Classes and Objects Practice Problem

P1. Create a class named 'Student' with String variable 'name' and integer variable 'roll_no'. Assign the value of roll_no as '2' and that of name as "John" by creating an object of the class Student.

```
class Student

{

        String name;

        int roll_no;

}




class ques1 extends Student

{

        public static void main(String args[])

        {

                Student st = new Student();

                st.name = "John";

                st.roll_no = 2;



                System.out.println(st.name);

                System.out.println(st.roll_no);

        }

}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac ques1.java

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ques1
John
2
```

P2. Assign and print the roll number, phone number and address of two students having names "Sam" and "John" respectively by creating two objects of class 'Student'.

```
class Student

{

        String name;

        int roll_number;

        String phone_number;

        String address;



}




class ques2 extends Student

{

        public static void main(String args[])

        {

                Student st = new Student();

                st.name = "Sam";

                st.roll_number = 1;

                st.phone_number = "123456789";

                st.address = "Downing Street UK";
```

```
        Student st2 = new Student();
        st2.name = "John";
        st2.roll_number = 2;
        st2.phone_number = "9876543211";
        st2.address = "Detroit USA";

        System.out.println("For Student named as :" + st.name);
        System.out.println("Roll Number: " + st.roll_number);
        System.out.println("Phone Number: " + st.phone_number);
        System.out.println("Address : "+st.address);

        System.out.println("For Student named as :" + st2.name);
        System.out.println("Roll Number: " + st2.roll_number);
        System.out.println("Phone Number: " + st2.phone_number);
        System.out.println("Address : "+st2.address);
    }
}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac ques2.java

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ques2
For Student named as :Sam
Roll Number: 1
Phone Number: 123456789
Address : Downing Street UK
For Student named as :John
Roll Number: 2
Phone Number: 9876543211
Address : Detroit USA
```

P3. Write a program to print the area and perimeter of a triangle having sides of 3, 4 and 5 units by creating a class named 'Triangle' without any parameter in its constructor.

```
class Triangle
{
	static int s1 = 3;
	static int s2 = 4;
	static int s3 = 5;

	Triangle()
	{
		int s = (Triangle.s1 + Triangle.s2 + Triangle.s3)/2;
		int area_o = s*(s-Triangle.s1)*(s - Triangle.s2)*(s-Triangle.s3);
		String area = String.format("%.2f" , Math.sqrt(area_o));
		System.out.println("Area of Triangle : " + area);
		System.out.println("Perimeter of Triangle : " + s*2);
	}
}

class ques3
{
	public static void main(String args[])
	{
		Triangle t1 = new Triangle();
	}
}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac ques3.java

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ques3
Area of Triangle : 6.00
Perimeter of Triangle : 12
```

P4. Write a program to print the area and perimeter of a triangle having sides of 3, 4 and 5 units by creating a class named 'Triangle' with constructor having the three sides as its parameters.

```
class Triangle
{
	Triangle()
	{}
	Triangle(int a , int b , int c)
	{
		int perimeter = a + b + c;
		int s = perimeter/2;
		int o_area = s*(s-a)*(s-b)*(s-c);
		String area = String.format("%.2f",Math.sqrt(o_area));
		System.out.println("Area of Triangle : " + area);
		System.out.println("Perimeter of Triangle : " + perimeter);
	}
}
```

```
class ques4
```

```
{
	public static void main(String args[])
	{
		Triangle t = new Triangle(3,4,5);
	}
}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ques4
Area of Triangle : 6.00
Perimeter of Triangle : 12

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>
```

