

**MAHADI HASAN MISHUK**

21 SOECE 11649

CE-D 48

**Linkedin**: [shorturl.at/kprvX](http://shorturl.at/kprvX) **Github**:[shorturl.at/ESWY8](http://shorturl.at/ESWY8)

-------------------------------------------------------------------------------------------------------------------------------

**Experiment - 2**

1. Implement Java program to create a class Student with data ‘name, city and age’ along with the method printData to display the data. Create the two objects s1 ,s2 to declare and access the values.

Code:

class Student

{

String name, city;

int age;

static int m;

void printData()

{

System.out.println("Student name = "+name);

System.out.println("Student city = "+city);

System.out.println("Student age = "+age);

}

}

class test

{

public static void main(String args[])

{

Student s1=new Student();

Student s2=new Student();

s1.name="Amit";

s1.city="Dehradun";

s1.age=22;

s2.name="Kapil";

s2.city="Delhi";

s2.age=23;

s2.printData();

s1.printData();

s1.m=20;

s2.m=22;

Student.m=27;

System.out.println("s1.m = "+s1.m);

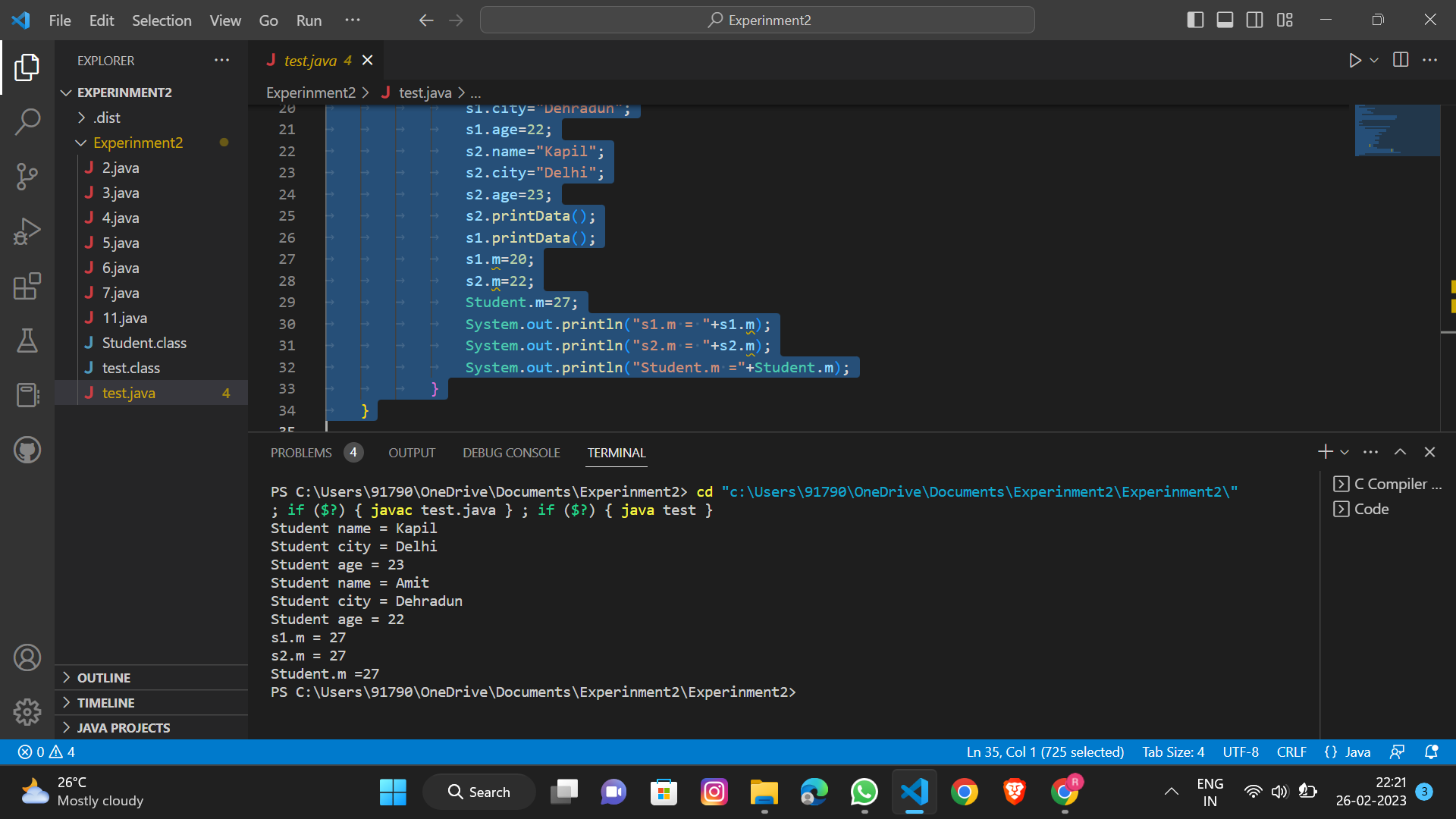
System.out.println("s2.m = "+s2.m);

