**CHANDIGARH UNIVERSITY**

**UNIVERSITY INSTITUTE OF ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



|  |  |
| --- | --- |
| **Submitted By:** Sahil Kaundal  **Submitted To:** Neeru Sharma | |
| **Subject Name** | Project Based Learning Java (Lab) |
| **Subject Code** | 20CSP-321 |
| **Branch** | Computer Science Engineering |
| **Semester** | 5th |

LAB INDEX

**NAME:** Sahil Kaundal **SUBJECT NAME:** PBLJ (Lab)

**UID:** 21BCS8197 **SUBJECT CODE:** 20CSP-321

**SECTION:** 20BCS\_WM-616/A

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr.No** | **Program** | **Date** | **Evaluation** | | | | **Sign** |
| **LW**  **(12)** | **VV**  **(10)** | **FW**  **(8)** | **Total**  **(30)** |
| 1. | Create an application to save the employee information using arrays. | 16/08/2022 |  |  |  |  |  |
| 2. | Design and implement a simple inventory control system for a small video rental store. | 20/08/2022 |  |  |  |  |  |
| 3. | Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance. | 27/08/2022 |  |  |  |  |  |
| 4. | Create a program to set view of Keys from Java Hashtable. | 27/09/2022 |  |  |  |  |  |
| 5. | Create a program to show the usage of Sets of Collection interface. | 27/09/2022 |  |  |  |  |  |
| 6. | Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed. | 04/10/2022 |  |  |  |  |  |
| 7. | Create a menu based Java application with the following options.1.Add an Employee2.Display All3.Exit | 13/10/2022 |  |  |  |  |  |

**Experiment 7**

**Student Name:** Sahil Kaundal **UID:** 21BCS8197

**Branch:** BE CSE (Lateral Entry) **Section/Group:** 616/A

**Semester:** 5th **Date of Performance:** 13/10/2022

**Subject Name:** PBLJ Lab **Subject Code:** 20CSP-321

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

Create a menu based Java application with the following options.1.Add an Employee2.Display All3.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.

1. **Task to be done/ Which logistics used:**

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.

1. **Apparatus / Simulator Used:**

* Eclipse IDE - (Java)
* NetBeans.
* JDK-8 or any.

1. **Programs/ Code:**

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.io.ObjectOutputStream;

import java.io.Serializable;

import java.util.ArrayList;

import java.util.Scanner;

@SuppressWarnings("serial")

class Employee implements Serializable{

int id;

String name;

float salary;

long contact\_no;

int email\_id;

public Employee(int id, String name, float salary, long contact\_no, int email\_id)

{

this.id = id;

this.name = name;

this.salary = salary;

this.contact\_no = contact\_no;

this.email\_id = email\_id;

}

public String toString()

{

return "\nEmployee Details :" + "\nID: " + this.id + "\nName: " + this.name + "\nSalary: " +

this.salary + "\nContact No: " + this.contact\_no + "\nEmail-id: " + this.email\_id;

}

}

public class Exp7

{

static void display(ArrayList<Employee> al)

{

System.out.println("\n--------------Details---------------\n");

System.out.println(String.format("%-10s%-15s%-10s%-20s%-10s", "ID","Name","Salary","Contact","Age"));

for(Employee e : al)

{

System.out.println(String.format("%-5s%-20s%-10s%-15s%-10s",e.id,e.name,e.salary,e.contact\_no,e.email\_id));

}

}

@SuppressWarnings("unchecked")

public static void main(String[] args)

{

int id;

String name;

float salary;

long contact\_no;

int email\_id;

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();

}

}

catch(Exception exp){

System.out.println(exp);

}

do

{

System.out.println("1). Add Employee \n" +

"2). Search Employee \n" +

"3). Edit Employee Details \n" +

"4). Delete Employee Details \n" +

"5). Display All Employees \n" +

"6). EXIT \n");

System.out.println("Enter Your Choice : ");

int ch = sc.nextInt();

switch(ch)

{

case 1:System.out.println("\nEnter Details: \n");

System.out.println("Enter ID :");

id = sc.nextInt();

System.out.println("Enter Name :");

name = sc.next();

System.out.println("Enter Salary :");

salary = sc.nextFloat();

System.out.println("Enter Contact :");

contact\_no = sc.nextLong();

System.out.println("Enter Age :");

email\_id = sc.nextInt();

al.add(new Employee(id, name, salary, contact\_no, email\_id));

display(al);

break;

case 2: System.out.println("Enter ID to Search Employee :");

id = sc.nextInt();

int i=0;

for(Employee e: al)

{

if(id == e.id)

{

System.out.println(e+"\n");

i++;

}

}

if(i == 0)

{

System.out.println("\nEmployee not available, Please enter a valid ID!!");

}

break;

case 3: System.out.println("\nEnter ID to EDIT the Details");

id = sc.nextInt();

int j=0;

for(Employee e: al)

