List of Concepts Involved:

- static keyword
- Class loading and How Java program actually executes
- Different Members in the java program
- Static variables
- Static Methods
- Static block
- Difference with respect to static and non static members of a class

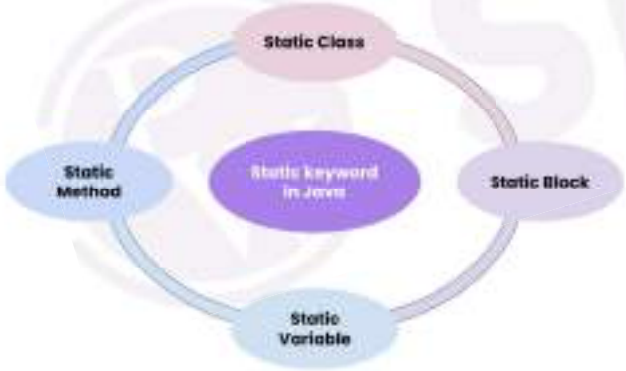
‘static’ keyword

The static keyword is mainly used for memory management in Java. A static keyword can be applied to variables, blocks, methods, and classes. The static keyword is a property of a class rather than an instance of the class. The static keyword is used for a constant variable or a method that is the same for every instance of a class.

Where is the "static" keyword applicable?

The static keyword is a non-access modifier in Java and is applicable for the following:

1. Variables
2. Methods
3. Blocks
4. Class



Static Variables

If we declare any variable as static, then it is called a static variable. When a variable is declared as static, then a single copy of that variable is created and shared among all of the objects at the class level. Static variables are global variables. All instances of the class share the same static variable.
We can create static variables at the class level only.

Why static?

It makes our program more efficient, as every object doesn't allocate separate memory to a static variable.

Static Method

A static method is a method that belongs to a class rather than an instance of a class. This means you can call a static method without creating an object of the class. Static methods are sometimes called class methods.

There are a few other reasons why you might want to use static methods:

- You can access static methods from outside of the class in which they are defined. This is not possible with non-static methods.
- Subclasses can override static methods, but non-static methods cannot.
- Static methods are executed when an instance of the class is created, whereas non-static methods are not.
- Static methods can be used to create utility classes that contain general-purpose methods.

Static Blocks

It is used to initialize static data members. It is used to initialize before the main method at the time of class loading. It gets executed only once when the class gets loaded. It is not necessary to execute it again when creating different objects after the first time.

Static Class

In Java, a "static class" is a class that can be instantiated without having to create an instance of the containing class. A static class is defined as a member of another class and can only access static members of the containing class.

How Java Program Actually executes:

Class Loading

In Java, classloading is the process of loading class files into the JVM (Java Virtual Machine) at runtime. It is responsible for loading classes from various sources, such as the file system, network, and databases, and making them available to the JVM for execution.

The class loading process in Java is divided into three phases: loading, linking, and initialization.

1. Loading: In the loading phase, the classloader locates the class file using the fully qualified class name, reads the class file, and converts it into a Class object. The Class object contains the metadata of the class, such as the fields, methods, and constructors.

2. Linking: In the linking phase, the JVM performs several operations on the Class object, such as verifying the class file's integrity, resolving symbolic references, and allocating memory for the class variables.

3. Initialization: In the initialization phase, the JVM initializes the class variables with their default values, and runs the class's static initialization block (if any).