**CODE:**

1. **Interface.java**

import java.rmi.\*;

public interface circleInterface extends Remote{

public double area(int radius) throws RemoteException;

public double perimeter(int radius) throws RemoteException;

}

1. **ImplementInterface.java**

import java.rmi.\*;

import java.rmi.server.\*;

@SuppressWarnings("serial")

public class circleImpl extends UnicastRemoteObject implements circleInterface{

public circleImpl() throws RemoteException {

super();

}

public double area(int radius) throws RemoteException {

double pi=3.14;

return pi\*radius\*radius;

}

public double perimeter(int radius) throws RemoteException {

double pi=3.14;

return 2\*pi\*radius;

}

}

1. **Server.java**

import java.rmi.\*;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

public class Server{

public Server() throws RemoteException {

super();

}

public static void main(String args[]) throws RemoteException {

Registry reg=LocateRegistry.createRegistry (101);

circleImpl ci=new circleImpl();

reg.rebind("circle", ci);

System.out.println("Server is waiting....");

}}

1. **Client.java**

import java.net.MalformedURLException;

import java.rmi.\*;

import java.util.Scanner;

public class Client{

public Client(){

}

public static void main(String args[]) throws RemoteException, MalformedURLException, NotBoundException{

int radius;

@SuppressWarnings("resource")

Scanner in=new Scanner(System.in);

System.out.println("Enter the radius of the circle");

radius=in.nextInt();

circleInterface c=(circleInterface) Naming.lookup("rmi://localhost:101/circle");

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

System.out.println("The perimeter of the circle is "+c.perimeter(radius));

}

}

**OUTPUT:**



