DEPARTMENT OF ACADEMIC AFFAIRS

Discover. Learn. Empower.

NAAC GRADE A+ ACCREDITED UNIVERSITY

WORKSHEET – 5

Student Name: Anjali Singh
UID: 20BCS9239
Branch: BE-CSE
Section/Group: 607 A
Semester: 5TH
Date :29/09/2022
Subject Name: Project based learning with Java
Subject Code: 20CSP-321

Aim: Collect Unique Symbols From Set of Cards

Software used : IntelliJ

Task to be done:

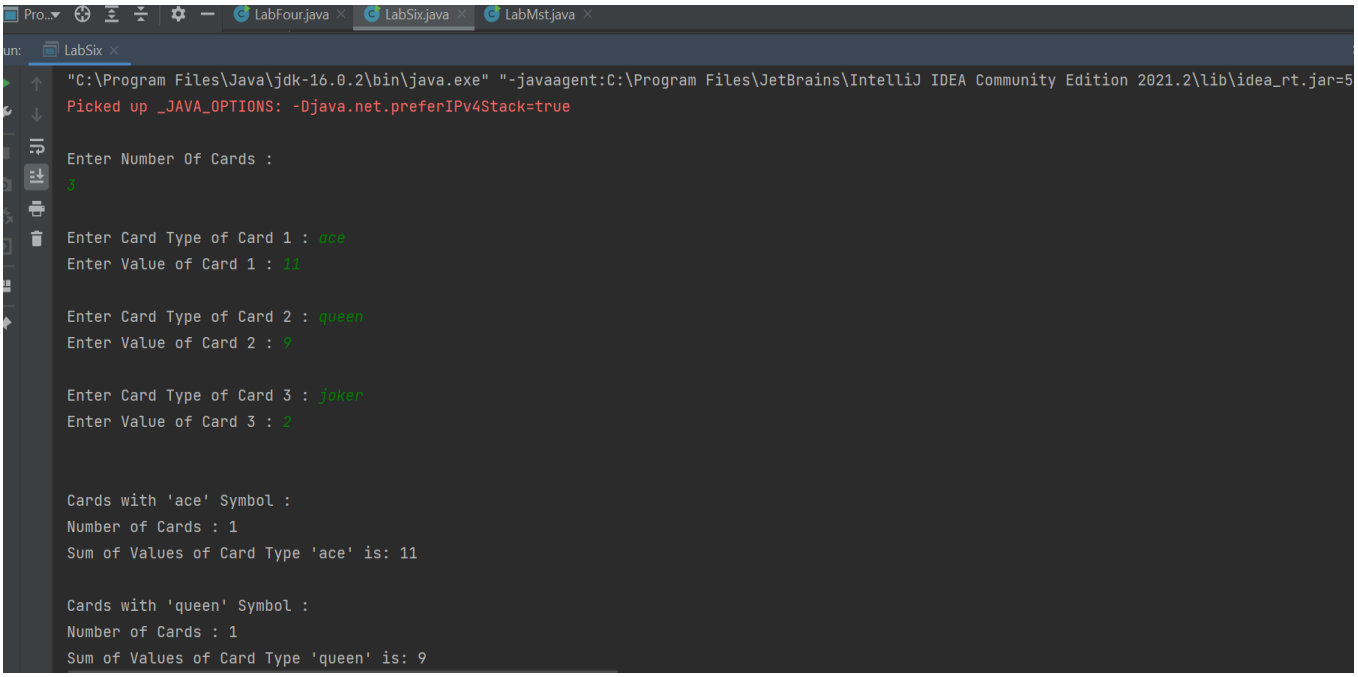
Playing cards during travel is a fun filled experience. For this game they wanted to collect all four unique symbols. Can you help these guys to collect unique symbols from a set of cards? Create Card class with attributes symbol and number. From our main method collect each card details (symbol and number) from the user. Collect all these cards in a set, since set is used to store unique values or objects. Once we collect all four different symbols display the first occurrence of card details in alphabetical order.

