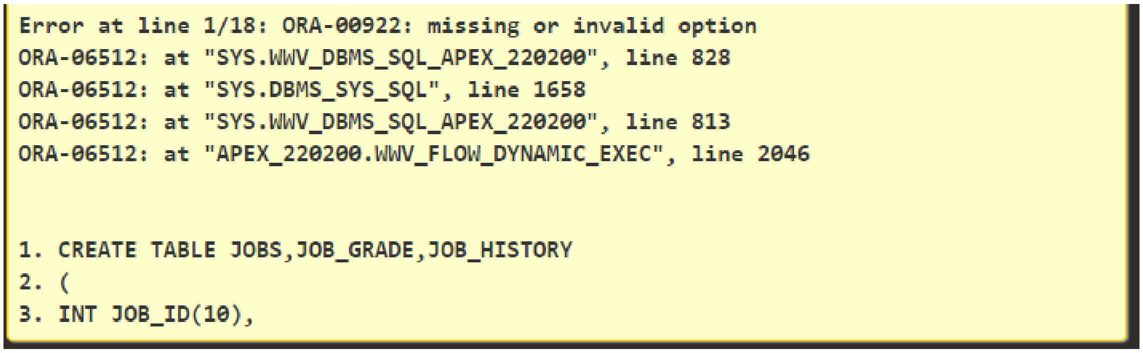
DBMS – PRACTISE 11_1

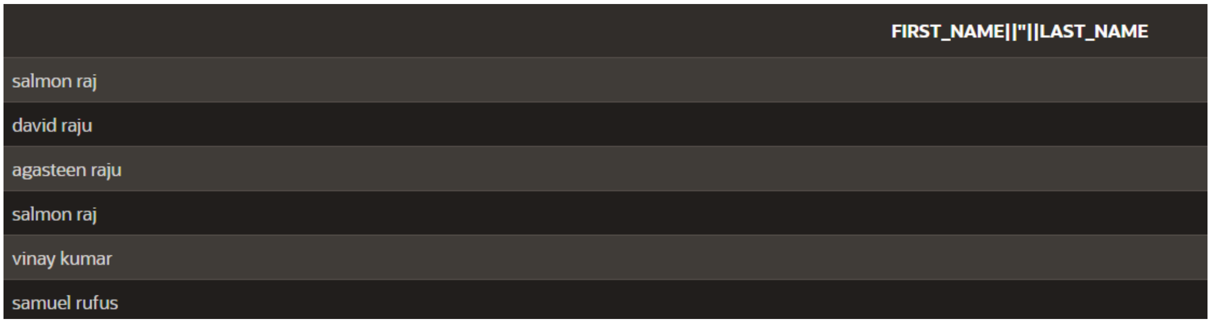
Problem: −Create a list of all tables whose first two characters in the name of the table is JO −The tables must be owned by the current Oracle User.

```
CREATE TABLE JOBS,JOB_GRADE,JOB_HISTORY

(

INT JOB_ID(10),

NAME VARCHAR(100),

);
```



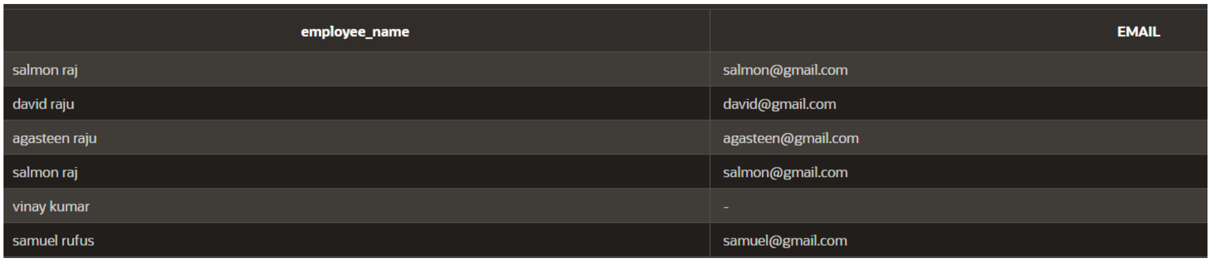
```
Error at line 1/18: ORA-00922: missing or invalid option
ORA-06512: at "SYS.WWV_DBMS_SQL_APEX_220200", line 828
ORA-06512: at "SYS.DBMS_SYS_SQL", line 1658
ORA-06512: at "SYS.WWV_DBMS_SQL_APEX_220200", line 813
ORA-06512: at "APEX_220200.WWV_FLOW_DYNAMIC_EXEC", line 2046


1. CREATE TABLE JOBS,JOB_GRADE,JOB_HISTORY
2. (
3. INT JOB_ID(10),
```

```
CREATE TABLE Emp

(

   first_name VARCHAR(100),

   last_name VARCHAR(100));

DESC Emp;


INSERT INTO Emp(first_name,last_name) VALUES ('vinay ','kumar');

INSERT INTO Emp(first_name,last_name) VALUES ('david','raju');

INSERT INTO Emp(first_name,last_name) VALUES ('samuel','rufus');

INSERT INTO Emp(first_name,last_name) VALUES ('agasteen','raju');

INSERT INTO Emp(first_name,last_name) VALUES ('salmon','raj');


SELECT first_name||' '||last_name

FROM Emp;
```



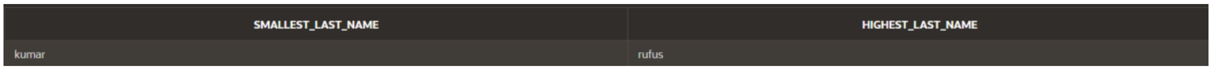
FIRST_NAME||"||LAST_NAME
salmon raj
david raju
agasteen raju
salmon raj
vinay kumar
samuel rufus

```
SELECT first_name ||'  '||last_name AS "employee_name",Email

FROM Emp;
```



employee_name	EMAIL
salmon raj	salmon@gmail.com
david raju	david@gmail.com
agasteen raju	agasteen@gmail.com
salmon raj	salmon@gmail.com
vinay kumar	-
samuel rufus	samuel@gmail.com

```
SELECT

   MIN(last_name) AS smallest_last_name,

   MAX(last_name) AS highest_last_name

FROM

   employees;
```

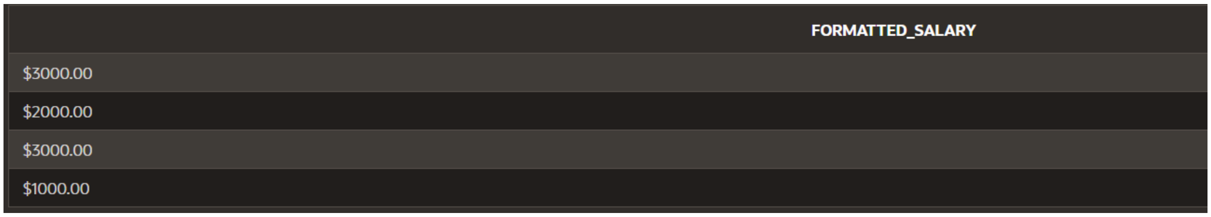


SMALLEST_LAST_NAME	HIGHEST_LAST_NAME
kumar	rufus

```
UPDATE Emp

SET salary=700

WHERE first_name='vinay';

UPDATE Emp

SET salary=100

WHERE first_name='david';
```

```
UPDATE Emp

SET salary=1000

WHERE first_name='samuel';

UPDATE Emp

SET salary=2000

WHERE first_name='agasteen';

UPDATE Emp

SET salary=3000

WHERE first_name='salmon';
```

