**Advanced Java**

**Assignment 1**

**Name: Matthew Yackiel** **Due: September 2, 2021**

Type all answers below each question. Leave a blank line between the question and its answer. Also leave a blank line before the next question. Give examples were appropriate. Read Chapter 1 in the book from Dr. Lu, my PowerPoint presentation called *Review of OOP Concepts in Java* before starting this assignment. Chapters 1 to 15 in the Deitel book contain topics covered in your first two Java classes if you purchased that book.

1. **What are Java’s two kinds of variables?**

**Primitive Types and Instance Types**

1. **What is Camel Case Naming?**

It is naming where words are seperated with capital letters. ex) oneTwoThreeFour

1. **What is the purpose of a constructor? Does each class have a constructor?**

**To construct objects (instances of the class). Yes every class has a constructor.**

1. **What is the default constructor?**

**It is a parameterless constructor that initializes an object. Usually the fields are set to default values.**

1. **What is the signature of a method?**

**The signature includes the name, the parameters, and the return type of a method.**

1. **What is the difference between overriding and overloading a method?**

**Overriding: Redefining a method from a parent class, such as toString.**

**Overloading: having multiple methods, or constructors with the same name but different signatures. Used to cover more cases.**

1. **What is the difference between a class and an object?**

A class defines object’s behavior and data.j An object is a single instance of a class.

1. **Define each of the following with regards to classes and objects. There are two questions below**. The terms in each part are related.
   1. Fields, Attributes and Instance Variables

Fields are specific to individual objects, they correspond to the attributes defined by the class. Instance variables are variables, that have non-primitive types. Objects of any type.

* 1. **Responsibilities, messages, methods**

1. **What is the difference between a break and continue statement?**

**Break and Continue are flow control structures. Continue is used in loops to continue on to the next iteration without finishing the current iteration. Break will move flow outside of the current code block. So moving to the end of a loop without finishing, or exiting a switch statement, or an if statement.**

1. **What does the keyword static mean in Java?**

**It is a lifetime modifier. Static specifies that this piece of code will exist as long as 1 instance of it exists. It is also shared amonst all instances.**

1. **What does it mean if a variable is declared to be private static final?**

**It cannot be directly accessed, it is shared amongst all instances, and it is immutable.**

1. **What is an enum type?**

**A data type that represents an enumeration. It can have all the structures that a class has or none of them. What it does have is a set of final static values that represent 0, 1, 2, 3, 4, … but are written with an identifier (caps locked). Such as,**

**MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY.**

**Could be an enum type for the days of the week.**

1. **What does Encapsulation and Information Hiding mean with regards to an Object-Oriented Programming Language?**

**Encapsulation is used to make code more reusable. Information hiding occurs when code is trying to reference a variable form outside of it’s scope.**

1. **What is inheritance and how do we use it in Object-Oriented Programming (OOP)?**

Inheritance is used to make a more specific use case for an already existing piece of code. All classes inherit the Objects class. We use it to use an already existing behavior of a Class while adding to, or overloading some of it’s behavior to suit our subclass.

1. **What are super classes and subclasses in OOP?**

**A super class is any class that is being inherited from. While a subclass is the class that inherits from the super class. These are also called Parent class, and Child class respectively.**

1. **What are interfaces and how are they used in object-oriented design and programming?**

**Interfaces are used to create common behaviour amongst a group of classes. For example we may have a Car interface, but have a truck class, a hybrid class, and a sedan class that all have different implementations of the Car interface.**

1. **What is the default access modifier in Java, public, private, or package?**

**Package, the code can be accessed freely within the package but not outside of the package. This is also sometimes called package-private.**

1. **When should you use the class BigDecimal?**

**When you need very specific decimal numbers.**

1. **What does it mean for an object to be immutable?**

**It is immutable if all of it’s Fields are final.**

1. **When would we use the Scanner class?**

**We use the scanner interface for user input**

1. **What is argument promotion and casting?**

**Argument promotion automatically promotes objects to a more specific type. Such as int to double. Casting is used to force a statement to become a specific type.**

1. **When would we use the class SecureRandom?**

Anytime a Random number is needed, escpecially in the case of password generation or cryptography.

