We created an online app, which will be used a company that sells vehicles (cars, trucks and electric cars). First we started by creating a main class of Vehicle and from there we created subclasses Car, Truck and Electric Car and all this these subclasses inherited from the main class(vehicle), we did this to show the concept of inheritance. The online app has methods, Browse vehicles. Browse Vehicles method has Browse cars, this method allows the user or a customer to see all the cars that the car company has in market at that time. The online app has also method of Browse trucks which did the same task of allowing the user to see the available trucks in the company and lastly there is also method of browsing electric cars which also allowed the user to see the available electric cars in the company. The online app has a class Motor­­Store which had methods that allowed the user to use the app efficiently . There is the method of purchasing vehicles(cars, trucks, electric cars). The method of purchasing a car allows the customer to buy the car online so that the car may be delivered to them if it’s at their interest, the app also have the method of purchasing a truck. The method also allows the user to buy a truck online and lastly there is method of purchasing electric car which like all purchase methods allowed the user to buy an electric car online. We also have a method of pawning a vehicle(car, truck, electric car). The method of pawning a truck allows a user to sell their truck to the company, the method will take that truck and add it to the available trucks in the company, similarly the method of pawn car allowed the user to sell their car to the company and that car will be added to the available cars in the company and lastly the method of pawn electric car allowed the user to sell their electric car to the company and that car will be added to the existing electric cars in the company. We sorted trucks available in store using insertion and selection sort. We sorted the trucks by comparing the prices and displaying the truck with the lowest price, we did this using insertion and selection sort. We searched for a certain truck in the company using linear search, the search looked for a specific truck in the available trucks( target) and it returned it if it matched the target truck else it returned -1 (if the target truck was not found). In our online app we used the scanner to read user input so we had a class that handled exceptions (InvalidPriceExceptions), this class is used to ensure that the customer does not enter a price less than zero. The online app has a class for user interaction(main class), this class allowed the user to use the online app as to how they wished to. It showed them the menu that the company had so it is where user input was asked so they can be able to use the app.