LAB # 8

OBJECTIVE

To study inheritance and using super to call superclass constructors

Question:

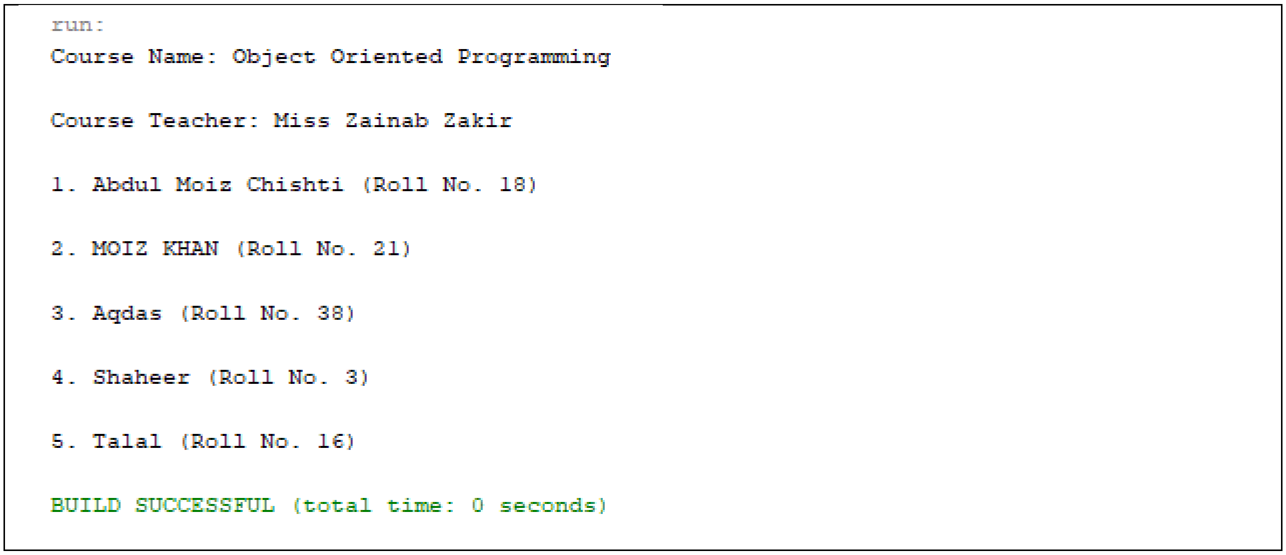
Develop a registration system for a University. It should consist of three classes namely Student, Teacher, and Course. For example, a student needs to have a name, roll number, address and GPA to be eligible for registration. Therefore choose appropriate data types for encapsulating these properties in a Student objects. Similarly a teacher needs to have name, address, telephone number, and a degree (or a list of degrees) he/she has received. Finally courses must have a name, students (5 students) registered for the course, and a teacher assigned to conduct the course. Create an object of Course with 5 Students and a Teacher. A call to a method, say printDetails(), of the selected course reference should print name of the course, details of the teacher assigned to that course, and names and roll numbers of the students enrolled with the course

Source Code:

```
package lab8task1;

/**
 *
 * @author Abdul Moiz Chishti
 */
public class Lab8Task1 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        Student std[] = new Student[5];
std[0] = new Student("Abdul Moiz Chishti",022,"London ",3.5);
std[1] = new Student("MOIZ KHAN",21,"Gulshan",4);
std[2] = new Student("Aqdas",38,"13-c",3.86);
std[3] = new Student("Shaheer",003,"lahore",3.6);
std[4] = new Student("Talal",16,"Islamabad",3.8);


Teacher teach = new Teacher( "Miss Zainab Zakir", "A-123, Juhar", "033*-*******", " Software Eng" );
```

```
Course c = new Course( "Object Oriented Programming", teach, std );

c.printDetails(); }}



class Student { String name; int rollNo; String add; double gpa;
    Student ( String name, int rollNo, String add, double gpa){
        this.name = name;
        this.rollNo = rollNo;
        this.add = add;
        this.gpa = gpa;

    }
   }
class Teacher { String name; String add; String tel; String deg;
        Teacher ( String name, String add, String tel, String deg ){
 this.name = name;
 this.add = add;
 this.tel = tel;
 this.deg = deg;


        }
}

class Course { String name;
    Student std[] = new Student[5];
    Teacher teach;


    Course ( String name, Teacher teach, Student std[] )
{
 this.name = name;
 this.std = std;
 this.teach = teach;

}

    void printDetails(){
 System.out.println("Course Name: "+this.name+"\n");
 System.out.println("Course Teacher: "+this.teach.name+"\n");
 for ( int i = 0 ; i < 5 ; i++ ){
 System.out.println((i+1)+". "+this.std[i].name+" (Roll No. "+this.std[i].rollNo+")"+"\n");
 }
}
}
```

