UCD Analytics Research & Implementation MIS40750 Programming Assignment

Deadline: 5pm Friday 17th February 2017.
Worth: 20% of the module.
Team: please form a team of 2-3 students for this project.

1 Specification

Suppose you work for an airline, and your job is to write software algorithm to allocate seats when passengers make bookings. For each flight, you'll be given the seating configuration of the plane. Then as bookings come in, your job will be to allocate the seats to the passengers. Each booking will be for one or more passengers, and naturally bookings of multiple passengers should be allocated seats together where possible.

The bookings will be provided in a CSV file where each line consists of one integer representing the number of passengers in the party and one name, the name of the person making the booking. You should allocate all seats for the booking in that name. You have to allocate the seats for one booking before looking at the next booking (i.e. the next line).

A database file will be provided representing the seating plan of the aircraft, the empty and occupied seats. After each booking you must update the database with the allocation you have made.

When a booking can be accommodated, you should allocate the seats – together if possible, but split up if necessary. When a booking cannot be accommodated at all, because there are too few free seats, you should not allocate seats.

After each booking has been processed (either allocated or refused) you must also update two metrics in the database:

- a number representing how many passengers have been refused outright (this is total passengers, not number of bookings that have been refused);
- and a number representing how many passengers are seated away from any other member of their party.

Fields for these metrics will already be present in the database.

Your program should be structured into multiple functions as appropriate.

Your program must include appropriate comments and tests for individual functions.

A sample database and sample bookings file are available. You should inspect these to understand their structure and format. Your program will be tested against other files in the same formats.

Grading criteria: firstly, the program must carry out its job correctly. Thereafter, marks will be awarded for submissions that include suitable tests, comments, and documentation (not excessive); appropriate decomposition of the problem into functions, using loops and data structures were appropriate; brevity and clarity; and good coding style.

2 Submission

Submission consists of (1) a single URL pointing to the Github repository for this project and (2) a brief statement that the work is your own and which aspects of the work were carried out by which team member. Both (1) and (2) should be typed directly as submission text in Blackboard.

Your Github repository must include a file named `README.md` with brief documentation and a single a single `.py` file named `seat_assign_12341234_56785678.py` where `12341234` and `56785678` are the ID numbers of the two students carrying out this assignment. Your repository can optionally include some test data.

The Github repository must include at least two commits by each member of the team. The commit history must demonstrate the gradual development of the project rather than including just a few very large commits. Guideline: commits should usually be of less than 50 lines (exception: commits which reformat or reorganise large amounts of existing code).

The user of your program will run it on the command line by typing `python seat_assign_12341234_56785678.py data.db bookings.csv`, where `data.db` is the name of an SQLite database and `bookings.csv` is a file representing the bookings, one per line. Your program must work with other database and bookings filenames as given on the command line.