COURSE OUTCOME 3

PROGRAM NO: 1

AIM:

To find area of different shapes using overloaded functions.

ALGORITHM:

Step 1: Start

Step 2: Define the main class

Step 3: Define methods with the same methodname that performs the area operation for each shape

Step 4: Display the areas of each shapes.

PROGRAM CODE:

| CO3Q1.java | ```java
import java.util.Scanner;

public class CO3Q1
{
    double area(float r)
    {
        double pi = 3.14;
        double ar;
        ar = pi*r*r;
        return ar;
    }
    double area(float h,float b)
    {
        double ar;
        ar = (h*b)/2;
        return ar;
    }
    double area(double s)
    {
        double ar;
        ar = s*s;
        return ar;
    }
``` |

```java
    double area(double l,double br)
    {
        double ar;
        ar = l*br;
        return ar;
    }
    public static void main(String[] args)
    {
        CO3Q1 obj = new CO3Q1();
        int ch;
        float r,h,b;
        double s,l,br;
        System.out.println("Enter the option:");
        System.out.println("1. Area of the circle");
        System.out.println("2. Area of the triangle");
        System.out.println("3. Area of the square");
        System.out.println("4. Area of the rectangle");
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the choice");
        ch = sc.nextInt();
        switch(ch)
        {
            case 1: System.out.println("Enter the radius:");
                r = sc.nextFloat();
                System.out.println(obj.area(r));
                break;
            case 2: System.out.println("Enter the height and breadth:");
                h = sc.nextFloat();
                b = sc.nextFloat();
                System.out.println(obj.area(h,b));
                break;
            case 3: System.out.println("Enter the length of the side:");
                s = sc.nextDouble();
                System.out.println(obj.area(s));
                break;
            case 4: System.out.println("Enter the length and breadth:");
                l = sc.nextDouble();
                br = sc.nextDouble();
                System.out.println(obj.area(l,br));
                break;
            default: System.out.println("Invalid choice.");
        }
    }
}
```

OUTPUT:

```
run:
Enter the option:
1. Area of the circle
2. Area of the triangle
3. Area of the square
4. Area of the rectangle
Enter the choice
1
Enter the radius:
5
78.5
BUILD SUCCESSFUL (total time: 13 seconds)
```

```
run:
Enter the option:
1. Area of the circle
2. Area of the triangle
3. Area of the square
4. Area of the rectangle
Enter the choice
2
Enter the height and breadth:
10
5
25.0
BUILD SUCCESSFUL (total time: 36 seconds)
```

```
run:
Enter the option:
1. Area of the circle
2. Area of the triangle
3. Area of the square
4. Area of the rectangle
Enter the choice
4
Enter the length and breadth:
4
6
24.0
BUILD SUCCESSFUL (total time: 10 seconds)
```

```
run:
Enter the option:
1. Area of the circle
2. Area of the triangle
3. Area of the square
4. Area of the rectangle
Enter the choice
3
Enter the length of the side:
5
25.0
BUILD SUCCESSFUL (total time: 10 seconds)
```

RESULT:

The program is successfully executed and the output is verified.

PROGRAM NO: 2

AIM:

To create a class 'Employee' with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class 'Teacher' that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.

ALGORITHM:

Step.1: Start the program.

Step.2: Define a class '*Employee*' with data members Empid, Name, Salary, Address and a constructor to initialize these members.

Step.3: Define a class '*Teacher*' that inherit the properties of class 'Employee' and contain its own data members Department, Subjects taught and constructors to initialize these data members and also include a method Display() to display all the data members.

Step.4: Define a main() to create array of objects for the class to display the details of 'N' teachers.

Step.5: Stop the program.

PROGRAM CODE:

| employee.java | ```class Employee2
{
    int Empid;
    String Name;
    float Salary;
    String Address;

    Employee2()
    {

    }
    public Employee2(int id, String name, float sal, String addr)
    {
        Empid = id;
        Name = name;
        Salary = sal;``` |

```java
            Address = addr;
        }

    }
class Teacher extends Employee2
{
    String department;
    String Subjects;

    public Teacher(int id, String name, float sal, String addr, String dept, String sub)
    {
        super(id, name, sal, addr);
        department = dept;
        Subjects = sub;

