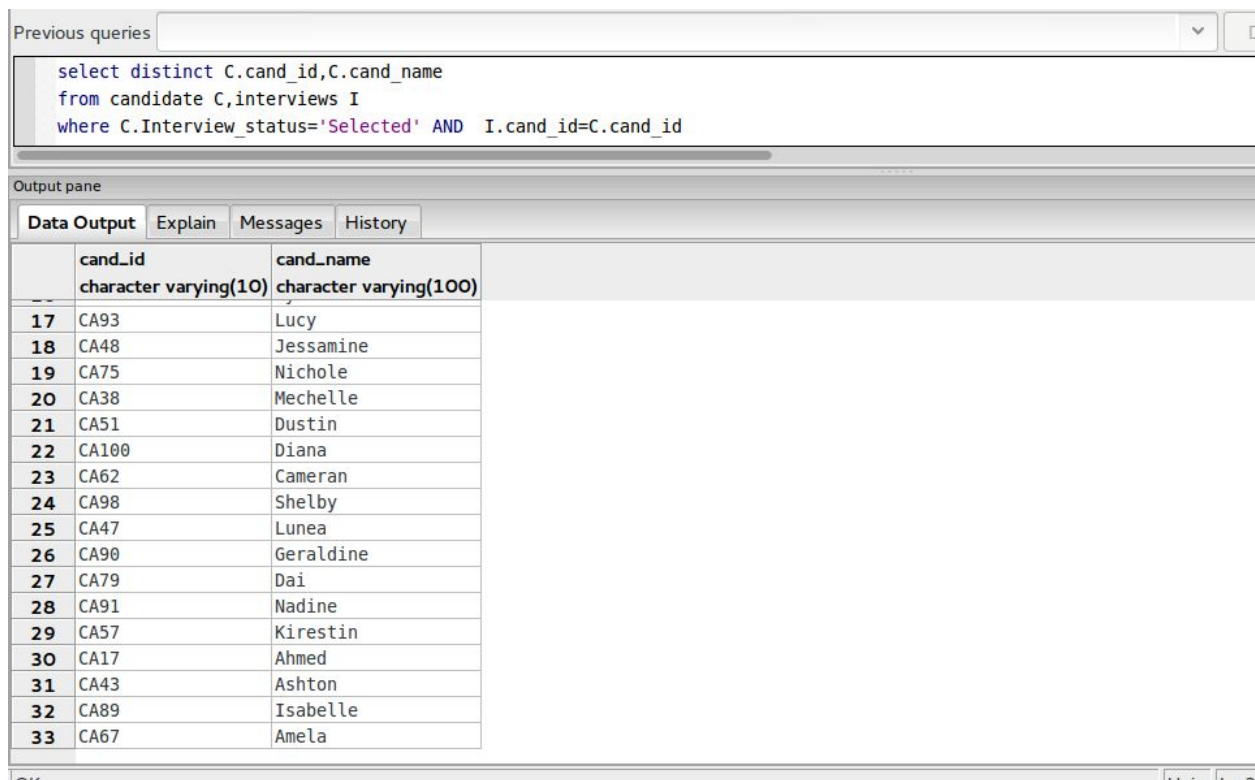


SQL_QUERIES

RECRUITMENT MANAGEMENT SYSTEM

1.Display all the candidate name ,id who are selected in interview

```
select distinct C.cand_id,C.cand_name
from candidate C,interviews I
where C.Interview_status='Selected' AND I.cand_id=C.cand_id
```



The screenshot shows a database query interface. At the top, there is a text area for the query: `select distinct C.cand_id,C.cand_name from candidate C,interviews I where C.Interview_status='Selected' AND I.cand_id=C.cand_id`. Below the query area is an "Output pane" with tabs for "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is selected, displaying a table with 33 rows. The table has two columns: "cand_id" (character varying(10)) and "cand_name" (character varying(100)). The rows are numbered 17 to 33, with the first row (17) being highlighted. The candidate names listed are Lucy, Jessamine, Nichole, Mechelle, Dustin, Diana, Cameran, Shelby, Luneia, Geraldine, Dai, Nadine, Kirestin, Ahmed, Ashton, Isabelle, and Amela.

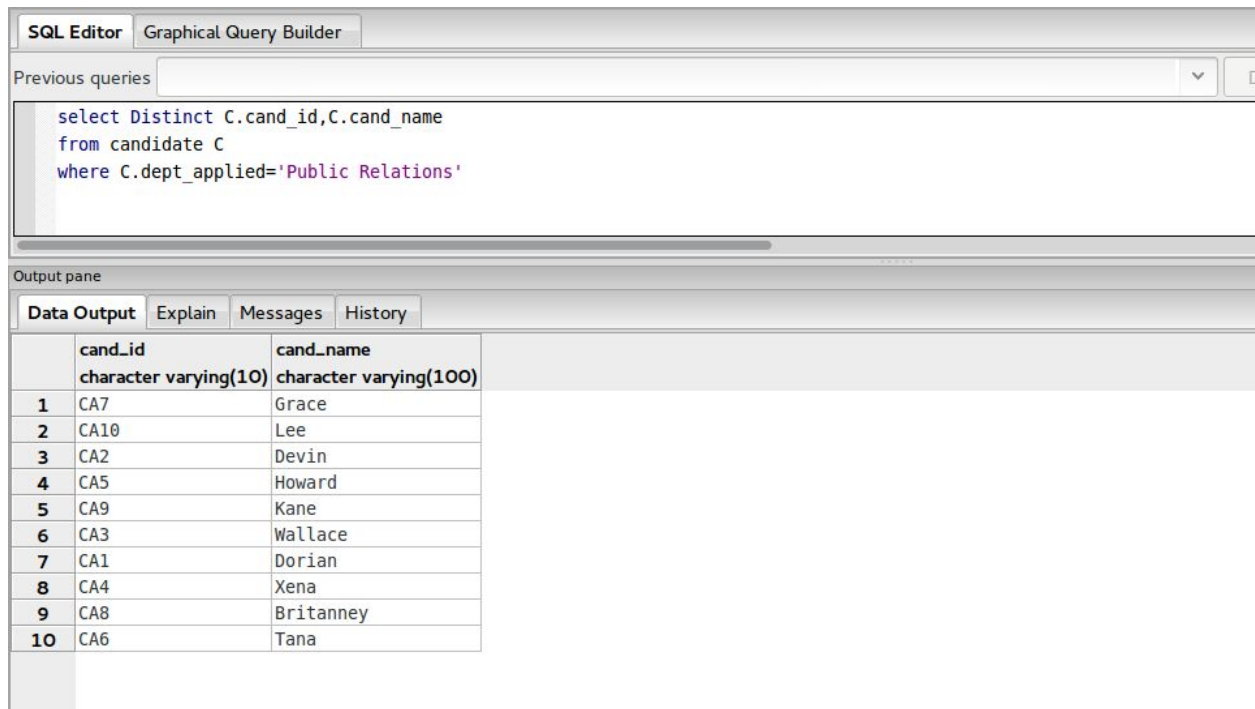
	cand_id character varying(10)	cand_name character varying(100)
17	CA93	Lucy
18	CA48	Jessamine
19	CA75	Nichole
20	CA38	Mechelle
21	CA51	Dustin
22	CA100	Diana
23	CA62	Cameran
24	CA98	Shelby
25	CA47	Luneia
26	CA90	Geraldine
27	CA79	Dai
28	CA91	Nadine
29	CA57	Kirestin
30	CA17	Ahmed
31	CA43	Ashton
32	CA89	Isabelle
33	CA67	Amela