System.out.println("Student.m ="+Student.m);

}

}

Output:



2. Implement a java program to create a class Employee display the employee details such as employee id, employee name, salary, age using Scanner class. Create Employee class to object and display employee details.

Code:

import java.util.Scanner;

class Employee

{

int empId;

String empName;

double salary;

int age;

public Employee(int empId, String empName, double salary, int age)

{

this.empId = empId;

this.empName = empName;

this.salary = salary;

this.age = age;

}

public void displayDetails()

{

System.out.println("Employee ID: " + empId);

System.out.println("Employee Name: " + empName);

System.out.println("Salary: " + salary);

System.out.println("Age: " + age);

}

}

class EmployeeTest

{

public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter employee ID: ");

int empId = scanner.nextInt();

System.out.print("Enter employee name: ");

String empName = scanner.next();

System.out.print("Enter employee salary: ");

double salary = scanner.nextDouble();

System.out.print("Enter employee age: ");

int age = scanner.nextInt();

Employee emp = new Employee(empId, empName, salary, age);

emp.displayDetails();

}

}

Output:

3. Design a class to represent a bank account. Which contains account number, name of the depositor, type of the account, balance amount in the account. Use constructors to assign initial values, to Deposit an amount, to Withdraw amount after checking balance, to display name and balance.

Code:

class BankAccount

{

private int accountNumber;

private String accountName;

private String accountType;

private double balance;

public BankAccount(int accountNumber, String accountName, String accountType, double balance)

{

this.accountNumber = accountNumber;

this.accountName = accountName;

this.accountType = accountType;

this.balance = balance;

}

public void deposit(double amount)

{

balance += amount;

}

public void withdraw(double amount)

{

if (balance >= amount)

{

balance -= amount;

}

else

{

System.out.println("Insufficient balance!");

}

}

public void display()

{

System.out.println("Name: " + accountName);

System.out.println("Balance: " + balance);

}

}

class Main

{

public static void main(String[] args)

{

BankAccount myAccount = new BankAccount(12345, "John Doe", "Checking", 1000.0);

myAccount.deposit(500.0);

