LAB # 02

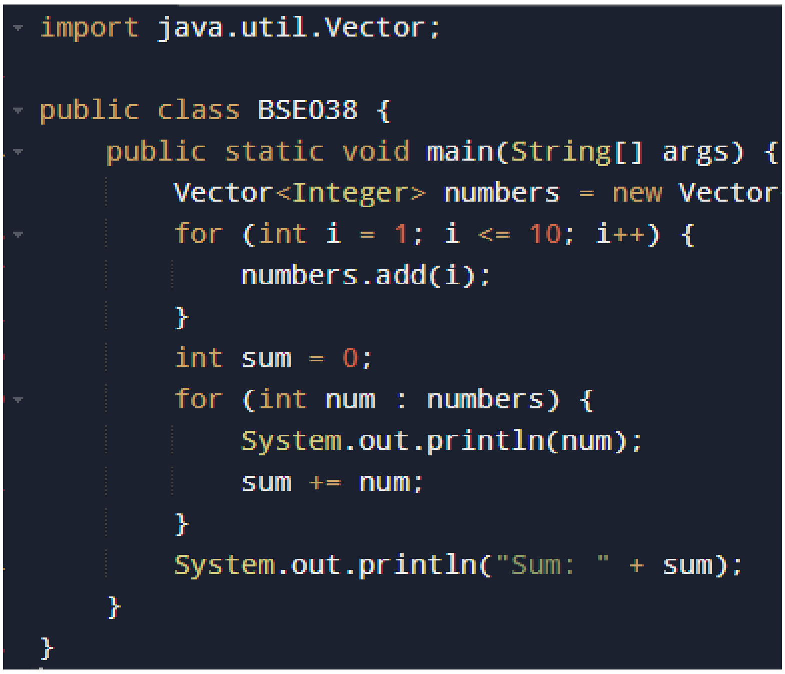
ArrayList and Vector in JAVA

OBJECTIVE: To implement ArrayList and Vector.

LAB TASKS

1. Write a program that initializes Vector with 10 integers in it. Display all the integers and sum of these integers.

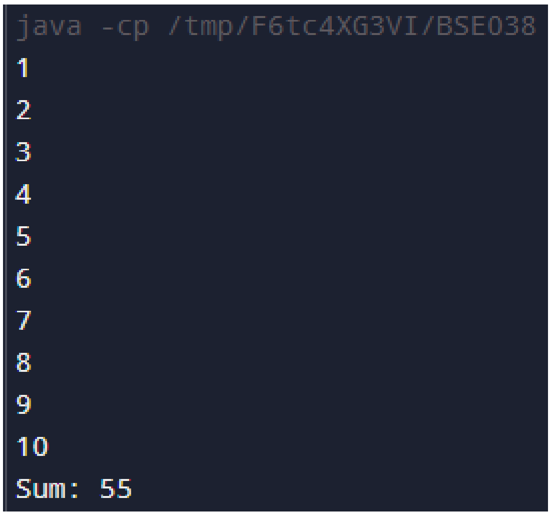
CODE :



```
import java.util.Vector;

public class BSE038 {
    public static void main(String[] args) {
        Vector<Integer> numbers = new Vector
        for (int i = 1; i <= 10; i++) {
            numbers.add(i);
        }
        int sum = 0;
        for (int num : numbers) {
            System.out.println(num);
            sum += num;
        }
        System.out.println("Sum: " + sum);
    }
}
```

