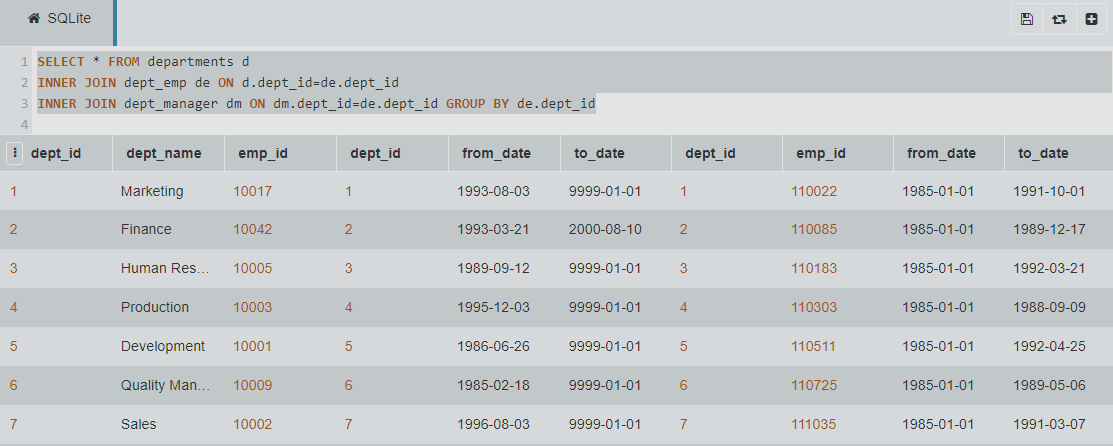
**Part 1**

**1. List all attributes present in the departments relation.**

SELECT \* from departments d

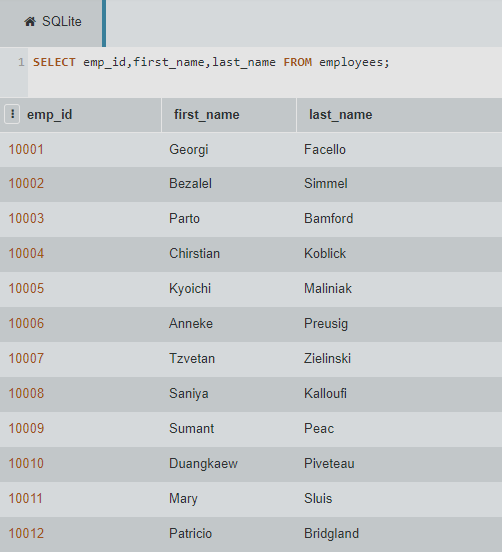
INNER JOIN dept\_emp de on d.dept\_id=de.dept\_id

INNER JOIN dept\_manager dm on dm.dept\_id=de.dept\_id group by de.dept\_id

****

**2. List all employee numbers of all past/current employees, their first and last names**

SELECT emp\_id,first\_name,last\_name from employees;



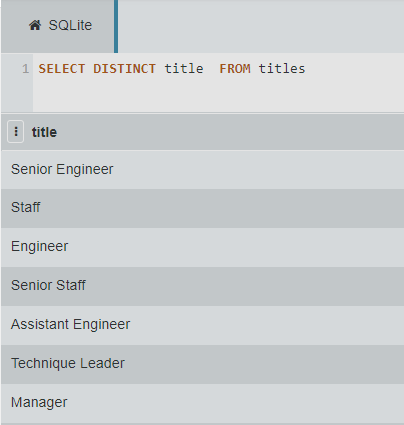
**3. List all job titles held by any past/current employees.**

SELECT \* from titles;



