WEEK-6

(A) AIM: Design a package to contain the class Student that contains data members such as name, roll number and another package contains the interface Sports which contains some sports information. Import these two packages in a package called Report which process both Student and Sport and give the report.

PROGRAM:

Contents of ***student/Student.java:***

```
package student;

public class Student {
  public String name;
  public int rollNumber;
  public Student(String name,int rollNumber){
    this.name = name;
    this.rollNumber = rollNumber;
  }
  public String getName(){
    return name;
  }
  public int getRollNumber(){
    return rollNumber;
  }
}
```

Contents of ***sports/Cricket.java:***

```
package sports;

public class Cricket implements spo {
  @Override
  public String getSportName(){
    return "Cricket";
  }
}
```

Contents of ***sport/Football.java:***

```
package sports;

public class Football implements spo {
  @Override
  public String getSportName(){
    return "Football";
  }
}
```

Contents of *sports/Hockey.java:*

```
package sports;

public class Hockey implements spo {
  @Override
  public String getSportName(){
    return "Hockey";
  }
}
```

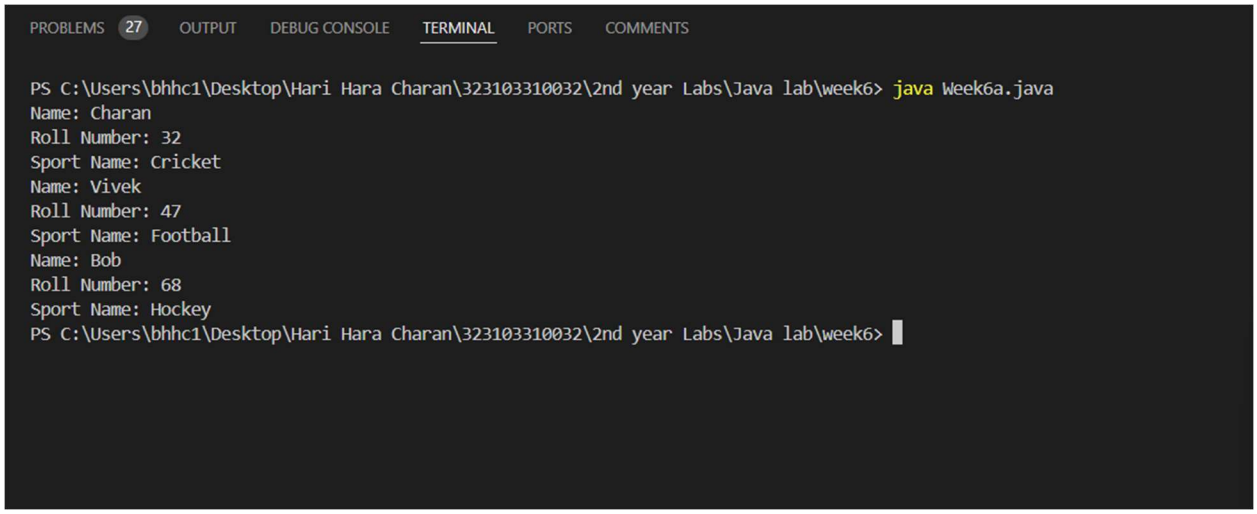
Contents of *sport/Spo.java:*

```
package sports;

public interface spo {

    String getSportName();

}
```

Contents of *Week6a.java:*

```
import report.rep;
import sports.Cricket;
import sports.Football;
import sports.Hockey;
import student.Student;

public class Week6a {
  public static void main(String args[]){
      // 1st report for student s1 who plays cricket
        Student s1 = new Student("Charan", 32);
        Cricket cri = new Cricket();
        rep r1 = new rep();
        r1.generateReport(s1, cri);
      // 2nd report for student s2 who plays football
        Student s2 = new Student("Vivek", 47);
        Football f = new Football();
        rep r2 = new rep();
        r2.generateReport(s2, f);
        Student s3 = new Student("Bob" , 68);
        Hockey h = new Hockey();
        rep r3 = new rep();
        r3.generateReport(s3, h);
  }
}
```

OUTPUT:



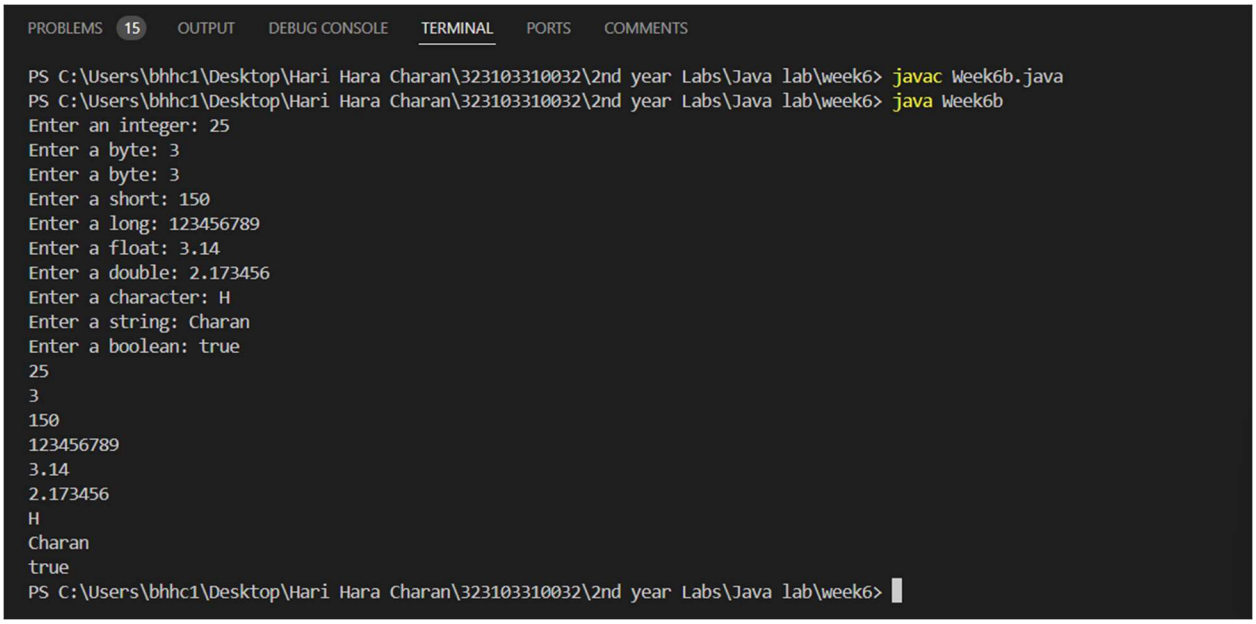
(B) AIM: Write a program that accepts values of different data types and convert them to corresponding wrapper classes and display using the vector

PROGRAM:

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

public class Week6b {
  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter an integer: ");
    Integer i = sc.nextInt();
    System.out.print("Enter a byte: ");
    Byte by = sc.nextByte();
    System.out.print("Enter a short: ");
    Short sh = sc.nextShort();
    System.out.print("Enter a long: ");
    Long l = sc.nextLong();
    System.out.print("Enter a float: ");
    Float f = sc.nextFloat();
    System.out.print("Enter a double: ");
    Double d = sc.nextDouble();
    System.out.print("Enter a character: ");
    Character c = sc.next().charAt(0);
    System.out.print("Enter a string: ");
    String s = sc.next();
    System.out.print("Enter a boolean: ");
    Boolean b = sc.nextBoolean();
    Vector<Object> v = new Vector<>();
```

```
        v.add(i);
        v.add(by);
        v.add(sh);
        v.add(l);
        v.add(f);
        v.add(d);
        v.add(c);
        v.add(s);
        v.add(b);
        for(Object item : v) {
          System.out.println(item);
        }
        sc.close();
    }
}
```

OUTPUT:



WEEK-7

(A) AIM: Write a program to generate a set of random numbers between two numbers x1 and x2, and x1>0.

PROGRAM:

```
import java.util.Random;
import java.util.Scanner;
public class Week7a {
  public static void main(String[] args) {
       Scanner sc = new Scanner(System.in);
       Random rand = new Random();
       System.out.println("enter the value of x1 (x1>0): ");
       int x1 = sc.nextInt();
       if(x1 < 0){
           System.out.println("x1 must be grater than 0");
       }
       System.out.println("enter the value of x2 : ");
       int x2 = sc.nextInt();
       if(x2 < x1){
           System.out.println("x2 must be grater than x1");
           sc.close();
           return;
       }
       System.out.println("how many random numbers do you want to generate:
");
       int count = sc.nextInt();
       System.out.print("Generate random numbers between "+ x1 + " and " + x2
+ ": ");
       for(int i = 0; i < count; i++) {
           int x = rand.nextInt(x2-x1+1) + x1;
           System.out.print(" " + x);
       }
       sc.close();
  }
}
```

OUTPUT:

```
PROBLEMS 15  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week7> javac Week7a.java
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week7> java Week7a
Enter the value of x1 (x1>0): 12
Enter the value of x2 : 8
x2 must be grater than x1
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week7>
```

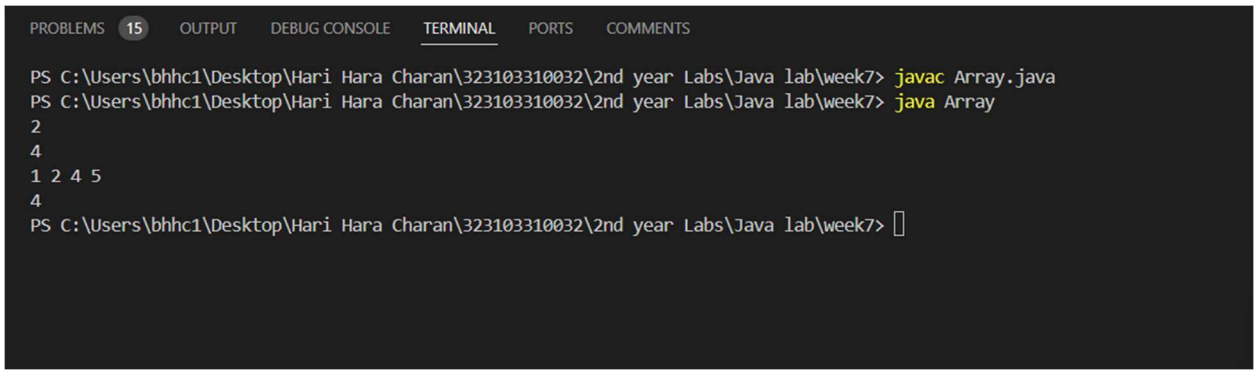
(B)AIM: Write a program to implement a new ArrayList class. It should contain add(), get(), remove(), size() methods. Use dynamic array logic.

PROGRAM:

```
public class Array {
    int[] array;
    int size;
    public Array() {
        array = new int[10];
        size = 0;
    }
    public void add(int x){
        if(size == array.length){
            resize();
        }
        array[size] = x;
        size++;
    }
    public int get(int index){
        if(index < 0 || index >= size){
            throw new ArrayIndexOutOfBoundsException("index out of bounds");
        }
        return array[index];
    }
    public void remove(int index){
        if(index < 0 || index >= size){
            throw new ArrayIndexOutOfBoundsException("index out of bounds");
        }
        for(int i = index; i < size-1; i++){
            array[i] = array[i+1];
        }
        size--;
    }
    public int size(){
        return size;
    }
    public void resize(){
        int[] newArray = new int[array.length*2];
        for(int i = 0; i < size; i++){
            newArray[i] = array[i];
        }
        array = newArray;
    }
    public void display(){
        for(int i = 0; i < size; i++){
            System.out.print(array[i] + " ");
        }
    }

    public static void main(String[] args) {
```

```
        Array list = new Array();
        list.add(1);
        list.add(2);
        list.add(3);
        list.add(4);
        list.add(5);
        System.out.println(list.get(1));
        list.remove(2);
        System.out.println(list.size());
        list.display();
        System.out.println();
        System.out.println(list.get(2));
    }
}
```

Output:



(C)AIM: Create an employee class containing at least 3 details along with Id, setters, and getters. Insert the employee objects dynamically key as employee id and value as it's corresponding object into a HashMap. Perform Id based search operation on the HashMap.

PROGRAM:

```
import java.util.HashMap;
import java.util.Scanner;

class Employee {
    private int id;
    private String name;
    private String department;
    private int salary;
    public Employee(int id, String name, String department, int salary) {
        this.id = id;
        this.name = name;
        this.department = department;
        this.salary = salary;
    }
```