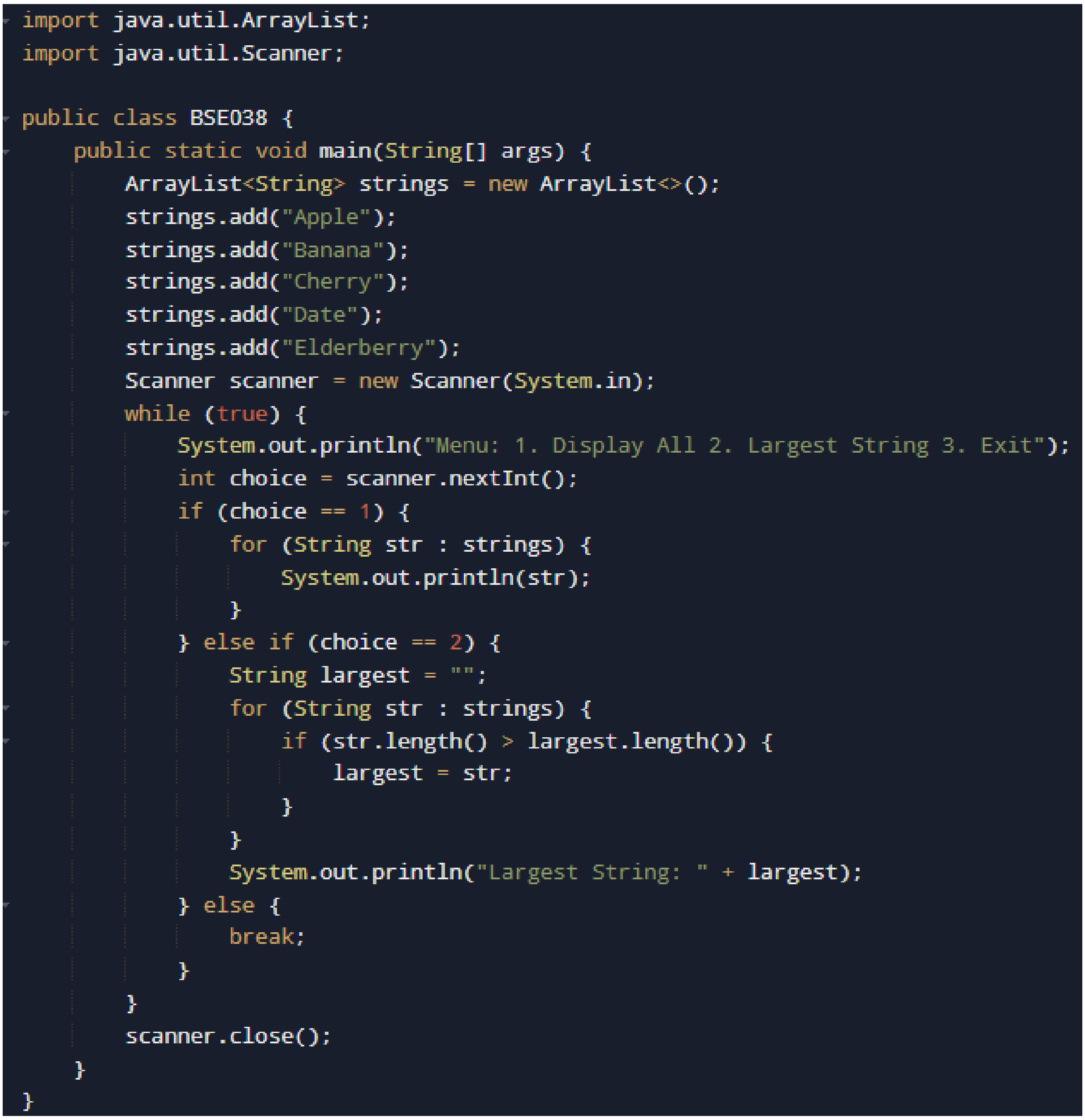
OUTPUT :



```
java -cp /tmp/F6tc4XG3VI/BSE038
1
2
3
4
5
6
7
8
9
10
Sum: 55
```

2. Create a ArrayList of string. Write a menu driven program which: a. Displays all the elements b. Displays the largest String

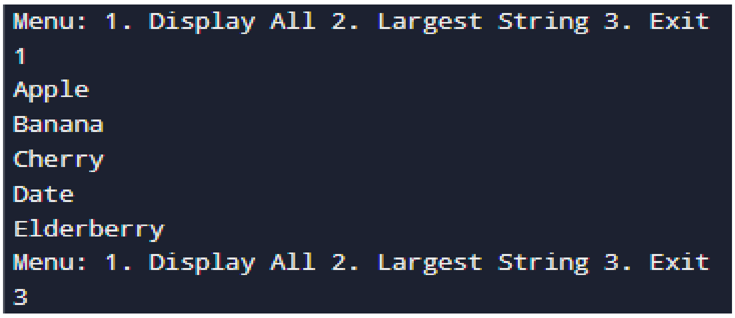
CODE :



```java
import java.util.ArrayList;
import java.util.Scanner;

public class BSE038 {
    public static void main(String[] args) {
        ArrayList<String> strings = new ArrayList<>();
        strings.add("Apple");
        strings.add("Banana");
        strings.add("Cherry");
        strings.add("Date");
        strings.add("Elderberry");
        Scanner scanner = new Scanner(System.in);
        while (true) {
            System.out.println("Menu: 1. Display All 2. Largest String 3. Exit");
            int choice = scanner.nextInt();
            if (choice == 1) {
                for (String str : strings) {
                    System.out.println(str);
                }
            } else if (choice == 2) {
                String largest = "";
                for (String str : strings) {
                    if (str.length() > largest.length()) {
                        largest = str;
                    }
                }
                System.out.println("Largest String: " + largest);
            } else {
                break;
            }
        }
        scanner.close();
    }
}
```

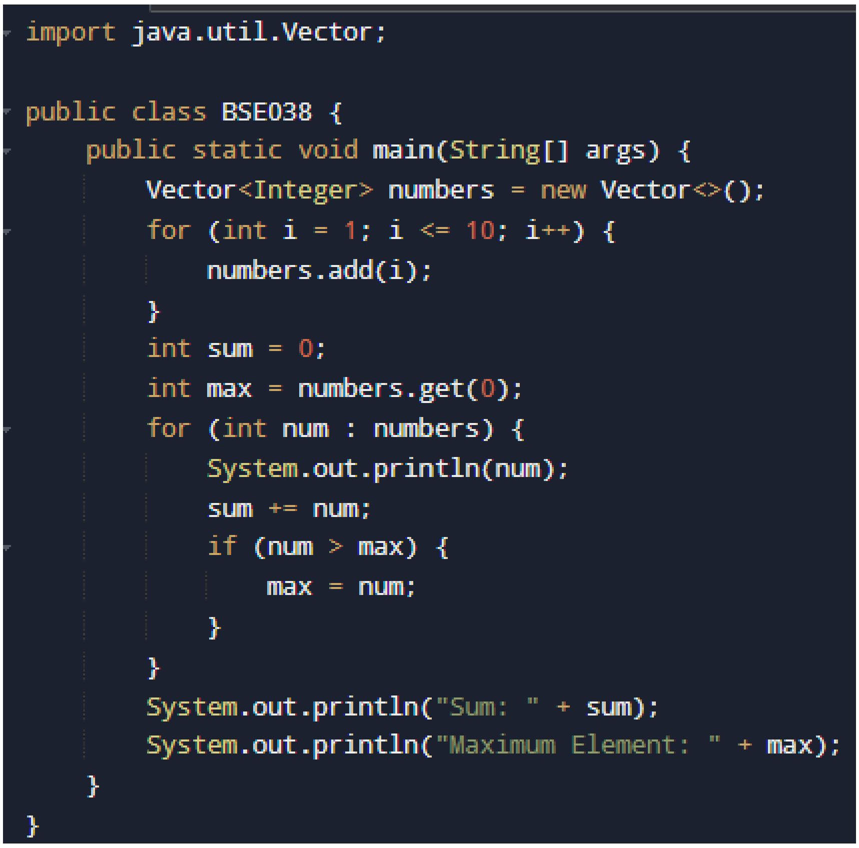
Output :



```
Menu: 1. Display All 2. Largest String 3. Exit
1
Apple
Banana
Cherry
Date
Elderberry
Menu: 1. Display All 2. Largest String 3. Exit
3
```

