static and final keyword in java

static and final keyword in java

1. The members that are declared with the static keyword inside a class are called static members in java.
2. A static variable is also known as class variable in Java.
3. In Java, static is a keyword that is used for memory management mainly.
4. Static keyword in Java can be applied with variables, methods, inner classes, and blocks.
5. We cannot declare a class with static keyword but the inner class can be declared as static.
6. Static members get memory once when the class is loaded into the memory. But instance members get the memory after the object creation of the class.
7. A local variable cannot be declared as static.

How to change value of Static Variable?

We can change the value of the static variable in Java by using

1. a constructor and
2. static block
3. but not inside a static method.

Static Method in Java

When a method is declared with the keyword ‘static’, it is called static method in Java.

A static method is also known as class method because, like a static variable, it is also tied to the class, not to an object of class.

As static method belongs to the class itself rather than an individual instance of a class, so we can call and execute it directly in the class, without creating an object of class.

Features of Static Method

1. A static method in a class can directly access other static members of the class.
2. It can be called directly within the same class and outside the class using the class name.
3. It cannot access instance (i.e. non-static) members of a class.
4. We cannot declare a static method and instance method with the same signature in the same class.
5. When we create a static method in the class, only one copy of the method is created in the memory and shared by all objects of the class. Whether you create 10 objects or 1 object, it does not matter.

Features of Static Method (Cont..)

6. The static method is always bound with compile time.

7. “**this**“ and “super” keywords are not allowed inside the static method or static area. We cannot use “this” keyword in the body of static method because static methods are associated with a class, not an instance.

8. Since we cannot override the static methods, we cannot use the super keyword in its body.

static block in Java.

1. When a block is declared with the **static** keyword, it is called static block in Java.
2. A static block is also known as static initialization block or static initializer block in Java.
3. It gets executed only once by JVM when the class is loaded into the memory by Java **ClassLoader**.

Example:

```
static  {
        Java code.
        }
```

Keypoints of static block:

1. Static Initialization Block is Executed before Main method.
2. **JVM** loads the corresponding dot class file (byte code) into memory.
3. During the dot class file loading into memory, static block is executed. After loading the dot class file, **JVM** calls the main method to start execution.
4. Therefore, static block is executed before the main method.
5. Dot class file is loaded into the memory only one time. So, only one time static block will be executed.
6. Instance block's execution depends upon the object creation.
7. Static block cannot access instance (non-static) variables and methods.

