CODING CHALLENGE-CAREERHUB

Problem Statement:

Create SQL Schema from the application, use the class attributes for table column names.

SQL SCHEMA CREATION:

CREATING TABLES-

```
create database careerhub;
use careerhub;
create table if not exists companies (
  CompanyID int primary key,
  CompanyName varchar(100),
  Location varchar(100)
);
create table if not exists jobs (
  JobID int primary key,
  CompanyID int,
  JobTitle varchar(100),
  JobDescription text,
  JobLocation varchar(100),
  Salary decimal(10,2),
  JobType varchar(50),
  PostedDate datetime,
  foreign key (CompanyID) references companies(CompanyID)
);
create table if not exists applicants (
  ApplicantID int primary key,
  FirstName varchar(100),
  LastName varchar(100),
  Email varchar(100),
  Phone varchar(20),
  Resume text
);
create table if not exists applications (
  ApplicationID int primary key,
  JobID int,
  ApplicantID int,
  ApplicationDate datetime,
  CoverLetter text,
  foreign key (JobID) references jobs(JobID),
  foreign key (ApplicantID) references applicants(ApplicantID)
);
```

INSERTING VALUES-

```
insert into Companies
values
(1, 'Hexaware', 'Chennai'),
(2, 'Google', 'Hyderabad'),
(3, 'Microsoft', 'Bangalore'),
(4, 'TCS', 'Mumbai'),
(5, 'Tech Mahindra', 'Pune'),
(6, 'Kaar', 'Chennai');
insert into Jobs
values
```

```
(101, 1, 'Software Developer', 'Backend developer role', 'Chennai', 70000, 'Full-time', '2024-06-01 10:00:00'),

(102, 2, 'Data Analyst', 'Analyze data and reports', 'Hyderabad', 85000, 'Full-time', '2024-06-03 11:00:00'),

(103, 3, 'Web Developer', 'Frontend development', 'Bangalore', 65000, 'Part-time', '2024-06-05 09:30:00'),

(104, 4, 'System Engineer', 'Maintain IT systems', 'Mumbai', 80000, 'Full-time', '2024-06-07 12:00:00'),

(105, 5, 'QA Engineer', 'Test automation', 'Pune', 60000, 'Contract', '2024-06-10 15:45:00'),

(106, 6, 'Cloud Engineer', 'Manage cloud infra', 'Chennai', 90000, 'Full-time', '2024-06-12 14:00:00');

insert into Applicants

values

(201, 'Rishitha', 'Vegesna', 'rishi@gmail.com', '9876543210', 'Experienced in Python and SQL'),

(202, 'Iswarya', 'K', 'iswarya@gmail.com', '9876543211', 'Java and Web Development'),

(203, 'Savitri', 'M', 'savitri@gmail.com', '9876543212', 'Data Analytics and Excel'),

(204, 'Rukhmini', 'S', 'rukhmini@gmail.com', '9876543213', 'C++ and Software Testing'),

(205, 'Bala', 'R', 'bala@gmail.com', '9876543214', 'Cloud and DevOps'),

(206, 'Sravya', 'T', 'sravya@gmail.com', '9876543215', 'UI/UX Design'),

(207, 'Charitha', 'V', 'charitha@gmail.com', '9876543216', 'AI and Machine Learning'),

(208, 'Harishini', 'D', 'harishini@gmail.com', '9876543217', 'System Security');

insert into Applications

values

(301, 101, 201, '2024-06-14 10:00:00', 'I am excited to join your company.'),
```

(302, 102, 202, '2024-06-15 11:30:00', 'I love working with data.'),

(303, 103, 203, '2024-06-15 12:00:00', 'I am passionate about frontend.'),

(304, 104, 204, '2024-06-16 09:00:00', 'Experienced in system monitoring.'),

(305, 101, 205, '2024-06-16 10:45:00', 'Looking forward to backend role.'),

(306, 106, 206, '2024-06-17 13:30:00', 'Skilled in cloud services.'),

(307, 102, 207, '2024-06-17 14:15:00', 'Strong interest in analytics.'),

(308, 105, 208, '2024-06-18 15:00:00', 'Keen on software testing.'),

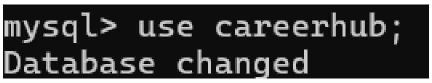
(309, 104, 203, '2024-06-18 16:30:00', 'System Engineer is my goal.');

Tasks:

1.Provide a SQL script that initializes the database for the Job Board scenario "CareerHub".

Create database careerhub;

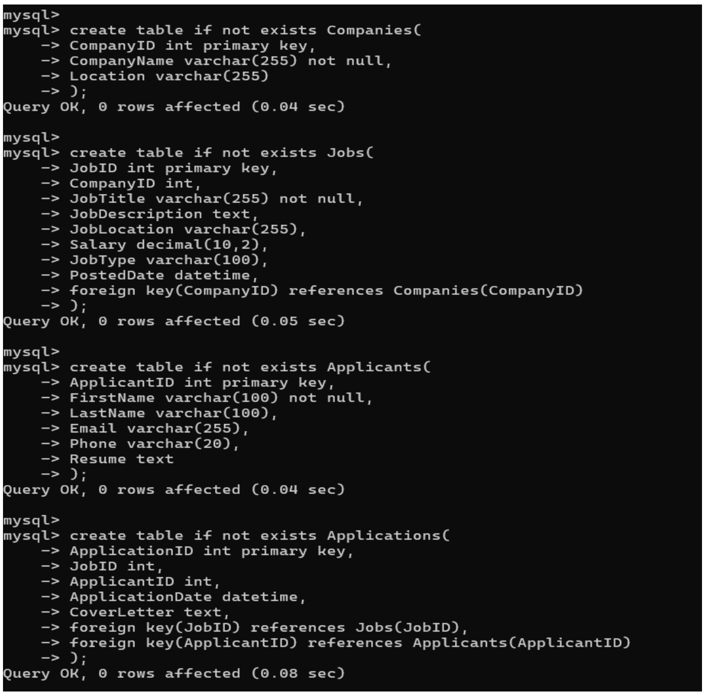
Use careerhub;



```
mysql> use careerhub;
Database changed
```

2. Create tables for Companies, Jobs, Applicants and Applications.

The CREATE TABLE scripts were already provided in the previous section under SQL Schema Creation.