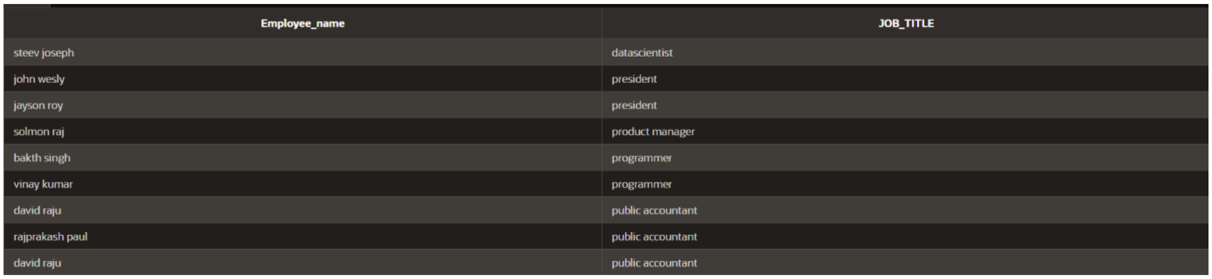
```
SELECT TO_CHAR(salary, '$9999.99') AS formatted_salary

FROM Emp

WHERE salary BETWEEN 700 AND 3000;
```



FORMATTED_SALARY
$3000.00
$2000.00
$3000.00
$1000.00

Problem: − Create a list of every employee and his related job title sorted by job_title

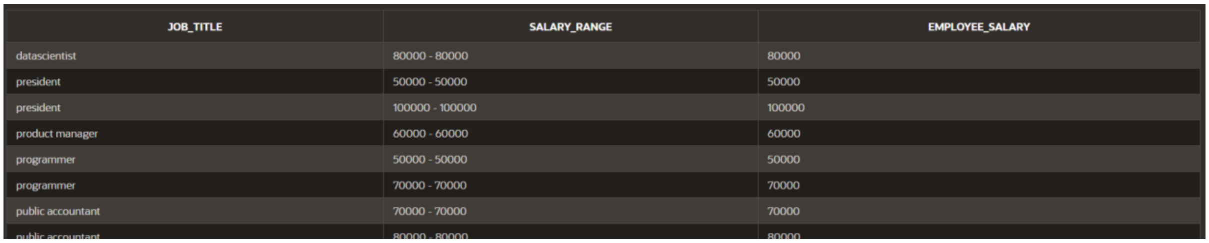
```
SELECT first_name||'  '||last_name AS "Employee_name",job_title

FROM Employ

ORDER BY job_title;
```



Employee_name	JOB_TITLE
steev joseph	datascientist
john wesly	president
jayson roy	president
solmon raj	product manager
bakth singh	programmer
vinay kumar	programmer
david raju	public accountant
rajprakash paul	public accountant
david raju	public accountant

```
SELECT job_title,MIN(salary) || ' - ' || MAX(salary) AS salary_range,salary AS employee_salary
```

```
FROM Employ

GROUP BY job_title, salary

ORDER BY job_title, salary;
```



JOB_TITLE	SALARY_RANGE	EMPLOYEE_SALARY
datascientist	80000 - 80000	80000
president	50000 - 50000	50000
president	100000 - 100000	100000
product manager	60000 - 60000	60000
programmer	50000 - 50000	50000
programmer	70000 - 70000	70000
public accountant	70000 - 70000	70000
public accountant	80000 - 80000	80000

CSA 0563 – DBMS

Cretaing a table by using CREATE

```
CREATE  TABLE STUDENT2

(

name VARCHAR(10),

std_id  NUMBER(10),

age  NUMBER(2),

course VARCHAR(10),

date_of_registration DATE

);
```

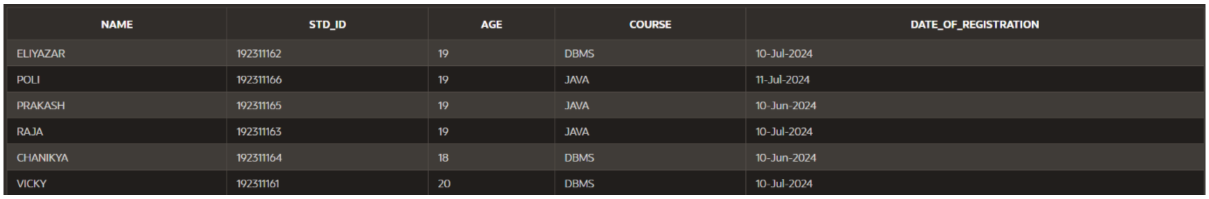
Inserting values into table by Using INSERT keyword

```
INSERT INTO STUDENT2 (name,std_id,age,course,date_of_registration)

VALUES ('ELIYAZAR',192311162,19,'DBMS',TO_DATE('10-07-2024','DD-MM-YYYY'));

INSERT INTO STUDENT2 (name,std_id,age,course,date_of_registration)

VALUES ('CHANIKYA',192311164,18,'DBMS',TO_DATE('10-06-2024','DD-MM-YYYY'));

INSERT INTO STUDENT2 (name,std_id,age,course,date_of_registration)

VALUES('CHANIKYA',192311164,18,'DBMS',TO_DATE('10-06-2024','DD-MM-YYYY'));

INSERT INTO STUDENT2 (name,std_id,age,course,date_of_registration)

VALUES('PRAKASH',192311165,19,'JAVA',TO_DATE('10-06-2024','DD-MM-YYYY'));
```

```
INSERT INTO STUDENT2 (name,std_id,age,course,date_of_registration)

VALUES('POLI',192311166,19,'JAVA',TO_DATE('11-07-2024','DD-MM-YYYY'));

INSERT INTO STUDENT2 (name,std_id,age,course,date_of_registration)

VALUES('VICKY',192311161,20,'DBMS',TO_DATE('10-07-2024','DD-MM-YYYY'));
```

Displaying the tablevalues by using SELECT

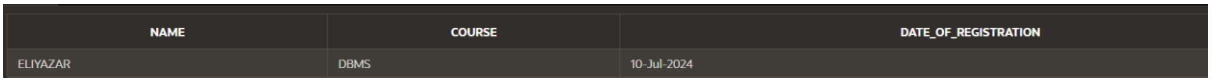
```
SELECT *

FROM STUDENT2;
```



NAME	STD_ID	AGE	COURSE	DATE_OF_REGISTRATION
ELIYAZAR	192311162	19	DBMS	10-Jul-2024
POLI	192311166	19	JAVA	11-Jul-2024
PRAKASH	192311165	19	JAVA	10-Jun-2024
RAJA	192311163	19	JAVA	10-Jul-2024
CHANIKYA	192311164	18	DBMS	10-Jun-2024
VICKY	192311161	20	DBMS	10-Jul-2024

Appluying condition **WHERE**

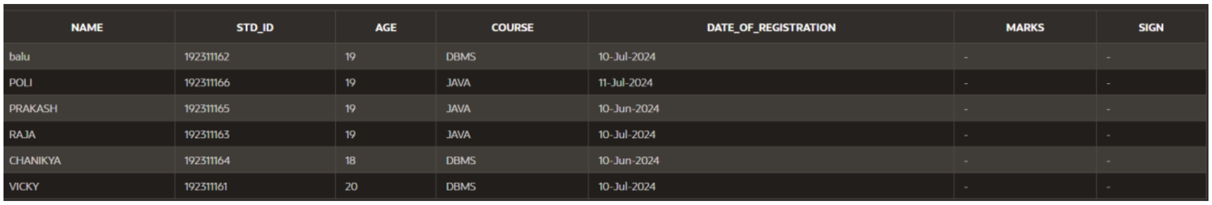
```
SELECT  name ,course,date_of_registration

FROM STUDENT2

WHERE std_id = 192311162;
```



NAME	COURSE	DATE_OF_REGISTRATION
ELIYAZAR	DBMS	10-Jul-2024

Adding another column to a table by using ALTER and ADD

```
ALTER TABLE STUDENT2

ADD marks FLOAT;
```

