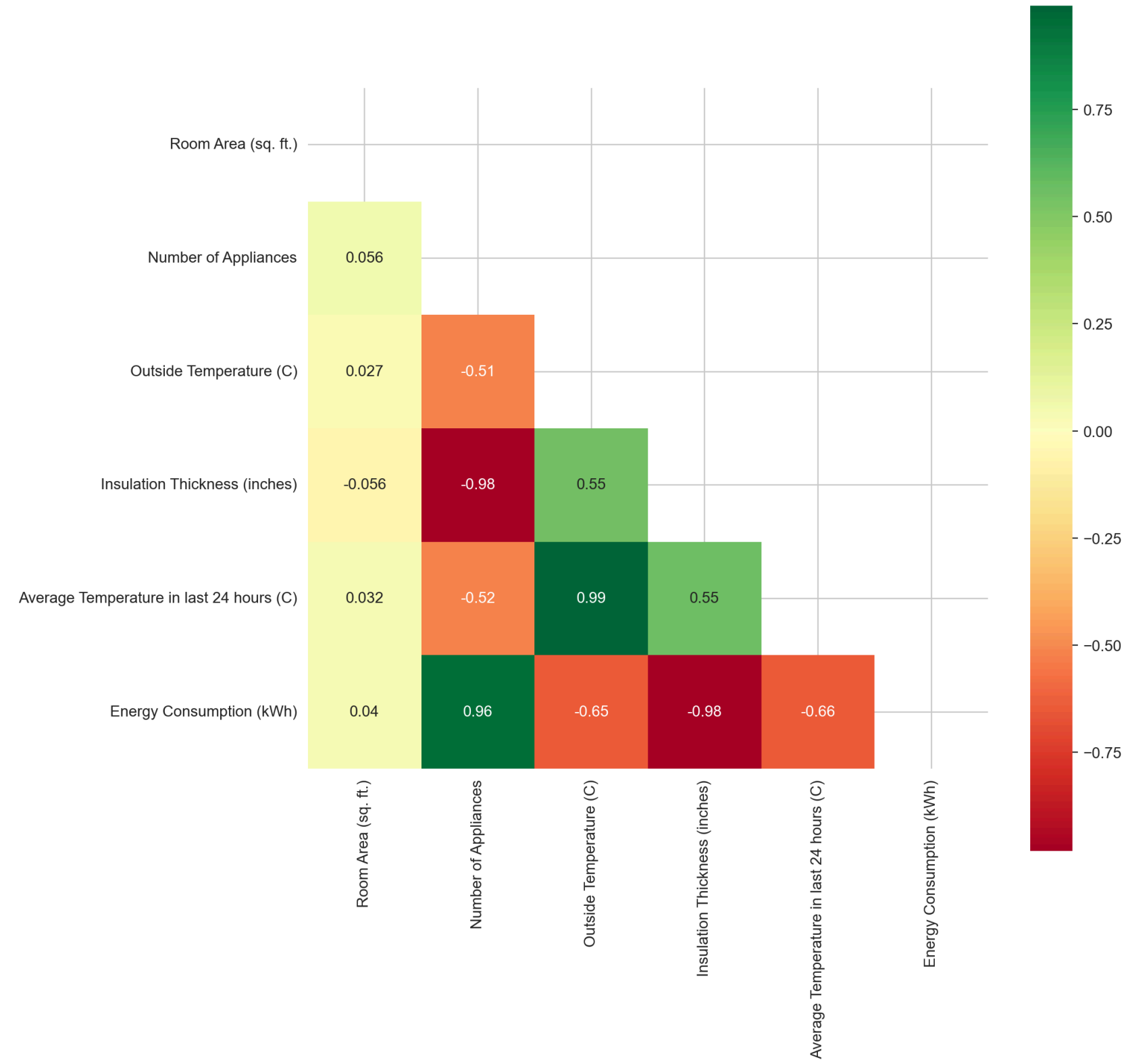
- *Room Area:* Since the majority of the data points lie on the right of the median, we can see the data is left skewed, not perfectly but we can say that.
- *Number of Appliances:* Here the distribution is multimodal i.e There are more than two peak
- *Outside Temperature:* Here the distribution is close to gaussian distribution
- *Insulation Thickness:* Here again we can
- *Average Temperature in last 24 hours:*
- *Energy Consumption:*

Creating a correlation heatmap and correlation matrix

In [11]:
```
# creating a variable containing all the numerical variables
corr = df[['Room Area (sq. ft.)',
           'Number of Appliances',
           'Outside Temperature (C)',
           'Insulation Thickness (inches)',
           'Average Temperature in last 24 hours (C)',
           'Energy Consumption (kWh)']].corr()

mask= np.triu(np.ones_like(corr, dtype=bool))

# defining the figure size
fig, ax = plt.subplots(figsize = (10,10))
# creating the heatmap and turning the annotation on
sns.heatmap(corr, mask=mask,
            cmap='RdYlGn',
            annot= True,
            center=0,
            square=True)
plt.show()
```