3. Write a program that initializes Vector with 10 integers in it. Display all the integers Sum of these integers. Find Maximum Element in VectorWrite a program that initialize five different strings and perform the following operations. a. Concatenate all five stings. b. Convert fourth string to uppercase. c. Find the substring from the concatenated string from 8 to onward

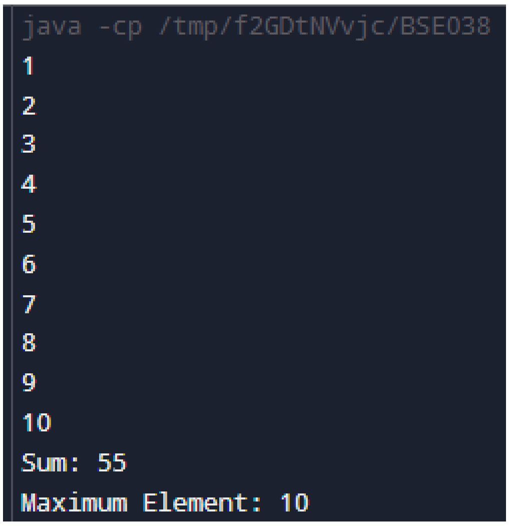
CODE:



```
import java.util.Vector;

public class BSE038 {
    public static void main(String[] args) {
        Vector<Integer> numbers = new Vector<>();
        for (int i = 1; i <= 10; i++) {
            numbers.add(i);
        }
        int sum = 0;
        int max = numbers.get(0);
        for (int num : numbers) {
            System.out.println(num);
            sum += num;
            if (num > max) {
                max = num;
            }
        }
        System.out.println("Sum: " + sum);
        System.out.println("Maximum Element: " + max);
    }
}
```

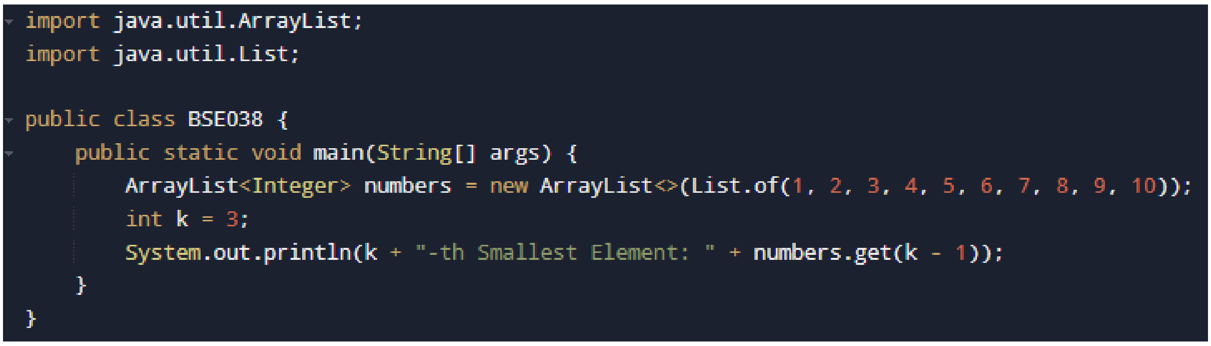
Output :



```
java -cp /tmp/f2GDtNVvjc/BSE038
1
2
3
4
5
6
7
8
9
10
Sum: 55
Maximum Element: 10
```

4. Find the k-th smallest element in a sorted ArrayList

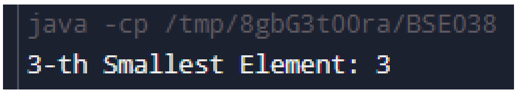
Code:



```java
import java.util.ArrayList;
import java.util.List;

public class BSE038 {
    public static void main(String[] args) {
        ArrayList<Integer> numbers = new ArrayList<>(List.of(1, 2, 3, 4, 5, 6, 7, 8, 9, 10));
        int k = 3;
        System.out.println(k + "-th Smallest Element: " + numbers.get(k - 1));
    }
}
```