33 rows returned

2.list all the candidate names and id who have applied for public relation department

```
select Distinct C.cand_id,C.cand_name
from candidate C
```

where C.dept_applied='Public Relations'



The screenshot shows a database query editor with two tabs: 'SQL Editor' and 'Graphical Query Builder'. The 'SQL Editor' tab is active, displaying the following SQL query:

```
select Distinct C.cand_id,C.cand_name
from candidate C
where C.dept_applied='Public Relations'
```

Below the query editor is the 'Output pane' with four tabs: 'Data Output', 'Explain', 'Messages', and 'History'. The 'Data Output' tab is active, showing a table with 10 rows of results. The table has two columns: 'cand_id' (character varying(10)) and 'cand_name' (character varying(100)).

	cand_id character varying(10)	cand_name character varying(100)
1	CA7	Grace
2	CA10	Lee
3	CA2	Devin
4	CA5	Howard
5	CA9	Kane
6	CA3	Wallace
7	CA1	Dorian
8	CA4	Xena
9	CA8	Britanney
10	CA6	Tana

10 rows returned

3.list all the details of employees in a Yahoo company

```
select distinct E.emp_id,E.emp_name
from employee E, company C
where E.c_id IN(select C.c_id
from company C
where C.c_name='Yahoo')
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select distinct E.emp_id,E.emp_name
from employee E, company C
where E.c_id IN(select C.c_id
from company C
where C.c_name='Yahoo')]
```

Output pane

Data Output Explain Messages History

	emp_id character varying(10)	emp_name character varying(100)
32	EF24	Baxter
33	EH41	Irma L. Cummings
34	EH57	Stacey F. Leonard
35	EI49	Sanford, Inez B.
36	EH69	Bell J. Ferrell
37	EH29	Brittanney W. Johnsto
38	EF44	Len
39	EF41	Kelly
40		

OK. Unix Ln 5, Col 24, Ch 132 40 rows. 11 ms

40 rows returned

4.list all types of a department of a company

```
select dept_type
from department D
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select dept_type  
from department D
```

Output pane

Data Output Explain Messages History

	dept_type character varying(30)
1	Public Relations
2	Asset Management
3	Customer Relations
4	Quality Assurance
5	Media Relations
6	Legal Department
7	Finances
8	Customer Service
9	Human Resources
10	Payroll
11	Tech Support
12	Research and Develo
13	Advertising
14	Sales and Marketing
15	Accounting
16	Strategy
17	Design
18	Operational

OK. Unix Ln 2, Col 7, Ch 20 18 rows. 12 ms

18 rows returned

5.display the locations of a chami company

```
select state,city  
from address A,company C  
where C.c_id=A.c_id AND c_name='Chami'
```

The screenshot shows a database management interface with two main tabs: "SQL Editor" and "Graphical Query Builder". The "SQL Editor" tab is active, displaying the following SQL query:

```
select state,city
from address A,company C
where C.c_id=A.c_id AND c_name='Chami'
```

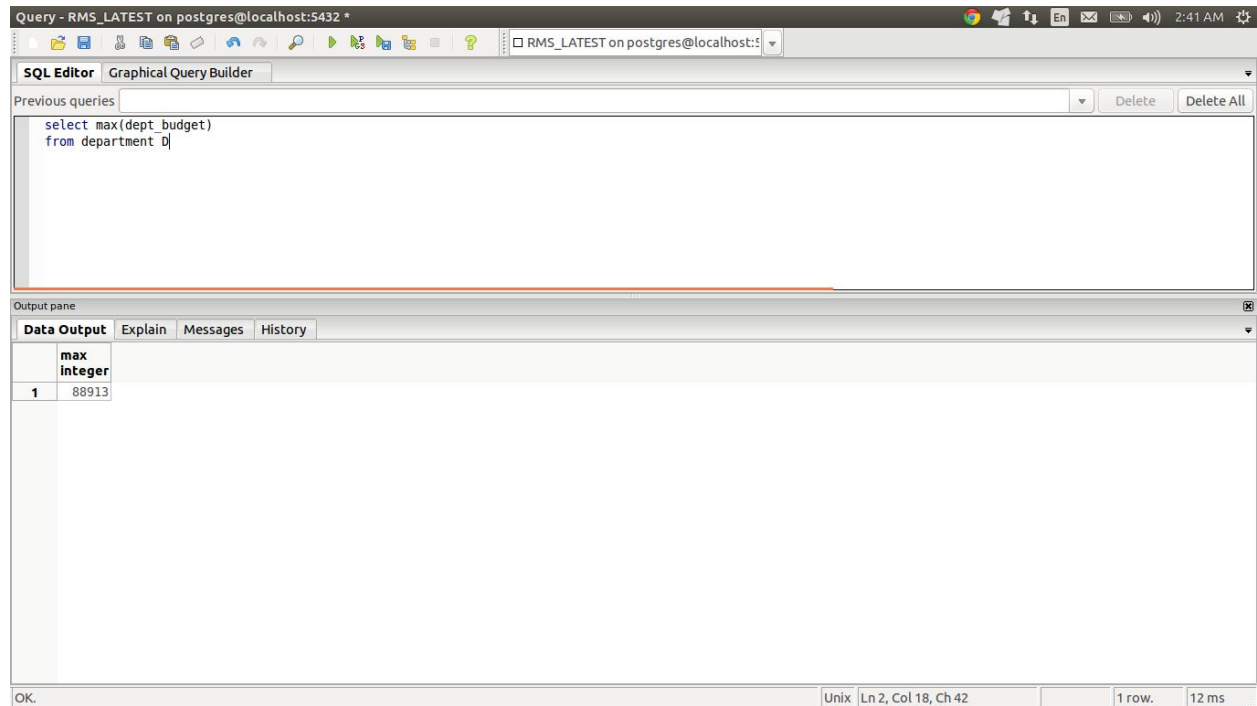
Below the query editor is the "Output pane", which has four sub-tabs: "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is selected, showing the results of the query in a table format. The table has two columns: "state" and "city", both with a data type of "character varying(100)". There are three rows of data:

	state	city
1	Puglia	Carpignano Salentino
2	OH	Columbus
3	YK	Whitehorse

3 rows returned

6.Display the maximum budget of a department from any company

```
select max(dept_budget)
from department D
```



