

## Indentation

Indentation is possibly the most important component of making your code readable. Look at this code and try to quickly determine what code belongs to which for loop:

```
public class BadIndentation {
public static void main(String[] args) {
    for (int i = 1; i <= omgBadIndentation; i++) {
    for (int j = 1; j <= i; j++) {
        System.out.println();
            gahWhyIsThisMethodCallAllTheWayOverHere();
    }

        int thisSeemsLikeAGoodSpotForThisVariable = 0;
        }
        isTheMethodOverYet();
    }

        whoKnows();
}
}
```

It's hard. Contrast that with this:

```
public class GoodIndentation {
    public static void main(String[] args) {
        for (int i = 1; i <= ahSoNice; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.println();
                thisMethodCallGoesRightHere();
            }
            int perfection = 1;
        }
        isTheMethodOverYet();
    }
    yeahAlmost();
}
}
```

It's much easier to see which code belongs in each class, method, and loop.

The rules for indentation are simple if you pay close attention to your curly braces. In Java, the convention is to open your curly braces at the end of the line that declares your control structure (class, method, loop, etc). For example:

```
public class JavaStyle {
    ...
}
```

(Note: Some people prefer “C-Style” indentation, which puts curly braces on the line after your declaration, as shown below. We allow this, but prefer that you use Java-style indentation for your Java code)

```
public class CStyle
{
    ...
}
```

From here, indentation is simple. Whenever you open a control structure with a curly brace, the following lines will be indented one more tab over (we suggest a tab size of 3-4 spaces-- jGrasp's default is 3). Maintain this indentation until one of two things happen.

1. Another control structure begins, in which case the following lines will be indented another tab over.

```
for (int i = 1; i <= 5; i++) {
    someCodeHere();
    afterTheLoopBeginsIndentSomeMore();
    for (int j = 1; j <= i; j++) {
        System.out.println();
        ...
    }
}
```

2. Your current control structure ends, as indicated by a closing curly brace. The closing curly brace and any subsequent lines should be indented to the same place as the control structure began.

```
for (int i = 1; i <= 5; i++) {
    for (int j = 1; j <= i; j++) {
        insideBothLoops();
    }
    insideJustTheOuterLoop();
}
outsideBothTheLoops();
```

Another note: These rules apply to comments as well. It can be tempting to put them at the beginning of lines (or elsewhere), but only a class comment should go at the beginning of a line.

```
// Class comments go here (but are hopefully longer than this)
public class MyPerfectProgram {

    // Method comments go here (again, hopefully longer...)
    public static void main(String[] args) {
        // Comments are indented the same as code
        confusingCode(); // Acceptable for short lines
    }
}
```

It's that easy. If you're having trouble or just want to be thorough, you can use the [Indenter Tool](#) before turning in your assignments, but we suggest you at least try it yourself first to get a feel for indenting.

Happy indenting!