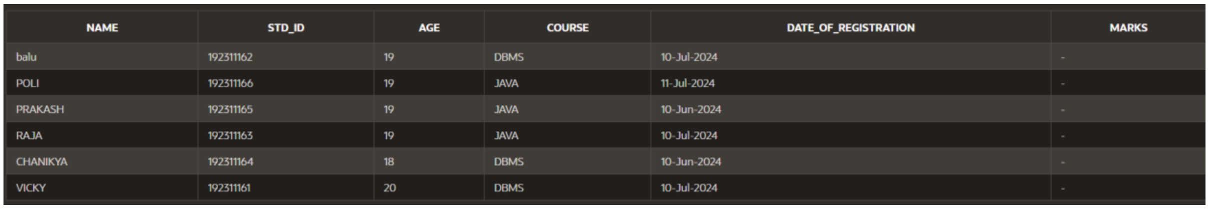


NAME	STD_ID	AGE	COURSE	DATE_OF_REGISTRATION	MARKS	SIGN
balu	192311162	19	DBMS	10-Jul-2024	-	-
POLI	192311166	19	JAVA	11-Jul-2024	-	-
PRAKASH	192311165	19	JAVA	10-Jun-2024	-	-
RAJA	192311163	19	JAVA	10-Jul-2024	-	-
CHANIKYA	192311164	18	DBMS	10-Jun-2024	-	-
VICKY	192311161	20	DBMS	10-Jul-2024	-	-

Deleting column name by using ALTER and DROP

```
ALTER TABLE STUDENT2
```

DROP (sign);



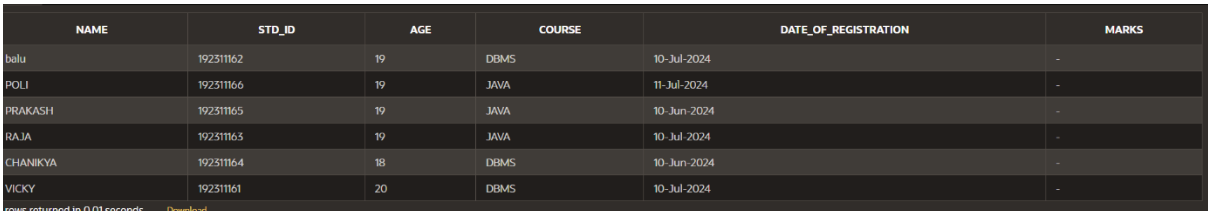
NAME	STD_ID	AGE	COURSE	DATE_OF_REGISTRATION	MARKS
balu	192311162	19	DBMS	10-Jul-2024	-
POLI	192311166	19	JAVA	11-Jul-2024	-
PRAKASH	192311165	19	JAVA	10-Jun-2024	-
RAJA	192311163	19	JAVA	10-Jul-2024	-
CHANIKYA	192311164	18	DBMS	10-Jun-2024	-
VICKY	192311161	20	DBMS	10-Jul-2024	-

Updating the specified row in a table by using UPDATE,SET

UPDATE STUDENT2

SET name=’balu’

WHERE std_id=192311162;



NAME	STD_ID	AGE	COURSE	DATE_OF_REGISTRATION	MARKS
balu	192311162	19	DBMS	10-Jul-2024	-
POLI	192311166	19	JAVA	11-Jul-2024	-
PRAKASH	192311165	19	JAVA	10-Jun-2024	-
RAJA	192311163	19	JAVA	10-Jul-2024	-
CHANIKYA	192311164	18	DBMS	10-Jun-2024	-
VICKY	192311161	20	DBMS	10-Jul-2024	-

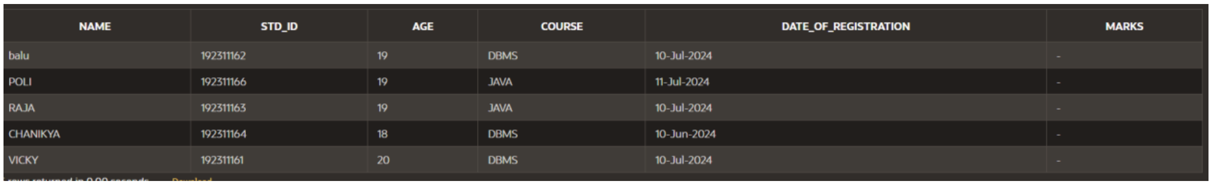
rows returned in 0.01 seconds Download

Deleting specifed row in a table by using DELETE keyword

DELETE FROM STUDENT2

WHERE name=’PRAKASH’

SELECT * FROM STUDENT2



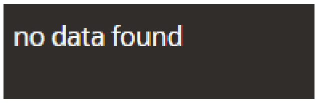
NAME	STD_ID	AGE	COURSE	DATE_OF_REGISTRATION	MARKS
balu	192311162	19	DBMS	10-Jul-2024	-
POLI	192311166	19	JAVA	11-Jul-2024	-
RAJA	192311163	19	JAVA	10-Jul-2024	-
CHANIKYA	192311164	18	DBMS	10-Jun-2024	-
VICKY	192311161	20	DBMS	10-Jul-2024	-

rows returned in 0.00 seconds Download

Deleting all the rows in a table by using TRUNCATE keyword

TRUNCATE TABLE STUDENT2;

SELECT * FROM STUDENT2;

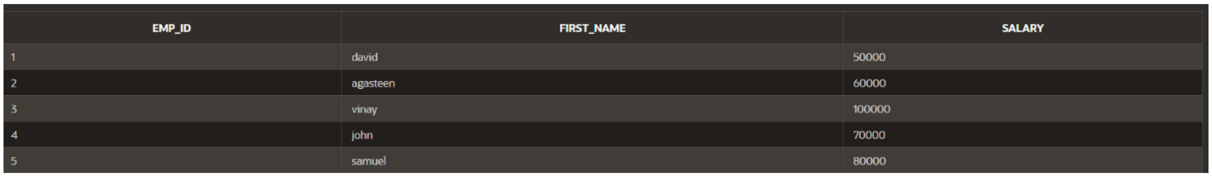


no data found

Selecting only some columns in the table by using SELECT

SELECT emp_id,first_name,salary

FROM Employee;



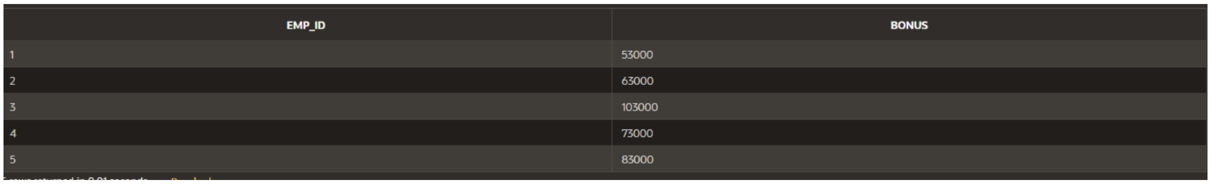
EMP_ID	FIRST_NAME	SALARY
1	david	50000
2	agasteen	60000
3	vinay	100000
4	john	70000
5	samuel	80000

ARTHMETIC OPERATORS :

(+-*/)

SELECT emp_id, salary+3000

FROM Employee;



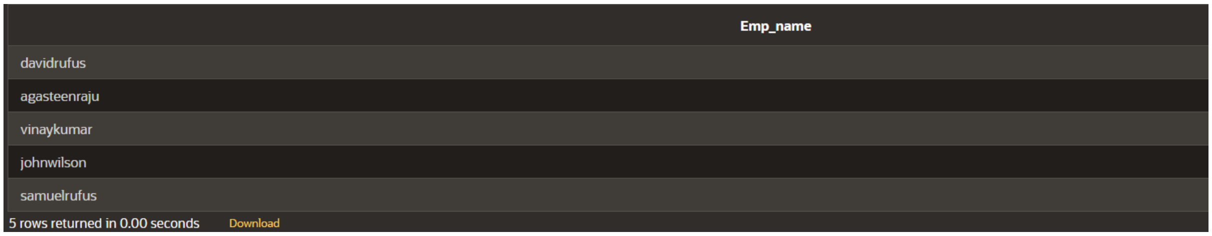
EMP_ID	BONUS
1	53000
2	63000
3	103000
4	73000
5	83000