1 row returned

7.list all the hourly employee in altavista company

```
select E.emp_name
from company C,employee E
where C.c_name='Altavista' AND C.c_id=E.c_id AND E.emp_id LIKE
'EH%'
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select E.emp_name  
from company C,employee E  
where C.c_name='Altavista' AND C.c_id=E.c_id AND E.emp_id LIKE 'EH%'
```

Output pane

Data Output Explain Messages History

	emp_name character varying(100)
1	Jacqueline
2	Helen
3	Jessamine
4	Zenia G. Winters
5	Abraham F. Zimmerman
6	Odysseus W. Decker
7	Nina C. Shepard
8	Giacomo A. Stuart
9	Cairo W. Gomez

OK. Unix Ln 3, Col 69, Ch 113 9 rows. 23 ms

9 rows returned

8.list all the full time employee in ADOBE

```
select E.emp_id,E.emp_name  
from employee E,fulltime_employee F,company C  
where E.emp_id=F.femp_id AND c_name='Adobe' And C.c_id=E.c_id
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor | Graphical Query Builder

Previous queries

```
select E.emp_id,E.emp_name
from employee E,fulltime_employee F,company C
where E.emp_id=F.femp_id AND C_name='Adobe' And C.c_id=E.c_id
```

Output pane

Data Output | Explain | Messages | History

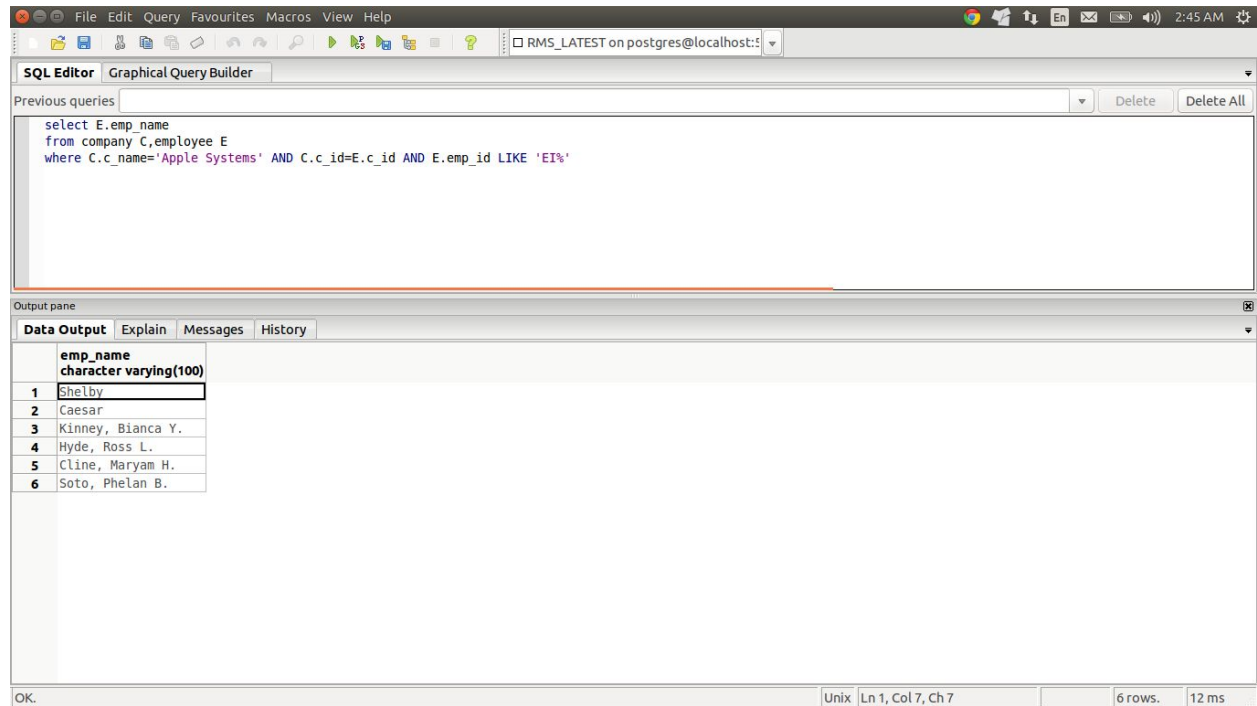
	emp_id character varying(10)	emp_name character varying(100)
1	EF5	Keaton
2	EF11	Jonah
3	EF18	Chaim
4	EF45	Caesar
5	EF55	Kevin Sargent
6	EF61	Colette Guy
7	EF68	Grace Rosa
8	EF95	Risa Maynard

OK. | Unix | Ln 1, Col 27, Ch 27 | 8 rows. | 13 ms

8 rows returned

9.list all the intern in APPLE SYSTEMS

```
select E.emp_name
from company C,employee E
where C.c_name='Apple Systems' AND C.c_id=E.c_id AND E.emp_id
LIKE 'EI%'
```

6 rows returned

10.list all the details of employee in a HR department in a cakewalk company

```
select E.emp_id,E.emp_name
from employee E
where E.dept_id IN (select D.dept_id
from department D,company C
where D.dept_type='Human Resources' AND C.c_name='Cakewalk')
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```

select E.emp_id,E.emp_name
from employee E
where E.dept_id IN (select D.dept_id
from department D,company C
where D.dept_type='Human Resources' AND C.c_name='Cakewalk')

```

Output pane

Data Output Explain Messages History

	emp_id character varying(10)	emp_name character varying(100)
14	EF99	Yael Cooke
15	EH34	Paula D. Weiss
16	EH44	Chantale P. Ferrell
17	EH54	Todd A. Pruitt
18	EH64	Laura G. Terry
19	EH74	Hakeem D. Roberson
20	EH84	Erin M. Noble
21	EH94	Claire G. Sandoval
22	EI34	Hyde, Ross L.
23	EI44	Battle, Warren E.
24	EI54	Lindsay, Neville E.
25	EI64	Whitehead, Carson D.
26	EI74	Soto, Phelan B.
27	EI84	Watkins, Hollee B.
28	EI94	Gibbs, George U.

OK. Unix Ln 5, Col 61, Ch 169 29 rows. 2 ms

29 rows returned

11.display the company id and its name which is at salt lake city location

