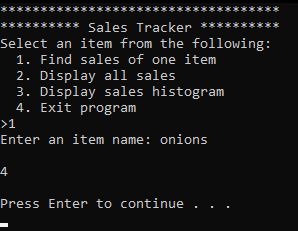
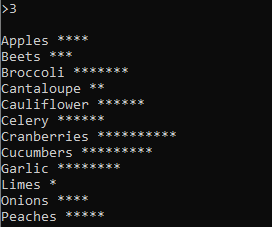
Taylor Colton

August 10, 2024

Project 3 Documentation: Corner Grocer

My code develops a sales tracking application that allows users to interact with item data through numerous capabilities by reading the data from a file. A set of items and the associated sales counts are managed by the SalesTracker class; these are kept in a std::map with integer values and string keys. The application provides a menu-driven interface that enables users to print a sales data histogram, view every item with its sales total, and query the frequency of particular goods. The program contains capabilities to back up the sales data to a file and guarantees that user input is evaluated and written correctly.

While developing my code I decided to install a linter, SonarLint by SonarSource, which showed me some things that I could improve in my code. Specifically regarding using maps. It taught me that using a string for a key can be memory intensive when retrieving values from a map, unless you use a transparent comparator like so:



The linter also gave me tips on catching exceptions and how to do so in a more memory-efficient manner. When catching an exception as a reference, memory isn’t allocated unnecessarily for it. The final improvement I made via the linter’s suggestions was to use *make\_unique* when creating a new object, rather than *new*. This automates memory management for the object.