**Mastering Arrays in Java – Topic 2: Traversing with Loops**

When you have **multiple values** stored in an array, the real magic happens when you can **visit each element** efficiently.  
This process is called **Array Traversal** — and in Java, **loops** are our superpower for it. 💡

**🔹 Why Loops Matter**

Without loops, printing a 100-element array means writing 100 lines of code 😵.  
With loops, it’s just a few lines — and works for **any size array**.

**🔹 Two Ways to Traverse**

1️⃣ **Classic for loop** – Gives you the index & value.  
2️⃣ **Enhanced for-each loop** – Cleaner & simpler for reading values.

💻 **Example:**

// Classic for loop

for (int i = 0; i < arr.length; i++) {

System.out.println("Index " + i + ": " + arr[i]);

}

// For-each loop

for (int value : arr) {

System.out.println(value);

}

**🧠 Pro Tips for Students**

* Always use .length in loops — never hardcode size.
* Use for loop when you **need the index**.
* Use for-each loop when you just **need values**.
* Be careful with loop conditions to avoid ArrayIndexOutOfBoundsException.

💬 **Question for You:**  
Which loop do you use more often in Java — for or for-each — and why?  
Let’s discuss in the comments 👇

#Java #Arrays #Programming #CodingTips #JavaDeveloper #LearnJava #100DaysOfCode #DataStructures