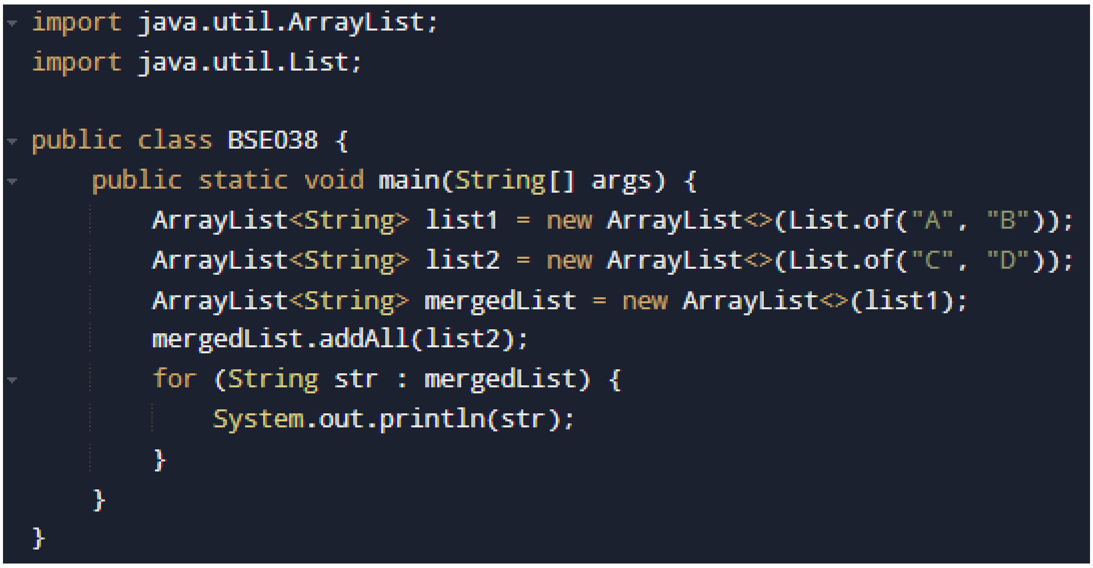
Output:



```
java -cp /tmp/8gbG3t00ra/BSE038
3-th Smallest Element: 3
```

5. Write a program to merge two ArrayLists into one.

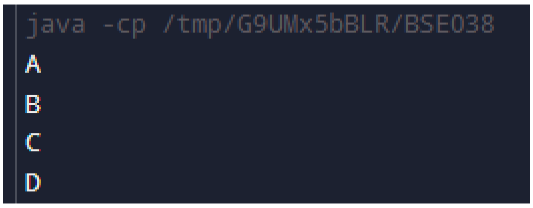
Code:



```java
import java.util.ArrayList;
import java.util.List;

public class BSE038 {
    public static void main(String[] args) {
        ArrayList<String> list1 = new ArrayList<>(List.of("A", "B"));
        ArrayList<String> list2 = new ArrayList<>(List.of("C", "D"));
        ArrayList<String> mergedList = new ArrayList<>(list1);
        mergedList.addAll(list2);
        for (String str : mergedList) {
            System.out.println(str);
        }
    }
}
```

Output:

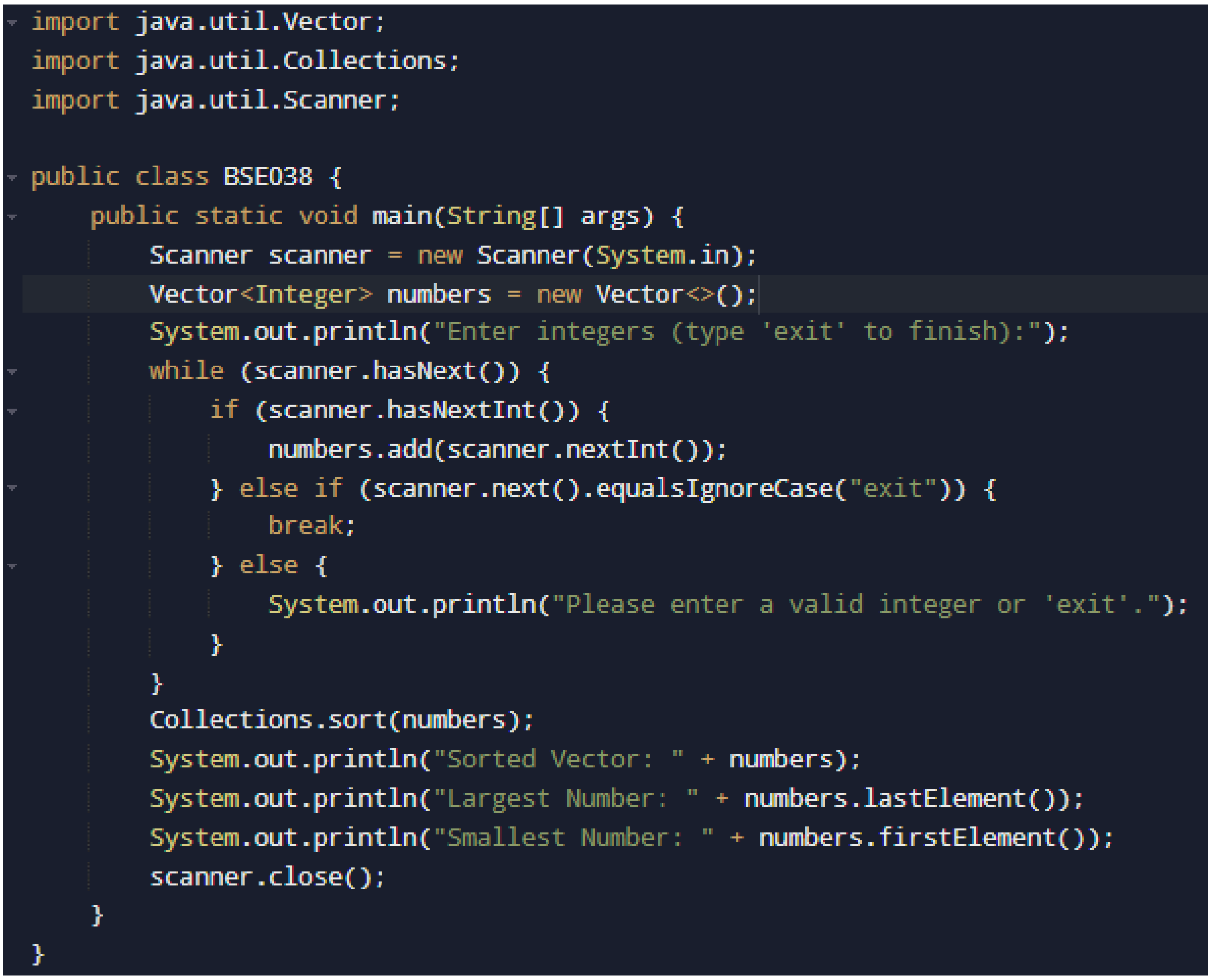


```
java -cp /tmp/G9UMx5bBLR/BSE038
A
B
C
D
```