**4. List all unique job titles found in the database.**

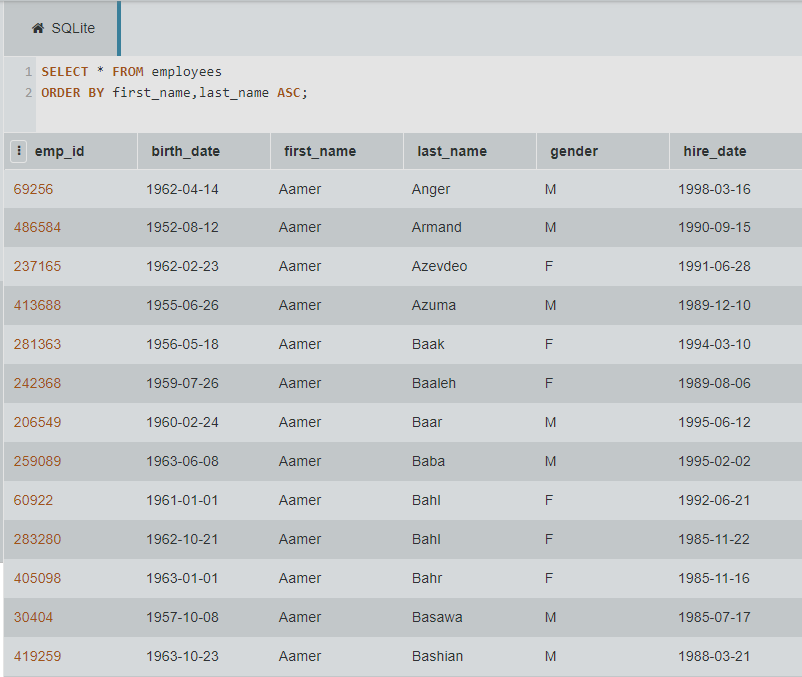
SELECT DISTINCT title from titles;

****

**5. List all past/current employees’ names ordered alphabetically in ascending order, i.e. first name and last name in alphabetical order.**

SELECT \* from employees

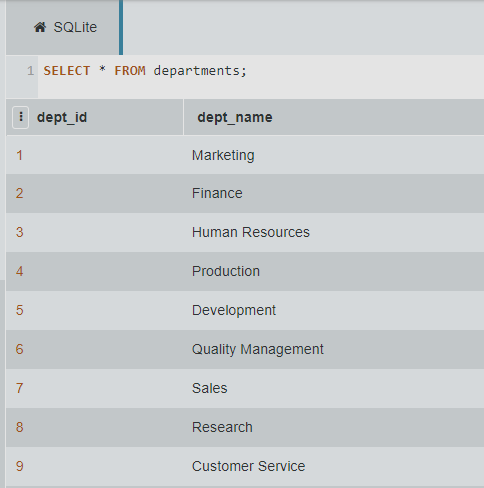
ORDER by first\_name, last\_name asc;



**Part 2**

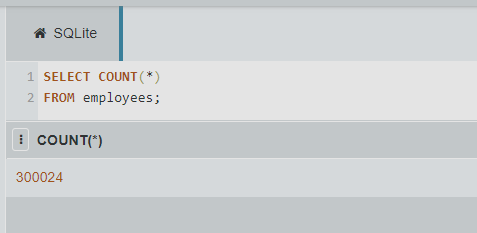
**1. List all department titles present in the database.**

select \* from departments

****

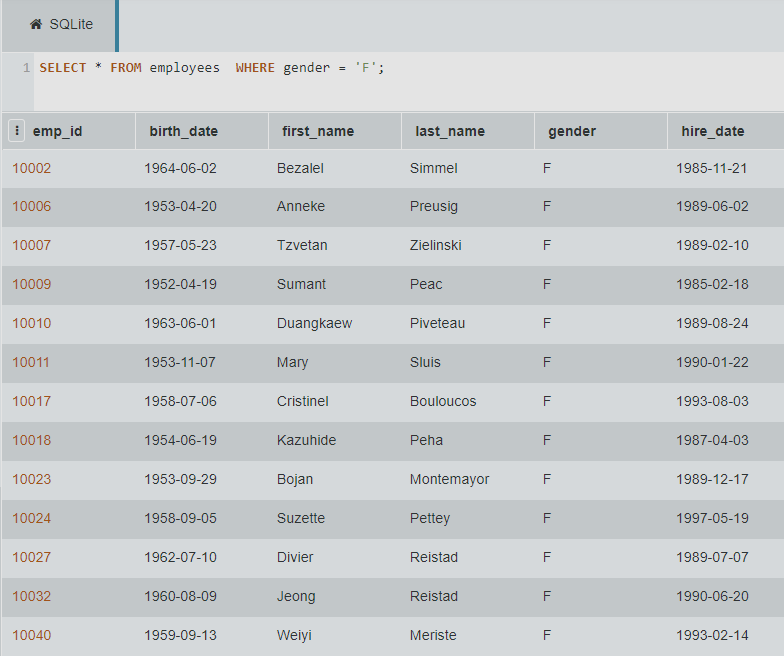
**2. List the total number of all employees (past/current).**

select COUNT(\*) from employees;