{

if(id == e.id)

{

j++;

do{

int ch1 =0;

System.out.println("\nEDIT Employee Details :\n" +

"1). Employee ID \n" +

"2). Name \n" +

"3). Salary \n" +

"4). Contact \n" +

"5). Age \n" +

"6). GO BACK\n");

System.out.println("Enter Your Choice : ");

ch1 = sc.nextInt();

switch(ch1)

{

case 1: System.out.println("\nEnter new Employee ID:");

e.id =sc.nextInt();

System.out.println(e+"\n");

break;

case 2: System.out.println("Enter new Employee Name:");

e.name =sc.nextLine();

System.out.println(e+"\n");

break;

case 3: System.out.println("Enter new Employee Salary:");

e.salary =sc.nextFloat();

System.out.println(e+"\n");

break;

case 4: System.out.println("Enter new Employee Contact No. :");

e.contact\_no =sc.nextLong();

System.out.println(e+"\n");

break;

case 5: System.out.println("Enter new Employee Age :");

e.email\_id =sc.nextInt();

System.out.println(e+"\n");

break;

case 6: j++;

break;

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

break;

}

}

while(j==1);

}

}

if(j == 0)

{

System.out.println("\nEmployee Details are not available, Please enter a valid ID!!");

}

break;

case 4: System.out.println("\nEnter Employee ID to DELETE: ");

id = sc.nextInt();

int k=0;

try{

for(Employee e: al)

{

if(id == e.id)

{

al.remove(e);

display(al);

k++;

}

}

if(k == 0)

{

System.out.println("\nEmployee Details are not available, Please enter a valid ID!!");

}

}

catch(Exception ex){

System.out.println(ex);

}

break;

case 5: try {

al = (ArrayList<Employee>)ois.readObject();

} catch (ClassNotFoundException e2) {

System.out.println(e2);

} catch (Exception e2) {

System.out.println(e2);

}

display(al);

break;

case 6: try {

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("\n Thanks For Using This Application");

sc.close();

System.exit(0);

break;

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

break;