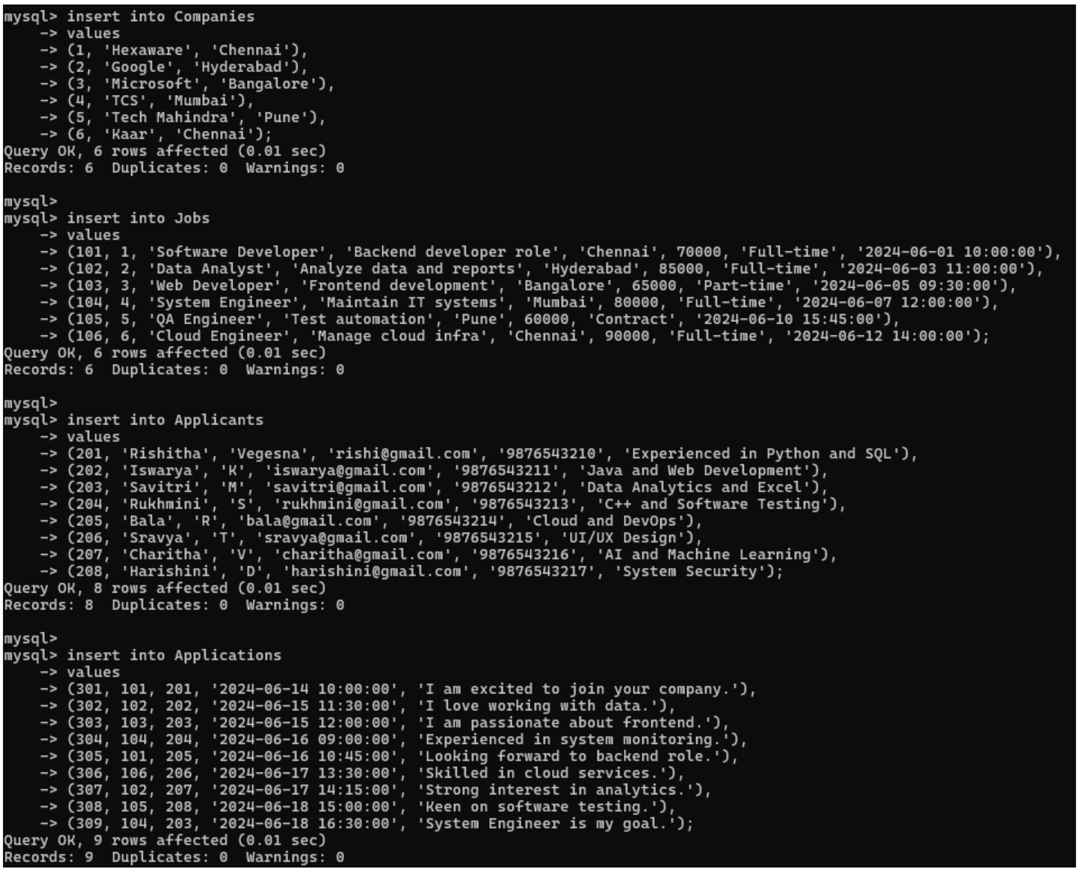


```
mysql>
mysql> create table if not exists Companies(
    -> CompanyID int primary key,
    -> CompanyName varchar(255) not null,
    -> Location varchar(255)
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql>
mysql> create table if not exists Jobs(
    -> JobID int primary key,
    -> CompanyID int,
    -> JobTitle varchar(255) not null,
    -> JobDescription text,
    -> JobLocation varchar(255),
    -> Salary decimal(10,2),
    -> JobType varchar(100),
    -> PostedDate datetime,
    -> foreign key(CompanyID) references Companies(CompanyID)
    -> );
Query OK, 0 rows affected (0.05 sec)

mysql>
mysql> create table if not exists Applicants(
    -> ApplicantID int primary key,
    -> FirstName varchar(100) not null,
    -> LastName varchar(100),
    -> Email varchar(255),
    -> Phone varchar(20),
    -> Resume text
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql>
mysql> create table if not exists Applications(
    -> ApplicationID int primary key,
    -> JobID int,
    -> ApplicantID int,
    -> ApplicationDate datetime,
    -> CoverLetter text,
    -> foreign key(JobID) references Jobs(JobID),
    -> foreign key(ApplicantID) references Applicants(ApplicantID)
    -> );
Query OK, 0 rows affected (0.08 sec)
```



```
mysql> insert into Companies
    -> values
    -> (1, 'Hexaware', 'Chennai'),
    -> (2, 'Google', 'Hyderabad'),
    -> (3, 'Microsoft', 'Bangalore'),
    -> (4, 'TCS', 'Mumbai'),
    -> (5, 'Tech Mahindra', 'Pune'),
    -> (6, 'Kaar', 'Chennai');
Query OK, 6 rows affected (0.01 sec)
Records: 6  Duplicates: 0  Warnings: 0

mysql>
mysql> insert into Jobs
    -> values
    -> (101, 1, 'Software Developer', 'Backend developer role', 'Chennai', 70000, 'Full-time', '2024-06-01 10:00:00'),
    -> (102, 2, 'Data Analyst', 'Analyze data and reports', 'Hyderabad', 85000, 'Full-time', '2024-06-03 11:00:00'),
    -> (103, 3, 'Web Developer', 'Frontend development', 'Bangalore', 65000, 'Part-time', '2024-06-05 09:30:00'),
    -> (104, 4, 'System Engineer', 'Maintain IT systems', 'Mumbai', 80000, 'Full-time', '2024-06-07 12:00:00'),
    -> (105, 5, 'QA Engineer', 'Test automation', 'Pune', 60000, 'Contract', '2024-06-10 15:45:00'),
    -> (106, 6, 'Cloud Engineer', 'Manage cloud infra', 'Chennai', 90000, 'Full-time', '2024-06-12 14:00:00');
Query OK, 6 rows affected (0.01 sec)
Records: 6  Duplicates: 0  Warnings: 0

mysql>
mysql> insert into Applicants
    -> values
    -> (201, 'Rishitha', 'Vegesna', 'rishi@gmail.com', '9876543210', 'Experienced in Python and SQL'),
    -> (202, 'Iswarya', 'K', 'iswarya@gmail.com', '9876543211', 'Java and Web Development'),
    -> (203, 'Savitri', 'M', 'savitri@gmail.com', '9876543212', 'Data Analytics and Excel'),
    -> (204, 'Rukhmini', 'S', 'rukhmini@gmail.com', '9876543213', 'C++ and Software Testing'),
    -> (205, 'Bala', 'R', 'bala@gmail.com', '9876543214', 'Cloud and DevOps'),
    -> (206, 'Sravya', 'T', 'sravya@gmail.com', '9876543215', 'UI/UX Design'),
    -> (207, 'Charitha', 'V', 'charitha@gmail.com', '9876543216', 'AI and Machine Learning'),
    -> (208, 'Harishini', 'D', 'harishini@gmail.com', '9876543217', 'System Security');
Query OK, 8 rows affected (0.01 sec)
Records: 8  Duplicates: 0  Warnings: 0

mysql>
mysql> insert into Applications
    -> values
    -> (301, 101, 201, '2024-06-14 10:00:00', 'I am excited to join your company.'),
    -> (302, 102, 202, '2024-06-15 11:30:00', 'I love working with data.'),
    -> (303, 103, 203, '2024-06-15 12:00:00', 'I am passionate about frontend.'),
    -> (304, 104, 204, '2024-06-16 09:00:00', 'Experienced in system monitoring.'),
    -> (305, 101, 205, '2024-06-16 10:45:00', 'Looking forward to backend role.'),
    -> (306, 106, 206, '2024-06-17 13:30:00', 'Skilled in cloud services.'),
    -> (307, 102, 207, '2024-06-17 14:15:00', 'Strong interest in analytics.'),
    -> (308, 105, 208, '2024-06-18 15:00:00', 'Keen on software testing.'),
    -> (309, 104, 203, '2024-06-18 16:30:00', 'System Engineer is my goal.');
Query OK, 9 rows affected (0.01 sec)
Records: 9  Duplicates: 0  Warnings: 0
```

3. Define appropriate primary keys, foreign keys, and constraints.

- companies: companyid as primary key
- jobs: jobid as primary key, and companyid as foreign key referencing companies(companyid)
- applicants: applicantid as primary key
- applications: applicationid as primary key, and

