Java static keyword

The **static keyword** in Java is used for memory management mainly. We can apply static keyword with variables, methods, blocks and nested

classes. The static keyword belongs to the class than an instance of the class.

1. **Java static variable**

If you declare any variable as static, it is known as a static variable.

- The static variable can be used to refer to the common property of all objects (which is not unique for each object), for example, the company name of employees, college name of students, etc.
- The static variable gets memory only once in the class area at the time of class loading.

2. **Java static method**

If you apply static keyword with any method, it is known as static method.

a. A static method belongs to the class rather than the object of a class.

b. A static method can be invoked without the need for creating an instance of a class.

c. A static method can access static data member and can change the value of it.

Q) Why is the Java main method static?

Ans) It is because the object is not required to call a static method. If it were a non-static method, JVM creates an object first then call main() method that will lead the problem of extra memory allocation.

3. **Java static block**
   a. Is used to initialize the static data member.

   b. It is executed before the main method at the time of class loading.

this keyword in Java

There can be a lot of usage of **Java this keyword**. In Java, this is a **reference variable** that refers to the current object.

Super Keyword in Java

The **super** keyword in Java is a reference variable which is used to refer immediate parent class object.

Final Keyword in Java

The **final keyword** in java is used to restrict the user. The java final keyword can be used in many context. Final can be:

1. variable
2. method
3. class

1. Java final variable

If you make any variable as final, you cannot change the value of final variable(It will be constant).

2. Java final method

If you make any method as final, you cannot override it.

3. Java final class

If you make any class as final, you cannot extend it.