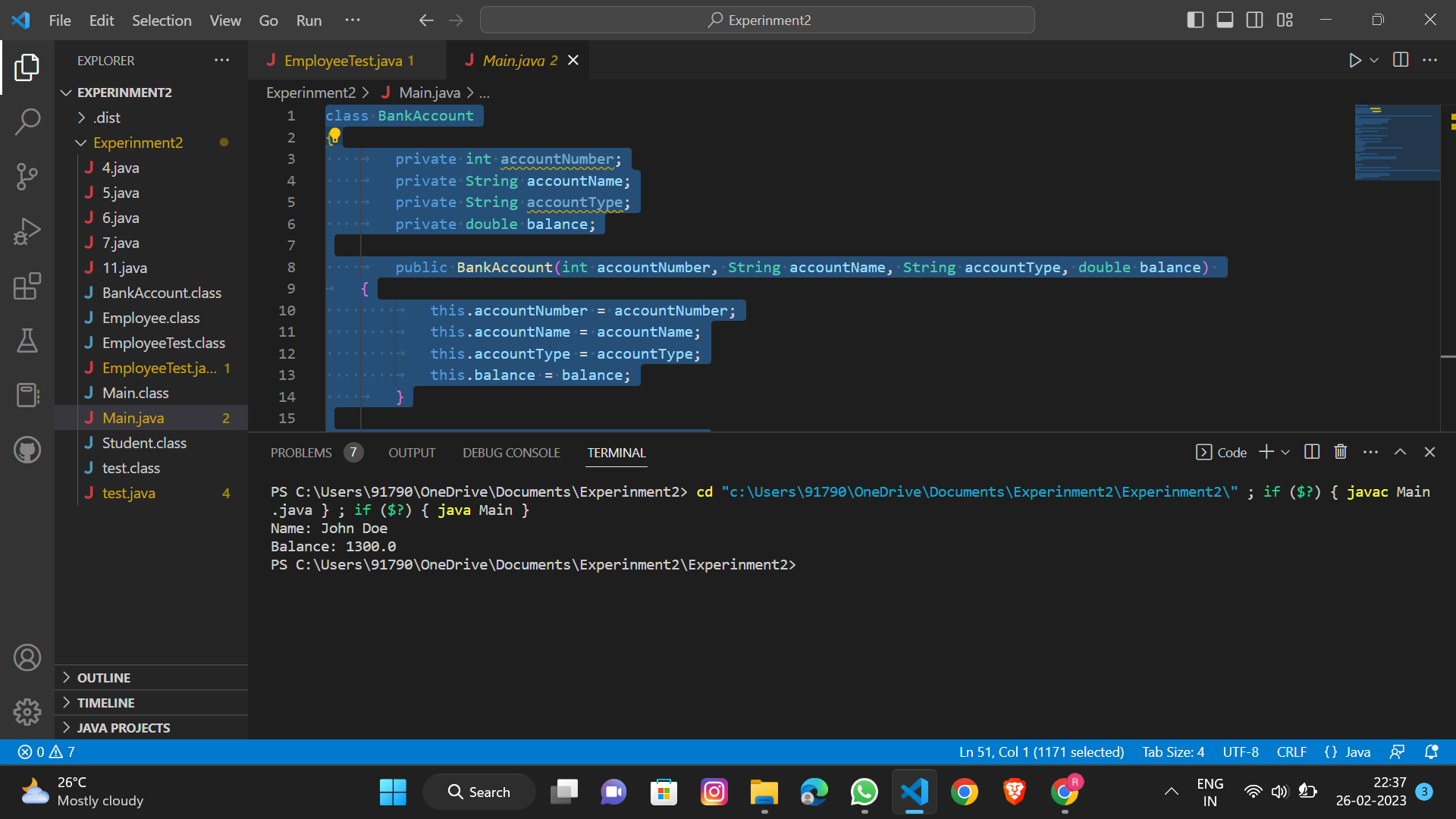
myAccount.withdraw(200.0);

myAccount.display();

}

}

Output:



4.Java Program to Find Area of Square, Rectangle and Circle using Method Overloading.

Code:

class AreaCalculator

{

public static void main(String[] args)

{

double squareArea = calculatesquare(5.0);

double rectangleArea = calculaterectangle(4.0, 6.0);

double circleArea = calculatecircle(3.0);

System.out.println("The area of the square is " + squareArea);

System.out.println("The area of the rectangle is " + rectangleArea);

System.out.println("The area of the circle is " + circleArea);

}

public static double calculatesquare(double side)

{

return side \* side;

}

public static double calculaterectangle(double length, double width)

{

return length \* width;

}

public static double calculatecircle(double radius)

