In [5]:

```
import pandas as pd
data = pd.read_csv("data.csv")
```

In [7]:

```
print(data.head())
```

```
   VIN (1-10)    County      City State  Postal Code  Model Year     Make  \
0  KM8K33AGXL      King   Seattle    WA      98103.0        2020  HYUNDAI
1  1C4RJYB61N      King   Bothell    WA      98011.0        2022     JEEP
2  1C4RJYD61P    Yakima    Yakima    WA      98908.0        2023     JEEP
3  5YJ3E1EA7J      King  Kirkland    WA      98034.0        2018    TESLA
4  WBY7Z8C5XJ  Thurston   Olympia    WA      98501.0        2018      BMW

            Model                   Electric Vehicle Type  \
0            KONA          Battery Electric Vehicle (BEV)
1  GRAND CHEROKEE  Plug-in Hybrid Electric Vehicle (PHEV)
2  GRAND CHEROKEE  Plug-in Hybrid Electric Vehicle (PHEV)
3         MODEL 3          Battery Electric Vehicle (BEV)
4              I3  Plug-in Hybrid Electric Vehicle (PHEV)

  Clean Alternative Fuel Vehicle (CAFV) Eligibility  Electric Range  \
0           Clean Alternative Fuel Vehicle Eligible             258
1             Not eligible due to low battery range              25
2             Not eligible due to low battery range              25
3           Clean Alternative Fuel Vehicle Eligible             215
4           Clean Alternative Fuel Vehicle Eligible              97

   Base MSRP  Legislative District  DOL Vehicle ID  \
0          0                  43.0       249675142
1          0                   1.0       233928502
2          0                  14.0       229675939
3          0                  45.0       104714466
4          0                  22.0       185498386

                  Vehicle Location  \
0     POINT (-122.34301 47.659185)
1     POINT (-122.20578 47.762405)
2  POINT (-120.6027202 46.5965625)
3     POINT (-122.209285 47.71124)
4     POINT (-122.89692 47.043535)

                                Electric Utility  2020 Census Tract
0   CITY OF SEATTLE - (WA)|CITY OF TACOMA - (WA)       5.303300e+10
1  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10
2                                     PACIFICORP       5.307700e+10
3  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10
4                         PUGET SOUND ENERGY INC       5.306701e+10
```

In [9]:

```
# Objective: Demonstrate Python basics with variables

# 1. Assign a string variable and print it
name = "Electric Vehicle Data Analysis"
print("String:", name)

# 2. Assign an integer variable and print it
total_entries = 150482
print("Integer:", total_entries)

# 3. Assign a float variable and print it
```

```
average_range = 125.5  # Example average range in miles
print("Float:", average_range)

# 4. Use a boolean variable to store True or False values based on a condition
is_large_dataset = total_entries > 100000
print("Boolean:", is_large_dataset)

# 5. Store multiple values in a list variable
vehicle_types = ["BEV", "PHEV"]
print("List:", vehicle_types)

# 6. Use a dictionary to store key-value pairs for more complex data
sample_vehicle = {
    "Make": "Tesla",
    "Model": "Model 3",
    "Electric Range": 215,
    "Type": "BEV"
}
print("Dictionary:", sample_vehicle)

# 7. Assign multiple variables at once for concise code
make, model, range_miles = "Hyundai", "Kona", 258
print("Multiple Variables:", make, model, range_miles)
```

```
String: Electric Vehicle Data Analysis
Integer: 150482
Float: 125.5
Boolean: True
List: ['BEV', 'PHEV']
Dictionary: {'Make': 'Tesla', 'Model': 'Model 3', 'Electric Range': 215, 'Type':
'BEV'}
Multiple Variables: Hyundai Kona 258
```

In [11]:

```
# Function to calculate carbon footprint
def calculate_carbon_footprint(energy_consumption, emission_factor):
    """Calculate carbon footprint given energy consumption (kWh) and emission fa
    return energy_consumption * emission_factor

# Example: Add a column with carbon footprint for an assumed energy consumption
energy_consumption = 1500  # Assume each vehicle consumes 1500 kWh/year
emission_factor = 0.5      # Assume emission factor is 0.5 kg CO₂ per kWh

data['Carbon Footprint (kg CO2)'] = calculate_carbon_footprint(energy_consumptio

# Lambda function to filter cities with carbon footprint below 400 kg CO2
sustainability_threshold = 400
filtered_cities = data[data['Carbon Footprint (kg CO2)'] < sustainability_thresh

print("Cities with Carbon Footprint below 400 kg CO2:")
print(filtered_cities)
```

```
Cities with Carbon Footprint below 400 kg CO2:
[]
```