    }

    Teacher()
    {

    }

    public void display()
    {
        System.out.println("Employee ID - "+Empid);
        System.out.println("Employee Name - "+Name);
        System.out.println("Salary - "+Salary);
        System.out.println("Address - "+Address);
        System.out.println("Department - "+department);
        System.out.println("Subject - "+Subjects);
    }

}
public class CO3Q2
{
    public static void main(String[] args)
    {
        int n;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of Teachers to be added:");
        n = sc.nextInt();
        Teacher obj[] = new Teacher[n];
        for (int i=0; i<n; i++)
        {
            obj[i] = new Teacher();
        }
        for(int i=0;i<n;i++)
        {
            System.out.println("\t*****");
            System.out.println("Enter Employee ID");
            obj[i].Empid = sc.nextInt();
```

```java
            sc.nextLine();
            System.out.println("Enter Employee Name");
            obj[i].Name = sc.nextLine();
            System.out.println("Enter Employee Salary");
            obj[i].Salary = sc.nextFloat();
            sc.nextLine();
            System.out.println("Enter Employee Address");
            obj[i].Address = sc.nextLine();
            System.out.println("Enter Employee Department");
            obj[i].department = sc.nextLine();
            System.out.println("Enter Employee Subject");
            obj[i].Subjects = sc.nextLine();
        }
    System.out.println("\t***************");
    System.out.println("Employee Details:-");
    for(int i=0;i<n;i++)
        obj[i].display();
    }
}
```

OUTPUT:

```
run:
Enter the number of Teachers to be added:
2
        *****
Enter Employee ID
101
Enter Employee Name
Dustin
Enter Employee Salary
60000
Enter Employee Address
London
Enter Employee Department
Literature
Enter Employee Subject
English
        *****
Enter Employee ID
102
Enter Employee Name
Millie
Enter Employee Salary
85000
Enter Employee Address
New York
Enter Employee Department
Science
Enter Employee Subject
Physics
        ***************
Employee Details:-
Employee ID - 101
Employee Name - Dustin
Salary - 60000.0
Address - London
Department - Literature
Subject - English
Employee ID - 102
Employee Name - Millie
Salary - 85000.0
Address - New York
Department - Science
Subject - Physics
BUILD SUCCESSFUL (total time: 1 minute 10 seconds)
```

RESULT:

The program is successfully executed and the output is verified.

PROGRAM NO: 3

AIM:

To create a class 'Person' with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class 'Employee' that inherits the properties of class Person and also contains its own data members like Empid, Company_name, Qualification, Salary and its own constructor. Create another class 'Teacher' that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.

ALGORITHM:

Step 1: Start

Step 2: Create a class named '*Person'* with data members name, gender, address and age

& a constructor to initialize them.

Step 3: Create a class named '*Employee'* which is derived from Person, with data members

empid, cmpnyname, qualification and sal & a constructor Employee() to initialize them.

Step 4: Create class named 'Teach" which is derived from Employee, with data members

subject, dept and tid ; a constructor to initilize members ; and a function named

display() to display details.

Step 5: Create an array of objects to display details.

Step 6: Stop

PROGRAM CODE:

| CO3Q3.java | ```java
import java.util.Scanner;


class Person
{
    String Name;
    String Gender;
    String Address;
    int Age;
``` |

```java
        Person()
        {

        }
        Person(String name, String gender, String addr, int age)
        {
            Name = name;
            Gender = gender;
            Address = addr;
            Age = age;
        }
}

class Employee extends Person
{
    int Empid;
    String Company_name;
    String Qualification;
    float Salary;
    Employee()
    {

    }
    public Employee(String name, String gender, String addr, int age)
    {
        super(name, gender, addr, age);
    }
    public Employee(int id,String name, String qual, float sal)
    {
        Empid = id;
        Company_name = name;
        Qualification = qual;
        Salary = sal;
    }

}
class Teacher extends Employee
{
    String Subject;
    String Department;
    String Teachersid;
    Teacher()
    {

    }
    Teacher(String sub, String dept, String id)
```

```java
        {
          Subject = sub;
          Department = dept;
          Teachersid = id;
        }
        public void display()
        {
          System.out.println("Name:" + Name);
            System.out.println("Age:" + Age);
            System.out.println("Gender:" + Gender);
            System.out.println("Address:" + Address);
            System.out.println("Emp id:" + Empid);
            System.out.println("Salary:" + Salary);
            System.out.println("Qualification:" + Qualification);
            System.out.println("Company Name:" + Company_name);
            System.out.println("Teacher id:" + Teachersid);
            System.out.println("Subject:" + Subject);
            System.out.println("Department:" + Department);
          System.out.println("\n\n");
        }
}

public class CO3Q3
{
    public static void main(String[] args)
    {
        int n;
        System.out.println("Enter the no. of Teachers:");
        Scanner sc = new Scanner(System.in);
        n = sc.nextInt();
        Teacher obj[] = new Teacher[n];
        for(int i=0;i<n;i++)
            obj[i] = new Teacher();
        sc.nextLine();
        for(int i=0;i<n;i++)
        {
            System.out.println("\t*****");
            System.out.println("Enter the name:");
            obj[i].Name = sc.nextLine();

            System.out.println("Enter the Age:");
            obj[i].Age = sc.nextInt();
            sc.nextLine();
            System.out.println("Enter the Gender:");
            obj[i].Gender = sc.nextLine();
            System.out.println("Enter the Address:");
```

```java
            obj[i].Address = sc.nextLine();
            System.out.println("Enter the Emp id:");
            obj[i].Empid = sc.nextInt();
            System.out.println("Enter the Salary:");
            obj[i].Salary = sc.nextFloat();
            sc.nextLine();
            System.out.println("Enter the Qualification:");
            obj[i].Qualification = sc.nextLine();
            System.out.println("Enter the Company Name:");
            obj[i].Company_name = sc.nextLine();
            System.out.println("Enter the Teacher id:");
            obj[i].Teachersid = sc.nextLine();
            System.out.println("Enter the Subject:");
            obj[i].Subject = sc.nextLine();
            System.out.println("Enter the Department:");
            obj[i].Department = sc.nextLine();
        }
        System.out.println("\t***************");
        System.out.println("Teachers Details:-\n");
        for(int i=0;i<n;i++)
            obj[i].display();
    }

}
```

