JAVA Programming Questions

Q.1 **What is the difference between equals() and == in Java?**

Equals() method is defined in Object class in Java and used for checking equality of two objects defined by business logic.

"==" or equality operator in Java is a binary operator provided by Java programming language and used to compare primitives and objects. *public boolean equals(Object o)* is the method provided by the Object class. The default implementation uses == operator to compare two objects. For example: method can be overridden like String class. equals() method is used to compare the values of two objects.

Q2. What are wrapper classes in Java?

Wrapper classes convert the Java primitives into the reference types (objects). Every primitive data type has a class dedicated to it. These are known as wrapper classes because they "wrap" the primitive data type into an object of that class. Refer to the below image which displays different primitive type, wrapper class and constructor argument.

Q.3 What is a marker interface?

A Marker interface can be defined as the interface having no data member and member functions. In simpler terms, an empty interface is called the Marker interface. The most common examples of Marker interface in Java are Serializable, Cloneable etc. The marker interface can be declared as follows.

Q4. What is constructor chaining in Java?

In Java, constructor chaining is the process of calling one constructor from another with respect to the current object. Constructor chaining is possible only through legacy where a subclass constructor is responsible for invoking the superclass' constructor first. There could be any number of classes in the constructor chain. Constructor chaining can be achieved in two ways:

1. Within the same class using this()
2. From base class using super()

.

Q.5 **What is final keyword in Java?**

final is a special keyword in Java that is used as a non-access modifier. A final variable can be used in different contexts such as:

- *final variable*

When the final keyword is used with a variable then its value can't be changed once assigned. In case the no value has been assigned to the final variable then using only the class constructor a value can be assigned to it.

- *final method*

When a method is declared final then it can't be overridden by the inheriting class.

- *final class*

When a class is declared as final in Java, it can't be extended by any subclass class but it can extend other class.

Q.6 **What is an infinite loop in Java? Explain with an example.**

An infinite loop is an instruction sequence in Java that loops endlessly when a functional exit isn't met. This type of loop can be the result of a programming error or may also be a deliberate action based on the application behavior. An infinite loop will terminate automatically once the application exits.

For example:

```
1   public class InfiniteForLoopDemo
2   {
3   public static void main(String[] arg) {
4   for(;;)
5   System.out.println("Welcome to Edureka!");
6   // To terminate this program press ctrl + c in the console.
7   }
8   }
```

Q.7 Why Java is not 100% Object-oriented?

ANS : Java is not 100% Object-oriented because it makes use of eight primitive data types such as boolean, byte, char, int, float, double, long, short which are not objects

Q8. What is a Map in Java?

In Java, Map is an interface of Util package which maps unique keys to values. The Map interface is not a subset of the main Collection interface and thus it behaves little different from the other collection types. Below are a few of the characteristics of Map interface:

1. Map doesn't contain duplicate keys.
2. Each key can map at max one value.

Q9 What is a classloader in Java?

The **Java ClassLoader** is a subset of JVM (Java Virtual Machine) that is responsible for loading the class files. Whenever a Java program is executed it is first loaded by the classloader. Java provides three built-in classloaders:

1. Bootstrap ClassLoader
2. Extension ClassLoader
3. System/Application ClassLoader

Q10 . What is collection class in Java? List down its methods and interfaces.