HOME TASKS

1. Create a Vector storing integer objects as an input. a. Sort the vector b. Display largest number c. Display smallest number

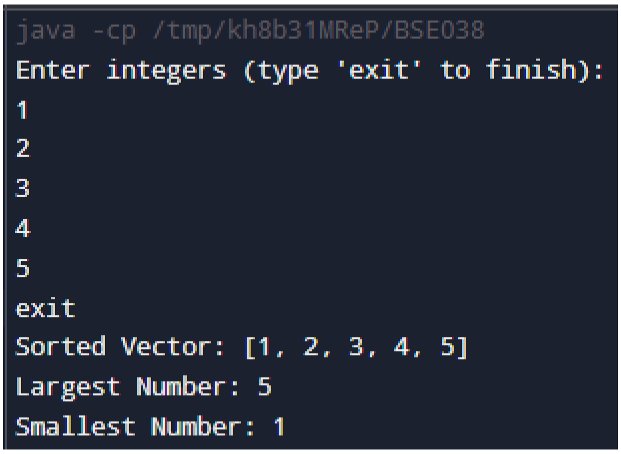
   Code:



```
import java.util.Vector;
import java.util.Collections;
import java.util.Scanner;

public class BSE038 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Vector<Integer> numbers = new Vector<>();
        System.out.println("Enter integers (type 'exit' to finish):");
        while (scanner.hasNext()) {
            if (scanner.hasNextInt()) {
                numbers.add(scanner.nextInt());
            } else if (scanner.next().equalsIgnoreCase("exit")) {
                break;
            } else {
                System.out.println("Please enter a valid integer or 'exit'.");
            }
        }
        Collections.sort(numbers);
        System.out.println("Sorted Vector: " + numbers);
        System.out.println("Largest Number: " + numbers.lastElement());
        System.out.println("Smallest Number: " + numbers.firstElement());
        scanner.close();
    }
}
```

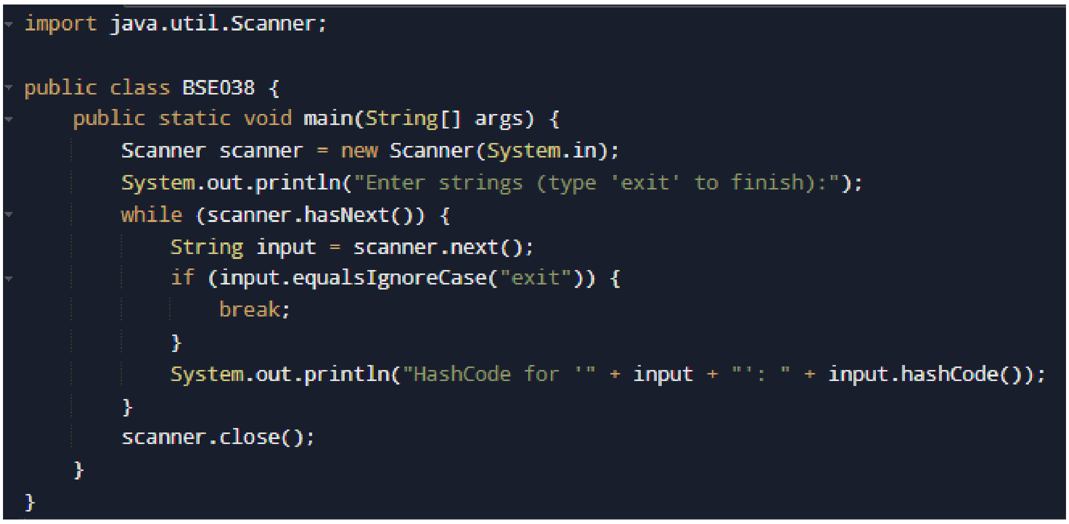
Output:



```
java -cp /tmp/kh8b31MReP/BSE038
Enter integers (type 'exit' to finish):
1
2
3
4
5
exit
Sorted Vector: [1, 2, 3, 4, 5]
Largest Number: 5
Smallest Number: 1
```

