Career Services Assignment 3 – Java Flash Cards

Points possible: 50

Category	Criteria	% of Grade
Completeness	All requirements of the assignment are complete.	100

Instructions: Research common Java interview questions online and create 20 flash cards from the information you find. Study your flash cards regularly to better prepare for interviews. Fill out the table below with the information you put on each of your flash cards.

Front of Card	Back of Card
What is JAVA?	High-level platform independent programming language.
What are the features of JAVA?	1. OOP concepts 2. Platform independent 3. High Performance 4. Multi-threaded
How does JAVA enable high performance?	It uses Just In Time compiler that converts instructions to bytecodes.
Name JAVA IDE's.	Eclipse and Netbeans
What is a Constructor?	Constructor is a method in a class with the same name as the class. It gets invoked when a new object corresponding to the class is created. It is created by default if not implicitly created. It can be overloaded (two or more in the same class) but if we create a constructor with parameters, we need to explicitly create a constructor with no parameters.
What is the difference between local variable and instance variable?	Local variable's scope exists within the method in which it was defined. Instance variables are defined outside the methods but inside the class and its scope exists throughout the class.
What is a class?	Class is where all Java code is written, and it includes methods and variables.
What is an Object?	Object is a member or an instance of a class.
What are OOP's concepts?	• Inheritance • Encapsulation • Polymorphism • Abstraction

	• Interface
What is Inheritance?	Inheritance is reusing properties of a class in another class to reduce repeating code. The original class is commonly known as the Super Class and the derived class is known as Sub Class.
What is Encapsulation?	It is a way of restricting users from directly modifying variables or data in a class by changing access modifiers to private.
What is Polymorphism?	It is a way of using a method in different ways. Each class that implements the method can implement it in their own way. The two types of polymorphism are: • Method Overloading (Static) • Method Overriding (Dynamic)
What is Method Overriding?	Type of Polymorphism with conditions: Same method name Same arguments Same return types
What is Method Overloading?	Type of Polymorphism with conditions: Same method name Different argument types Same or different return type
What is Interface?	Interface is a file that contains the method signature and no method body. It is a way to achieve multiple inheritance which is not possible with classes.
What is Abstract Class	It is a class that cannot be instantiated.
What is the difference between Array and Array List?	Array has a fixed size while Array List has dynamic size. Array requires index to add elements to it while Array List uses the add() method
What is the difference between String, StringBuilder and StringBuffer?	-String is immutable/constant so it is used when the data should not change constantly. -StringBuilder is mutable and stores string values in a stack. Strings can be added to and removed from a StringBuilder. -StringBuffer is mutable just like StringBuilder but it is synchronized and thread-safe, hence, is slower than StringBuilder
Explain the scope of four access specifiers.	Public: is accessible to all classes in a package and in other packages. Private: is accessible only within the class. Protected: is accessible to all classes within a package and sub classes in other packages.

	Default: is accessible to all classes within a package.
What are the differences between Lists, Sets and Maps	L**ists:** allow duplicates, add and store elements in order, allow null values **Sets:** do not allow duplicates, are unordered, can have one null value **Maps:** have key-value pairs where the values can be duplicated but the keys cannot **Common Implementations:** • **Lists:** ArrayList, LinkedList, Vector • **Sets:** HashSet, LinkedHashSet, TreeSet • **Maps:** HashMap, LinkedHashMap, TreeMap, Hashtable