```

select A.c_id,C.c_name
from address A,company C
where A.city='Salt Lake City' AND A.c_id=C.c_id

```


)

The screenshot shows a PostgreSQL SQL Editor window titled "Query - RMS_LATEST on postgres@localhost:5432 *". The editor contains the following SQL query:

```
SELECT E.emp_name
from employee E,fulltime_employee F
where F.femp_id=E.emp_id AND F.salary IN( select max(F1.salary)
from fulltime_employee F1) UNION
select E.emp_name
from employee E,hourly_employee H
where H.hemp_id=E.emp_id AND H.hourly_wages IN( select max(H1.hourly_wages)
from hourly_employee H1
)
```

The "Output pane" at the bottom shows the results of the query, which are 10 rows of employee names. The first column is labeled "emp_name" and "character varying(100)".

emp_name character varying(100)
1 Barbara E. Lynch
2 Ralph O. Carroll
3 Raja C. Stuart
4 Patricia
5 Jonah Z. Finch
6 Graiden T. Curtis
7 Catherine Nichols
8 Rhea
9 Iilana P. Castro
10 Henry U. Wagner

At the bottom of the window, the status bar indicates "OK.", "Unix | Ln 5, Col 18, Ch 221", "10 rows.", and "12 ms".

10 rows returned

13.display the experience of all full time employee in a lycos

```
select F.experience,F.femp_id
from fulltime_employee F,company C,employee E
where C.c_id=E.c_id AND C.c_name='Lycos' AND E.emp_id =F.femp_id
```


Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select I.iemp_id,I.stipend,E.emp_name
from intern I,employee E
where I.stipend IN (select max(stipend)
from intern I) AND E.emp_id=I.iemp_id
```

Output pane

Data Output Explain Messages History

	iemp_id character varying(10)	stipend integer	emp_name character varying(100)
1	EI64	29905	Whitehead, Carson D.

OK. Unix Ln 5, Col 7, Ch 168 1 row. 12 ms

1 row returned

15.display the company details with grades in desc

```
select max(c_grade),c_name,C.c_id
from company C
group by c_id ,c_name
order by c_grade DESC
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select max(c_grade),c_name,C.c_id
from company C
group by c_id ,c_name
order by c_grade DESC
```

Output pane

	max Integer	c_name character varying(20)	c_id character varying(10)
82	3	MAQ	C66
83	3	Google	C39
84	2	Pearson	C61
85	2	Snapdeal	C69
86	2	Amazon	C97
87	2	Pearson	C83
88	1	Pearson	C49
89	1	Finale	C55
90	1	Borland	C14
91	1	Nvidia	C63
92	1	Finale	C57
93	1	Macromedia	C93
94	1	MAQ	C52
95	1	Nvidia	C100
96	1	Finale	C79
97	1	Google	C40
98	1	Cakewalk	C3
99	1	Pearson	C77
100	1	Yahoo	C16

OK. Unix Ln 4, Col 22, Ch 93 100 rows. 12 ms

100 rows returned

16.display the top 5 company details according to desc order of grades

```
select c_grade,c_name
from company
order by c_grade DESC
limit 5
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select c_grade, c_name
from company
order by c_grade DESC
limit 5
```

Output pane

Data Output Explain Messages History

	c_grade	c_name
	integer	character varying(20)
1	10	Sibelius
2	10	Altavista
3	10	Apple Systems
4	10	Nvidia
5	10	Cakewalk

OK. Unix Ln 1, Col 7, Ch 7 5 rows. 13 ms

5 rows returned

17. display the employee id with max hours worked

```
select H.hemp_id
from hourly_employee H, hourly_wage H1
where H1.hours_worked IN (select max(H.hours_worked)
from hourly_wage H, hourly_employee H1
where H.hourly_wages=H1.hourly_wages
) AND H.hourly_wages=H1.hourly_wages
```


Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor | Graphical Query Builder

Previous queries

```
select H.hemp_id
from hourly_employee H, hourly_wage H1
where H1.hours_worked IN (select max(H.hours_worked)
from hourly_wage H, hourly_employee H1
where H.hourly_wages=H1.hourly_wages
) AND H.hourly_wages=H1.hourly_wages
```

Output pane

Data Output | Explain | Messages | History

	hemp_id character varying(10)
1	EH10
2	EH20
3	EH35
4	EH45
5	EH55
6	EH65
7	EH75
8	EH85
9	EH95

OK. Unix Ln 4, Col 38, Ch 146 9 rows. 12 ms

9 rows returned

18.how many candidates got selected by company(Google)

```
select DISTINCT C.cand_id,C.cand_name
from candidate C,company C1
where C1.c_name='Google' AND c.c_id=C1.c_id AND
c.interview_status='Selected'
```

SQL Editor Graphical Query Builder

Previous queries select max(c_grade),c_name from company C group by c_id,c_name order by c_grade DESC

```

select DISTINCT C.cand_id,C.cand_name
from candidate C,company C1
where C1.c_name='Google' AND c.c_id=C1.c_id AND c.interview_status='Selected'

```

Output pane

Data Output Explain Messages History

	cand_id character varying(10)	cand_name character varying(100)
1	CA57	Kirestin
2	CA23	Karleigh
3	CA53	Paul

3 rows returned

19.list the cand_id and cand_name selected by google or yahoo

```

select DISTINCT C.cand_id,C.cand_name
from candidate C,company C1
where C1.c_name='Google' AND C.c_id=C1.c_id AND
C.interview_status='Selected'
UNION
(select DISTINCT C.cand_id,C.cand_name
from candidate C,company C1
where C1.c_name='Yahoo' AND C.c_id=C1.c_id AND
C.interview_status='Selected')

```

Previous queries `select max(c_grade),c_name from company C group by c_id,c_name order by c_grade DESC`

```

select DISTINCT C.cand_id,C.cand_name
from candidate C,company C1
where C1.c_name='Google' AND C.c_id=C1.c_id AND C.interview_status='Selected'
UNION
(select DISTINCT C.cand_id,C.cand_name
from candidate C,company C1
where C1.c_name='Yahoo' AND C.c_id=C1.c_id AND C.interview_status='Selected')

```

Output pane

Data Output Explain Messages History

	cand_id character varying(10)	cand_name character varying(100)
1	CA23	Karleigh
2	CA4	Xena
3	CA41	Edward
4	CA53	Paul
5	CA57	Kirestin
6	CA62	Cameran
7	CA74	Kalia
8	CA96	Madeson

8 rows returned

20.candidates rejected by google company

```

select distinct C.interview_status,C.cand_id
from candidate C,company C1
where C1.c_name='Google' AND C.interview_status='Rejected' AND
C.c_id=C1.c_id

```

SQL Editor Graphical Query Builder

Previous queries: select max(c_grade),c_name from company C group by c_id ,c_name order by c_grade DESC

```
select distinct C.interview_status,C.cand_id
from candidate C,company C1
where C1.c_name='Google' AND C.interview_status='Rejected' AND C.c_id=C1.c_id
```

Output pane

Data Output Explain Messages History

	interview_status character varying(20)	cand_id character varying(10)
1	Rejected	CA40
2	Rejected	CA70
3	Rejected	CA27
4	Rejected	CA69

4 rows returned

21.candidate kept on process by Yahoo company

```
select distinct C.interview_status,C.cand_id
from candidate C,company C1
where C1.c_name='Yahoo' AND C.interview_status='On Process' AND
C.c_id=C1.c_id
```

SQL Editor Graphical Query Builder

Previous queries select max(c_grade),c_name from company C group by c_id ,c_name order by c_grade DESC

```

select distinct C.interview_status,C.cand_id
from candidate C,company C1
where C1.c_name='Yahoo' AND C.interview_status='On Process' AND C.c_id=C1.c_id

```

Output pane

Data Output Explain Messages History

	interview_status character varying(20)	cand_id character varying(10)
1	On Process	CA16
2	On Process	CA54
3	On Process	CA86

3 rows returned

22.List employee name in alphabetical order with name starting with 'A' OR 'B' in Borland

```

select DISTINCT emp_name
from employee,company C
where (emp_name LIKE 'A%' OR emp_name LIKE 'B%') AND
C.c_name='Borland' AND C.c_id=employee.c_id
order by emp_name

```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor | Graphical Query Builder

Previous queries

```
select DISTINCT emp_name
from employee,company C
where (emp_name LIKE 'A%' OR emp_name LIKE 'B%') AND C.c_name='Borland' AND C.c_id=employee.c_id
order by emp_name
```

Output pane

Data Output | Explain | Messages | History

	emp_name character varying(100)
1	Abdul
2	Alisa W. Ramos
3	Arsenio D. Drake
4	Bradley Q. Klein
5	Brianna

OK. Unix Ln 4, Col 18, Ch 165 5 rows. 12 ms

5 rows returned

23.list candidate name starting with 'L'

```
select distinct cand_name
from candidate
where cand_name LIKE 'L%'
order by cand_name
```

The screenshot shows a software interface with two main tabs: "SQL Editor" and "Graphical Query Builder". The "SQL Editor" tab is active, displaying a SQL query in a text area. Above the text area, a "Previous queries" list shows a previous query: `select max(c_grade),c_name from company C group by c_id ,c_name order by c_grade DESC`. Below the SQL Editor is an "Output pane" with four sub-tabs: "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is selected, showing a table with the results of the query. The table has two columns: "cand_name" (character varying(100)) and an implicit row index. The results are:

	cand_name character varying(100)
1	Lacota
2	Lee
3	Lucy
4	Lunea

4 rows returned

24. List companies with their addresses starting with 'H'

```
select distinct C.c_id,C.c_name
from address A,company C
where city LIKE 'H%' AND C.c_id=A.c_id
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select distinct C.c_id,C.c_name  
from address A,company C  
where city LIKE 'H%' AND C.c_id=A.c_id
```

Output pane

Data Output Explain Messages History

	c_id character varying(10)	c_name character varying(20)
1	C74	Macromedia
2	C27	Google
3	C75	Flipkart
4	C100	Nvidia

OK. Unix Ln 3, Col 39, Ch 96 4 rows. 12 ms

4 rows returned

25. List intern with names ending with 'a'

```
select E.emp_name  
from intern B,employee E  
where B.iemp_id=E.emp_id AND E.emp_name LIKE '%a'
```


Previous queries `select max(c_grade),c_name from company C group by c_id ,c`

```

select E.emp_name
  from intern B,employee E
 where B.iemp_id=E.emp_id AND E.emp_name LIKE '%a'

```

Output pane

Data Output Explain Messages History

	emp_name character varying(100)
1	Serena
2	Bertha
3	Galena
4	Cynthia
5	Zephania
6	Francesca
7	Idola
8	Carissa
9	Brianna
10	Joshua

10 rows returned

26.List candidates born before 2000.

```

select C.cand_name
  from candidate C
 where TO_CHAR(C.dob,'YYYY') <'2000'

```

<pre>select C.cand_name from candidate C where TO_CHAR(C.dob, 'YYYY') < '2000'</pre>	
Output pane	
<div> <div>Data Output</div> <div>Explain</div> <div>Messages</div> <div>History</div> </div>	
	cand_name character varying(100)
11	Molly
12	Emily
13	Sylvia
14	Lacota
15	Paul
16	Aurelia
17	Dillon
18	Cameran
19	Willow
20	Branden
21	Ava
22	Vernon
23	Baxter
24	Bert
25	Madeson
26	Shelby
27	Diana

27 rows returned

27.display employee detail who are working in altavista located in middelkerke

```
select E.emp_name
from employee E,company C,address A
where C.c_id=E.c_id AND A.city='Middelkerke' AND C.c_name='Altavista'
AND A.c_id=C.c_id
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select E.emp_name  
from employee E,company C,address A  
where C.c_id=E.c_id AND A.city='Middelkerke' AND C.c_name='Altavista' AND A.c_id=C.c_id
```

Output pane

Data Output Explain Messages History

	emp_name character varying(100)
1	Norman
2	Jacqueline
3	Idola
4	Wallace Wells
5	Zenia G. Winters
6	Nina C. Shepard
7	Dejesus, Warren T.
8	Santana, Holmes H.

