Programming Assignment #1

Warehouse Bins

CS 2308.255 Spring 2020
Instructor: Jill Seaman

Due: Monday, 2/3/2020: upload electronic copy by 11:59pm!

Each bin in a warehouse holds a quantity of the same type of parts. Your program should use a structure that keeps the following data for each part:
- Description of the part kept in the bin
- Number of parts in the bin
- Price of the part

The program should have an array of 10 bins, initialized with the following data:

Part Description	Quantity	Price
Chain	10	8.99
Handlebar	5	19.95
Pedals	15	7.99
Chainring	21	8.99
Cassette	7	10.99
Wheel	5	53.99
Saddle	5	29.95
Derailleur	25	11.99
Seat post	18	7.49
Tire	12	9.99

When the program runs, it should repeat a loop that performs the following steps: The user should see a list of what each bin holds and how many parts are in each bin. The user can choose to either quit the program or select a bin. When a bin is selected, the user can either add parts to it or remove parts from it. The loop then repeats, showing the updated bin data on the screen.

When the user quits the program, the program should display a message indicating the total value of the current inventory, and which part has the maximum quantity. Please

see sample output in the file **output1.txt** on the class website. Note: your output should look <u>at least</u> as nice as mine (data lined up in columns, money formatted to two decimal points, etc).

Your program must include four functions:
- one to display the parts and quantity (NOT the prices),
- one to process the user's selection (to add or remove a certain number of parts for a given bin),
- one to determine the total value of the current inventory, and
- one to determine which part has the highest quantity.

These functions should take an array of Category (and perhaps other values) as arguments.

Additional Requirements:

- Your program **must compile** and run, otherwise you will receive a score of 0.

- Your program must give correct results for any valid input data. You should make it easy for another programmer to change the part names, quantity, prices, number of parts and if so the program should still give correct results.

- Do not accept input that would make the quantity go below 0 (give an error message instead).

- If more than one part shares the maximum quantity, your program should output any one of them.

- Please do not use any features of C++ that we have not yet covered in class (use features from Chapters 1-7, 11 only). Do not use classes!

- **Style**: See the Style Guidelines document on the course website. Especially pay attention to the **comments** required for the top of the file and for each function. The grader will (potentially) deduct points if your program violates the style guidelines.

Logistics:

Name your file **assign1_xxxxx.cpp** where xxxxx is your TX State NetID (your txstate.edu email id).

Submit an **electronic copy** using the Assignments tool on the Canvas website for this class (canvas.txstate.edu).

See the assignment policy on the course website (cs.txstate.edu/~js236/cs2308) for more details, including late deadlines and penalties.