In [12]:
```
# creating a correlation matrix
corr.corr()
```

Out[12]:

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Average Temperature in last 24 hours (C)	Energy Consumption (kWh)
Room Area (sq. ft.)	1.000000	0.105114	-0.174426	-0.111625	-0.171919	0.116939
Number of Appliances	0.105114	1.000000	-0.898384	-0.999791	-0.900605	0.996932
Outside Temperature (C)	-0.174426	-0.898384	1.000000	0.904398	0.999965	-0.928880
Insulation Thickness (inches)	-0.111625	-0.999791	0.904398	1.000000	0.906562	-0.998065
Average Temperature in last 24 hours (C)	-0.171919	-0.900605	0.999965	0.906562	1.000000	-0.930755
Energy Consumption (kWh)	0.116939	0.996932	-0.928880	-0.998065	-0.930755	1.000000

Inference:

1. *Room Area vs. Number of Appliances:*

- There is a weak positive correlation between the room area and the number of appliances. Larger room areas tend to have slightly more appliances.

2. *Room Area vs. Outside Temperature:*

- There is a weak negative correlation between room area and outside temperature. Larger room areas may be associated with slightly lower outside temperatures.

3. *Room Area vs. Insulation Thickness:*

- There is a weak negative correlation between room area and insulation thickness. Larger room areas may have slightly thinner insulation.

4. *Room Area vs. Average Temperature in the last 24 hours:*

- There is a weak negative correlation between room area and the average temperature in the last 24 hours. Larger room areas may be associated with slightly lower average temperatures.

5. *Room Area vs. Energy Consumption:*

- There is a weak positive correlation between room area and energy consumption. Larger room areas may be associated with slightly higher energy consumption.

6. *Number of Appliances vs. Outside Temperature:*

- There is a strong negative correlation between the number of appliances and outside temperature. These don't have a lot of correlation between them.

7. *Number of Appliances vs. Insulation Thickness:*

- There is a very strong negative correlation between the number of appliances and insulation thickness. Buildings with more appliances tend to have thinner insulation.

8. *Number of Appliances vs. Average Temperature in the last 24 hours:*

- There is a strong negative correlation between the number of appliances and the average temperature in the last 24 hours. Buildings with more appliances tend to have lower average temperatures.

9. *Number of Appliances vs. Energy Consumption:*

- There is a very strong positive correlation between the number of appliances and energy consumption. Buildings with more appliances tend to have higher energy consumption.

10. *Outside Temperature vs. Insulation Thickness:*

- There is a strong positive correlation between outside temperature and insulation thickness. Buildings with higher outside temperatures tend to have thicker insulation.

11. *Outside Temperature vs. Average Temperature in the last 24 hours:*

- There is an extremely strong positive correlation between outside temperature and the average temperature in the last 24 hours. Higher outside temperatures coincide with higher average temperatures in the last 24 hours.

12. *Outside Temperature vs. Energy Consumption:*

- There is a strong negative correlation between outside temperature and energy consumption. Higher outside temperatures tend to be associated with lower energy consumption.

13. *Insulation Thickness vs. Average Temperature in the last 24 hours:*

- There is a strong positive correlation between insulation thickness and the average temperature in the last 24 hours. Thicker insulation is associated with higher average temperatures.

14. *Insulation Thickness vs. Energy Consumption:*

- There is an extremely strong negative correlation between insulation thickness and energy consumption. Thicker insulation is associated with significantly lower energy consumption.

15. *Average Temperature in the last 24 hours vs. Energy Consumption:*

- There is a strong negative correlation between the average temperature in the last 24 hours and energy consumption. Higher average temperatures in the last 24 hours are associated with lower energy consumption.

Required Analysis: A)

- The primary feature with the highest correlation with Energy Consumption is Number of Appliances with the value of 0.999965
  - High Positive Correlation: Building with more appliances will require more energy to operate hence it would make sense that more appliances will lead to higher Energy Consumption.
- The secondary feature with the highest correlation with Energy Consumption is Insulation Thickness with the value of -0.998065
  - High Negative Correlation: Building that have thicker insulation will have overall lower Energy consumption as the insulation will help maintaining a stable temperature indoors.

Required Analysis: B)

Identifying

- In order to decide which feature may not contribute significantly to prediction accuracy, we will be looking at the correlation matrix again
- Room Area would be the one feature that may not contribute a lot to the prediction accuracy as it has the correlation value of 0.116939, which is comparetively lower than the rest of the features
- Since Room Area has barely any impact on Energy consumption directly but it does contribute indirectly as higher Room Area has a weak correlation to the Number of Appliances
- in terms of logical thinking, it would require more energy to cooldown or heatup a room with a larger area but since the range of the room area from 270 sq.ft to 277 sq.ft. there is not any drastic increase in Room Area that would highly impact the energy consumption.