OUTPUT:

```
run:
Enter the no. of Teachers:
2
        *****
Enter the name:
Mike
Enter the Age:
19
Enter the Gender:
Male
Enter the Address:
Germany
Enter the Emp id:
101
Enter the Salary:
54000
Enter the Qualification:
MCA
Enter the Company Name:            Teachers Details:-
Google
Enter the Teacher id:               Name:Mike
101                                 Age:19
Enter the Subject:                  Gender:Male
Science                             Address:Germany
Enter the Department:               Emp id:101
Physics                             Salary:54000.0
        *****                       Qualification:MCA
Enter the name:                     Company Name:Google
Nancy                               Teacher id:101
Enter the Age:                      Subject:Science
20                                  Department:Physics
Enter the Gender:
Female
Enter the Address:
Australia
Enter the Emp id:
102                                 Name:Nancy
Enter the Salary:                   Age:20
70000                               Gender:Female
Enter the Qualification:            Address:Australia
Msc                                 Emp id:102
Enter the Company Name:             Salary:70000.0
Microsoft                           Qualification:Msc
Enter the Teacher id:               Company Name:Microsoft
102                                 Teacher id:102
Enter the Subject:                  Subject:Statistics
Statistics                          Department:Maths
Enter the Department:
Maths
        **************
```

RESULT:

The program is successfully executed and the output is verified.

PROGRAM NO: 4

AIM:

To write a program has class Publisher, Book, Literature and Fiction. Read the information and print the details of books from either the category, using inheritance.

ALGORITHM:

Step 1: Start

Step 2:Create a class named 'Publisher' with data members pname, pid; a constructor named

Publisher().

Step 3: Create a class named 'Book' which is derived 'Publisher' with data members nop, price; a constructor named Book().

Step 4: Create a class named 'literature' which is derived from Book with data members title, author; a constructor; a function show() to display details.

