30/07/2023 *AUTOMATED TESTING - OO Programming*

Lab practice Using Java programming languages

STUDENT NAME | COURSE NAME | INSTRUCTOR NAME/CLASS TIME/PERIOD

**ASSIGNMENT No. 3 – Using JAVA**

Exercises:

1. Write a java program to create a simple calculator using a do while structure to create an option menu, the program has to ask the user to enter two numbers. Create the methods sum, difference, product, division, and average to perform the operations based on the user choice and these methods have to return the result to print. The program provide option to ask the user if they want to perform a new operation or exit.
2. Create a class Student with the states (attributes): Name, address, phone number, Student ID. Also define the behavior of student as learn, perform assignment, read, attendance, do presentation (Exemple: My name is {Name}). After that create 3 instances of the class Student and present the instance (My name is .....)
3. Create a super class called Car.
4. The Car class has the following fields and methods. ◦int speed; ◦double regularPrice; ◦String color; ◦ double getSalePrice; getSalePrice()
5. Create a sub class of Car class and name it as Truck. The Truck class has the following fields ◦int weight; and method. ◦double getSalePrice(); **//Ifweight > 2000, 10% discount. Otherwise, 20% discount.**
6. Create a subclass of Car class and name it as Sedan. The Sedan class has the following fields and methods. ◦int length; ◦double getSalePrice(); **//Iflength>20feet,5%discount,Otherwise,10%discount**
7. Create MyOwnAutoShop class which contains the main() method. Perform the following within the main() method.

◦Create an instance of Sedan class and initialize all the fields with appropriate values. Use super(...) method in the constructor for initializing the fields of the superclass.

◦Create two instances of the Ford class and initialize all the fields with appropriate values. Use super(...) method in the constructor for initializing the fields of the super class.

◦Create an instance of Car class and initialize all the fields with appropriate values.

Display the sale prices of all instance.

1. Create a class called Book to represent a book. A Book should include four pieces of information as instance variables‐a book name, an ISBN number, an author name and a publisher. Your class should have a constructor that initializes the four instance variables. Provide a mutator method and accessor method (query method) for each instance variable. In addition, provide a method named getBookInfo() that returns the description of the book as a String (the description should include all the information about the book). You should use this keyword in member methods and constructor. Write a test application named BookTest to create 5 objects for 5 elements for class Book to demonstrate the class Book's capabilities.