In [7]:

```
import pandas as pd
ev_data = pd.read_csv("data.csv")
vehicle_makes = pd.Series(ev_data['Make'].unique())
print("Unique Vehicle Makes:")
print(vehicle_makes)
ev_projects_df = ev_data[['Make', 'Model Year', 'Electric Vehicle Type', 'Electr
print("\nSubset of the Electric Vehicle Data:")
print(ev_projects_df)
# Accessing specific columns
print("\nList of Vehicle Makes:")
print(ev_data['Make'].head())

# Filtering vehicles with electric range greater than 200 miles
high_range_vehicles = ev_data[ev_data['Electric Range'] > 200]
print("\nVehicles with Electric Range Greater than 200 Miles:")
print(high_range_vehicles[['Make', 'Electric Range']].head())

# Adding a new column to indicate long-range vehicles
ev_data['Long Range'] = ev_data['Electric Range'] > 150
print("\nDataFrame with Long Range Column:")
print(ev_data[['Make', 'Electric Range', 'Long Range']].head())

# Grouping by Electric Vehicle Type and calculating the average electric range
average_range_by_type = ev_data.groupby('Electric Vehicle Type')['Electric Range
print("\nAverage Electric Range by Vehicle Type:")
print(average_range_by_type)
```

```
Unique Vehicle Makes:
0                  HYUNDAI
1                     JEEP
2                    TESLA
3                      BMW
4                 CHRYSLER
5                     FORD
6                   TOYOTA
7                     AUDI
8                   NISSAN
9                      KIA
10               CHEVROLET
11              VOLKSWAGEN
12                    FIAT
13                    MINI
14                   SMART
15                  RIVIAN
16                   VOLVO
17                 PORSCHE
18                   HONDA
19              MITSUBISHI
20                  SUBARU
21                POLESTAR
22           MERCEDES-BENZ
23                CADILLAC
24                  JAGUAR
25                 LINCOLN
26                 GENESIS
27                   LUCID
28                   LEXUS
29                  FISKER
30                   MAZDA
31              LAND ROVER
32                   TH!NK
33          AZURE DYNAMICS
34              ALFA ROMEO
35    WHEEGO ELECTRIC CARS
36                 BENTLEY
dtype: object

Subset of the Electric Vehicle Data:
       Make  Model Year                   Electric Vehicle Type  \
0   HYUNDAI        2020          Battery Electric Vehicle (BEV)
1      JEEP        2022  Plug-in Hybrid Electric Vehicle (PHEV)
2      JEEP        2023  Plug-in Hybrid Electric Vehicle (PHEV)
3     TESLA        2018          Battery Electric Vehicle (BEV)
4       BMW        2018  Plug-in Hybrid Electric Vehicle (PHEV)
5     TESLA        2020          Battery Electric Vehicle (BEV)
6  CHRYSLER        2017  Plug-in Hybrid Electric Vehicle (PHEV)
7     TESLA        2020          Battery Electric Vehicle (BEV)
8     TESLA        2018          Battery Electric Vehicle (BEV)
9     TESLA        2023          Battery Electric Vehicle (BEV)

   Electric Range  Base MSRP
0             258          0
1              25          0
2              25          0
3             215          0
4              97          0
5             266          0
```

```
6              33          0
7             291          0
8             215          0
9               0          0

List of Vehicle Makes:
0    HYUNDAI
1       JEEP
2       JEEP
3      TESLA
4        BMW
Name: Make, dtype: object

Vehicles with Electric Range Greater than 200 Miles:
      Make  Electric Range
0  HYUNDAI              258
3    TESLA              215
5    TESLA              266
7    TESLA              291
8    TESLA              215

DataFrame with Long Range Column:
      Make  Electric Range  Long Range
0  HYUNDAI              258        True
1     JEEP               25       False
2     JEEP               25       False
3    TESLA              215        True
4      BMW               97       False

Average Electric Range by Vehicle Type:
Electric Vehicle Type
Battery Electric Vehicle (BEV)            78.608902
Plug-in Hybrid Electric Vehicle (PHEV)    30.655471
Name: Electric Range, dtype: float64
```

In [9]:

```
import numpy as np
import pandas as pd

# Load the CSV file into a DataFrame
ev_data = pd.read_csv("data.csv")

# Extract the 'Electric Range' column as a NumPy array
electric_range = ev_data['Electric Range'].to_numpy()

# 1. Calculate the sum of the Electric Range
total_range = np.sum(electric_range)
print("Total Electric Range (in miles):", total_range)

# 2. Calculate the mean Electric Range
mean_range = np.mean(electric_range)
print("Mean Electric Range (in miles):", mean_range)

# 3. Calculate the standard deviation of Electric Range
std_range = np.std(electric_range)
print("Standard Deviation of Electric Range:", std_range)

# 4. Reshape the array into a 2D array with 10 rows for further analysis
reshaped_range = electric_range[:50].reshape(10, 5)  # Taking the first 50 value
print("\nReshaped Array (10x5):")
print(reshaped_range)
```

```
Total Electric Range (in miles): 10214393
Mean Electric Range (in miles): 67.87783921000518
Standard Deviation of Electric Range: 96.22968895839267

Reshaped Array (10x5):
[[258  25  25 215  97]
 [266  33 291 215   0]
 [215  19  25 220 220]
 [ 19 204  84  26  84]
 [ 13 249  38 103  25]
 [322 215  19 125 330]
 [ 84 222 239  84   0]
 [ 73  84 259   0   0]
 [ 75  20 150  16  32]
 [220  76  33  32  21]]
```

Code for Handling Missing Values:

In [11]:

```python
import pandas as pd

# Load the dataset
ev_data = pd.read_csv("data.csv")

# 1. Check for missing values
print("Missing Values Summary:")
print(ev_data.isnull().sum())

# 2. Remove rows with missing values
ev_data_cleaned = ev_data.dropna()
print("\nData after Removing Rows with Missing Values:")
print(ev_data_cleaned.head())

# 3. Impute missing values using the mean for the 'Electric Range' column
ev_data['Electric Range'] = ev_data['Electric Range'].fillna(ev_data['Electric R
print("\nData after Imputing Missing Values in 'Electric Range':")
print(ev_data['Electric Range'].head())

# 4. Forward fill missing values
ev_data_ffill = ev_data.fillna(method='ffill')
print("\nData after Forward Fill:")
print(ev_data_ffill.head())

# 5. Flag missing data by creating an indicator column
ev_data['Range Missing'] = ev_data['Electric Range'].isnull().astype(int)
print("\nData with 'Range Missing' Column:")
print(ev_data[['Electric Range', 'Range Missing']].head())
```

```
Missing Values Summary:
VIN (1-10)                                             0
County                                                 3
City                                                   3
State                                                  0
Postal Code                                            3
Model Year                                             0
Make                                                   0
Model                                                  0
Electric Vehicle Type                                  0
Clean Alternative Fuel Vehicle (CAFV) Eligibility      0
Electric Range                                         0
Base MSRP                                              0
Legislative District                                 341
DOL Vehicle ID                                         0
Vehicle Location                                       7
Electric Utility                                       3
2020 Census Tract                                      3
dtype: int64

Data after Removing Rows with Missing Values:
   VIN (1-10)    County      City State  Postal Code  Model Year     Make  \
0  KM8K33AGXL      King   Seattle    WA      98103.0        2020  HYUNDAI   
1  1C4RJYB61N      King   Bothell    WA      98011.0        2022     JEEP   
2  1C4RJYD61P    Yakima    Yakima    WA      98908.0        2023     JEEP   
3  5YJ3E1EA7J      King  Kirkland    WA      98034.0        2018    TESLA   
4  WBY7Z8C5XJ  Thurston   Olympia    WA      98501.0        2018      BMW   

            Model                   Electric Vehicle Type  \
0            KONA          Battery Electric Vehicle (BEV)   
1  GRAND CHEROKEE  Plug-in Hybrid Electric Vehicle (PHEV)   
2  GRAND CHEROKEE  Plug-in Hybrid Electric Vehicle (PHEV)   
3         MODEL 3          Battery Electric Vehicle (BEV)   
4              I3  Plug-in Hybrid Electric Vehicle (PHEV)   

  Clean Alternative Fuel Vehicle (CAFV) Eligibility  Electric Range  \
0           Clean Alternative Fuel Vehicle Eligible             258   
1             Not eligible due to low battery range              25   
2             Not eligible due to low battery range              25   
3           Clean Alternative Fuel Vehicle Eligible             215   
4           Clean Alternative Fuel Vehicle Eligible              97   

   Base MSRP  Legislative District  DOL Vehicle ID  \
0          0                  43.0       249675142   
1          0                   1.0       233928502   
2          0                  14.0       229675939   
3          0                  45.0       104714466   
4          0                  22.0       185498386   

                  Vehicle Location  \
0     POINT (-122.34301 47.659185)   
1     POINT (-122.20578 47.762405)   
2  POINT (-120.6027202 46.5965625)   
3     POINT (-122.209285 47.71124)   
4     POINT (-122.89692 47.043535)   

                               Electric Utility  2020 Census Tract  
0   CITY OF SEATTLE - (WA)|CITY OF TACOMA - (WA)       5.303300e+10  
1  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10  
2                                    PACIFICORP       5.307700e+10  
```

```
3  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10
4                         PUGET SOUND ENERGY INC       5.306701e+10

Data after Imputing Missing Values in 'Electric Range':
0    258
1     25
2     25
3    215
4     97
Name: Electric Range, dtype: int64
```

```
C:\Users\ACER\AppData\Local\Temp\ipykernel_6408\1653729744.py:21: FutureWarning:
DataFrame.fillna with 'method' is deprecated and will raise in a future version.
Use obj.ffill() or obj.bfill() instead.
  ev_data_ffill = ev_data.fillna(method='ffill')
```

```
Data after Forward Fill:
   VIN (1-10)    County      City State  Postal Code  Model Year     Make  \
0  KM8K33AGXL      King   Seattle    WA      98103.0        2020  HYUNDAI   
1  1C4RJYB61N      King   Bothell    WA      98011.0        2022     JEEP   
2  1C4RJYD61P    Yakima    Yakima    WA      98908.0        2023     JEEP   
3  5YJ3E1EA7J      King  Kirkland    WA      98034.0        2018    TESLA   
4  WBY7Z8C5XJ  Thurston   Olympia    WA      98501.0        2018      BMW   

            Model                   Electric Vehicle Type  \
0            KONA          Battery Electric Vehicle (BEV)   
1  GRAND CHEROKEE  Plug-in Hybrid Electric Vehicle (PHEV)   
2  GRAND CHEROKEE  Plug-in Hybrid Electric Vehicle (PHEV)   
3         MODEL 3          Battery Electric Vehicle (BEV)   
4              I3  Plug-in Hybrid Electric Vehicle (PHEV)   

  Clean Alternative Fuel Vehicle (CAFV) Eligibility  Electric Range  \
0           Clean Alternative Fuel Vehicle Eligible             258   
1             Not eligible due to low battery range              25   
2             Not eligible due to low battery range              25   
3           Clean Alternative Fuel Vehicle Eligible             215   
4           Clean Alternative Fuel Vehicle Eligible              97   

   Base MSRP  Legislative District  DOL Vehicle ID  \
0          0                  43.0       249675142   
1          0                   1.0       233928502   
2          0                  14.0       229675939   
3          0                  45.0       104714466   
4          0                  22.0       185498386   

                  Vehicle Location  \
0     POINT (-122.34301 47.659185)   
1     POINT (-122.20578 47.762405)   
2  POINT (-120.6027202 46.5965625)   
3     POINT (-122.209285 47.71124)   
4     POINT (-122.89692 47.043535)   

                               Electric Utility  2020 Census Tract  
0   CITY OF SEATTLE - (WA)|CITY OF TACOMA - (WA)       5.303300e+10  
1  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10  
2                                     PACIFICORP       5.307700e+10  
3  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10  
4                         PUGET SOUND ENERGY INC       5.306701e+10  

Data with 'Range Missing' Column:
   Electric Range  Range Missing
0             258              0
1              25              0
2              25              0
3             215              0
4              97              0
```

Code for Data Preprocessing:

In [13]:

```
from sklearn.preprocessing import MinMaxScaler, StandardScaler
import pandas as pd

# 1. Normalization using Min-Max Scaling for 'Electric Range' and 'Base MSRP'
scaler = MinMaxScaler()
```

