

## **PART 1: MULTIPLE CHOICE QUESTIONS**

*Each question is worth two marks. Indicate answers in the FIRST PAGE of the answer booklet provided.*

### **Question 1**

Answer the following question based on the hierarchy given below:

```
public interface Chargeable { }  
public abstract class Parcel implements Chargeable { }  
public class Present extends Parcel { }
```

Which of the following Java statement will compile **correctly**?

- a) `Parcel c = new Parcel();`
- b) `Chargeable c = new Present();`
- c) `Parcel p = new Chargeable();`
- d) `Present p = new Parcel();`

### **Question 2**

Inheritance indicates a/an \_\_\_\_\_ relationship.

- a) is-a
- b) has-a
- c) both A & B
- d) none of the above

### **Question 3**

Java classes can

- a) extend multiple superclasses and implement multiple interfaces
- b) extend only one superclass and implement multiple interfaces
- c) extend multiple superclasses and implement a single interface
- d) extend only one superclass and implement a single interface

### **Question 4**

The try blocks contain code that could possibly

- a) throw an exception
- b) catch an exception
- c) display an exception
- d) none of the above

### **Question 5**

When the method `readLine()` of a `BufferedReader` object tries to read beyond the end of a text file, it

- a) throws `IOException`
- b) throws `EndOfFileException`
- c) returns a value of -1
- d) returns a value of null

### **Question 6**

Which of the following is classified as checked exception?

- a) `Exception`.
- b) `ClassCastException`.
- c) `NullPointerException`.
- d) `NumberFormatException`.

### Question 7

Which of the following classes enable input and output of entire objects to or from a file?

- I. SerializedInputStream      II. SerializedOutputStream      III. FileReader  
IV. ObjectInputStream      V. ObjectOutputStream      VI. FileWriter

- a) I and II
- b) IV and V
- c) III and VI
- d) I, II, IV, and V

### Question 8

Given the following code:

```
class MyThread extends Thread {  
    public static void main(String [] args) {  
        MyThread t1 = new MyThread();  
        MyThread t2 = new MyThread();  
        t1.start();  
        System.out.print("one. ");  
        t2.start();  
        System.out.print("two. ");  
    }  
  
    public void run() {  
        System.out.print("Thread ");  
    }  
}
```

What is the result of this code?

- a) Compilation fails
- b) An exception occurs at runtime.
- c) Thread one. Thread two.
- d) The output cannot be determined.

### Question 9

Given the following code:

```
public class MyRunnable implements Runnable {  
    public void run() {  
        // some code here  
    }  
}
```

Which of these will create and start this thread?

- a) new Runnable(MyRunnable).start();
- b) new Thread(MyRunnable).run();
- c) new Thread(new MyRunnable()).start();
- d) new MyRunnable().start();

### Question 10

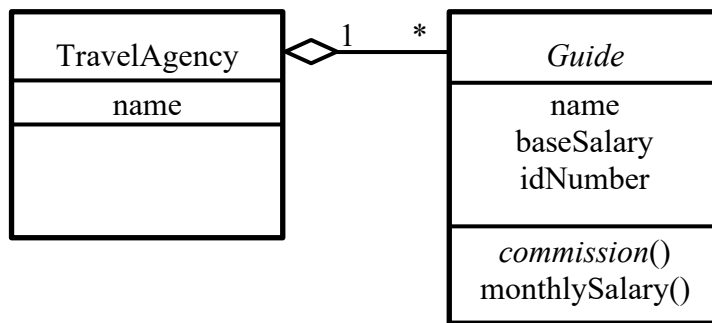
Choose the **incorrect** statement from the ones below, in respect of graphics components:

- a) many Swing components are written in Java;
- b) AWT components rely heavily on the native operating system;
- c) components that rely on the native operating system are referred to as lightweight components.
- d) Swing components look the same no matter what operating system they are being used on;

## PART 2: WRITTEN QUESTIONS

### Question 1

Examine the UML class diagram provided below for a Guide class of a travel agency. Each tour guide has a **name**, of type String, a **base salary** of type double, and a unique 3-digit **id number**, of type integer, from 100 to 999. The class TravelAgency is responsible for the creation of Guide objects. It therefore allocates a unique identification number for each new Guide object.