P5. Write a program to print the area of two rectangles having sides (4,5) and (5,8) respectively by creating a class named 'Rectangle' with a method named 'Area' which returns the area and length and breadth passed as parameters to its constructor.

```
class Rectangle
{
	int a;
	int b;
	Rectangle()
	{}

	Rectangle(int a , int b)
	{
		this.a = a;
		this.b = b;
```

```
            System.out.println("Area of Rectangle : "+ area_Method(this.a , this.b));
        }

        int area_Method(int x , int y)
        {
            int area = x*y;
            return area;
        }
}

class ques5
{
        public static void main(String args[])
        {
            Rectangle r = new Rectangle(5,6);
        }
}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac ques5.java

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ques5
Area of Rectangle : 30
```

P6. Write a Java class Clock for dealing with the day time represented by hours, minutes, and seconds. Your class must have the following features:

- **Three instance variables for the hours (range 0 - 23), minutes (range 0 - 59), and seconds (range 0 - 59).**
- **Three constructors:**
  - **default (with no parameters passed; is should initialize the represented time to 12:0:0)**
  - **a constructor with three parameters: hours, minutes, and seconds.**
  - **a constructor with one parameter: the value of time in seconds since midnight (it should be converted into the time value in hours, minutes, and seconds)**
- **Instance methods:**
  - **a *set*-method method setClock() with one parameter *seconds* since midnight (to be converted into the time value in hours, minutes, and seconds as above).**
  - ***get*-methods getHours(), getMinutes(), getSeconds() with no parameters that return the corresponding values.**
  - ***set*-methods setHours(), setMinutes(), setSeconds() with one parameter each that set up the corresponding instance variables.**
  - **method tick() with no parameters that increments the time stored in a Clock object by one second.**
  - **method addClock() accepting an object of type Clock as a parameter. The method should add the time represented by the parameter class to the time represented in the current class.**
  - **Add an instance method toString() with no parameters to your class. toString() must return a String representation of the Clock object in the form "(hh:mm:ss)", for example "(03:02:34)".**
  - **Add an instance method tickDown() which decrements the time stored in a Clock object by one second.**
  - **Add an instance method subtractClock() that takes one Clock parameter and returns the difference between the time represented in the current Clock object and the one represented by the Clock parameter. Difference of time should be returned as an clock object.**

Write a separate class ClockDemo with a main() method. The program should:

- **instantiate a Clock object firstClock using one integer *seconds* since midnight obtained from the keyboard.**
- **tick the clock ten times by applying its *tick()* method and print out the time after each tick.**
- **Extend your code by appending to it instructions instantiating a Clock object secondClock by using three integers (hours, minutes, seconds) read from the keyboard.**
- **Then tick the clock ten times, printing the time after each tick.**
- **Add the secondClock time in firstClock by calling method addClock.**
- **Print both clock objects calling toString method**

Create a reference thirdClock that should reference to object of difference of firstClock and secondClock by calling the method subtractClock().

