**Objective**

Create a helper namespace header file that will provide an easy-to-use interface for frequently used functionality.

**Namespace**

Create a new Helper.h header file. Within it, declare a namespace named 'Helper'. Inside the namespace, declare AND define (within the namespace body { }) your first helper method:

**int GetValidatedInt(const char\* strMessage, int nMinimumRange = 0, int nMaximumRange = 0);**

This function should first display the provided message to the screen then use cin to get an int from the user. Error check and validate to ensure a legal integer was entered. If not, clear the cin buffer using clear() and ignore() and try again (Note: the buffer still needs to be cleared even if this step was successful).

If a legal integer was entered, check its value to see if it is within the provided minimum and maximum range (inclusive). If the user input was within the range, return the integer. If not, get input again. The range check should be ignored if BOTH the minimum and maximum parameters are 0, so any valid integer is acceptable.

You should be using a loop so that the method does not return until a valid and within range (if the range is not 0) integer was totally provided.

**Customize**

Create at least two more helper functions that will handle generic code that you find yourself consistently re-writing in SPR. Some helper ideas include:

* Return random number within a range
* Print a menu (array of strings) and return the user’s selection
* Clear and ignore the cin input buffer
* Print a provided integer in binary, hex, or oct
* Copy string reference