Java™

Java Flow Controls Part 2

Content

1. Sequential Flow
2. Decision-Making Statements (Conditional)/Selection
   a. if
   b. if-else
   c. if-else if-else ladder
3. Looping Statements (Iteration)
   a. for loop
   b. while loop
   c. do-while loop
   d. Nested Loop
4. Jump Statements
   a. break
   b. continue
5. Labeling Loops

Types of Control Flow

Types of control flow in programming:

- Sequential
- Conditional
- Iterative

What is Looping/Iteration

Println(“Hello”);

Println(“Hello”);

Println(“Hello”);

or

?

Do-While Loop

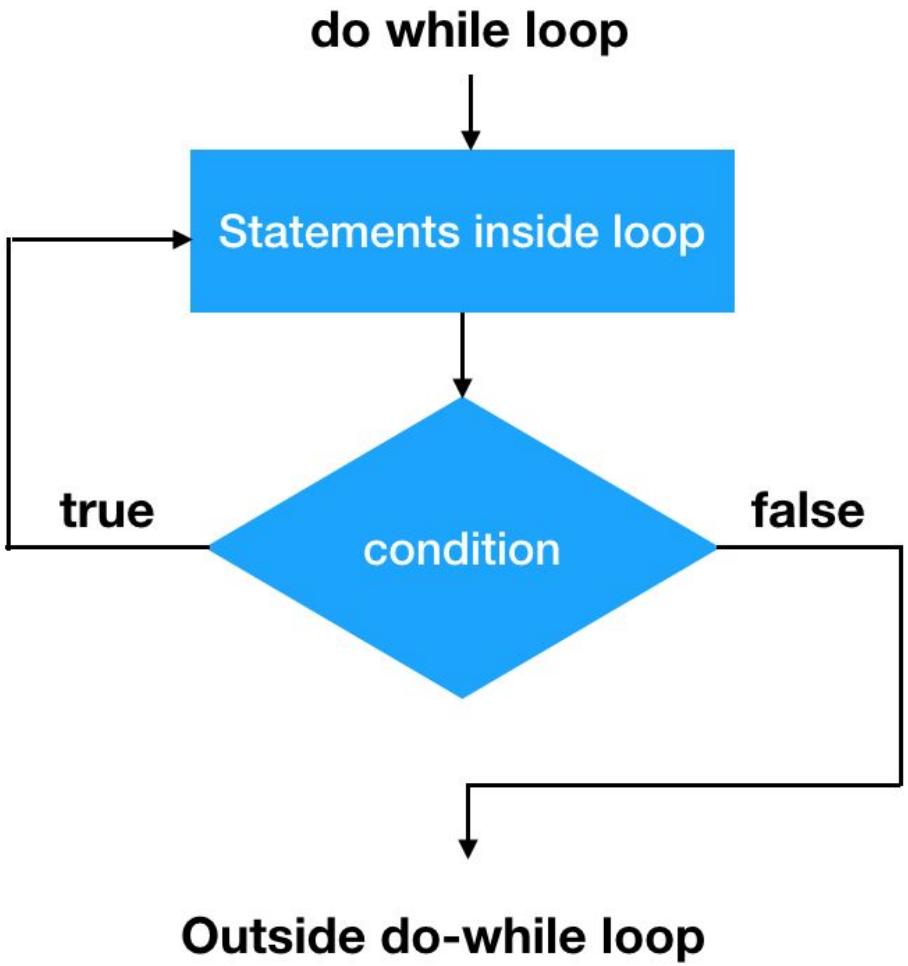
A do-while loop in Java is a control flow statement that allows code to be executed at least once, and then repeated as long as a condition is true.

Syntax:

```
do {
   // code block to be executed
} while (condition);
```

Note: The semicolon after while(condition); is required.

Do-While Loop



Do-while Loop

```
int i = 10;
do {
   System.out.println("Inside loop: ");
   i++;
} while (i < 5);
```

Q1. Print "Welcome to C-DAC Patna" 10 times.

Q2. Print 1 to 5.

