**Collection**

1 – Student Marks

**Create a Class StudentMarks with a main method. Use Scanner to accept marks scored by Students.**

**Add the marks to an ArrayList. (two or more students can have the same marks)**

**Display the highest mark scored by the student.**

**Display the Average marks scored by the Students**

**Display the marks stored by the 3rd Student**

**Sort the marks and display the marks using Iterator.**

Sample Input

Enter number of Students

4

Enter Marks

40 60 60 50

Sample Output

Highest Marks : 60

Average Marks : 52.5

Marks : 1-40 2-60 3-60 4-50

3rd Student marks : 60

Sorted : 1-40 2-50 3-60 4-60

**2 – Cricket Score Analyzer**

**Create a Class ScoreAnalyzer with a private instance variable 'runsData' of type LinkedList.**

**The linkedlist will store runs scored by individual players in One Day Cricket match.**

**Create a constructor and initialize runsData variable.**

**• Create a method addRunsToList which accepts runs scored by a player as parameter and stores it in the List.**

**• Create a method calcRunRate which returns the runRate (Consider all 50 overs were played by a team).**

**• Create a method lowestRunsScored to return the lowest runs scored by a player.**

**• Create a method displayRuns to display the runs scored by all players.(use for each loop)**

**Create class TestScoreAnalyzer with a main method.**

**• Create an object of ScoreAnalyzer**

**• Use Scanner to get the runs scored and use addRuns method to store the runs**

**• Display the runs scored by all players**

**• Display the runrate**

**• Display the lowest runs scored**

**• Display the count of player who did batting**

Sample input

Enter Runs

20 30 150 50 2

Sample Output

Runs Scored : 1-20 2-30 3-150 4-50 6-2

Runrate : 5.04

Lowest Runs : 2

Count of player who batted: 5

**3 – ScoreCard**

**Create a class ScoreCard with a main method to analyze the runs scored by batsmen of a team.**

**The input data format is given below. The program needs to output as given below. Use a concrete class of Map to achieve the functionality.**

**Sample Input**

**-------------**

**Enter Runs Scored**

**Rahane-20 Rahul-30 Kohli-150 Dhoni-50 Lokesh-2**

**Sample Output**

**-------------**

**Players who batted**

**Dhoni**

**Kohli**

**Lokesh**

**Rahane**

**Rahul**

**Scores by Players**

**Dhoni : 50**

**Kohli : 150**

**Lokesh : 2**

**Rahane : 20**

**Rahul : 30**

**Total Score : 202**

**Name of Highest Scorer : Kohli**

**Runs Scored by Dhoni : 50**

**4 – Patient List - Comparable and Comparator**

**Create a class Patient with private instance variables patientId (int), name(String) and age(int). Include parameterized Constructor and Getters/Setters**

**Create a class TestPatientList with a main method.**

**• Create 3 objects of patient and add to an ArrayList**

**• Sort the List based on Name which should be the natural order and display using for each loop**

**• Sort the List based on age which should be the alternate sorting order and display using Iterator**

**• Add all the objects in to a TreeSet and display the TreeSet**

**• Create a static method getPatientAge which accepts name of the patient and TreeSet of patient. Method should return the age of patient. Invoke this method in the main method.**

**5 – Duplicate Patient - HashCode and Equals**

**Use the Patient class created in earlier assignment and modify if needed.**

**Create a class TestPatientSet with a main method.**

**• Create 3 objects of patient and add to a LinkedHashSet. One of the patient should be a duplicate**

**• Display the size of the Set.**

**• Display the patient details for all patients**

6 – States

A File named states.txt contains the following data

Karnataka

Punjab

Telangana

Maharashtra

TamilNadu

Uttar Pradesh

Punjab

Karnataka

Kerala

Delhi

**Create a program to store the names of states of India in a LinkedHashSet named stateSet by reading the file. ( Use readLine method of BufferedReader )**

**The program should do the following:**

**• Display the total number of states**

**• Remove Delhi from the Set**

**• Display the states which have name starting with K**

**• Display the states in a sorted Order ( Use another Set)**