RDBMS Coding Challenge Date:18-4-2024

1.Creating database:

Create database exam;

Use exam;

o/p:Database Changed

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

Creating table name companies

create table companies(company_id integer(20) primary key,company_name varchar(30),Location varchar(30));

o/p: Query OK, 0 rows affected, 1 warning (0.03 sec)

```
mysql> create table companies(company_id integer(20) primary key,company_name varchar(30),Location varchar(30));
Query OK, 0 rows affected, 1 warning (0.03 sec)
```

Creating table jobs:

create table jobs (job_id integer(20),company_id integer(20),job_title varchar(30),job_description text(50),job_location varchar(30),salary decimal(10,2),job_type varchar(30),posted_date date);

o/p:Query OK, 0 rows affected, 2 warnings (0.02 sec)

Creating applicants table:

create table applicants(applicant_id integer(20) primary key,first_name varchar(30),last_name varchar(30),email varchar(30),phone varchar(20),resume text(100));

o/pQuery OK, 0 rows affected, 1 warning (0.01 sec)

Creating table applications:

create table applications(application_id integer(30) primary key,job_id integer(30),applicant_id integer(20),application_date date,cover_letter text(100));

o/p:Query OK, 0 rows affected, 3 warnings (0.02 sec)

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

enforcing foreign key constraints:

alter table jobs add constraint fk1 foreign key(company_id) references companies(company_id);

mysql> alter table applications add constraint fk2 foreign key(job_id) references jobs(job_id);

alter table applications add constraint fk3 foreign key(applicant_id) references applicants(applicant_id);

inserting data:

```
mysql> INSERT INTO companies (company_id, company_name, Location) VALUES
    -> (1, 'Hexaware', 'Pune'),
    -> (2, 'Infosys', 'Mumbai'),
    -> (3, 'Accenture', 'Hyderabad'),
    -> (4, 'TCS', 'Chennai'),
    -> (5, 'Pennent', 'Vizag'),
    -> (6, 'Amedus', 'Bangalore');
```

tables

companies:

```
+-------------+--------------------+-------------+
| company_id  | company_name       | Location    |
+-------------+--------------------+-------------+
|           1 | Hexaware           | Pune        |
|           2 | Infosys            | Mumbai      |
|           3 | Accenture          | Hyderabad   |
|           4 | TCS                | Chennai     |
|           5 | Pennent            | Vizag       |
|           6 | Amedus             | Bangalore   |
|           7 | RMS Technologies   | Hyderabad   |
+-------------+--------------------+-------------+
7 rows in set (0.00 sec)
```

jobs:

```
mysql> select * from jobs;
+--------+------------+-----------+--------------------+--------------+----------+-----------+-------------+
| job_id | company_id | job_title | job_description    | job_location | salary   | job_type  | posted_date |
+--------+------------+-----------+--------------------+--------------+----------+-----------+-------------+
|    101 |          1 | Developer | developing software | pune         | 50000.00 | full-time | 2022-04-12  |
|    102 |          1 | Testing   | testing software    | pune         | 20000.00 | part-time | 2023-06-12  |
|    103 |          2 | developer | Developing software | Chennai      | 80000.00 | full-time | 2021-07-24  |
|    104 |          2 | Testing   | testing software    | hyderabad    | 20000.00 | full-time | 2022-12-27  |
|    105 |          3 | Testing   | testing software    | Hyderabad    | 60000.00 | part-time | 2022-03-01  |
|    106 |          3 | developer | developing software | bangalore    | 70000.00 | part-time | 2023-07-11  |
|    107 |          4 | developer | developing software | Chennai      | 45000.00 | full-time | 2023-06-12  |
|    108 |          4 | Testing   | testing software    | bhuvaneswar  | 30000.00 | part-time | 2022-02-01  |
|    109 |          5 | developer | developing software | bhuvaneswar  | 25000.00 | full-time | 2022-09-17  |
|    110 |          5 | Testing   | testing software    | vizag        | 55000.00 | part-time | 2023-07-17  |
+--------+------------+-----------+--------------------+--------------+----------+-----------+-------------+
10 rows in set (0.00 sec)
```

Applicants:

```
mysql> select * from applicants;
+--------------+------------+-----------+--------------------+------------+---------+
| applicant_id | first_name | last_name | email              | phone      | resume  |
+--------------+------------+-----------+--------------------+------------+---------+
|          501 | vishnu     | charan    | vcptes44@gmail.com | 1234574337 | resume1 |
|          502 | soma       | shekar    | shek@gmail.com     | 9495743371 | resume2 |
|          503 | satya sai  | pavan     | pavan@gmail.com    | 9848652314 | resume3 |
+--------------+------------+-----------+--------------------+------------+---------+
3 rows in set (0.00 sec)
```

Appliacations:

application_id	job_id	applicant_id	application_date	cover_letter
201	101	501	2023-05-15	coverletter1
202	102	502	2023-06-20	coverletter2
203	103	503	2023-07-10	coverletter3
204	101	501	2023-08-05	coverletter4
205	102	503	2023-09-12	coverletter5
206	104	503	2023-10-18	coverletter6
207	107	501	2023-11-25	coverletter7
208	108	501	2023-12-30	coverletter8
209	109	503	2024-01-10	coverletter9
210	110	503	2024-02-22	coverletter10

10 rows in set (0.00 sec)

Queries:

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

Create database if not exists exam

Create table applica

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 j.job_title,count(application_id) as number

from jobs j join applications a on a.job_id=j.job_id

group by j.job_id;

o/p:

job_title	number
Developer	2
Testing	2
developer	1
Testing	1
developer	1
Testing	1
developer	1
Testing	1

8 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 c.company_name,j.job_title,c.location,j.salary from companies c join jobs j on c.company_id=j.company_id

-> where j.salary between 20000 and 50000;

o/p:

company_name	job_title	location	salary
Hexaware	Developer	Pune	50000.00
Hexaware	Testing	Pune	20000.00
Infosys	Testing	Mumbai	20000.00
TCS	developer	cityX	45000.00
TCS	Testing	cityX	30000.00
Pennent	developer	Vizag	25000.00

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

```
mysql> select j.job_title,c.company_name,app.application_date  from
    ->    companies c join jobs j on c.company_id=j.company_id
    ->    join applications app on app.job_id=j.job_id
    ->    join applicants a on a.applicant_id=app.applicant_id
    ->    where app.applicant_id=501;
```

o/p:

job_title	company_name	application_date
Developer	Hexaware	2023-05-15
Developer	Hexaware	2023-08-05
developer	TCS	2023-11-25
Testing	TCS	2023-12-30

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 c.company_name,c.company_id,avg(salary)
    -> from companies c join jobs j on c.company_id=j.company_id
    -> group by company_id;
```

o/p:

company_name	company_id	avg(salary)
Hexaware	1	35000.000000
Infosys	2	50000.000000
Accenture	3	65000.000000
TCS	4	37500.000000
Pennent	5	40000.000000

5 rows 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.

```
mysql> SELECT c.company_id, c.company_name, COUNT(*) AS job_number
    -> FROM jobs j JOIN companies  c ON j.company_id = c.company_id
    -> GROUP BY c.company_id, c.company_name
    -> ORDER BY job_number  DESC
    -> LIMIT 1;
```

o/p:

```
+------------+--------------+------------+
| company_id | company_name | job_number |
+------------+--------------+------------+
|          1 | Hexaware     |          2 |
+------------+--------------+------------+
1 row 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.applicant_id,a.first_name,c.company_name
    ->     from companies c join jobs j on j.company_id=c.company_id
    ->     join applications app on j.job_id=app.job_id
    ->     join applicants a on a.applicant_id=app.applicant_id
    ->     where c.location='cityX' and year(j.posted_date)<2020  ;
```

o/p:

```
+--------------+------------+--------------+
| applicant_id | first_name | company_name |
+--------------+------------+--------------+
|          503 | satya sai  | Infosys      |
+--------------+------------+--------------+
```

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

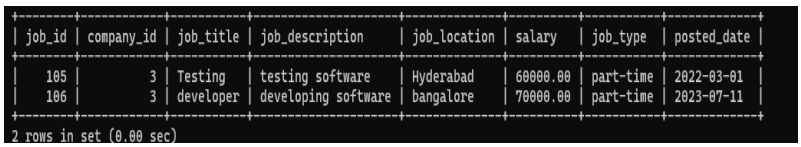
```
select job_title from jobs where salary between 60000 and 80000;
```

```
+-----------+
| job_title |
+-----------+
| developer |
| Testing   |
| developer |
+-----------+
3 rows in set (0.00 sec)
```

12. Find the jobs that have not received any applications

mysql> select * from jobs j where j.job_id not in(select job_id from applications);

o/p:



job_id	company_id	job_title	job_description	job_location	salary	job_type	posted_date
105	3	Testing	testing software	Hyderabad	60000.00	part-time	2022-03-01
106	3	developer	developing software	bangalore	70000.00	part-time	2023-07-11

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

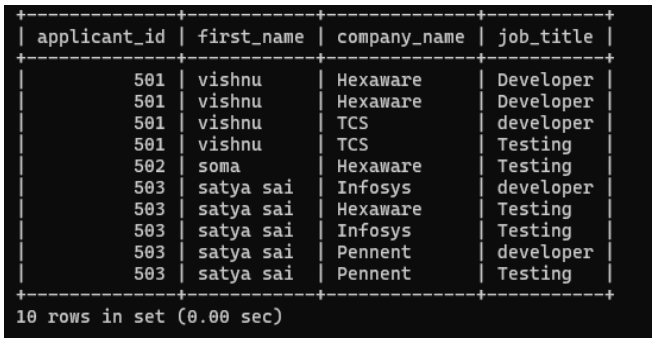
select a.applicant_id,a.first_name,c.company_name,j.job_title

-> from companies c join jobs j on j.company_id=c.company_id

-> join applications app on j.job_id=app.job_id

-> join applicants a on a.applicant_id=app.applicant_id

-> order by a.applicant_id;



applicant_id	first_name	company_name	job_title
501	vishnu	Hexaware	Developer
501	vishnu	Hexaware	Developer
501	vishnu	TCS	developer
501	vishnu	TCS	Testing
502	soma	Hexaware	Testing
503	satya sai	Infosys	developer
503	satya sai	Hexaware	Testing
503	satya sai	Infosys	Testing
503	satya sai	Pennent	developer
503	satya sai	Pennent	Testing

10 rows in set (0.00 sec)

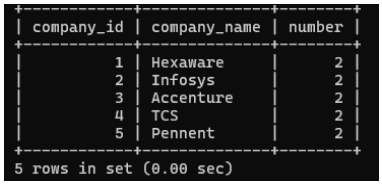
14. Retrieve a list of companies along with the count of jobs they have posted, even if they have not received any applications

select c.company_id,c.company_name,count(j.job_id) as number

-> from jobs j join companies c on c.company_id=j.company_id

-> group by c.company_id,c.company_name;

o/p:



company_id	company_name	number
1	Hexaware	2
2	Infosys	2
3	Accenture	2
4	TCS	2
5	Pennent	2

5 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

```
mysql> select  a.applicant_id,a.first_name,c.company_name,j.job_title  from
    ->     applicants a left join applications app on a.applicant_id=app.applicant_id
    ->      right join jobs j on j.job_id=app.job_id
    ->      join companies c on c.company_id=j.company_id;
```

applicant_id	first_name	company_name	job_title
501	vishnu	Hexaware	Developer
501	vishnu	Hexaware	Developer
502	soma	Hexaware	Testing
503	satya sai	Hexaware	Testing
503	satya sai	Infosys	developer
503	satya sai	Infosys	Testing
NULL	NULL	Accenture	Testing
NULL	NULL	Accenture	developer
501	vishnu	TCS	developer
501	vishnu	TCS	Testing
503	satya sai	Pennent	developer
503	satya sai	Pennent	Testing

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

```
 SELECT DISTINCT c.company_id,c.company_name
    -> FROM     companies c join jobs j on c.company_id = j.company_id
          Where j.salary>(select avg(salary) from jobs);
```

o/p:

company_id	company_name
1	Hexaware
2	Infosys
3	Accenture
5	Pennent

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

```
mysql>      select  a.applicant_id,concat(a.first_name,'  ',a.last_name) as name,c.location as name
from
    ->     applicants a join applications app on a.applicant_id=app.applicant_id
    ->      join jobs j on j.job_id=app.job_id
    ->     join companies c on c.company_id=j.company_id;
```

o/p:

applicant_id	name	name
501	vishnu charan	Pune
502	soma shekar	Pune
503	satya sai pavan	Mumbai
501	vishnu charan	Pune
503	satya sai pavan	Pune
503	satya sai pavan	Mumbai
501	vishnu charan	cityX
501	vishnu charan	cityX
503	satya sai pavan	Vizag
503	satya sai pavan	Vizag

10 rows in set (0.00 sec)

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

mysql> select * from jobs where job_title like 'Engi%' or job_title like 'De%';

o/p:

job_id	company_id	job_title	job_description	job_location	salary	job_type	posted_date
101	1	Developer	developing software	pune	50000.00	full-time	2022-04-12
103	2	developer	Developing software	cityX	80000.00	full-time	2019-01-01
106	3	developer	developing software	bangalore	70000.00	part-time	2023-07-11
107	4	developer	developing software	cityX	45000.00	full-time	2023-06-12
109	5	developer	developing software	bhuvaneswar	25000.00	full-time	2022-09-17

5 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.first_name,j.job_title from applicants a left join applications app on a.applicant_id=app.applicant_id

left join jobs j on j.job_id=app.job_id;

o/p:

first_name	job_title
vishnu	Developer
vishnu	Developer
vishnu	developer
vishnu	Testing
soma	Testing
satya sai	developer
satya sai	Testing
satya sai	Testing
satya sai	developer
satya sai	Testing
raja	NULL

11 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

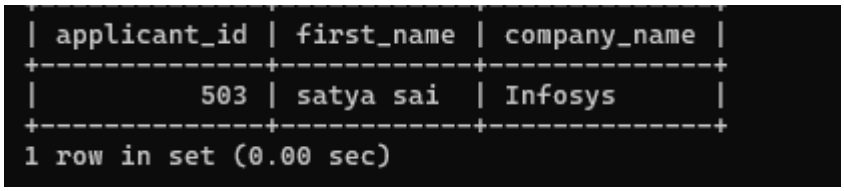
mysql> select a.applicant_id,a.first_name,c.company_name

-> from companies c join jobs j on j.company_id=c.company_id

-> join applications app on j.job_id=app.job_id

-> join applicants a on a.applicant_id=app.applicant_id

-> where j.job_location='cityX' and year(j.posted_date)<2021;



applicant_id	first_name	company_name
503	satya sai	Infosys

1 row in set (0.00 sec)