CST126 W19 Lab 2 Bottles

# Introduction

In this lab we will be practicing the use of structures and classes by storing and printing a collection of bottles. (I got the information about bottles here: http://www.bottlebooks.com/determin.htm

# Learning objectives

* Reviewing Structures
* Classes
* Const correctness

# Requirements

* Store and print a list of bottles that is part of someone’s collection. You will get the list of bottles from a text file. I’ve provided a couple text files for you. We will grade with big.txt.
* Your text file reading doesn’t need to be fancy. The data is space delimited.
* There are a maximum of 30 records in the file. I made a constant for this.
* You may use the string class for any text variables.
* Use functions and methods in a logical way. Don’t have big functions or methods.
* Your classes and structures must:
  + Each have their own .h and .cpp. Only methods that are 1 line may be inline (in the .h)
  + Be const correct. Don’t underestimate the work of this the first-time round. (It will be easier if you make it a habit and just start const correct.)
  + Your .h must either use #ifndef or #pragma once. If you use #pragma once you must provide a comment that describe how it works and demonstrates that you understand it.
  + Note: Since you are adding .h and .cpp for your classes you will be pushing new files to github. You can do it!
* You will have at least 1 **class** to store the information about a single bottle.
  + The information we care about is:
    - * The color – One word to describe the color.
      * The type – One word to describe the type of bottle: i.e. flask, syrup.
      * The age – an integer to indicate how old the bottle is.
      * The value – a double to indicate how much the bottle is worth.
  + To stay const correct you will need getter functions for all the member data. These functions can be defined in the .h since they will only be a line long.
  + A print method that prints a single bottle.
  + A default constructor that blanks or zeros all the data.
  + A constructor that takes all 4 pieces of data (color, type, age, value) and calls the initialization function below.
  + You will need a method to initialize a bottle. As a hint here is the signature from mine:

void init (const string & color, const string & cat, int age, double value);

* You will have another **class** or **structure** to represent the collection of bottles. It will be built as you read the file and add bottles.
  + It has the following member data.
    - An array for your bottles. It is at most 30 bottles (see note above about file size.)
    - An integer to indicate how many bottles you actually have at this moment. (So for example the test.txt file has 7 as the size.
  + A method to provide the total value of the collection. This will simply loop through the array of bottles and add the values.
  + A method to print the collection. It will also print the count of bottles and the total value.
  + Note that I said you can do this as a structure. You can also do it as a class with private member data. If you do this you can get some extra credit because this will be harder. I strongly suggest that you get it to work as a structure before adding the private member data. (Also, I’d suggest moving one piece of data to private at a time to limit your exposure to errors.)
* Please do this incrementally. In the program, I added comments with most of the steps that I did. My steps are only a suggestion. Your steps will likely be smaller as this is your first time at classes. Just do a small bit and compile and test.

# Output

Here is an example of the output from my program for test.txt . You don’t have to copy it exactly. Just be self-documenting and not ugly.

