SE116 – LAB#2 2021-2022 SPRING

Aim: Program development using classes and methods in Java programming language.

TO DO @ LAB:

1. Implement a sample program to provide its users with the following operations of a **scientific calculator**: power, square root, maximum, and logarithm. In your **source code**, you are not expected to define new methods for these operations. You will just define a `main` method in a sample `class`. In `main`, use `switch-case` (or `if-else`) statements to select the operations. First of all, print out the menu (the list of options from which the user can choose), then read the user's choice for the operation and read also the related argument values of the operation, then call the corresponding method of class `Math` to compute the result of the operation, and finally print out the result. By the way, use an **infinite loop** in your program and terminate it upon the user's related choice.
In order to handle the scientific calculator operations, you are supposed to call the corresponding `static` methods provided by class `Math`. The corresponding methods for this Lab work are the following ones:

- `pow`
- `sqrt`
- `max`
- `log`

These methods take `double` parameters and return `double`.

Please consider the following sample menu displayed at the beginning of the program:

```
A sample scientific calculator is running now ...
Here are the menu options for your choice [1 through 5] :
*** Press 1 to calculate POWER       ***
*** Press 2 to calculate SQUARE ROOT ***
*** Press 3 to calculate MAXIMUM     ***
*** Press 4 to calculate LOGARITHM   ***
*** Press 5 to TERMINATE the program ***
Your choice? :
```

TO DO @ HOME:

2.
A. Modify your program to contain a package with the following two classes: `ScientificCalculator` and `Driver`.
The class `ScientificCalculator` will contain the following static methods: `displayMenu`, `calculatePower`, `calculateSquareRoot`, `calculateMaximum`, and `calculateLogarithm`. The method `displayMenu` prints out the menu; whereas the other methods compute the corresponding operations. To implement these methods, you are free in your design to calculate the related operations (either code the algorithms of the operations yourself or call the corresponding `static` methods of class `Math`).
The class `Driver` will just contain a `main` method to test the methods of class `ScientificCalculator`. Illustrate the `main` method similar to the one in your first program.

B. Modify your class `ScientificCalculator` to contain one more static method called `calculateFactorial`. This method shall take a nonnegative integer parameter and return its factorial value. Try to use a recursive algorithm to compute the factorial value.