CPSC 121

Week 12
Smart Pointers

Lab 10 - Practice with Smart Pointers

Create a class that manages a dice matching game. There are 3 dice randomly being rolled. Create a default constructor that randomly assigns values to the dice(1-6), and a deconstructor that prints when the object is being destroyed. Create functions that display the dice, and allows the user to reroll one of the dice. Additionally check to see how many of the dice match.

For this lab you must dynamically allocate an instance of the classes using only smart pointers. The object will be managed by using two shared pointers. The shared pointers will be tested to see if they point to an object then *reset()* and then display the number of shared objects in its grouping. Please refer to the example output on the back.

Useful Code:

```
#include <time.h>
srand(time(0));                    //seed the timer to reset
rollADie = rand() % 6 + 1;         // generate number in the range of 1 - 6
```

Sample Cases To Consider		
Dice #1 is 3 **Dice #2 is 3** **Dice #3 is 3** **all three dice are equal**	**Dice #1 is 2** **Dice #2 is 2** **Dice #3 is 5** **two dice are equal**	**Dice #1 is 3** **Dice #2 is 6** **Dice #3 is 4** **no dice are equal**

Example Output:

===Creating Shared Pointer #1 to manage the object===
Dice #1 is 4
Dice #2 is 1
Dice #3 is 1
two dice are equal

===Now creating Shared Pointer #2 to manage the same object===
Pick a dice to reroll
1
Dice #1 is 1
Dice #2 is 1
Dice #3 is 1
all three dice are equal

==Checking if pointers are being utilized==
Ptr 1 currently points to an object
Ptr 2 currently points to an object

Ptr 1's # of references in shared grouping: 2
Ptr 2's # of references in shared grouping: 2

===Releasing Pointer #1===
==Checking if pointers are being utilized==
Ptr 1 currently points to no object
Ptr 2 currently points to an object

Ptr 1's # of references in shared grouping: 0
Ptr 2's # of references in shared grouping: 1

===Releasing Pointer #2===
No shared Pointers left to manage dice object, Deconstructor called on dice object!
==Checking if pointers are being utilized==
Ptr 1 currently points to no object
Ptr 2 currently points to no object

Ptr 1's # of references in shared grouping: 0
Ptr 2's # of references in shared grouping: 0