2. Write a java program which takes user input and gives hashcode value of those inputs using hashCode () method.

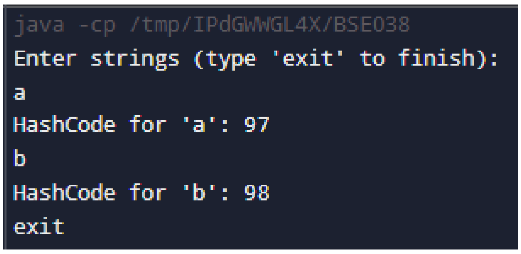
   Code:



```java
import java.util.Scanner;

public class BSE038 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter strings (type 'exit' to finish):");
        while (scanner.hasNext()) {
            String input = scanner.next();
            if (input.equalsIgnoreCase("exit")) {
                break;
            }
            System.out.println("HashCode for '" + input + "': " + input.hashCode());
        }
        scanner.close();
    }
}
```

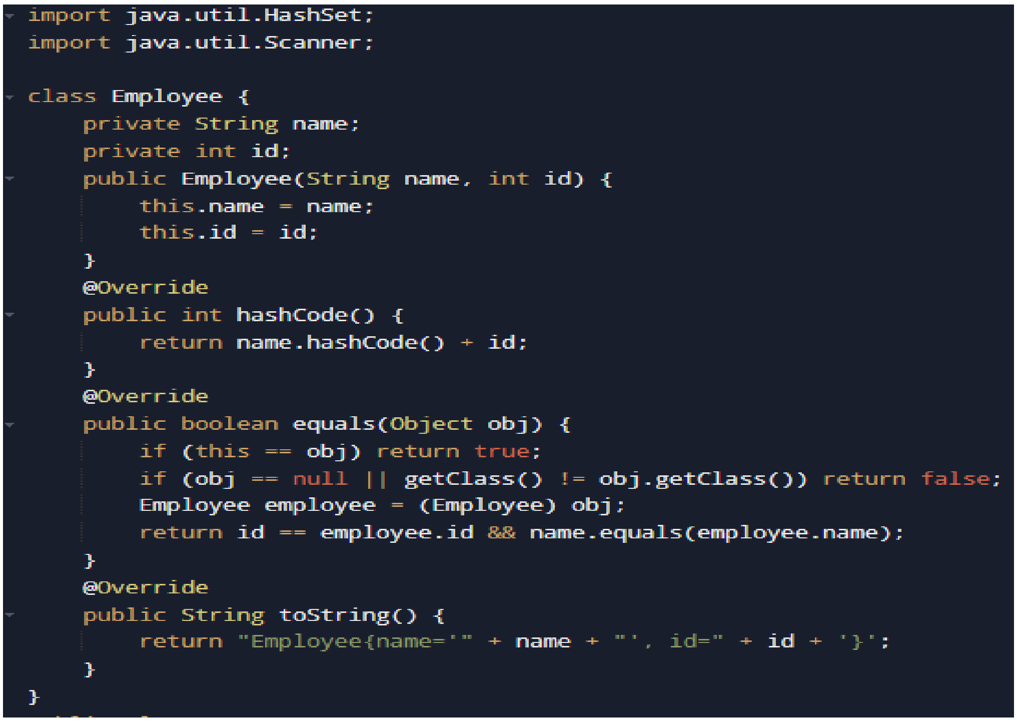
Output:



```
java -cp /tmp/IPdGWWGL4X/BSE038
Enter strings (type 'exit' to finish):
a
HashCode for 'a': 97
b
HashCode for 'b': 98
exit
```

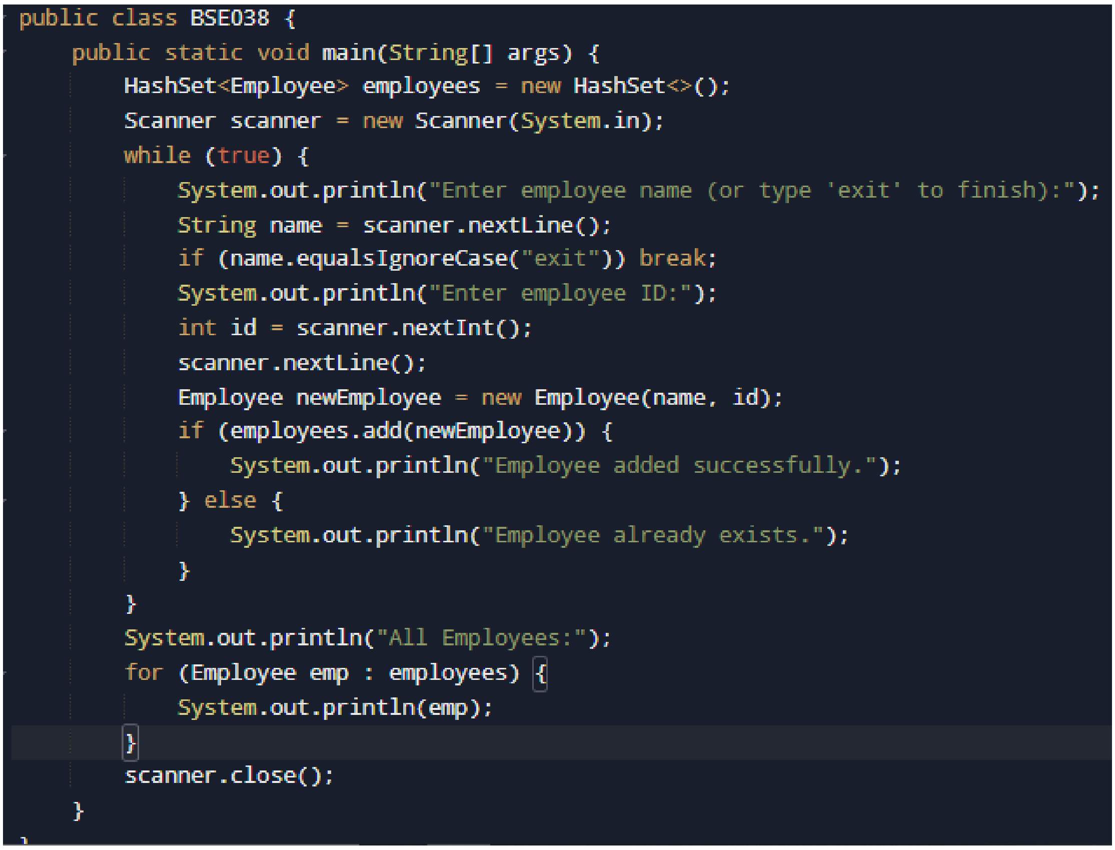
3. Create a java project, suppose you work for a company that needs to manage a list of employees. Each employee has a unique combination of a name and an ID. Your goal is to ensure that you can track employees effectively and avoid duplicate entries in your system.