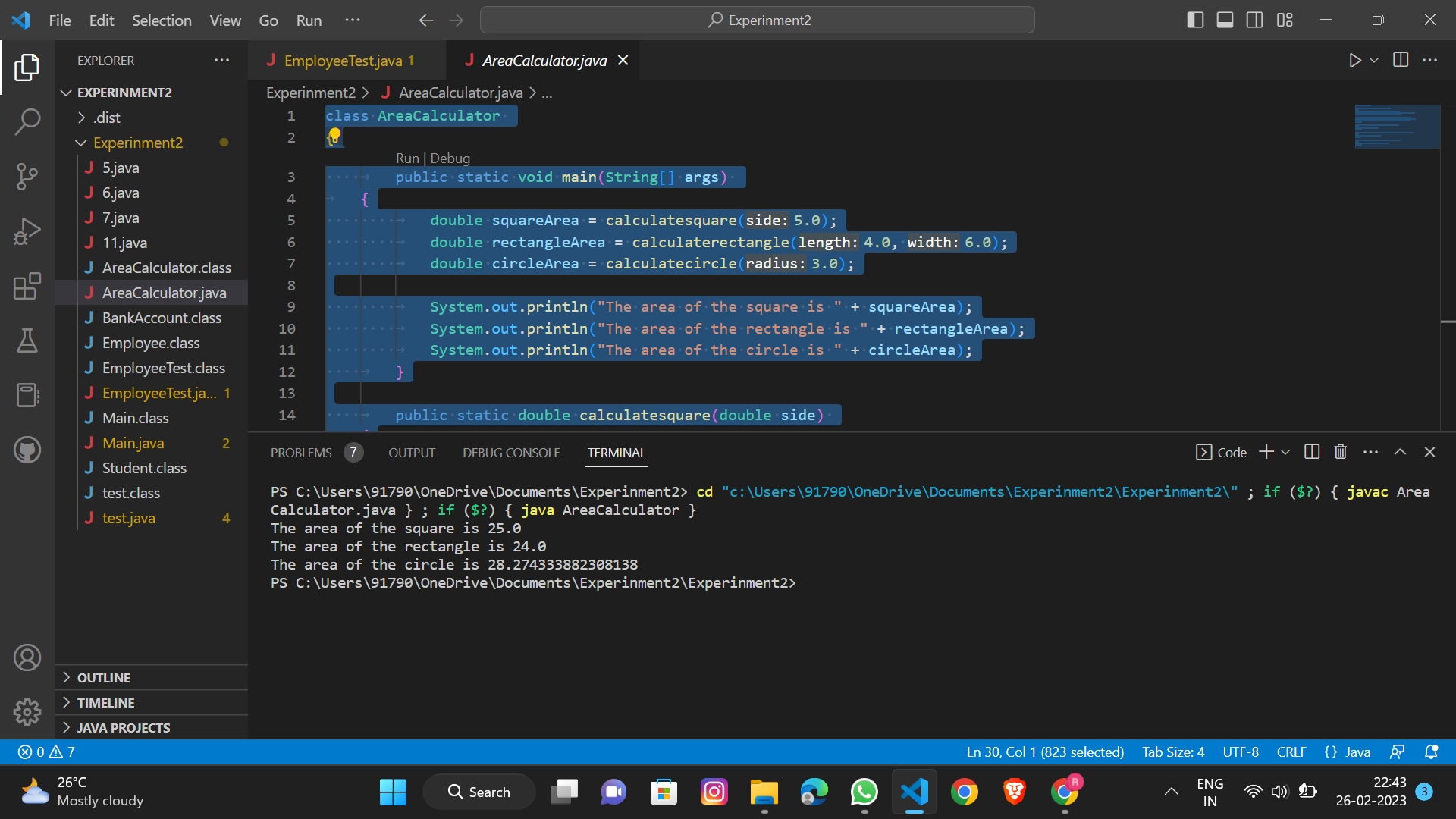
{

return Math.PI \* radius \* radius;

}

}

Output:



5.Implement a java program create class Box with necessary member for calculate volume of Box and display it with method name Volume. Create Box class object and to display volume of Box.

Code:

class Box

{

private double length;

private double width;

private double height;

public Box(double length, double width, double height)

{

this.length = length;

this.width = width;

this.height = height;

}

public double getVolume()

{

double volume = length \* width \* height;

return volume;

}

public static void main(String[] args)

{

Box box = new Box(10.0, 5.0, 2.0);

