A car dealership would like to be able to identify customers that are more likely to purchase certain types of cars.

The dealership inventory system stores the following information about cars:

* Make
* Model
* Year
* Color
* Price (in USD)
* Size (see below)

E.g.: 2010 blue Volvo sedan costing $21,100

The inventory system also tracks the space available in the lot due to limitations in available space; cars are classified as being:

* Compact: small space
* Mid-sized: medium space
* SUV/Minivan/Truck/etc.: large space

E.g., 24 compact spaces remaining, 10 mid-size spaces, 3 large spaces

The warehouse of the dealership also stores information related to the types of cars purchased by each customer (or those that have expressed interest in a given model):

* Age
* Income
* Gender
* Purchase/Interest (car)

E.g.: 20-30 year old males with >$90K income prefer convertibles (correlated through interest and purchases). These rules are updated daily to ensure accurate information about potential customers.

Design and implement the following:

1. Write a function that will take as input a customer profile and, based on the internal rules stored in the warehouse related to car correlation, return an ordered list (by price) of cars that the customer is most likely to purchase.
2. Write a function that updates the dealerships available inventory. It should take as input a list of cars to be added and/or removed from the system, representing a bulk order to be processed by the dealership. The method should scan all incoming cars and ensure that there is space available. If not it should return an error. If it can store them the cars should then be added to the inventory.