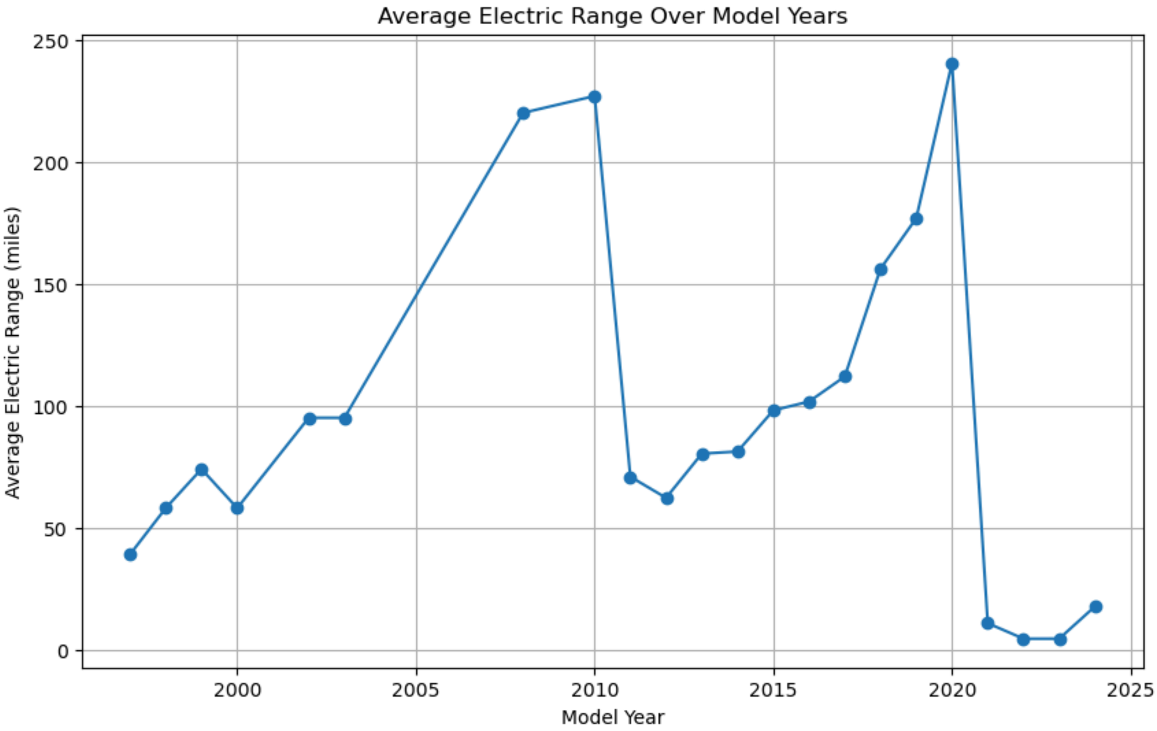
```
ev_data[['Electric Range', 'Base MSRP']] = scaler.fit_transform(ev_data[['Electr
print("\nNormalized Data:")
print(ev_data[['Electric Range', 'Base MSRP']].head())

# 2. Encoding categorical variables using one-hot encoding for 'Electric Vehicle
ev_data_encoded = pd.get_dummies(ev_data, columns=['Electric Vehicle Type'])
print("\nData with One-Hot Encoding:")
print(ev_data_encoded.head())

# 3. Feature Engineering: Creating a new column 'Long Range'
ev_data['Long Range'] = ev_data['Electric Range'] > 0.75  # Assuming normalized
print("\nData with 'Long Range' Feature:")
print(ev_data[['Electric Range', 'Long Range']].head())
```

```
Normalized Data:
   Electric Range  Base MSRP
0        0.765579        0.0
1        0.074184        0.0
2        0.074184        0.0
3        0.637982        0.0
4        0.287834        0.0

Data with One-Hot Encoding:
   VIN (1-10)    County      City State  Postal Code  Model Year     Make  \
0  KM8K33AGXL      King   Seattle    WA      98103.0        2020  HYUNDAI
1  1C4RJYB61N      King   Bothell    WA      98011.0        2022     JEEP
2  1C4RJYD61P    Yakima    Yakima    WA      98908.0        2023     JEEP
3  5YJ3E1EA7J      King  Kirkland    WA      98034.0        2018    TESLA
4  WBY7Z8C5XJ  Thurston   Olympia    WA      98501.0        2018      BMW

            Model Clean Alternative Fuel Vehicle (CAFV) Eligibility  \
0            KONA           Clean Alternative Fuel Vehicle Eligible
1  GRAND CHEROKEE             Not eligible due to low battery range
2  GRAND CHEROKEE             Not eligible due to low battery range
3         MODEL 3           Clean Alternative Fuel Vehicle Eligible
4              I3           Clean Alternative Fuel Vehicle Eligible

   Electric Range  Base MSRP  Legislative District  DOL Vehicle ID  \
0        0.765579        0.0                  43.0       249675142
1        0.074184        0.0                   1.0       233928502
2        0.074184        0.0                  14.0       229675939
3        0.637982        0.0                  45.0       104714466
4        0.287834        0.0                  22.0       185498386

                  Vehicle Location  \
0     POINT (-122.34301 47.659185)
1     POINT (-122.20578 47.762405)
2  POINT (-120.6027202 46.5965625)
3     POINT (-122.209285 47.71124)
4     POINT (-122.89692 47.043535)

                               Electric Utility  2020 Census Tract  \
0   CITY OF SEATTLE - (WA)|CITY OF TACOMA - (WA)       5.303300e+10
1  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10
2                                     PACIFICORP       5.307700e+10
3  PUGET SOUND ENERGY INC||CITY OF TACOMA - (WA)       5.303302e+10
4                         PUGET SOUND ENERGY INC       5.306701e+10

   Range Missing  Electric Vehicle Type_Battery Electric Vehicle (BEV)  \
0              0                                               True
1              0                                              False
2              0                                              False
3              0                                               True
4              0                                              False

   Electric Vehicle Type_Plug-in Hybrid Electric Vehicle (PHEV)
0                                              False
1                                               True
2                                               True
3                                              False
4                                               True

Data with 'Long Range' Feature:
   Electric Range  Long Range
```

```
0        0.765579        True
1        0.074184       False
2        0.074184       False
3        0.637982       False
4        0.287834       False
```