5 rows returned in 0.01seconds Download

CONCATENATE two column name as one name by using AS :

SELECT first_name | | last_name AS Emp_name

FROM Employee;



Emp_name
davidrufus
agasteenraju
vinaykumar
johnwilson
samuelrufus

5 rows returned in 0.00 seconds Download

DESCRIBE Keyword

desc keyword

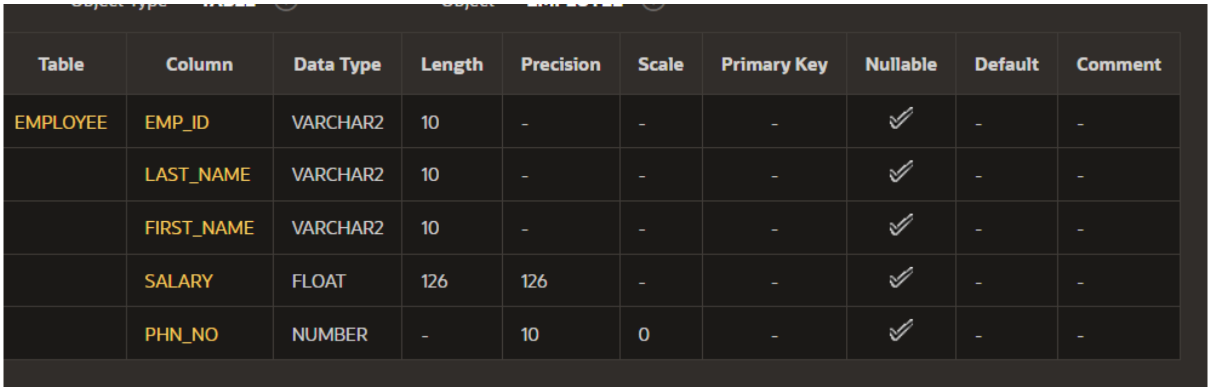


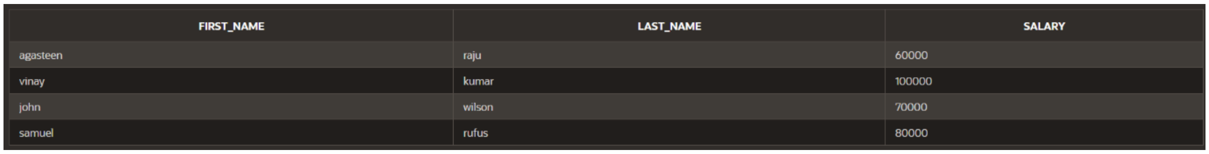
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	EMP_ID	VARCHAR2	10	-	-	-	✓	-	-
	LAST_NAME	VARCHAR2	10	-	-	-	✓	-	-
	FIRST_NAME	VARCHAR2	10	-	-	-	✓	-	-
	SALARY	FLOAT	126	126	-	-	✓	-	-
	PHN_NO	NUMBER	-	10	0	-	✓	-	-

Using WHERE clause

SELECT first_name,last_name,salary

FROM Employee

WHERE salary > 50000.0;



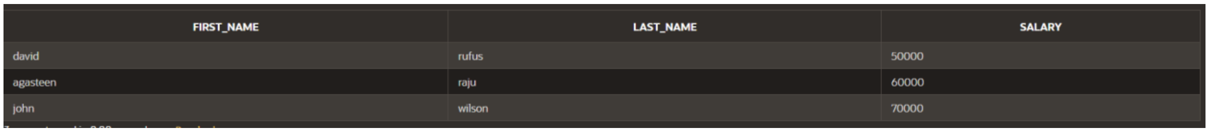
FIRST_NAME	LAST_NAME	SALARY
agasteen	raju	60000
vinay	kumar	100000
john	wilson	70000
samuel	rufus	80000

Using BETWEEN and AND operator

SELECT first_name,last_name ,salary

FROM Employee

WHERE salary BETWEEN 35000.0 AND 75000.0;



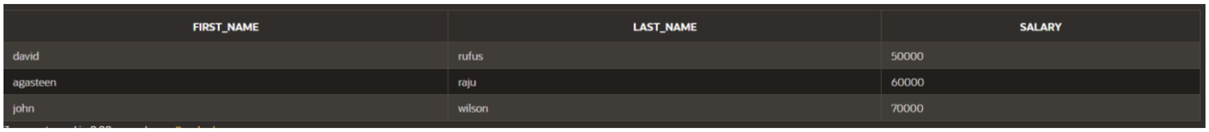
FIRST_NAME	LAST_NAME	SALARY
david	rufus	50000
agasteen	raju	60000
john	wilson	70000

By using IN operator

SELECT first_name,last_name,salary

FROM Employee

WHERE emp_id IN (2,5);



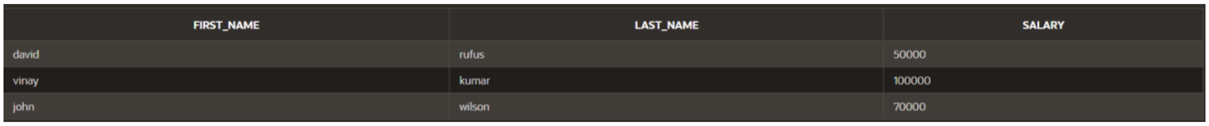
FIRST_NAME	LAST_NAME	SALARY
david	rufus	50000
agasteen	raju	60000
john	wilson	70000

By using NOT IN operator

SELECT first_name,last_name,salary

FROM Employee

WHERE emp_id NOT IN (2,5);



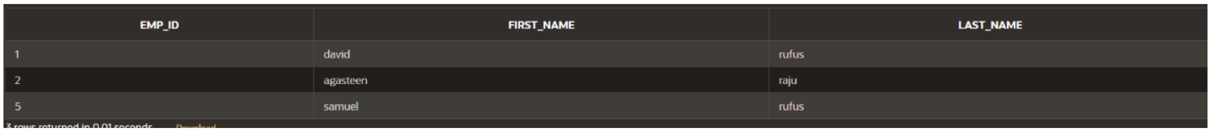
FIRST_NAME	LAST_NAME	SALARY
david	rufus	50000
vinay	kumar	100000
john	wilson	70000

By using LIKE operator

SELECT first_name,last_name,salary

FROM Employee

WHERE last_name LIKE ‘r%’;



EMP_ID	FIRST_NAME	LAST_NAME
1	david	rufus
2	agasteen	raju
5	samuel	rufus

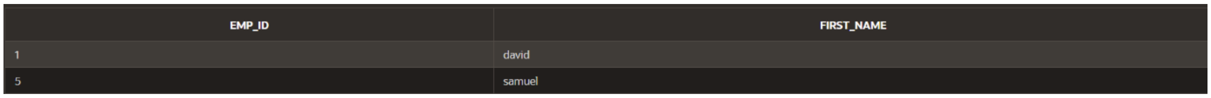
3 rows returned in 0.01 seconds Download

By using OR operator

SELECT emp_id,first_name

FROM Employee

WHERE emp_id=’1’ OR emp_id = ‘5’;



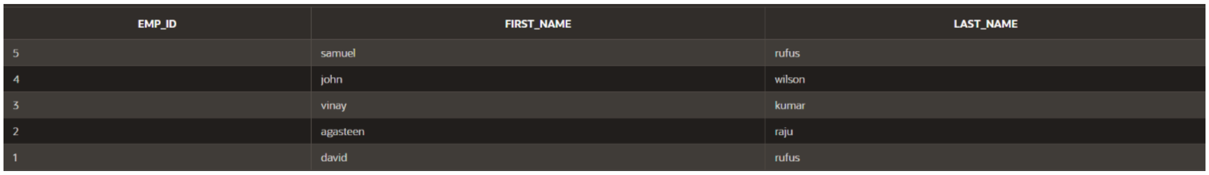
EMP_ID	FIRST_NAME
1	david
5	samuel

Arranging or ordering the tablein ASCENDING and DESCENDING ORDER by using ORDER BY

