

Capstone Homework #1 design with tests.

Summary: In this exercise you will do some program design using automated tests

Due: Thursday April 28 at 11:59 pm

Production code:

write a production code function to accomplish the following task, (you can of course have helper functions/methods that are intended to be private, but there should be a single public function that accomplishes the task and gets tested)

Your function needs to calculate the total to charge a customer at checkout. The function should take the users state and a list of records* of items to be purchased, it should return the total that the user will be charged (including sales tax).

This method **cannot/must not** be used to process refunds

Let's limit the states we do business in to New Jersey. Delaware and Pennsylvania. Use their two character abbreviations as a parameter in the function. We will limit the types of goods we sell (in the record type field) to these three for simplicity: "Wic Eligible food", Clothing, and "everything else"

lets talk in class briefly about how these three states differ.

Summary:

DE: no sales tax

NJ: Sales tax rate is 6.6%

PA: Sales tax rate is 6% (we'll ignore the local tax)

WIC Eligible food is tax exempt in all three states

Clothing is exempt from in PA, and NJ **except** Fur clothing is taxable normally in NJ, we'll call it taxable if the string fur appears in a clothing item name

Everything else is covered under the default tax rates

Tests:

write automated tests to thoroughly test the function your are writing above.

Academic Deliverables:

Upload the production and test code to github and invite your instructor to the github repository. Include a readme.md that tells me what your production function is and how to run your tests.

The usual languages are fine for this project, if you want to use something unusual check with me.

*struct in go/rust/c/c++, data class in python, a record or something similar in java – with data members but no useful methods, put the basic record class in your file so your tests can create test work. **The record should have three fields, the item name, the price, and the item type.**