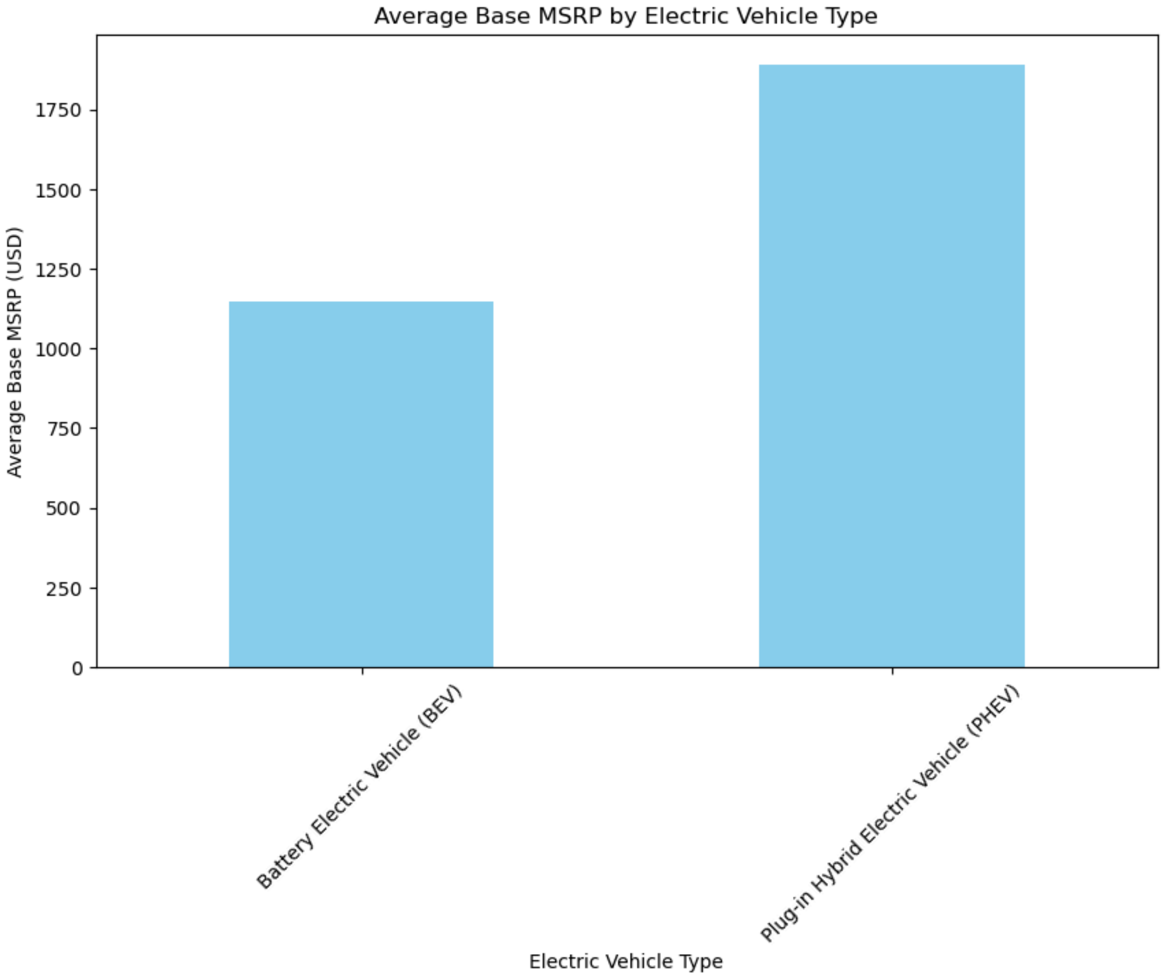
In [15]:

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the data
ev_data = pd.read_csv("data.csv")

# Group data by Model Year and calculate the average Electric Range for each yea
avg_range_per_year = ev_data.groupby('Model Year')['Electric Range'].mean()

# Plotting the line graph
plt.figure(figsize=(10, 6))
plt.plot(avg_range_per_year.index, avg_range_per_year.values, marker='o', linest
plt.title("Average Electric Range Over Model Years")
plt.xlabel("Model Year")
plt.ylabel("Average Electric Range (miles)")
plt.grid(True)
plt.show()
```



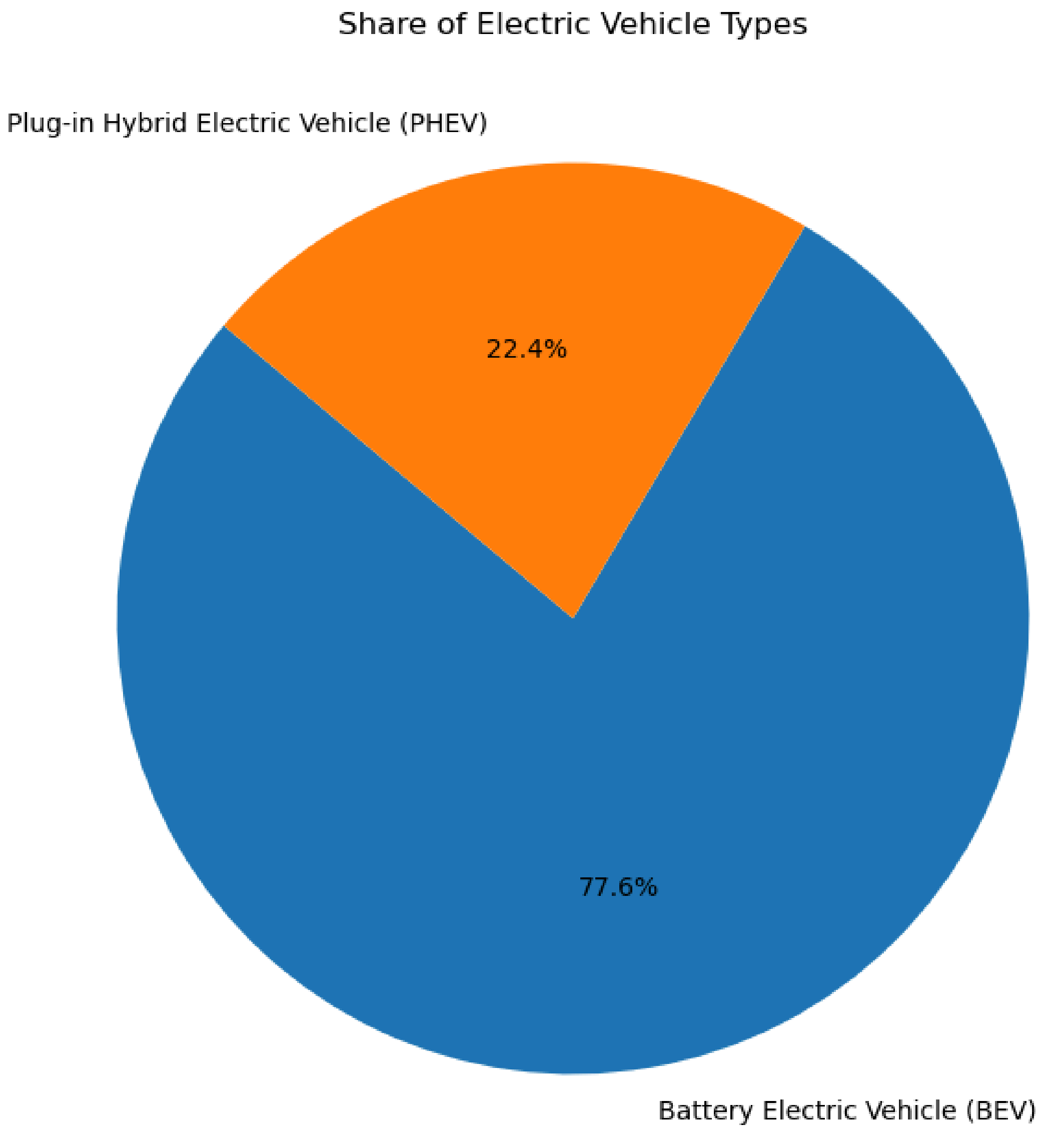
In [17]:

```python
# Group by Electric Vehicle Type and calculate the average Base MSRP
avg_msrp_per_type = ev_data.groupby('Electric Vehicle Type')['Base MSRP'].mean()

# Plotting the bar chart
plt.figure(figsize=(10, 6))
avg_msrp_per_type.plot(kind='bar', color='skyblue')
plt.title("Average Base MSRP by Electric Vehicle Type")
plt.xlabel("Electric Vehicle Type")
plt.ylabel("Average Base MSRP (USD)")
plt.xticks(rotation=45)
plt.show()
```



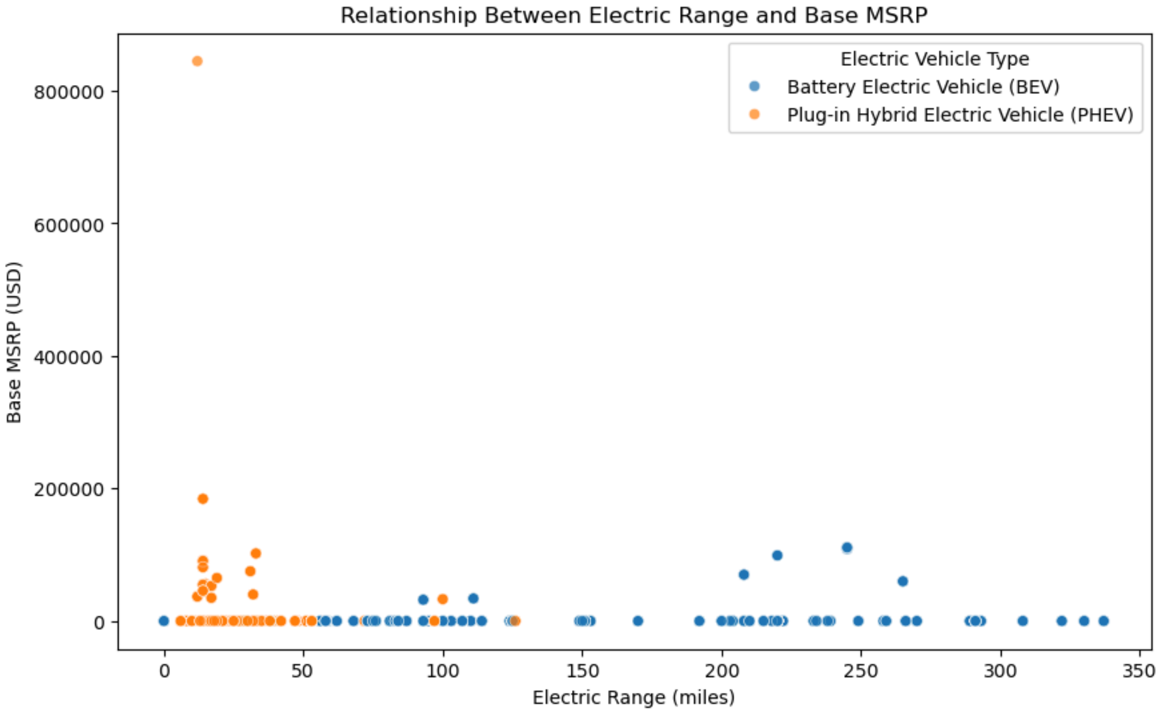
In [19]:

```
# Count the number of vehicles by type
vehicle_type_counts = ev_data['Electric Vehicle Type'].value_counts()

