

CSCA20 - Lab 10

Putting it all together/Building up step by step

Learning Objectives

This lab is sort of a meta-lab, that will let you demonstrate your skills in sections based on where you are and what skills you need to demo. The lab will build up through the skill categories of this course, and let you demonstrate the most advanced set of skills you're able to master.

Prelab

This week, we will have 4 tests to write, so all of the tutorial time will be taken up with testing. So you will want to arrive with the lab complete, or at least the parts of the lab you plan to demonstrate. If you need help getting the labs done, you can ask during the tutorials for lab 9, or during office hours, or on Piazza.

Demonstration & Evaluation

This lab lets you demonstrate all skills in the course. The lab is broken up into 4 parts:

- Part A demonstrates all of the foundational skills
- Part B demonstrates all of the cores skills
- Part C demonstrates all of the advanced skills
- Part D demonstrates all of the extension skills

In the tutorial, you will get an opportunity to demonstrate any part you need to, and it is possible to demonstrate multiple parts. But keep in mind that we're testing whether you wrote and understood all the code yourself, it's not about having the code done as much as it is about having done it yourself.

The Scenario

The UTSC (Unbelievably Tiring Semester of Coding) society has decided to use the material you've been building for the last few labs, because hopefully you've already got big parts of the code working, and can re-use code. So instead of a fully new scenario, we're building off the scenario from labs 8-9.

Data Files

You have been provided with several CSV files:

- `budgets.csv` has info on the budget for a shopper
- `foods.csv` has information on foods including the price per item, location of the food's source, and whether it is classified as a fruit
- `people.csv` has information on customers including names, ages, addresses, the number of trips they have made to the store, and whether or not they are a member
- `purchases.csv` has information on which customer purchased which item, in what quantity and on what day

Completing the lab

PART A: Foundational Skills

- Create a menu that asks the user to input the name of a customer (or 'QUIT' to quit)
- For each customer, ask the user how many items the customer purchased
- For each item, ask the user the price
- Print the name of the user and the total they spent

PART B: Core Skills

- Create a menu that asks the user to input the name of a customer (or 'QUIT' to quit)
- For each customer, ask the user how many times the customer visited the store
- For each visit, ask the day of the week, and the total amount spent that day
- Ask the user for a discount day of the week (50% off)¹
- Tell the user the total amount spent by the customer
- You must write and use a function that takes a list of strings representing days of the week for each visit, a list of floats representing amount spent for each visit, and a string representing the discount day, and returns the total spent by that user, summing the values for all days, but applying a 50% discount to all discount days

As an example, if the input was ['MON', 'TUE', 'MON', 'WED'], [1.00, 2.00, 3.00, 4.00], 'MON'. Then the output would be 8.00 $((1.00 * 0.5) + 2.00 + (3.00 * 0.5) + 4.00)$

¹yes, I know this is a bit odd, but the idea is that they user somehow gets back 50% of the amount they spent that day... I sure wish my grocery store did that

PART C: Advanced Skills

- Write a function that takes a CSV file open for reading, a string representing the column to be used as keys, and another string representing the column to be used as values, and produces a dictionary mapping keys to sets of values (e.g., passing in a handle for `purchases.csv`, `person_id`, `day` would return a dictionary mapping customer ids to the set of days on which they made a purchase)
- Use the function (and any other helper functions you wish to create) to produce a system where a user can enter the names of csv files, and produce printed output for the following:
 - all foods purchased by a customer (queried by name)
 - all days on which a customer shopped (queried by name)
 - all customers who have spent more than their budget

PART D: Extension Skills

- In addition to the requirements of PART C, your code should also use `sql_tools` and `csv` import to produce CSV files, as well as `matplotlib` to create scatter plots for the following queries
 - Customer budget vs price of items purchased
 - Customer age vs number of items purchased
- The user should also be able to limit the queries by the following criteria
 - Include all purchases
 - Include only purchases made by members
 - Include only purchases of fruits

Hints

- If you're planning to demonstrate more than one part, have separate copies of your files, so that if something goes wrong when you're editing an earlier part it doesn't impact your later parts
- The parts get progressively more difficult, and we're having to test a lot of pieces at once because we're trying to get it all done in a single lab. So if you can't demonstrate a particular level, don't panic, you'll get another chance on the exam, where we can break things down into simpler pieces, and you'll have more time.