1. **What is the java.time package? When was it added to Java?**

**Time is used to represent instants in time, it was added in Java SE 8.**

1. **What is the scope of an identifier?**

**The scope is the region of code where an identifier lives. If you are outside of the identifier’s scope it is as if that identifier does not exist there.**

1. **What does it mean to hide a variable in a class? (Hidden Variables)**

**It is when a subclass does not declare all of it’s parent’s data types. In the subclass those data types are now “hidden”.**

1. **What is the meaning and use of the keyword this?**

**“this” is used to reference a variable inside of a more specific scope, it can also be a reference to an Object that is calling a method, or to a class constructor in the case of a class definition.**

1. **What are the benefits of the get and set methods?**

**You can better ensure data integrity by having checks within set methods to ensure a value isn’t being passed that could throw an error. Get methods are used to prevent direct access to the data itself, only the values that the data holds.**

1. **What is polymorphism? Give an example.**

**Polymorphism describes situation in which something exists in several different forms. For example, method overloading and overriding are both example os polymorphism. The toString method specifically is likely implemented differently for thousands of different classes, but maintains the same signature.**

1. **What is an abstract class and abstract method? When do we use them?**

**An abstract class is a class that has some amount of abstract methods and variables. Abstract methods, are methods without an implementation. These are used to provide abstraction, users of code don’t need to know the exact implementation of it, they simply need to know how to use it.**

1. **What does the instanceof operator do? When do we use it?**

**This operator is used to determine if an Object has the specified Type. It returns a boolean operator and takes the form (object instanceOf Type)**

1. **When do we use exception handling?**
2. **What is the difference between checked and uncheck exceptions?**

Checked exceptions occur at compile time, and unchecked exceptions occur at runtime.

1. **What does it mean to chain an exception?**

**Occurs when code throws another exception, in response to an exception.**

1. **What does it mean to throw early and catch late with regards to exception handling? (see** [**https://howtodoinjava.com/best-practices/java-exception-handling-best-practices/**](https://howtodoinjava.com/best-practices/java-exception-handling-best-practices/)**)**

**You should catch and exception as soon as you can, but deal with it once you have as much information as possible.**

1. **What is the Java Event Queue?**

**This is a platform independat class that queues events sent by other classes.**

1. **When should you use the StringBuilder class instead of the String class? Explain why.**

**When I need to constantly append new pieces to a String it is better to use string builder because StringBuilder handles the procedural construction of Strings more efficiently than a String.**

1. **What is the StringTokenizer class and when do we use it? How does it compare to the Scanner class?**

**I would use StringTokenizer when I need to pull strings that are seperated by a common substring, called a delimiter out of a large piece of text. Where as I would use Scanner if I needed to parse data of different types. Not just strings.**

1. **What is the difference between the java.io and java.nio packages?**

**Java.io is stream oriented where as java.nio is buffer oriented, it stores the data in a buffer while keeping a channel open for 2 way communication. Java.io can only receive OR send data. It cannot do both at once.**

1. **What is the difference between Pass-By-Value and Pass-by-Reference? Why is this relevant to Java?**

**Pass by value means that we are passing the data itself and not the reference to the actual data. Pass-by-reference passes the actual location of the data, so any method that changes that data would change that data for all other structures down the line. Pass-by-value lets us pass an Object into a method without worrying for it’s data integrity.**

1. **How does Java pass parameters? Explain.**

**Java never passes Objects, it only passes a soft copy of the object to a method. Essentially we give the data and behavior that the object has, without passing the acual object itself. This way mehtods won’t change objects. Primitive types are always passed by value.**