CPT187 Final Project Instructions Version Date: March 25, 2024

Each student in CPT187 will design, plan, and code a Final Project which counts for 25% of each student's overall grade in the course.

Project Name: Collectors Management Tool

Project Purpose:
A Python program designed to help collectors manage their collection, no matter what they are collecting. Their collection could be items like stamps, coins, art, music, videos, anything of value. The Python program should be designed to allow collectors to customize the database and program for any type of collection, including custom input forms and output records. The project program needs to be written using the skills learned in CPT87. The program should be designed to run in Windows providing a user-friendly graphical interface without any error messages or exceptions. All input data must be validated, and output must be neat and professional. The program must be well documented in the same style that we use in CPT187 Lab Assignments.

Minimum Requirements:
- The program should manage:
    - Categories of items in collection
    - Item name
    - Item description
    - Name of item's source, contact, address, phone, email, phone
    - Price paid for item
    - Current value of item
    - Location where the item is stored, loaned, or displayed
    - Other data points the student developer wants to add

- Required specifications for this project (subject to change):
    - Object-Oriented Programming (OOP)
    - Relational database using SQLite
    - Graphic User Interface (GUI) using tkinter
    - User authentication for admin and users (at least two roles but could be more)
    - Users must be able to add and edit any data they create in the program
    - Only the admin can inactivate any record, rather than delete records
    - Users must be able to search all records (by substring) to edit only their data
    - Admin must be able to search all records (by substring) for edit or inactivation
    - User customizable input forms and output reports
    - Create customized database tables for customized input forms and output reports
- The Python program must be saved as **`collectorsTool.py`**
- Database must be saved in a SQLite SQL file named **`collectorsTool.sqlite`**
- All input data must be validated
- Exception handlers must be coded throughout the program where relavant
- The project program must be well documented including student's name and a project description in the header, as well as explanations for each function and blocks of code
- Both the **`collectorsTool.py`** program and **`collectorsTool.sqlite`** files will be run from the same folder/directory; both files must be submitted by the student for grading

Extra Credit:

- Submit completed project for grading at least one-week prior to the deadline; notify the instructor by email and include in **`extra_credit.txt`** documentation file to receive credit
- CPT187 student added extra features not specified in the Final Project assignment instructions; explain in **`extra_credit.txt`** documentation file

Application Structure:

To leave CPT187 students as much flexibility as possible, the project program structure expectations are limited to the following:

- The interface must be original Python code written by the CPT187 student submitting the project
- Do not use any non-original utilities or other code not created by you in your project *unless* the utility or code is well attributed to the source including providers name and URL to their website download and documentation pages
- The database must be in be saved as an SQLite relational database

Submission Requirements:

All student submissions to Blackboard must include the following files which can run in the same folder/directory on the user's computer:

- Project program named **`collectorsTool.py`**
- Database file named **`collectorsTool.sqlite`**
- Any custom modules created by student
- Explanation documentation of extra credit in **`extra_credit.txt`** file

Disclaimer:

Your instructor reserves the right to change these instructions or the grading rubric as necessary at any time. CPT187 students will be notified of any changes by way of email and announcements on our Blackboard class page.

Questions and Suggestions:

Your feedback about the final project is encouraged. Address all questions and/or suggestions to your instructor at sandewbs@my.gvltec.edu.

Attachments:

Final Project Grading Rubric

CPT187: Object-Oriented Logic and Design
FINAL PROJECT GRADING RUBRIC (v24.1)

Student:

	REQUIRED FINAL PROJECT SPECIFICATIONS:	Points Earned	Points Available	Meets Specifications **100%**	Mostly Meets **80%**	Somewhat Meets **70%**	Needs Improvement **60%**	Does NOT Meet Specs **0%**
1	Project works according to specifications		10					
2	SQLite relational database well designed and normalized		10					
3	Code does not generate any errors or exceptions		10					
4	Project is logical and well designed for ease of operation		10					
5	Interface input screens allow for access to all relavent fields		10					
6	Users and admin can search all data by substring		10					
7	Users can edit data they created but not other users data		10					
8	Output reports are detailed and proferssional		10					
9	Proper data validation of all input data		10					
10	Well commented code documentation per instructions		10					
	TOTAL REQUIRED POINTS		**100**					
	EXTRA CREDIT (*extra_credit.txt documentation file required):							
11	Submitted at least one week early*		10					
12	Additional Feature 1; student added; must be documented*		10					
13	Additional Feature 2; student added; must be documented*		10					
14	Additional Feature 3; student added; must be documented*		10					
15	Additional Feature 4; student added; must be documented*		10					
	TOTAL BONUS POINTS		**50**					
	TOTAL FINAL PROJECT POINTS		**150**					

This grading rubric is subject to change by the instructor