```
    public int getId(){
        return id;
    }
    public void setId(int id) {
        if(id>0 || id<10){
            this.id = id;
        }
        else {
            System.out.println("Invalid ID");
        }
    }
    public String getName(){
        return name;
    }
    public void setName(String name){
        String[] names = name.split("[0123456789]");
        if(names.length == 1){
            this.name = name;
        }
        else{
            System.out.println("Invalid Name");
        }
    }
    public String getDepartment(){
        return department;
    }
    public void setDepartment(String department){
        String[] names = name.split("[0123456789]");
        if(names.length == 1){
            this.department = department;
        }
        else{
            System.out.println("Invalid Name");
        }
    }
    public int getSalary(){
        return salary;
    }
    public void setSalary(int salary){
        this.salary = salary;
    }

    public void display(){
        System.out.println("Employee ID: "+id);
        System.out.println("Employee Name: "+name);
        System.out.println("Employee Department: "+department);
        System.out.println("Employee Salary: "+salary);
    }

}
public class Week7c {
    public static void main(String[] args) {
        HashMap<Integer, Employee> employees = new HashMap<>();
```

```
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of employees: ");
        int n = sc.nextInt();
        for(int i = 0; i < n; i++) {
            System.out.println("enter details for employee #" + (i + 1));
            System.out.println("Id: ");
            int id = sc.nextInt();
            System.out.println("Name: ");
            String name = sc.next();
            System.out.println("Department: ");
            String department = sc.next();
            System.out.println("Salary: ");
            int salary = sc.nextInt();
            Employee employee = new Employee(id, name, department, salary);
            employees.put(id, employee);
        }
        System.out.println("Enter the employee ID: ");
        int searchId = sc.nextInt();
        if(employees.containsKey(searchId)){
            System.out.println("Employee found");
            Employee found = employees.get(searchId);
            found.display();
        }
        else{
            System.out.println("Employee not found");
        }
        sc.close();
        }
    }
```

OUTPUT:

```
PROBLEMS 14  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week7> javac Week7c.java
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week7> java Week7c
Enter the number of employees: 3
Enter details for employee 1
Id: 1
Name: Charan
Department: CSE
Salary: 30000
Salary: 30000
Enter details for employee 2
Id: 5
Name: Bob
Department: Chemical
Salary: 50000
Enter details for employee 3
Id: 9
Name: Fread
Department: IT
Salary: 300000
Enter the employee ID: 1
Employee found
Employee ID: 1
Employee Name: Charan
Employee Department: CSE
Employee Salary: 30000
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week7>
```

AIM: Write a program that reads file name from the user then displays information about that file, also read the contents from the file in byte stream to count the number of alphabets, numeric values, and special symbols. Write these statistics into another file using byte streams

PROGRAM:

```
import java.util.Scanner;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
public class FileAnalyzer {
  public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter path of the file to be analyzed: ");
      String filename = sc.nextLine();
      File file = new File(filename);
      if (file.exists()) {
          System.out.println("File name "+file.getName());
          System.out.println("absolute path "+file.getAbsolutePath());
          System.out.println("writeable "+file.canWrite());
          System.out.println("readable "+file.canRead());
          System.out.println("file size in bytes "+file.length());
      }
      else {
          System.out.println("File not found");
      }
      int alphabets = 0;
      int digits = 0;
      int special = 0;

      try(FileInputStream fis = new FileInputStream(filename)){
          int bytedata;
          while((bytedata = fis.read())!=-1){
              if((bytedata >= 'A' && bytedata <= 'Z') || (bytedata >= 'a' &&
bytedata <= 'z')){
                  alphabets++;
              }
              else if((bytedata >= '0' && bytedata <= '9')){
                  digits++;
              }
              else{
                  special++;
              }
          }
      }
      catch(IOException e){
          System.out.println(e.getMessage());
      }
      System.out.println("Alphabets: " + alphabets);
```

```
        System.out.println("Digits: " + digits);
        System.out.println("Special: " + special);

        String outputfilename = "example1.txt";
        try (FileOutputStream fos = new FileOutputStream(outputfilename)) {
            String result = "Alphabets: " + alphabets + "\n" +
                    "Digits: " + digits + "\n" +
                    "Special Characters: " + special + "\n";
            fos.write(result.getBytes());
            System.out.println("Statistics written to " + outputfilename);
        } catch (IOException e) {
            System.out.println("Error writing to the file: " + e.getMessage());
        }

        sc.close();

    }
}
```

OUTPUT:

```
PROBLEMS 14  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week8> javac FileAnalyzer.java
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week8> java FileAnalyzer
Enter path of the file to be analyzed: superMarket.csv
File name superMarket.csv
absolute path C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week8\superMarket.csv
writeable true
readable true
file size in bytes 124
Alphabets: 54
Digits: 35
Special: 35
Statistics written to example1.txt
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\Java lab\week8>
```