****

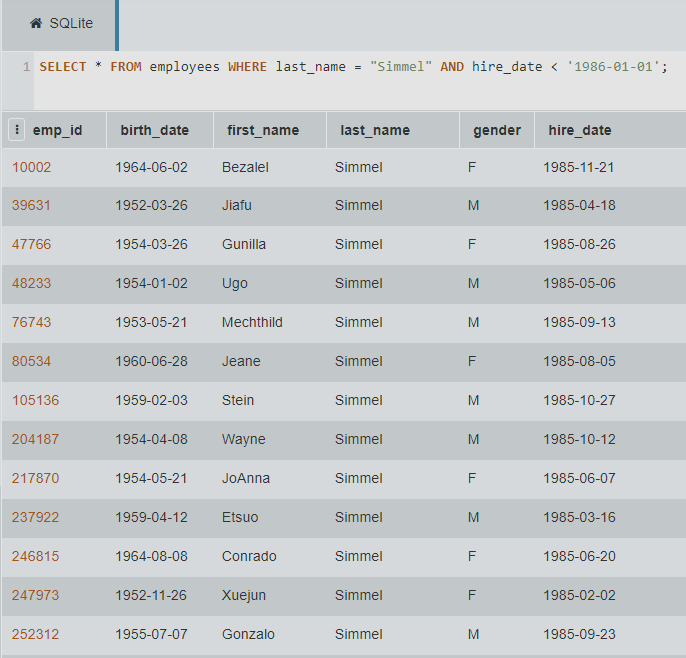
**3. List female employees (past/current) together with all other relation attributes.**

select \* from employees WHERE gender = 'F'

****

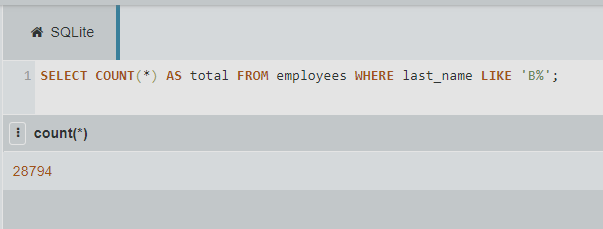
**4. List past/current employees hired prior to 1986-01-01 with the surname Simmel**

select \* from employees WHERE last\_name = 'Simmel' and hire\_date < '1986-01-01'

****

**5. How many past/current employees’ last name begins with the capital letter B? Use a column alias total with B to output your results.**

select count(\*) as total from employees WHERE last\_name like 'B%'