}

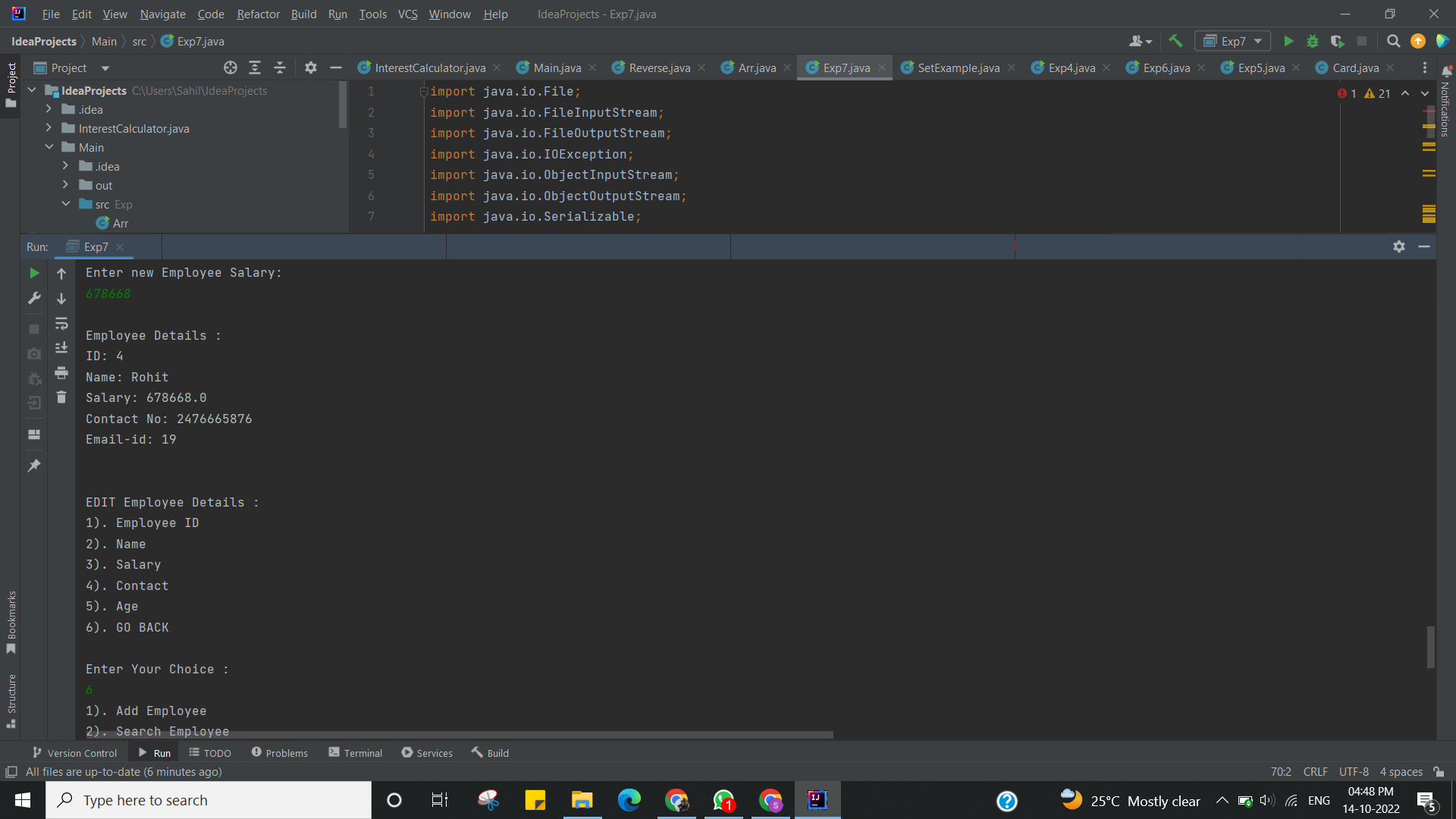
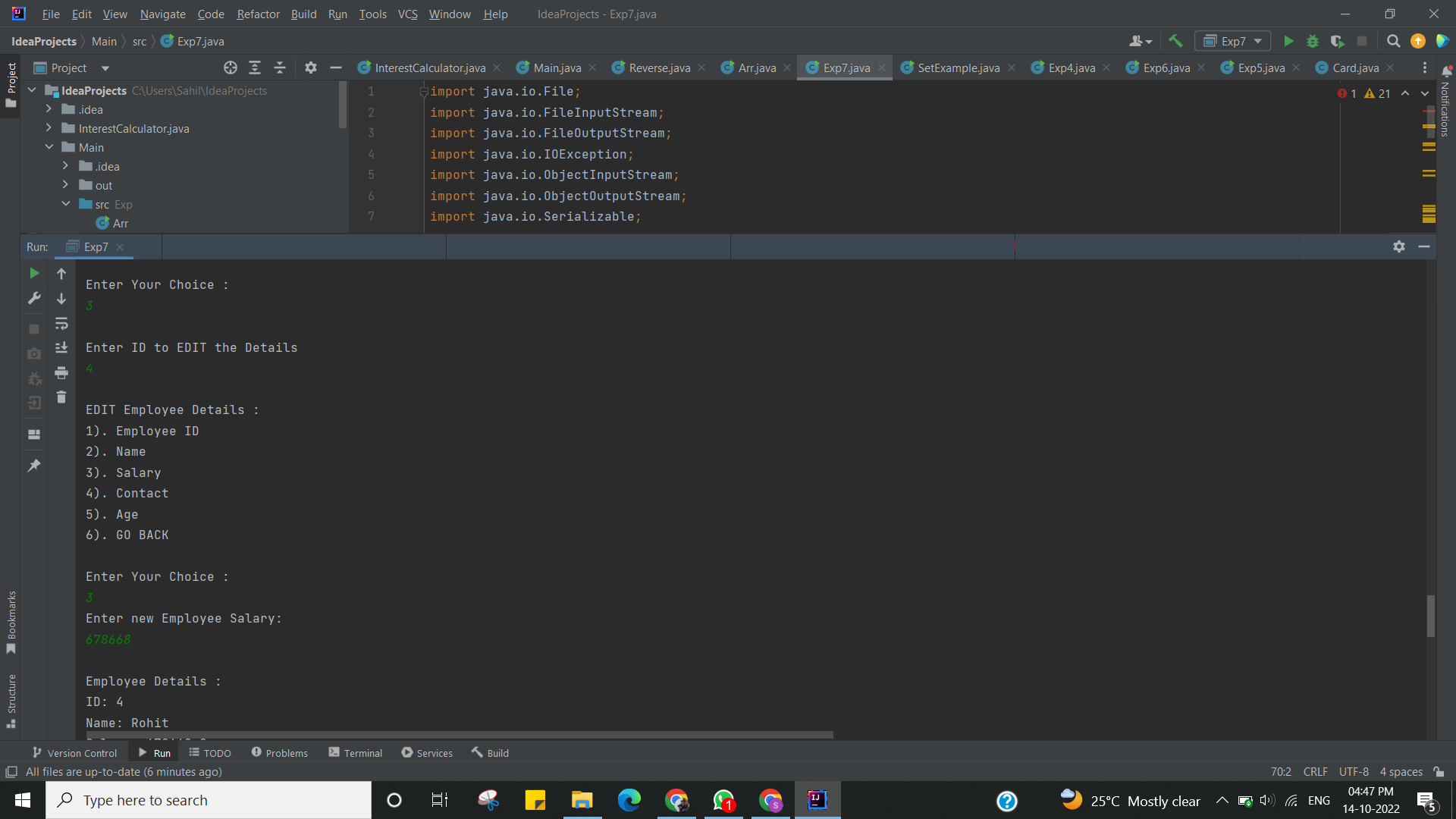
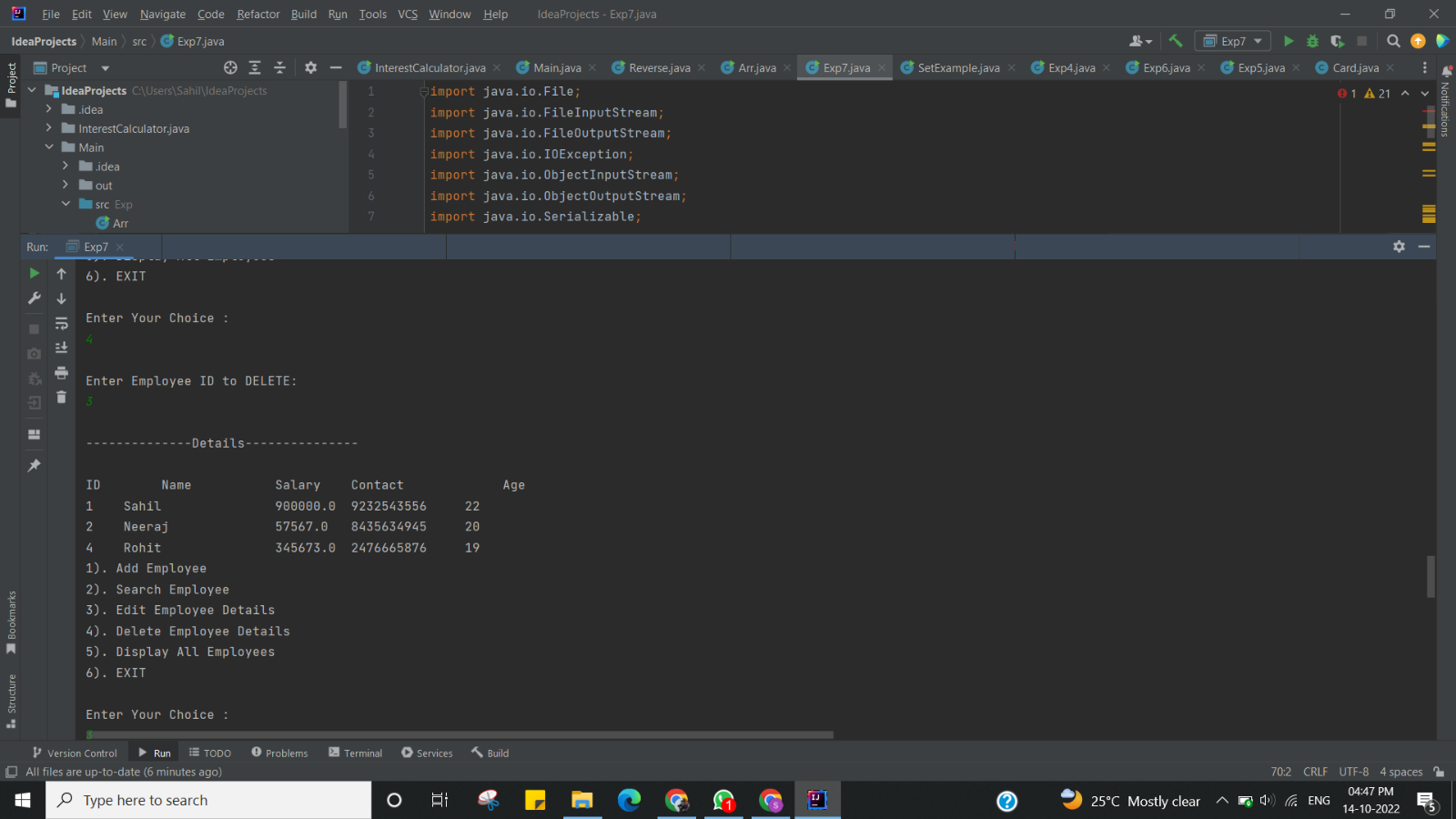