Mitigating

- Keeping all those into consideration, I would still be keeping the Room Area feature it it would probably help us explain some of the variance on the dependent variable and making an assumption that it will not change the results drastically

Performing Multiple Linear Regression

Data Preprocessing

In [13]:
```
# turning the categorical variables into 0 and 1
df_new = pd.get_dummies(df, columns=['Building Type', 'HVAC System'],
                        drop_first=True)
# looking at the encoded dataframe
df_new.head()
```

Out[13]:

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Average Temperature in last 24 hours (C)	Energy Consumption (kWh)	Building Type_ Residential	HVAC System_ Split AC	HVAC System_ Window AC
0	270	15	30	2.0	28	380	True	False	False
1	273	20	28	1.5	26	420	False	True	False
2	276	18	26	1.8	24	390	True	False	True
3	276	12	32	2.5	32	320	True	False	False
4	276	22	20	1.2	18	480	False	True	False

In [14]:
```
# creating a variable containing independent variables (Xs)
X = df_new.drop("Energy Consumption (kWh)",axis=1)
# creating a variable containing dependent variable (Y)
Y = df_new['Energy Consumption (kWh)']
display(X.head(),Y.head())
```

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Average Temperature in last 24 hours (C)	Building Type_ Residential	HVAC System_ Split AC	HVAC System_ Window AC
0	270	15	30	2.0	28	True	False	False
1	273	20	28	1.5	26	False	True	False
2	276	18	26	1.8	24	True	False	True
3	276	12	32	2.5	32	True	False	False
4	276	22	20	1.2	18	False	True	False

```
0    380
1    420
2    390
3    320
4    480
Name: Energy Consumption (kWh), dtype: int64
```

Splitting the dataset

splitting the dataset into training and testing sets

X_train, X_test,y_train, y_test = train_test_split(X,Y, test_size=0.2, random_state=1)

Building Multiple Linear Regression Model

creating the model

mlr = LinearRegression()

fitting the variables

mlr.fit(X_train, y_train)

In [15]:
```
# creating the model
mlr = LinearRegression()
# fitting the variables
mlr.fit(X, Y)
```

Out[15]:
```
▾ LinearRegression
LinearRegression()
```

checking the accuracy of the model

y_pred = mlr.predict(X_test) #rmse = np.sqrt(mean_squared_error(y_test, y_pred)) print(f"Mean squared error on the testing dataset is {rmse}")

In [16]:
```
# checking the accuracy of the model
y_pred = mlr.predict(X)
rmse = np.sqrt(mean_squared_error(Y, y_pred))
print(f"Mean squared error on the testing dataset is {rmse}")
```

```
Mean squared error on the testing dataset is 5.974304053772277
```

In [17]:
```
print("Coefficients:",mlr.coef_)
print("Intercept:",mlr.intercept_)
```

```
Coefficients: [ 7.13841205e-02 -1.67074708e+00 -3.34951686e+00 -1.27965071e+02
  2.22303293e+00 -1.10739380e+00 -5.55699838e+00  4.29009588e-01]
Intercept: 673.8486030780857
```

Testing on new datapoints

In [18]:
```
# Define the data for the points
data = {
    'Room Area (sq. ft.)': [279, 277, 276],
    'Number of Appliances': [16, 22, 14],
    'Outside Temperature (C)': [20, 15, 25],
    'Insulation Thickness (inches)': [1.7, 1.5, 2.2],
    'Average Temperature in last 24 hours (C)': [19, 14, 26],
    'Building Type_ Residential': [True, False, True],
    'HVAC System_ Split AC': [False, True, False],
    'HVAC System_ Window AC': [False, False, True],
    'Energy Consumption (kWh)': [385, 425, 350]
}

# Create a DataFrame
df_points = pd.DataFrame(data)
display(df_points.head())
```

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Average Temperature in last 24 hours (C)	Building Type_ Residential	HVAC System_ Split AC	HVAC System_ Window AC	Energy Consumption (kWh)
0	279	16	20	1.7	19	True	False	False	385
1	277	22	15	1.5	14	False	True	False	425
2	276	14	25	2.2	26	True	False	True	350

In [19]:
```
# creating a variable containing independent variables (Xs)
X_points = df_points.drop("Energy Consumption (kWh)",axis=1)
# creating a variable containing dependent variable (Y)
Y_points = df_points['Energy Consumption (kWh)']
display(X_points.head(),Y_points.head())
```

	Room Area (sq. ft.)	Number of Appliances	Outside Temperature (C)	Insulation Thickness (inches)	Average Temperature in last 24 hours (C)	Building Type_ Residential	HVAC System_ Split AC	HVAC System_ Window AC
0	279	16	20	1.7	19	True	False	False
1	277	22	15	1.5	14	False	True	False
2	276	14	25	2.2	26	True	False	True

```
0    385
1    425
2    350
Name: Energy Consumption (kWh), dtype: int64
```

In [20]:
```
ypred_points = mlr.predict(X_points)
mse_test = mean_squared_error(Y_points, ypred_points)
print("Mean Squared Error (MSE) for the Test Set:", mse_test)
```

```
Mean Squared Error (MSE) for the Test Set: 623.0621796629695
```