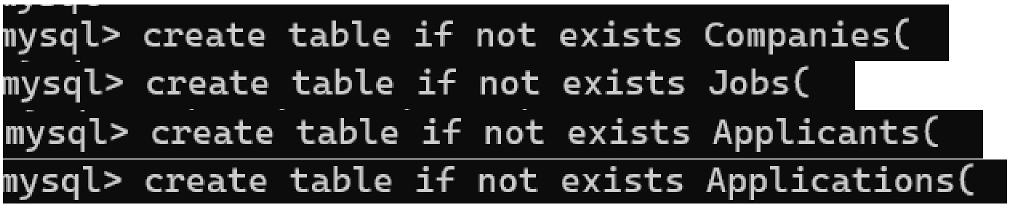
  jobid as foreign key referencing jobs(jobid)

  applicantid as foreign key referencing applicants(applicantid)

4. Ensure the script handles potential errors, such as if the database or tables already exist.

To prevent errors when the script is run multiple times, the

CREATE TABLE statements include IF NOT EXISTS



```
mysql> create table if not exists Companies(
mysql> create table if not exists Jobs(
mysql> create table if not exists Applicants(
mysql> create table if not exists Applications(
```

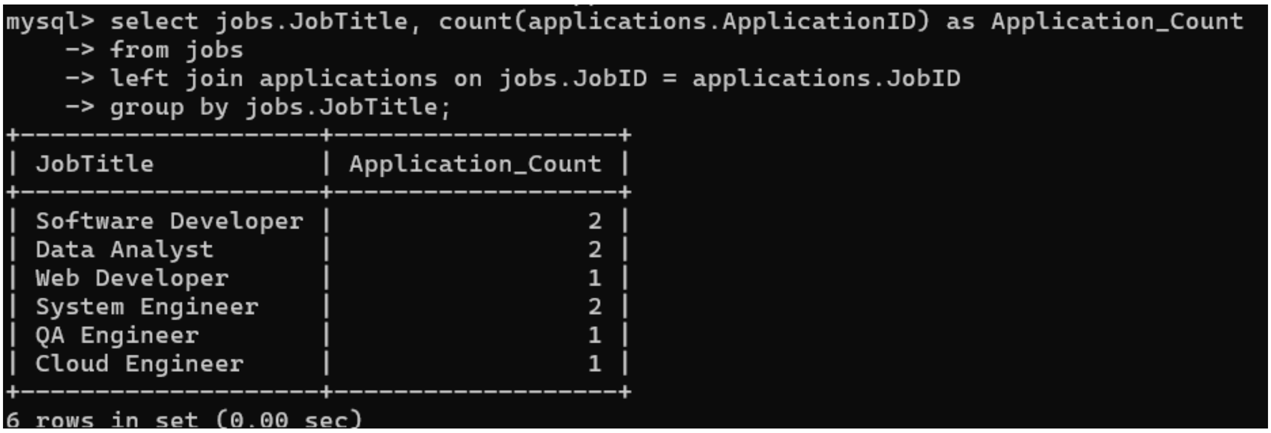
5. Write an SQL query to count the number of applications received for each job listing in the "Jobs" table. Display the job title and the corresponding application count. Ensure that it lists all jobs, even if they have no applications.

select jobs.jobtitle,count(applications.applicationid) as application_count

from jobs

left join applications on jobs.jobid=applications.jobid

group by jobs.jobtitle;



```
mysql> select jobs.JobTitle, count(applications.ApplicationID) as Application_Count
    -> from jobs
    -> left join applications on jobs.JobID = applications.JobID
    -> group by jobs.JobTitle;
+--------------------+-------------------+
| JobTitle           | Application_Count |
+--------------------+-------------------+
| Software Developer |                 2 |
| Data Analyst       |                 2 |
| Web Developer      |                 1 |
| System Engineer    |                 2 |
| QA Engineer        |                 1 |
| Cloud Engineer     |                 1 |
+--------------------+-------------------+
6 rows in set (0.00 sec)
```

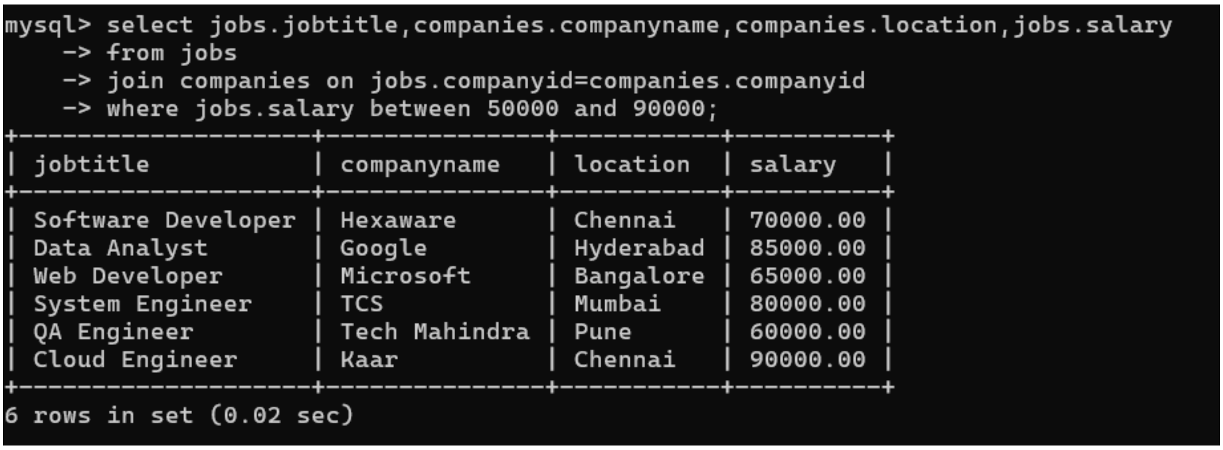
6. Develop an SQL query that retrieves job listings from the "Jobs" table within a specified salary range. Allow parameters for the minimum and maximum salary values. Display the job title, company name, location, and salary for each matching job.

Select jobs.jobtitle,companies.companyname,companies.location,jobs.salary

from jobs

join companies on jobs.companyid=companies.companyid

where jobs.salary between 50000 and 90000;



```
mysql> select jobs.jobtitle,companies.companyname,companies.location,jobs.salary
    -> from jobs
    -> join companies on jobs.companyid=companies.companyid
    -> where jobs.salary between 50000 and 90000;
+--------------------+---------------+-----------+----------+
| jobtitle           | companyname   | location  | salary   |
+--------------------+---------------+-----------+----------+
| Software Developer | Hexaware      | Chennai   | 70000.00 |
| Data Analyst       | Google        | Hyderabad | 85000.00 |
| Web Developer      | Microsoft     | Bangalore | 65000.00 |
| System Engineer    | TCS           | Mumbai    | 80000.00 |
| QA Engineer        | Tech Mahindra | Pune      | 60000.00 |
| Cloud Engineer     | Kaar          | Chennai   | 90000.00 |
+--------------------+---------------+-----------+----------+
6 rows in set (0.02 sec)
```

7. Write an SQL query that retrieves the job application history for a specific applicant. Allow a parameter for the ApplicantID, and return a result set with the job titles, company names, and application dates for all the jobs the applicant has applied to.