Output:



```
run:
Course Name: Object Oriented Programming

Course Teacher: Miss Zainab Zakir

1. Abdul Moiz Chishti (Roll No. 18)

2. MOIZ KHAN (Roll No. 21)

3. Aqdas (Roll No. 38)

4. Shaheer (Roll No. 3)

5. Talal (Roll No. 16)

BUILD SUCCESSFUL (total time: 0 seconds)
```

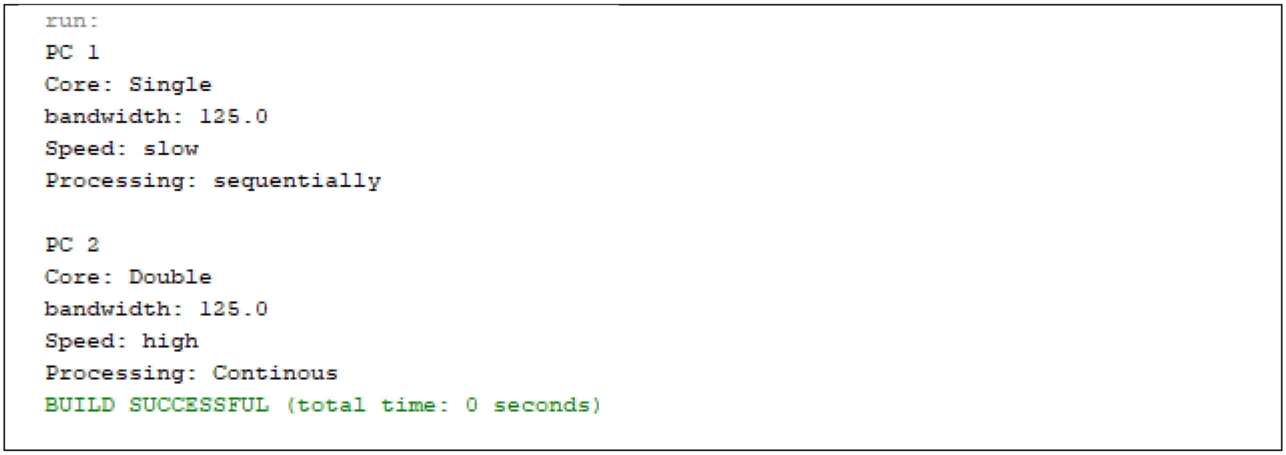
Question:

Create a class called computers and two classes MyComputer and YourComputer which inherits computer class. Define appropriate features of their processor in the classes. Create another class processor as a composite Inheritance class of computer. Write a method which prints the differences between the processors of two computers. Example: single core: bandwidth = 125GByte/s speed = slow processing = sequentially

Source Code:

```
package lab8task2;

/**
 *
 * @author Abdul Moiz Chishti
 */
public class Lab8Task2 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    System.out.println("PC 1");
    MyComputer mc=new MyComputer();
    mc.processing("Single", 125, "slow", "sequentially");
    mc.configuration();

    System.out.println("\nPC 2");
    YourComputer yc=new YourComputer();
    yc.processing("Double", 125, "high", "Continous");
    yc.configuration();



    }

}
public class MyComputer extends Computer {
     public void processing(String core,double bw,String speed,String prg){

        this.core=core;
        this.bw=bw;
        this.speed=speed;
        this.prg=prg;


    }
    public void configuration(){
        System.out.println("Core: "+core);
        System.out.println("bandwidth: "+bw);
        System.out.println("Speed: "+speed);
        System.out.println("Processing: "+prg);

    }
}
public class YourComputer extends Computer {
     public void processing(String core,double bw,String speed,String prg){

        this.core=core;
        this.bw=bw;
        this.speed=speed;
        this.prg=prg;


    }
    public void configuration(){
        System.out.println("Core: "+core);
        System.out.println("bandwidth: "+bw);
        System.out.println("Speed: "+speed);
        System.out.println("Processing: "+prg);

    }

}
public class Processor {



}
public class Computer {

    Processor obj =new Processor();

    public String core;double bw;String speed;String prg;
   public void processing (){
   }
}
```

Output:



```
run:
PC 1
Core: Single
bandwidth: 125.0
Speed: slow
Processing: sequentially

PC 2
Core: Double
bandwidth: 125.0
Speed: high
Processing: Continous
BUILD SUCCESSFUL (total time: 0 seconds)
```