06-While Loop

While Loop

A while loop is an entry-controlled control flow structure in programming that repeatedly executes a block of code as long as a specified condition remains true. The loop continues to iterate while the condition is true and terminates once the condition evaluates to false.

Basic Syntax of While Loop:

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

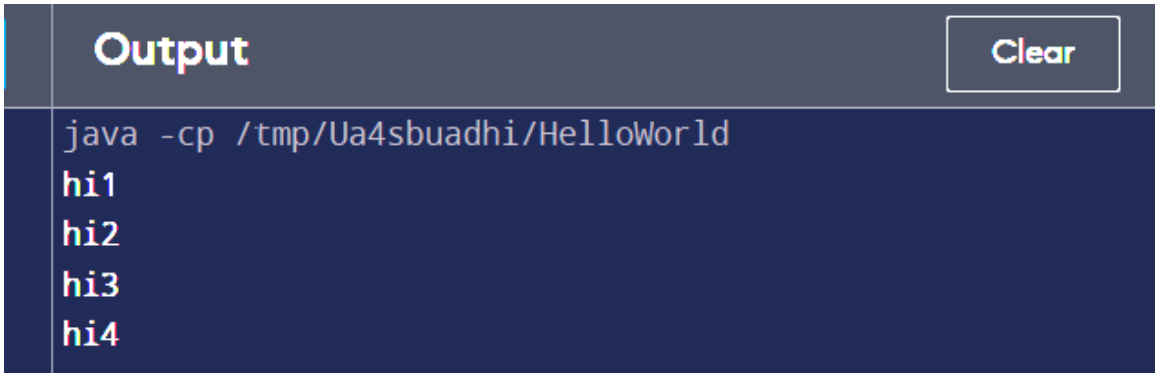
In this structure, the condition is checked before each iteration. If the condition evaluates to true, the loop body (code block) is executed. If the condition evaluates to false, the loop terminates.

Example: Printing "Hi" Multiple Times

To print "Hi" four times using a while loop, you can use a counter variable that increments with each iteration. This counter helps control the number of times the loop executes, preventing infinite loops.

Example Code:

```
int i = 0; // Initialize counter variable
while (i < 4) { // Loop continues while i is less than 4
    System.out.println("Hi");
    i++; // Increment counter
}
```



In this example:

- The counter variable i is initialized to 0.

- The condition i < 4 ensures the loop runs only four times.
- The System.out.println("Hi"); statement prints "Hi" during each iteration.
- The counter i increments by 1 after each iteration, eventually making the condition false and terminating the loop.

Nested While Loops: Printing "Hello" After "Hi"

To print "Hello" three times after each "Hi", you can use nested while loops. A nested loop runs inside another loop, with the inner loop completing all its iterations before the outer loop proceeds to the next iteration.

Example Code with Explanation:

```
int i = 0; // Outer loop counter
while (i < 4) {
    System.out.println("Hi");
    int j = 0; // Inner loop counter
    while (j < 3) {
        System.out.println("Hello " + (j + 1)); // Preceding with the number of times printed
        j++; // Increment inner loop counter
    }
    i++; // Increment outer loop counter
}
```

In this example:

- The outer loop (controlled by i) prints "Hi" four times.
- For each iteration of the outer loop, the inner loop (controlled by j) prints "Hello" three times, preceded by the number indicating the print count.