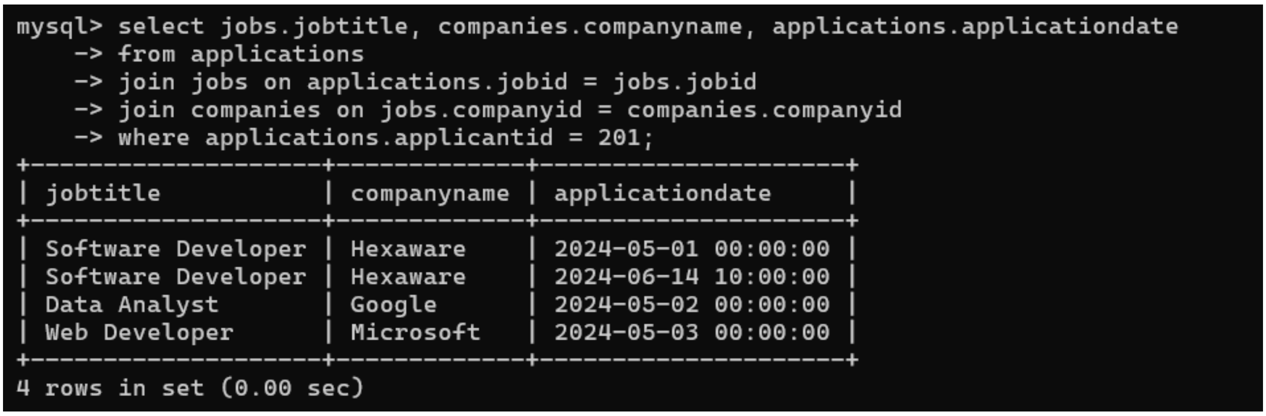
select jobs.jobtitle, companies.companyname, applications.applicationdate

from applications

join jobs on applications.jobid = jobs.jobid

join companies on jobs.companyid = companies.companyid

where applications.applicantid = 201;



```
mysql> select jobs.jobtitle, companies.companyname, applications.applicationdate
    -> from applications
    -> join jobs on applications.jobid = jobs.jobid
    -> join companies on jobs.companyid = companies.companyid
    -> where applications.applicantid = 201;
+--------------------+-------------+---------------------+
| jobtitle           | companyname | applicationdate     |
+--------------------+-------------+---------------------+
| Software Developer | Hexaware    | 2024-05-01 00:00:00 |
| Software Developer | Hexaware    | 2024-06-14 10:00:00 |
| Data Analyst       | Google      | 2024-05-02 00:00:00 |
| Web Developer      | Microsoft   | 2024-05-03 00:00:00 |
+--------------------+-------------+---------------------+
4 rows in set (0.00 sec)
```

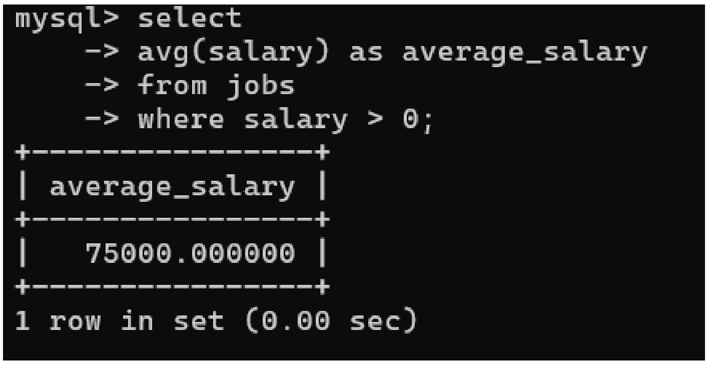
8. Create an SQL query that calculates and displays the average salary offered by all companies for job listings in the "Jobs" table. Ensure that the query filters out jobs with a salary of zero.

select

avg(salary) as average_salary

from jobs

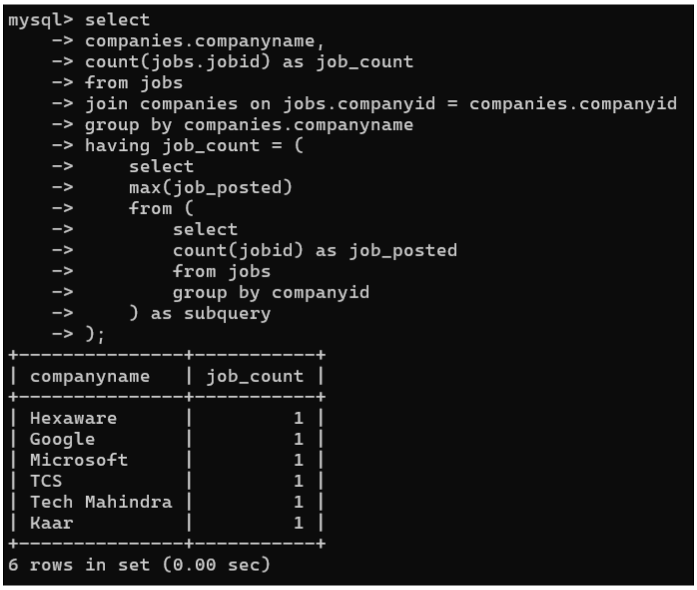
where salary > 0;



```
mysql> select
    -> avg(salary) as average_salary
    -> from jobs
    -> where salary > 0;
+----------------+
| average_salary |
+----------------+
|   75000.000000 |
+----------------+
1 row in set (0.00 sec)
```

9. Write an SQL query to identify the company that has posted the most job listings. Display the company name along with the count of job listings they have posted. Handle ties if multiple companies have the same maximum count.

```
select
companies.companyname,
count(jobs.jobid) as job_count
from jobs
join companies on jobs.companyid = companies.companyid
group by companies.companyname
having job_count = (
    select
    max(job_posted)
    from (
        select
        count(jobid) as job_posted
        from jobs
        group by companyid
    ) as subquery
);
```



```
mysql> select
    -> companies.companyname,
    -> count(jobs.jobid) as job_count
    -> from jobs
    -> join companies on jobs.companyid = companies.companyid
    -> group by companies.companyname
    -> having job_count = (
    ->     select
    ->     max(job_posted)
    ->     from (
    ->         select
    ->         count(jobid) as job_posted
    ->         from jobs
    ->         group by companyid
    ->     ) as subquery
    -> );
+---------------+-----------+
| companyname   | job_count |
+---------------+-----------+
| Hexaware      |         1 |
| Google        |         1 |
| Microsoft     |         1 |
| TCS           |         1 |
| Tech Mahindra |         1 |
| Kaar          |         1 |
+---------------+-----------+
6 rows in set (0.00 sec)
```

10. Find the applicants who have applied for positions in companies located in 'CityX' and have at least 3 years of experience.

select a.firstname, a.lastname

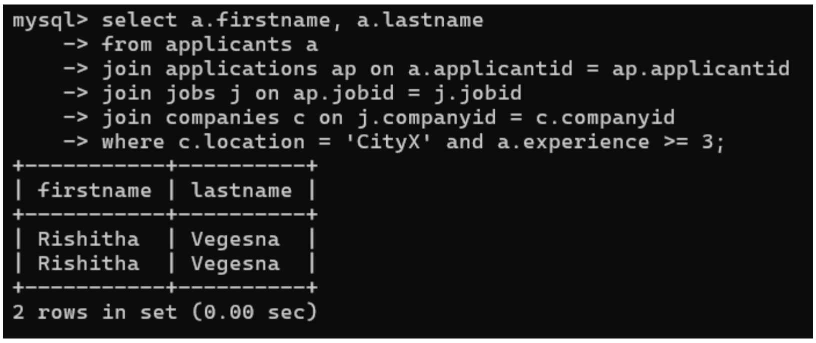
from applicants a

join applications ap on a.applicantid = ap.applicantid

join jobs j on ap.jobid = j.jobid

join companies c on j.companyid = c.companyid

where c.location = 'CityX' and a.experience >= 3;