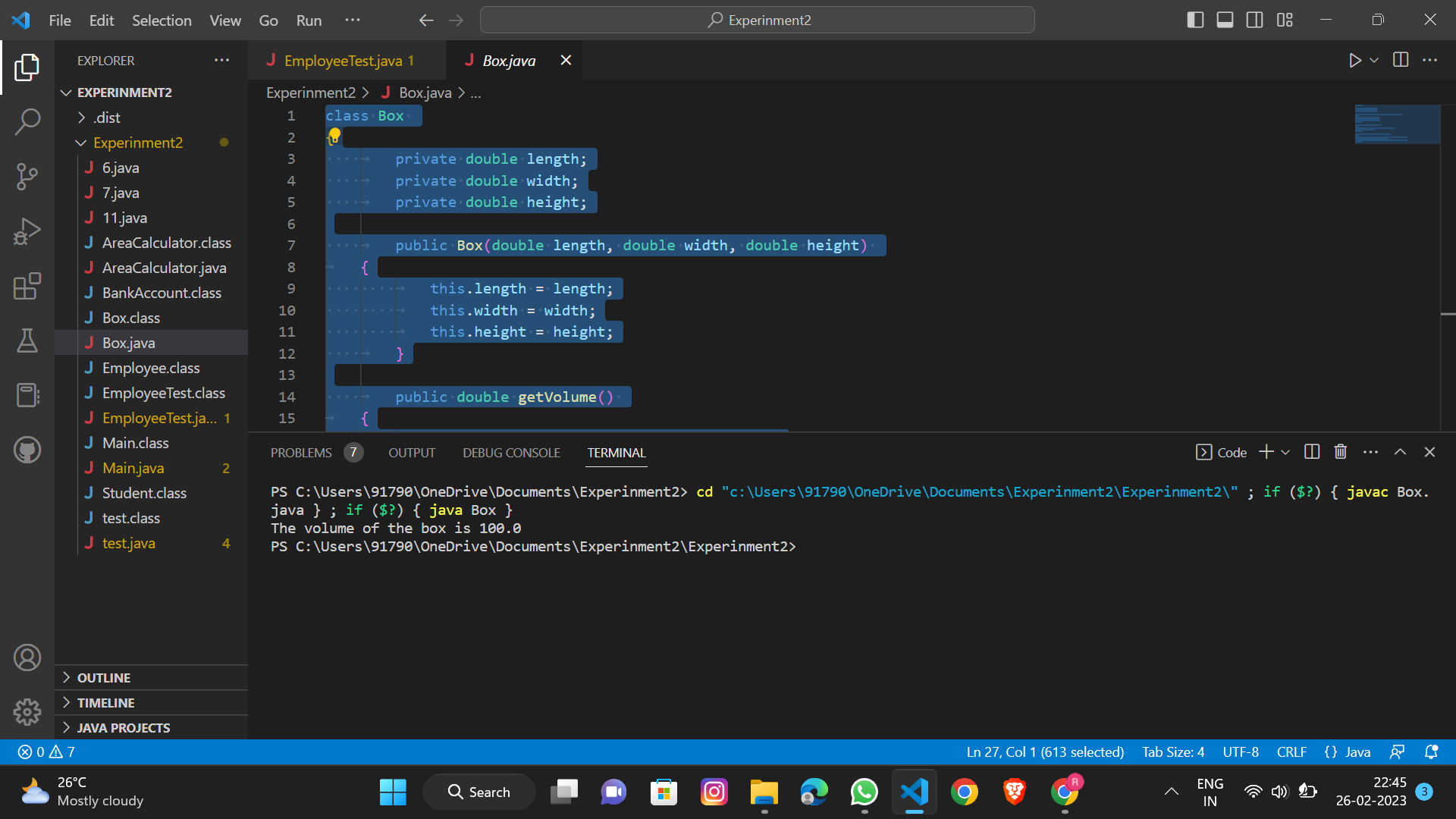
double volume = box.getVolume();

System.out.println("The volume of the box is " + volume);

}

}

Output:



6.Implement java program to create class Person with data member like age, name ,address, mobile number. and declare them as private along with method to take input of data member getinput() and for display details show(). Create Person class object and display details.

Code:

import java.util.Scanner;

class Person

{

private int age;

private String name;

private String address;

private String mobileNumber;

public void getInput()

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter age: ");

age = scanner.nextInt();

System.out.print("Enter name: ");

name = scanner.next();

System.out.print("Enter address: ");

address = scanner.next();

System.out.print("Enter mobile number: ");

mobileNumber = scanner.next();

}

public void show()

{

System.out.println("Name: " + name);

System.out.println("Age: " + age);

System.out.println("Address: " + address);

System.out.println("Mobile number: " + mobileNumber);

}

public static void main(String[] args)

{

Person person = new Person();