# Plotting the pie chart
plt.figure(figsize=(8, 8))
vehicle_type_counts.plot(kind='pie', autopct='%1.1f%%', startangle=140)
plt.title("Share of Electric Vehicle Types")
plt.ylabel("")  # Remove the default y-axis label
plt.show()
```



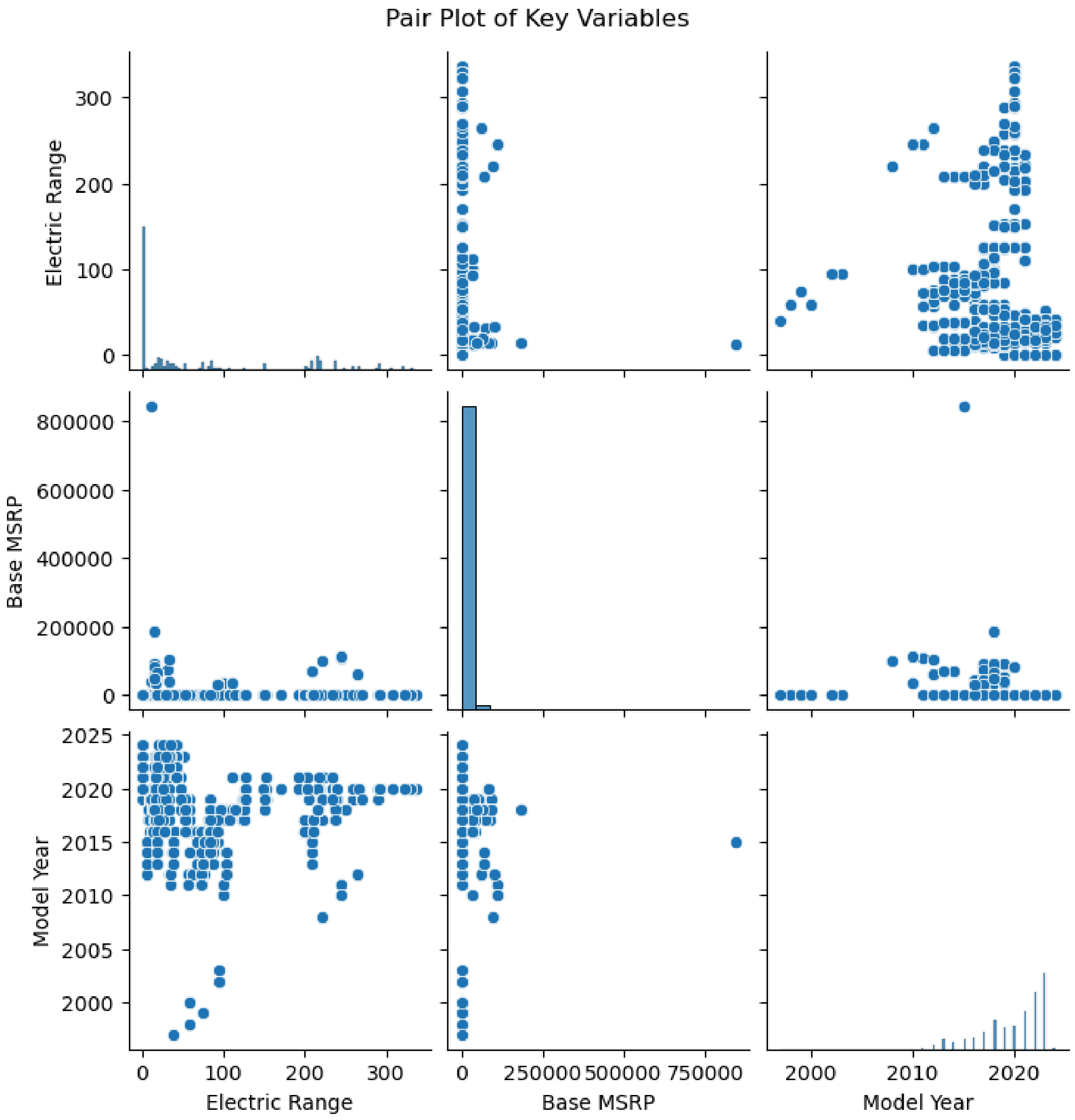
In [21]:

```
import seaborn as sns

# Scatter plot using Seaborn
plt.figure(figsize=(10, 6))
sns.scatterplot(data=ev_data, x='Electric Range', y='Base MSRP', hue='Electric V
plt.title("Relationship Between Electric Range and Base MSRP")
plt.xlabel("Electric Range (miles)")
plt.ylabel("Base MSRP (USD)")
plt.legend(title='Electric Vehicle Type')
plt.show()
```

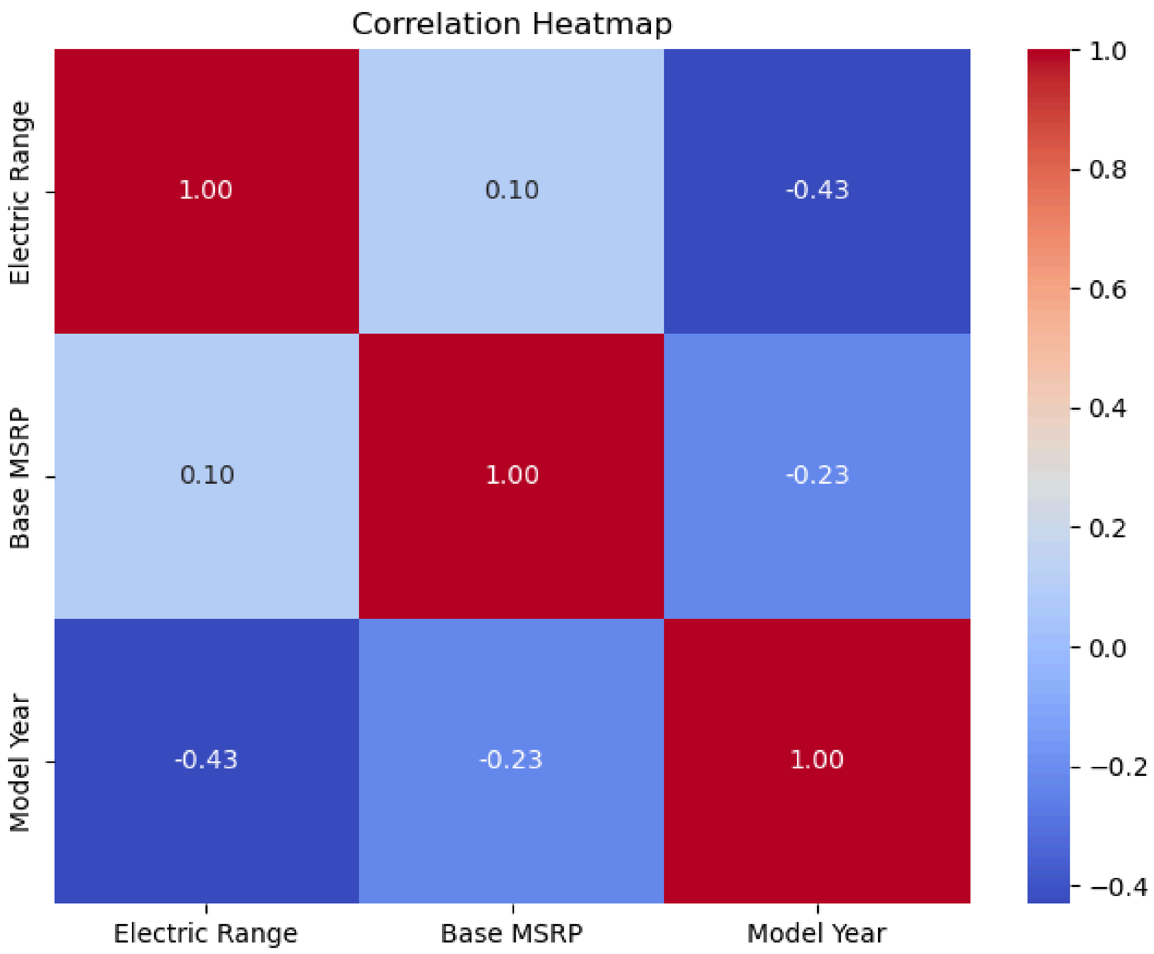


In [30]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the data
ev_data = pd.read_csv("data.csv")

# Pair plot for Electric Range, Base MSRP, and Model Year
sns.pairplot(ev_data[['Electric Range', 'Base MSRP', 'Model Year']].dropna())
plt.suptitle("Pair Plot of Key Variables", y=1.02)
plt.show()
```

