Lesson 03 – Java Programs, Scanner Class Methods, Local and Global Variables

a. Java hierarchy
   - **package** (conceptually a folder)
   - **class**
   - **field** (global variable)
   - **method**
   - **statement**

b. using the **Scanner class** to do input. The following are methods in the Scanner class:
   - `next()` //get a String delimited by whitespace
   - `nextLine()` //get rest of line as a String
   - `nextInt()` //get an int
   - `nextDouble()` //get a double
   - `hasNext()` //useful to check for end-of-input

c. writing **sentinel-controlled loops** using the `hasNext()` method
   - Examples:
     - `while (input.hasNext())`
     - `while (inFile.hasNext())`
     - `while (choice != 4)`

d. Java **comments**
   - single line comment
     ```
     //calculate taxes and return net pay
     double totalRevenue; //totalRevenue is an accumulator
     ```
   - multiple line comment
     ```
     /*
     alsdkfjlsdjflsdfjksdfj
     lasdjljsdlfjlsjflsdjflsdfjlsdfjlsdfjlsdf
     */
     ```

e. **local** and **global** variables
   - **local**
     1. declared within a method
     2. not automatically initialized
   - **global**
     1. declared at class level (outside of any method)
     2. automatically initialized to the default for the data type
       - **primitive types**
         - numeric -> 0
         - char -> null
         - boolean -> false
       - **reference types**
         - all reference types are initialized to null
f. named constants
   1. all caps with words separated by an underscore (_)
   2. use of final keyword

   ```java
   final double PI = 3.14159;
   final int CLASS_SIZ = 32;
   final double FED_TAX_RATE = .20;
   ```

g. using methods [modularization]
   • declaring a method
     1. method header
        • method return type
           1. void
           2. primitive data type
           3. reference type [class name]
        • method name
        • parameter list

        ```java
        public static void main(String args[])
        ```

     2. method body
        • Statements inside curly braces {   }
          statements always end with a semi-colon (;) in Java

     • invoking (calling) a method
       1. To call a method in Java use the method name and an argument list
          The argument list will contain 0 or more items.

       2. Examples:
          • initialize();
          • Math.pow( 2, 3 );
          • sum( 1, 3, 8, 4, 9 );
          • outFile.println("Girl Scout Troop 10");
          • System.out.println(num1 + " + " + num2 + " = " + (num1+num2) );
            (num1+num2) is an expression Java evaluates during output