Part 1:

1. SELECT \* FROM departments as d join dept\_manager as dm ON d.dept\_id = dm.dept\_id JOIN dept\_emp as de on d.dept\_id=de.dept\_id
2. SELECT emp\_id,first\_name,last\_name FROM employees
3. SELECT dept\_name FROM departments
4. SELECT DISTINCT title FROM titles ORDER BY title ASC
5. SELECT first\_name,last\_name FROM employees ORDER by first\_name ASC,last\_name ASC

Part 2:

1. SELECT COUNT(\*) FROM employees WHERE hire\_date="1991-05-01"
2. SELECT e.emp\_id, count(e.emp\_id) FROM employees as e JOIN titles as t ON e.emp\_id=t.emp\_id GROUP BY e.emp\_id HAVING COUNT(e.emp\_id) > 2
3. SELECT \* FROM employees as e JOIN titles as t ON e.emp\_id=t.emp\_id JOIN dept\_emp as de on de.emp\_id=e.emp\_id JOIN dept\_manager as dm on dm.emp\_id=e.emp\_id JOIN departments as d ON d.dept\_id=de.dept\_id OR d.dept\_id=dm.dept\_id JOIN salaries as s on s.emp\_id=e.emp\_id WHERE gender="F"
4. SELECT \* from employees WHERE last\_name="Simmel" and hire\_date < "1986-01-01"
5. SELECT COUNT(\*) as "total with B" from employees WHERE last\_name LIKE "B%"
6. Employee Training Table

DROP TABLE IF EXISTS emp\_training;

CREATE TABLE IF NOT EXISTS emp\_training(

trainer\_no INTEGER PRIMARY KEY AUTOINCREMENT,

first\_name TEXT NOT NULL,

last\_name TEXT NOT NULL,

t\_module TEXT

);

1. Insert Data

INSERT INTO emp\_training (first\_name,last\_name,t\_module) VALUES ('Joe', 'Bloggs', 'Google Docs');

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

1. DROP TABLE IF EXISTS emp\_training
2. ALTER TABLE employees ADD COLUMN email\_address varchar(20);
3. UPDATE employees SET email\_address="gfacello@gmail.com" WHERE emp\_id=10001;

Part 3:

1. SELECT d.dept\_id, e.gender, COUNT(e.emp\_id) as num\_empGender from departments as d JOIN dept\_manager as dm on d.dept\_id=dm.dept\_id JOIN employees as e on e.emp\_id=dm.emp\_id GROUP by d.dept\_id, e.gender ORDER by d.dept\_id ASC
2. SELECT e.gender, AVG(s.salary) as avg\_salary, t.title from employees as e JOIN titles as t on e.emp\_id=t.emp\_id JOIN salaries as s on e.emp\_id=s.emp\_id where t.title="Technique Leader" GROUP by gender
3. SELECT count(\*) As current\_employees from employees as e JOIN salaries as s on e.emp\_id=s.emp\_id WHERE CURRENT\_DATE < s.to\_date and s.salary > 90000 and s.salary < 90040
4. SELECT e.last\_name, e.first\_name, s.salary from employees as e INNER JOIN salaries as s on e.emp\_id=s.emp\_id WHERE CURRENT\_DATE < s.to\_date and s.salary > 90000 GROUP by e.last\_name, e.first\_name ORDER by e.last\_name DESC, e.first\_name DESC
5. SELECT e.first\_name, e.last\_name, s.from\_date, s.to\_date, s.salary from employees as e JOIN salaries as s on e.emp\_id=s.emp\_id where e.emp\_id=10012
6. Multiple Questions Answer
   1. What is the degree of this table?
      1. For the salaries table given above, the degree is 4. That is there are 4 attributes (columns/fields) in this table.
   2. What column(s), if any, make(s) up the primary key?
      1. There is no primary key.
   3. What column(s), if any, make(s) up the foreign key?
      1. emp\_id is a foreign key for the employees table
7. A composite key is two or more columns in a table that are combined to uniquely identify each row in the table. The columns must be combined to create a unique identifier, but individually they do not guarantee uniqueness.

For dept\_manager and dept\_emp tables: There are emp\_id and dept\_id columns. These columns are likely to repeat the same data, but when combined they form a unique key.

For salaries and titlies tables: In this table, we need to consider the emp\_id and dates. Because the employee may have different duties or salaries on different dates.

https://www.geeksforgeeks.org/composite-key-in-sql/