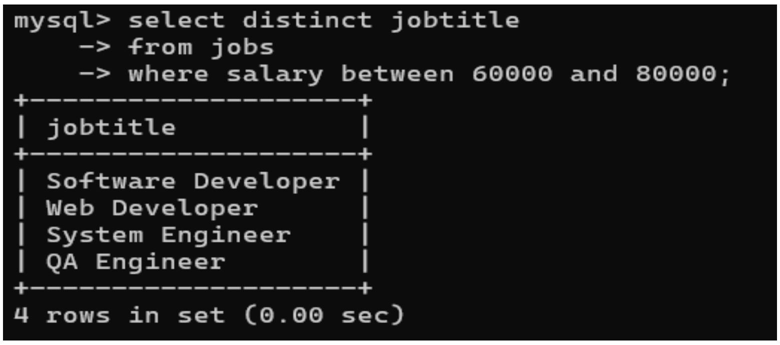
```
mysql> select a.firstname, a.lastname
    -> from applicants a
    -> join applications ap on a.applicantid = ap.applicantid
    -> join jobs j on ap.jobid = j.jobid
    -> join companies c on j.companyid = c.companyid
    -> where c.location = 'CityX' and a.experience >= 3;
+-----------+----------+
| firstname | lastname |
+-----------+----------+
| Rishitha  | Vegesna  |
| Rishitha  | Vegesna  |
+-----------+----------+
2 rows in set (0.00 sec)
```

11. Retrieve a list of distinct job titles with salaries between $60,000 and $80,000.

select distinct jobtitle

from jobs

where salary between 60000 and 80000;



```
mysql> select distinct jobtitle
    -> from jobs
    -> where salary between 60000 and 80000;
+--------------------+
| jobtitle           |
+--------------------+
| Software Developer |
| Web Developer      |
| System Engineer    |
| QA Engineer        |
+--------------------+
4 rows in set (0.00 sec)
```

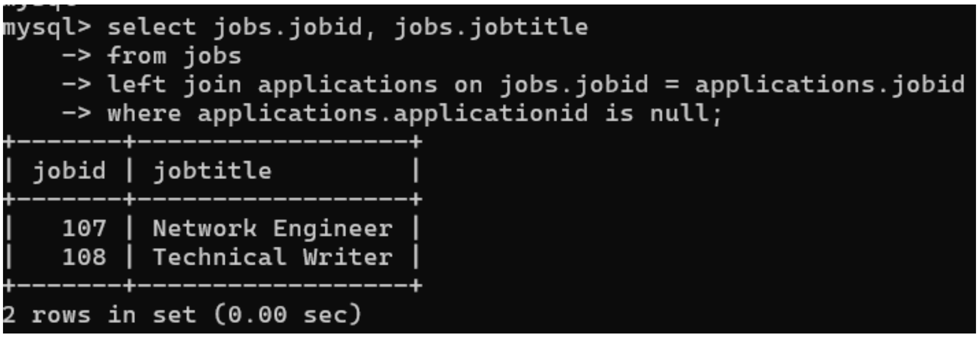
12. Find the jobs that have not received any applications.

select jobs.jobid, jobs.jobtitle

from jobs

left join applications on jobs.jobid = applications.jobid

where applications.applicationid is null;



```
mysql> select jobs.jobid, jobs.jobtitle
    -> from jobs
    -> left join applications on jobs.jobid = applications.jobid
    -> where applications.applicationid is null;
+-------+-------------------+
| jobid | jobtitle          |
+-------+-------------------+
|   107 | Network Engineer  |
|   108 | Technical Writer  |
+-------+-------------------+
2 rows in set (0.00 sec)
```

13. Retrieve a list of job applicants along with the companies they have applied to and the positions they have applied for.

select

applicants.applicantid,

applicants.firstname,

applicants.lastname,

companies.companyname,

jobs.jobtitle

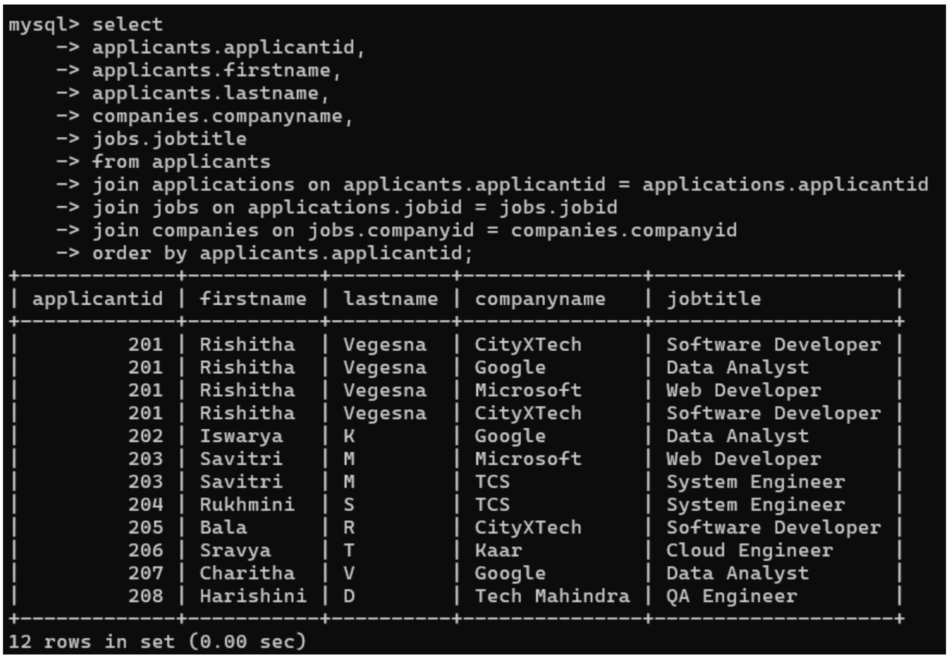
from applicants

join applications on applicants.applicantid = applications.applicantid

join jobs on applications.jobid = jobs.jobid

join companies on jobs.companyid = companies.companyid

order by applicants.applicantid;



```
mysql> select
    -> applicants.applicantid,
    -> applicants.firstname,
    -> applicants.lastname,
    -> companies.companyname,
    -> jobs.jobtitle
    -> from applicants
    -> join applications on applicants.applicantid = applications.applicantid
    -> join jobs on applications.jobid = jobs.jobid
    -> join companies on jobs.companyid = companies.companyid
    -> order by applicants.applicantid;
+-------------+-----------+----------+---------------+--------------------+
| applicantid | firstname | lastname | companyname   | jobtitle           |
+-------------+-----------+----------+---------------+--------------------+
|         201 | Rishitha  | Vegesna  | CityXTech     | Software Developer |
|         201 | Rishitha  | Vegesna  | Google        | Data Analyst       |
|         201 | Rishitha  | Vegesna  | Microsoft     | Web Developer      |
|         201 | Rishitha  | Vegesna  | CityXTech     | Software Developer |
|         202 | Iswarya   | K        | Google        | Data Analyst       |
|         203 | Savitri   | M        | Microsoft     | Web Developer      |
|         203 | Savitri   | M        | TCS           | System Engineer    |
|         204 | Rukhmini  | S        | TCS           | System Engineer    |
|         205 | Bala      | R        | CityXTech     | Software Developer |
|         206 | Sravya    | T        | Kaar          | Cloud Engineer     |
|         207 | Charitha  | V        | Google        | Data Analyst       |
|         208 | Harishini | D        | Tech Mahindra | QA Engineer        |
+-------------+-----------+----------+---------------+--------------------+
12 rows in set (0.00 sec)
```