Step 5: Create a class named 'fiction' which is derived from Book with data members bname, auth; a constructor; a function display() to print details.

Step 6: Print a menu defining the type of genres; if literature create an object of literature

type and object of type fiction if fiction is chosen.

Step 7 : Stop

PROGRAM CODE:

| CO3Q4.java | ```java
import java.util.Scanner;


public class CO3Q4
{
    public class publisher
    {
        String pub_name;
        publisher(){}
        publisher(String name)
        {
            pub_name = name;
        }
    }
    static public class Book extends publisher
    {
``` |
| --- | --- |

```java
        String book_name;
        Book(){}
        Book(String bname, String pname)
        {
          super(pname);
          book_name = bname;
        }
        public void display()
        {
          System.out.println("\n\n*********\n\nBook Details");
          System.out.println("Publisher:"+ pub_name);
          System.out.println("Book:"+book_name);
        }
      }
      static public class literature extends Book
      {
        String book_genre;

        public literature() {}
        literature(String name,String book,String genre)
        {
          super(name,book);
          book_genre = genre;
          super.display();
          System.out.println("Genre:"+book_genre);
        }

      }
      static public class fiction extends Book
      {
        String book_genre;
        fiction(){}
        fiction(String name,String book,String genre)
        {
          super(name,book);
          book_genre = genre;
          super.display();
          System.out.println("Genre:"+book_genre);
        }
      }


      public static void main(String[] args)
      {
        String name,book,genre;
```

```
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter the book details...");
            System.out.println("Enter the Book name:");
            book = sc.nextLine();
            System.out.println("Enter the Publisher name:");
            name = sc.nextLine();
            System.out.println("Enter the Genre name:");
            genre = sc.nextLine();
            fiction obj;
            literature ob;
            if(genre.toLowerCase().equals("fiction"))
            {
               obj = new fiction(name,book,genre);
            }
            else if(genre.toLowerCase().equals("literature"))
               ob = new literature(name,book,genre);
            else
               System.out.println("Enter Fiction or Literature");
        }
}
```

OUTPUT:

```
run:
Enter the book details...
Enter the Book name:
Harry Potter
Enter the Publisher name:
Pottermore Publishing
Enter the Genre name:
Fiction


**********

Book Details
Publisher:Harry Potter
Book:Pottermore Publishing
Genre:Fiction
BUILD SUCCESSFUL (total time: 22 seconds)
```

RESULT:

The program is successfully executed and the output is verified.

PROGRAM NO: 5

AIM:

To create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student.

ALGORITHM:

Step.1: Start the program.

Step.2: Define a class '*Student*' which will read a student's academic information from the user.

Step.3: Define another class '*Sports*' that extends '*Student*' and reads the sports data of the student.

Step.4: Define another interface '*Results*' that extends '*Sports*'and has a *Display()* to display the profile, academic score and sports score of the student.

Step.5: Define a main () method to create objects for the above classes and to call the associated member methods.

Step.6: Stop the program.

PROGRAM CODE:

CO3Q5.java	import java.util.Scanner;
	interface results
	{
	void getdata();
	int display();
	}
	class Student implements results
	{
	int std_id,std_tmark;
	String std_name;
	@Override
	public void getdata()
	{
	Scanner sc = new Scanner(System.in);

```java
        System.out.println("Enter the name of the student:");
        std_name = sc.nextLine();
        System.out.println("Enter the student id:");
        std_id = sc.nextInt();
        System.out.println("Enter total academic mark:");
        std_tmark = sc.nextInt();
    }
    @Override
    public int display()
    {
        System.out.println("\t--------Student details--------");
        System.out.println("Student name: " +std_name);
        System.out.println("Student id: " +std_id);
        System.out.println("Total mark:" +std_tmark);
        return std_tmark;
    }
}

class Sports implements results
{
    int tmarks;
    @Override
    public void getdata()
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the marks obtained in sports:");
        tmarks = sc.nextInt();
    }
    @Override
    public int display()
    {
        System.out.println("Marks obtained in Sports:"+ tmarks);
        return tmarks;
    }

