1. **MyArrayList.java:** In this Program I have created a class called MyArrayList in which I have created an Object of type ArrayList. This Object will be created as when the class MyArrayList is instatiated.

Some methods are created to add, change, and remove content from the ArrayList.

Here I have used set(), add(), remove() and get() methods from ArrayList.

1. **MyArrayList2.java**: in this program I have designed a system which takes “Action” as an input from the user and then decides which constructor to invoke. Here I have used Multilevel inheritance.

In this example MyArrayList2 is the highest class which is then extended by HandleInputs and further HandleInputs is extended by Child class.

MyArrayList2 class is used only to instatiate Arraylist class and make new Object newAl.

HandleInputs class is used to store all instance variables which will be then used in the Child class to perform certain actions.

In the main method, we are using scanner class to take input from user to understand what operation user wants to perform on the Arraylist. Depending upon the user input, program will decide which constructor to invoke and which method to execute.

To avoid ArrayOutOfBoud Exception I have added a while loop which will keep the control of the program in the try -catch block until user enters a valid input.

1. **Employee.java:** This program aims towards making a system to enter new employees with their employee Id. The parent class Employee takes care of creating the Object of type HashMap which holds the record of an employee in the form of key value pair. And only Admin class will have authority to make any changes to this record.

Admin class is FINAL, hence it can not be extended by any other class making all these methods secure.

Add:- new employee can not be added directly through main method. To add a new employee, first add method of Admin class is called. The admin then will enter the name of the new employee and automatically employee number will be given to it.

Similarly an employee will be removed with the help of methods from Admin class.

1. **MenuCard.java**: This program tries to mimic the ordering system at any restaurant. The Parent class MenuCard is used to initialize the variables and objects. Here we are using two different HashMap object to create two different views. One for the Manager of the restaurant where manager would see the manu as key value pairs of menuitem with its code. On the other hand, guest will see as a key value pair of menu item and its price. This code has some bugs in it and might not work as expected for now, however I will be fixing it very shortly.
2. **Lottery.java**:In this particular example I have made use of one of Set’s property.

Sets are unordered in nature, and hence we can not expect the same order as we inserted. This property will be best used to draw lucky winners.

The program takes 10 inputs from the user as participants and then printouts top 3. As Sets are unordered we get shuffled results.