Chapter-5

What is loop? How many kinds of loops are there in Java?

*Loops* are constructs that control repeated executions of a block of statements. The concept

of looping is fundamental to programming. Java provides three types of loop statements:

**while** loops, **do**-**while** loops, and **for** loops.

What is While loop?

*A* **while** *loop executes statements repeatedly while the condition is true.* The part of the loop that contains the statements

to be repeated is called the *loop body.* A one-time execution of a loop body is referred to

as an *iteration (*or *repetition) of the loop.* Each loop contains a *loop-continuation-condition*

The syntax for the **while** loop is:

**while** (loop-continuation-condition) {

// Loop body

Statement(s);

}

N.B. The **while** loop repeatedly executes the statements in the loop body when the

**loop-continuation-condition** evaluates to **true**.

What are the loop Design Strategies?

Consider three steps when

writing a loop.

Step 1: Identify the statements that need to be repeated.

Step 2: Wrap these statements in a loop like this:

**while** (**true**) {

Statements;

}

Step 3: Code the **loop-continuation-condition** and add appropriate statements for

controlling the loop.

**while** (loop-continuation-condition) {

Statements;

Additional statements for controlling the loop;

}

What is sentinel value?

In programming, **sentinel value** is a special **value** that is used to terminate a loop. A loop that uses a sentinel value to control its execution is called a

*Sentinel-controlled loop.*

Describe Do-while-loop

*A* **do-while** *loop is the same as a* **while** *loop except that it executes the loop body*

*first and then checks the loop continuation condition.*

The **do**-**while** loop is a variation of the **while** loop. Its syntax is:

**do** {

// Loop body;

Statement(s);

} **while** (loop-continuation-condition);

What are the differences between Do-while-loop and while-loop?

* In [Java While loop](https://www.tutorialgateway.org/java-while-loop/), condition is tested at the beginning of the loop and if the condition is True then only statements in that loop will be executed. So, While loop executes the code block **only** if the condition is True.
* In [Java Do While loop](https://www.tutorialgateway.org/java-do-while-loop/), condition is tested at the end of the loop so Do While executes the statements in the code block at least once even if the condition Fails.

What is for loop?

*A* **for** *loop has a concise syntax for writing loops.* A **for** loop can be used to simplify the preceding loop as:

**for** (i = initialValue; i < endValue; i++)

// Loop body

...

}

In general, the syntax of a **for** loop is:

**for** (initial-action; loop-continuation-condition;

action-after-each-iteration) {

// Loop body;

Statement(s);

}

N.B. The **for** loop statement starts with the keyword **for**, followed by a pair of parentheses

enclosing the control structure of the loop. This structure consists of **initial-action**,

**loop-continuation-condition**, and **action-after-each-iteration**. The control

structure is followed by the loop body enclosed inside braces. The **initial-action**, **loopcontinuation-**

**condition**, and **action-after-each-iteration** are separated by

semicolons.

Which Loop to Use?

*You can use a* **for** *loop, a* **while** *loop, or a* **do-while** *loop, whichever is convenient.*

The **while** loop and **for** loop are called *pretest loops* because the continuation condition

is checked before the loop body is executed. The **do-while** loop is called a *posttest loop*

because the condition is checked after the loop body is executed. The three forms of loop

statements—**while**, **do-while**, and **for**—are expressively equivalent; that is, you can write

a loop in any of these three forms.

Describe Nested Loops?

*A loop can be nested inside another loop.*

*Nested loops* consist of an outer loop and one or more inner loops. Each time the outer loop is

repeated, the inner loops are reentered, and started anew.

What is break and continue loop?

*The* **break** *and* **continue** *keywords provide additional controls in a loop.* Two keywords, **break** and **continue**, can be used in loop statements to provide additional

controls. Using **break** and **continue** can simplify programming in some cases.

**Note**

The **continue** statement is always inside a loop. In the **while** and **do-while** loops,

the **loop-continuation-condition** is evaluated immediately after the **continue**

statement. In the **for** loop, the **action-after-each-iteration** is performed,

then the **loop-continuation-condition** is evaluated, immediately after the

**continue** statement.

**CHAPTER SUMMARY**

**1.** There are three types of repetition statements: the **while** loop, the **do-while** loop, and

the **for** loop*.*

**2.** The part of the loop that contains the statements to be repeated is called the *loop body.*

**3.** A one-time execution of a loop body is referred to as an *iteration of the loop.*

**4.** An *infinite loop* is a loop statement that executes infinitely.

**5.** In designing loops, you need to consider both the *loop control structure* and the loop

body.

**6.** The **while** loop checks the **loop-continuation-condition** first. If the condition

is **true**, the loop body is executed; if it is **false**, the loop terminates.

**7.** The **do-while** loop is similar to the **while** loop, except that the **do-while** loop executes

the loop body first and then checks the **loop-continuation-condition** to

decide whether to continue or to terminate.

**8.** The **while** loop and the **do-while** loop often are used when the number of repetitions

is not predetermined.

**9.** A *sentinel value* is a special value that signifies the end of the loop.

**10.** The **for** loop generally is used to execute a loop body a fixed number of times.

**11.** The **for** loop control has three parts. The first part is an initial action that often initializes

a control variable. The second part, the **loop-continuation-condition**,

determines whether the loop body is to be executed. The third part is executed after

each iteration and is often used to adjust the control variable. Usually, the loop control

variables are initialized and changed in the control structure.

**12.** The **while** loop and **for** loop are called *pretest loops* because the continuation condition

is checked before the loop body is executed.

**13.** The **do-while** loop is called a *posttest loop* because the condition is checked after the

loop body is executed.

**14.** Two keywords, **break** and **continue**, can be used in a loop.

**15.** The **break** keyword immediately ends the innermost loop, which contains the break.

**16.** The **continue** keyword only ends the current iteration.