CU
CHANDIGARH
UNIVERSITY

CHANDIGARH
UNIVERSITY
Discover. Learn. Empower.

Experiment 2.4

Student Name: Lipakshi	**UID: 20BCS5082**
Branch: CSE	**Section/Group: 607-B**
Semester:5th	**Subject Name: PBLJ Lab**

1. **Aim/Overview of the practical:** 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.

2. **Software/Hardware Requirements:** Eclipse IDE-(Java), Netbeans

3. **Steps for experiment/practical/Code:**

```
import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.ObjectInputStream;
```

```java
import java.io.ObjectOutputStream;

import java.io.Serializable;

import java.util.ArrayList;

import java.util.Scanner;


@SuppressWarnings("serial")
class Employee implements Serializable{

    int id;

    int age;

    String name;

    float salary;

    long contact_no;

    String email_id;

    public Employee(int id, String name, float salary,int age)
    {
        this.id = id;

        this.name = name;

        this.salary = salary;

        this.age=age;

    }
    public String toString()
```

```java
        {
return "\nEmployee Details :" + "\nID: " + this.id + "\nName: " + this.name + "\nSalary: " +
                this.salary + "\nAge: "+this.age;

        }


    }
    public class main
    {
        static void display(ArrayList<Employee> al)
        {
            System.out.println("\n-------------Employee List--------------\n");
            System.out.println(String.format("%-10s%-15s%-10s%-20s",
    "ID","Name","Salary","Age"));
            for(Employee e : al)
            {

System.out.println(String.format("%-5s%-20s%-10s%-15s",e.id,e.name,e.salary,e.age));
            }
        }
        @SuppressWarnings("unchecked")
        public static void main(String[] args)
        {
```

```java
int id;
    String name;
    float salary;
    int age;
    Scanner sc = new Scanner(System.in);
    ArrayList<Employee> al = new ArrayList<Employee>();
    File f = null;
    FileInputStream fis = null;
    ObjectInputStream ois = null;
    FileOutputStream fos = null;
    ObjectOutputStream oos =null;
    try{
        f = new File("N:/Java Work Space/Eclipse Programs/Employee
Management Tool/src/EmployeeDataList1.txt");
        if(f.exists())
        {
            fis = new FileInputStream(f);
            ois = new ObjectInputStream(fis);
            al = (ArrayList<Employee>)ois.readObject();
        }
```

```java
        }
    catch(Exception exp){
            System.out.println(exp);

        }
        do
        {
            System.out.println("\n***Welcome to the Employee Management
System****\n");
            System.out.println("1). Add Employee \n" +


                "2). Display all Employees working in this company\n" +
                "3). EXIT\n");
        System.out.println("Enter your choice : ");
        int ch = sc.nextInt();
        switch(ch)
        {
            case 1:System.out.println("\nEnter the following details to ADD list:\n");
                System.out.println("Enter ID :");
                id = sc.nextInt();
                System.out.println("Enter Name :");
                name = sc.next();
                System.out.println("Enter Salary :");
```

```java
salary = sc.nextFloat();
                    System.out.println("Enter Age :");
                    age = sc.nextInt();
                    al.add(new Employee(id, name, salary, age));
                    display(al);
                    break;
                case 2: try {
                    al = (ArrayList<Employee>)ois.readObject();



            } catch (ClassNotFoundException e2) {
                System.out.println(e2);
            } catch (Exception e2) {
                System.out.println(e2);
            }
                display(al);
                break;
            case 3: try {
```

```java
        fos = new FileOutputStream(f);

                oos = new ObjectOutputStream(fos);

                oos.writeObject(al);

            } catch (IOException e1) {

    e1.printStackTrace();


        }

            catch(Exception e2){

                e2.printStackTrace();

        }

            finally{

                try {

                    fis.close();

                    ois.close();

                    fos.close();

                    oos.close();

                } catch (Exception e1) {

                    e1.printStackTrace();

                }

            }

                System.out.println("\nYou have chosen EXIT !! Saving Files and
    closing the tool.");
```

```
sc.close();

                System.exit(0);

                break;


    default : System.out.println("\nEnter a correct choice from the List :");

                break;

        }

    }

    while(true);

    }

}
```

4. Result/Output/Writing Summary:

```
***Welcome to the Employee Management System****

1). Add Employee
2). Display all Employees working in this company
3). EXIT


Enter your choice :
1


Enter the following details to ADD list:


Enter ID :
1002
Enter Name :
Aman
Enter Salary :
3000000
Enter Age :
23


--------------Employee List---------------


ID          Name              Salary    Age
1001 Ajay                     150000.0  26
1002 Aman                     3000000.0 23
```

```
--------------Employee List---------------

ID          Name              Salary    Age
1001 Ajay                     150000.0  26
1002 Aman                     3000000.0 23


***Welcome to the Employee Management System****

1). Add Employee
2). Display all Employees working in this company
3). EXIT


Enter your choice :
3
```

```
You have chosen EXIT !! Saving Files and closing the tool.

Process finished with exit code 0
```

Learning outcomes (What I have learnt):

1. I learnt the OOPS concepts of java thoroughly.

2. Got to know more about the control flow of the Code.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			