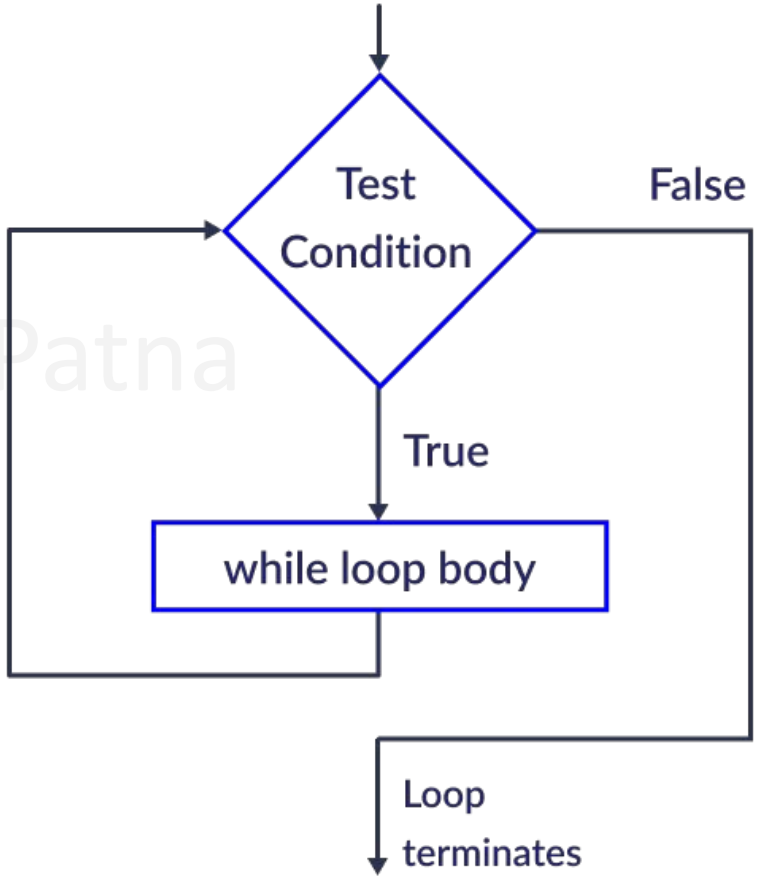
Q3. Countdown from 5 to 1.

while Loop

A while loop in Java is a control flow statement that repeats a block of code as long as a specified condition is true. If the condition is false initially, the loop body won't run even once. Useful when you don't know how many times the loop should run.

Syntax:

```
while (condition) {
   // code to be executed
}
```



while Loop

```
int i = 1;
while (i <= 5) {
   System.out.println(i);
   i++;
}
```

Q1. Print even numbers from 2 to 10.

Q2. Sum of first 5 numbers.

Q3. Print multiplication table of 3

Common Mistakes

- Forgetting to update the variable inside the loop.
- Creating an infinite loop by mistake.
- The condition must become False after some time, otherwise, loop runs forever.
- Using while loop when a for loop is better suited for fixed iterations.

Break

- The break statement is used to exit a loop or switch statement immediately, regardless of the condition. Once break is encountered, control jumps outside the loop or switch block.

- Break used in:
  In loops (for, while, do-while) → to stop the loop early
  In switch statements → to prevent fall-through

```
int i = 1;
while(i <= 10) {
   if(i == 4) {
      break;
   }
   System.out.println(i);
   i++;
}
```

What will be output of above code ?

Continue

- The continue statement is used to skip the current iteration of a loop and move to the next iteration directly. Unlike break, it does not exit the loop—it just skips the rest of the loop body for that iteration.

```
int i = 0;
while(i < 5) {
   i++;
   if(i == 3) {
      continue;
   }
   System.out.println(i);
}
```


Guess Output ?

Q. Print numbers from 1 to 10, but skip multiples of 3 using continue.

Nested Loops

- A nested loop is a loop inside another loop.
  - The outer loop controls the number of rows (or main repetitions).
  - The inner loop controls the number of columns (or detailed actions within each outer loop).

```
int i = 1;
while(i <= 3) {
   int j = 1;
   while(j <= 4) {
      System.out.print("Hello");
      j++;
   }
   System.out.println();
   i++;
}
```

Guess Output?

Infinite Loops

- An infinite loop is a loop that never stops running because the termination condition is never met or is missing altogether.
- Infinite loops can be intentional (like in games, servers) or accidental (programming mistake).