}
public class CO3Q5
{
    public static void main(String[] args)
    {
```

```
                        int mark;
                        Student ob = new Student();
                        Sports obj = new Sports();
                        ob.getdata();
                        obj.getdata();
                        mark = ob.display();
                        mark = mark + obj.display();
                        System.out.println("Marks(Academic+Sports)="+ mark);
                  }
            }
```

OUTPUT:

```
run:
Enter the name of the student:
Tokyo
Enter the student id:
501
Enter total academic mark:
200
Enter the marks obtained in sports:
25
        --------Student details--------
Student name: Tokyo
Student id: 501
Total mark:200
Marks obtained in Sports:25
Marks(Academic+Sports)=225
BUILD SUCCESSFUL (total time: 26 seconds)
```

RESULT:

The program is successfully executed and the output is verified.

PROGRAM NO: 6

AIM:

To create an interface having prototypes of functions area() and perimeter(). Create two classes Circle and Rectangle which implements the above interface. Create a menu driven program to find area and perimeter of objects.

ALGORITHM:

Step.1: Start the program.

Step.2: Define an interface '*prop*' with methods to read inputs and calculate area and perimeter.

Step.3: Define a class '*Rectangle*' that extends *Circle* to initialize its data members l, b and to calculate and display the area and perimeter of a rectangle.

Step.4: Define a main () to create objects for the above classes to invoke its member methods to print the results.

Step.5: Stop the program.

PROGRAM CODE:

| CO3Q6.java | ```java
import java.util.Scanner;
interface prop{
   void getdata();
   void area();
   void perimeter();
}
class Circle implements prop{
   double pi = 3.14;
   double r;
   Scanner sc = new Scanner(System.in);
   @Override
   public void getdata(){
      System.out.println("Enter the radius of the circle:");
      r = sc.nextDouble();
   }
   @Override
   public void perimeter(){
      System.out.println("Perimeter of the circle: "+(2*pi*r));
``` |
|---|---|

```java
        }
        @Override
        public void area(){
            System.out.println("Perimeter of the circle: "+(pi*r*r));
        }
    }
    class Rectangle implements prop{
        double l,b;
        Scanner sc = new Scanner(System.in);
        @Override
        public void getdata(){
            System.out.println("Enter the length of the rectangle:");
            l = sc.nextDouble();
            System.out.println("Enter the breadth of the rectangle:");
            b = sc.nextDouble();
        }
        @Override
        public void area(){
            System.out.println("Perimeter of a rectangle: "+(l*b));
        }
        @Override
        public void perimeter(){
            System.out.println("Perimeter of a rectangle: "+(2*(l+b)));
        }
    }
    public class CO3Q6 {
        public static void main(String[] args) {
            int ch;
            Scanner sc = new Scanner(System.in);
            Circle ob = new Circle();
            Rectangle obj = new Rectangle();
            do{
                System.out.println("\n1.Circle\n2.Rectangle\n3.exit");
                System.out.println("Enter your choice:");
                ch = sc.nextInt();
                switch(ch){
                    case 1 :ob.getdata();
                            ob.area();
                            ob.perimeter();
                            break;
                    case 2 :obj.getdata();
                            obj.area();
                            obj.perimeter();
                            break;
                    case 3 :System.out.println("Exited...");
                            System.exit(0);
```

```
                }
            }while(true);
        }
    }
```

OUTPUT:

```
run:

1.Circle
2.Rectangle
3.exit
Enter your choice:
1
Enter the radius of the circle:
5
Perimeter of the circle: 78.5
Perimeter of the circle: 31.400000000000002

1.Circle
2.Rectangle
3.exit
Enter your choice:
2
Enter the length of the rectangle:
10
Enter the breadth of the rectangle:
6
Perimeter of a rectangle: 60.0
Perimeter of a rectangle: 32.0

1.Circle
2.Rectangle
3.exit
Enter your choice:
3
Exited...
BUILD SUCCESSFUL (total time: 20 seconds)
```

RESULT:

The program is successfully executed and the output is verified.

PROGRAM NO: 7

AIM:

To prepare bill with the given format using calculate method from interface.

Order No.
Date:

Product Id	Name	Quantity	unit price	Total
101	A	2	25	50
102	B	1	100	100
			Net. Amount	150

ALGORITHM:

Step.1: Start the program.