   Code:



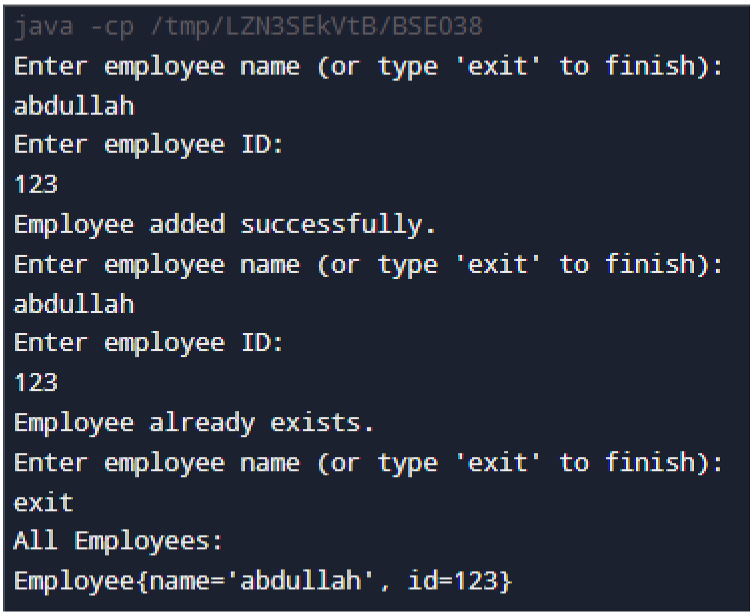
```java
import java.util.HashSet;
import java.util.Scanner;

class Employee {
    private String name;
    private int id;
    public Employee(String name, int id) {
        this.name = name;
        this.id = id;
    }
    @Override
    public int hashCode() {
        return name.hashCode() + id;
    }
    @Override
    public boolean equals(Object obj) {
        if (this == obj) return true;
        if (obj == null || getClass() != obj.getClass()) return false;
        Employee employee = (Employee) obj;
        return id == employee.id && name.equals(employee.name);
    }
    @Override
    public String toString() {
        return "Employee{name='" + name + "', id=" + id + '}';
    }
}
```



```
public class BSE038 {
    public static void main(String[] args) {
        HashSet<Employee> employees = new HashSet<>();
        Scanner scanner = new Scanner(System.in);
        while (true) {
            System.out.println("Enter employee name (or type 'exit' to finish):");
            String name = scanner.nextLine();
            if (name.equalsIgnoreCase("exit")) break;
            System.out.println("Enter employee ID:");
            int id = scanner.nextInt();
            scanner.nextLine();
            Employee newEmployee = new Employee(name, id);
            if (employees.add(newEmployee)) {
                System.out.println("Employee added successfully.");
            } else {
                System.out.println("Employee already exists.");
            }
        }
        System.out.println("All Employees:");
        for (Employee emp : employees) {
            System.out.println(emp);
        }
        scanner.close();
    }
}
```

