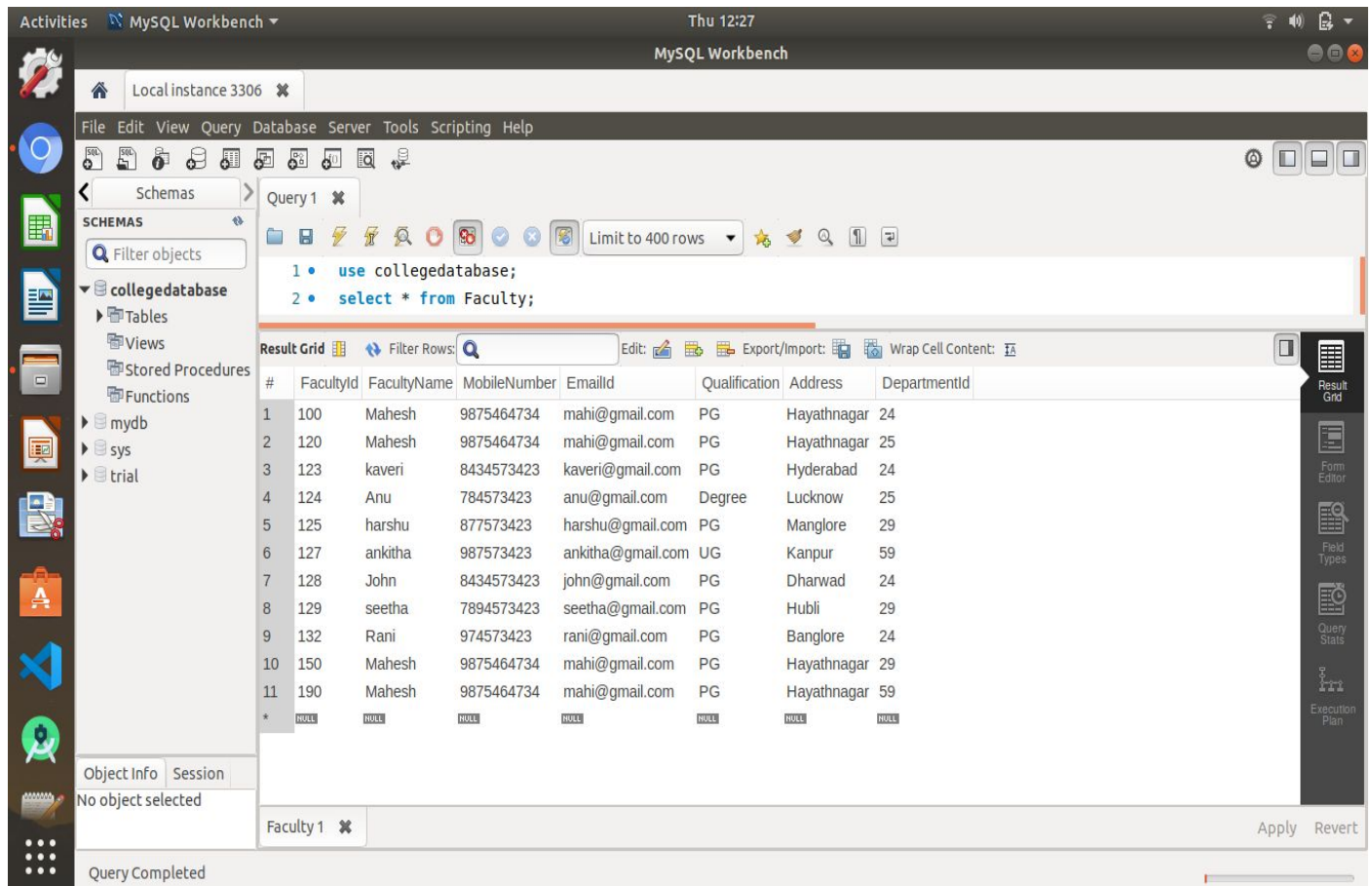


Assignment-4

Tables that are used for writing queries:

Faculty:



The screenshot displays the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'collegedatabase' selected, with 'Tables' expanded. The 'Query Editor' shows a query with two lines: 'use collegedatabase;' and 'select * from Faculty;'. The 'Result Grid' displays the following data:

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	100	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	24
2	120	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	25
3	123	kaveri	8434573423	kaveri@gmail.com	PG	Hyderabad	24
4	124	Anu	784573423	anu@gmail.com	Degree	Lucknow	25
5	125	harshu	877573423	harshu@gmail.com	PG	Manglore	29
6	127	ankitha	987573423	ankitha@gmail.com	UG	Kanpur	59
7	128	John	8434573423	john@gmail.com	PG	Dharwad	24
8	129	seetha	7894573423	seetha@gmail.com	PG	Hubli	29
9	132	Rani	974573423	rani@gmail.com	PG	Banglore	24
10	150	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	29
11	190	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	59
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The bottom status bar indicates 'Query Completed'.

Department:

The screenshot displays the MySQL Workbench interface. The top status bar shows 'Thu 12:05' and 'MySQL Workbench'. The main window is titled 'Local instance 3306'. The left sidebar shows the 'SCHEMAS' tree with 'Tables' expanded, listing 'Admissions', 'Columns', 'Indexes', 'Foreign Keys', 'Triggers', and 'Department'. The 'Department' table is selected. The central query editor shows the following SQL query:

```
1 • use collegedatabase;
2 • select * from Department;
```

The 'Result Grid' at the bottom displays the query results. The table has columns: #, DepartmentId, DepartmentName, and HOD. The results are as follows:

#	DepartmentId	DepartmentName	HOD
1	20	CSE	Rajendra
2	24	CSE	JayKumar
3	25	ECE	jay
4	29	ECE	jay
5	30	CSE	Umesh Yadav
6	31	CSE	Paddikal
7	32	ECE	KOhli
8	33	EEE	Devilliers
9	34	EEE	Rohith
10	35	EEE	Boomra
11	59	ECE	jay
*	NULL	NULL	NULL

The bottom status bar indicates 'Query Completed'.

Admissions:

Activities MySQL Workbench Thu 12:06

MySQL Workbench

Local instance 3306

File Edit View Query Database Server Tools Scripting Help

Schemas

Query 1

Limit to 400 rows

```
1 • use collegedatabase;
2 • select * from Admissions;
```

Result Grid

#	StudentId	DateOfAdmission	Fee	StudentDOB	StudentPhone	StudentEmail	StudentAddress	Admission ID
1	20	1999-09-30	456000	1987-06-23	7578534667	aish@email.com	jaipur	12
2	23	1999-09-30	456000	1987-06-23	7578534667	aish@email.com	jaipur	15
3	187	2010-01-01	70000	1999-03-22	7063578435	mona@email.com	Hyderguda	112
4	100	2015-02-21	40000	2002-05-12	8342343534	monica@email.com	Hyderabad	120
5	145	2013-06-12	50000	2001-02-02	8123456775	anvesha@email.com	Banglore	124
6	143	2016-04-11	20000	1998-12-19	8342354546	ansh@email.com	kanpur	127
7	200	2010-10-10	40000	2003-03-14	9343563456	erica@email.com	Secunderabad	156
8	123	2011-08-04	30000	2000-08-20	9435464561	srija@email.com	Lucknow	167
9	287	2010-01-01	70000	1999-03-22	7063578435	mona@email.com	Hyderguda	212
10	200	2015-02-21	40000	2002-05-12	8342343534	monica@email.com	Hyderabad	220
11	245	2013-06-12	50000	2001-02-02	8123456775	anvesha@email.com	Banglore	224
12	243	2016-04-11	20000	1998-12-19	8342354546	ansh@email.com	kanpur	227
13	300	2010-10-10	40000	2003-03-14	9343563456	erica@email.com	Secunderabad	256
14	223	2011-08-04	30000	2000-08-20	9435464561	srija@email.com	Lucknow	267

Object Info Session

Schema: collegedatabase

Admissions 21

Query Completed

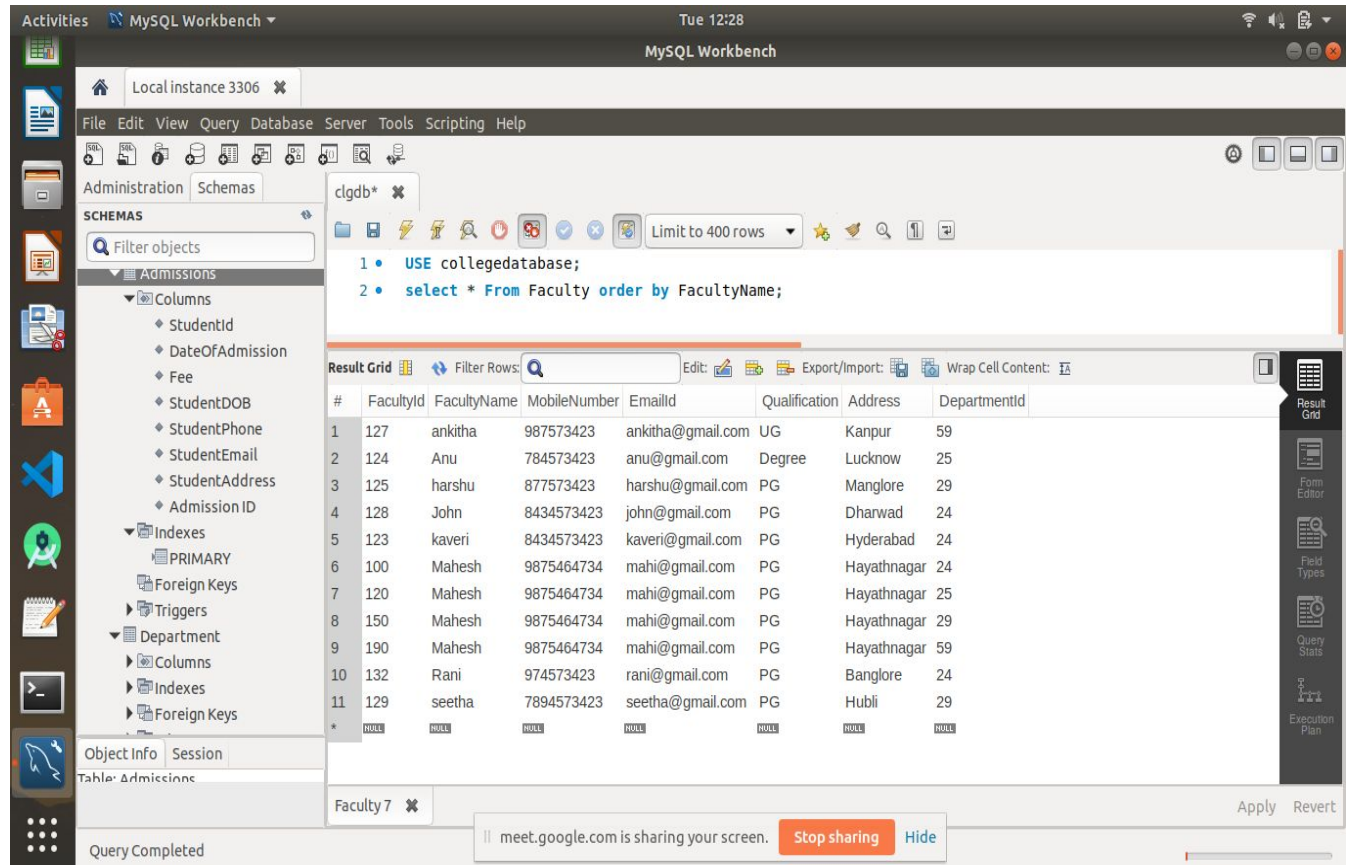
Apply Revert

Question 1: Order by clause:

Query: `Select * from Faculty order by FacultyName;`

Explanation: ordered in ascending order of faculty names

Output:



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

```
1 • USE collegedatabase;  
2 • select * From Faculty order by FacultyName;
```

The query results are displayed in the Result Grid, showing 11 rows of data. The columns are: #, FacultyId, FacultyName, MobileNumber, EmailId, Qualification, Address, and DepartmentId. The data is sorted by FacultyName in ascending order.

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	127	ankitha	987573423	ankitha@gmail.com	UG	Kanpur	59
2	124	Anu	784573423	anu@gmail.com	Degree	Lucknow	25
3	125	harshu	877573423	harshu@gmail.com	PG	Manglore	29
4	128	John	8434573423	john@gmail.com	PG	Dharwad	24
5	123	kaveri	8434573423	kaveri@gmail.com	PG	Hyderabad	24
6	100	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	24
7	120	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	25
8	150	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	29
9	190	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	59
10	132	Rani	974573423	rani@gmail.com	PG	Banglore	24
11	129	seetha	7894573423	seetha@gmail.com	PG	Hubli	29

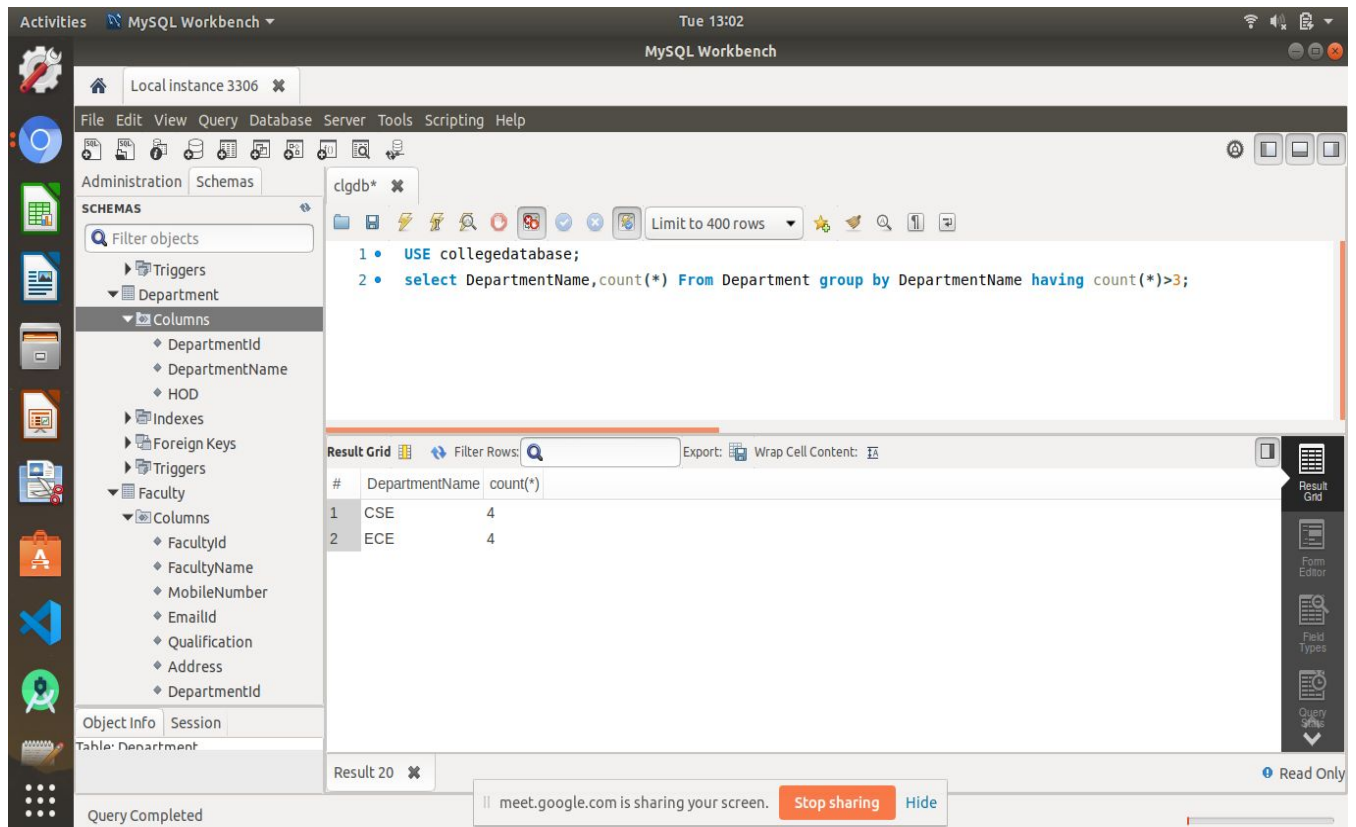
The interface also shows the Schemas pane on the left, the Object Info pane, and a status bar at the bottom indicating "Query Completed".

Question 2: Group by and having

Query: Select DepartmentName,count(*) From Department group by DepartmentName having count(*)>3;

Explanation: Group by Departmentname and those have occurrences more than 3.

Output:



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

```
1 • USE collegedatabase;
2 • select DepartmentName,count(*) From Department group by DepartmentName having count(*)>3;
```

The Results tab is selected, displaying the following data in a table:

#	DepartmentName	count(*)
1	CSE	4
2	ECE	4

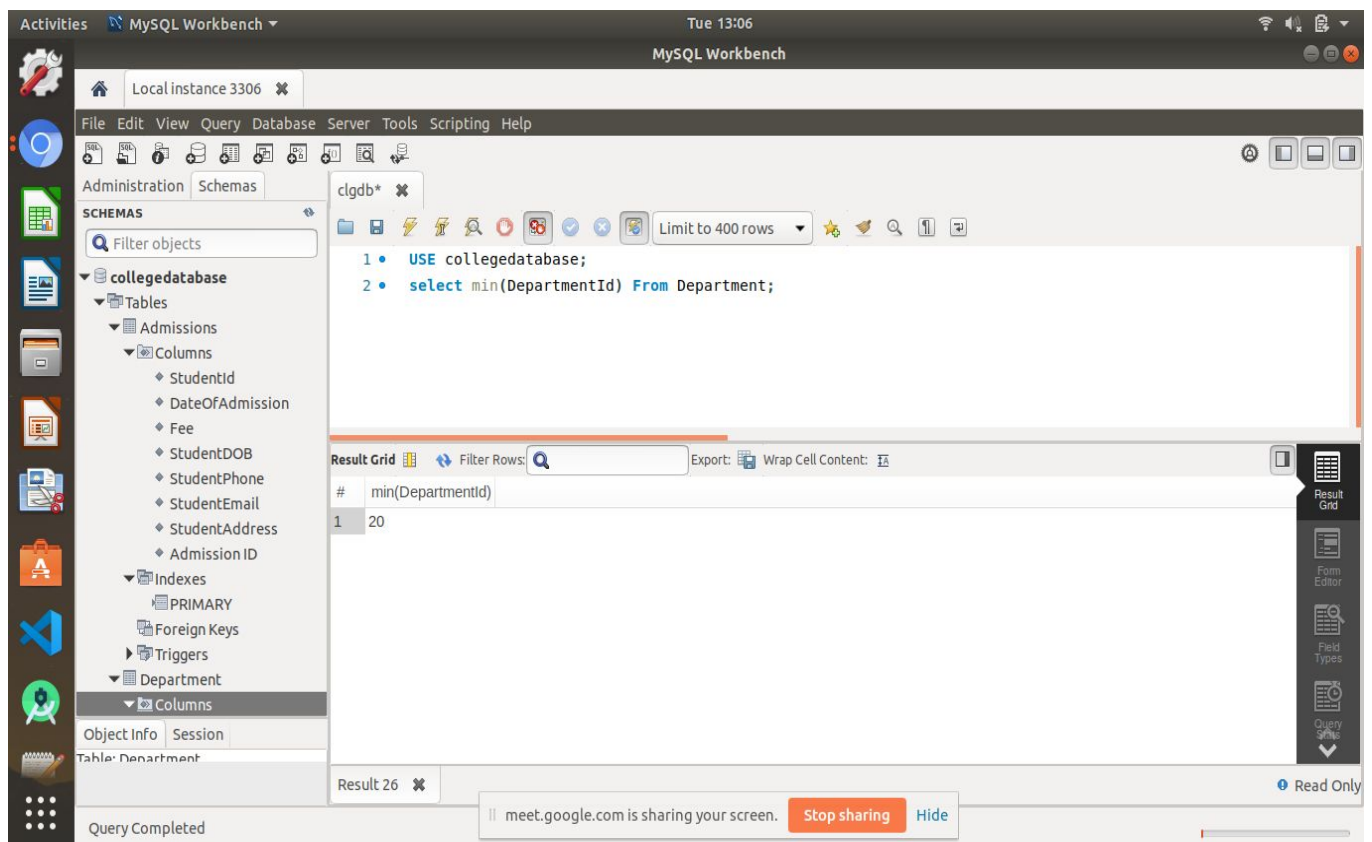
The interface also shows the Schemas pane on the left with the 'Department' table expanded, and a status bar at the bottom indicating 'Query Completed'.

Question 3: Aggregate Functions

a).Query: Select min(DepartmentId) From Department;

Explanation: gives minimum departmentId

Output:



The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'collegedatabase' expanded, showing tables like 'Admissions', 'Department', and 'Indexes'. The 'Department' table is selected. The main query editor contains the following SQL code:

```
1 • USE collegedatabase;
2 • select min(DepartmentId) From Department;
```

The 'Result Grid' at the bottom shows the output of the query:

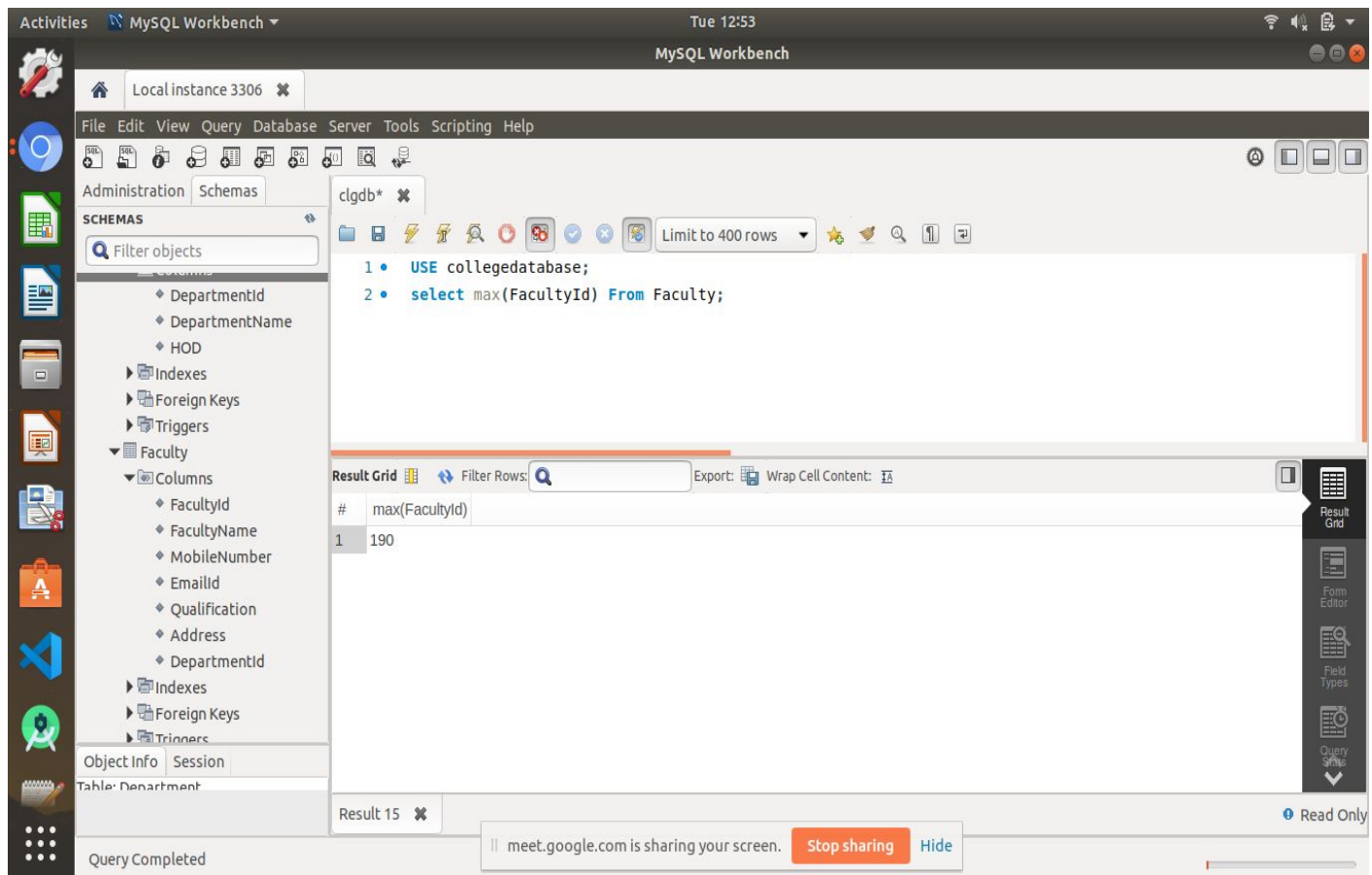
#	min(DepartmentId)
1	20

The status bar at the bottom indicates 'Query Completed'.

b).Query: `Select max(FacultyId) From Faculty;`

Explanation: gives maximum FacultyId

Output:



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

```
1 • USE collegedatabase;  
2 • select max(FacultyId) From Faculty;
```

The result grid displays the output of the query:

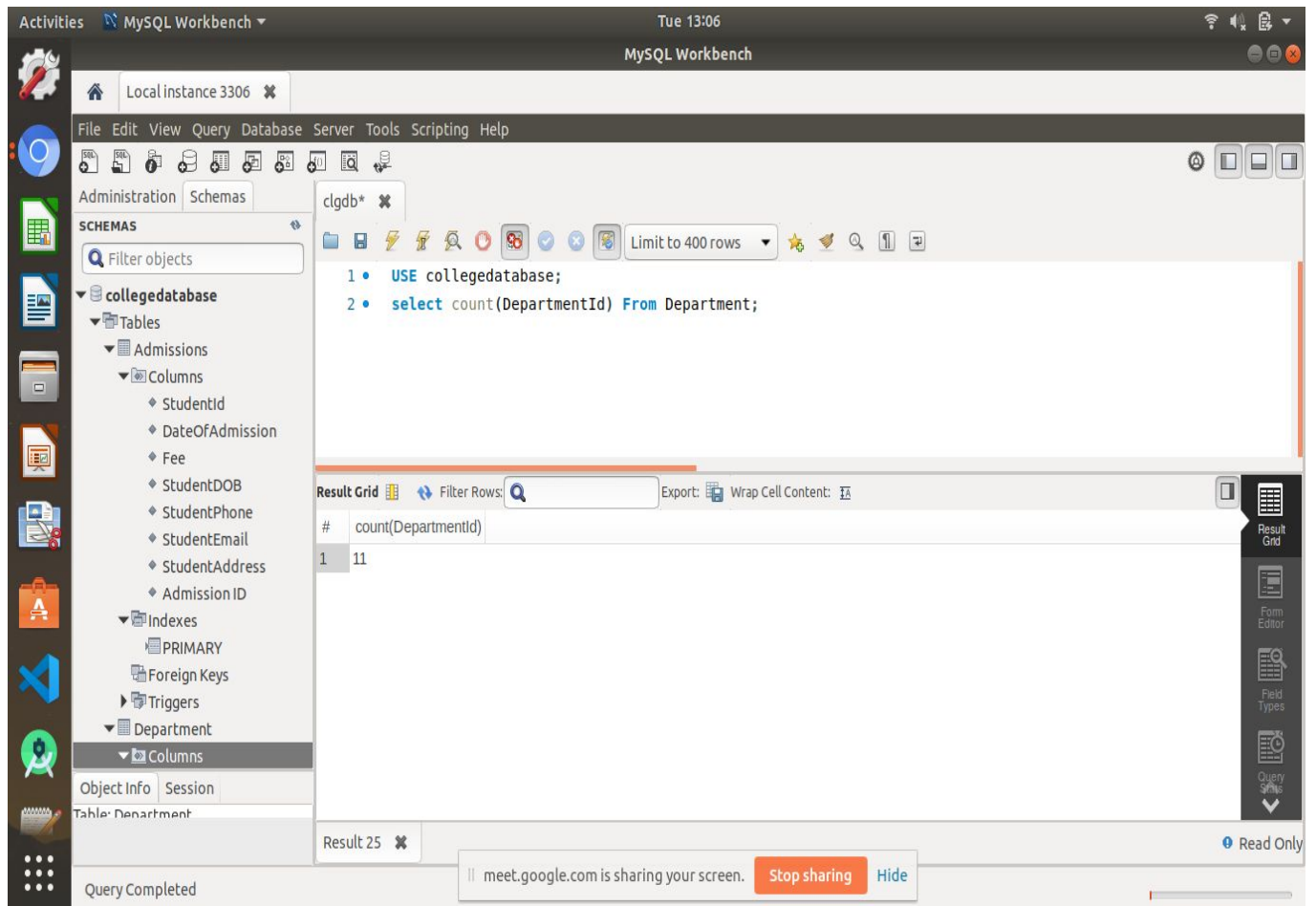
#	max(FacultyId)
1	190

The status bar at the bottom indicates "Query Completed". A notification at the bottom right states "meet.google.com is sharing your screen." with buttons for "Stop sharing" and "Hide".

c).Query: Select count(DepartmentId) From Department;

Explanation: gives number of entries in department table

Output:



The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'collegedatabase' schema with various tables and columns. The main query editor contains the following SQL code:

```
1 • USE collegedatabase;
2 • select count(DepartmentId) From Department;
```

The 'Result Grid' at the bottom shows the output of the query:

#	count(DepartmentId)
1	11

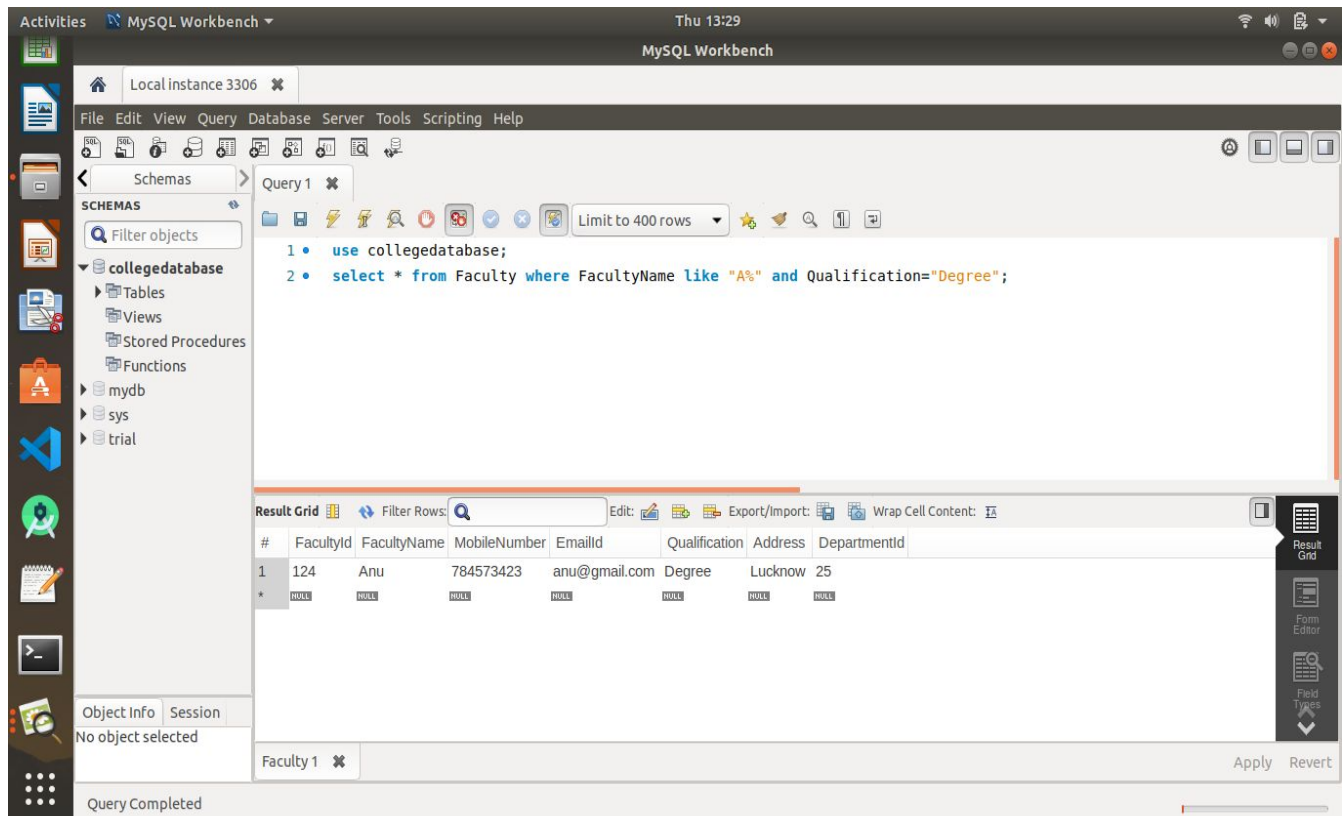
The status bar at the bottom indicates 'Query Completed'.

Question 4: Logical operators especially with LIKE

a).Query: select * from Faculty where FacultyName like "A%" and Qualification="Degree";

Explanation: FacultyName starting with letter A and qualification degree.

Output:



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

```
1 • use collegedatabase;
2 • select * from Faculty where FacultyName like "A%" and Qualification="Degree";
```

The Results tab is selected, displaying the following data in the Result Grid:

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	124	Anu	784573423	anu@gmail.com	Degree	Lucknow	25
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The status bar at the bottom indicates "Query Completed".

b).Query: select * from Admissions where Fee>30000 and StudentAddress="Bangalore";

Explanation: gives Student details whose fee is greater than 30000 and stays in Bangalore.

Output:

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

```
1 • USE collegedatabase;
2 • select * From Admissions where Fee>30000 and StudentAddress="Banglore";
3
```

The Results Grid displays the output of the query, showing 3 rows of data. The columns are: #, StudentId, DateOfAdmission, Fee, StudentDOB, StudentPhone, StudentEmail, StudentAddress, and Admission ID.

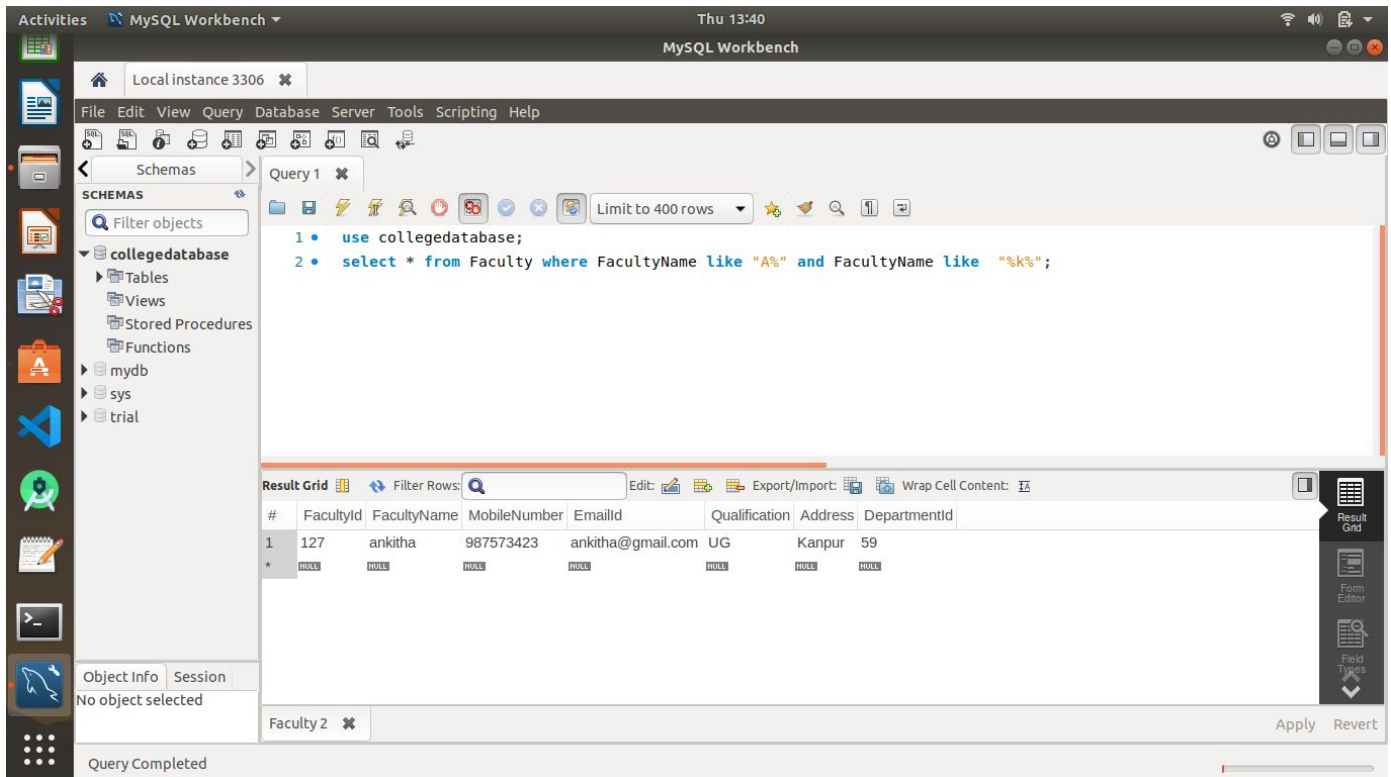
#	StudentId	DateOfAdmission	Fee	StudentDOB	StudentPhone	StudentEmail	StudentAddress	Admission ID
1	145	2013-06-12	50000	2001-02-02	8123456775	anvesha@emial.com	Banglore	124
2	245	2013-06-12	50000	2001-02-02	8123456775	anvesha@emial.com	Banglore	224
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The status bar at the bottom indicates "Query Completed" and "Admissions 30". A notification bar at the bottom right shows "meet.google.com is sharing your screen." with "Stop sharing" and "Hide" buttons.

c).Query: select * from Faculty where FacultyName like "A%" and FacultyName like "%k%";

Explanation: FacultyName starting with letter A and contains letter k in FacultyName

Output:



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

```
1 • use collegedatabase;
2 • select * from Faculty where FacultyName like "A%" and FacultyName like "%k%";
```

The Results Grid displays the output of the query. It shows a table with 8 columns: #, FacultyId, FacultyName, MobileNumber, EmailId, Qualification, Address, and DepartmentId. The first row contains the data for FacultyId 127, FacultyName 'ankitha', MobileNumber 987573423, EmailId 'ankitha@gmail.com', Qualification 'UG', Address 'Kanpur', and DepartmentId 59. A second row is marked with an asterisk and shows NULL values for all columns.

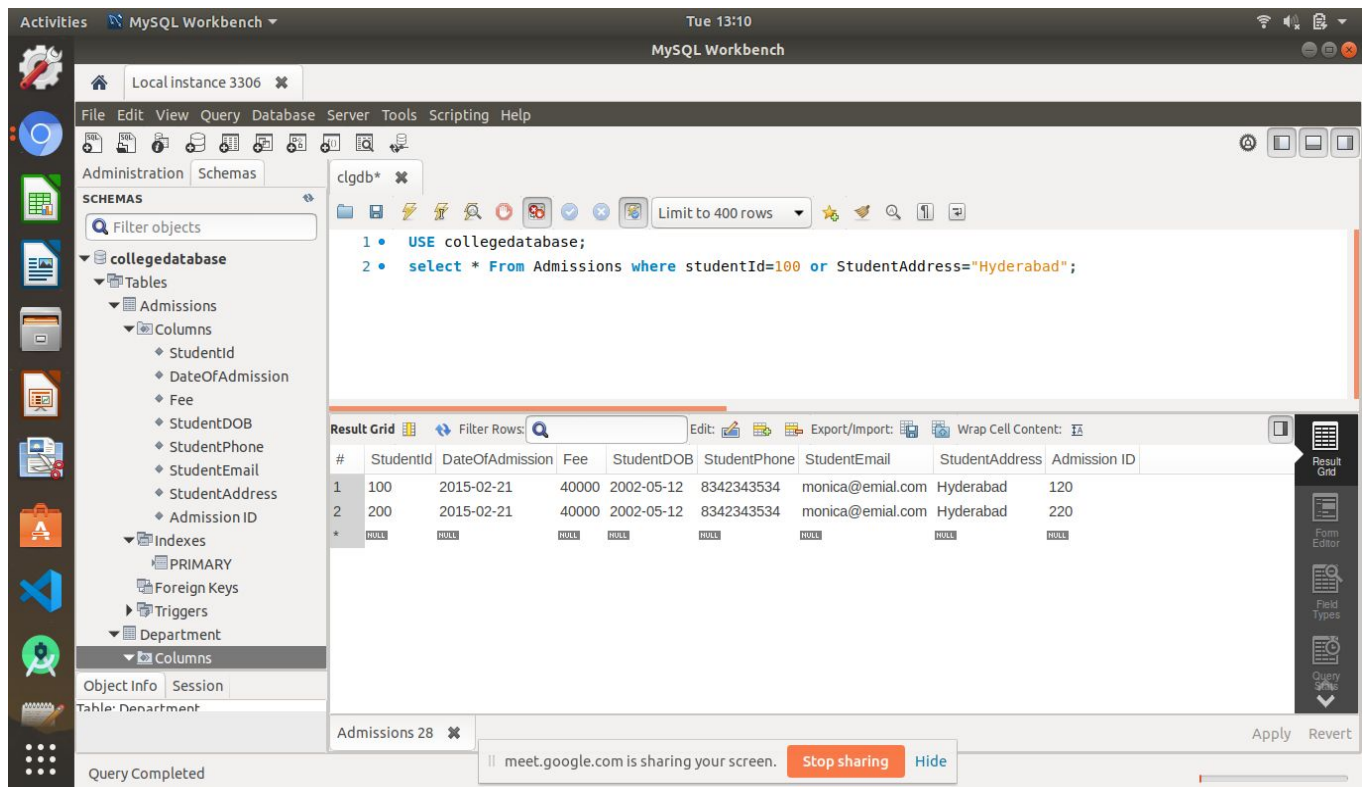
#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	127	ankitha	987573423	ankitha@gmail.com	UG	Kanpur	59
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The status bar at the bottom indicates "Query Completed".

d).Query: select * from Admissions where Fee>30000 and StudentAddress="Bangalore";

Explanation: gives Student details whose id is 100 or or who stays in Hyderabad.

Output:



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

```
1 • USE collegedatabase;
2 • select * From Admissions where studentId=100 or StudentAddress="Hyderabad";
```

The Results Grid displays the output of the query, showing 2 rows of data. The columns are: #, StudentId, DateOfAdmission, Fee, StudentDOB, StudentPhone, StudentEmail, StudentAddress, and Admission ID.

#	StudentId	DateOfAdmission	Fee	StudentDOB	StudentPhone	StudentEmail	StudentAddress	Admission ID
1	100	2015-02-21	40000	2002-05-12	8342343534	monica@emial.com	Hyderabad	120
2	200	2015-02-21	40000	2002-05-12	8342343534	monica@emial.com	Hyderabad	220

The interface also shows the Schemas pane on the left, listing the 'collegedatabase' and its tables, including 'Admissions'. The bottom status bar indicates 'Query Completed'.

e).Query: select * from Faculty
where FacultyId=
ANY(select FacultyId from Faculty where
DepartmentId=24);

Output:

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

```
1 • USE collegedatabase;
2 • SELECT *
3 FROM Faculty
4 WHERE FacultyId = ANY (SELECT FacultyId FROM Faculty WHERE DepartmentId=24);
5
```

The Results Grid shows the output of the query, listing faculty members with DepartmentId 24. The data is as follows:

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	100	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	24
2	123	kaveri	8434573423	kaveri@gmail.com	PG	Hyderabad	24
3	128	John	8434573423	john@gmail.com	PG	Dharwad	24
4	132	Rani	974573423	rani@gmail.com	PG	Banglore	24
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The interface also shows the Schemas pane on the left, the Object Info pane at the bottom, and a status bar indicating "Query Completed". A notification at the bottom states "meet.google.com is sharing your screen." with "Stop sharing" and "Hide" buttons.

f).Query: select * from Faculty
where DepartmentId=
ALL(select DepartmentId from Faculty where
DepartmentId=24);

Output:

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

```
1 • USE collegedatabase;
2 • SELECT * From Faculty where DepartmentId=all(select DepartmentId From Faculty where DepartmentId=24);
3
```

The Result Grid displays the following data:

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	100	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	24
2	123	kaveri	8434573423	kaveri@gmail.com	PG	Hyderabad	24
3	128	John	8434573423	john@gmail.com	PG	Dharwad	24
4	132	Rani	974573423	rani@gmail.com	PG	Banglore	24
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The interface also shows the Schemas pane on the left, the Object Info pane at the bottom, and a status bar indicating "Query Completed". A notification at the bottom states "meet.google.com is sharing your screen." with "Stop sharing" and "Hide" buttons.

Question 4: At least 4 Nested queries specific to your Database, out of which at least 2 should have multiple subquery.

a).Query:

```
select * from Faculty where DepartmentId=
(select DepartmentId From
Department where DepartmentId=
(select MAX(DepartmentId) from Department));
```

Output:

The screenshot shows the MySQL Workbench interface. The 'Query Editor' contains the following SQL query:

```
1 • use collegedatabase;
2 • SELECT * FROM Faculty
3 • WHERE DepartmentId = (
4 • SELECT DepartmentId FROM Department WHERE DepartmentId = (
5 • SELECT MAX(DepartmentId) FROM Department));
6
7
8
9
10
```

The 'Result Grid' displays the output of the query, showing two rows of faculty data. The first row is for FacultyId 127 (Ankitha) and the second row is for FacultyId 190 (Mahesh). Both are associated with DepartmentId 59.

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	127	ankitha	987573423	ankitha@gmail.com	UG	Kanpur	59
2	190	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	59

The interface also shows the 'SCHEMAS' panel on the left, listing the database structure, and the 'Object Info' panel at the bottom, indicating the schema is 'collegedatabase'.

b).Query:

```
select * from Department
where DepartmentId in
(select DepartmentId from Department where
DepartmentName="ECE" and DepartmentId in
(select DepartmentId from Faculty where FacultyId=
(select max(FacultyId) from Faculty)));
```

Output:

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

```
1 • use collegedatabase;
2 • SELECT * FROM Department where DepartmentId in
3 • (select DepartmentId from Department where DepartmentName="ECE" and DepartmentId in
4 • (select DepartmentId from Faculty where FacultyId=
5 • (select max(FacultyId) from Faculty)));
6
7
```

The Results Grid shows the following data:

#	DepartmentId	DepartmentName	HOD
1	59	ECE	jay

The Action Output pane shows the following log:

#	Time	Action	Message	Duration / Fetch
42	11:43:16	SELECT * FROM Faculty where FacultyId in (select max(Fa...	1 row(s) returned	0.00068 sec / 0.000...
43	11:46:58	use collegedatabase	0 row(s) affected	0.00025 sec
44	11:46:58	SELECT * FROM Department where DepartmentId in (sel...	Error Code: 1052. Column 'DepartmentId' in field ...	0.00039 sec
45	11:50:04	use collegedatabase	0 row(s) affected	0.00044 sec
46	11:50:04	SELECT * FROM Department where DepartmentId in (sel...	1 row(s) returned	0.00095 sec / 0.000...

The status bar at the bottom indicates "Query Completed".

c).Query:
select * from Faculty where FacultyId in
(select max(FacultyId) from Faculty where FacultyId <>
(select max(FacultyId) from Faculty where
DepartmentId in
(select DepartmentId from Department where
HOD="jay"))));

Output:

The screenshot displays the MySQL Workbench interface. The 'Query Editor' window contains the following SQL query:

```
1 • use collegedatabase;
2 • SELECT * FROM Faculty where FacultyId in
3   (select max(FacultyId) from Faculty where FacultyId <>
4   (select max(FacultyId) from Faculty where DepartmentId in
5   (select DepartmentId from Department where HOD="jay"))));
```

The 'Result Grid' shows the output of the query. It contains one row of data for FacultyId 150, with columns for FacultyName, MobileNumber, EmailId, Qualification, Address, and DepartmentId.

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	150	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	29

The 'Action Output' window at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
42	11:43:16	SELECT * FROM Faculty where FacultyId in (select max(Fa	1 row(s) returned	0.00068 sec / 0.000

The status bar at the bottom indicates 'Query Completed'.

d).Query:
Select * from Faculty where DepartmentId=
any(select DepartmentId From Faculty where
DepartmentId=25);

Output:

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

```
1 • USE collegedatabase;  
2 • SELECT * From Faculty where DepartmentId=  
3   any(select DepartmentId From Faculty where DepartmentId=25);  
4
```

The Results window displays the output of the query in a table format:

#	FacultyId	FacultyName	MobileNumber	EmailId	Qualification	Address	DepartmentId
1	120	Mahesh	9875464734	mahi@gmail.com	PG	Hayathnagar	25
2	124	Anu	784573423	anu@gmail.com	Degree	Lucknow	25
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The interface also shows the Schemas pane on the left, the Query toolbar, and the Status bar at the bottom indicating "Query Completed". A notification at the bottom states "meet.google.com is sharing your screen." with "Stop sharing" and "Hide" buttons.