Great question, Gyanenjay! 💡

**❓ Is Thread.currentThread().isInterrupted() true by default?**

👉 **No**, it returns **false by default**.

**🧠 Why?**

Because when a thread starts running, its **"interrupted flag" is false by default** — it means the thread has **not been interrupted** yet.

So unless you explicitly call:

thread.interrupt();

...the isInterrupted() method will always return **false**.

**✅ Example:**

public class CheckInterrupt {

public static void main(String[] args) {

Thread t = new Thread(() -> {

System.out.println(Thread.currentThread().isInterrupted()); // false

});

t.start();

}

}

**🔁 If you want it to return true:**

You need to call t.interrupt() before checking:

Thread t = new Thread(() -> {

Thread.currentThread().interrupt(); // Set the interrupt flag

System.out.println(Thread.currentThread().isInterrupted()); // true

});

t.start();

Let me know if you want to test interrupt with sleep() or join() too — those are classic interview trick zones!