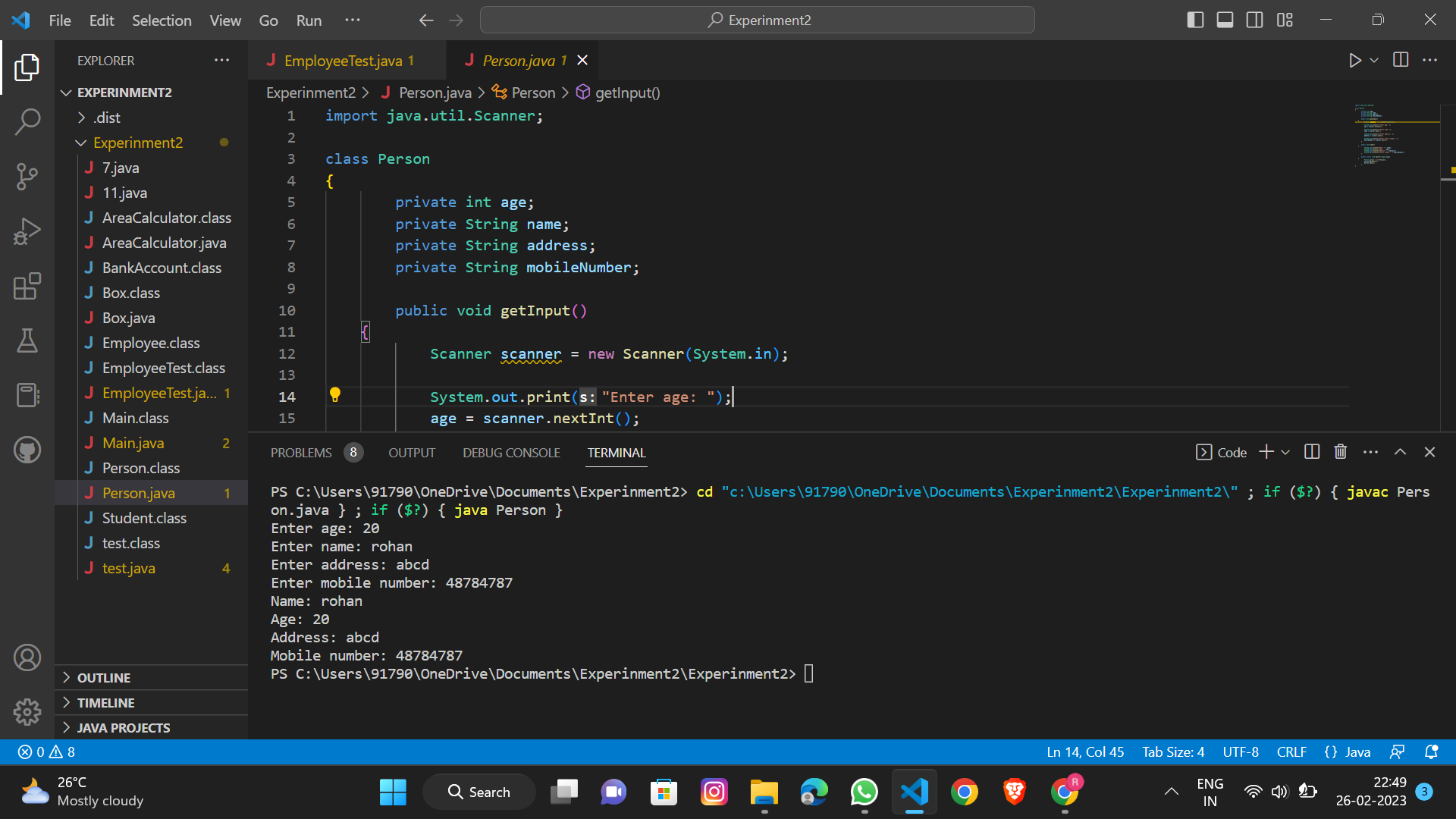
person.getInput();

person.show();

}

}

Output:



7. Consider 2 program problem in which add condition for which employee get highest salary and display that salary with necessary employee details.

Code:

import java.util.Scanner;

class Employee

{

private String name;

private int age;

private double salary;

public Employee(String name, int age, double salary)

{

this.name = name;

this.age = age;

this.salary = salary;

}

public String getName()

{

return name;

}

public int getAge()

{

return age;

}

public double getSalary()

{

return salary;

}

}

class Main1

{

public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of employees: ");

int numEmployees = scanner.nextInt();

Employee highestPaidEmployee = null;

double highestSalary = Double.NEGATIVE\_INFINITY;

for (int i = 0; i < numEmployees; i++)

{

System.out.println("Enter details for employee " + (i+1) + ":");

System.out.print("Name: ");

String name = scanner.next();

System.out.print("Age: ");

int age = scanner.nextInt();

System.out.print("Salary: ");

double salary = scanner.nextDouble();

if (salary > highestSalary)

{

highestPaidEmployee = new Employee(name, age, salary);

highestSalary = salary;

}

}

System.out.println("Employee with highest salary:");

System.out.println("Name: " + highestPaidEmployee.getName());

System.out.println("Age: " + highestPaidEmployee.getAge());

System.out.println("Salary: " + highestPaidEmployee.getSalary());

}

}

Output: