

SQL CODING CHALLENGE ----- Assignment 2 -----

CareerHub, The Job Board (SHRINIDHI.S.S)

1. Provide a SQL script that initializes the database for the Job Board scenario “CareerHub”.
2. Create tables for Companies, Jobs, Applicants and Applications.
3. Define appropriate primary keys, foreign keys, and constraints.
4. Ensure the script handles potential errors, such as if the database or tables already exist.

```
create database careerhub;  
use careerhub;
```

```
CREATE TABLE Companies (  
    CompanyID INT PRIMARY KEY,  
    CompanyName VARCHAR(20),  
    Location VARCHAR(20));  
  
CREATE TABLE Jobs (  
    JobID INT PRIMARY KEY,  
    CompanyID INT,  
    JobTitle VARCHAR(255),  
    JobDescription TEXT,  
    JobLocation VARCHAR(255),  
    Salary DECIMAL(10, 2),  
    JobType VARCHAR(50),  
    PostedDate DATETIME,  
    FOREIGN KEY (CompanyID) REFERENCES Companies(CompanyID));
```

```
CREATE TABLE Applicants (  
    ApplicantID INT PRIMARY KEY,  
    FirstName VARCHAR(255),
```

LastName VARCHAR(255),

Email VARCHAR(255),

Phone VARCHAR(20),

Resume TEXT);

CREATE TABLE 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));

INSERT INTO Companies (CompanyID, CompanyName, Location) VALUES

(1, 'Company 1', 'New York'),

(2, 'Company 2', 'Georgia'),

(3, 'Company 3', 'Chicago'),

(4, 'Company 4', 'Paris'),

(5, 'Company 5', 'Canada');

INSERT INTO Jobs (JobID, CompanyID, JobTitle, JobDescription, JobLocation, Salary, JobType, PostedDate) VALUES

(11, 1, 'Software Engineer', 'We are looking for a skilled and experienced software engineer to join our team.', 'New York', 80000.00, 'Full-time', '2024-04-15 09:00:00'),

(22, 2, 'Marketing Specialist', 'Seeking a creative marketing specialist to develop and implement marketing strategies.', 'Georgia', 70000.00, 'Full-time', '2024-04-16 10:30:00'),

(33, 3, 'Data Analyst', 'We are hiring a data analyst to analyze and interpret complex data sets.', 'Chicago', 75000.00, 'WFH', '2024-04-17 11:45:00'),

(44, 4, 'UX/UI Designer', 'Join our team as a UX/UI designer to create engaging user experiences.', 'Paris', 85000.00, 'Part-time', '2024-04-18 13:15:00'),

(55, 5, 'Accountant', 'We are seeking an experienced accountant to manage financial transactions.', 'Canada', 70000.00, 'Full-time', '2024-04-19 14:45:00');

INSERT INTO Applicants (ApplicantID, FirstName, LastName, Email, Phone, Resume)
VALUES

(101, 'Tim', 'David', 'timdavid@example.com', '123-456-7890', 'Experienced software engineer with expertise in Java, Python, and SQL.'),

(102, 'Tim', 'Brook', 'timbrook@example.com', '987-654-3210', 'Marketing specialist with 3 years of experience developing and executing digital marketing campaigns.'),

(103, 'Jos', 'Butler', 'josbutler@example.com', '456-789-0123', 'Data analyst with strong analytical skills and proficiency in statistical analysis tools'),

(104, 'Harry', 'Edward', 'harryedward@example.com', '321-654-0987', 'Creative UX/UI designer with a passion for user-centered design.'),

(105, 'Niall', 'Horon', 'niallhoron@example.com', '789-012-3456', 'Detail-oriented accountant with a CPA certification and 4 years of experience in financial accounting.');

INSERT INTO Applications (ApplicationID, JobID, ApplicantID, ApplicationDate, CoverLetter) VALUES

(1001, 11, 101, '2024-04-20 09:30:00', 'Dear Hiring Manager, I am excited to apply for the Software Engineer position at Example Company 1...'),

(1002, 22, 102, '2024-04-21 10:45:00', 'To Whom It May Concern, I am writing to express my interest in the Marketing Specialist position at Example Company 2...'),

(1003, 33, 103, '2024-04-22 11:00:00', 'Hello, I am applying for the Data Analyst position at Example Company 3. I believe my analytical skills and experience make me a strong candidate...'),

(1004, 44, 104, '2024-04-23 12:15:00', 'Dear Hiring Team, I am writing to apply for the UX/UI Designer position at Example Company 4. With a strong background in user experience design...'),

(1005, 55, 105, '2024-04-24 13:30:00', 'Dear Hiring Committee, I am excited to submit my application for the Accountant position at Example Company 5. With years of experience in financial management...');

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Administration
- Schemas

SQL File 3*

```
1 create database careerhub;
2 use careerhub;
3 CREATE TABLE Companies (
4   CompanyID INT PRIMARY KEY,
5   CompanyName VARCHAR(20),
6   Location VARCHAR(20));
7 CREATE TABLE Jobs (
8   JobID INT PRIMARY KEY,
9   CompanyID INT,
10  JobTitle VARCHAR(255),
11  JobDescription TEXT,
12  JobLocation VARCHAR(255),
13  Salary DECIMAL(10, 2);
```

Output

#	Time	Action	Message	Duration / Fetch
1	09:41:27	create database careerhub	1 row(s) affected	0.047 sec
2	09:41:38	use careerhub	0 row(s) affected	0.015 sec
3	09:45:55	CREATE TABLE Companies (CompanyID INT PRIMARY KEY, CompanyName VARC...	0 row(s) affected	0.109 sec
4	09:47:43	CREATE TABLE Jobs (JobID INT PRIMARY KEY, CompanyID INT, JobTitle VARCH...	0 row(s) affected	0.062 sec
5	09:48:33	CREATE TABLE Applicants (ApplicantID INT PRIMARY KEY, FirstName VARCHAR...	0 row(s) affected	0.047 sec
6	09:49:23	CREATE TABLE Applications (ApplicationID INT PRIMARY KEY, JobID INT, Appli...	0 row(s) affected	0.078 sec

Object Info Session

ENG IN 09:49 02-05-2024

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Administration
- Schemas

SQL File 3*

```
38 INSERT INTO Companies (CompanyID, CompanyName, Location) VALUES
39 (1, 'Company 1', 'New York'),
40 (2, 'Company 2', 'Georgia'),
41 (3, 'Company 3', 'Chicago'),
42 (4, 'Company 4', 'Paris'),
43 (5, 'Company 5', 'Canada');
44 INSERT INTO Jobs (JobID, CompanyID, JobTitle, JobDescription, JobLocation, Salary, JobType, PostedDate) VA
45 (11, 1, 'Software Engineer', 'We are looking for a skilled and experienced software engineer to join our t
46 (22, 2, 'Marketing Specialist', 'Seeking a creative marketing specialist to develop and implement marketin
47 (33, 3, 'Data Analyst', 'We are hiring a data analyst to analyze and interpret complex data sets.', 'Chica
48 (44, 4, 'UX/UI Designer', 'Join our team as a UX/UI designer to create engaging user experiences.', 'Paris
49 (55, 5, 'Accountant', 'We are seeking an experienced accountant to manage financial transactions.', 'Canad
50 INSERT INTO Applicants (ApplicantID, FirstName, LastName, Email, Phone, Resume) VALUES
51 (101, 'Tim', 'David', 'tim@timdavid.com', '123.456.7890', 'Experienced software engineer with extensiv
```

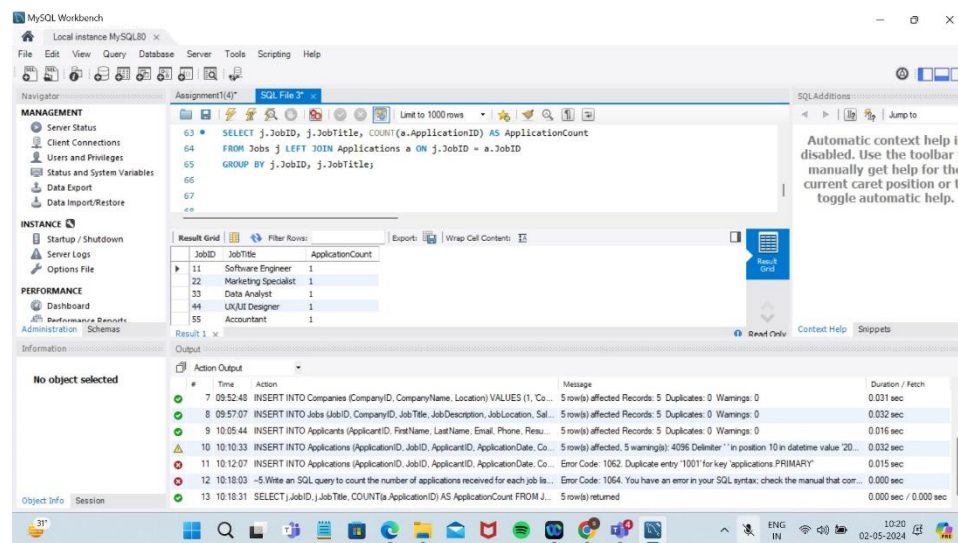
Output

#	Time	Action	Message	Duration / Fetch
5	09:48:33	CREATE TABLE Applicants (ApplicantID INT PRIMARY KEY, FirstName VARCHA...	0 row(s) affected	0.047 sec
6	09:49:23	CREATE TABLE Applications (ApplicationID INT PRIMARY KEY, JobID INT, Appl...	0 row(s) affected	0.078 sec
7	09:52:48	INSERT INTO Companies (CompanyID, CompanyName, Location) VALUES (1, Co...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.031 sec
8	09:57:07	INSERT INTO Jobs (JobID, CompanyID, JobTitle, JobDescription, JobLocation, Sal...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.032 sec
9	10:05:44	INSERT INTO Applicants (ApplicantID, FirstName, LastName, Email, Phone, Resu...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
10	10:10:33	INSERT INTO Applications (ApplicationID, JobID, ApplicantID, ApplicationDate, Co...	5 row(s) affected. 5 warning(s): 4096 Delimiter '"' in position 10 in datetime value '20...	0.032 sec

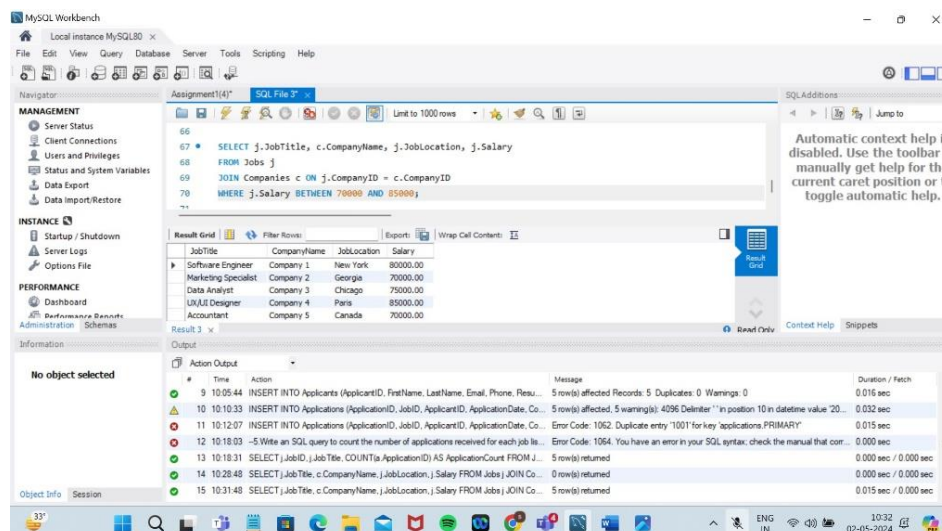
Object Info Session

ENG IN 12:08 02-05-2024

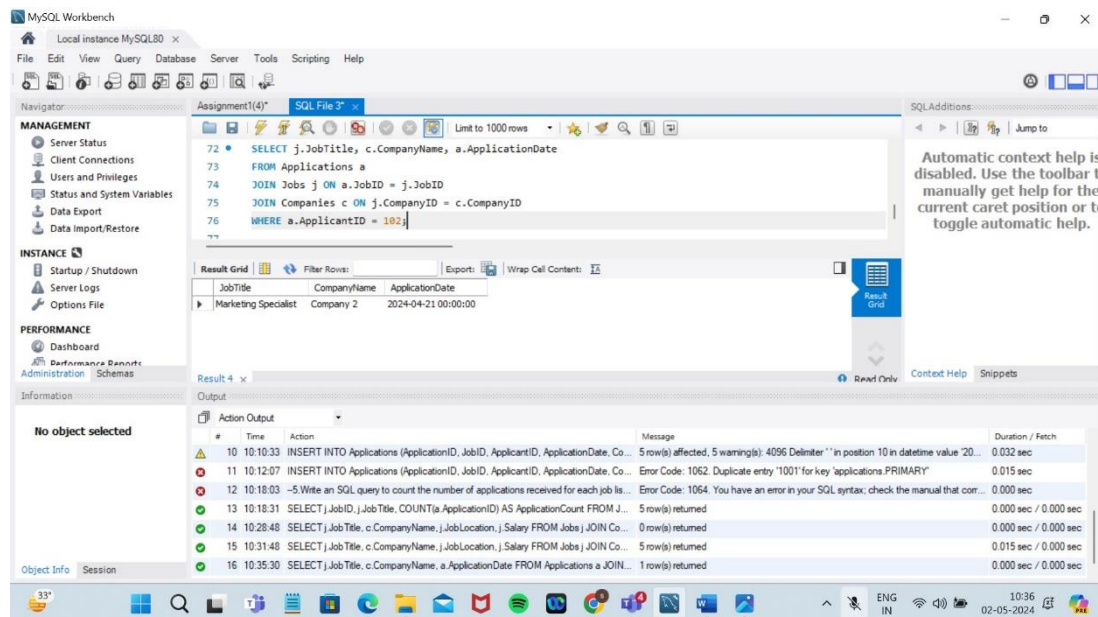
5. Write an SQL query to count the number of applications received for each job listing in the "Jobs" table. Display the job title and the corresponding application count. Ensure that it lists all jobs, even if they have no applications.



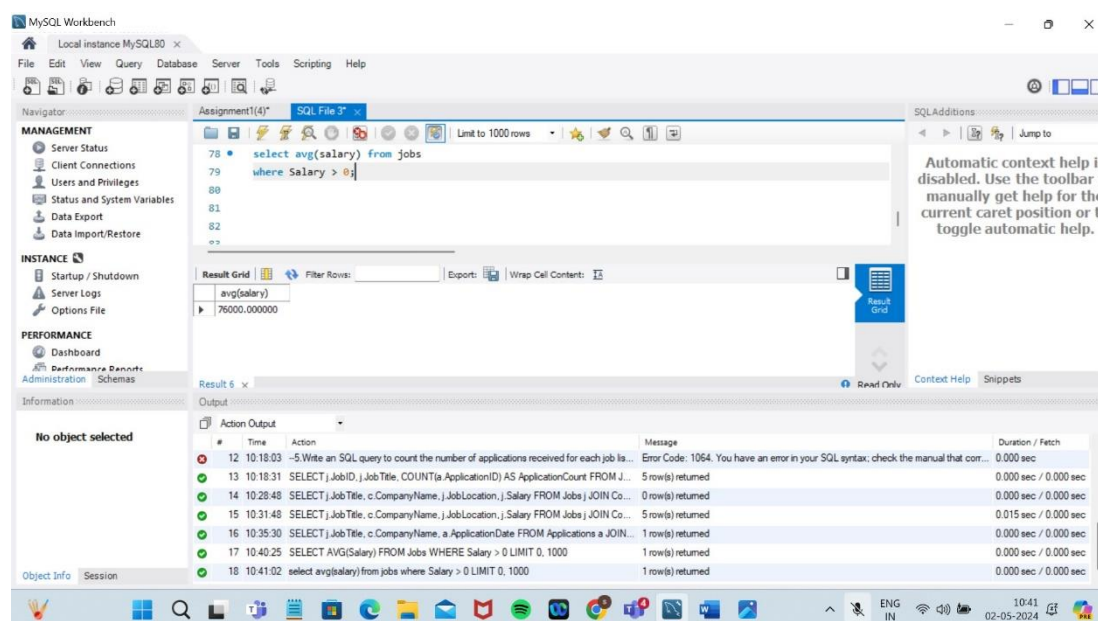
6. Develop an SQL query that retrieves job listings from the "Jobs" table within a specified salary range. Allow parameters for the minimum and maximum salary values. Display the job title, company name, location, and salary for each matching job.



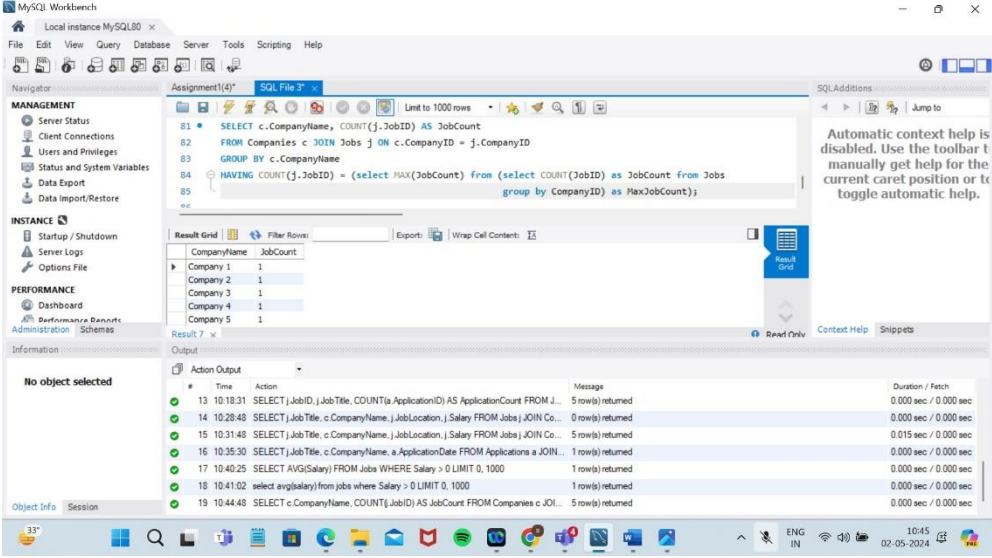
7. Write an SQL query that retrieves the job application history for a specific applicant. Allow a parameter for the ApplicantID, and return a result set with the job titles, company names, and application dates for all the jobs the applicant has applied to.



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.



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.



The screenshot shows MySQL Workbench with a query window titled 'Assignment1(4)*'. The query is as follows:

```

11 SELECT c.CompanyName, COUNT(j.JobID) AS JobCount
12 FROM Companies c JOIN Jobs j ON c.CompanyID = j.CompanyID
13 GROUP BY c.CompanyName
14 HAVING COUNT(j.JobID) = (select MAX(JobCount) from Jobs
15                          group by CompanyID) as MaxJobCount);

```

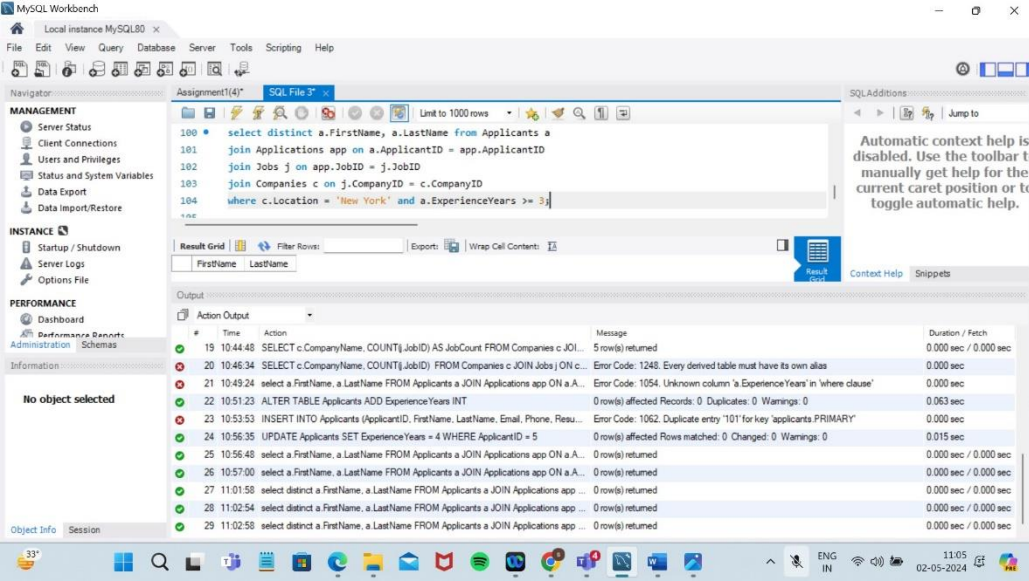
The Results grid shows the following data:

CompanyName	JobCount
Company 1	1
Company 2	1
Company 3	1
Company 4	1
Company 5	1

The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
13	10:18:31	SELECT j.JobID, j.JobTitle, COUNT(a.ApplicationID) AS ApplicationCount FROM J...	5 row(s) returned	0.000 sec / 0.000 sec
14	10:28:48	SELECT j.JobTitle, c.CompanyName, j.JobLocation, j.Salary FROM Jobs j JOIN Co...	0 row(s) returned	0.000 sec / 0.000 sec
15	10:31:48	SELECT j.JobTitle, c.CompanyName, j.JobLocation, j.Salary FROM Jobs j JOIN Co...	5 row(s) returned	0.015 sec / 0.000 sec
16	10:35:30	SELECT j.JobTitle, c.CompanyName, a.ApplicationDate FROM Applications a JOIN...	1 row(s) returned	0.000 sec / 0.000 sec
17	10:40:25	SELECT AVG(Salary) FROM Jobs WHERE Salary > 0 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
18	10:41:02	select avg(salary) from jobs where Salary > 0 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
19	10:44:48	SELECT c.CompanyName, COUNT(j.JobID) AS JobCount FROM Companies c JOI...	5 row(s) returned	0.000 sec / 0.000 sec

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



The screenshot shows MySQL Workbench with a query window titled 'SQL File 3*'. The query is as follows:

```

100 select distinct a.FirstName, a.LastName from Applicants a
101 join Applications app on a.ApplicantID = app.ApplicantID
102 join Jobs j on app.JobID = j.JobID
103 join Companies c on j.CompanyID = c.CompanyID
104 where c.Location = 'New York' and a.ExperienceYears >= 3;

```

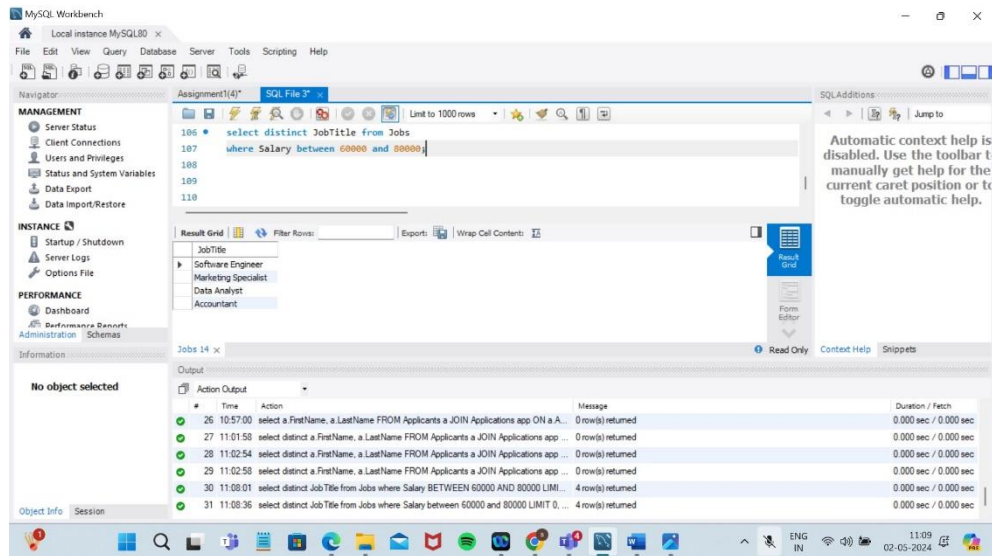
The Results grid shows the following data:

FirstName	LastName

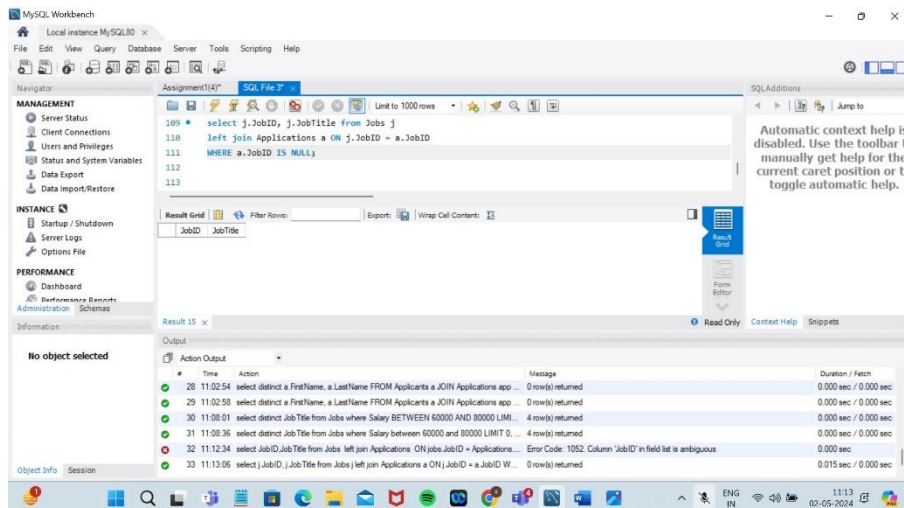
The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
19	10:44:48	SELECT c.CompanyName, COUNT(j.JobID) AS JobCount FROM Companies c JOI...	5 row(s) returned	0.000 sec / 0.000 sec
20	10:46:34	SELECT c.CompanyName, COUNT(j.JobID) FROM Companies c JOIN Jobs j ON c...	Error Code: 1248. Every derived table must have its own alias	0.000 sec
21	10:49:24	select a.FirstName, a.LastName FROM Applicants a JOIN Applications app ON a.A...	Error Code: 1054. Unknown column 'a.ExperienceYears' in 'where clause'	0.000 sec
22	10:51:23	ALTER TABLE Applicants ADD ExperienceYears INT	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.063 sec
23	10:53:53	INSERT INTO Applicants (ApplicantID, FirstName, LastName, Email, Phone, Resu...	Error Code: 1062. Duplicate entry '101' for key 'applicants.PRIMARY'	0.000 sec
24	10:56:35	UPDATE Applicants SET ExperienceYears = 4 WHERE ApplicantID = 5	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.015 sec
25	10:57:00	select a.FirstName, a.LastName FROM Applicants a JOIN Applications app ON a.A...	0 row(s) returned	0.000 sec / 0.000 sec
26	10:57:00	select a.FirstName, a.LastName FROM Applicants a JOIN Applications app ON a.A...	0 row(s) returned	0.000 sec / 0.000 sec
27	11:01:59	select distinct a.FirstName, a.LastName FROM Applicants a JOIN Applications app ...	0 row(s) returned	0.000 sec / 0.000 sec
28	11:02:54	select distinct a.FirstName, a.LastName FROM Applicants a JOIN Applications app ...	0 row(s) returned	0.000 sec / 0.000 sec
29	11:02:58	select distinct a.FirstName, a.LastName FROM Applicants a JOIN Applications app ...	0 row(s) returned	0.000 sec / 0.000 sec

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



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



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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

113 select a.FirstName, a.LastName, c.CompanyName, j.JobTitle
114 from Applicants a
115 join Applications app on a.ApplicantID = app.ApplicantID
116 join Jobs j on app.JobID = j.JobID
117 join Companies c on j.CompanyID = c.CompanyID;

```

The Result Grid shows the following data:

FirstName	LastName	CompanyName	JobTitle
Tim	David	Company 1	Software Engineer
Tim	Brook	Company 2	Marketing Specialist
Jos	Butler	Company 3	Data Analyst
Harry	Edward	Company 4	UX/UI Designer
Niall	Heron	Company 5	Accountant

The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
29	11:02:58	select distinct a.FirstName, a.LastName FROM Applicants a JOIN Applications app ...	0 row(s) returned	0.000 sec / 0.000 sec
30	11:08:01	select distinct JobTitle from Jobs where Salary BETWEEN 60000 AND 80000 LIM...	4 row(s) returned	0.000 sec / 0.000 sec
31	11:08:36	select distinct JobTitle from Jobs where Salary between 60000 and 80000 LIMIT 0 ...	4 row(s) returned	0.000 sec / 0.000 sec
32	11:12:34	select JobID, JobTitle from Jobs left join Applications ON jobs.JobID = Applications...	Error Code: 1052. Column 'JobID' in field list is ambiguous	0.000 sec
33	11:13:06	select j.JobID, j.JobTitle from Jobs j left join Applications a ON j.JobID = a.JobID W...	0 row(s) returned	0.015 sec / 0.000 sec
34	11:16:22	select a.FirstName, a.LastName, c.CompanyName, j.JobTitle from Applicants a join ...	5 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

119 select c.CompanyName, COUNT(j.JobID) as JobCount
120 from Companies c
121 left join Jobs j on c.CompanyID = j.CompanyID
122 group by c.CompanyName;

```

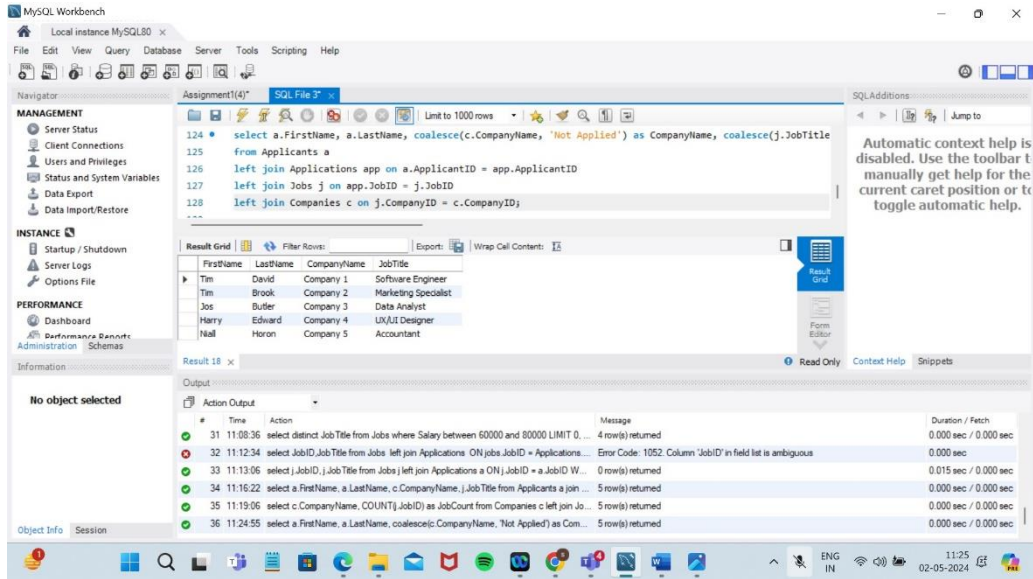
The Result Grid shows the following data:

CompanyName	JobCount
Company 1	1
Company 2	1
Company 3	1
Company 4	1
Company 5	1

The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
30	11:08:01	select distinct JobTitle from Jobs where Salary BETWEEN 60000 AND 80000 LIM...	4 row(s) returned	0.000 sec / 0.000 sec
31	11:08:36	select distinct JobTitle from Jobs where Salary between 60000 and 80000 LIMIT 0 ...	4 row(s) returned	0.000 sec / 0.000 sec
32	11:12:34	select JobID, JobTitle from Jobs left join Applications ON jobs.JobID = Applications...	Error Code: 1052. Column 'JobID' in field list is ambiguous	0.000 sec
33	11:13:06	select j.JobID, j.JobTitle from Jobs j left join Applications a ON j.JobID = a.JobID W...	0 row(s) returned	0.015 sec / 0.000 sec
34	11:16:22	select a.FirstName, a.LastName, c.CompanyName, j.JobTitle from Applicants a join ...	5 row(s) returned	0.000 sec / 0.000 sec
35	11:19:06	select c.CompanyName, COUNT(j.JobID) as JobCount from Companies c left join Jo...	5 row(s) returned	0.000 sec / 0.000 sec

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



The screenshot shows the MySQL Workbench interface with a SQL query in the editor and its results in the Result Grid.

SQL Query:

```

124 select a.FirstName, a.LastName, coalesce(c.CompanyName, 'Not Applied') as CompanyName, coalesce(j.JobTitle
125 from Applicants a
126 left join Applications app on a.ApplicantID = app.ApplicantID
127 left join Jobs j on app.JobID = j.JobID
128 left join Companies c on j.CompanyID = c.CompanyID

```

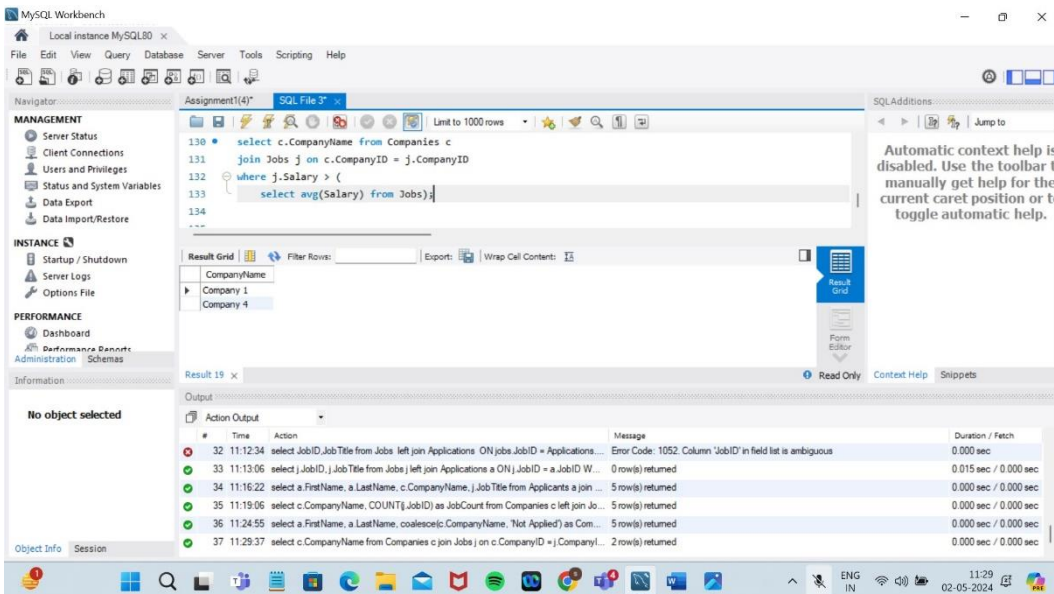
Result Grid:

FirstName	LastName	CompanyName	JobTitle
Tim	David	Company 1	Software Engineer
Tim	Brook	Company 2	Marketing Specialist
Jos	Butler	Company 3	Data Analyst
Harry	Edward	Company 4	UX/UI Designer
Niall	Heron	Company 5	Accountant

Action Output:

#	Time	Action	Message	Duration / Fetch
31	11:08:36	select distinct JobTitle from Jobs where Salary between 60000 and 80000 LIMIT 0...	4 row(s) returned	0.000 sec / 0.000 sec
32	11:12:34	select JobID, JobTitle from Jobs left join Applications ON jobs.JobID = Applications...	Error Code: 1052. Column 'JobID' in field list is ambiguous	0.000 sec
33	11:13:06	select JobID, JobTitle from Jobs left join Applications a ON JobID = a.JobID W...	0 row(s) returned	0.015 sec / 0.000 sec
34	11:16:22	select a.FirstName, a.LastName, c.CompanyName, j.JobTitle from Applicants a join ...	5 row(s) returned	0.000 sec / 0.000 sec
35	11:19:06	select c.CompanyName, COUNT(j.JobID) as JobCount from Companies c left join Jo...	5 row(s) returned	0.000 sec / 0.000 sec
36	11:24:55	select a.FirstName, a.LastName, coalesce(c.CompanyName, 'Not Applied') as Com...	5 row(s) returned	0.000 sec / 0.000 sec

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



The screenshot shows the MySQL Workbench interface with a SQL query in the editor and its results in the Result Grid.

SQL Query:

```

130 select c.CompanyName from Companies c
131 join Jobs j on c.CompanyID = j.CompanyID
132 where j.Salary > (
133 select avg(Salary) from Jobs)

```

Result Grid:

CompanyName
Company 1
Company 4

Action Output:

#	Time	Action	Message	Duration / Fetch
32	11:12:34	select JobID, JobTitle from Jobs left join Applications ON jobs.JobID = Applications...	Error Code: 1052. Column 'JobID' in field list is ambiguous	0.000 sec
33	11:13:06	select JobID, JobTitle from Jobs left join Applications a ON JobID = a.JobID W...	0 row(s) returned	0.015 sec / 0.000 sec
34	11:16:22	select a.FirstName, a.LastName, c.CompanyName, j.JobTitle from Applicants a join ...	5 row(s) returned	0.000 sec / 0.000 sec
35	11:19:06	select c.CompanyName, COUNT(j.JobID) as JobCount from Companies c left join Jo...	5 row(s) returned	0.000 sec / 0.000 sec
36	11:24:55	select a.FirstName, a.LastName, coalesce(c.CompanyName, 'Not Applied') as Com...	5 row(s) returned	0.000 sec / 0.000 sec
37	11:29:37	select c.CompanyName from Companies c join Jobs j on c.CompanyID = j.CompanyID...	2 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
181 select concat(a.FirstName, ' ', a.LastName) as FullName,
182        concat(a.City, ' ', a.State) as Location from Applicants a;
183
184
185
```

The Results grid displays the following data:

FullName	Location
Tim David	New York, A
Tim Brook	Georgia, B
Jos Butler	Chicago, C
Harry Edward	Paris, D
Nial Horan	Canada, E

The Output tab shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
50	11:38:03	UPDATE Applicants SET City = 'New York' WHERE ApplicantID = 101	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
51	11:38:05	UPDATE Applicants SET City = 'Georgia' WHERE ApplicantID = 102	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
52	11:38:07	UPDATE Applicants SET City = 'Chicago' WHERE ApplicantID = 103	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
53	11:38:10	UPDATE Applicants SET City = 'Paris' WHERE ApplicantID = 104	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
54	11:38:12	UPDATE Applicants SET City = 'Canada' WHERE ApplicantID = 105	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec
55	11:38:16	select concat(a.FirstName, ' ', a.LastName) as FullName, concat(a.City, ' ', a.St...	5 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
183
184 select * from Jobs
185 where JobTitle like '%Developer%' or JobTitle like '%Engineer%';
186
187
```

The Results grid displays the following data:

JobID	CompanyID	JobTitle	JobDescription	JobLocation	Salary	JobType	PostedDate
11	1	Software Engineer	We are looking for a skilled and experienced sof...	New York	80000.00	Full-time	2024-04-15 09:...

The Output tab shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
51	11:38:05	UPDATE Applicants SET City = 'Georgia' WHERE ApplicantID = 102	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
52	11:38:07	UPDATE Applicants SET City = 'Chicago' WHERE ApplicantID = 103	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
53	11:38:10	UPDATE Applicants SET City = 'Paris' WHERE ApplicantID = 104	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
54	11:38:12	UPDATE Applicants SET City = 'Canada' WHERE ApplicantID = 105	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec
55	11:38:16	select concat(a.FirstName, ' ', a.LastName) as FullName, concat(a.City, ' ', a.St...	5 row(s) returned	0.000 sec / 0.000 sec
56	11:41:37	select * from Jobs where JobTitle like '%Developer%' or JobTitle like '%Engineer%';	1 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

187 select a.FirstName, a.LastName, j.JobTitle
188 from Applicants a
189 left join Applications app on a.ApplicantID = app.ApplicantID
190 left join Jobs j on app.JobID = j.JobID;
191

```

The Results grid displays the following data:

FirstName	LastName	JobTitle
Tim	David	Software Engineer
Tim	Brook	Marketing Specialist
Jos	Butler	Data Analyst
Harry	Edward	UX/UI Designer
Niall	Horon	Accountant

The Output tab shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
52	11:38:07	UPDATE Applicants SET City = 'Chicago' WHERE ApplicantID = 103	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
53	11:38:10	UPDATE Applicants SET City = 'Paris' WHERE ApplicantID = 104	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
54	11:38:12	UPDATE Applicants SET City = 'Canada' WHERE ApplicantID = 105	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec
55	11:38:16	select concat(a.FirstName, ' ', a.LastName) as FullName, concat(a.City, ' ', a.St...	5 row(s) returned	0.000 sec / 0.000 sec
56	11:41:37	select * from Jobs where JobTitle like '%Developer%' or JobTitle like '%Engineer%' LI...	1 row(s) returned	0.000 sec / 0.000 sec
57	11:43:37	select a.FirstName, a.LastName, j.JobTitle from Applicants a left join Applications ap...	5 row(s) returned	0.015 sec / 0.000 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.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

192 select a.FirstName, a.LastName, c.CompanyName from Applicants a
193 join Applications app on a.ApplicantID = app.ApplicantID
194 join Jobs j on app.JobID = j.JobID
195 join Companies c on j.CompanyID = c.CompanyID
196 where c.Location = 'Newyork' and a.ExperienceYears > 2;
197

```

The Results grid displays the following data:

FirstName	LastName	CompanyName
-----------	----------	-------------

The Output tab shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
55	11:38:16	select concat(a.FirstName, ' ', a.LastName) as FullName, concat(a.City, ' ', a.St...	5 row(s) returned	0.000 sec / 0.000 sec
56	11:41:37	select * from Jobs where JobTitle like '%Developer%' or JobTitle like '%Engineer%' LI...	1 row(s) returned	0.000 sec / 0.000 sec
57	11:43:37	select a.FirstName, a.LastName, j.JobTitle from Applicants a left join Applications ap...	5 row(s) returned	0.015 sec / 0.000 sec
58	11:46:17	select a.FirstName, a.LastName, c.CompanyName from Applicants a join Applicatio...	0 row(s) returned	0.000 sec / 0.000 sec
59	11:48:53	select a.FirstName, a.LastName, c.CompanyName from Applicants a join Applicatio...	0 row(s) returned	0.000 sec / 0.000 sec
60	11:50:19	select a.FirstName, a.LastName, c.CompanyName from Applicants a join Applicatio...	0 row(s) returned	0.000 sec / 0.000 sec