OK. Unix Ln 3, Col 70, Ch 124 8 rows. 12 ms

8 rows returned

28.Count the number of locations where company borland is situated.

```
select count(*)  
from company C,address A  
where C.c_name='Borland' AND C.c_id=A.c_id
```

SQL Editor Graphical Query Builder

Previous queries select max(c_grade),c_name from company C group by c_id ,c

```

SELECT count(*)
from company C,address A
where C.c_name='Borland' AND C.c_id=A.c_id

```

Output pane

Data Output Explain Messages History

	count
	bigint
1	3

1 rows returned

29. Count the number of full time employees working in HR department who have interviewed candidates.

```

select distinct E.emp_id
from employee E,interviews I
where I.emp_id=E.emp_id AND E.emp_id IN (select F1.femp_id
from fulltime_employee F1
)

```

The screenshot shows a SQL Editor window with a toolbar at the top. The window has two tabs: "SQL Editor" and "Graphical Query Builder". The "SQL Editor" tab is active, displaying a query. Above the query editor, there is a "Previous queries" section showing a previous query: `select max(c_grade),c_name from company C group by c_id ,c_name order by c_grade DESC`. The current query in the editor is:

```
SELECT distinct E.emp_id
from employee E,interviews I
where I.emp_id=E.emp_id AND E.emp_id IN (select F1.femp_id
from fulltime_employee F1)
```

Below the query editor is an "Output pane" with four tabs: "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is selected, showing the results of the query in a table format:

	emp_id character varying(10)
1	EF39
2	EF19
3	EF9
4	EF29
5	EF40
6	EF8
7	EF49

7 rows returned

30.Display corresponding number of candidates interviewed by HR employees and HR employees must be full time employees. (order should be displayed in ascending)

```
select count(*),l.emp_id
from interviews l
group by l.emp_id
having l.emp_id LIKE 'EF%'
order by count(*)
```

SQL Editor Graphical Query Builder

Previous queries `select max(c_grade),c_name from company C group by c_id ,c_name`

```

SELECT count(*),I.emp_id
from interviews I
group by I.emp_id
having I.emp_id LIKE 'EF%'
order by count(*)

```

Output pane

Data Output Explain Messages History

	count bigint	emp_id character varying(10)
1	1	EF40
2	1	EF8
3	12	EF49
4	13	EF39
5	13	EF19
6	14	EF9
7	14	EF29

7 rows returned

31.Display corresponding number of employees working in respective departments in Microsoft.(ascending order of their counts)

```

select count(*),D.dept_type
from employee E,company C,department D
where C.c_name='Microsoft' AND C.c_id=E.c_id AND E.dept_id=D.dept_id
group by D.dept_type
order by count(*)

```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor | Graphical Query Builder

Previous queries

```
select count(*),D.dept_type
from employee E,company C,department D
where C.c_name='Microsoft' AND C.c_id=E.c_id AND E.dept_id=D.dept_id
group by D.dept_type
order by count(*)
```

Output pane

Data Output | Explain | Messages | History

	count	dept_type
	bigint	character varying(30)
1	7	Customer Relations
2	8	Public Relations
3	15	Payroll

OK. Unix Ln 6, Col 18, Ch 177 3 rows. 12 ms

3 rows returned

32.Display count of rejections from each company

```
select count(*),C.c_name
from candidate A,company C
where A.c_id=C.c_id AND A.interview_status='Rejected'
group by C.c_name
order by count(*)
```

Previous queries `select max(c_grade),c_name from company C group by c_id ,c_name order by`

```

SELECT  count(*),C.c_name
from  candidate A,company C
where A.c_id=C.c_id AND A.interview_status='Rejected'
group by C.c_name
order by count(*)

```

Output pane

Data Output Explain Messages History

	count bigint	c_name character varying(20)
1	1	Lavasoft
2	1	Apple Systems
3	2	Borland
4	2	Sibelius
5	2	Adobe
6	2	Chami
7	2	Altavista
8	4	Microsoft
9	4	Google
10	5	Lycos
11	5	Yahoo
12	6	Cakewalk

12 rows returned

33.Display names of most experienced employee (experience >5)

```

select E.emp_name
from employee E,fulltime_employee F
where F.femp_id=E.emp_id AND F.experience >5

```


Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor Graphical Query Builder

Previous queries

```
select E.emp_name
from employee E,fulltime_employee F
where F.femp_id=E.emp_id AND F.experience >5
```

Output pane

Data Output Explain Messages History

	emp_name character varying(100)
11	Palmer
12	Edan
13	Kelly
14	Jeremy
15	Jerome Dawson
16	Risa Forbes
17	Colette Guy
18	Olivia Tillman
19	Karleigh Estes
20	Catherine Nichols
21	Dakota Pope
22	Sonya Kinney
23	Anne Burris
24	Gannon Cherry
25	Hollee Avila
26	Hayden Dominguez
27	Risa Maynard
28	Herman Tate

OK. Unix Ln 3, Col 45, Ch 99 28 rows. 12 ms

28 rows returned

34.How many candiates have appllied for company having grade greater than 8 and got rejected. (Be detailed in showing results like displaying even the company details)

```
select C.cand_name,A.c_name,A.c_id,L.city
from candidate C,company A,address L
where A.c_grade >8 AND C.interview_status IN
(select  C1.interview_status
from candidate C1
where C1.interview_status='Rejected')
AND C.c_id=A.c_id AND L.c_id=C.c_id
```

SQL Editor Graphical Query Builder

Previous queries: select max(c._grade),c._name from company C group by c._id,c._name order by c._grade DESC

```

SELECT C.cand_name,A.c_name,A.c_id,L.city
from candidate C,company A,address L
where A.c_grade >8 AND C.interview_status IN
(select C1.interview_status
from candidate C1
where C1.interview_status='Rejected') AND C.c_id=A.c_id AND L.c_id=C.c_id

```

Output pane

Data Output Explain Messages History

	cand_name character varying(100)	c_name character varying(20)	c_id character varying(10)	city character varying(100)
1	Margaret	Yahoo	C4	Meeuwen
2	Kane	Apple Systems	C9	San Rafael
3	Alfonso	Altavista	C15	Middelkerke
4	Shoshana	Altavista	C15	Middelkerke
5	Dylan	Cakewalk	C17	Reading
6	Zephania	Cakewalk	C17	Reading
7	Ciaran	Google	C27	Hamburg
8	Quentin	Microsoft	C30	Quesada
9	Branden	Sibelius	C36	Lafayette
10	Adena	Sibelius	C36	Lafayette

10 rows returned

35.Display corresponding number of candidates interviewed by HR employees

```

select count(*),l.emp_id
from interviews l
group by l.emp_id
order by count(*)

