Chapter 6 CRT - Sophia Chen - CompSci 20

2. The method declaration is the first line of a method, containing its name, return type, and any parameters needed for the method. On the other hand, the method body is the statements that implement a method.
3. The access modifier is the type of keyword that is used to change the access level of a method.
4. Another word used for describing the access level of a method is visibility.
5. var1: has a local scope to the main() method.
   var2: has a local scope to the for loop in the main() method.
   Var3: has a local scope to the method1() method
   var4: has a local scope to the for loop in the method1() method

6.
   A. public static int getVowels (String input) {
      }

   B. public static int extractDigit (int input) {
      }

   C. public static String insertString (String input, int input1) {
      }

7. a) The compiler is able to distinguish one method from another by looking at the differences of method names, and the number, order, and type of parameters that make up each method.
   b) Two methods in the same class can have the same name, as long as their parameters are not the exact same. This is called method overloading, where they can have the same name, but the order, number, and types of parameters must be different in each.

8. a) The return statement is used to send a value back to the calling statement from the method.
   b) The return statement is only able to return one value back to the calling statement at a time.
   c) The declaration of a method returning a value has a specified data type that the method will return, such as int, String, or double. On the other hand, the declaration of a method that will not return a value simply has the keyword “void” in place of the specified data type, as it will not need to return anything to the calling statement.

9. The error in the code lies in the line “doSomething();”. This line calls the method with a return value, but does not explicitly use or store the value anywhere, essentially making this line of code unnecessary.