SESSION - II

Part I

Lab Number 5: Strings

Lab Exercises

1. Design a class which represents a student. Every student record is made up of the following fields.

i) Registration number (int)

ii) Full Name (String)

iii) Date of joining (Gregorian calendar)

iv) Semester (short)

v) GPA (float)

vi) CGPA (float)

Whenever a student joins he will be given a new registration number. Registration number is calculated as follows. If year of joining is 2012 and he is the 80th student to join then his registration number will be 1280.

Write member functions to do the following.

a) Provide default and parameterized constructors to this class

b) Write display method which displays the record. Test the class by writing suitable main method.

c) Create an array of student record to store minimum of 5 records in it. Input the records and display them.

Code:

```
import java.util.* ;

class Student{

        int reg ;

        String name ;

        short sem ;
```

```
float gpa ;
float cgpa ;
Date date ;
static int counter = 0 ;
public Student(){
        counter++ ;
        reg = 0 ;
        name = "" ;
        sem = 0 ;
        gpa = 0.0f ;
        cgpa = 0.0f ;
        GregorianCalendar gc = new GregorianCalendar();
        this.date = gc.getTime() ;
        int year = date.getYear();
        System.out.println(year+1900) ;

}
public Student(String fullname, short semester, float gpa , float cgpa ){
        counter++ ;
        sem = semester ;
        name = fullname ;
        this.gpa = gpa ;
        this.cgpa = cgpa ;
        GregorianCalendar gc = new GregorianCalendar();
        this.date = gc.getTime() ;
        this.reg = ((date.getYear() +1900)%100)*100 + counter ;


}
```

```
	void display(){
		System.out.println("Details of the student are") ;
		System.out.println("Name: "+name+"\t GPA: "+gpa+"\tCGPA: "+cgpa+"\t Date of Joining: "+date+"\tReg_No: "+reg);
	}
}
public class studentdemo{
	public static void main(String[] args){
		Student[] stu = new Student[5] ;
		Scanner sc = new Scanner(System.in);
		float gp,cg ;
		short se ;
		String name = "" ;
		for(int i = 0 ; i<5 ; i++){
			System.out.println("Enter the name of the student: ");
			name = sc.nextLine();
			System.out.println("Enter GPA");
			gp= sc.nextFloat();
			System.out.println("Enter CGPA");
			cg = sc.nextFloat();
			System.out.println("Enter Semester");
			se = sc.nextShort();
			stu[i] = new Student(name,se,gp,cg);
			sc.nextLine();
		}
		for(int i = 0 ; i<5 ; i++){
			stu[i].display();
			System.out.println();
		}
	}
}
```

Test Case:

```
student@lplab-Lenovo-Product:~/190905513$ java studentdemo
Enter the name of the student:
Danish
Enter GPA
8
Enter CGPA
9
Enter Semester
3
Enter the name of the student:
Nalin
Enter GPA
8
Enter CGPA
10
Enter Semester
3
Enter the name of the student:
Abhinav
Enter GPA
9
Enter CGPA
10
Enter Semester
3
Enter the name of the student:
Raghav
Enter GPA
10
Enter CGPA
10
Enter Semester
3
Enter the name of the student:
Srisai
Enter GPA
8
Enter CGPA
9
Enter Semester
3
```

```
Enter the name of the student:
Raghav
Enter GPA
10
Enter CGPA
10
Enter Semester
3
Enter the name of the student:
Srisai
Enter GPA
8
Enter CGPA
9
Enter Semester
3
Details of the student are
Name: Danish     GPA: 8.0       CGPA: 9.0       Date of Joining: Sat Feb 13 08:38:15 IST 2021  Reg_No: 2101

Details of the student are
Name: Nalin      GPA: 8.0       CGPA: 10.0      Date of Joining: Sat Feb 13 08:38:26 IST 2021  Reg_No: 2102

Details of the student are
Name: Abhinav    GPA: 9.0       CGPA: 10.0      Date of Joining: Sat Feb 13 08:38:37 IST 2021  Reg_No: 2103

Details of the student are
Name: Raghav     GPA: 10.0      CGPA: 10.0      Date of Joining: Sat Feb 13 08:38:48 IST 2021  Reg_No: 2104

Details of the student are
Name: Srisai     GPA: 8.0       CGPA: 9.0       Date of Joining: Sat Feb 13 08:39:05 IST 2021  Reg_No: 2105

student@lplab-Lenovo-Product:~/190905513$
```

Lab No. 7: Interfaces and Multithreading

Lab Exercises

5. Design an interface called Series with the following methods

a. Get Next (returns the next number in series)

b. reset(to restart the series)

c. set Start (to set the value from which the series should start)

Design a class named By Twos that will implement the methods of the interface Series such that it generates a series of numbers, each two greater than the previous one. Also design a class which will include the main method for referencing the interface.

Code:

```
import java.util.* ;
interface Series {
        int getnext() ;
        void reset() ;
        void setstart(int x) ;

}
class ByTwos implements Series {
        int start ;
        int current ;
        public ByTwos(){
                start = 0 ;
                current = 0;
        }
        public int getnext(){
                current+=10 ;
                return current ;
        }
        public void reset(){
                current = start ;
```

```
        }
        public void setstart(int x){
                current = x ;
                start = x ;
        }
}

public class series{
        public static void main(String[] args){
                ByTwos bt = new ByTwos();
                for(int i =0 ; i<3 ; i++)
                        System.out.println("Next is: " +bt.getnext()) ;
                ByTwos bt1 = new ByTwos() ;
                bt1.setstart(7);
                for(int i =0 ; i<3 ; i++)
                        System.out.println("Next is: " +bt1.getnext()) ;
                bt1.reset();
                System.out.println("Value after reset is: "+ bt1.current) ;
        }

}
```

Test Case: