

In [32]:

Task-4

```
### Build an EV recommendation class which allow users to input their budget, desired range, and battery capacity.
### The class should then return the top three EVs matching their criteria.
```

```
import pandas as pd
```

```
#Class is defined here
```

```
class EVFinder:
```

```
# This is the constructor method – it runs when you create an object from the class.
```

```
def __init__(self, dataframe):
    self.df=dataframe
```

This is a method(a function inside class) and it is taking three things from user mentioned in that.

```
def find_best_evs(self, budget, desired_range, min_battery_capacity):
```

#This line filters the EVs that meet all three conditions and creates a new filter table

```
filtered=self.df[
    (self.df["Minimal price (gross) [PLN]" ]<=budget) &
    (self.df["Range (WLTP) [km]" ]>=desired_range) &
    (self.df["Battery capacity [kWh]" ]>=min_battery_capacity)
```

]

```
# This sorts the filtered EV by range(biggest range will come first) first then by battery in descending order
```

```
sorted_evs=filtered.sort_values(  
    by=["Range (WLTP) [km]", "Battery capacity [kWh]"],  
    ascending=[False, False]
```

)

```
# This returns top 3 EVs with selected columns
```

```
return sorted_evs[["Car full name",
                  "Minimal price (gross) [PLN]",
```

```
        "Range (WLTP) [km]",
        "Battery capacity [kWh]"
    ]].head(3)
```

```
In [33]: # This is where we will load our dataset
df=pd.read_excel("FEV-data-Excel.xlsx")

try:
    # We have defined input values here.
    budget = int(input("Enter your maximum budget (PLN): "))
    desired_range = int(input("Enter your minimum range (km): "))
    min_battery_capacity = float(input("Enter your minimum battery capacity (kWh): "))

    # We will create the object of EVFinder here
    finder = EVFinder(df)

    #This calls the method defined inside the class, filters and sorts the cars, and returns the top 3 as a table.
    top_evs = finder.find_best_evs(budget, desired_range, min_battery_capacity)

    # Reset index and display as a clean 3x4 table
    top_evs.reset_index(drop=True, inplace=True)

    # Display the table
    print("Top 3 EVs:")
    display(top_evs)

except ValueError:
    print("Please enter valid numeric inputs.")

# NOTE- The table displayed below based on assumption mentioned below for reference and can be changed by altering values.
# Enter your maximum budget (PLN): 250000
# Enter your minimum range (km): 300
# Enter your minimum battery capacity (kWh): 50
```

Top 3 EVs:

	Car full name	Minimal price (gross) [PLN]	Range (WLTP) [km]	Battery capacity [kWh]
0	Tesla Model 3 Long Range	235490	580	75.0
1	Volkswagen ID.3 Pro S	179990	549	77.0
2	Volkswagen ID.4 1st	202390	500	77.0

In []: