**Interview Questions**

**JAVA-111: Session 4**

**Answering interview questions is crucial in your journey of applied learning. Review them to ensure your understanding of important topics covered in the previous session and to prepare yourself for upcoming challenges. Remember that it's important to answer these questions on your own before viewing the solution. The solutions are hyperlinked to community posts on our platform.**

**Note: The questions below have been sourced from previous interviews**

**Questions**

1. [**What are looping statements and how do they work in Java?**](https://crio.do/learn/crio-community/topic/what-are-looping-statements-and-how-do-they-work-in-java/252933)
2. [**What are branching statements and how do they work in Java?**](https://crio.do/learn/crio-community/topic/what-are-branching-statements-and-how-do-they-work-in-java/252934)

**Additional Readings**

* [**for loop in java**](https://www.programiz.com/java-programming/for-loop)
* [**Java while and do-while loop**](https://www.programiz.com/java-programming/do-while-loop#syntax-while)
* [**Jump Statements in java**](https://www.scholarhat.com/tutorial/java/java-jump-statements-break-continue-return)
* [**Difference between break and return statements**](https://www.linkedin.com/pulse/difference-between-return-break-statements-muhammad-awais#:~:text=%22Return%22%20affects%20the%20entire%20function,efficient%20and%20error%2Dfree%20code.)

**✅ 1. What are looping statements and how do they work in Java?**

**📌 Definition:**

Looping statements are used to **repeat a block of code** multiple times **as long as a condition is true**. They help reduce code repetition.

**🔄 Types of Loops in Java:**

| **Loop Type** | **Description** |
| --- | --- |
| for loop | Repeats code a fixed number of times. |
| while loop | Repeats code **while a condition is true**. |
| do-while loop | Like while, but it runs **at least once**, even if the condition is false. |

**✅ Example 1 – for loop:**

java

CopyEdit

for (int i = 1; i <= 5; i++) {

System.out.println(i);

}

🔹 Output: 1 2 3 4 5

**✅ Example 2 – while loop:**

java

CopyEdit

int i = 1;

while (i <= 5) {

System.out.println(i);

i++;

}

**✅ Example 3 – do-while loop:**

java

CopyEdit

int i = 1;

do {

System.out.println(i);

i++;

} while (i <= 5);

**✅ 2. What are branching statements and how do they work in Java?**

**📌 Definition:**

Branching statements are used to **change the flow** of execution in a program. They let you **exit a loop**, **skip an iteration**, or **return from a method**.

**🔁 Types of Branching Statements:**

| **Statement** | **Description** |
| --- | --- |
| break | Exits a loop or switch statement. |
| continue | Skips the rest of the loop **for current iteration** and moves to the next. |
| return | Exits from the current method and optionally returns a value. |

**✅ Example – break:**

java

CopyEdit

for (int i = 1; i <= 10; i++) {

if (i == 5) break;

System.out.println(i);

}

🔹 Output: 1 2 3 4 (loop stops at 5)

**✅ Example – continue:**

java

CopyEdit

for (int i = 1; i <= 5; i++) {

if (i == 3) continue;

System.out.println(i);

}

🔹 Output: 1 2 4 5 (skips 3)

**✅ Example – return:**

java

CopyEdit

public static void greet() {

System.out.println("Hello!");

return;

}

**🔁 1. for Loop in Java**

**📌 Used when the number of iterations is known.**

**✅ Syntax:**

java

CopyEdit

for (initialization; condition; update) {

// code block

}

**✅ Example:**

java

CopyEdit

for (int i = 1; i <= 5; i++) {

System.out.println(i);

}

**🧠 Output:**

CopyEdit

1

2

3

4

5

**🔄 2. while and do-while Loops in Java**

**✅ while Loop:**

* **Condition is checked first**
* Loop runs **only if the condition is true**

**Syntax:**

java

CopyEdit

while (condition) {

// code block

}

**Example:**

java

CopyEdit

int i = 1;

while (i <= 5) {

System.out.println(i);

i++;

}

**✅ do-while Loop:**

* Code block runs **at least once**
* Then the condition is checked

**Syntax:**

java

CopyEdit

do {

// code block

} while (condition);

**Example:**

java

CopyEdit

int i = 1;

do {

System.out.println(i);

i++;

} while (i <= 5);

**🔁 3. Jump Statements in Java**

Jump statements control the flow of execution.

**🔸 Types:**

| **Statement** | **Use** |
| --- | --- |
| break | Exits a loop or switch block early |
| continue | Skips the rest of current loop iteration |
| return | Exits from the current method |

**✅ Examples:**

java

CopyEdit

// break example

for (int i = 1; i <= 5; i++) {

if (i == 3) break;

System.out.println(i);

}

// Output: 1 2

// continue example

for (int i = 1; i <= 5; i++) {

if (i == 3) continue;

System.out.println(i);

}

// Output: 1 2 4 5

// return example

public static void greet() {

System.out.println("Hello");

return; // ends the method here

}

**❓ 4. Difference Between break and return Statements**

| **Feature** | **break** | **return** |
| --- | --- | --- |
| Purpose | Exits a loop or switch block | Exits a method completely |
| Used in | Loops (for, while, do-while), switch | Any method |
| Affects | Only loop or switch control flow | Ends method and optionally returns a value |
| Example | break; inside a loop | return 5; inside a method |

**📌 Summary Chart:**

| **Loop Type** | **Entry Check** | **Runs at least once?** | **Best For** |
| --- | --- | --- | --- |
| for | Yes | No | Known iteration count |
| while | Yes | No | Unknown repetition |
| do-while | After first | Yes | Run at least once |