# Experiment No: 2.1

**Student Name: Kamal Ale Magar** **UID: 21BCS10155**
**Branch: CSE** **Section/Group: 616/B**
**Semester: 6th** **Date of Performance: 19/02/24**
**Subject: Project Based Learning in Java with Lab** **Subject Code: 21CSH-319**

## 1. Aim:

Create a program to collect and store all the cards to assist the users in finding all the cards in a given symbol using Collection interface.

## Task:

Collect and Group Cards

Write a program to collect and store all the cards to assist the users in finding all the cards in a given symbol.

This cards game consist of N number of cards. Get N number of cards details from the user and store the values in Card object with the attributes symbol and number.

Store all the cards in a map with symbol as its key and list of cards as its value. Map is used here to easily group all the cards based on their symbol.

Once all the details are captured print all the distinct symbols in alphabetical order from the Map. For each symbol print all the card details, number of cards and their sum respectively.

## 2. Objective:

• To learn about concept of Hashing.

• To learn about HashMap..

## 3. Input/Apparatus Used:

VS Code

## 4. Procedure/Algorithm/Pseudocode:

Step 1: Create a card class having  symbol and number.

Step 2: Add details of cards in hashmap.

Step 3: Check if card numbers are already present in hashmap then ignore.

Step 4: If it is not present then replace previous details and insert modified details.

Step 5: Then after iteration print number of cards in each symbol and sum of number in each symbol.

## 5. Code:

```
import java.util.*;


class Card {
    String symbol;
    int number;

    Card(String symbol, int number) {
        this.symbol = symbol;
        this.number = number;
    }

    @Override
    public String toString() {
        return symbol + " " + number;
    }
}
```

```
public class CardCollector {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter Number of Cards:");
        int n = scanner.nextInt();
        scanner.nextLine(); // Consume newline

        Map<String, List<Card>> cardMap = new TreeMap<>();

        for (int i = 0; i < n; i++) {
            System.out.println("Enter card " + (i + 1) + ":");
            String symbol = scanner.nextLine().trim();
            int number = scanner.nextInt();
            scanner.nextLine(); // Consume newline
            Card card = new Card(symbol, number);
            cardMap.computeIfAbsent(symbol, k -> new ArrayList<>()).add(card);
        }

        System.out.println("Distinct Symbols are :");
        for (String symbol : cardMap.keySet()) {
            System.out.print(symbol + " ");
        }
        System.out.println();

        for (Map.Entry<String, List<Card>> entry : cardMap.entrySet()) {
            String symbol = entry.getKey();
            List<Card> cards = entry.getValue();
            System.out.println("Cards in " + symbol + " Symbol");
```

```
            int sum = 0;
            for (Card card : cards) {
                System.out.println(card);
                sum += card.number;
            }
            System.out.println("Number of cards : " + cards.size());
            System.out.println("Sum of Numbers : " + sum);
        }
        scanner.close();
    }
}
```

## 6. Result/Output:

```
PS E:\All\STUDY\Sixth Semester\Java\Code>  & 'C:\Program Files\Java\jdk-18.0.2.1\bin\java.
sers\HP\AppData\Roaming\Code\User\workspaceStorage\0fd6c3ab0343839c2213661dd4637541\redhat
Enter Number of Cards:
13
Enter card 1:
s
1
Enter card 2:
s
12
Enter card 3:
s
13
Enter card 4:
d
4
Enter card 5:
c
5
Enter card 6:
h
5
Enter card 7:
h
7
```

```
Enter card 8:
c
3
Enter card 9:
c
2
Enter card 10:
h
9
Enter card 11:
s
7
Enter card 12:
d
4
Enter card 13:
d
3
Distinct Symbols are :
c d h s
Cards in c Symbol
c 5
c 3
c 2
Number of cards : 3
```

```
 Number of cards : 3
 Sum of Numbers : 10
 Cards in d Symbol
 d 4
 d 4
 d 3
 Number of cards : 3
 Sum of Numbers : 11
 Cards in h Symbol
 h 5
 h 7
 h 9
 Number of cards : 3
 Sum of Numbers : 21
 Cards in s Symbol
 s 1
 s 12
 s 13
 s 7
 Number of cards : 4
 Sum of Numbers : 33
```

## 7. Learning Outcomes:

- Learnt about the concept of hashing.
- Learn about how to used hashmap.
- Learn about to how to utilize the concept of hashing and hashmap in java