**Worksheet 7 - Methods**

1. Write true or false for each statement below.

|  |  |
| --- | --- |
| **Statement** | **True/False** |
| A method is a collection of statements |  |
| A method has a name which must be a valid identifier |  |
| Methods make our code more organised |  |
| The *main* is a special method |  |
| A method must return a value |  |
| A method must have input parameters |  |
| Methods in a class run in the sequence in which they are written |  |
| A method which returns *void* is a method which is empty |  |

1. For each method signature below, mark whether it is valid (compiles) or not.

|  |  |
| --- | --- |
| **Method Signature** | **✓ OR 🗶** |
| public static myNiceMethod() |  |
| public static void myNiceMethod() |  |
| public static int myNiceMethod(int i) |  |
| public static void if() |  |
| public static myMethod(String s) |  |
| public static float myMethod(String s) |  |

1. a) Write a method called *alarm* that prints the word "Alarm!" multiple times on separate lines. The method should accept an integer parameter that specifies how many times the output line is printed.

b) Test your method by calling it from the main method. Before calling it, ask the user for the integer.

1. a) Write a method called sum100 that returns the sum of the integers from 1 to 100.

b) Test the method by calling it from the main method, and afterwards outputting the result.

1. a) Write a method *isEven* that determines whether an integer is even or odd. The method should take an integer argument and return true if the integer is even. If the integer is odd, it should return false.

b) Test your method by calling it from the main method. Before calling it, ask the user for the integer. After calling it, show the result (“is even” or “is odd”) to the user.

1. a) Write a method *multiple* that determines, for a pair of integers, whether the second integer is a multiple of the first. The method should take two integer arguments, return true if the second is a multiple of the first, and return false otherwise.

b) Test your method by calling it from the main method. Before calling it, ask the user for the integers. After calling it, show the result (“is multiple” or “is not multiple”) to the user.

1. a) Write a method called *average* that accepts three integers and returns their average as a floating point value.

b) Test your method by calling it from the main method. Before calling it, ask the user for the integers. After calling it, show the result to the user.