****

**6. Create a new table called emp\_training with 3 columns:**

**• trainer\_no: this should be the primary key and is of type integer and needs to implemented as an auto-increment.**

**• first\_name: this data type is varchar(30) and should not be NULL**

**• last\_name: this data type is varchar(30) and should not be NULL**

**• t\_module: this data type is varchar(20)**

CREATE table emp\_training (

trainer\_no int AUTO\_INCREMENT,

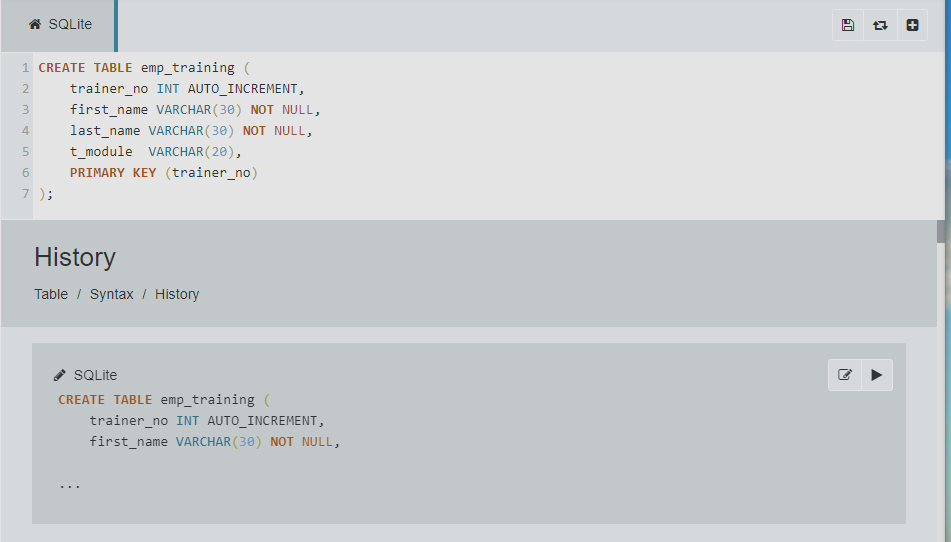
first\_name varchar(30) not null,

last\_name varchar(30) not null,

t\_module varchar(20),

PRIMARY key (trainer\_no)

);

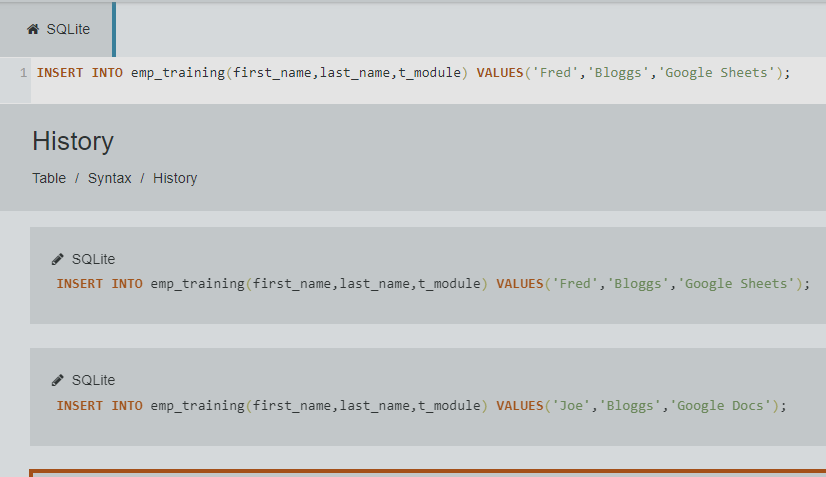
****

**7. Insert 2 new rows into the emp\_training table:**

**Row 1: fname: Joe lname: Bloggs module: Google Docs**

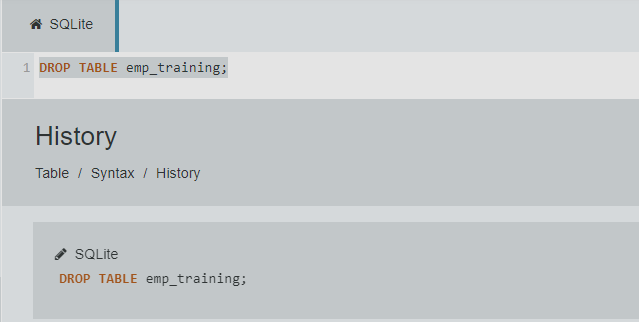
**Row 2: fname: Fred lname: Bloggs module: Google Sheets**

INSERT INTO emp\_training(first\_name, last\_name,t\_module) values('Fred','Bloggs','Google Sheets');

****

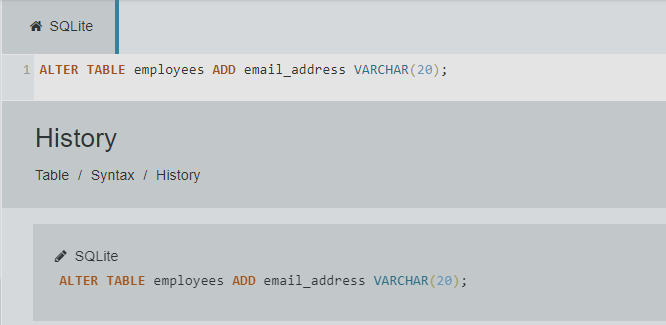
**8. The organisation no longer wishes to record the employees training within the database. Therefore, delete the newly created emp\_training table.**

drop TABLE emp\_training

****

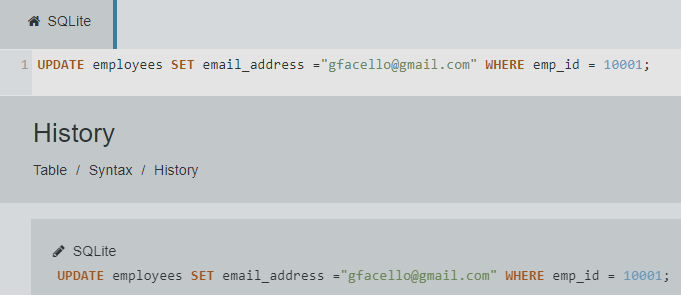
**9. Alter the employees table to include an email\_address field with a data type of varchar(20).**

alter TABLE employees add email\_address varchar(20)