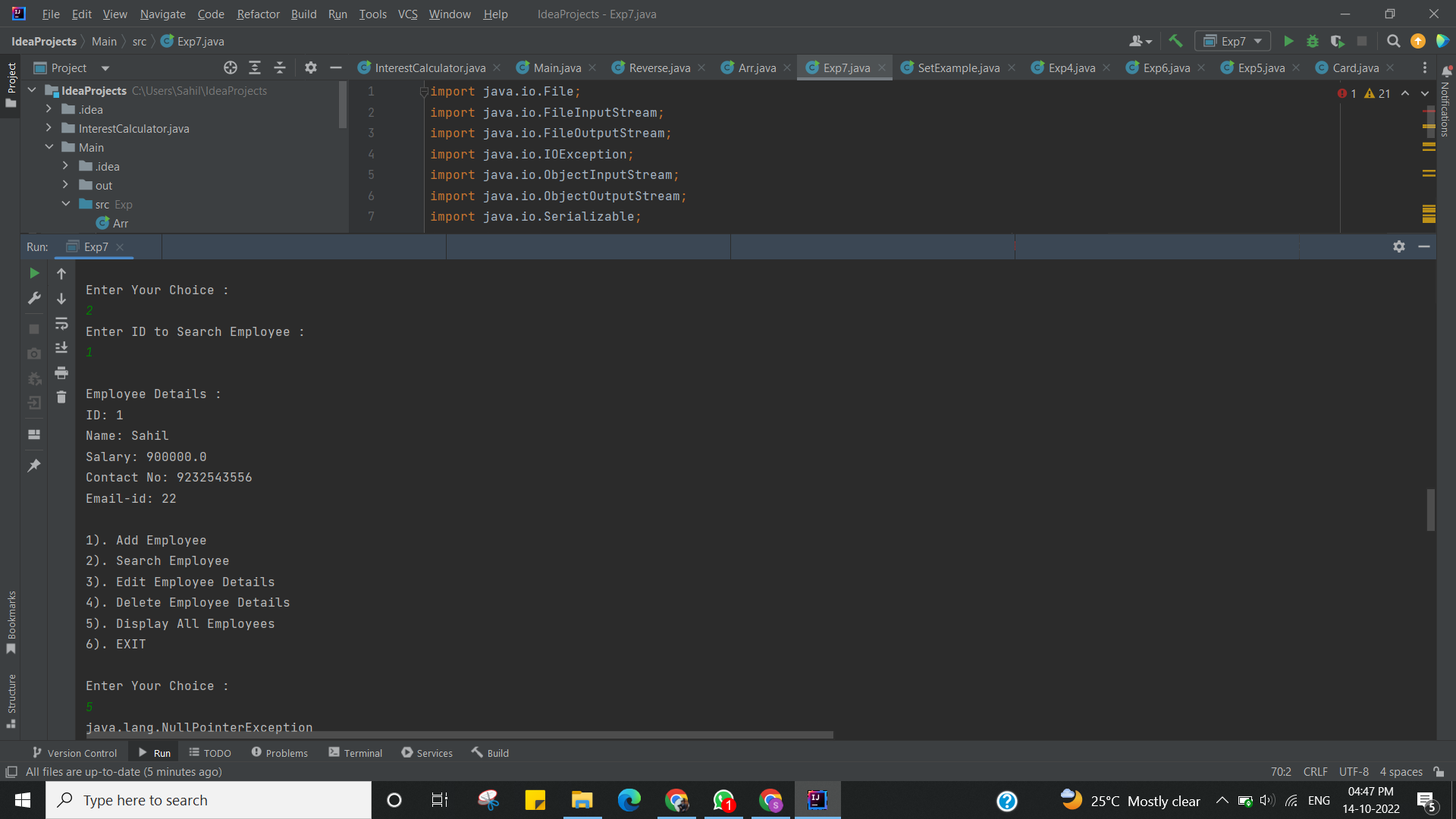
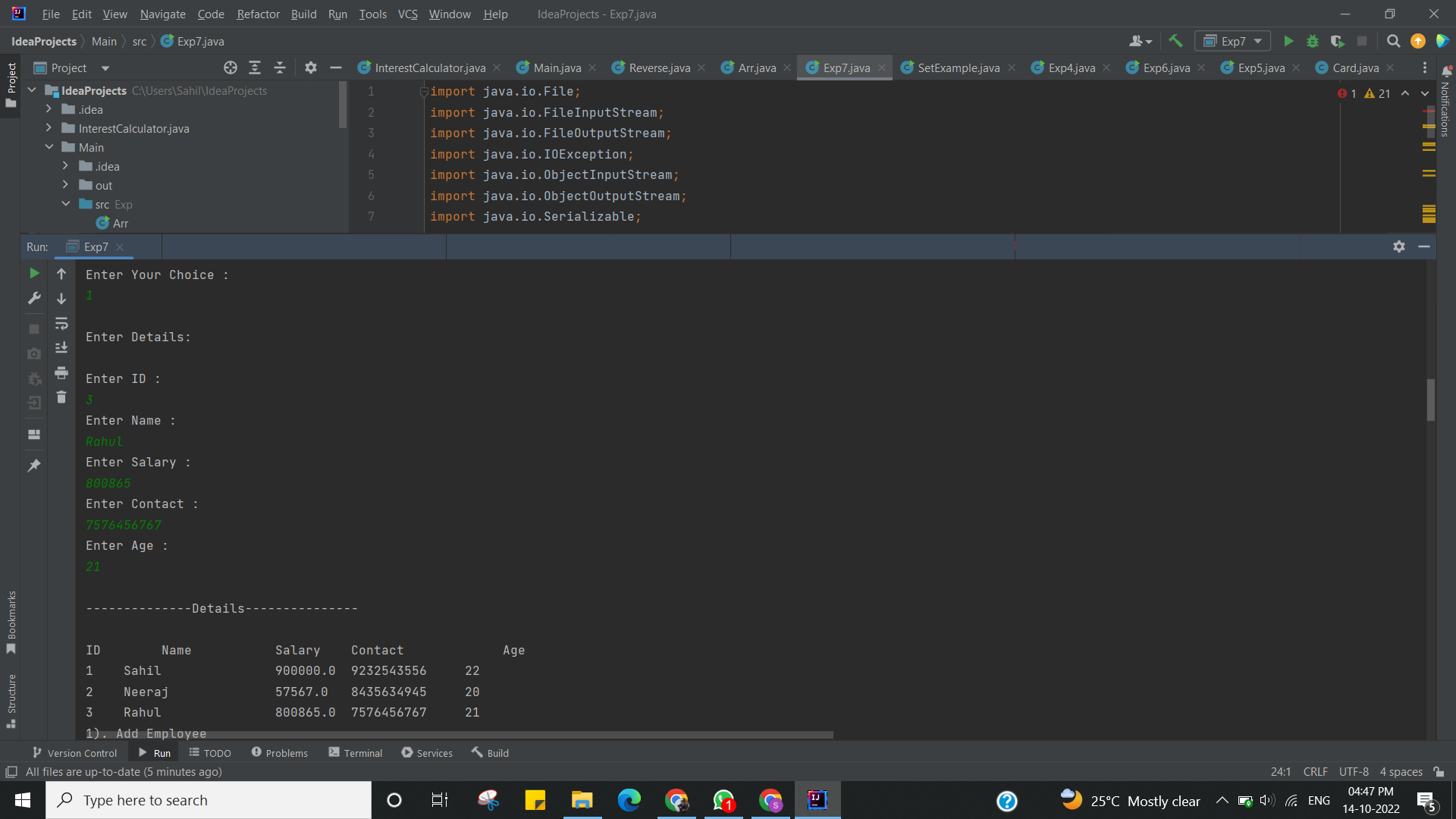
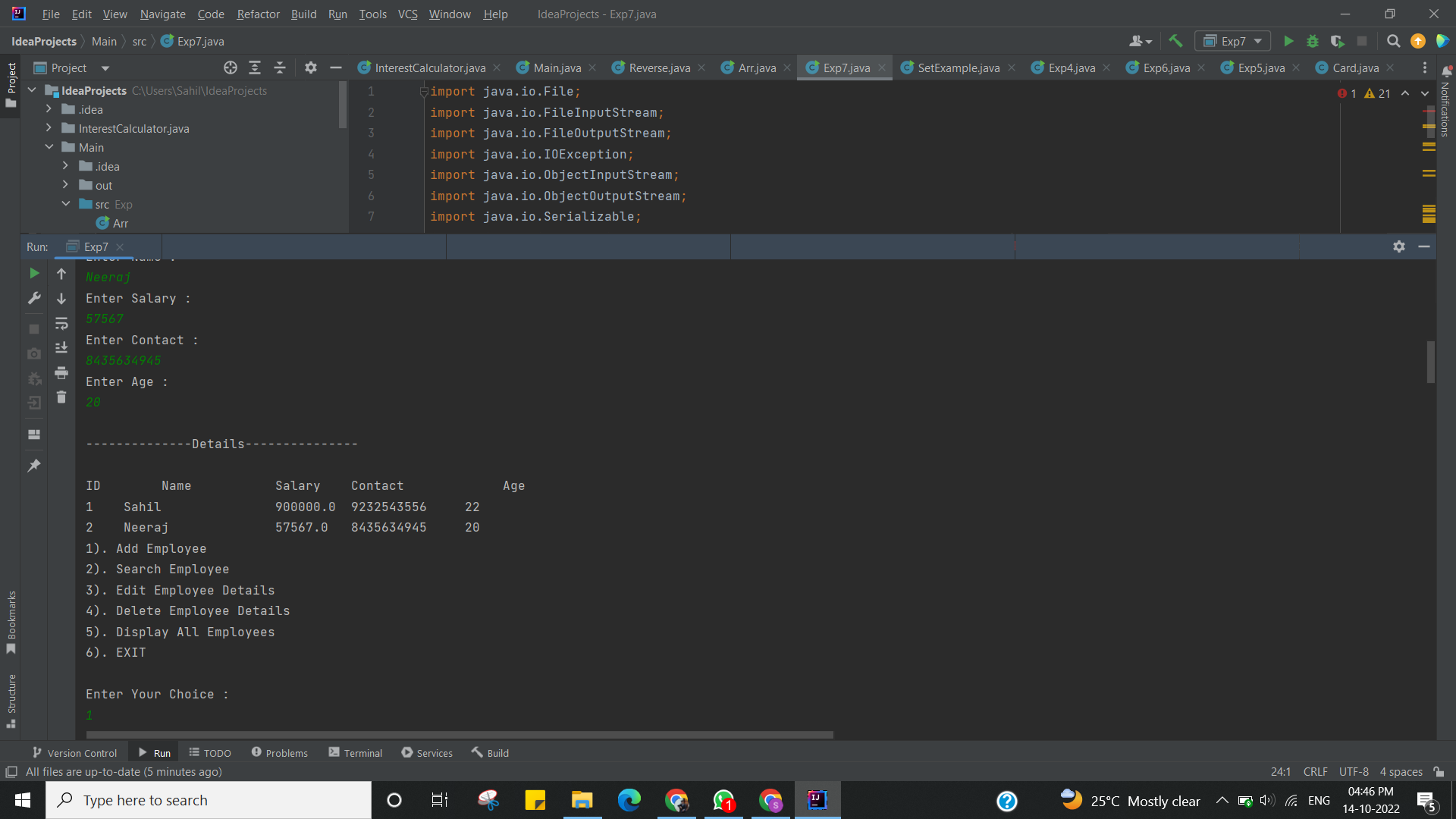
