# PRE-SELECTION FOR TEST AUTOMATION MENTORING PARTICIPATION: JAVA

# Task 1

Design object model for given domain.

Please demonstrate usage of:

* Classes (abstract classes - if applicable)
* Interfaces
* Inheritance
* Polymorphism
* Encapsulation
* Collections framework.

Each class, method and variable should have meaningful name and purpose. You need to determine which classes are required to solve the task. Inheritance should be used only when it makes sense. Classes should be grouped into corresponding packages. Work with console should be minimized (only required data for input, output correspond the info in task). It is expected that you will choose certain domain from the list below, define classes hierarchy for that domain and implement it using OOP (inheritance or implementing interfaces). Each class should have methods and fields defined by you. Your program should create objects of different types in chosen domain and combine them into collection (use collections framework). Next, will be required to process data of that collection by certain rules.

**Domain options (please choose and implement one of them).**

1. **Flower shop.** Define flowers hierarchy. Create several flower-objects. Create a bouquet and determine its price.
2. **Candy gift basket.** Define hierarchy of candies. Create several candy-objects. Collect basket of the candies and determine its weight. Sort the candies by predefined criteria (define criteria by yourself). Find the candy which will match predefined criteria (define criteria by yourself).
3. **Domestic electrical appliances.** Define hierarchy of electrical appliances. “Plug in” some of them. Calculate power consumption. Sort electrical appliances by predefined criteria (choose criteria by yourself). Find appliance which will match predefined criteria (choose criteria by yourself).
4. **Chef.** Define hierarchy of vegetables. Make a salad. Count calories. Sort vegetables in salad using some criteria (choose yourself). Find vegetables in salad which will match predefined criteria (define criteria by yourself).
5. **Port.** Define ships hierarchy. Create port. Calculate total load capacity. Sort ports’ ships by predefined criteria (define criteria by yourself). Find a ship that will match predefined criteria (you could define one or several criteria).
6. **Taxi fleet.** Define hierarchy of passenger cars. Create taxi fleet. Calculate total cost of taxi fleet. Sort cars by fuel consumption. Find the car which will match predefined criteria (define criteria by yourself).
7. **Game room.** Create game room for children of different age. Room should have fixed number of toys within the allocated budget. Rooms should have toys of similar type but designed for children of different age (e.g. small, medium and large cars, balls, dolls). Sort toys in the room using one of the criteria. Find toys which will match predefined criteria (define criteria by yourself).
8. **Holiday tours**. Create a set of traveling offers to the client by different types (traveling, excursions, shopping, cruise etc.). Please take into consideration variations of transport, meals, number of days. Implement possibility of choosing and sorting of offers.

# Task 2

For the object model implemented in task 1, it is needed to design custom exception classes and implement handling of possible exceptional situations. For example, there is no element that meets required criteria in the collection, impossible to delete an element etc. Leave your explanation in commented blocks of your exception class. Create at least 3 custom exceptions and use 5 built-in exceptions minimum.