

Lab Exercises - 3

Exercise 1 – Display the following menu to the user:

- 1) Create a new file
- 2) Display the file
- 3) Add a new item to the file

Make a selection to 1, 2 or 3

Ask the user to enter 1, 2 or 3. If they select anything other than 1, 2 and 3, it should display a suitable error message. If they select 1, ask the user to enter a school subject and save it to a new file called "Subjects.txt". It should overwrite any existing file with a new file. If they select 2, display the contents of "Subjects.txt" file. If they select 3, ask the user to enter a new subject and save it to the file and then display the entire content of the file. Run the programme several times to test the options.

Exercise 2 - Create a simple maths quiz that will ask the user for their name and then generate two random questions. Store their name, the questions that were asked, their answers and their final score in a .csv file. Whenever the programme is run it should add to the csv file, and not overwrite anything.

Exercise 3 - (Function and .csv files) Create the following menu:

- 1) Add to file
- 2) View all records
- 3) Quit programme

Enter the number of your selection:

If the user selects 1, allow them to add to a file called Salaries.csv which will store their name and salary. If they select 2 it should display all records in the Salaries.csv file. If they select 3 it should stop the programme. If they select an incorrect option, they should see an error message. They should keep returning to the menu until they select option 3.

Exercise 4 – (function and csv) In Python, it is not technically possible to directly delete a record from a .csv file. Instead, you need to save the file to a temporary list in Python, make the changes to the list and then overwrite the original file with the temporary list.

Change the previous programme to allow you to do this. Your menu should now look like this:

- 1) Add to file
- 2) View all records
- 3) Delete a record
- 4) Quit programme

Exercise 5 - Sum a Collection of Numbers

Create a program that sums all of the numbers entered by the user while ignoring any input that is not a valid number. Your program should display the current sum after each number is entered. It should display an appropriate message after each non-numeric input, and then continue to sum any

additional numbers entered by the user. Exit the program when the user enters a blank line. Ensures that your program works correctly for both integer and floating-point numbers.

SQLite3:

Exercise 17 – Create an SQL database called PhoneBook1 that contains a table called Names with the following data:

| ID | First Name | Surname | Phone Number |
|----|------------|----------|--------------|
| 1 | Simon | Pierre | 0142678 9056 |
| 2 | Katarina | Iglesias | 0203456 7078 |
| 3 | Derrick | Brown | 0122345 8765 |
| 4 | John | Smith | 0112653 2312 |
| 5 | Mark | Isaac | 01416571383 |

Exercise 18 – Using the phonebook database, write a programme that will display the following menu

```
Main menu
```

```
1) View phone book
2) Add to phone book
3) Search for surname
4) Delete person from phone book
5) Quit
Enter your selection
```

If the user selects 1, they should be able to view the entire phonebook. If they select 2, it should allow them to add a new person to the phonebook. If they select 3, it should ask them for a surname and then display only the record of people with the same surname. If they select 4, it should ask for an ID and then delete that record from the table. If they select 5, it should end the programme. Finally, the programme should display a suitable message if they enter an incorrect selection from the menu. They should return to the menu after each action until they select 5.