**Core Java – Collections**

Q1. Write a java program as per the below given specification:

1. Design a Dish class having dishID, dishName, creationTime as the attributes.

To initialise the attributes of the class create a parametrizable constructor. Also override the toString() method to print the object details on the console.

1. Design a Menu class having menuList as an attribute which is a static ArrayList.
2. Load the menu list with Dish object comprising some sample dishes using static block.

Design a Restaurant class comprising main method.

It should give a menu-driven interface to a user as follows:

Press 1 to display Dishes.

Press 2 to search Dish.

iv) On the click of 1 all the dishes offered by the restaurant should be displayed to a user.

1. On the click of 2 it prompt the user to enter the dishID, if the ID enetered matches with the dish, its details should be printed on the console.
2. If dishId not present then throw suitable custom Exception.

Q2. Write a java program to maintain a Phone book using Map.

Program should give a menu driven interface as shown below:

Press 1 to Add new phone book entry

Press 2 to Search a Phone Number

Press 3 to Quit.

On the click of 1 a user is prompted to enter his name and phone number.

The details entered by the user should be maintained in a Map phonebook, where the name is the key and the phone number is the value.

On the click of 2 a user is prompted to enter the name of the user whose number need to be searched.

Once the name is entered correctly his number is displayed on the console. Throw relevant exception if name is not in the map.

On the click of 3 program terminates.

*Note : Use HashMap to store phone book entries.*

*Q3)* Implement the Question 1 using Java 8 Stream API

Q4) Write a java program as per the below given specification:

1. Declare a class Employee having properties namely
   1. name (type String) dept (type String ) and salary (type double)
   2. declare a constructor to initialize the properties and generate setter and getters for all the properties.
2. Declare a class StreamFilterTest having main() method declare an object of ArrayList and store four objects of Employee class.
3. After storing Employee object convert the ArrayList object into Stream object and perform following operations
   1. Display total number of Employee inside each department.
   2. Display total salary of all the Employee.
   3. Display Employee information whose salary is greater than 20000.
   4. Display Employee information whose salary is less than 20000.
   5. Display information of those Employee whose name begins with any user input token.
   6. Display Employee information which name contains more than five characters.
   7. Display only the name of the Employee after transforming the name into uppercase.

Note : Apply concept of stream and stream related operations like map and filter