Discussion 11

Databases

Make sure you installed DB Browser for SQLite

Tips on working with databases

READ READ READ READ READ READ the documentation. Lots of nuance and little details.
Documentation:
https://www.sqlite.org/docs.html
Good examples:
https://www.w3schools.com/sql/default.asp

Useful things to read: SELECT, WHERE, JOIN, INSERT, IF NOT EXISTS, how to insert values into tables, seeing what gets returned by a Select call, etc.

Dropping

DROP TABLE IF EXISTS *Tablename* will **completely wipe out** your table. Not useful if you want to keep adding to your table every time your code runs (hint hint final project).

Instead, you can use:

CREATE TABLE IF NOT EXISTS *Tablename*

This will create a table if one doesn’t exist, but will not override an existing one.

Tables with two different keys

Sometimes you may want to have two tables linked by a shared key.

E.g., people and their favourite movie

Name	Favourite Movie
Jane	0
Jack	0
Jenny	3
Jacob	1
Josefina	2

id	Title	Release Date	Genre
0	Exam	2009	Horror
1	Pulp Fiction	1994	Crime
2	Parasite	2019	Comedy / Thriller
3	Celda 211	2009	Prison
4	Monsters Inc	2001	Animation

Your Task: Animal Hospital!

You have been hired by the great, but not-tech-savvy, local animal hospital! You will help them with a series of tasks.

Task 1: Create the database and add in Fluffle

1. Create a new table in the database with the following fields:

   Pet id, name (string), species_id (number), age (integer), cuteness (integer), aggressiveness (number)

1. Populate it with the following entry:

   Name = Fluffle, species = “Rabbit”, age = 3, cuteness = 90, aggressiveness = 100

(You will find what species_id to enter for Fluffle by looking at the table “Species”, which we create for you. Read code / animal_hospital.db for details)

Task 1 Output

pet_id	name	species_id	age	cuteness	aggressivity
Filter	Filter	Filter	Filter	Filter	Filter
0	Fluffle	0	3	90	100

Task 2: Merge the hospitals!

Your animal hospital and another one are merging together. Import their patient list into your table! They provided their patient list in a JSON file.

Task: Read in the JSON file and add the pets to your database.

You will need to search the Species table to know what id to put in for each species.

Task 3: Help the intern!

You have a new intern who is ready to work on curing animals. However, they are new to the job and do not know how to handle aggressive pets. Return a list of patients they can help with.

Task: Filter the database for pets whose aggressiveness is 10 or lower. Return a list of tuples, the tuples consisting of the name of the animals who are the least aggressive.

Task 4 (Optional): For tax reasons, report the number of animals you have!

Count number of each species in the database and print this out in the terminal. Example output will be:

“Rabbit = 1

Hare = 1

Dog = 2

etc.”