==========

```
import java.util.Scanner;


class Clock
{
	int hours;
	int minutes;
	int seconds;

	Clock()
	{
		hours = 12;
		minutes = 0;
		seconds = 0;
	}
```

```
	Clock(int hours , int minutes , int seconds)
	{
		this.hours = hours;
		this.minutes = minutes;
		this.seconds = seconds;
	}

	Clock(int seconds)
	{
		int t_seconds = seconds%60;
		int t_minutes = seconds/60;
		int t_hours = t_minutes/60;

		this.seconds = t_seconds;
		this.minutes = t_minutes%60;
		this.hours = t_hours;
	}

	//Methods
	void setClock(int seconds)
	{
		int t_seconds = seconds%60;
		int t_minutes = seconds/60;
		int t_hours = t_minutes/60;
```

```
            this.seconds = t_seconds;
            this.minutes = t_minutes%60;
            this.hours = t_hours;
        }

        int getHour()
        {
            return hours;
        }

        int getMinutes()
        {
            return minutes;
        }

        int getSeconds()
        {
            return seconds;
        }

        void setHours(int hours)
        {
            this.hours = hours;
        }
```

```
	void setMinutes(int minutes)
	{
		this.minutes = minutes;
	}

	void setSeconds(int seconds)
	{
		this.seconds = seconds;
	}

	void tick()
	{
		int temp=0;
		temp = this.hours*3600 + this.minutes*60 + this.seconds + 1;
		setClock(temp);
	}

	void addClock(Clock x)
	{
		this.hours = this.hours + x.hours;
		this.minutes = this.minutes + x.minutes;
		this.seconds = this.seconds + x.seconds;
	}
```

```
	public String toString()
	{
		return "(" +hours+":"+minutes+":"+seconds+")";
	}

	void tickDown()
	{
		int temp=0;
		temp = this.hours*3600 + this.minutes*60 + this.seconds - 1;
		setClock(temp);
	}

	Clock subtactClock(Clock x)
	{
		Clock y = new Clock();
		y.hours = this.hours - x.hours;
		y.minutes = this.minutes - x.minutes;
		y.seconds = this.seconds - x.seconds;

		return y;
	}

}
```

```
class ClockDemo
{
	public static void main(String args[])
	{
		Scanner sc = new Scanner(System.in);

		System.out.print("Enter total seconds to set firstClock : ");
		int fc = sc.nextInt();
		Clock firstClock = new Clock(fc);

		for(int i=0;i<10;i++)
		{
			firstClock.tick();

			System.out.println(firstClock.toString());

			System.out.println("====================");
		}

		System.out.println("Enter hours, minutes , seconds to set secondClock : ");
		System.out.print("Hours : ");
		int sc_hour = sc.nextInt();

		while(!(sc_hour>=0 && sc_hour<24))
		{
			System.out.println("Please Enter Hour in the range of 00-23");
```

```
                System.out.print("Hours : ");
                sc_hour = sc.nextInt();
        }

        System.out.print("Minutes : ");
        int sc_minute = sc.nextInt();

        while(!(sc_minute>=0 && sc_minute<=60))
        {
                System.out.println("Please Enter Minute in the range of 00-60");
                System.out.print("Minutes : ");
                sc_minute = sc.nextInt();
        }

        System.out.print("Seconds : ");
        int sc_second = sc.nextInt();

        while(!(sc_second>=0 && sc_second<=60))
        {
                System.out.println("Please Enter Seconds in the range of 00-60");
                System.out.print("Second : ");
                sc_second = sc.nextInt();
        }

        Clock secondClock = new Clock(2,40,33);
```

```
            for(int i=0;i<10;i++)
            {
                secondClock.tick();

                System.out.println(secondClock.toString());

                System.out.println("====================");
            }

            secondClock.addClock(firstClock);

            System.out.println(firstClock.toString());

            System.out.println(secondClock.toString());

            Clock thirdClock = secondClock.subtactClock(firstClock);

            System.out.println("Third Clock : " + thirdClock.toString());
        }
}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac ClockDemo.java

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ClockDemo
Enter total seconds to set firstClock : 7200
(2:0:1)
====================
(2:0:2)
====================
(2:0:3)
====================
(2:0:4)
====================
(2:0:5)
====================
(2:0:6)
====================
(2:0:7)
====================
(2:0:8)
====================
(2:0:9)
====================
(2:0:10)
====================
Enter hours, minutes , seconds to set secondClock :
Hours :
```

==========

P7. Write a Java class Complex for dealing with complex number. Your class must have the following features:

- **Instance variables :**
  - **realPart for the real part of type double**
  - **imaginaryPart for imaginary part of type double.**
- **Constructor:**
  - **public Complex (): A default constructor, it should initialize the number to 0, 0)**
  - **public Complex (double realPart, double imaginaryPart): A constructor with parameters, it creates the complex object by setting the two fields to the passed values.**
- **Instance methods:**
  - **public Complex add (Complex otherNumber): This method will find the sum of the current complex number and the passed complex number. The methods returns a new Complex number which is the sum of the two.**

