**COLLEGE OF COMPUTING AND INFORMATICS**

**CSEB5133 / CSEB534 JAVA PROGRAMMING**

**SEM 2 2024/2025**

**LAB 1: OBJECT AND CLASS**

**Objectives**

To test students on their understanding of the concept of Inheritance and Polymorphism.

**Lab Evaluation & Submission Instructions**

1. This is an individual lab exercise.
2. You are compulsory to complete **ALL QUESTIONS for Level Easy and Moderate** exceptLevel Challenging (Self-Lab Revision Exercise).
3. **Your lab exercises will be evaluated and graded only during the lab session**. Once you receive your marks, submit your complete program to the Brighten platform.
4. Late submissions or changes after grading will not be considered.
5. Do attach this code segment in all files:

/\*Subject code : CSEB5133 / CSEB534 Java Programming

Section : 01A or 01B

Student name : XXX

Student ID no: XXX

Question no : XXX \*/

**LEVEL: EASY**

**Question**

*Source: Semester 1, 2013/2014, Midterm Test*

There are two (2) errors in the following program. Identify and fix the errors.

|  |
| --- |
| public class ShowErrors {  public static void main(String[] args) {  C c = new C(5);  System.out.println(c.value);  }  }    class C {  private int value;  } |

Tasks:

1. Fix the errors without using an accessor method.
2. Fix the errors by using an accessor method.
3. Rewrite your solution in (ii) by using mutator method.

[5 marks]

**LEVEL: MODERATE**

**Question**

*Source: Section C, Final examination SEMESTER II 2013/2014*

1. Create a class named HotelRoom that includes an integer field for the room number and a double field for the nightly rental rate. Include get and set methods for these fields and a constructor that requires an integer argument representing the room number. The constructor sets the room rate based on the room number; rooms numbered 299 and below are RM190.00 per night, and others are RM250.00 per night. Include a no-argument method to display all fields’ value of a HotelRoom object. For example, an invocation to the display() method prints this sentence: "Regular hotel room – room number 288, rental rate is RM190.00 per night."

[10 marks]

1. Create a subclass named Suite whose constructor requires a room number and adds a RM150.00 surcharge to the regular hotel room rate, which again is based on the room number. Overrides the method display() in superclass so that it prints relevant information of a Suite room. For example, "Suite hotel room – room number 318, rental rate is RM400.00 per night."

[6 marks]

1. Write an application named UseHotelRoom that creates an object of each class (use your own values to set to the object, user input is not required). Invoke the method display() from each object you created.

[4 marks]

**LEVEL: CHALLENGING (SELF-LAB REVISION EXERCISE)**

**Question**

*Source: Final Examination Sem 1 20132014*

1. A class called Author is designed to contain:

* Three *private* instance variables: name(String), email(String) and gender(char of either ‘m’ or ‘f’)
* One constructor to initialize the name, emailand gende*r* with its parameter values.
* Acessor methods for all fields and mutator methods only for *email* (no mutator for nameand genderas these attributes cannot be changed)
* A toString()method that returns "-*author-name- (-gender-) at –email-*", e.g., "Khadijah Osman (f) at khadijaho@gmail.com".

1. A class called Book is designed to contain:

* Four *private* instance variables: name(*String*), author(of the class Author, assume that each book has one and only one author), price(*double*), and qtyInStock(*int*).
* Two constructors:
* public Book (String name, Author author, double price) {...}
* public Book (String name, Author author, double price, int qtyInStock) {...}
* public methods getName(), getAuthor(), getPrice(), setPrice(), getQtyInStock(), setQtyInStock().
* toString() that returns "-book-name- by -author-name- $-price-", e.g., "Programming with Java by Khadijah Osman $62.00".

1. Write the Authorclass
2. Write the Bookclass
3. Write a test program called TestBook to do the following:
   1. Create two instances of Book using both of the constructors. Use your own values.
   2. Print the name and email of the author from both of the Book instances.
   3. Using the toString() method of Book, print the information of both books.