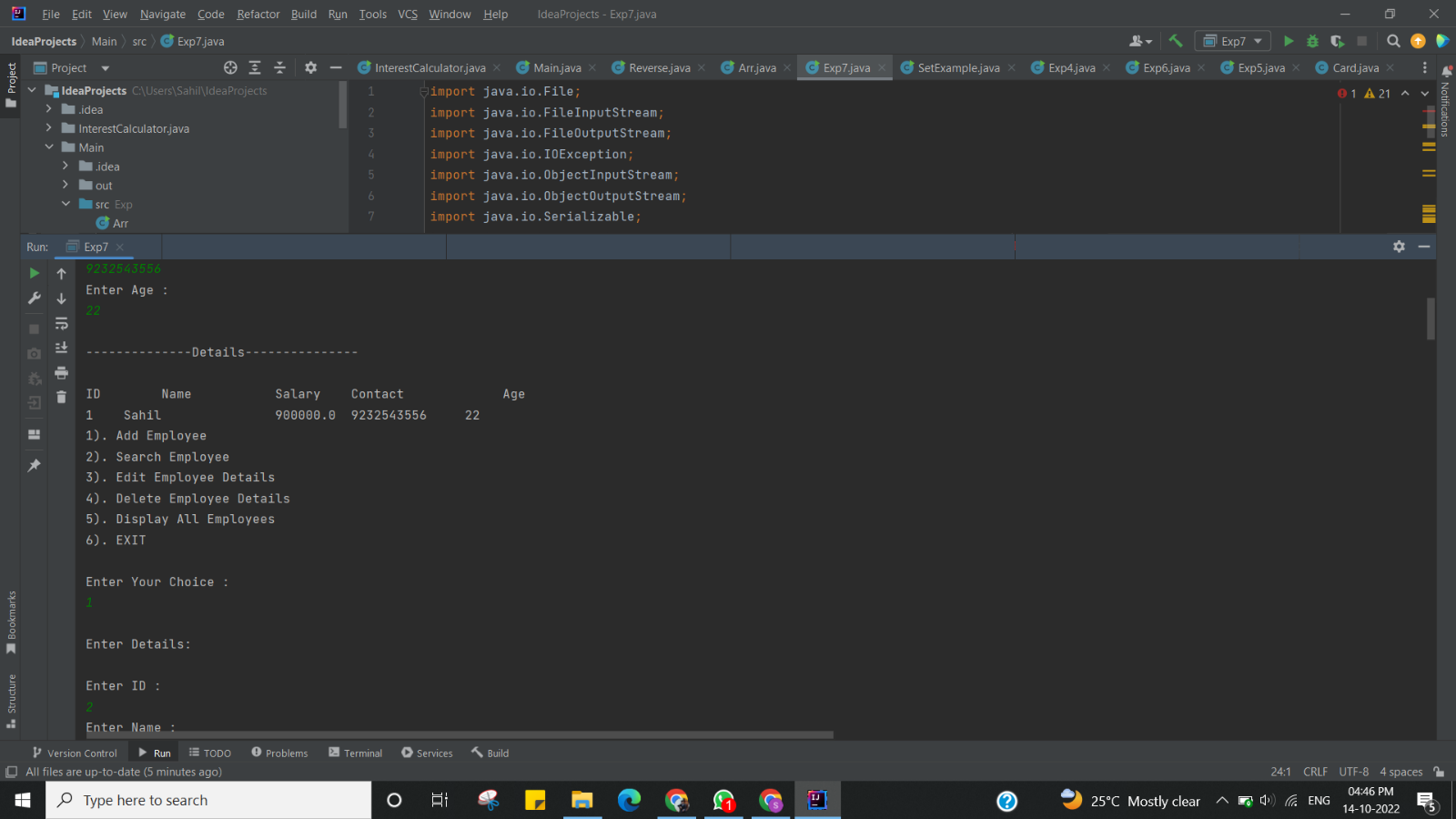
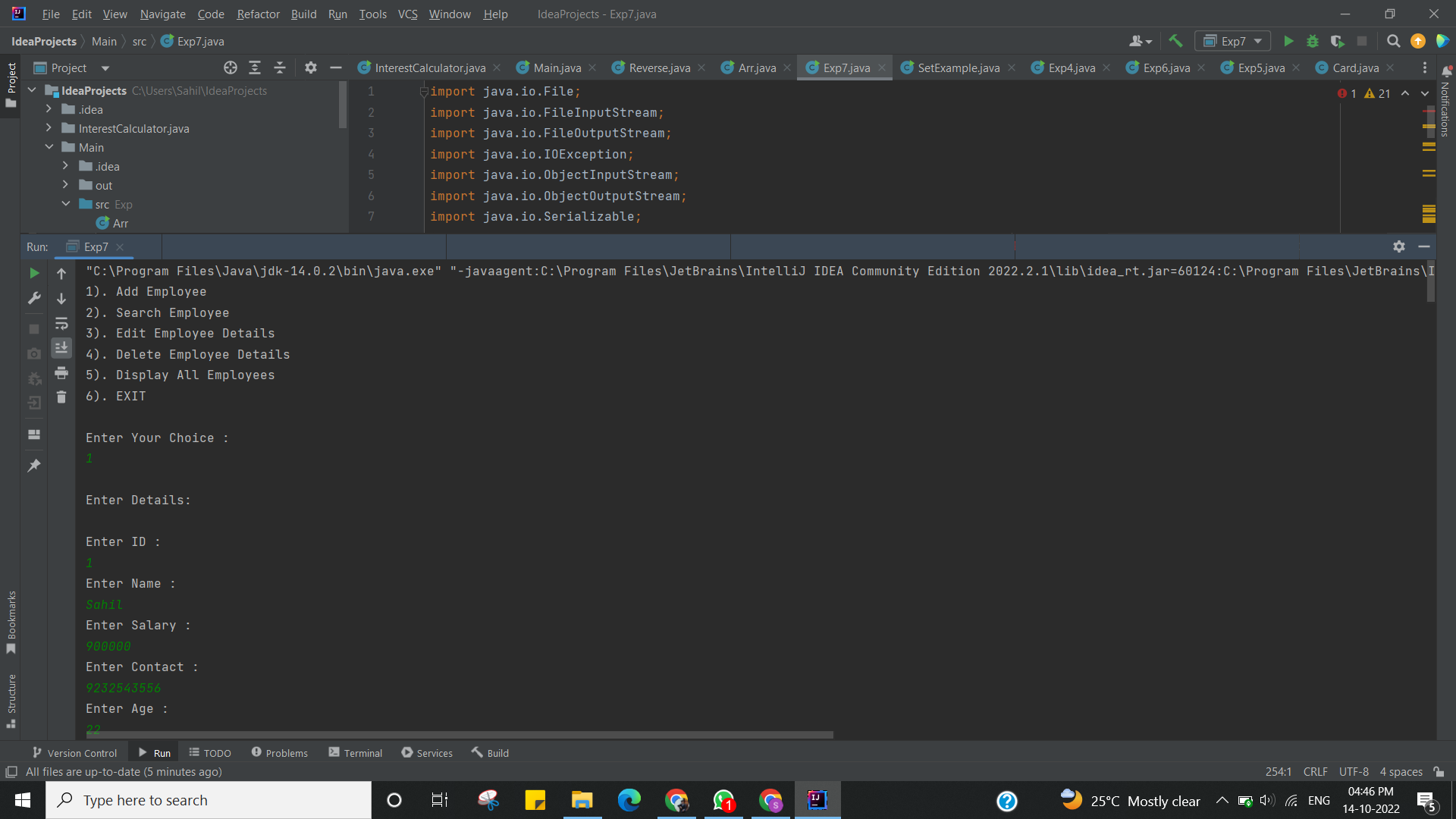
}

while(true);

}

}

**5. Result/Output/Writing Summary:**



I have successfully done this program.

**Learning Outcomes (What I have learnt):**

* Learnt while loop.
* Learnt the concept of switch concept.
* Learnt Switch case
* Learnt Database connectivity.
* Learnt how to write menu-driven program.

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

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |
| 2. |  |  |  |