CENT 110 – Final Project

The final project, project 2, will be a group project that involves creating a GUI (Graphical User Interface) program that is capable of reading both JSON and XML input files and putting the data into a table. Here are the groups for the project:

Group 1 : Anthony, Dennis, Connor
Group 2 : Junjie, Irene, Michael
Group 3 : Bradley, Bryson, Brandon
Group 4 : Craig, Les, Kyle
Group 5 : Leo, Jeremy, Austin

Every group gets a common folder under Laulima>Resources to share code and make final submission.

Pyside

The GUI program will be written using Python Pyside. Pyside is the free implementation of the PyQT framework.

Setup

You are supplied with a program "query_sales_gui.py" that can query the database file "book_sales.db". The database file can be created from the command line using:

sqlite3 book_sales.db <init.sql

with the "init.sql" that is provided. But, you can also just use the attached "book_sales.db" file. sqlite3 can be installed using:

sudo apt install sqlite3

"query_sales_gui.py" is a PySide program that can connect to the database file and run queries on the database. Once a query is run, the program can save to either JSON or XML files.

Your job is to come up with "proj2.py" which uses the QtGui.QTableWidget table to map the data from a JSON or XML file to a table and display the table.

The idea is to create different JSON (or XML) files by running different queries in "query_sales_gui.py" and saving to different filenames. Then, "proj2.py" can open any of those files and dynamically reset the table to match that data.

Basic Requirements

A completed project must be able to perform the following:

1. Read in a JSON file and place the contents in a QTableWidget object that will serve as the main table. The main table should resize itself so that the number of rows in the table is equal to the number of items from the JSON file. The QTableWidget will scroll, so you don't have to worry about a file that has more rows than can be displayed in the size you allot for the table. Also, the column widths should adjust to the actual data.

2. Read in a XML file and place the contents in the main table.

3. Sort by each of the columns. When a column menu is clicked the order is changed to either ascending or descending. When sorting by price, make sure to sort the items by their numerical value. Right now the prices are treated as Strings, not floats. So 14.99 is considered less than 2.50.

4. Using the items in the main table, write to a HTML file that displays the table data in a HTML table.

5. Provide a way for the user to add additional data to the main table, one row at a time.

6. Add a feature to save the main table contents to a JSON file with the same contents.

7. Add a feature to save the main table contents to an XML file with the same contents.

Grading

- 70 points – The project implements all the requirements above. You will present each feature and also any extra credit features you incorporated. Your presentation will include an explanation of the most difficult problem the group encountered, as well as how the problem was dealt with.
- 10 points – the instructor will evaluate your personal contribution to the project.
- 10 points – Detailed algorithm for the program.
- 10 points – Answers to the following questions:
  1. What are the two advantages of storing data in JSON format?
  2. What is an advantage of using XML over JSON? What is an advantage of using JSON over XML?
  3. What are the two reasons to create a GUI program versus a console-based program?
  4. What is the advantage of using a database to store data?

Extra Credit Options – Maximum extra credit is 10 points

1. (1 point) – Add a feature to open a CSV file in addition to a JSON or XML file.
2. (1 point) – Add a feature to save the main table contents to a CSV file in addition to a XML file.
3. (3 points) – Add a feature so that when opening an XML file or a JSON file, the contents can be added(appended) to the existing table, instead of replacing the existing table contents.
4. (2 points) – Add a feature that can delete the first row of the table or delete the last row of the table.
5. (3 points) – Add a feature that can delete all the selected rows of the table.
6. (10 points) A program that basically combines the two programs – "query_sales_gui.py" and "proj2.py" and skips the step of creating JSON/XML files and goes directly from query to making a new table.

Due Date

The group presentations will be held on Dec 6. The final version of the code must be uploaded by 11:45 am on that day. All other material (besides the code) must be submitted by 5:00 pm that day. All the items submitted should be in a folder called **Project 2 Final Version** inside your group folder.