



CMSC 131

Fall 2018

Commenting

- Recall: Two styles for comments
- What is the purpose?
- Where do we put them?
 - At the top of every class
 - Above every method
 - Defines “contract”
 - Pre- and Post-conditions
 - Frequently when declaring a variable
 - Throughout code, in places where clarifications are helpful
- How many comments should we write?
- We will be grading for comments starting with project #4

Floating Point Calculations

Example: FloatingPoints.java
compareDoubles, squareRoot

- What went wrong here?
- Under what circumstances does “roundoff error” occur?
- Why does my calculator appear to be better at this?

How can we “fix” this?
tolerance in our calculation

Pro Tips:

- Don't use == to compare result of calculation with theoretical result
- Use integers whenever possible (e.g. representing money)

Java Packages

- What's a package and why is this useful?
- Declaring the package for a class
- Accessing members of other classes
 - Within the current package
 - From a distant package
 - Using “fully-qualified name”
 - Using import
- Import with wildcard
 - Avoid this (naming collisions)
- `Java.lang` is special

Java Libraries

- What are they and why do we care?
- Where is the documentation?
- Let's study the API for a couple common classes:
 - String
 - Math