- **public Complex subtract (Complex otherNumber): This method will find the difference of the current complex number and the passed complex number. The methods returns a new Complex number which is the difference of the two.**
- **public Complex multiply (Complex otherNumber): This method will find the product of the current complex number and the passed complex number. The methods returns a new Complex number which is the product of the two.**
- **public Complex divide (Complex otherNumber): This method will find the ... of the current complex number and the passed complex number. The methods returns a new Complex number which is the ... of the two.**
- **public void setRealPart (double realPart): Used to set the real part of this complex number.**
- **public void setImaginaryPart (double realPart): Used to set the imaginary part of this complex number.**
- **public double getRealPart(): This method returns the real part of the complex number**
- **public double getImaginaryPart(): This method returns the imaginary part of the complex number**
- **public String toString(): This method allows the complex number to be easily printed out to the screen**

Write a separate class ComplexDemo with a main() method and test the Complex class methods.

==========

```
class Complex

{

        double realPart;

        double imaginaryPart;



        public Complex()

        {
```

```
        this.realPart = 0;
        this.imaginaryPart = 0;
}

public Complex(double realPart , double imaginaryPart)
{
        this.realPart = realPart;
        this.imaginaryPart = imaginaryPart;
}

public Complex Complexadd(Complex otherNumber)
{
        Complex x = new Complex();
        x.realPart = this.realPart + otherNumber.realPart;
        x.imaginaryPart = this.imaginaryPart + otherNumber.imaginaryPart;
        return x;
}

public Complex Complexsub(Complex otherNumber)
{
        Complex x = new Complex();
        x.realPart = this.realPart - otherNumber.realPart;
```

```
        x.imaginaryPart = this.imaginaryPart - otherNumber.imaginaryPart;
        return x;
}

public Complex Complexmul(Complex otherNumber)
{
        Complex x = new Complex();
        x.realPart = this.realPart * otherNumber.realPart;
        x.imaginaryPart = this.imaginaryPart * otherNumber.imaginaryPart;
        return x;
}

        public Complex Complexdiv(Complex otherNumber)
{
        Complex x = new Complex();
        x.realPart = this.realPart / otherNumber.realPart;
        x.imaginaryPart = this.imaginaryPart / otherNumber.imaginaryPart;
        return x;
}

public void setRealPart(double realPart)
{
```

```
        this.realPart = realPart;
}

public void setImaginaryPart(double imaginaryPart)
{
        this.imaginaryPart = imaginaryPart;
}

public double getRealPart()
{
        return this.realPart;
}

public double getImaginaryPart()
{
        return this.imaginaryPart;
}

public String toString()
{
        return this.realPart + " + i" + this.imaginaryPart;
}
```

```
}

class ComplexDemo
{
	public static void main(String args[])
	{
		Complex c1 = new Complex();
		Complex c2 = new Complex(2,3);

		System.out.println(c1.getRealPart() + " + i" + c1.getImaginaryPart());
		System.out.println(c2.getRealPart() + " + i" +c2.getImaginaryPart());

		c1.setRealPart(12);
		c1.setImaginaryPart(5);
		System.out.println(c1.getRealPart() + " + i" +c1.getImaginaryPart());

		Complex c3 = c1.Complexadd(c2);

		System.out.println(c3.getRealPart() + " + i"  +c3.getImaginaryPart());

		System.out.println("Using Stringto Method");
```

```
        System.out.println(c1);


    }
}
```

```
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac ComplexDemo.java

C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java ComplexDemo
0.0 + i0.0
2.0 + i3.0
12.0 + i5.0
14.0 + i8.0
Using Stringto Method
12.0 + i5.0
```

P8. Write a Java class Author with following features:

- **Instance variables :**
  - **firstName for the author's first name of type String.**
  - **lastName for the author's last name of type String.**
- **Constructor:**
  - **public Author (String firstName, String lastName): A constructor with parameters, it creates the Author object by setting the two fields to the passed values.**
- **Instance methods:**
  - **public void setFirstName (String firstName): Used to set the first name of author.**
  - **public void setLastName (String lastName): Used to set the last name of author.**
  - **public double getFirstName(): This method returns the first name of the author.**
  - **public double getLastName(): This method returns the last name of the author.**
  - **public String toString(): This method printed out author's name to the screen**

Write a Java class Book with following features:

- **Instance variables :**
  - **title for the title of book of type String.**

- **author for the author's name of type String.**
- **price for the book price of type double.**

· **Constructor:**

- **public Book (String title, Author name, double price): A constructor with parameters, it creates the Author object by setting the the fields to the passed values.**

· **Instance methods:**

- **public void setTitle(String title): Used to set the title of book.**
- **public void setAuthor(String author): Used to set the name of author of book.**
- **public void setPrice(double price): Used to set the price of book.**
- **public double getTitle(): This method returns the title of book.**
- **public double getAuthor(): This method returns the author's name of book.**
- **public String toString(): This method printed out book's details to the screen**

Write a separate class BookDemo with a main() method creates a Book titled "Developing Java Software" with authors Russel Winderand price 79.75. Prints the Book's string representation to standard output (using System.out.println).

==========

```
interface metho1

{

        public void setFirstName(String firstName);

        public void setLastName(String lastName);

        public String getFirstName();

        public String getLastName();

        public String toString();

}


class Author implements metho1
```

```
{
        String firstName;
        String lastName;

        Author()
        {}

        Author(String firstName , String lastName)
        {
                this.firstName = firstName;
                this.lastName = lastName;
        }

        public void setFirstName(String firstName)
        {
                this.firstName = firstName;
        }

        public void setLastName(String lastName)
        {
                this.lastName = lastName ;
        }
```

```
	public String getFirstName()
	{
		return firstName;
	}

	public String getLastName()
	{
		return lastName;
	}

	public String toString()
	{
		return this.firstName + " "+ this.lastName;
	}

}

interface metho2
{
	public void setTitle(String title);
	public void setAuthor(String author);
```

```
        public void setPrice(double price);
        public void getTitle();
        public void getAuthor();
        public String toString();
}

class Book
{
        String title;
        String author;
        double price;

        public Book()
        {}

        public Book(String title , String author , double price)
        {
                this.title = title;
                this.author = author;
                this.price = price;
        }
```

```
public void setTitle(String title)
{
        this.title = title;
}

public void setAuthor(String author)
{
        this.author = author;
}

public void setPrice(double price)
{
        this.price = price;
}

public String getTitle()
{
        return this.title;
}

public String getAuthor()
{
```

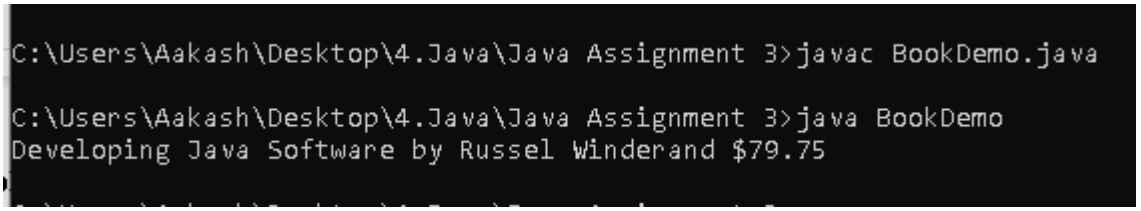
```
        return this.author;
    }

    public String toString()
    {
        return this.title +" by " + this.author +" $"+this.price;
    }

}

class BookDemo
{
    public static void main(String args[])
    {
        Book b = new Book("Developing Java Software","Russel Winderand",79.75);

        System.out.println(b);
    }
}
```


C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>javac BookDemo.java
C:\Users\Aakash\Desktop\4.Java\Java Assignment 3>java BookDemo
Developing Java Software by Russel Winderand $79.75