****

**10. Update the email address of Georgi Facello to gfacello@gmail.com, where emp\_no equals to 10001.**

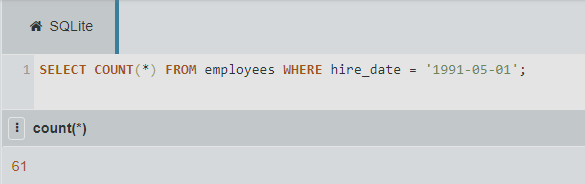
UPDATE employees set email\_address ='gfacello@gmail.com' WHERE emp\_id = 10001

****

**Part 3**

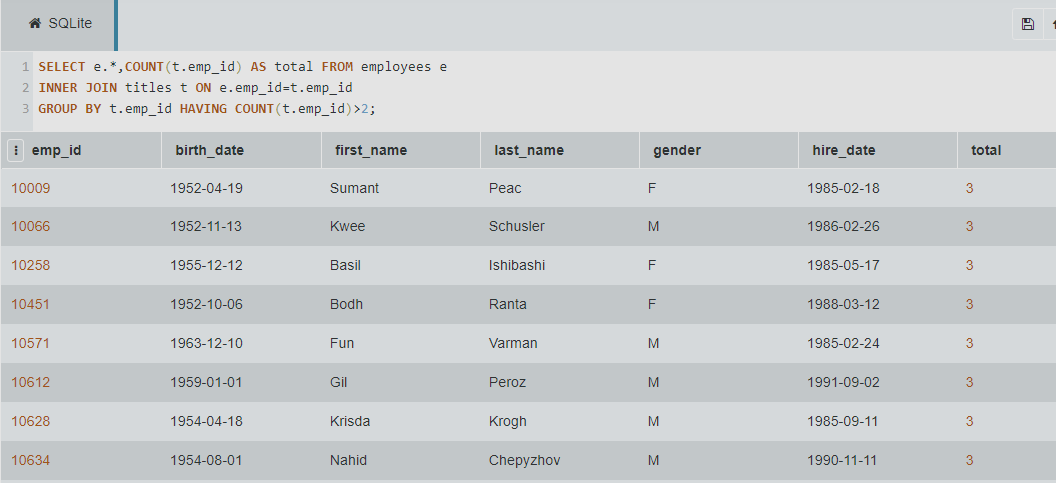
**1. The number of all employees that started on 1991-05-01.**

SELECT Count(\*) from employees WHERE hire\_date = '1991-05-01';

****

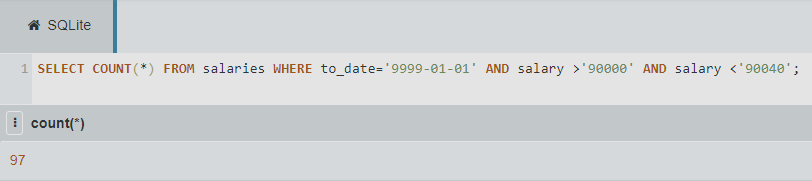
**2. List all emp\_no who have had strictly more than 2 titles and display the total number of the titles they have had.**

SELECT e.\*,count(t.emp\_id) as total from employees e INNER JOIN titles t on e.emp\_id=t.emp\_id GROUP by t.emp\_id HAVING count(t.emp\_id)>2;



