TODAY’S TOPIC:

• Last time we took an initial look at Java syntax.

• Today we will look at Java naming conventions.
ALWAYS REMEMBER...

"Write your code as if the next guy to maintain it is a homicidal maniac who knows where you live."

(Sierra and Bates, "Sun Certified Programmer and Developer for Java")

Source: http://thedarkknight.warnerbros.com/dvdsite/

NAMING RULES

Like any civilized programming language, Java has naming rules, that are enforced by the compiler:

1. Must start with a letter or _ or $
2. Following characters must be letters, numbers, _ or $
3. Cannot contain special characters or spaces.
   (Apparently $ and _ are not as special as you think).
4. Cannot be a reserved word.
5. Are case sensitive
And just like in real life, the rules aren’t strict enough. There are additional conventions that are used by most programmers.

1. Even if Sun says, $ is a special character, don’t use it.
2. Actually _, is pretty special too, so don’t use it either, well most of the time anyway.
3. It’s really cool to uppercase the first letter of each word so do that.
4. It’s helpful to differentiate what we are naming, so use different conventions for different things.
5. This is the 21st Century, don’t use short non-descriptive names like x or num.

Class & Interface Names
- Upper case first letter.
- Upper case the first letter of each word.
- Typically are nouns (they are objects after all).
- Example: String, Die, BankAccount, Employee, JPanel

Variable Names
- Lower case first letter.
- Upper case the first letter of each word after that.
- Typically are nouns (they are attributes).
- Example: numberOfSides, accountNumber, employeeld
NAMING CONVENTIONS

• Constant Names
  – Upper case the entire name.
  – Use _ between words.
  – Typically are nouns.
  – Example:  PI, DEFAULT_WIDTH, TAX_RATE

• Package Names
  – Lower case the entire name.
  – Example:  java.lang, userinterface, com.tools

NAMING CONVENTIONS

• Method Names
  – Lower case first letter.
  – Upper case the first letter of each word after that.
  – Typically are verbs (they are actions on an object).
  – Example:  roll, getAge, calculateAirSpeed
COMMENTING FOR JAVADOC

• Included in the JDK is a Javadoc program.
  
  – This program will create the Java APIs for your code, provided you include the correct comment structure.
  
  – To do this you can use special block comments, with predefined tags using @

• You may include a comment block at the top of your code to document the class itself:
  /**
   * The Die class represents a die of variable sides, and can be rolled to return a random value.
   * @author Michael J. Holmes
   * @version 2.0 Jan 8, 2015.
   */

• You may also include a comment block for each method:
  /**
   * Rolls the die to get a random value.
   * @return A random value between 1 and the number of sides.
   */
DIE ROLLER EXAMPLES

• Let’s review a couple more program examples using our Die objects.

USING THE DIE API

• Let’s take a look at the Application Programmer's Interface (API) for the Die class.