PopQuiz 2 - `SELECT`, `WHERE`, `ORDER BY`, and `LIMIT`

Database Theory & Applications CSC 330 (Spring 2024)

README

Please execute all queries against the provided `test.db` database. You can download the SQLite file from here: tinyurl.com/foods-sql.

Check that `test.db` in the current working directory contains the tables in `foods.sql`, `episodes`, `food_types`, `foods`, and `foods_episodes`.

```
.tables
```

```
episodes        food_types      foods           foods_episodes
```

Task 1: Basic WHERE clause usage

```
-- Find all foods with an id of 5. Print id and name.
SELECT id, name FROM foods WHERE id=5;
```

Task 2: Using logical operators in WHERE clause

```
-- Select all foods whose type_id is either 1 (Bakery) or 2 (Cereal).
-- Print the name as "Name" and the type_id as "Type"
SELECT name AS "Name", type_id AS "Type"
FROM foods WHERE type_ID=1 OR type_ID=2;
```

Task 3: String manipulation with WHERE clause

```
-- Find foods whose names start with 'Ch' and contain 'e'.
-- Print the names with the header:
--       Names that start with 'Ch' and contain 'e'
SELECT name AS "Names that start with 'Ch' and contain 'e'"
FROM foods WHERE name LIKE "Ch%e%";
```

Task 4: Using arithmetic expressions in SELECT

```
-- Find the name of each food along with its name length.
-- Display only the records 10 to 20
-- Show ‘Name‘ and ‘Length of Name‘ in the header
SELECT name as NAME, LENGTH(name) AS "Length of Name"
FROM foods LIMIT 10,10;
```

Task 5: Employing the IN operator

```
-- Select all foods whose type_id is either 3 (Chicken/Fowl) or 4
-- (Dairy) using the IN operator. Show only 10 rows, no offset.
SELECT name AS NAME, type_id AS TYPE FROM foods
WHERE type_id IN (3,4);
```

Task 6: Combining string functions and WHERE clause

```
-- Select foods whose names end with ’s’ and are at least 20 characters
-- long. Pick an appropriate header name.
SELECT name AS ’NAME LENGTH at least 5, ends in ‘s‘’ FROM foods
WHERE LENGTH(name) >= 20 AND (name LIKE ’%s’);
```

Task 7: Exploring the ORDER BY clause

```
-- List all foods in ascending order of their type_id, and descending
-- order of their names. Display only 10 rows starting after 100 rows.
SELECT type_id AS TYPE, name AS NAME FROM foods
ORDER BY type_id ASC, name DESC LIMIT 10 OFFSET 100;
```

Task 8: Using aggregate functions

```
-- Find the average length of food names.
SELECT AVG(LENGTH(name)) AS ’Average length
of name’ FROM foods;
```

Task 9: The GLOB pattern matching

```
-- Select foods whose names start with 'P' and contain an 'e'.
-- Use the GLOB operator instead of LIKE and * instead of %
--  SELECT name AS "NAME starts with P and contains e" FROM foods
--         WHERE name GLOB 'P*e*';
SELECT name AS "NAME starts with P and contains e" FROM foods
WHERE name GLOB 'P*e*';
```

Task 10: Complex WHERE with logical and relational operators

```
-- Find all foods whose name starts with B and are not of type_id 2
-- (Cereal) or type_id 1 (Bakery).
SELECT name AS NAME, type_id AS TYPE FROM foods
        WHERE name LIKE 'B%' AND type_id NOT IN (1,2);
SELECT name AS NAME, type_id AS TYPE FROM foods
       WHERE name LIKE 'B%' AND (type_id != 1 AND type_id != 2);
```