Assignment 2

Inheritance, Package and Interface

1. Each customer of a bank has customer id, name, and current loan amount and phone number. One can change the attributes like name, phone number. A customer may ask for loan of certain amount. It is granted provided the sum of current loan amount and asked amount does not exceed credit limit (fixed amount for all customer). A customer may be a privileged account holder. For such customers, credit limit is higher. Once a loan is sanctioned, necessary updates should be made. Any type of customer should be able to find his credit limit, current loan amount and amount of loan he can seek.

Design and implement the classes.

2. For every person in an institute details like name, address (consists of premises number, street, city, pin and state), phone number, e-mail id are maintained. A person is either a student or a faculty or an office-staff. For student, roll number and course of study are also be maintained. For faculty, employee id, department and specialisation are to be stored. One should be able to view the object details and set the attributes. For address and phone number, one may change it partially depending on the choice. Design and implement the classes.

3. For a library management system design BookList, MemberList and Transaction packages. Booklist package will have the support to store book information in the list like book id, title, total number of copies purchased, and number of copies currently available. One can add book in list (verifying uniqueness of book id), change the attribute values (particularly, increase/decrease copies purchased, available as and when required), display particular book information (for a book id) and also total list. MemberList package will provide the service for maintaining member information. Member information includes member id (unique), name, date of birth and number of books currently issued to him. There is a limit on number of books one can have at a point of time (it is same for all members). Transaction package maintains a list of transaction. A transaction entry in the list keeps member id, book id of the book being issued. Supports are to be provided to modify the entries. An entry with member id 'xxxx' can be used for adding a new entry.

Using the packages, develop a system that can do the following:

i) Add new book in booklist ii) Add more copies for a book iii) Show all book details iv) Show details of a book v) Add member in the list vi) show all members vii) show details of a member viii) Issue a book (check book validity and availability, check member validity and eligibility to get a book, once passes through the validations add an entry into transaction list and update counts in corresponding booklist and memberlist entries) ix) book return book (check the validity of corresponding issue with book id and member id and once passes through the validations update the transaction entry by marking member id as 'xxxx' and update counts in corresponding booklist and memberlist entries)

Consider the **list as arrays**. While working with arrays it is to be ensured that use of indices out of the range is reported.

4. Consider the packages designed in previous question. Design an interface to ensure that the library management must have the option i) add book ii) search book iii) view all book iv) add member v) search a member vii) view all members viii) issue book ix) return book.

Design the system by implementing the interface.

5. Design a student class with roll, name and score. Support must be there to set the score. Score is non-negative and cannot exceed 100. If the marks is less than 40, the candidate will be declared as failed. For invalid score an exception has to be raised. User of set score method will decide about the measures to deal with the exception.

6. Consider a wrapper class for a numeric basic type. Check the support for the following: conversion from i) basic type to object ii) object to basic type iii) basic type to String iv) String (holding numeric data) to numeric object v) object to String.

7. Take a String input that contains multiple words. Do the following: i) number of times ‘a’, ‘an’, ‘the’ appears ii) number of times “and”, “or” appears iii) whether it starts with “The” or not iv) put the String into an array of characters v) display the tokens in the String (tokens are the substrings separated by space or @ or .)