14. Retrieve a list of companies along with the count of jobs they have posted, even if they have not received any applications.

select

companies.companyid,

companies.companyname,

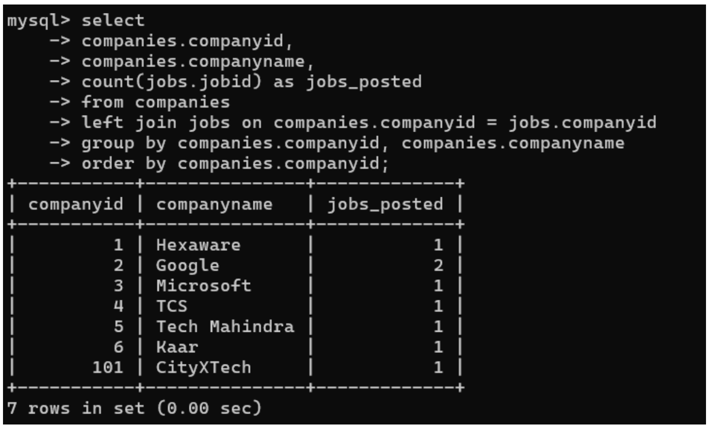
count(jobs.jobid) as jobs_posted

from companies

left join jobs on companies.companyid = jobs.companyid

group by companies.companyid, companies.companyname

order by companies.companyid;



```
mysql> select
    -> companies.companyid,
    -> companies.companyname,
    -> count(jobs.jobid) as jobs_posted
    -> from companies
    -> left join jobs on companies.companyid = jobs.companyid
    -> group by companies.companyid, companies.companyname
    -> order by companies.companyid;
+-----------+---------------+-------------+
| companyid | companyname   | jobs_posted |
+-----------+---------------+-------------+
|         1 | Hexaware      |           1 |
|         2 | Google        |           2 |
|         3 | Microsoft     |           1 |
|         4 | TCS           |           1 |
|         5 | Tech Mahindra |           1 |
|         6 | Kaar          |           1 |
|       101 | CityXTech     |           1 |
+-----------+---------------+-------------+
7 rows in set (0.00 sec)
```

15. List all applicants along with the companies and positions they have applied for, including those who have not applied.

select

applicants.applicantid,

applicants.firstname,

applicants.lastname,

companies.companyname,

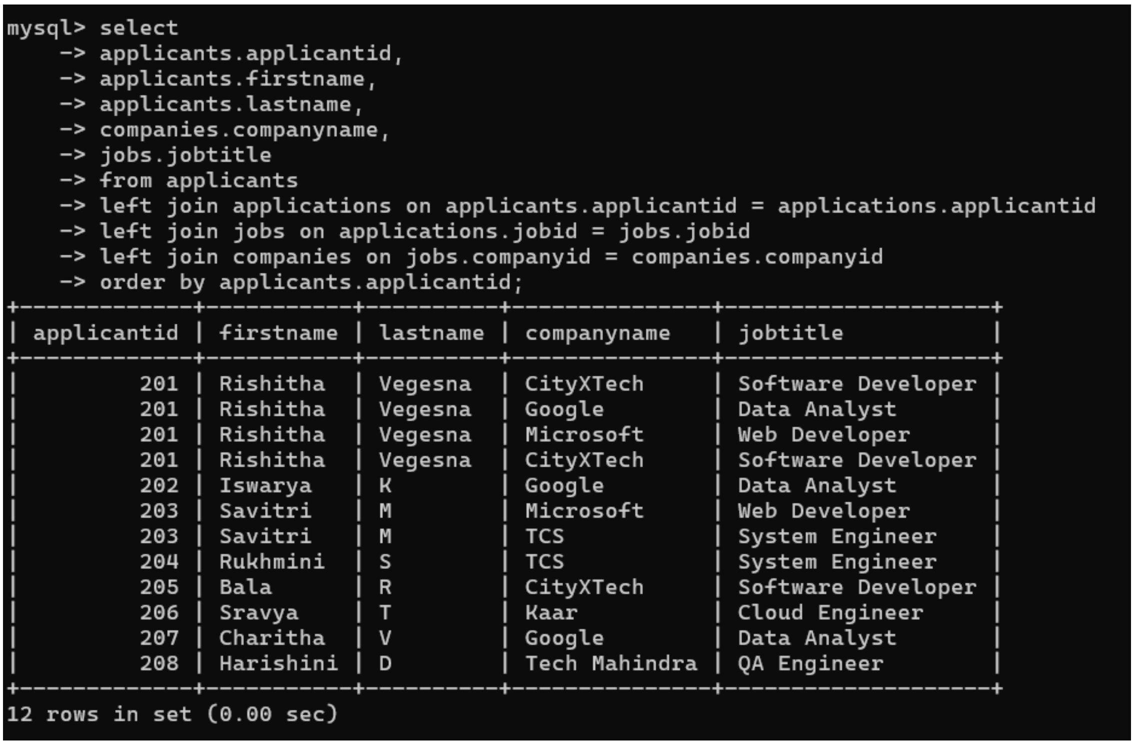
jobs.jobtitle

from applicants

left join applications on applicants.applicantid = applications.applicantid

left join jobs on applications.jobid = jobs.jobid

left join companies on jobs.companyid = companies.companyid order by applicants.applicantid;



```
mysql> select
    -> applicants.applicantid,
    -> applicants.firstname,
    -> applicants.lastname,
    -> companies.companyname,
    -> jobs.jobtitle
    -> from applicants
    -> left join applications on applicants.applicantid = applications.applicantid
    -> left join jobs on applications.jobid = jobs.jobid
    -> left join companies on jobs.companyid = companies.companyid
    -> order by applicants.applicantid;
+-------------+-----------+----------+---------------+--------------------+
| applicantid | firstname | lastname | companyname   | jobtitle           |
+-------------+-----------+----------+---------------+--------------------+
|         201 | Rishitha  | Vegesna  | CityXTech     | Software Developer |
|         201 | Rishitha  | Vegesna  | Google        | Data Analyst       |
|         201 | Rishitha  | Vegesna  | Microsoft     | Web Developer      |
|         201 | Rishitha  | Vegesna  | CityXTech     | Software Developer |
|         202 | Iswarya   | K        | Google        | Data Analyst       |
|         203 | Savitri   | M        | Microsoft     | Web Developer      |
|         203 | Savitri   | M        | TCS           | System Engineer    |
|         204 | Rukhmini  | S        | TCS           | System Engineer    |
|         205 | Bala      | R        | CityXTech     | Software Developer |
|         206 | Sravya    | T        | Kaar          | Cloud Engineer     |
|         207 | Charitha  | V        | Google        | Data Analyst       |
|         208 | Harishini | D        | Tech Mahindra | QA Engineer        |
+-------------+-----------+----------+---------------+--------------------+
12 rows in set (0.00 sec)
```