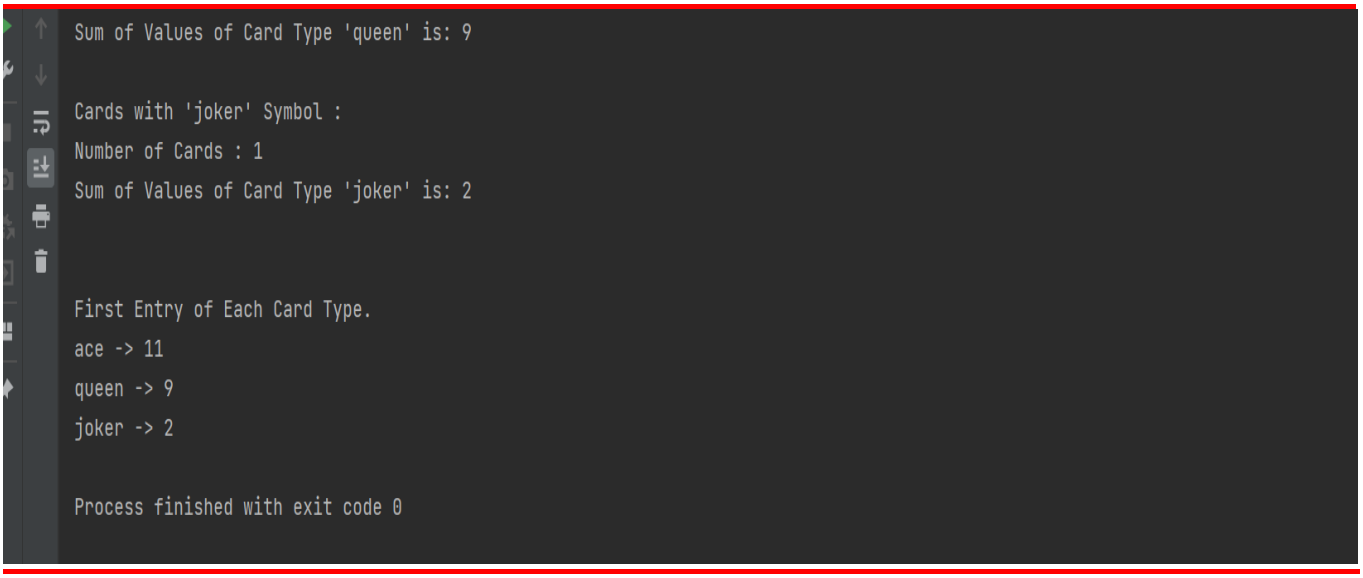
CODE

```
package com.nirvana;

import java.util.*;

public class LabSix {
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);
        List<Integer> cardValue = new ArrayList<Integer>();
        HashMap<String, List<Integer>> mp = new HashMap<>();
        HashMap<String, Integer> firstEntry = new HashMap<>();

        int totalNoOfCards, value;
        System.out.println("\nEnter Number Of Cards : ");
        totalNoOfCards = input.nextInt();
        String cardType;

        for(int index = 1; index <= totalNoOfCards; index++){
            System.out.print("\nEnter Card Type of Card " + index + " : ");
            cardType = input.next();
            System.out.print("Enter Value of Card " + index + " : ");
            value = input.nextInt();

            if(mp.containsKey(cardType)){
                if(cardValue.contains(value)){
                    System.out.println("No same card type have same
value.\nProcess terminated.");
                    System.exit(0);
                }
                cardValue = mp.get(cardType);
                cardValue.add(value);
            }else{
                cardValue = new ArrayList<Integer>();
                firstEntry.put(cardType,value);
                cardValue.add(value);
                mp.put(cardType, cardValue);
            }
        }
        System.out.println();

        int noOfSameCard=0,sumOfSameCard=0;
        for(Map.Entry getData : mp.entrySet()){
            System.out.println("\nCards with '" + getData.getKey() + "'
Symbol : ");
            ArrayList<Integer> symbolArray = (ArrayList<Integer>)
```

```
getData.getValue();
            Iterator itr= symbolArray.iterator();
            while(itr.hasNext()){
                noOfSameCard++;
                int num = (int) itr.next();
                sumOfSameCard += num;
            }

            System.out.println("Number of Cards : " + noOfSameCard);
            System.out.println("Sum of Values of Card Type
'"+getData.getKey()+"' is: "+ sumOfSameCard);
            sumOfSameCard = 0;
            noOfSameCard = 0;
        }

        System.out.println("");
        System.out.println("\nFirst Entry of Each Card Type.");
        for(Map.Entry getData : firstEntry.entrySet()){
            System.out.println(getData.getKey() + " ->
"+getData.getValue());
        }
    }
}
```

OUTPUT


LabFour.java
LabSix.java
LabMst.java
LabSix
"C:\Program Files\Java\jdk-16.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2021.2\lib\idea_rt.jar=5
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Enter Number Of Cards :
3
Enter Card Type of Card 1 : ace
Enter Value of Card 1 : 11
Enter Card Type of Card 2 : queen
Enter Value of Card 2 : 9
Enter Card Type of Card 3 : joker
Enter Value of Card 3 : 2
Cards with 'ace' Symbol :
Number of Cards : 1
Sum of Values of Card Type 'ace' is: 11
Cards with 'queen' Symbol :
Number of Cards : 1
Sum of Values of Card Type 'queen' is: 9


Sum of Values of Card Type 'queen' is: 9
Cards with 'joker' Symbol :
Number of Cards : 1
Sum of Values of Card Type 'joker' is: 2
First Entry of Each Card Type.
ace -> 11
queen -> 9
joker -> 2
Process finished with exit code 0

LEARNING OUTCOME

1.Learned about collection framework.

2.Hashmap, tree map and list were implemented.