

Supplement: Statement Labels and Breaking with Labels
For Introduction to Java Programming
By Y. Daniel Liang

Every statement in Java can have an optional label as an identifier. Labels are often associated with loops. You can use a **break** statement with a label to break out of the labeled loop, and a **continue** statement with a label to break out of the current iteration of the labeled loop.

The **break** statement given below, for example, breaks out of the outer loop if (**i * j > 50**) and transfers control to the statement immediately following the outer loop.

```
outer:
    for (int i = 1; i < 10; i++) {
        inner:
            for (int j = 1; j < 10; j++) {
                if (i * j > 50)
                    break outer;

                System.out.println(i * j);
            }
    }
```

If you replace **break outer** with **break** in the preceding statement, the **break** statement would break out of the inner loop and continue to stay inside the outer loop.

The following continue statement breaks out of the inner loop if (**i * j > 50**) and starts a new iteration of the outer loop if **i < 10** is true after **i** is incremented by 1.

```
outer:
    for (int i = 1; i < 10; i++) {
        inner:
            for (int j = 1; j < 10; j++) {
                if (i * j > 50)
                    continue outer;

                System.out.println(i * j);
            }
    }
```

If you replace **continue outer** with **continue** in the preceding statement, the **continue** statement would break out of the current iteration of the inner loop and continue the next iteration of the inner loop if **j < 10** is true after **j** is incremented by 1.