**Project Description**

Simulate the realization of "Customer Information Management Software" based on the text interface.

The software can insert, modify and delete customer objects (implemented by array), and can print customer schedules

**Purpose**

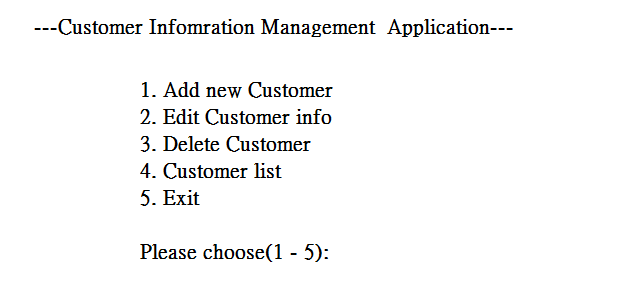
To test my understanding of Java language, knowledge of Oracle Object Programming and how to design a software application.

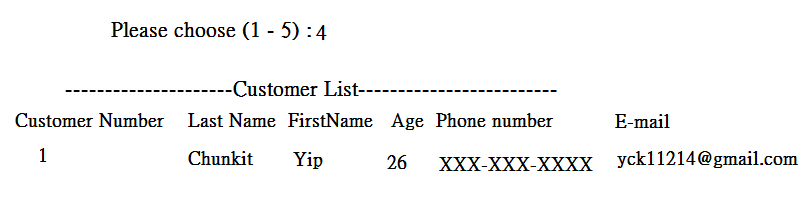
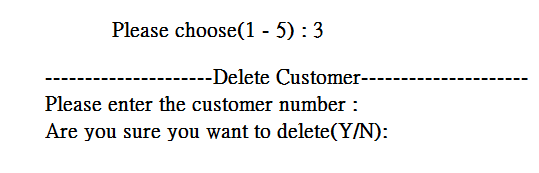
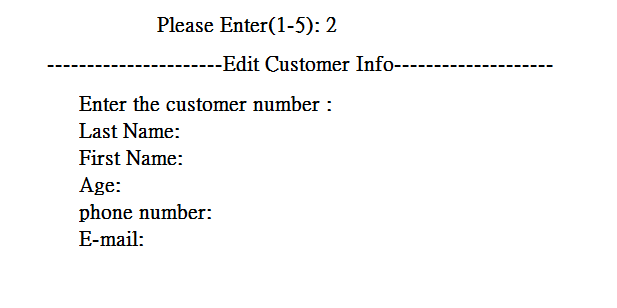
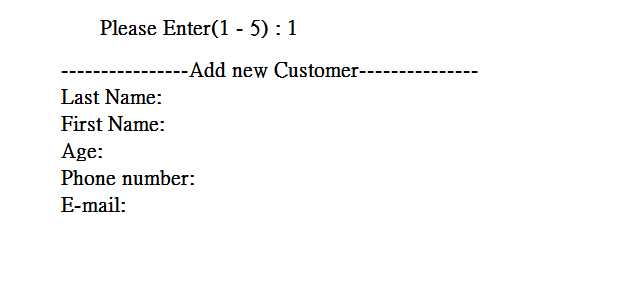
**Project requirements:**

Each customer's information is stored in the Customer object.

* Use an array of Customer type to record all current customers.
* After each "Add Customer" (Menu 1), the Customer object is added to the array.
* After each "Edit customer" (menu 2), the modified Customer object replaces the original object in the array.
* After each "delete customer" (menu 3), the Customer object is removed from the array.
* When executing "Customer List" (Menu 4), the information of all customers in the array will be listed.

**The simulation of application**

****



* CustomerView will be the user interface which display the menu and handling user input actions and located in view module
* Customer is instance class which encapsulating customer information (Encapsulation)
* CustomerList is the management module of Customer object, internally manages a group of Customer objects with an array, and provides corresponding methods of adding, editing, deleting, and traversing for CustomerView to use
* CMUtility is a class to store method to run CustomerView

**Customer:** instance class which encapsulating customer information (Encapsulation)

Customer class should have those following Instance variables:

* String lastname
* String firstname
* int age:
* String phonenumber
* String email
* Constructor
* get/set method
* getCustomerInfo() method : to display customer information

**CustomerList:** is the management module of Customer object, internally manages a group of Customer objects with an array, and provides corresponding methods of adding, editing, deleting, and traversing for CustomerView to use

Instance variables:

* Customer (To initialize the customer array)
* addCustomer() method (to add Customer)
* replaceCustomer() method (to replace specific customer)
* deleteCustomer() method(to delete specific customer)
* getAllCustomers() method(to return all customer)
* getCustomer() method(to return specific customer)
* getTotal() method(to return the number of customers

**CMUtility**: a class to store method to run CustomerView

* Static variable Scanner
* readMenuSelection() method(check is user input the right keyword)
* readChar() (read a character from the keyword and use it to return)
* readInt() method (read an integer with a length of no more than 2 digits)
* readInt(int defaultValue) method (read an integer with a length of no more than 2 digits and use it to return)
* readString(int limit) method (Read a string that does not exceed limit from the keyboard)
* readString(int limit , String defaultValue) method (Read a string that does not exceed limit from the keyboard and use it to return)
* readConfirmSelection() method (Used to confirm the selected input , reads Y or N)
* readKeyBoard() method (check the scanner input to see any illegal input)

**CustomerView:** will be the user interface which display the menu and handling user input actions and located in view module

* Instance variables

CustomerList customerList

* Constructor (Set the one customer to test)
* enterMainMenu() method (Display the interface of Customer Information Management Application)
* addNewCustomer() method (Add new Customer)
* editCustomer() method (Edit Customer information using customer id)
* deleteCustomer() method (Delete customer information using customer id)
* listAllCustomer() method (List all the Customer information)
* main() method