SELECT emp_id,first_name,last_name

FROM Employee

ORDER BY emp_id DESC;

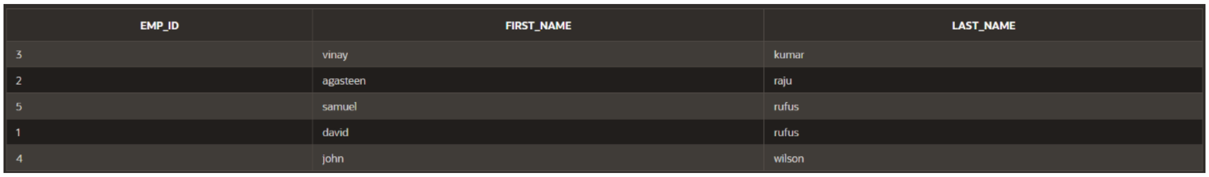


EMP_ID	FIRST_NAME	LAST_NAME
5	samuel	rufus
4	john	wilson
3	vinay	kumar
2	agasteen	raju
1	david	rufus

SELECT emp_id,first_name,last_name

FROM employee

ORDER BY last_name ;



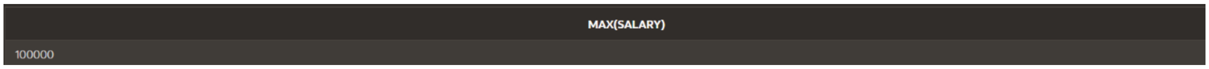
EMP_ID	FIRST_NAME	LAST_NAME
3	vinay	kumar
2	agasteen	raju
5	samuel	rufus
1	david	rufus
4	john	wilson

GROUP FUNCTIONS :

MAX :

SELECT MAX(salary)

FROM Employee;

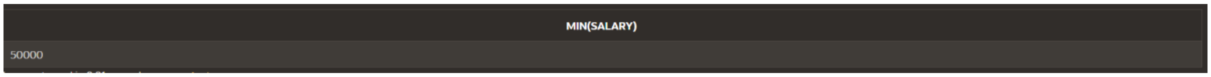


MAX(SALARY)
100000

MIN:

SELECT MIN(salary)

FROM Employee;

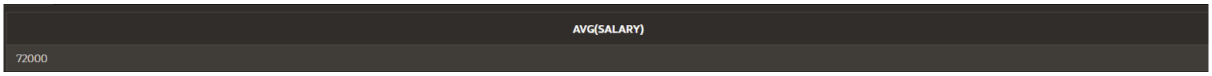


MIN(SALARY)
50000

AVERAGE :

SELECT AVG(salary)

FROM Employee;

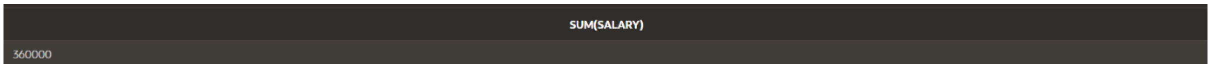


AVG(SALARY)
72000

SUM :

SELECT SUM(salary)

FROM Employee;



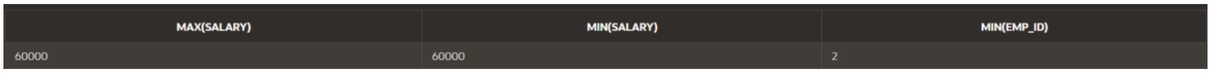
SUM(SALARY)
360000

More than one GROUP function :

SELECT MAX(Salary),MIN(Salary),MIN(emp_id)

FROM Employee

WHERE last_name='raju';

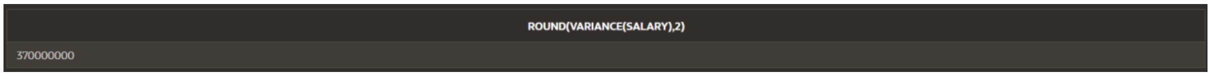


MAX(SALARY)	MIN(SALARY)	MIN(EMP_ID)
60000	60000	2

VARIANCE :

SELECT ROUND(VARIANCE(salary),2)

FROM Employee;

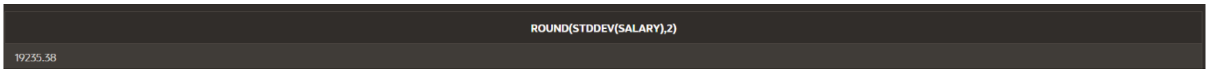


ROUND(VARIANCE(SALARY),2)
370000000

STANDARD DEVIATION :

SELECT ROUND(STDDEV(salary),2)

FROM Employee;

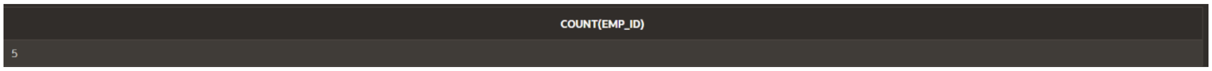


ROUND(STDDEV(SALARY),2)
19235.38

COUNT :

SELECT COUNT(emp_id)

FROM Employee;



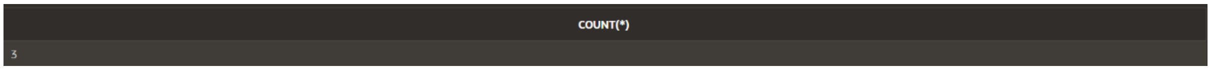
COUNT(EMP_ID)
5

COUNT(*)

SELECT COUNT(*)

FROM Employee

WHERE emp_id > 2;

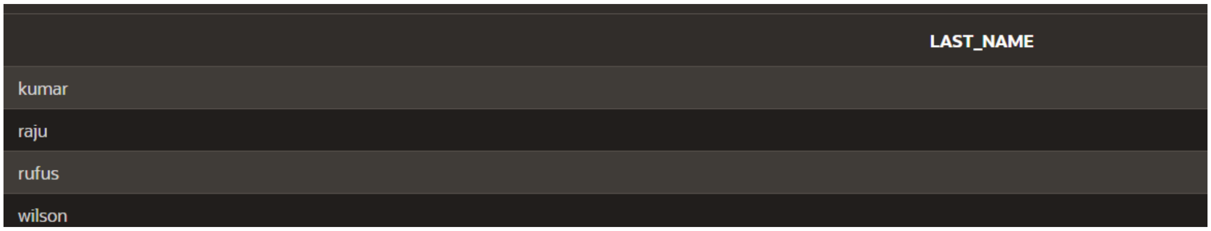


COUNT(*)
3

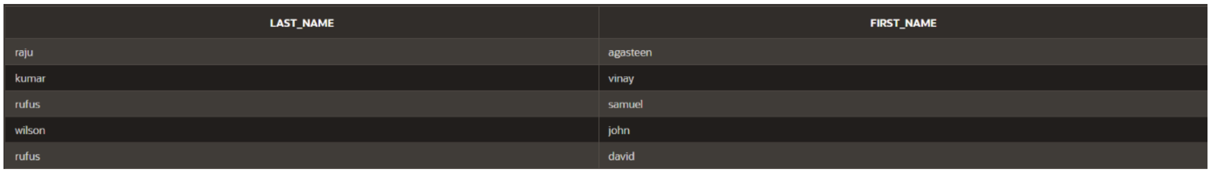
DISTINCT

SELECT DISTINCT last_name

FROM Employee;



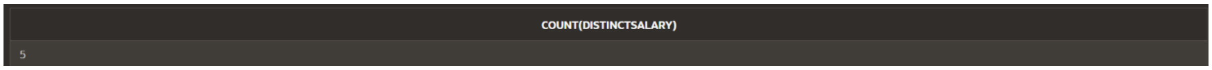
LAST_NAME
kumar
raju
rufus
wilson



LAST_NAME	FIRST_NAME
raju	agasteen
kumar	vinay
rufus	samuel
wilson	john
rufus	david

SELECT COUNT(DISTINCT salary)

