Programming 1

Assignment 3

Due Date: Oct-12 (Sat) 23:59:59. **Late submission will directly be marked as 0**.

Submission: Please submit one .java file for each task (3 .java files in total). Do not zip your files but submit one single file each time (Omnivox accepts multi-submission).

Full Score: 100

Requirement:

1. Make sure you write documentation for each task, (don't forget the @author).
2. **Keep your code clean** (indent, right spacing, appropriate identifier, camel case). Each part that is not clean will get -1 penalty.
3. Add appropriate comments if it is necessary.
4. Make sure you understand your own code and can explain it in front of another person. Student may be randomly selected to ask to explain her/his code. **Failing to explain your own code will be heavily panelized**.

Task 1:

Create a project **SimpleCalculator** which contains

Method:

1. **double calcResult(double num1, double num2)**: calculate addition, with two numbers: **num1** and **num2**
2. **double calcResult(double num1, double num2, char oper)**: calculate +, -, *, or / based on the **oper**, with two numbers: **num1** and **num2**. **(Assume the user will only input +, -, *, or /)**
3. **double calcResult(String formula)**: the formula (**num1 oper num2**) looks like: "31 * 2" or "3.14 + 7.1", (there is a space between **num1** and **oper**, and a space between **oper** and **num2**), you have to extract the **num1**, the **oper**, the **num2** from the string, and then calculate the result.

In the **main()**, ask the user to input numbers, and the operator, or a formula, then call the 3 methods to calculate the result, and then print the result.

```
run:
Please enter two numbers, separated by space: 3.14 2
Please enter the operator (+, -, *, or /): *
Please enter a formular, e.g.: "3.14 * 2": 15.78 / 4.19

Calling the first method  : 3.14 + 2.00 = 5.14
Calling the second method : 3.14 * 2.00 = 6.28
Calling the third method  : 15.78 / 4.19 = 3.77
BUILD SUCCESSFUL (total time: 8 seconds)
```

Task 2:

Create a project **CaseConverter** which contains

Method:

1. **String toTitleCase(String word)**: convert the case of the string **word** as "Xxxxx"
2. **String convertCase(String word)**: convert the case of the string **word** as "Xxxxx" (you should call **toTitleCase()** method)
3. **String convertCase(String word, char caseType)**: convert the case of the string **word** based on the second parameter: '**u**' '**U**': uppercase "XXXXX", '**l**' '**L**': lowercase "xxxxx", '**t**' '**T**' titlecase "Xxxxx" . **(Assume the user will only input 'u' 'U', 'l' 'L', or 't' 'T')**

Hint: You should first convert the **caseType** variable to either uppercase or lowercase, so it can accept both 'u' and 'U', 'l' and 'L', 't' and 'T'.

In the **main()**, ask the user to input a word, and choose the case type, then call the two **convertCase** methods to convert the word, and then print the result.

```
run:
Please enter a word: hElLo
Please choose the case you want to convert:
1. 'l' or 'L' for lowercase
2. 'u' or 'U' for uppercase
3. 't' or 'T' for titlecase
u

Original word                          : hElLo
Calling the first convertCase method : Hello
Calling the second convertCase method: HELLO
BUILD SUCCESSFUL (total time: 5 seconds)
```

Task 3

Create a project **ModuloChecker** which contains

Method:

1. **int letterToNumber(char c)**: take an English letter, (a-z and A-Z), then convert the letter into a number, ‘a’ or ‘A’ -> 0, ‘b’ or ‘B’ -> 1, ….. ‘z’ or ‘Z’ to 25.
2. **boolean isDivisible(int num)**: take one number and check if the number is divisible by 3.
3. **boolean isDivisible(int num, int base)**: take a number and a base, check if the number is divisible by the **base**.
4. **boolean isDivisible(char letter)**: take an English letter, (a-z and A-Z), then convert the letter into a number, ‘a’ or ‘A’ -> 0, ‘b’ or ‘B’ -> 1, ….. ‘z’ or ‘Z’ to 25, (call **letterToNumber** method), and check if the that number is divisible by 3.
5. **boolean isDivisible(char letter, int base)**: take an English letter, (a-z and A-Z), then convert the letter into a number, ‘a’ or ‘A’ -> 0, ‘b’ or ‘B’ -> 1, ….. ‘z’ or ‘Z’ to 25, and check if the that number is divisible by the **base**.
6. **boolean isDivisible(char letter, char base)**: take an English letter, (a-z and A-Z), then convert the letter into a number, ‘a’ or ‘A’ -> 0, ‘b’ or ‘B’ -> 1, ….. ‘z’ or ‘Z’ to 25, and check if the that number is divisible by the **base**, (the base should also be converted into a number by using **letterToNumber** method).

In the **main()**, ask the user to input a number, and a base, then call the first two **isDivisible** methods to check if the number is divisible, and then print the result. After that, ask the user to input an English letter, a number base, and an English letter base, and then call the last three **isDivisible** method check if the number is divisible, and then print the result

```
run:
Please enter a number and a base, separated by space: 9 3
9 is divisible by 3

Please enter an English letter: G
Please enter a number base and an English letter base, separated by space: 5 b

'G' is divisible by 5
'G' is divisible by 'b'
```