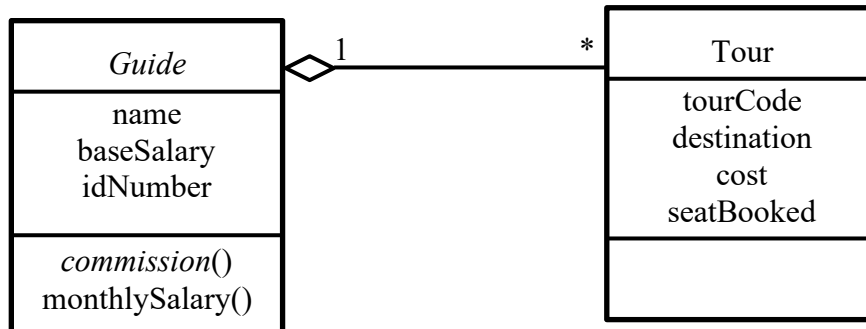


The Guide class is an abstract class, and contains an abstract method named *commission*, which returns the commission earned by each guide. The subclasses of Guide class will have its own implementation of computing the commission. We also need a method that returns the *monthly salary* of a guide, which is the sum of the *baseSalary* and *commission* earned for the month.

- Write the source code for the Guide class, by providing
  - class header, and declaration of appropriate attributes;
  - a constructor which accepts values for all three attributes
  - a reader (getter) method and a writer (setter) method for each of the attributes.
- Define a method in Guide which overrides the `toString` method inherited from class Object. The `toString` method should return the **name** and **id number** of the guide, as well as the **monthly salary** as shown below:  
Guide named `<name>` (`<idNumber>`) earns \$`[monthlySalary]` per month.  
For example,  
Guide named **James Bond** (**123**) earns \$**3550.00** per month.
- Define a method in Guide which overrides the `equals` method of class Object.  
**NOTE:** Two guides are deemed to be equal if they have the same **id number**.
- Do not forget to include the abstract method, *commission*, and the method `monthlySalary`.

## Question 2

The following diagram shows an aggregation (whole-part) relationship between `Guide` and `Tour`. A **collection** would be required inside `Guide` to implement this design.



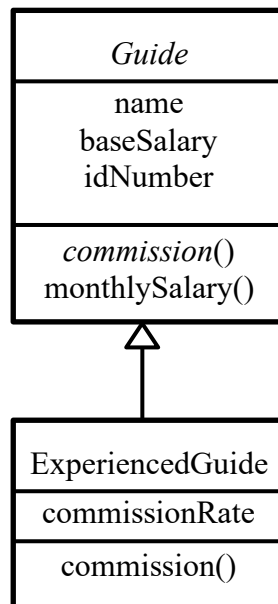
**Choose one** of the collection types: **Set**, **List** and **Map**, and write the **additional source code** needed in the `Guide` class to implement the collection of `Tours`:

- Specify additional attribute, and its initialization, as well as the corresponding setter, and getter method
- Include a method in `Guide` which allows a `Tour` object to be added to the collection.
- Include a method which accepts a `String` representing a tour code, and returns the tour matching this tour code. If there is no such tour with the parametric tour code, *null* should be returned. You must use stream and lambda expression to complete this task.
- Include a method which returns a string containing the details of all tours led by the guide, one per line. This method accepts a string as parameter, which determine whether the tours should return in original order or sorted according to cost. You must use stream and lambda expression to complete this task.

**NOTE:** Class `Tour` provides the attributes of a tour organised by the travel agency. You do NOT need to create the `Tour` class. You may assume that the class have been implemented correctly, and made available to you. The class has the methods: constructor with four parameters, getter, and setter methods for the respective attributes, and has correctly overridden the `equals` and `toString` methods of class `Object`.

### Question 3

An `ExperiencedGuide` is a special kind of guide, as shown in the following diagram:



An `ExperiencedGuide` will have a **commission rate** (a double, whose value varies from 1% to 10%, inclusive). For each tour that the experienced guide leads, he/she will be given a commission of (total amount of the tour \* commission rate). For example, for a tour of 20 people (i.e. seat booked is 20) and with cost of \$2000.00, and an experienced guide with a commission rate of 2%, then he/she will get a commission of  $20 * 2000 * 0.02 = 800.0$ .

Write the source code for the `ExperiencedGuide` class, by providing

- class header, and declaration of additional attribute;
- a constructor which accepts values for all attributes
- a reader (getter) method and a writer (setter) method for the attribute.
- Implement the `commission` method as explained above. You must use stream and lambda expression to complete this task.