16. Find companies that have posted jobs with a salary higher than the average salary of all jobs.

select distinct

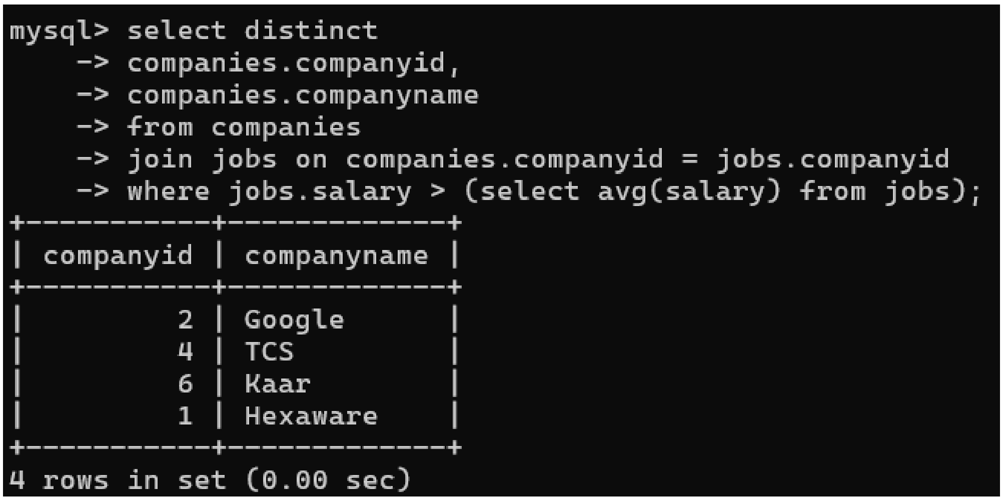
companies.companyid,

companies.companyname

from companies

join jobs on companies.companyid = jobs.companyid

where jobs.salary > (select avg(salary) from jobs);



```
mysql> select distinct
    -> companies.companyid,
    -> companies.companyname
    -> from companies
    -> join jobs on companies.companyid = jobs.companyid
    -> where jobs.salary > (select avg(salary) from jobs);
+-----------+-------------+
| companyid | companyname |
+-----------+-------------+
|         2 | Google      |
|         4 | TCS         |
|         6 | Kaar        |
|         1 | Hexaware    |
+-----------+-------------+
4 rows in set (0.00 sec)
```

17. Display a list of applicants with their names and a concatenated string of their city and state.

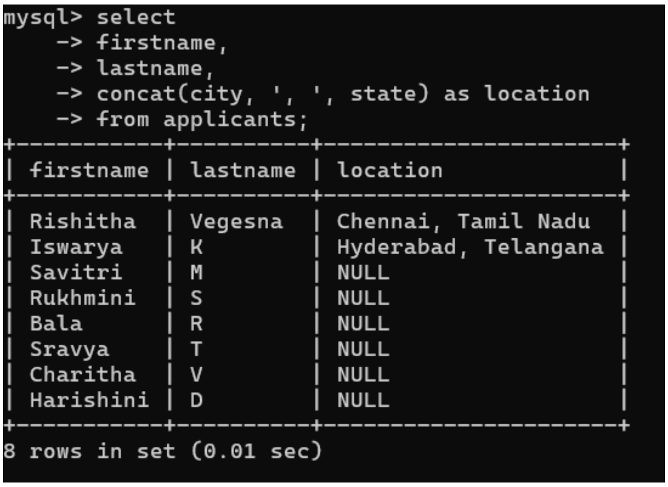
select

firstname,

lastname,

concat(city, ', ', state) as location

from applicants;



```
mysql> select
    -> firstname,
    -> lastname,
    -> concat(city, ', ', state) as location
    -> from applicants;
+-----------+----------+----------------------+
| firstname | lastname | location             |
+-----------+----------+----------------------+
| Rishitha  | Vegesna  | Chennai, Tamil Nadu  |
| Iswarya   | K        | Hyderabad, Telangana |
| Savitri   | M        | NULL                 |
| Rukhmini  | S        | NULL                 |
| Bala      | R        | NULL                 |
| Sravya    | T        | NULL                 |
| Charitha  | V        | NULL                 |
| Harishini | D        | NULL                 |
+-----------+----------+----------------------+
8 rows in set (0.01 sec)
```

18. Retrieve a list of jobs with titles containing either 'Developer' or 'Engineer'.

select

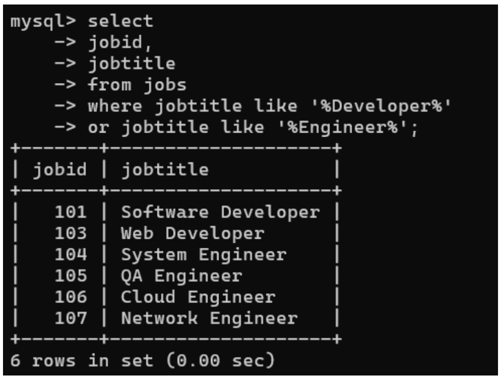
jobid,

jobtitle

from jobs

where jobtitle like '%Developer%'

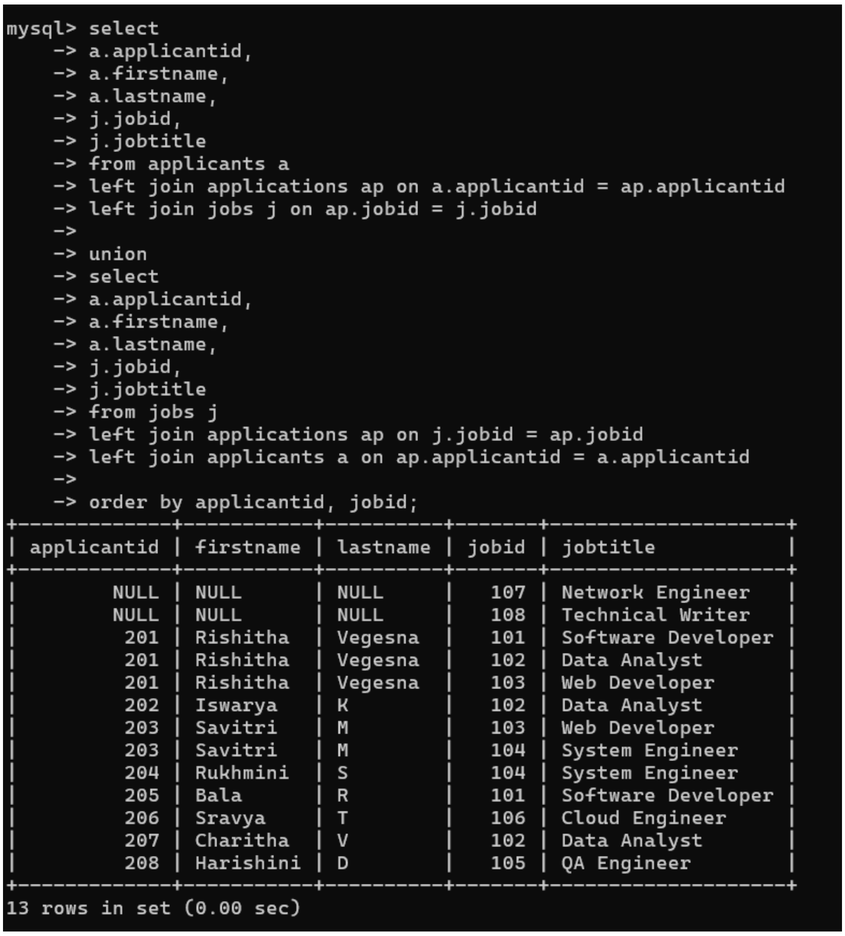
or jobtitle like '%Engineer%';



```
mysql> select
    -> jobid,
    -> jobtitle
    -> from jobs
    -> where jobtitle like '%Developer%'
    -> or jobtitle like '%Engineer%';
+-------+--------------------+
| jobid | jobtitle           |
+-------+--------------------+
|   101 | Software Developer |
|   103 | Web Developer      |
|   104 | System Engineer    |
|   105 | QA Engineer        |
|   106 | Cloud Engineer     |
|   107 | Network Engineer   |
+-------+--------------------+
6 rows in set (0.00 sec)
```