```

SQL Editor

Graphical Query Builder

Previous queries

select max(c_grade),c_name from company C group by c_id ,c_name

SELECT count(*),I.emp_id

from interviews I

group by I.emp_id

order by count(*)

Output pane

Data Output

Explain

Messages

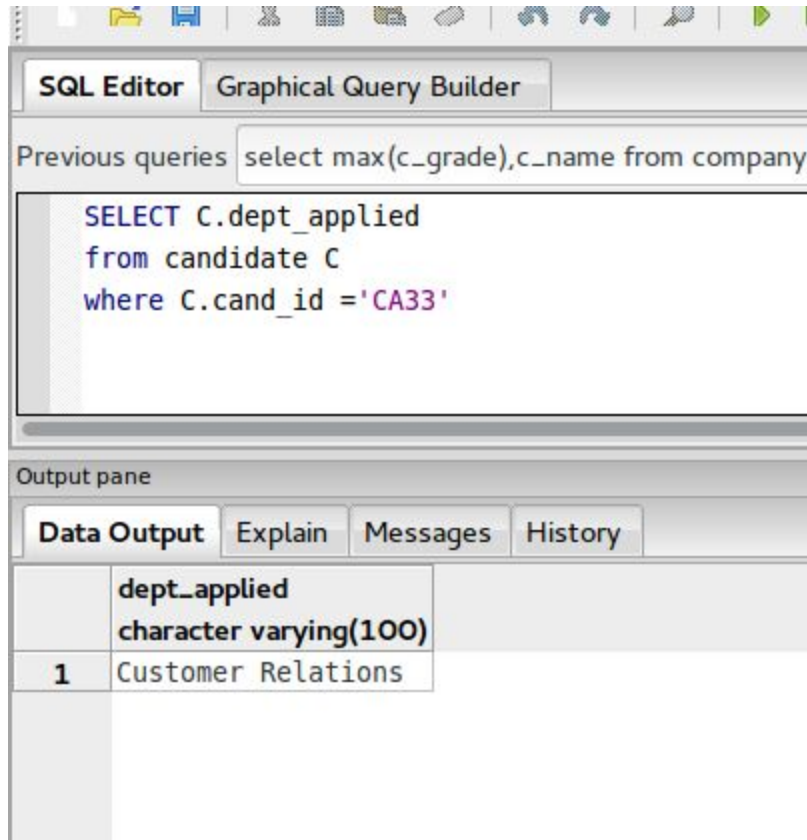
History

	count bigint	emp_id character varying(10)
1	1	EF8
2	1	EF40
3	12	EF49
4	13	EF39
5	13	EF19
6	14	EF29
7	14	EF9
8	16	EH19
9	16	EH9

9 rows returned

36.display the department applied by "CA33"

```
select C.dept_applied
from candidate C
where C.cand_id ='CA33'
```



1 row returned

37. List the departments where there are some employees who work only for 5 hours. (display with employee names)

```
select D.dept_type,E.emp_name,E.emp_id
from department D,employee E,hourly_wage H,hourly_employee H1
where E.dept_id=D.dept_id AND H1.hemp_id=E.emp_id AND
H.hours_worked IN
(select H.hours_worked
from hourly_wage H
where H.hours_worked=5)
AND H1.hourly_wages=H.hourly_wages
```

Query - RMS_LATEST on postgres@localhost:5432 *

SQL Editor | Graphical Query Builder

Previous queries

```

from department D,employee E,hourly_wage H,hourly_employee H1
where E.dept_id=D.dept_id AND H1.hemp_id=E.emp_id AND H.hours_worked IN
(select H.hours_worked
from hourly_wage H
where H.hours_worked=5)
AND H1.hourly_wages=H.hourly_wages

```

Output pane

Data Output | Explain | Messages | History

	dept_type character varying(30)	emp_name character varying(100)	emp_id character varying(10)
1	Media Relations	Jacqueline	EH5
2	Media Relations	Iona	EH15
3	Media Relations	Kylan	EH25
4	Media Relations	Reece W. Hensley	EH30
5	Media Relations	Zenia G. Winters	EH40
6	Media Relations	Camilla A. Burch	EH50
7	Media Relations	Xanthus C. Charles	EH60
8	Media Relations	Nichole R. Flynn	EH70
9	Media Relations	Nina C. Shepard	EH80
10	Media Relations	Quin X. Daniels	EH90
11	Media Relations	Martena G. Cunningha	EH100

OK. Unix Ln 8, Col 1, Ch 309 11 rows. 13 ms

11 rows returned

38. How many candidates did HR employee with ID 'EF19' interviewed

```

select count(*),l.emp_id
from interviews l
where l.emp_id='EF19'
group by l.emp_id
order by count(*)

```

The screenshot shows an SQL Editor window with a toolbar at the top. Below the toolbar are two tabs: "SQL Editor" and "Graphical Query Builder". The "SQL Editor" tab is active, displaying a query in a text area. Above the text area is a "Previous queries" section showing a previous query: "select max(c_grade),c_name from company C group by c_id ,c_name order by c_grade". Below the text area is an "Output pane" with four tabs: "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is active, showing a table with the results of the query.

```

SELECT count(*),I.emp_id
from interviews I
where I.emp_id='EF19'
group by I.emp_id
order by count(*)