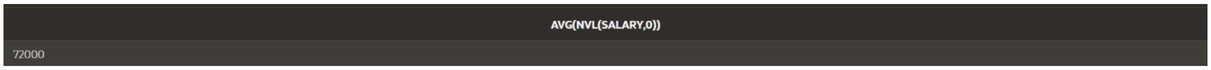
FROM Employee;



COUNT(DISTINCTSALARY)
5

SELECT AVG(NVL(salary,0))

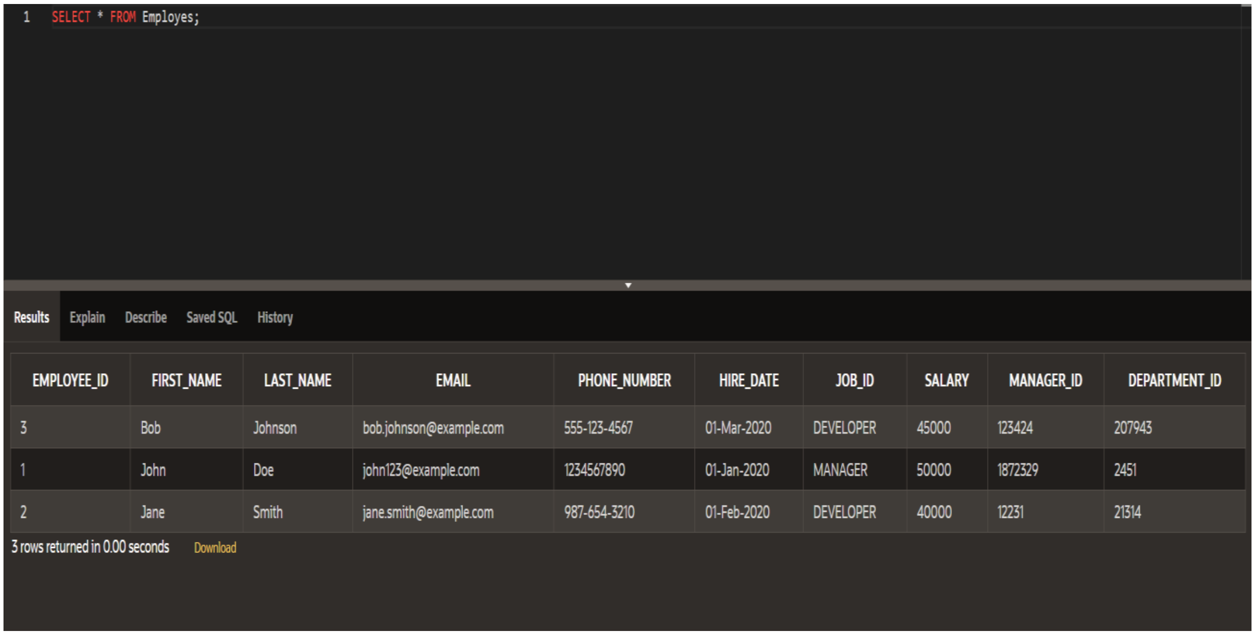
FROM Employee;



AVG(NVL(SALARY,0))
72000

Section 9

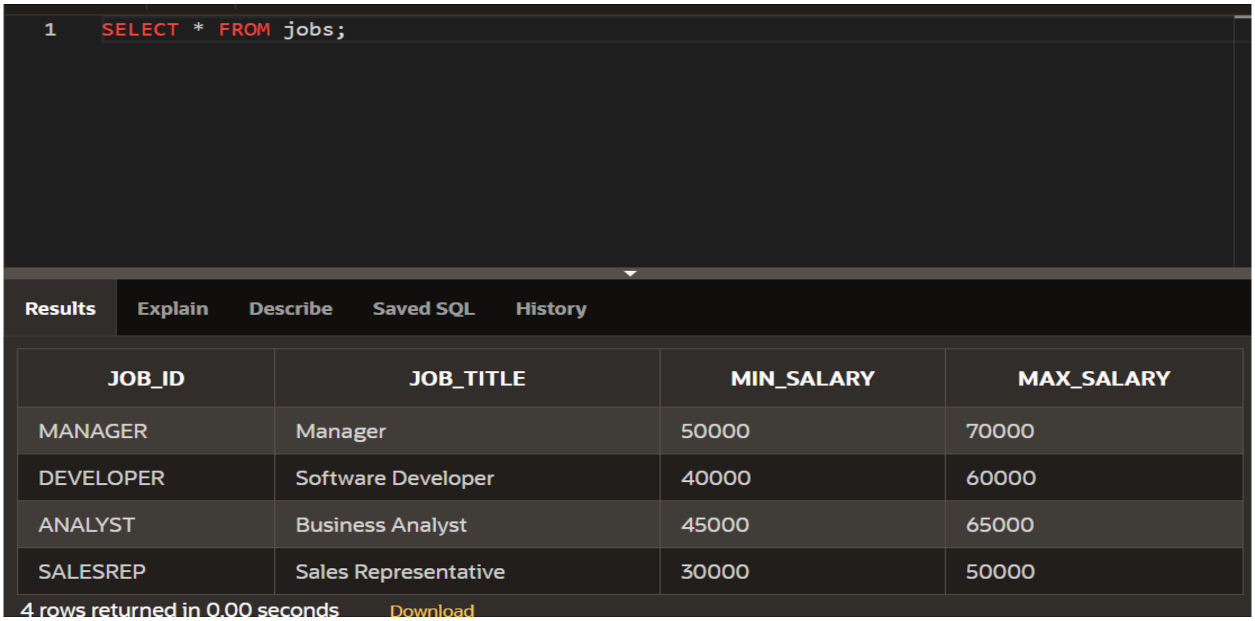
<u>DP 9.1:</u>



```sql
SELECT * FROM Employes;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	MANAGER_ID	DEPARTMENT_ID
3	Bob	Johnson	bob.johnson@example.com	555-123-4567	01-Mar-2020	DEVELOPER	45000	123424	207943
1	John	Doe	john123@example.com	1234567890	01-Jan-2020	MANAGER	50000	1872329	2451
2	Jane	Smith	jane.smith@example.com	987-654-3210	01-Feb-2020	DEVELOPER	40000	12231	21314

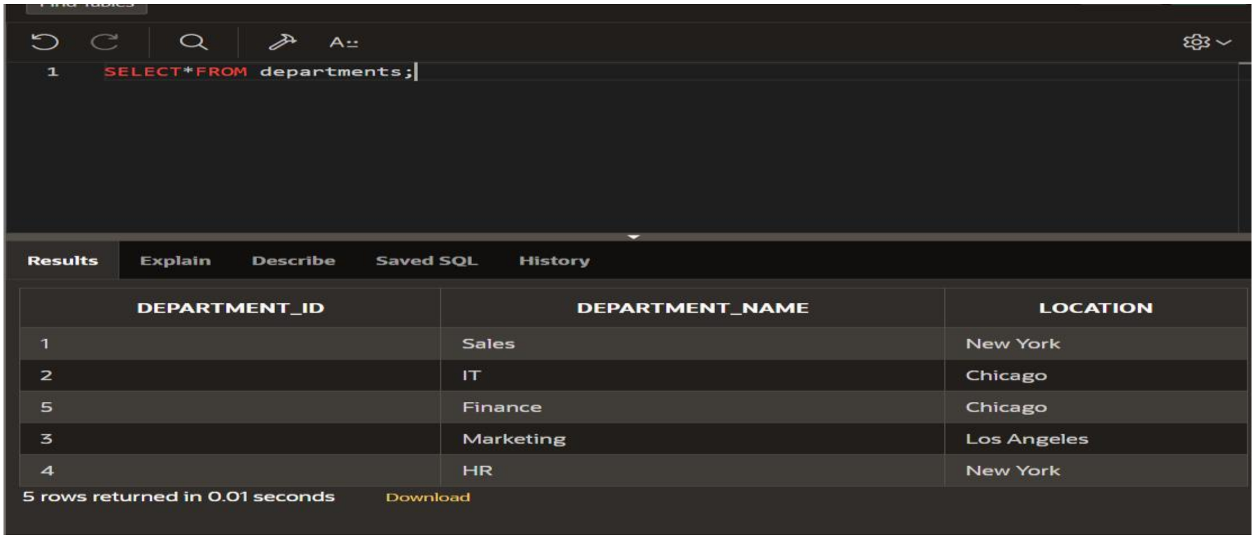
3 rows returned in 0.00 seconds