19. Retrieve a list of applicants and the jobs they have applied for, including those who have not applied and jobs without applicants.

```
select
a.applicantid,
a.firstname,
a.lastname,
j.jobid,
j.jobtitle
from applicants a
left join applications ap on a.applicantid = ap.applicantid
left join jobs j on ap.jobid = j.jobid
union
select
a.applicantid,
a.firstname,
a.lastname,
j.jobid,
j.jobtitle
from jobs j
left join applications ap on j.jobid = ap.jobid
left join applicants a on ap.applicantid = a.applicantid
order by applicantid, jobid;
```



```
mysql> select
    -> a.applicantid,
    -> a.firstname,
    -> a.lastname,
    -> j.jobid,
    -> j.jobtitle
    -> from applicants a
    -> left join applications ap on a.applicantid = ap.applicantid
    -> left join jobs j on ap.jobid = j.jobid
    ->
    -> union
    -> select
    -> a.applicantid,
    -> a.firstname,
    -> a.lastname,
    -> j.jobid,
    -> j.jobtitle
    -> from jobs j
    -> left join applications ap on j.jobid = ap.jobid
    -> left join applicants a on ap.applicantid = a.applicantid
    ->
    -> order by applicantid, jobid;
+-------------+-----------+----------+-------+--------------------+
| applicantid | firstname | lastname | jobid | jobtitle           |
+-------------+-----------+----------+-------+--------------------+
|        NULL | NULL      | NULL     |   107 | Network Engineer   |
|        NULL | NULL      | NULL     |   108 | Technical Writer   |
|         201 | Rishitha  | Vegesna  |   101 | Software Developer |
|         201 | Rishitha  | Vegesna  |   102 | Data Analyst       |
|         201 | Rishitha  | Vegesna  |   103 | Web Developer      |
|         202 | Iswarya   | K        |   102 | Data Analyst       |
|         203 | Savitri   | M        |   103 | Web Developer      |
|         203 | Savitri   | M        |   104 | System Engineer    |
|         204 | Rukhmini  | S        |   104 | System Engineer    |
|         205 | Bala      | R        |   101 | Software Developer |
|         206 | Sravya    | T        |   106 | Cloud Engineer     |
|         207 | Charitha  | V        |   102 | Data Analyst       |
|         208 | Harishini | D        |   105 | QA Engineer        |
+-------------+-----------+----------+-------+--------------------+
13 rows in set (0.00 sec)
```

20. List all combinations of applicants and companies where the company is in a specific city and the applicant has more than 2 years of experience. For example: city=Chennai

select

applicants.applicantid,

applicants.firstname,

applicants.lastname,

companies.companyid,

companies.companyname,

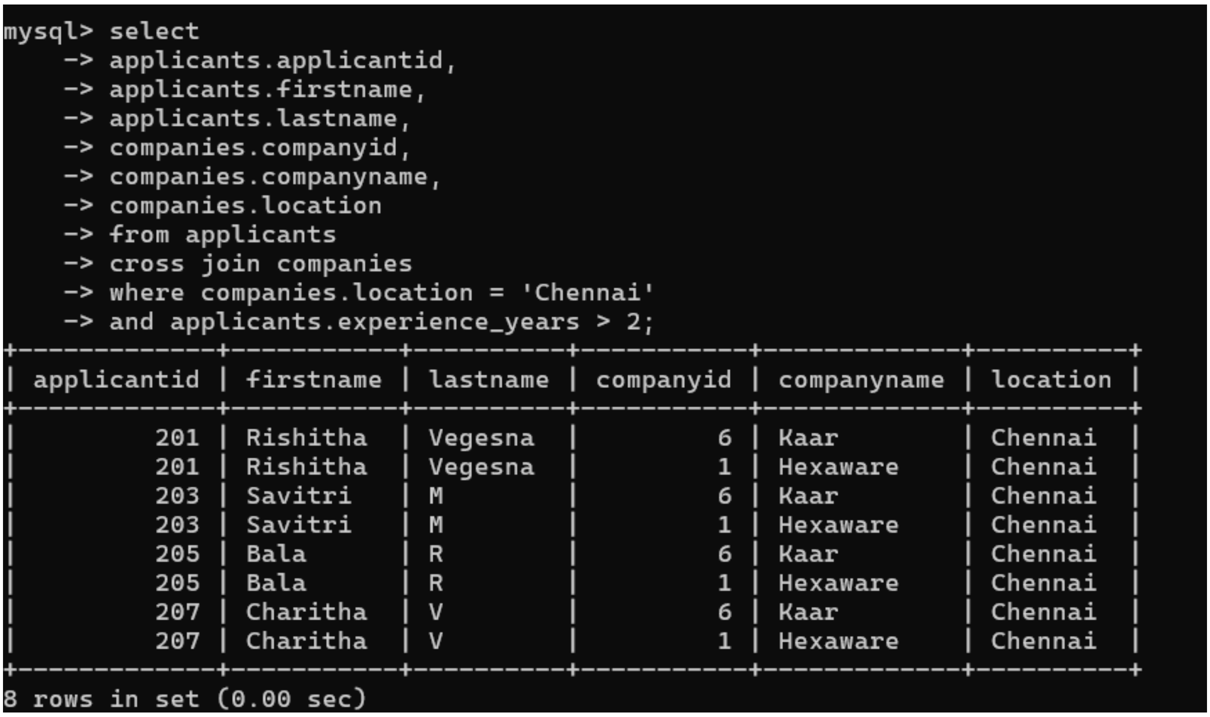
companies.location

from applicants

cross join companies

where companies.location = 'Chennai'

and applicants.experience_years > 2;



```
mysql> select
    -> applicants.applicantid,
    -> applicants.firstname,
    -> applicants.lastname,
    -> companies.companyid,
    -> companies.companyname,
    -> companies.location
    -> from applicants
    -> cross join companies
    -> where companies.location = 'Chennai'
    -> and applicants.experience_years > 2;
+-------------+-----------+----------+-----------+-------------+----------+
| applicantid | firstname | lastname | companyid | companyname | location |
+-------------+-----------+----------+-----------+-------------+----------+
|         201 | Rishitha  | Vegesna  |         6 | Kaar        | Chennai  |
|         201 | Rishitha  | Vegesna  |         1 | Hexaware    | Chennai  |
|         203 | Savitri   | M        |         6 | Kaar        | Chennai  |
|         203 | Savitri   | M        |         1 | Hexaware    | Chennai  |
|         205 | Bala      | R        |         6 | Kaar        | Chennai  |
|         205 | Bala      | R        |         1 | Hexaware    | Chennai  |
|         207 | Charitha  | V        |         6 | Kaar        | Chennai  |
|         207 | Charitha  | V        |         1 | Hexaware    | Chennai  |
+-------------+-----------+----------+-----------+-------------+----------+
8 rows in set (0.00 sec)
```