# CST8110 Assignment #2 [5%]

**This assignment relates to the following Course Learning Requirements:**

CLR 1: Define, analyze, and document the logic of a solution to a given problem.

CLR 2: Implement the solution to a given problem by writing the appropriate code in a high-level language (Java).

CLR 4: Install and use the Java Development, Runtime Environment, and documentation libraries.

CLR 7: Create documentation and a Java solution for programming problems that adhere to the submission standard identified within the timeframe given in the problem description.

Objective of this Assignment:

The following is an exercise to help you understand what you have learned in this module. It will allow you to familiarize yourself with the module content as well as practice the skills required to develop software using the Java programming language. You will also be asked to use Scanner and Math classes from the Java API, write a single class application with 3 methods to input and output a single number.   
***In this assignment you will read an input from the user using a Scanner object and display the result based on either of the following three cases: display the same the number read from the user if the number is between 1 and 10, display 1 if the number is less than 1, display 10 if the input number is greater than 10. You can do this use appropriate you of the max and min function between either 1, 10 and the input number.***

**You are free to use any logic to achieve the goal but you are not allowed to use conditional statements like if, if/else, or switch statements or any equivalent short form of any of them.**

# Pre-Assignment Instructions:

1. To prepare you for this assignment, please complete the content associated to the module.
2. Complete all module reading.
3. Using the book “Java How to Program” read Chapter 3, Appendices D & I (Online only)
4. Complete all reference video tutorials
5. Develop this program locally on your machine in a plain text editor or other IDE

**Assignment Tasks/Steps:**

* Create a class called **TestMax**
  1. Declare two instance variables of type int: **minNum** and **maxNum**
     1. Assign the value 1 to minNum, and the value 10 to maxNum
  2. Define three methods with the following details:
     1. **public int inputNum()**
        1. Prompt the user to enter a number (i.e. use either of print/println/printf to print an appropriate message)
        2. Uses a **Scanner** object to allow a user to enter an integer number (you may assume they will always enter an integer)
        3. The method should return the number input by the user
     2. **public void displayNum(int userNum)**
        1. Use the **min and max** methods of the **Math** class (see Lab2) for the minNum and maxNum variables

Hint: think about how to use of min and max function with userNum, minNum and maxNum to display either the userNum if it is between 1 and 10, or display 1 if userNum is less than 1, or display 10 if userNum is greater than 10.

**You are free to use any logic to achieve the goal but you are not allowed to use conditional statements like if, if/else, or switch statements or any equivalent short form of any of them.**

* + - 1. Use **printf** method from System.out to display the result
    1. **public static void main(String[] args)**
       1. Declare a variable of type TestMax (name of the class) (i.e. class name myVar;)
       2. Initialize and instantiate the TestMaxobject (myVar= new ….)
       3. Using your TestMax object:
          1. call the inputNum method to read a number from the user
          2. pass the result from the previous step to the displayNum method.
  1. If needed, you may store the interim values in local variables.
  2. Sample test cases to check your logic:
     1. input: 10 [output: 10]
     2. input: 6 [output: 6]
     3. input: 3 [output: 3]
     4. input: -1 [output: 1]
     5. input: 0 [output: 1]

**Submit Your Assignment**

Your assignment is to be submitted on Brightspace as Java files. It should be submitted with the following guidelines:

1. Include the file header using the template provided in Assignment 1
2. Use the following expected style guidelines:
   * 1. Use proper braces for all classes and methods **(stick to your style in all methods)**
     2. Indent your code using 4 spaces (no tabs)
     3. Follow naming conventions as instructed in the steps for all class, variable, and method identifiers
     4. Use appropriate whitespace for readability
     5. Comment your code (no less than one comment per class, and one comment per method)
3. The file must be named **TestMax.java**
4. Method names must match those in the requirements above, exactly.

Feel free to make any assumptions required, in order to implement a working code. Make sure the code delivers expected successful output. Highlight any assumption you may make while submitting the assessment.

**Assignment Grading Rubric (5%)**

|  |  |
| --- | --- |
| Criteria / Marks | Points |
| Program compiles | /1 |
| Program executes and outputs as expected from the updates performed by the learner  (see sample input/output as a guidance) | /1.5 |
| Header file is inserted at top of file and all variables and place holder and information replaced with learner’s information | /0.5 |
| The style guidelines are followed:  i. Use proper style braces for all classes and methods (stick to your style in all methods)  ii. Indent your code using 4 spaces (no tabs)  iii. Follow naming conventions for all class, variable, and method identifiers  iv. Use appropriate whitespace for readability  v. Comment your code (no less than one comment per class, and one comment per method) | /1.5 |
| Method names match those in the requirements exactly | /0.5 |
| Comments |  |
| Total Points | /5 |