do-while loop

If the conditional expression controlling a **while** loop is initially false, then the body of the loop will not be executed at all. However, sometimes it is desirable to execute the body of a **while** loop at least once, even if the conditional expression is false to begin with. In other words, there are times when you would like to test the termination expression at the end of the loop rather than at the beginning. Fortunately, Java supplies a loop that does just that: the **do-while**.

Objectives

Upon completion of this topic, we will be able to:

- Understand and use the do-while loop

do…. while Loop

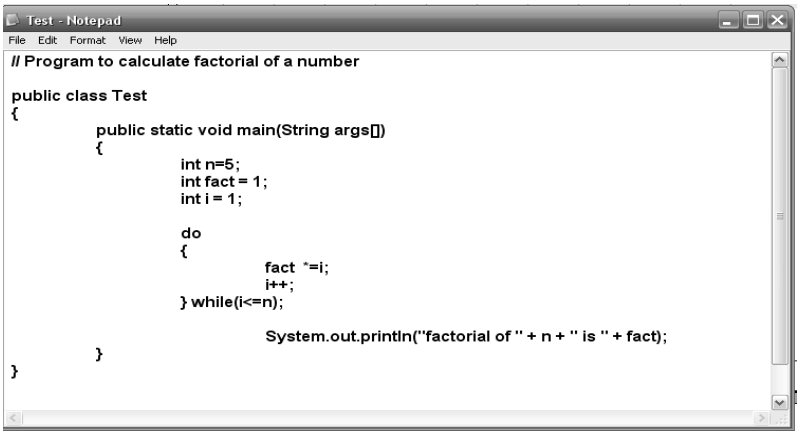
The **do-while** loop always executes its body at least once, because its conditional expression is at the bottom of the loop. Its general form is:

do {

// body of loop

} while (*condition*);

Each iteration of the **do-while** loop first executes the body of the loop and then evaluates the conditional expression. If this expression is true, the loop will repeat. Otherwise, the loop terminates. As with all of Java's loops, *condition* must be a boolean expression.



do…while loop example

The **do-while** loop is especially useful when you process a menu selection, because you will usually want the body of a menu loop to execute at least once. Consider the following program which implements a very simple help system for Java's selection and iteration statements:

```
// Using a do-while to process a menu selection
class Menu {
    public static void main(String args[])
    throws java.io.IOException {
        char choice;
        do {
            System.out.println("Help on:");
            System.out.println(" 1. if");
            System.out.println(" 2. switch");
            System.out.println(" 3. while");
            System.out.println(" 4. do-while");
            System.out.println(" 5. for\\n");
            System.out.println("Choose one:");
            choice = (char) System.in.read();
        } while( choice < '1' || choice > '5');
        System.out.println("\\n");
        switch(choice) {
        case '1':
            System.out.println("The if:\\n");
            System.out.println("if(condition) statement;");
            System.out.println("else statement;");
            break;
        case '2':
            System.out.println("The switch:\\n");
```

```
                System.out.println("switch(expression) {");
                System.out.println(" case constant:");
                System.out.println(" statement sequence");
                System.out.println(" break;");
                System.out.println(" // ...");
                System.out.println("}");
                break;
            case '3':
                System.out.println("The while:\\n");
                System.out.println("while(condition) statement;");
                break;
            case '4':
                System.out.println("The do-while:\\n");
                System.out.println("do {");
                System.out.println(" statement;");
                System.out.println("} while (condition);");
                break;
            case '5':
                System.out.println("The for:\\n");
                System.out.print("for(init; condition; iteration)");
                System.out.println(" statement;");
            }
        }
    }
```

Summary

Here are the key takeaways:

- The **do-while** loop is a post-checked while loop.
- The **do-while** loop is especially useful when you process a menu selection, because you will usually want the body of a menu loop to execute at least once.