2:
Method declaration gives its usage via name, return type (string, integer, boolean), and parameters while method body is the code

3:
Access modifier

4:
Visibility

5:
Var1:
In the “main” method

Var2:
In the for loop which is in the main method

Var3:
In the “method1” method

Var4:
In the for loop which is in the method1 method

6:
A:
private static int getVowels(String[] args)

B:
private static int extractDigit(int[] ints)

C:
private static int insertString(String[] args , int[] ints)

7:
A:
Types of values it gets

B:
No, if 2 methods had the same name and one was called upon, there would be an error or only one of the 2 methods would run

8:
A:
Returns a value

B:
Returns 1 value max

C:
private static String extractDigit(String[] args)
private static void extractDigit(String[] args)
Void = no value returned
String = String value returned

9:
Method “main” is not in a class unlike method “doSomething” which is in class the “MethodCallExample”

11:

A) True (in vocab list on the page before)
B) False (you don't need to declare the method again to call upon it)
C) False (void means no value returned)
D) False (access modifiers only change how it can be accessed)
E) True (on page before)
F) True (things like loops, methods and classes have curly brackets)
G) False (limited to only the method it was declared in)
H) True (can be changed if passed to a method)
I) False (2+ methods with the same name in a class, 1o different methods are okay)
J) True (thats why its called a “return” statement)
K) True (states initial requirements)
L) False (postcondition is what has to be true after code is executed)