Step.2: Define an interface *calc* with a method calculate().

Step.3: Define a class *bill* that implements *calc* to calculate the total amount for each product and has methods to generate a bill as given in the question.

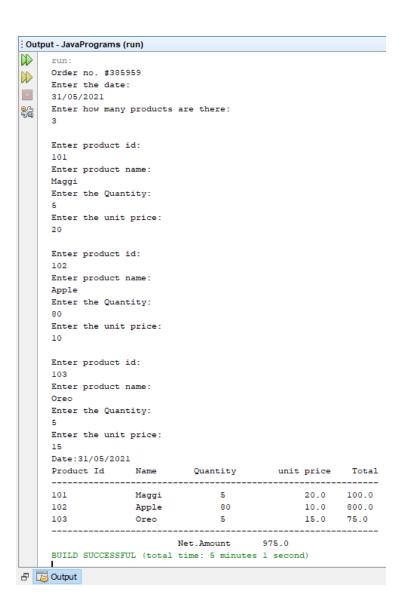**Step.4**: Define a main () to create objects for the class.

Step.5: Invoke the above methods by passing the data collected from user to generate the required bill.

Step.6: Stop the program.

PROGRAM CODE:

| CO3Q7. java | ```import java.util.Scanner;
interface calc{
    void calculate();
}
class bill implements calc{
    String date,name,p_id;
    int quantity;``` |
|------|------|

```java
        double unit_price,total,namount=0;
        Scanner sc = new Scanner(System.in);
        public void getdata(){
            System.out.println("\nEnter product id:");
            p_id = sc.nextLine();
            System.out.println("Enter product name:");
            name = sc.nextLine();
            System.out.println("Enter the Quantity:");
            quantity = sc.nextInt();
            System.out.println("Enter the unit price:");
            unit_price = sc.nextDouble();
        }
        @Override
        public void calculate(){
            total = quantity * unit_price;
        }
        public void display(){
            System.out.println(p_id+"\t\t"+name+"\t\t"+quantity+"\t\t"+unit_price+"\t"+total);
        }
}
public class CO3Q7 {
    public static void main(String[] args) {
        int n,i;
        double namount=0,t;
        int ran;
        String date;
        t = Math.random() *1000000;
        ran = (int) t;
        Scanner sc = new Scanner(System.in);
        System.out.println("Order no. #"+ran);
        System.out.println("Enter the date:");
        date = sc.nextLine();
        System.out.println("Enter how many products are there:");
        n = sc.nextInt();
        bill ob[] = new bill[n];
        for(i=0;i<n;i++)
            ob[i] = new bill();
        for(i=0;i<n;i++){
            ob[i].getdata();
            ob[i].calculate();
        }
        System.out.println("Date:"+date);
        System.out.println("Product Id \tName\t   Quantity\t   unit price\t Total ");
        System.out.println("-------------------------------------------------------------");
        for(i=0;i<n;i++){
            ob[i].display();
```

```
                    namount += ob[i].total;
            }
            System.out.println("-------------------------------------------------------------");
            System.out.println("\t\t\tNet.Amount\t"+ namount);


        }}
```

OUTPUT:

```
Output - JavaPrograms (run)
  run:
  Order no. #385959
  Enter the date:
  31/05/2021
  Enter how many products are there:
  3

  Enter product id:
  101
  Enter product name:
  Maggi
  Enter the Quantity:
  5
  Enter the unit price:
  20

  Enter product id:
  102
  Enter product name:
  Apple
  Enter the Quantity:
  80
  Enter the unit price:
  10

  Enter product id:
  103
  Enter product name:
  Oreo
  Enter the Quantity:
  5
  Enter the unit price:
  15
  Date:31/05/2021
  Product Id      Name        Quantity        unit price   Total
  -------------------------------------------------------------
  101             Maggi           5               20.0     100.0
  102             Apple          80               10.0     800.0
  103             Oreo            5               15.0     75.0
  -------------------------------------------------------------
                       Net.Amount        975.0
  BUILD SUCCESSFUL (total time: 5 minutes 1 second)

  Output
```

RESULT:

The program is successfully executed and the output is verified.