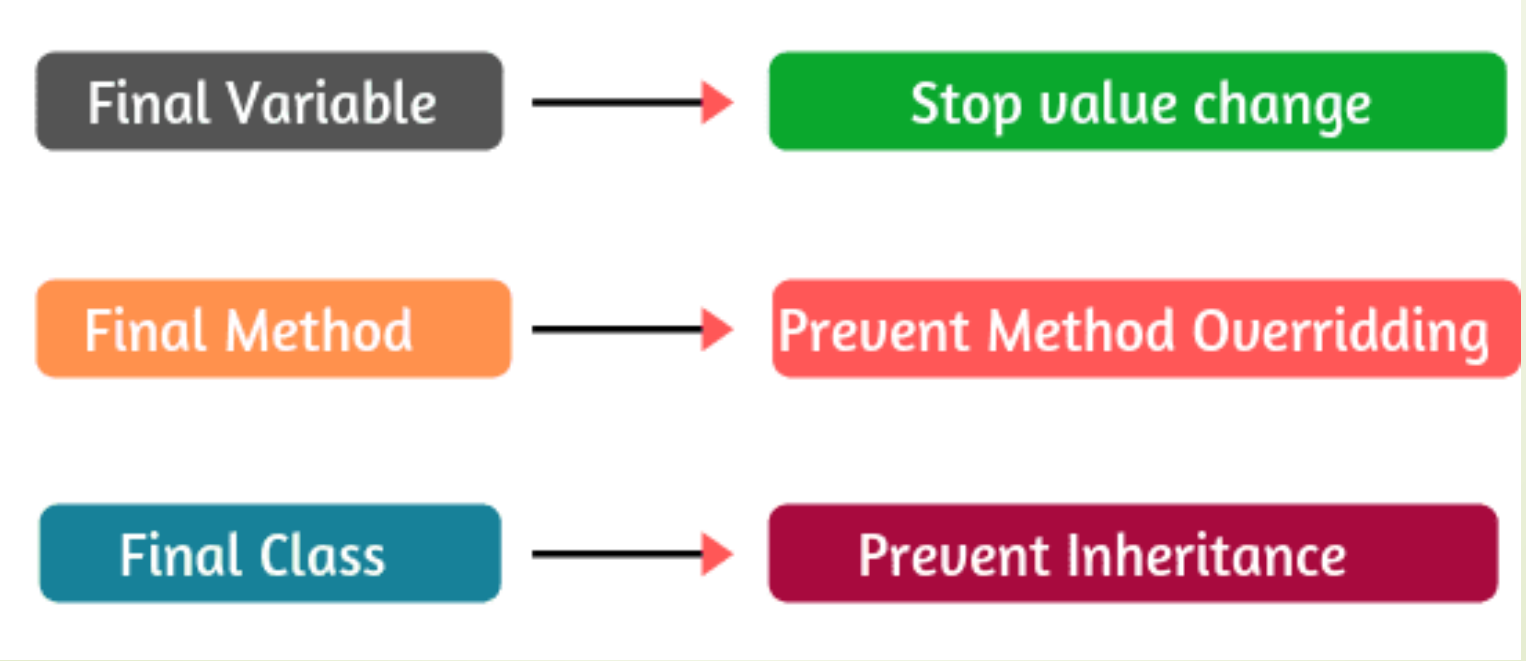
Use of Static Block in Java

1. The purpose of using a static initialization block is to write that logic inside static block that is executed during the class loading.
2. It is mostly used for changing default value of static variables.
3. It is used to initialize static variables of the class.

Final Keyword in Java

Java Final keyword has three different uses:

1. To declare a constant.
2. To prevent inheritance.
3. To prevent method from being overridden.



Final Variable in Java

1. A variable declared with a final keyword is known as a final variable in Java.

2. Final variable means a constant (value cannot be modified).

3. When the final variable once initialized, the value of the final variable can never be changed.

   For example:

   final float pi = 3.14f;

4. A final keyword can be applied to **local** variables, **instance** variables, and **static** variables.

Final Variable in Java

5. A final variable that is created inside the constructor, method, or block is known as a local final variable. It must be initialized once where it is created.
6. A variable that is declared as final and not initialized at a time of declaration is known as a blank final variable in Java.
7. A blank final variable must be initialized in the constructor of the class otherwise, we will get a compilation error.
8. Once the final variable is initialized in the constructor, it cannot be assigned a new value.
9. When a blank final variable is declared as static and not initialized at a time of declaration, it can only be initialized in the static block.
10. Static final variable cannot be initialized in the constructor of the class.

Final Reference Variable

1. A final variable that is declared as a reference to an object is known as reference final variable.
2. There is nothing like a final object in Java.
3. If a final reference variable refers to an object then it does not mean that the object is final.
4. It simply means that the reference variable cannot refer to another object.

Final Method in Java

A method that is declared with the final keyword is known as final method in Java.

A final method cannot be overridden in Java.

Method declared as final, a subclass can call the final method of superclass without any issues but it cannot override it.

Final Class in Java

1. A class that is declared with a final keyword is known as final class in Java.
2. Java classes declared as a final cannot be extended (inherited).
3. If you do not want to be a subclass, declare it final.

   For example:

   String class.

There are two ways to make a class as final.

1. The first way to make a class final is to use the final keyword in the class declaration.
2. The second way is to declare all of its constructors as private.

Summary of Final Keyword

- ✓ A constructor cannot be final.
- ✓ A block cannot be final.
- ✓ A local final variable must be initialized at the time of declaration.
- ✓ We cannot change the value of a final variable after initialization.
- ✓ We cannot override a final method.
- ✓ A final class cannot be extended(inherited).
- ✓ We can create the object for a final class but cannot extend it.
- ✓ If the method parameters are declared as final, the value of these parameters cannot be changed.