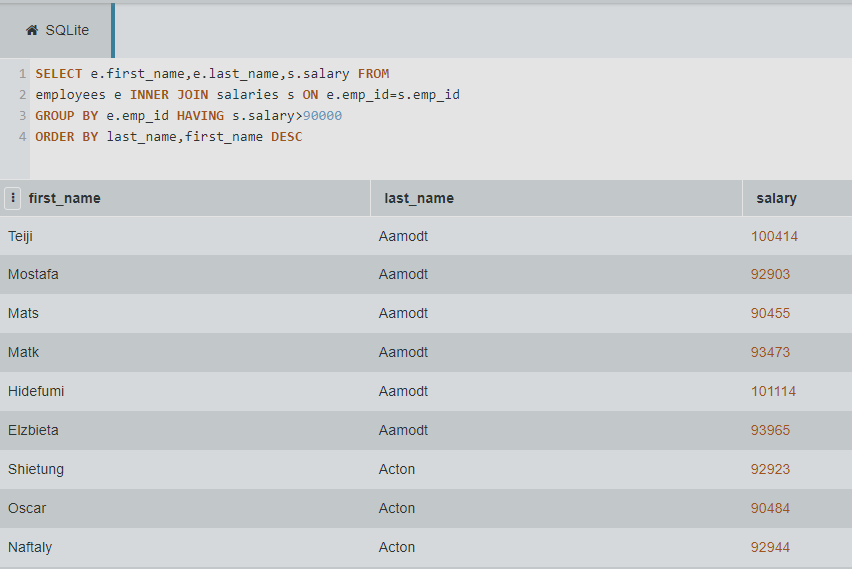
**3. The number of employees that have a current salary (i.e. to\_date equals to 9999-01-01) between 90000 and 90040**

SELECT Count(\*) from salaries WHERE to\_date ='9999-01-01' and salary >'90000' and salary < '90040'

****

**4. List all unique employees’ last and first names (using GROUP BY method) that have a current salary (i.e. to\_date equals to 9999-01-01) greater than 90000, outputting both names in descending order (sort by the last name first and then the first name) and also displaying their current salaries (using the INNER JOIN method).**

SELECT e.first\_name,e.last\_name,s.salary from employees e INNER JOIN salaries s on e.emp\_id=s.emp\_id GROUP by e.emp\_id HAVING s.salary>90000 ORDER by last\_name,first\_name desc

****

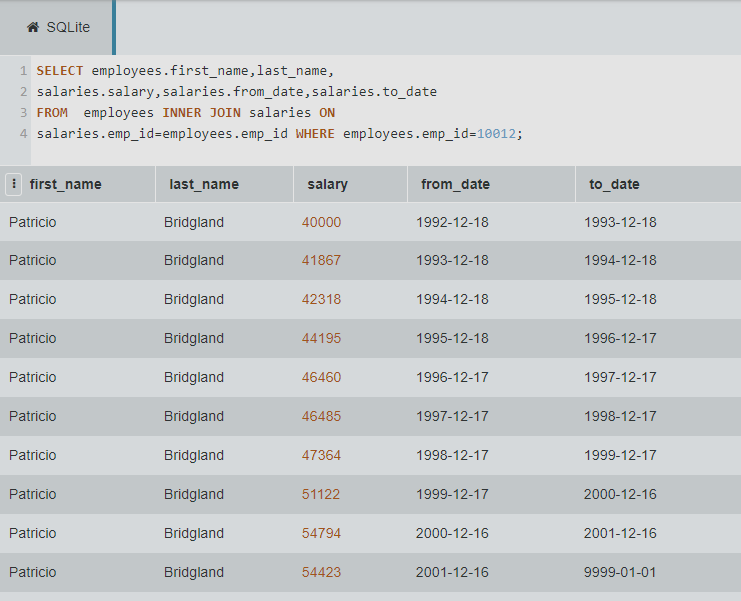
**5. First name, last name, all salary dates and related amounts for the employee with employee number 10012.**

SELECT employees.first\_name,last\_name,

salaries.salary,salaries.from\_date,salaries.to\_date

from employees INNER join salaries on

salaries.emp\_id=employees.emp\_id WHERE employees.emp\_id=10012;

****

**6. In relation to the table named salaries in Figure 1 above:**

**a. What is the degree of this table?**

degree 4

**b. What column(s), if any, make(s) up the primary key?**

(emp\_id, from\_date)

**c. What column(s), if any, make(s) up the foreign key?**

(emp\_id)

**7. In the given schema, the tables dept\_emp, dept\_manager, salaries, titles have composite keys. Explain for each relation why this is the case?**

In the tables dept emp, dept manager, salaries, titles, compound keys were used because it is not possible to use a single column of a table to uniquely identify their records.