**[Background](https://cs50.harvard.edu/x/2020/psets/7/houses/" \l "background)**

Hogwarts is in need of a student database. For years, the professors have been maintaing a CSV file containing all of the students’ names and houses and years. But that file didn’t make it particularly easy to get access to certain data, such as a roster of all the Ravenclaw students, or an alphabetical listing of the students enrolled at the school.

The challenge ahead of you is to import all of the school’s data into a SQLite database, and write a Python program to query that database to get house rosters for each of the houses of Hogwarts.

[**Specification**](https://cs50.harvard.edu/x/2020/psets/7/houses/#specification)

In import.py, write a program that imports data from a CSV spreadsheet.

* Your program should accept the name of a CSV file as a command-line argument.
  + If the incorrect number of command-line arguments are provided, your program should print an error and exit.
  + You may assume that the CSV file will exist, and will have columns name, house, and birth.
* For each student in the CSV file, insert the student into the students table in the students.db database.
  + While the CSV file provided to you has just a name column, the database has separate columns for first, middle, and last names. You’ll thus want to first parse each name and separate it into first, middle, and last names. You may assume that each person’s name field will contain either two space-separated names (a first and last name) or three space-separated names (a first, middle, and last name). For students without a middle name, you should leave their middle name field as NULL in the table.

In roster.py, write a program that prints a list of students for a given house in alphabetical order.

* Your program should accept the name of a house as a command-line argument.
  + If the incorrect number of command-line arguments are provided, your program should print an error and exit.
* Your program should query the students table in the students.db database for all of the students in the specified house.
* Your program should then print out each student’s full name and birth year (formatted as, e.g., Harry James Potter, born 1980 or Luna Lovegood, born 1981).
  + Each student should be printed on their own line.
  + Students should be ordered by last name. For students with the same last name, they should be ordered by first name.