```

	count bigint	emp_id character varying(10)
1	13	EF19

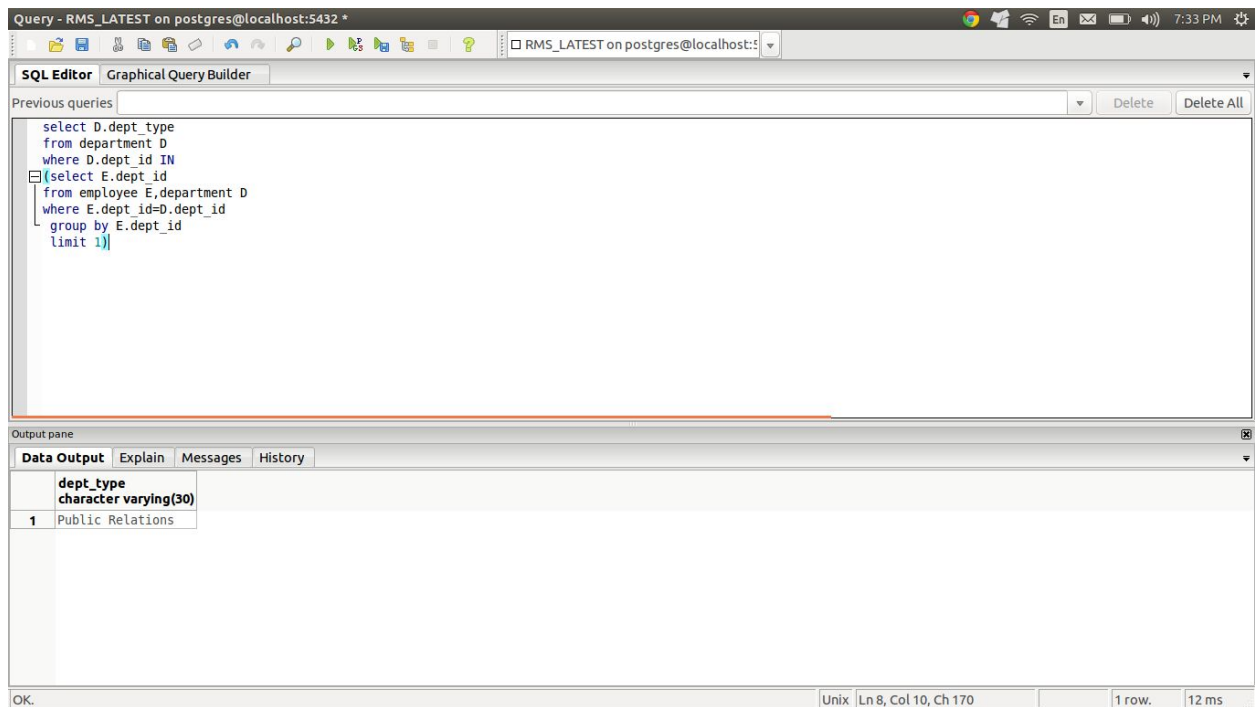
1 row returned

39) Name the Department having maximum number of employees.

```

select D.dept_type
from department D
where D.dept_id IN
(select E.dept_id
from employee E,department D
where E.dept_id=D.dept_id
group by E.dept_id
limit 1)

```

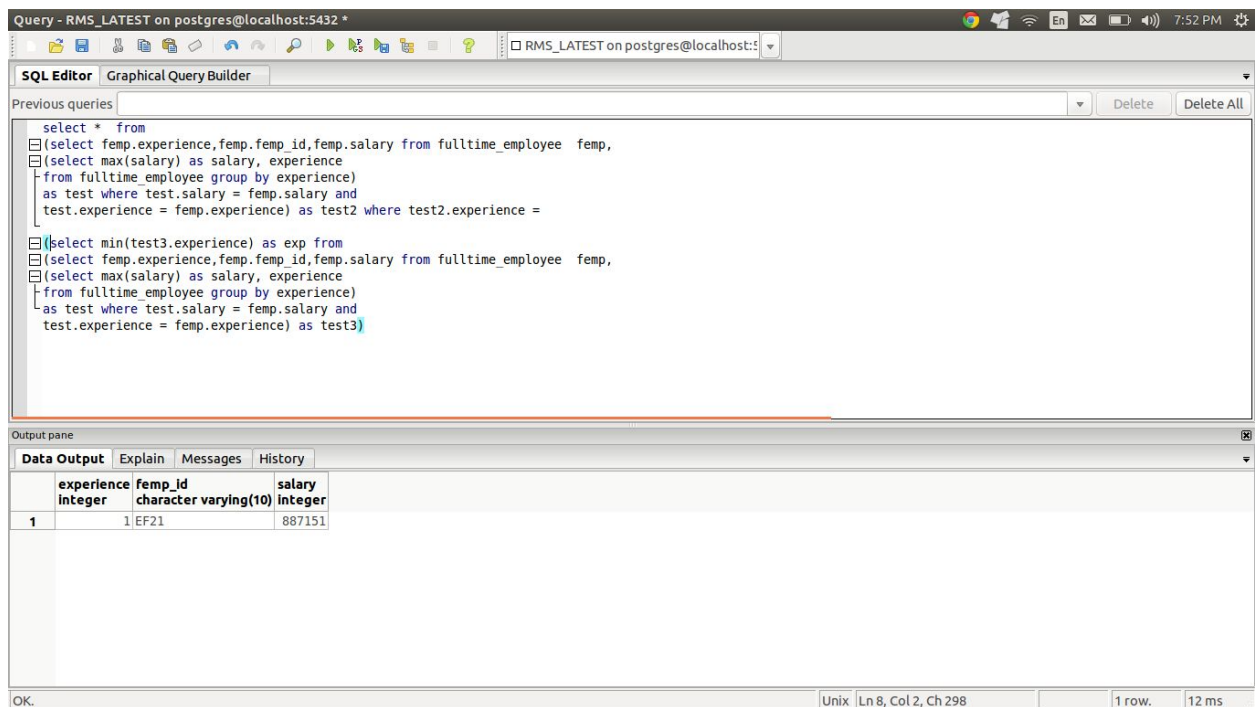


1 row returned

40. Pick a talented employee (Talented in the sense less experience but having high salary)

```
select * from
(select femp.experience,femp.femp_id,femp.salary from fulltime_employee
femp,
(select max(salary) as salary, experience
from fulltime_employee group by experience)
as test where test.salary = femp.salary and
test.experience = femp.experience) as test2 where test2.experience =
(select min(test3.experience) as exp from
```

(select femp.experience,femp.femp_id,femp.salary from fulltime_employee
femp,
(select max(salary) as salary, experience
from fulltime_employee group by experience)
as test where test.salary = femp.salary and
test.experience = femp.experience) as test3)



The screenshot shows a PostgreSQL SQL Editor window titled "Query - RMS_LATEST on postgres@localhost:5432 *". The editor contains a complex SQL query with multiple subqueries and joins. The query is as follows:

```
select * from
(select femp.experience,femp.femp_id,femp.salary from fulltime_employee femp,
(select max(salary) as salary, experience
from fulltime_employee group by experience)
as test where test.salary = femp.salary and
test.experience = femp.experience) as test2 where test2.experience =
(select min(test3.experience) as exp from
(select femp.experience,femp.femp_id,femp.salary from fulltime_employee femp,
(select max(salary) as salary, experience
from fulltime_employee group by experience)
as test where test.salary = femp.salary and
test.experience = femp.experience) as test3)
```

The output pane shows the results of the query, which is a single row with three columns: experience, femp_id, and salary. The values are 1, EF21, and 887151 respectively.

experience	femp_id	salary
Integer	character varying(10)	Integer
1	1 EF21	887151

The status bar at the bottom indicates "OK.", "Unix", "Ln 8, Col 2, Ch 298", "1 row.", and "12 ms".

1 row returned