```sql
SELECT * FROM jobs;
```

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
MANAGER	Manager	50000	70000
DEVELOPER	Software Developer	40000	60000
ANALYST	Business Analyst	45000	65000
SALESREP	Sales Representative	30000	50000

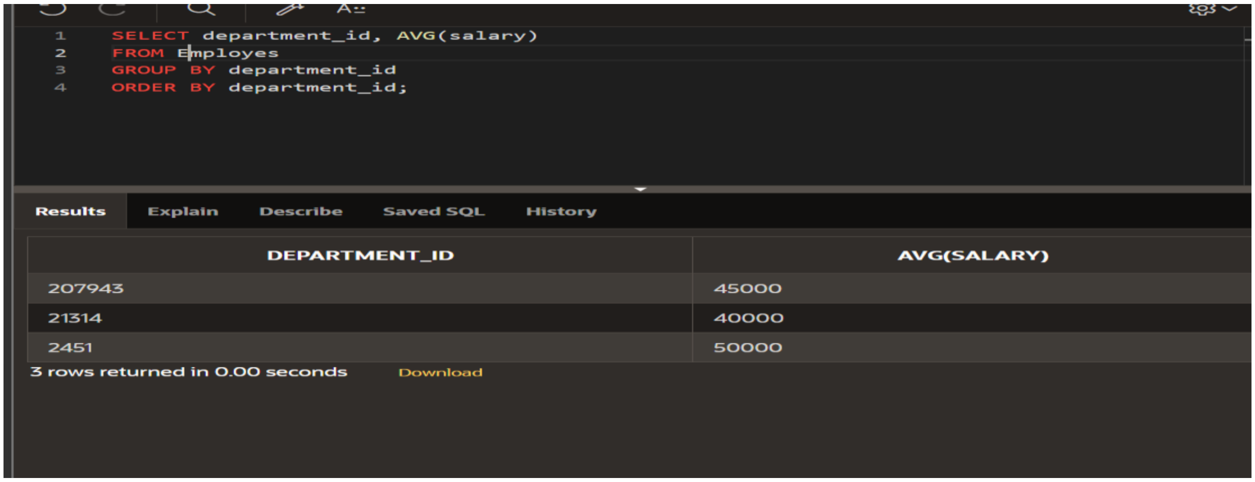
4 rows returned in 0.00 seconds



```sql
SELECT*FROM departments;
```

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION
1	Sales	New York
2	IT	Chicago
5	Finance	Chicago
3	Marketing	Los Angeles
4	HR	New York

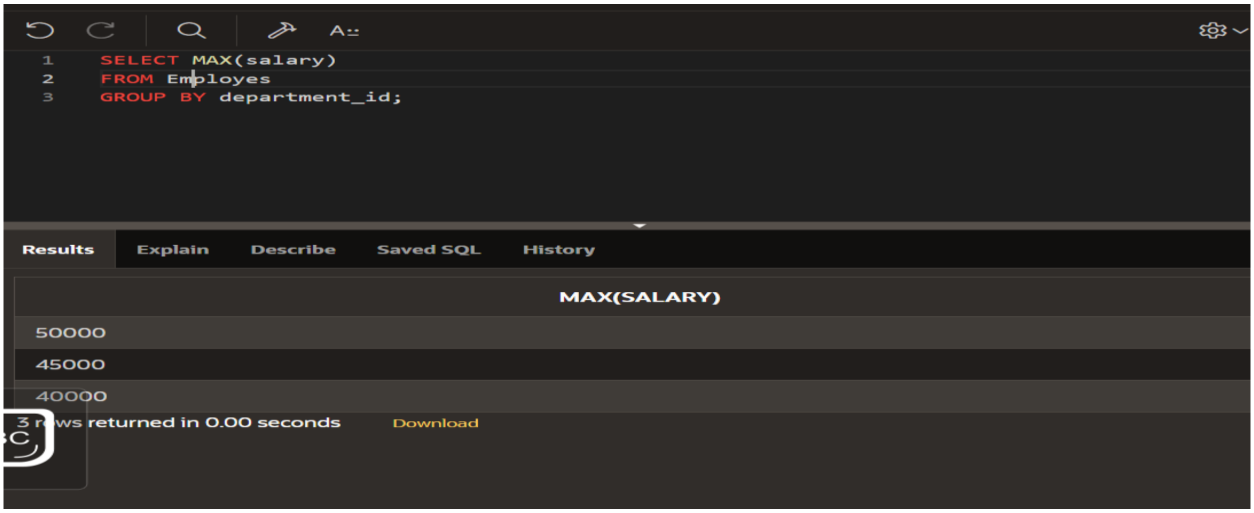
5 rows returned in 0.01 seconds   Download



```sql
SELECT department_id, AVG(salary)
FROM Employes
GROUP BY department_id
ORDER BY department_id;
```

DEPARTMENT_ID	AVG(SALARY)
207943	45000
21314	40000
2451	50000

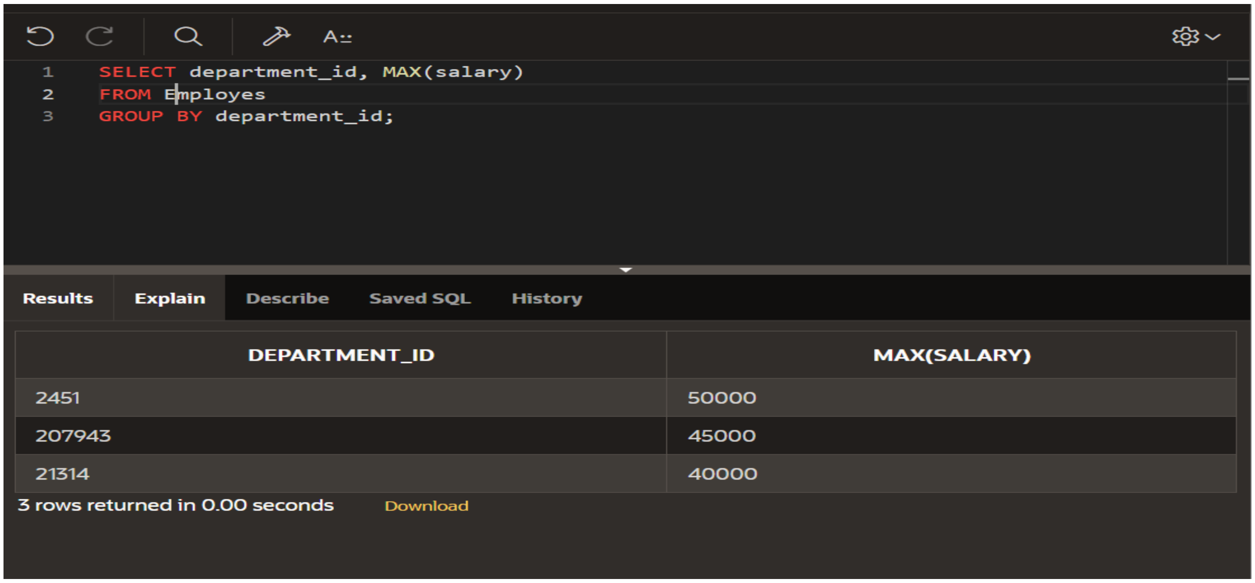
3 rows returned in 0.00 seconds   Download



