**Iteration Statements**

Java’s iteration**for**,**while**, andstatements**do-while**.Thesestatementsarecreate what we commonly call *loops.* As you probably know, a loop repeatedly executes the same set of instructions until a termination condition is met. As you will see, Java has a loop to fit any programming need.

While

The **while** loop is Java’sfundamentalloopingmoststatement. It repeats a statement or block while its controlling expression is true. Here is its general form:

While (*condition*) {

// body of loop

}

The *condition* can be any Boolean expression. The body of the loop will be executed as long as the conditional expression is true. When *condition* becomes false, control passes to the next line of code immediately following the loop. The curly braces are unnecessary if only a single statement is being repeated.

do-while

As you just saw, 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.

Systex:

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.

// Demonstrate the do-while loop. class DoWhile {

public static void main(String args[]) {

int n = 10; do {

System.out.println("tick " + n); n--;

} while(n > 0);

}

}

For

You were introduced to a simple form of the **for** loop in Chapter 2. As you will see, it is a powerful and versatile construct. Here is the general form of the **for** statement:

for(*initialization*; *condition*; *iteration*) {

// body

}

If only one statement is being repeated, there is no need for the curly braces.

The **for** loop operates as follows. When the loop first starts, the *initialization* portion of the loop is executed. Generally, this is an expression that sets the value of the *loopcontrol variable,* which acts as a counter that controls the loop.. Next, *condition* is evaluated. This must be a Boolean expression. It usually tests the loop control variable against a target value. If this expression is true, then the body of the loop is executed. If it is false, the loop terminates. Next, the *iteration* portion of the loop is executed. This is usually an expression that increments or decrements the loop control variable.

// Demonstrate the for loop. class ForTick {

public static void main(String args[]) { int n;

for(n=10; n>0; n--) System.out.println("tick " + n);

}

}