Here is a detailed set of notes based on the document about **Static and Final Keywords in Java**:

Static Keyword in Java

1. **Definition**:
    - Members declared with the `static` keyword inside a class are called static members.
    - `Static` is primarily used for memory management.
    - Static elements belong to the class rather than any object of the class.
2. **Key Points**:
    - A **static variable** is also called a **class variable**.
    - Static can be applied to variables, methods, inner classes, and blocks.
    - A **static class** cannot be declared, but an **inner class** can be static.
    - Static members are allocated memory only once, when the class is loaded, whereas instance members are allocated when the object is created.
    - Local variables cannot be declared as `static`.

Changing the Value of Static Variables

- Methods to change the value:
    1. Constructor
    2. Static block
- Cannot change static variable value inside a static method.

Static Methods

1. **Definition**:
    - Declared using the `static` keyword.
    - Known as class methods since they are tied to the class and not to any specific object.
2. **Features**:
    - Access other static members directly.
    - Can be called directly using the class name or within the same class.
    - Cannot access instance (non-static) members.
    - Cannot declare static and instance methods with the same signature in the same class.
    - Only one copy of a static method is shared across all instances of the class.
    - Bound at compile time.
    - **`this` and `super`** keywords are not allowed in static methods.
    - Cannot be overridden.

Static Block

1. **Definition**:

- A block declared with the `static` keyword, also called a static initialization block.
- Executes only once when the class is loaded by the JVM.

2. **Key Points**:
   - Executed before the main method.
   - The JVM loads the class file into memory and executes the static block during this process.
   - Static blocks cannot access instance variables or methods.
3. **Uses**:
   - To initialize static variables.
   - To set default values for static variables.

Final Keyword in Java

- **Uses**:
  1. To declare constants.
  2. To prevent inheritance.
  3. To prevent method overriding.

Final Variable

1. **Definition**:
   - Declared with the `final` keyword; its value cannot be modified after initialization.
2. **Key Points**:
   - Applicable to local, instance, and static variables.
   - A **blank final variable** (uninitialized) must be initialized in the constructor.
   - A static blank final variable can only be initialized in a static block.
3. **Final Reference Variable**:
   - A reference declared as `final` cannot point to another object but the object it references can still be modified.

Final Method

- A method declared as `final` cannot be overridden by subclasses.
- A subclass can call a `final` method from the superclass.

Final Class

1. **Definition**:
   - Declared with the `final` keyword to prevent inheritance.
2. **Key Points**:
   - Example: `String` class in Java.

    - All constructors of a final class can be private to enforce immutability.

Summary of Final Keyword

- A constructor cannot be declared `final`.
- A block cannot be `final`.
- A local `final` variable must be initialized at the time of declaration.
- Final variables cannot be reassigned.
- Final methods cannot be overridden.
- Final classes cannot be extended but objects can still be created.

These notes provide a structured and detailed overview of the concepts related to the `static` and `final` keywords in Java. Let me know if you'd like any section expanded or explained further!