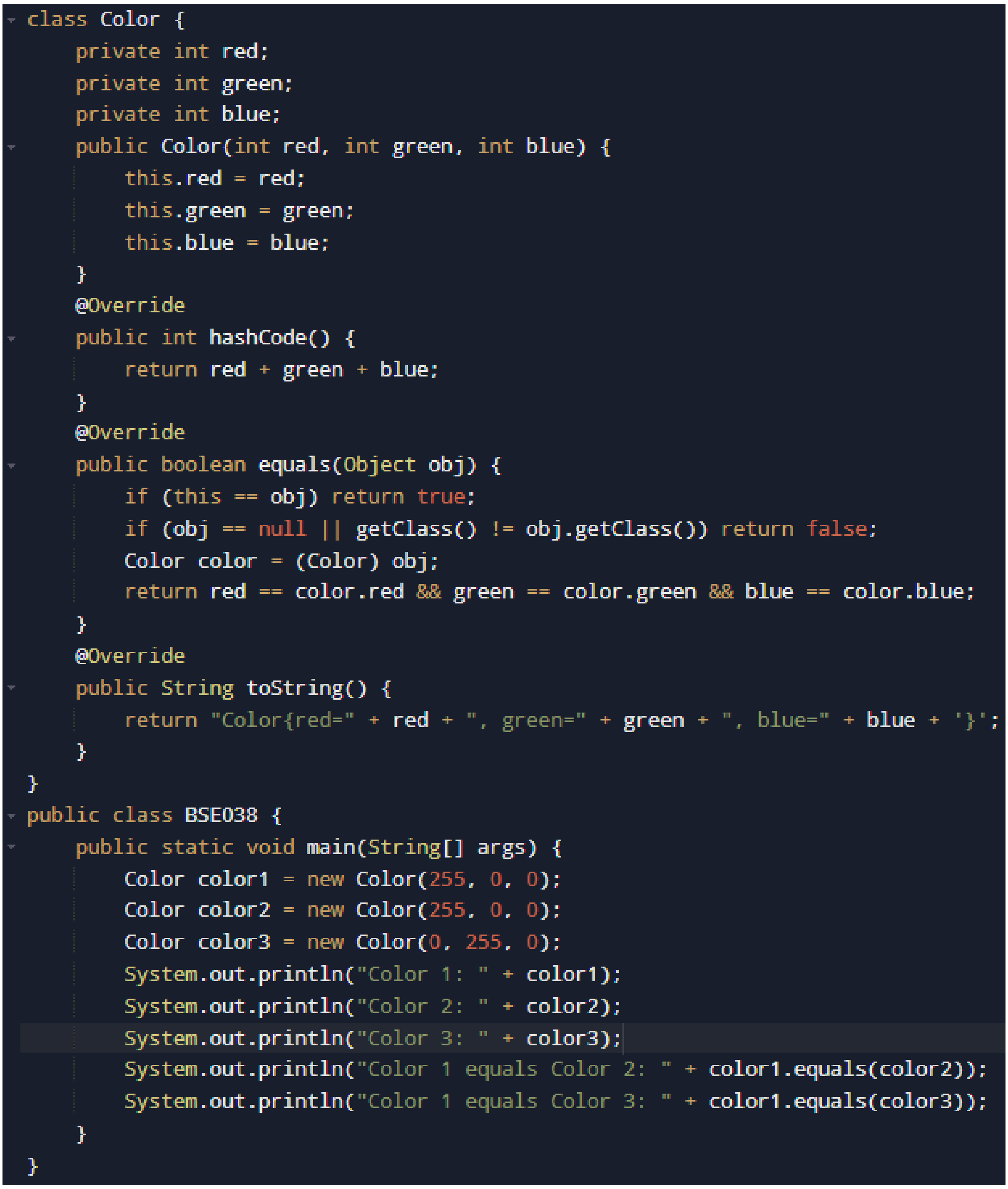
Output:



```
java -cp /tmp/LZN3SEkVtB/BSE038
Enter employee name (or type 'exit' to finish):
abdullah
Enter employee ID:
123
Employee added successfully.
Enter employee name (or type 'exit' to finish):
abdullah
Enter employee ID:
123
Employee already exists.
Enter employee name (or type 'exit' to finish):
exit
All Employees:
Employee{name='abdullah', id=123}
```

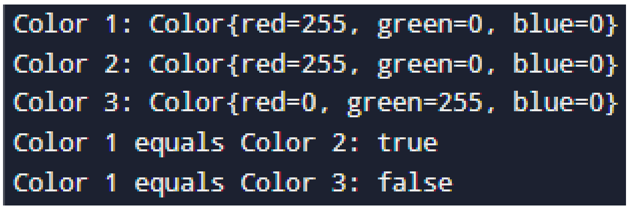
4. Write a Java program to reverse only the vowels in a string

Code:



```java
class Color {
    private int red;
    private int green;
    private int blue;
    public Color(int red, int green, int blue) {
        this.red = red;
        this.green = green;
        this.blue = blue;
    }
    @Override
    public int hashCode() {
        return red + green + blue;
    }
    @Override
    public boolean equals(Object obj) {
        if (this == obj) return true;
        if (obj == null || getClass() != obj.getClass()) return false;
        Color color = (Color) obj;
        return red == color.red && green == color.green && blue == color.blue;
    }
    @Override
    public String toString() {
        return "Color{red=" + red + ", green=" + green + ", blue=" + blue + '}';
    }
}
public class BSE038 {
    public static void main(String[] args) {
        Color color1 = new Color(255, 0, 0);
        Color color2 = new Color(255, 0, 0);
        Color color3 = new Color(0, 255, 0);
        System.out.println("Color 1: " + color1);
        System.out.println("Color 2: " + color2);
        System.out.println("Color 3: " + color3);
        System.out.println("Color 1 equals Color 2: " + color1.equals(color2));
        System.out.println("Color 1 equals Color 3: " + color1.equals(color3));
    }
}
```

Output:



```
Color 1: Color{red=255, green=0, blue=0}
Color 2: Color{red=255, green=0, blue=0}
Color 3: Color{red=0, green=255, blue=0}
Color 1 equals Color 2: true
Color 1 equals Color 3: false
```