**Lab Exercise**

**Objectives:**

Objectives of this lab exercise are as follows;

Learning…

1. How to use Inheritance
2. How to use polymorphism
3. How to use function overriding and overloading
4. How to declare interfaces and Abstract classes.
5. Usage of getter and setter.
6. Write a console based program to implement polymorphism using inheritance. Consider the example of Shape as base class with method show(). And then a child class Circle and Rectangle which inherit the base class Shape and override its method ***show()*.** Add one more Method with the name of getInfo(). This method would display the class name in which it is implemented. Do not override this method. When you will call the method ***getInfo()*** with child object it would still show the name of the base class, which implies that method has been directly inherited and was not overridden.
7. Repeat the same example in Question No.1 and make the Shape class Abstract and then implement its methods in Derived classes Circle and Rectangle. This time keep the method ***show()*** as abstract. And you would notice if an abstract method is not implemented in derived class then you get a compile time error.
8. Write a console based application. Declare a class Student. Declare its private data members as ***strStudentName***, ***iStudentRegistrationNo***, ***strBatchName***. Write getter and setter for these data members. Set and Get the values using these Properties. Use ***this*** keyword while setting the values.
9. Write a subclass called **SubClass** that is derived from **SuperClass** and that adds an integer data field called data2 and a public method called ***checkCondition***() that will check if data1 is equal to 10 and data2 is equal to 15, the ***checkCondition*** () method should return “Condition True!”.

Also, create methods called ***setData2***() and ***getData2***() for setting and retrieving the value of data1 and data2, as well as a constructor that accepts arguments for the starting values of data1 and data2. data1 is data member of SuperClass.

1. Write a Console based application. Create a class ***MyMath***. Within this Class write three overloaded methods ***add ()*.** The First method takes two integer parameters and adds them and returns result. The second overloaded method takes three parameters and adds them. The third one takes four parameters and adds them and returns the sum.
2. Write a java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.
3. Write the interface Drawable with only one abstract method draw() and the interface Measurable with abstract methods getArea() and getPerimeter(). Write a class that implements these two interfaces.