**🔷 STATIC Keyword in Java (Detailed Notes)**

The static keyword in Java is a **non-access modifier** used for:

* **Variables**
* **Methods**
* **Blocks**
* **Nested classes**

It is used to indicate that a **member belongs to the class rather than instances** of the class.

**✅ 1. Static Variables (Class Variables)**

**➤ Definition:**

A **static variable** is shared among **all instances** of a class. It's initialized only **once** when the class is loaded into memory.

**➤ Syntax:**

class Example {

static int count = 0; // Static variable

}

**➤ Characteristics:**

* Memory allocated only once at **class loading time**.
* Shared across all objects of the class.
* Accessed using **class name** or **object reference**.

**➤ Example:**

class **Student** {

static String college = "ABC College"; // shared by all objects

int rollNo;

**Student(int rollNo)** {

this.rollNo = rollNo;

}

void show() {

System.out.println(rollNo + " " + college);

}

}

public class TestStaticVariable {

public static void main(String[] args) {

Student s1 = new Student(101);

Student s2 = new Student(102);

s1.show();

s2.show();

}

}

**✅ 2. Static Methods**

**➤ Definition:**

Static methods belong to the class, not to any object of the class.

**➤ Syntax:**

static return\_type methodName() {

// method body

}

**➤ Characteristics:**

* Can be called without creating an object.
* Can **only access static data** directly.
* Cannot use this or super keywords.
* Mostly used for utility or helper methods.

**➤ Example:**

class **MathUtils** {

**static int square(int x)** {

return x \* x;

}

}

public class TestStaticMethod {

public static void main(String[] args) {

System.out.println(MathUtils.square(5)); // Direct class access

}}

**✅ 3. Static Blocks**

**➤ Definition:**

A static block is a **block of code inside a class** that runs **once** when the class is loaded.

**➤ Syntax:**

static {

// initialization code

}

**➤ Use:**

* Used to initialize **static variables**.
* Used for **complex static initialization logic**.
* Executes **before the main method** and any constructor.

**➤ Example:**

class **StaticBlockExample** {

static int a;

**static** {

System.out.println("Static block called");

a = 10;

}

public static void main(String[] args) {

System.out.println("Main method called");

System.out.println("Value of a: " + a);

}

}

**➤ Output:**

Static block called

Main method called

Value of a: 10

**✅ Summary of Usage**

| **Use Case** | **Keyword** | **Purpose** |
| --- | --- | --- |
| **Shared variable** | static int count; | Common for all objects |
| **Utility method** | static void show() | No need for object creation |
| **One-time config** | static { ... } | Initialization during class load |