In Java, the collection is a framework that acts as an architecture for storing and manipulating a group of objects. Using Collections you can perform various tasks like searching, sorting, insertion, manipulation, deletion, etc. Java collection framework includes the following:
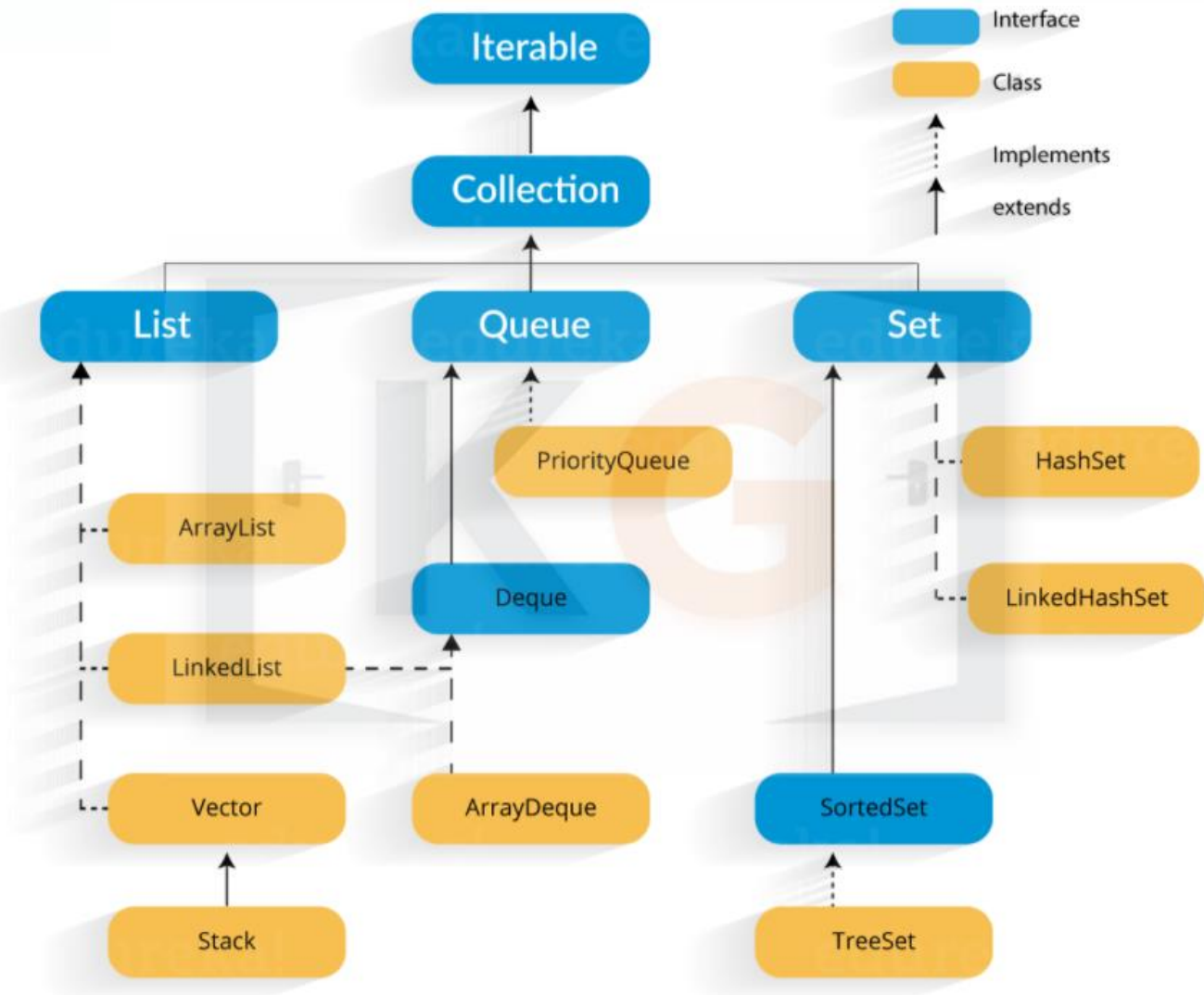
- Interfaces
- Classes
- Methods

The below image shows the complete hierarchy of the Java Collection.

Iterable

Collection

Interface		
Class		
Implements		
extends		

List

Queue

Set

PriorityQueue

ArrayList

HashSet

Deque

LinkedList

LinkedHashSet

Vector

ArrayDeque

SortedSet

Stack

TreeSet