In this experiment, we used the topological ordering algorithm to generate all possible linear sequences (total orderings) from a set of incomplete preferences on a group of electric cars. The aim was to simulate the decision-making process of a Decision Maker (DM) with incomplete preferences on available cars and to observe how these preferences result in different sortings. The set of alternatives considered includes six electric car models:

- **Tesla Model 3**

**- Polestar 2**

- **Nissan Leaf**

- **Ford Mustang Mach-E**

- **Audi e-tron**

- **Volkswagen ID.4**

- **Hyundai Kona Electric**

Incomplete preferences are represented as a set of oriented arcs in a graph, where a direct arc between two models indicates that the DM prefers one car over the other. The preference relations used to construct the graph are as follows:

* **Tesla Model 3** is preferred to **Nissan Leaf** and **Ford Mustang Mach-E**.
* **Polestar 2** is preferred to **Audi e-tron** and **Volkswagen ID.4.**
* **Audi e-tron** is preferred to **Nissan Leaf**.
* **Ford Mustang Mach-E** is preferred to **Hyundai Kona Electric**.
* **Tesla Model 3** is preferred to **Hyundai Kona Electric**.

These relationships do not cover all possible pairs of cars, so the DM provided an incomplete set of preferences. This implies that there are several possible total sortings, and the topological sorting algorithm was used to generate all sequences that respect the imposed preferences. Some of the reasons for the preferences chosen are:

* **Tesla Model 3** is preferred over both **Nissan Leaf** and **Ford Mustang Mach-E** because of its superior autonomy compared to most electric vehicles on the market and for the advanced autonomous driving technology and software system, considered an industry leader.
* **Polestar 2** is preferred over the **Audi e-tron** because it offers a more modern design, more engaging driving dynamics and an integrated Android infotainment system, while the **Audi e-tron** is more luxury-oriented. At the same time, I prefer the **Polestar 2** to the **Volkswagen ID.4**, a more practical but less powerful SUV.
* The **Audi e-tron** is preferred over the **Nissan Leaf** because it has more luxury and performance than the more affordable **Nissan Leaf**, making it a better choice for those seeking premium quality.
* **The Ford Mustang Mach-E** is preferred over the **Hyundai Kona Electric** because of its superior performance and the fact that it has a sporty design that distinguishes it from more compact cars like the **Kona**.
* **Volkswagen ID.4** is preferred over the **Hyundai Kona Electric** because it offers a more spacious and versatile driving experience, making it a better choice for families or those looking for a compact SUV with a little more space than the **Kona**.

Using Python code to perform topological sorting, we generated all possible linear orders that respect the given preferences as much as possible. Each topological sort represents a valid ranking of the alternatives where no car is ranked higher than another if it violates the specified preferences. Here are some example linear orders generated:

1. ['Tesla Model 3', 'Polestar 2', 'Audi e-tron', 'Nissan Leaf', 'Ford Mustang Mach-E', 'Volkswagen ID.4', 'Hyundai Kona Electric']
2. ['Tesla Model 3', 'Polestar 2', 'Volkswagen ID.4', 'Audi e-tron', 'Ford Mustang Mach-E', 'Nissan Leaf', 'Hyundai Kona Electric']
3. ['Polestar 2', 'Tesla Model 3', 'Audi e-tron', 'Nissan Leaf', 'Ford Mustang Mach-E', 'Volkswagen ID.4', 'Hyundai Kona Electric']
4. ['Tesla Model 3', 'Ford Mustang Mach-E', 'Polestar 2', 'Audi e-tron', 'Volkswagen ID.4', 'Nissan Leaf', 'Hyundai Kona Electric']

In total, 63 orders were generated. However, there are orders in which certain pairs of cars are arranged differently than the preferences explicitly defined by the DM. These changes in relative priorities between the models are called ‘jumps’. A jump occurs when an ordering relates two in a way not specified by the initial preferences. In this specific case are (in brackets, you can see in which number of the sorting output the following jumps were found):

* **Ford Mustang Mach-E** positioned before **Audi e-tron**.
* **Volkswagen ID.4** positioned before **Audi e-tron**.
* **Audi e-tron** positioned before **Volkswagen ID.4**.
* **Ford Mustang Mach-E** positioned before **Polestar 2**.
* **Volkswagen ID.4** positioned before **Audi e-tron**.
* **Tesla Model 3** positioned before **Polestar 2**.
* **Audi e-tron** positioned before **Tesla Model 3**.
* **Volkswagen ID.4** positioned before **Audi e-tron**.
* **Audi e-tron** positioned before **Volkswagen ID.4**.

In scenarios like this, it is important to consider how much the ‘jumps’ influence the final decision. If jumps concern alternatives for which one has weaker or insignificant preferences, the impact on the outcome might be limited. Conversely, if they violate strong preferences (such as the preference between Tesla Model 3 and Polestar 2), they may require further analysis. In this specific case, clear preferences placed the Tesla Model 3 at the top and the Polestar 2 preferred over the Audi e-tron. These preferences were respected in all generated orders, demonstrating that the algorithm succeeds in preserving the main relationships indicated by the Decision Maker. However, outside of these strong preferences, the relationships between the other cars (such as Nissan Leaf, Ford Mustang Mach-E, Volkswagen ID.4, and Hyundai Kona Electric) were less well defined, leading to a significant variety of placements among them.

In conclusion, the application of the topological sorting algorithm to the case of the seven electric cars demonstrated it’s ability to handle incomplete preferences and generate a set of valid sortings. However, the results highlight the importance of providing more detailed preferences to achieve a more stable sorting. In the decision-making context, the algorithm is particularly useful for exploring different scenarios, but in order to arrive at a final choice, it would be useful to reinforce existing preferences or provide further guidance.