**Encapsulation / Abstraction**

Exercise 1:

Create a class called Author is designed as follows:

It contains:

• Three private instance variables: name (String), email (String), and gender (char of either ‘m’ or ‘f’);

• One constructor to initialize the name, email and gender with the given values;

And, a class called Book is designed as follows:

It contains:

• Four private instance variables: name (String), author (of the class Author you have just created), price (double), and qtyInStock (int). Assuming that each book is written by one author.

• One constructor which constructs an instance with the values given.

• Getters and setters: getName(), getAuthor(), getPrice(), setPrice(), getQtyInStock(), setQtyInStock(). Again there is no setter for name and author.

Write the class Book (which uses the Author class written earlier).

Try:

1. Printing the book name, price and qtyInStock from a Book instance. (Hint: aBook.getName())

2. After obtaining the “Author” object, print the Author (name, email & gender) of the book.

#### ****Inheritance****

Exercise 2:

Declare one same variable in both parent and child class and try to access the variable of parent class in child class.

Exercise 3:

Create a class called Person with a member variable name. Save it in a file called Person.java

Create a class called Employee who will inherit the Person class.The other data members of the employee class are annual salary (double), the year the employee started to work, and the national insurance number which is a String.Save this in a file called Employee.java

Your class should have a reasonable number of constructors and accessor methods.

Write another class called TestEmployee, containing a main method to fully test your class definition.

Exercise 4:

A HighSchool application has two classes: the Person superclass and the Student subclass. Using inheritance, in this lab you will create two new classes, Teacher and CollegeStudent. A Teacher will be like Person but will have additional properties such as salary (the amount the teacher earns) and subject (e.g. “Computer Science”, “Chemistry”, “English”, “Other”). The CollegeStudent class will extend the Student class by adding a year (current level in college) nd major (e.g. “Electrical Engineering”, “Communications”, “Undeclared”).

#### ****Overriding / Polymorphism****

Exercise 5:

Write a program and use overloaded methods for printing different types of array (integer, double and character).

Exercise 6:

Create a base class Fruit which has name ,taste and size as its attributes. A method called eat() is created which describes the name of the fruit and its taste. Inherit the same in 2 other class Apple and Orange and override the eat() method to represent each fruit taste.

Exercise 7:

Write a program to create a class named shape. In this class we have three sub classes circle, triangle and square each class has two member function named draw () and erase (). Create these using polymorphism concepts.

#### ****String,StringBuffer****

Exercise 8:

Write a Program that will check whether a given String is Palindrome or not

Exercise 9:

Write a Program to understand the difference between == and equals method in a String

Exercise 10:

Write a Program which displays a menu, with 2 options. 1. String Functions and 2. StringBuffer functions.If option1 is chosen, it should demonstrate the use of 5 String functions. If option 2 is selected, it should demonstrate the use of 5 StringBuffer functions