

Name – Sachita Seth

UID – 22BCS12150

Section and Group – IoT_635 (B)

Q.2. Create a program to collect and store all the cards to assist the users in finding all the cards in a given symbol using the Collection interface.

```
import java.util.*;
```

```
class Card {
```

```
    String symbol;
```

```
    String value;
```

```
    Card(String symbol, String value) {
```

```
        this.symbol = symbol;    this.value  
= value;
```

```
    }
```

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

```
}
```

```
public class CardCollection {    static Map<String, List<Card>>
```

```
cardMap = new HashMap<>();    static Scanner scanner = new
```

```
Scanner(System.in);
```

```

public static void addCard() {
    System.out.print("Enter Card Symbol (e.g., Hearts, Spades): ");
    String symbol = scanner.nextLine();
    System.out.print("Enter Card Value (e.g., Ace, King, 10): ");
    String value = scanner.nextLine();

    cardMap.putIfAbsent(symbol, new ArrayList<>());
cardMap.get(symbol).add(new Card(symbol, value));
    System.out.println("Card Added Successfully!");
}

public static void findCardsBySymbol() {
    System.out.print("Enter Symbol to Search: ");
    String symbol = scanner.nextLine();

    List<Card> cards = cardMap.get(symbol);    if
(cards != null && !cards.isEmpty()) {
    System.out.println("Cards in " + symbol + ":");
    for (Card card : cards) {
        System.out.println(card);
    }
    } else {
        System.out.println("No cards found for this symbol!");
    }
}

```

```

public static void displayAllCards() {
    if (cardMap.isEmpty()) {
        System.out.println("No Cards Available!");
        return;
    }

    for (Map.Entry<String, List<Card>> entry : cardMap.entrySet()) {
        System.out.println("\nSymbol: " + entry.getKey());        for (Card
        card : entry.getValue()) {
            System.out.println(card);
        }
    }
}

public static void main(String[] args) {
    while (true) {
        System.out.println("\n1. Add Card\n2. Find Cards by Symbol\n3.
        Display All Cards\n4. Exit");

        System.out.print("Enter your choice: ");

        int choice = scanner.nextInt();
        scanner.nextLine();

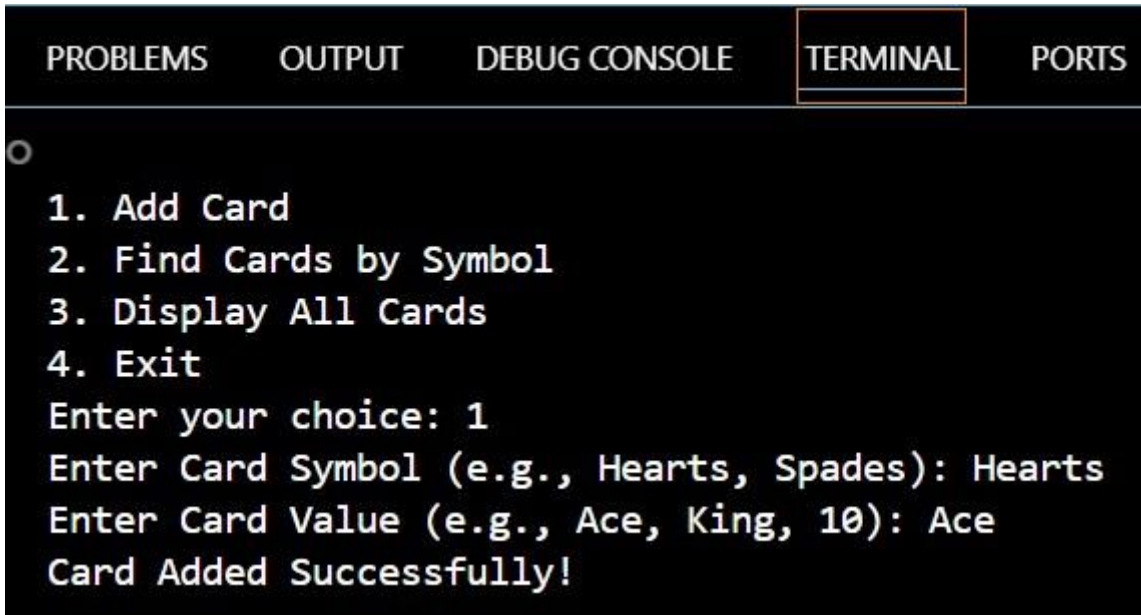
        switch (choice) {
            case 1
            -> addCard();
            case 2 ->

```

```
findCardsBySymbol();           case 3 ->
displayAllCards();

    case 4 -> {
        System.out.println("Exiting...");
        System.exit(0);
    }
    default -> System.out.println("Invalid Choice! Try Again.");
}
}
}
}
```

Output:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
○
1. Add Card
2. Find Cards by Symbol
3. Display All Cards
4. Exit
Enter your choice: 1
Enter Card Symbol (e.g., Hearts, Spades): Hearts
Enter Card Value (e.g., Ace, King, 10): Ace
Card Added Successfully!
```