Exp – 5

Ques 1. Write a Java program to calculate the sum of a list of integers using autoboxing and unboxing. Include methods to parse strings into their respective wrapper classes (e.g., Integer.parseInt()).

Code:

Ayush_S_Nair_22BCS14823.java > ...

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

public class Ayush_S_Nair_22BCS14823 {
    Run | Debug
    public static void main(String[] args) {
        String[] inputNumbers = {"5", "15", "25", "35", "45"};
        List<Integer> intList = convertToIntegerList(inputNumbers);

        int totalSum = computeSum(intList);

        System.out.println("Total Sum: " + totalSum);
    }

    public static List<Integer> convertToIntegerList(String[] inputNumbers) {
        List<Integer> resultList = new ArrayList<>();
        for (String number : inputNumbers) {
            resultList.add(Integer.valueOf(number));
        }
        return resultList;
    }

    public static int computeSum(List<Integer> intList) {
        int sum = 0;
        for (Integer value : intList) {
            sum += value;
        }
        return sum;
    }
}

```

Output:

```
PS D:\WebD> cd "d:\WebD\"
Total Sum: 125
PS D:\WebD>
```

Ques 2. Create a Java program to serialize and deserialize a Student object. The program should: Serialize a Student object (containing id, name, and GPA) and save it to a file. Deserialize the object from the file and display the student details. Handle FileNotFoundException, IOException, and ClassNotFoundException using exception handling.

Code:

```java
import java.io.*;

class Student implements Serializable {
    private int id;
    private String name;
    private double gpa;

    public Student(int id, String name, double gpa) {
        this.id = id;
        this.name = name;
        this.gpa = gpa;
    }

    public void display() {
        System.out.println("ID: " + id);
        System.out.println("Name: " + name);
        System.out.println("GPA: " + gpa);
    }
}

public class Ayush_S_Nair_22BCS14823_2 {
    Run | Debug
    public static void main(String[] args) {
        Student student = new Student(id:101, name:"Ayush S Nair", gpa:9.9);
        String filename = "student.ser";

        try (ObjectOutputStream out = new ObjectOutputStream(new FileOutputStream(filename))) {
            out.writeObject(student);
            System.out.println(x:"Student object serialized.");
        } catch (IOException e) {
            System.out.println("Error during serialization: " + e.getMessage());
        }

        try (ObjectInputStream in = new ObjectInputStream(new FileInputStream(filename))) {
            Student deserializedStudent = (Student) in.readObject();
            System.out.println(x:"Deserialized Student:");
            deserializedStudent.display();
        } catch (IOException | ClassNotFoundException e) {
            System.out.println("Error during deserialization: " + e.getMessage());
        }
    }
}
```

Output:

```
PS D:\WebD> cd "d:\WebD\" ;
Student object serialized.
Deserialized Student:
ID: 101
Name: Ayush S Nair
GPA: 9.9
PS D:\WebD>
```

Ques 3. Create a menu-based Java application with the following options. 1.Add an Employee 2. Display All 3. Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id,

designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.

Code:

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

class Employee implements Serializable {
    private int id;
    private String name;
    private String designation;
    private double salary;

    public Employee(int id, String name, String designation, double salary) {
        this.id = id;
        this.name = name;
        this.designation = designation;
        this.salary = salary;
    }

    public void display() {
        System.out.println("ID: " + id);
        System.out.println("Name: " + name);
        System.out.println("Designation: " + designation);
        System.out.println("Salary: " + salary);
        System.out.println(x:"----------------------");
    }
}

public class Ayush_S_Nair_22BCS14823_3 {
    private static final String FILE_NAME = "employees.dat";

    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Employee> employees = loadEmployees();
```

```
        while (true) {
            System.out.println(x:"1. Add Employee");
            System.out.println(x:"2. Display All");
            System.out.println(x:"3. Exit");
            System.out.print(s:"Choose an option: ");
            int choice = scanner.nextInt();
            scanner.nextLine();

            if (choice == 1) {
                System.out.print(s:"Enter ID: ");
                int id = scanner.nextInt();
                scanner.nextLine();
                System.out.print(s:"Enter Name: ");
                String name = scanner.nextLine();
                System.out.print(s:"Enter Designation: ");
                String designation = scanner.nextLine();
                System.out.print(s:"Enter Salary: ");
                double salary = scanner.nextDouble();
                scanner.nextLine();

                employees.add(new Employee(id, name, designation, salary));
                saveEmployees(employees);
                System.out.println(x:"Employee added successfully.");
            } else if (choice == 2) {
                for (Employee emp : employees) {
                    emp.display();
                }
            } else if (choice == 3) {
                System.out.println(x:"Exiting...");
                break;
```

```java
            } else if (choice == 3) {
                System.out.println(x:"Exiting...");
                break;
            } else {
                System.out.println(x:"Invalid option. Try again.");
            }
        }
        scanner.close();
    }

    private static List<Employee> loadEmployees() {
        try (ObjectInputStream in = new ObjectInputStream(new FileInputStream(FILE_NAME))) {
            return (List<Employee>) in.readObject();
        } catch (IOException | ClassNotFoundException e) {
            return new ArrayList<>();
        }
    }

    private static void saveEmployees(List<Employee> employees) {
        try (ObjectOutputStream out = new ObjectOutputStream(new FileOutputStream(FILE_NAME))) {
            out.writeObject(employees);
        } catch (IOException e) {
            System.out.println("Error saving employees: " + e.getMessage());
        }
    }
}
```

Output:

```
PS D:\WebD> cd "d:\WebD\" ; if ($?) { javac Ayush_S_Nair_22BCS14823_3.java
Note: Ayush_S_Nair_22BCS14823_3.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
1. Add Employee
2. Display All
3. Exit
Choose an option: 1
Enter ID: 12
Enter Name: Rajan
Enter Designation: Associate Software Engineer
Enter Salary: 1000000
Employee added successfully.
1. Add Employee
2. Display All
3. Exit
Choose an option: 1
Enter ID: 15
Enter Name: Ayush
Enter Designation: Data Analyst
Enter Salary: 1500000
Employee added successfully.
1. Add Employee
2. Display All
3. Exit
```

```
Choose an option: 2
ID: 12
Name: Rajan
Designation: Software Engineer
Salary: 1000000.0
----------------------
ID: 12
Name: Rajan
Designation: Associate Software Engineer
Salary: 1000000.0
----------------------
ID: 15
Name: Ayush
Designation: Data Analyst
Salary: 1500000.0
----------------------
1. Add Employee
2. Display All
3. Exit
Choose an option: 3
Salary: 1500000.0
----------------------
1. Add Employee
2. Display All
3. Exit
Choose an option: 3
Exiting...
PS D:\WebD>
```