Code Style

Randy Shepherd
NYU
Correctness is not Sufficient

- In mathematics, it’s simply important to get the right answer.
  In programming, this is not enough.
- Writing software is part science and part craft!
Programming Languages are for Humans

- The reason programming languages were invented was to make working with computers easier for humans.
- Not only when writing, but (perhaps more importantly) when reading!
- “Always code as if the person who ends up maintaining your code is a violent psychopath who knows where you live.”

- Jeff Atwood “Coding Horror”

  http://www.codinghorror.com/blog/2008/06/coding-for-violent-psychopaths.html
Good Style...

- Makes you a better programmer
- Reduces bugs in your code
- Reduces debugging time
- Reduces cognitive overhead
- Improves your grades
- Earns the love and adoration of your future coworkers
What is 'Good Style'?

- Good style is a subjective matter, and is difficult to define.
- The following are usually considered as part of style:
  → layout
  → indentation
  → naming
  → capitalization
Compare:

```java
if (hours < 24 && minutes < 60 && seconds < 60) {
    return true;
} else {
    return false;
}
```

or

```java
if (hours < 24 && minutes < 60 && seconds < 60) {
    return true;
} else {
    return false;
}
```

with something like

```java
if (hours < 24
    && minutes < 60
    && seconds < 60
)
{
    return true
;}
else
{
    return false
;}
```
# Naming & Capitalization

<table>
<thead>
<tr>
<th>Identifier type</th>
<th>Rules for naming</th>
<th>Examples</th>
</tr>
</thead>
</table>
| **Classes**     | Class names should be nouns in **upperCamelCase**, with the first letter of every word capitalised. Use whole words — avoid acronyms and abbreviations (unless the abbreviation is much more widely used than the long form, such as URL or HTML). | • class Raster;  
• class ImageSprite; |
| **Methods**     | Methods should be verbs in **lowerCamelCase**; that is, with the first letter lowercase and the first letters of subsequent words in uppercase. | • run();  
• runFast();  
• getBackground(); |
| **Variables**   | Local variables, instance variables, and class variables are also written in **lowerCamelCase**. Variable names should not start with underscore (_) or dollar sign ($) characters, even though both are allowed. This is in contrast to other coding conventions that state that underscores should be used to prefix all instance variables. Variable names should be short yet meaningful. The choice of a variable name should be mnemonic — that is, designed to indicate to the casual observer the intent of its use. One-character variable names should be avoided except for temporary "throwaway" variables. Common names for temporary variables are i, j, k, m, and n for integers; c, d, and e for characters. | • int i;  
• char c;  
• float myWidth; |
| **Constants**   | Constants should be written in uppercase characters separated by underscores. Constant names may also contain digits if appropriate, but not as the first character. | • final static int MAX_PARTICIPANTS = 10; |
In Conclusion

- Code should look neat.
- Code should be consistently formatted.
- Things should be named intelligently.
- Code should be self-documenting.