

SQL Exercise

Sakila

Use the Sakila database on maria - ssh to: mariadb.ict.op.ac.nz, then connect to the database: `mysql -u username -p`

1. What is the average replacement cost of a film?
2. Design a query to list the titles of each film and its language_id.
3. List the staff at each store (First name, last name, store number)
4. Design a query to show the number of films in each language.

SQLite

Set up a sqlite database

These instructions will take you through the process to create and use a sqlite database from a supplied .sql file. The short-cut instructions are a - e, more detailed 1-6 :

- a) Get a copy of the SQLite shell
 - b) Extract it into a directory that you're going to work from or one that's in your shell
 - c) Get a copy of the sql script file for the database
 - d) Start SQLite, creating the database file
 - e) Load the .sql file into the database
1. Create a local folder on your computer called sqlite
 2. Download a copy of SQLite either from I: or sqlite.org and extract it into your sqlite folder
The executable should be sqlite3.exe
 3. Download the pizza database from the I: drive or Teams
 4. Open a command prompt and issue these commands (H: is whatever the name of your drive). The final command should show both sqlite3.exe and pizza.sql in the directory
 - H:
 - cd h:\sqlite
 - dir

We will work with a database called “pizza” which has the following schema

Person (name, age, gender)	name is a key
Frequents (name, pizzeria)	(name, pizzeria) is a joint key
Eats (name, pizza)	(name, pizza) is a joint key
Serves (pizzeria, pizza, price)	(pizzeria, pizza) is a joint key

5. Now create the sqlite database file with:

➤ `sqlite3 pizza.sqlite`

6. SQLite, like all DBMS's, has a number of DBMS specific commands to manage and manipulate databases. Most SQLite specific commands start with a dot.

Load and confirm the database

- a) Create the database from the script
- b) Check that the tables exist
- c) Check the schema of the tables: person, frequents, eats and serves
- d) Confirm that data is in the tables

➤ `.read pizza.sql`
➤ `.tables`
➤ `.schema Person`
➤ `.schema Frequents`
➤ `.schema Eats`
➤ `.schema Serves`
➤ `Select * from Person;`
➤ `Select * from Frequents;`
➤ `Select * from Eats;`
➤ `Select * from Serves;`
➤ `.quit`
➤ `dir`

This last command should show that you now have a new file `pizza.sqlite`. This is the sqlite database file. To access it you issue the command

➤ `sqlite3 pizza.sqlite`

Single Table Exercises using Pizza

Working with this database. Note down the query in the space provided. The schema is:

Person (name, age, gender)	name is a key
Frequents (name, pizzeria)	(name, pizzeria) is a joint key
Eats (name, pizza)	(name, pizza) is a joint key
Serves (pizzeria, pizza, price)	(pizzeria, pizza) is a joint key

I recommend you turn headers on and set the column mode for readability, type .help for system commands

- .header on
- .mode column

1. Find all the places that serve pepperoni

```
pizzeria
Pizza Hut
Little Caesars
Straw Hat
New York Pizza
```

sqlite> Select pizzeria from Serves Where pizza='pepperoni';

2. Display all people sorted by age:

```
name|age|gender
Dan|13|male
Amy|16|female
Ian|18|male
Ben|21|male
Fay|21|female
Gus|24|male
Hil|26|female
```

sqlite> select * from Person order by age;

3. Display a list of who eats which pizza by listing the pizza first then the person's name. Sort this list by the pizza names then the person's name

```
pizza|name
cheese|Ben
cheese|Dan
cheese|Eli
cheese|Gus
cheese|Hil
mushroom|Amy
mushroom|Dan
mushroom|Fay
mushroom|Gus
pepperoni|Amy
pepperoni|Ben
```

sqlite> select pizza, name from Eats order by pizza, name;

4. Display the prices at the various pizza's by listing the pizzeria, the pizza and the price. Sort by Pizzeria, Price then Pizza

```
pizzeria|pizza|price
Chicago Pizza|cheese|7.75
Chicago Pizza|supreme|8.5
Dominos|cheese|9.75
Dominos|mushroom|11
Little Caesars|cheese|7
Little Caesars|mushroom|9.25
Little Caesars|sausage|9.5
Little Caesars|pepperoni|9.75
New York Pizza|cheese|7
New York Pizza|pepperoni|8
```

select pizzeria, pizza, price from Serves order by pizzeria, price, pizza;

5. What is the average price of a pizza for each pizzeria?

```
pizzeria|avg(price)
Chicago Pizza|8.125
Dominos|10.375
Little Caesars|8.875
New York Pizza|7.83333333
Pizza Hut|11.25
Straw Hat|9.0
```

```
sqlite> select pizzeria, AVG(price) AS avg_price from Serves
...> group by pizzeria
...> order by pizzeria;
```

6. How many people eat cheese pizzas?

```
Number of Cheese Eaters
5
```

```
select COUNT(*) from Eats where pizza='cheese';
```

7. List all of the pizzeria's with "Pizza" in their name.

```
pizzeria
Chicago Pizza
New York Pizza
Pizza Hut
```

```
sqlite> select distinct pizzeria from Serves where pizzeria LIKE '%pizza'
```

- Tip: To remove duplicate records use the keyword "distinct" after select.
- e.g.: select distinct name from eats;

8. Find all of the females under 20

```
name|age
Amy|16
```

```
sqlite> select * from Person where gender='female' AND age<20;
```

9. Find all male customers in their 20's showing their name and age, sorted by name.

```
name|age
Ben|21
Gus|24
```

```
sqlite> select name, age from Person where gender ='male' AND age BETWEEN 20 AND 29;
```

10. Get the maximum, minimum and average price of pizza's in each pizzeria, sort by pizzeria – note the labels in this output

```
pizzeria|MAX|MIN|AVERAGE
Chicago Pizza|8.5|7.75|8.125
Dominos|11|9.75|10.375
Little Caesars|9.75|7|8.875
New York Pizza|8.5|7|7.833333
Pizza Hut|12|9|11.25
Straw Hat|9.75|8|9.0
```

```
select pizzeria, MAX(price), MIN(price), AVG(price) from Serves group by pizzeria order by pizzeria;
```