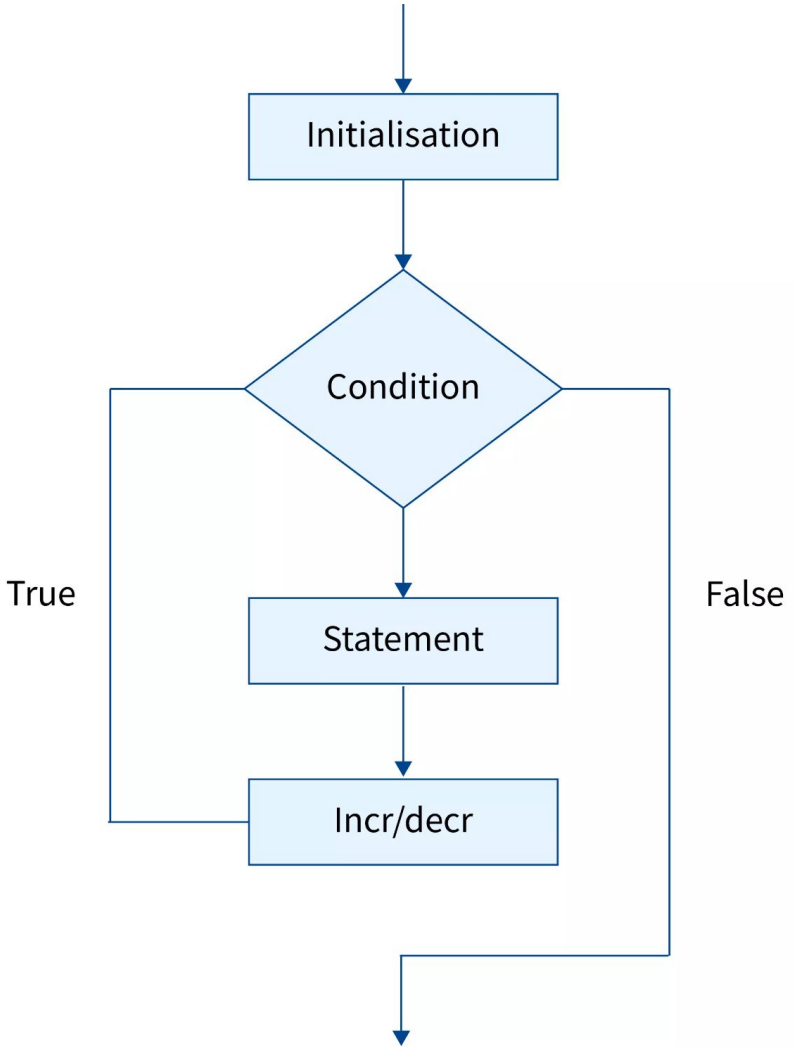
```
while(true) {
   System.out.println("This will run forever");
}
```

for loop

- A for loop in Java is a control structure used to repeat a block of code a fixed number of times. It's best used when the number of iterations is known in advance.

Syntax:

```
for(initialization; condition; update) {
   // Code to be executed
}
```



for loop

```
for(int i = 1; i <= 5; i++) {
   System.out.println("Count: " + i);
}
```

1. Print “Welcome” 10 times
2. Print numbers from 0 to 4
3. Print even numbers between 2 and 10
4. Print odd numbers between 1 and 10
5. Print numbers from 10 to 1 in reverse
6. Print square of numbers from 1 to 5
7. Count from 1 to 20, skipping every 2 numbers
8. Print first 10 multiples of 7

Nested for Loops

```
for(int i = 1; i <= 3; i++) {
   for(int j = 1; j <= 4; j++) {
      System.out.print("* ");
   }
   System.out.println(); // Move to next line
}
```

1. Print the following pattern using for loop:

```
* * *
* * *
* * *
```


2. Print this pattern using numbers:

```
1
12
123
1234
```

Labeling Loops:

- A labeled loop in Java allows you to name a loop so that you can specify which loop to break or continue when using nested loops.

- This is especially useful when you have nested loops and want to control the outer loop from inside the inner loop.

```
labelName:
for( ... ) {
   for( ... ) {
      if(condition) {
         break labelName;  // exits the outer loop
      }
   }
}
```

Assignments

1. Write a program to find the sum of the first 5 natural numbers using a **while** loop.

2. Print the multiplication table of 3 (from 3×1 to 3×10) using a **while** loop.

3. Print numbers in reverse order from 20 down to 1 using a **for** loop.

4. Print natural numbers starting from 1 to 10. The loop should stop when the number reaches 7. Use a **while** loop and **break** statement.

5. Print numbers from 10 down to 1, but skip the number 7 using a **while** loop and **continue**.

6. Print numbers from 1 to 100. Use a **for** loop and **break** it if any number divisible by 11 is encountered.

7. Keep accepting numbers from the user until they enter 0. After that, print the total sum of all entered numbers.

8. Print numbers from 1 to 100. If a number is divisible by both 3 and 5, print "Skip" instead of the number using a for loop and continue.

Optional Assignments

9. Print numbers in a 3×3 grid using nested loops.

```
1 2 3
1 2 3
1 2 3
```

10. Print the following number triangle using nested loops:

```
1
1 2
1 2 3
```


11. Print a left-aligned triangle using *:

```
*
* *
* * *
* * * *
```

12. Print a right-aligned pyramid using *:

```
   *
  * *
 * * *
* * * *
```

Eclipse Installation

- Follow this video Guide and try to Install Eclipse IDE:
  https://www.youtube.com/watch?v=i3uK--LXQU8

THANK YOU!!

C-DAC Patna