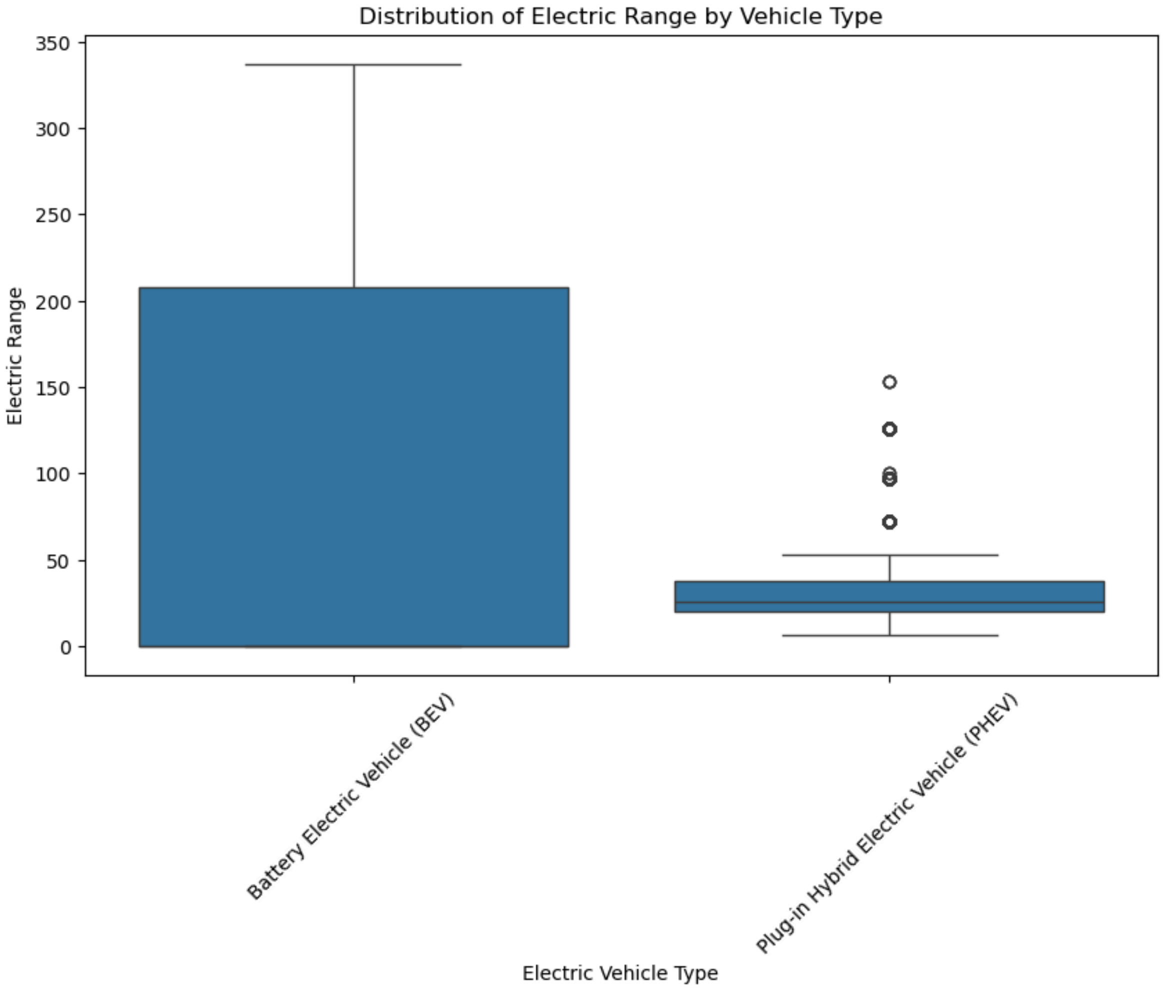


```
In [32]: corr_matrix = ev_data[['Electric Range', 'Base MSRP', 'Model Year']].corr()

         # Plotting the heatmap
         plt.figure(figsize=(8, 6))
         sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', fmt='.2f')
         plt.title("Correlation Heatmap")
         plt.show()
```



In [34]:

```
plt.figure(figsize=(10, 6))
sns.boxplot(data=ev_data, x='Electric Vehicle Type', y='Electric Range')
plt.title("Distribution of Electric Range by Vehicle Type")
plt.xticks(rotation=45)
plt.show()
```


Distribution of Electric Range by Vehicle Type
Electric Range
350
300
250
200
150
100
50
0
Battery Electric Vehicle (BEV)
Plug-in Hybrid Electric Vehicle (PHEV)
Electric Vehicle Type

In [ ]: