**Building Nicer Programs**

This lab is intended to allow you to practice writing methods as well as calling them.

1. Match the three components of a method declaration to the method below.
   1. Arguments
   2. Method Name
   3. Return Type

\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_\_\_\_

def multiplyNumbers(num1, num2):

return num1 \* num2

1. What is the keyword used when you wish for a method to return no data?
2. What is the term for when a class has more than one method declared with the same name?
3. What is the keyword used to create an empty method?
4. What is the name of the method of the following method declaration?

def random():

pass

1. What could be the return type of the following method declaration?

def longerString(word):

return word + word

1. What is the list of arguments of the following method declaration?

def min(a, b):

pass

1. What could the method declaration look like for the following method call?

action = action.addSuffix(“start”)

1. What could the method declaration look like for the following method call?

val = add(6, 53)

1. What could the method declaration look like for the following method call?

depositMoney(2000)

1. What could the method declaration look like for the following method call?

depositMoney(2000.0)

**\*Note:** For each of the following questions, you should actually write them in your IntelliJ in order to test them. Then, copy and paste your methods below each question.

1. Write a program with two methods both called add one that accepts two integer variables and the other accepts two double variables both called a and b. The add methods should return the result of a and b added together. Call these methods in your main method to test their functionality.
2. Jack is inventing a new language, “Jack Latin”, based on English according to the following rules. Every English word that starts with a vowel now has a -yay added to the end of it. So, “apple” is now “apple-yay”. Every word that starts with a consonant now has -ay added to the end of it. So, “kiwi” is now “kiwi-ay”. Write a program with a method called convertToJackLatin that takes any English word and converts it to JackLatin.

The input and output should be formatted as such:

*Please input a word you wish to translate to Jack Latin: banana*

*“banana” in Gao Latin is: banana-ay*

1. Write a program with a method called makeAThird that accepts two double variables that represent the lengths of two poles called poleOne and poleTwo. Imagine cutting one of these two poles to make three poles in total. The makeAThird method calculates the maximum possible length of the shortest pole among the three. Call the method in your main method to test its functionality. (Make sure to add static to the modifier of your method header)

The input and output should be formatted as such:

*Please input the size of the first pole: 6.8*

*Please input the size of the second pole: 10.0*

*The length of all three poles is at least: 5.0*

*NOTE: The length of each pole is at least 5.0 because if we cut the 10.0 long pole in half, we end up with a 6.8 pole and two 5.0 poles.*

| Input | Output |
| --- | --- |
| 5.0, 7.0 | 3.5 |
| 15.0, 6.0 | 6.0 |
| 10.0, 3.9 | 3.9 |
| 1.0, 1.5 | 0.75 |

1. Write a program with a method called minutesBetween that accepts two times (HH:MM) as Strings called timeOne and timeTwo and calculates the number of minutes between those two times. If the first time is earlier than the second time, assume that both times are PM. If the first time is later than the second time, assume that the first time is AM and the second one is PM.

The input and output should be formatted as such:

*Please input the size of the first time: 10:45*

*Please input the size of the second pole: 1:15*

*The number of minutes between these two times is: 150*