COSC2531 Programming Fundamentals

Assignment 1 - Code formatting rules

Overview

Just like any other document, they way you lay out your code affects its readability. The reader of your code is going to be someone who knows Java programming (either future you, or another programmer) and they'll be expecting consitency in layout, as well as visual markers to aid them understanding what you have created.

There are a number of rules that you should follow for your assignment, as well as suggestions on how to layout your code to improve readability.

We will assess your code against these rules when marking your assignment, so take care. We hope you find them helpful in making your code easier to read and understand.

Consistency

Java requires the use of braces (`{ }`) to show a hierarchical relationship between control structures and the code they contain.

There are two common ways to lay them out…

```
if (x < 10) {
    System.out.println("x is less than 10");
} else {
    System.out.println("x is not less than 10");
}
```

or

```
if (x < 10)
{
    System.out.println("x is less than 10");
}
else
{
    System.out.println("x is not less than 10");
}
```

Your team can choose either layout, but all of your code must follow your choice.

Mandatory rules

Many of the rules specify spacing to help readers more easily parse your source code.

1 You should apply a single space either side of operators and after keywords (if, while etc)

```
if (x < 10) {
    y = x + 10;
}
```

not

```
if(x<10) {
    y=x+10;
}
```

2 You should have two blank lines between methods

```
    public LandVille(int row, int col)
    {
        // A: constructor method (create and initialize land ar
ray, hasHouse variable)
    }
    public void displayLand()
    {
        // B: displayLand method
    }
```

not

```
    public LandVille(int row, int col)
    {
        // A: constructor method (create and initialize land array, hasHouse variable)
    }
    public void displayLand()
    {
        // B: displayLand method
    }
```

3 Break the code in your methods into paragraphs

In the same way a chapter in a book isn't one continuous wall of text, the code in your methods should also be broken up into chunks (by single blank lines). Each chunk achieves some small goal. For example

```
// read in values

Scanner in = new Scanner(System.in);

System.out.println("enter the first number");
int x = in.nextInt();

System.out.println("enter the second number");
int y = in.nextInt();
```

not

```
// read in values
```

```
Scanner in = new Scanner(System.in);
System.out.println("enter the first number");
int x = in.nextInt();
System.out.println("enter the second number");
int y = in.nextInt();
```

Deciding what chunking you should do takes some practice.

4 Comments

Comments should always be on their own line, and not at the end of a line of code.

```
// Ensure x is non negative
x = Math.abs(x);
```

not

```
x = Math.abs(x); // Ensure x is non negative
```