Core Java Week!

Questions and important topics

Static vs instance

A static variable is a variable that belongs to the class; an instance variable is a variable that belongs to the object (instance of the class). When creating class instances, each object gets its own copies of the instance variables, while static variables are shared among the objects.

```
public class Test {
    int x;
    static int y = 100;
    public Test(int x) { this.x = x; }
    public static void main(String[] args) {
        Test t1 = new Test(8);
        Test t2 = new Test(3);
        // True, y is static and shared between t1 and t2
        System.out.println(t1.y == t2.y);
        // False, t1 and t2 get their own copies of non-static x
        System.out.println(t1.x == t2.x);
    }
}
```

When and where can i use the static keyword

-Use static keyword when using methods or variables that are independent of instances/objects.

What does the final keyword mean when applied to a variable vs method vs class

A final variable is a variable that can only be initialized once; variables of a primitive type cannot be reassigned a new value, and a variable assigned to an object cannot be reassigned to another object (although the object's contents can still be modified).

```
final int x = 5;
x = 2; // Does not compile
final ArrayList<Integer> list = new ArrayList<Integer>();
list.add(9); // This is fine
list = new ArrayList<Integer>(); // This is not fine
```

A final method is a method that cannot be overridden.

```
public final int method() { … }

@Override
public int method() {} // Does not compile
```

A final class is a class that cannot be extended (cannot have any subclasses).

```
public final class Animal { … }
public class Cat extends Animal { … } // Does not compile
```

Explain lambdas

-Enables functional programming in java, and support for parallel processing.
-Lambda expressions are functions that exist in isolation; not tied to any object.
-Lambda expressions can be treated as “values”.

What are the benefits of Java

-Write Once, Run Anywhere
-OOP, Popular language, Rich library of APIs, High Level Language (auto memory allocation, auto garbage collection)

What is OOP?

-Object Oriented Programing (APIE -Abstraction, Polymorphism, Inheritance, Encapsulation)

How can i spin up a thread?

-There are two ways to create an instance of a thread.
1. Implement Runnable interface and pass the class instance into Thread constructor
2. Extend Thread interface and override the run() method.

-Then use [thread_name].start()

Can a class be private

Yes, but only nested classes can be declared private (a nested class is a class within a class). Declaring a private class outside a class is pointless as nothing can use/access it.

What are some annotations associated with JUnit

-@Test, @Before, @After

What is a JUnit test case

-A unit test case written with JUnit to make sure that written code meets its design and works as intended. Tests individual methods.

What’s the difference between JUnit and assert methods

-JUnit Assert is used to test expected outputs
-Java Assert statements are used to declare expected boolean conditions.

What are some benefits of Maven

-Get package dependencies easily
-Forces a standard directory structure

Why use getter and setter methods

Getter and setter methods are necessary when the members of a class are set as private and that nothing other than the object can look at or modify them. They control how modifications are made and how certain properties are accessed.

What is a switch statement and what values can work with it

-A switch statement is a multi-way branch statement, that executes code according to an inputted case value. Switch statements work with byte, short, char, and int data types. Also works with string and enums

Is Java pass by value or pass by reference? What does this mean?
-- pass by value --
https://www.javaworld.com/article/2077424/learn-java/learn-java-does-java-pass-by-reference-or-pass-by-value.html

What is reflection and what are its benefits?

Describe the Collection API -
https://www.ntu.edu.sg/home/ehchua/programming/java/J5c_Collection.html

How can we use Streams in Java? https://www.sitepoint.com/java-8-streams-filter-map-reduce/

Important classes in the Java API
String
Object
Thread
Serializable: An interface that marks an object as being serializable.
Runnable: An interface that an object must implement in order to be run on an instance of the Thread class.
Predicate

Important keywords - https://www.geeksforgeeks.org/access-and-non-access-modifiers-in-java/
Transient
Synchronized
Final
Static
Default
Private
Public
Protected

Static

Terms & concepts
Primitive data type
Wrapper class: A type of class that is used to wrap one of the primitive data types, this is needed when we need to pass these types by reference or into methods or collections that only accept objects.
Casting: The process of converting between data types. Comes in two varieties, implicit such as putting an int type into a double are handled automatically, explicit is when the result might cause errors such as a double into an int, in the latter case an explicit declaration in need for the code to compile.

Autoboxing: Process by which the compiler automatically converts primitives to their wrapper classes and vice-versa

Shadowing
Oop principles
Polymorphism
Thread: A singular “thread of execution”.
Multithreading: A program that uses multiple threads.
Design pattern
Lambda
Interface
String pool
Heap
List
Set
Map
Queue
Vector
Stack
Exception
Stack trace
Unchecked exception
Checked exception
Try
Catch
Finally
Propagate
Throw
Throws

try-with-resources
Abstract class: A class that cannot be instantiated; note that abstract classes can have instance variables and concrete implementations. Abstract classes must be extended and the subclass must be concrete in order to be instantiated.
Functional interface: An interface with one function prototype.
Marker interface: An interface with no function prototypes.
Serialization