Static Keyword

Assignment

Question 1- Why do we need need static keyword in Java Explain with an example

Ans-

The ‘static’ keyword in Java is used to declare variables and methods that belong to a class rather than an instance of the class. This allows them to be accessed directly through the class without having to create an instance. For example, ‘Math.PI’ is a ‘static’ variable in the ‘Math’ class that can be accessed directly without creating an instance of the ‘Math’ class.

Question 2- What is class loading and how does the Java program actually executes ?

Ans-

Class loading is the process of loading a Java class into memory when it is referenced in a program. The Java Virtual Machine (JVM) is responsible for the loading, linking, and initialization of classes during program execution.

When a Java program is executed, the JVM first looks for the main class specified in the program’s entry point. The JVM then loads the class into memory using a three-step process:

1. Loading: The JVM locates the bytecode for the class, which is typically stored in a file on the file system or in a JAR file. The JVM reads the bytecode and creates an internal representation of the class.
2. Linking: The JVM performs a series of checks and modifications to the class to prepare it for execution. This includes verifying that the class is well-formed, resolving references to other classes, and allocating memory for static variables.
3. Initializing: The JVM initializes the class by executing the static initialization blocks and initializing static variables.

Once a class has been loaded and initialized, it can be used by the program. If a class is referenced but has not yet been loaded, the JVM will load it on demand. If the JVM is unable to load a class, it will throw a ClassNotFoundException.

Overall, class loading is a key aspect of Java’s dynamic nature, allowing programs to load classes and execute code on demand.

Question 3- Can we mark a local variable as static.

Ans-

Yes, we can mark a local variable as static in java.

However, doing so would result in a compilation error because the static keyword is used to declare class-level variables and methods, not local variables. Local variables are limited in scope to the block of code in which they are declared and cannot be accessed outside of that block.

Therefore, only class-level variables and methods can be marked as static in Java.

Question 4- Why is the static block executed before the main method in java?

Ans-

The static block is executed before the main method in Java because it is part of the class initialization process, which occurs before the class is instantiated or any of its methods are called. The static block initializes the static variable and performs other necessary setup tasks before the main method, which is the entry point of the program,is called.

Question 5- Why is a static method also called a class method?

Ans-

A static method is also called a class method in Java because it belongs to the class and not to an instance of the class. This means that it can be called directly on the class itself, without needing to create an object of the class. Since it is associated with the class as a whole, rather than with any particular instance of the class, it is commonly referred to as a class method.

Question 6- What is the use of static blocks in java ?

Ans-

Static blocks in Java are used to initialize static variables or perform any static initialization tasks that need to be executed before the class is used. These blocks are executed only once when the class is loaded into memory, and they run before the main method or any other static methods or variables in the class. This allows developers to ensure that any necessary initialization tasks are completed before the class. This allows developers to ensure that any necessary initialization tasks are completed before the class is used, ensuring that the class behaves correctly throughout its lifetime.

Question 7- Difference between static and Instance variables

Ans-

A static variable is a variable that is associated with the class, not with any specific instance of the class. A static variable is shared across all instances of the class and can be accessed directly through the class name without needing to create an instance of the class.

On the other hand, an instance variable is a variable that is associated with each instance of the class. Each instance of the class has its own copy of the instance variable. Instance variables are accessed through an instance of the class.

In summary, the main difference between static and instance variables in Java are:

1. Scope: Static variables are associated with the class, while instance variables are associated with each instance of the class.
2. Access: Static variables can be accessed directly through the class name, while instance variables are accessed through an instance of the class.
3. Sharing: Static variables are shared across all instances of the class, while each instance of the class has its own copy of an instance variable.

Question 8- Difference between static and non static members

Ans-

In short and summarized format, Static members belong to the class itself and are shared among all instances of the class, while non-static members belong to individual instances of the class and have separate values for each instance.