```sql
SELECT MAX(salary)
FROM Employes
GROUP BY department_id;
```

MAX(SALARY)
50000
45000
40000

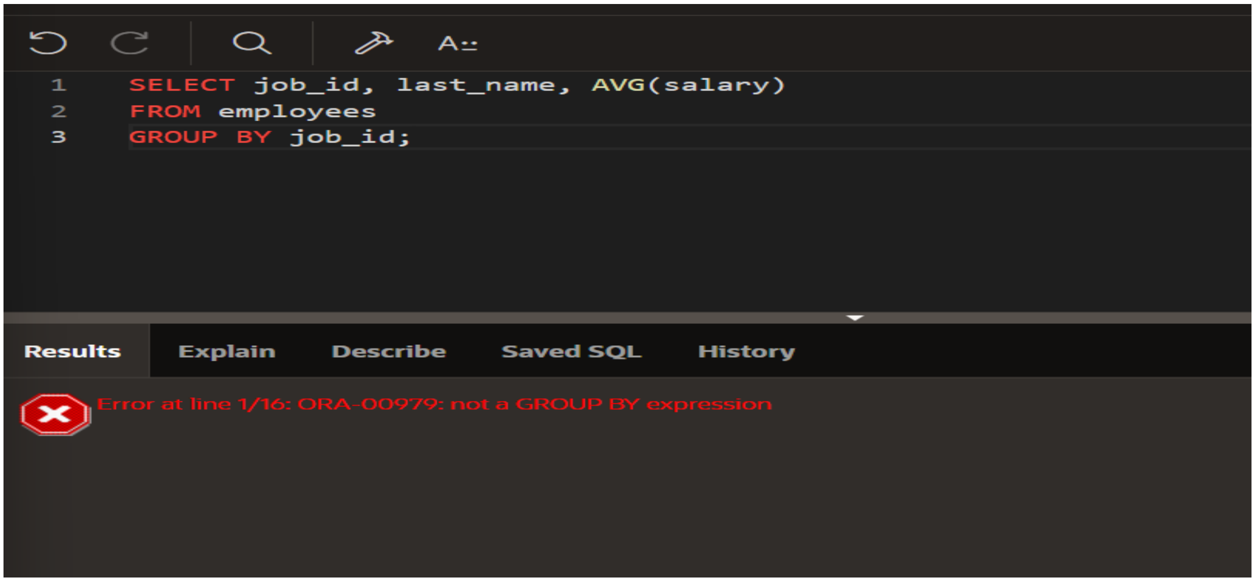
3 rows returned in 0.00 seconds   Download



```
SELECT department_id, MAX(salary)
FROM Employes
GROUP BY department_id;
```

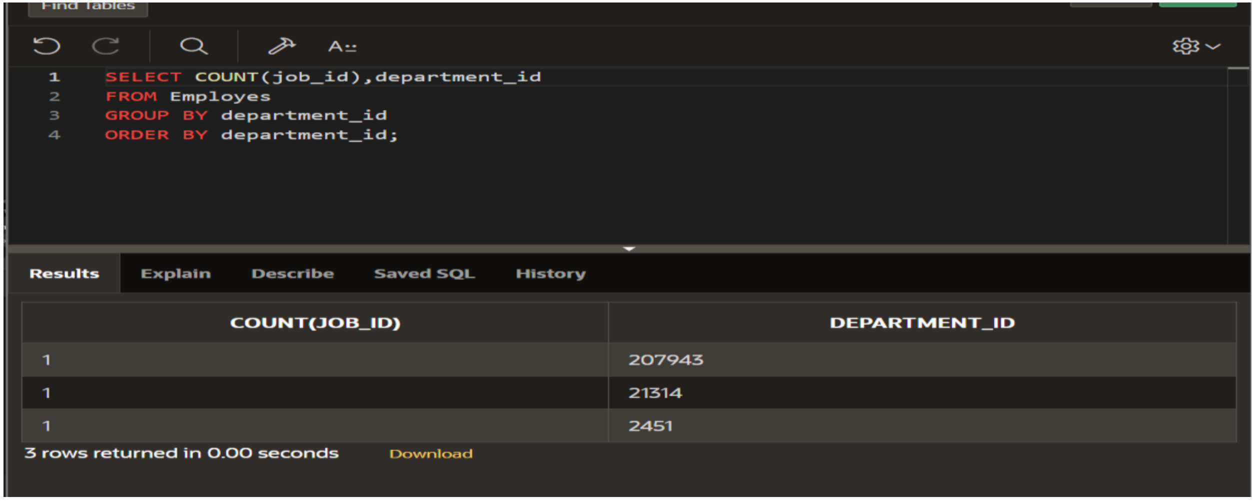
DEPARTMENT_ID	MAX(SALARY)
2451	50000
207943	45000
21314	40000

3 rows returned in 0.00 seconds Download



```
SELECT job_id, last_name, AVG(salary)
FROM employees
GROUP BY job_id;
```

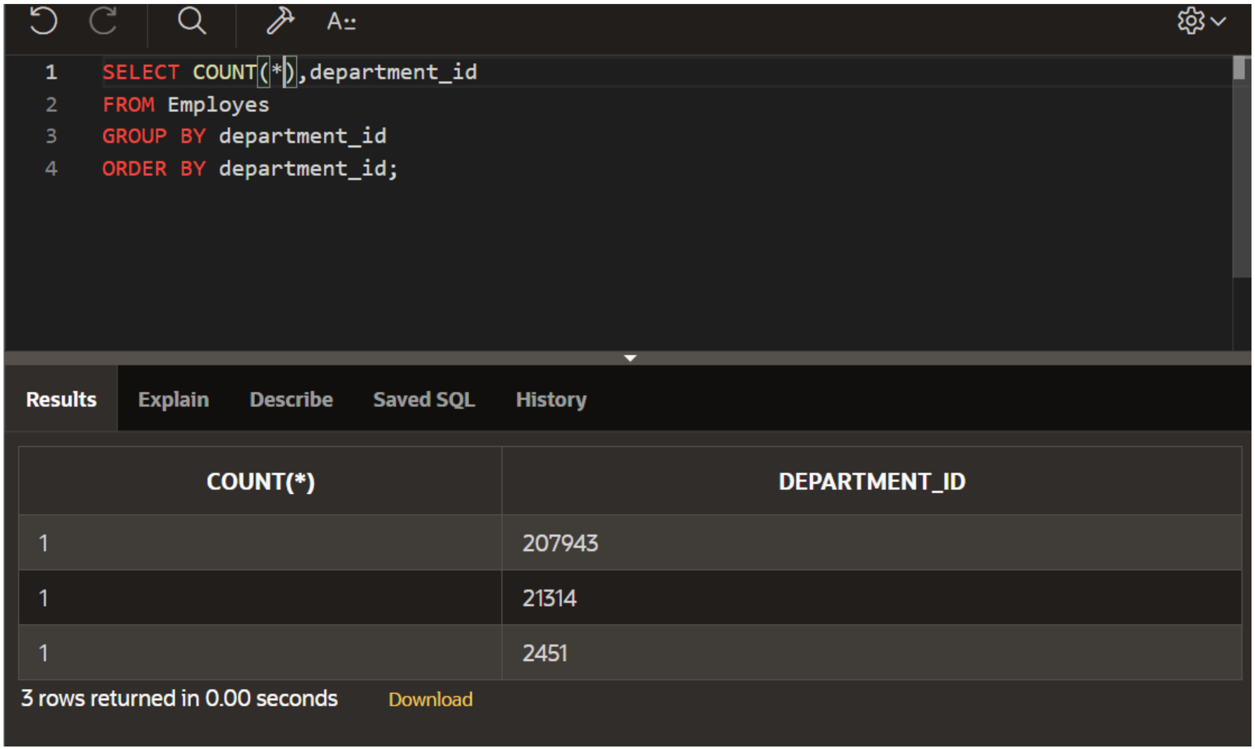
Error at line 1/16: ORA-00979: not a GROUP BY expression



```
SELECT COUNT(job_id),department_id
FROM Employes
GROUP BY department_id
ORDER BY department_id;
```

COUNT(JOB_ID)	DEPARTMENT_ID
1	207943
1	21314
1	2451

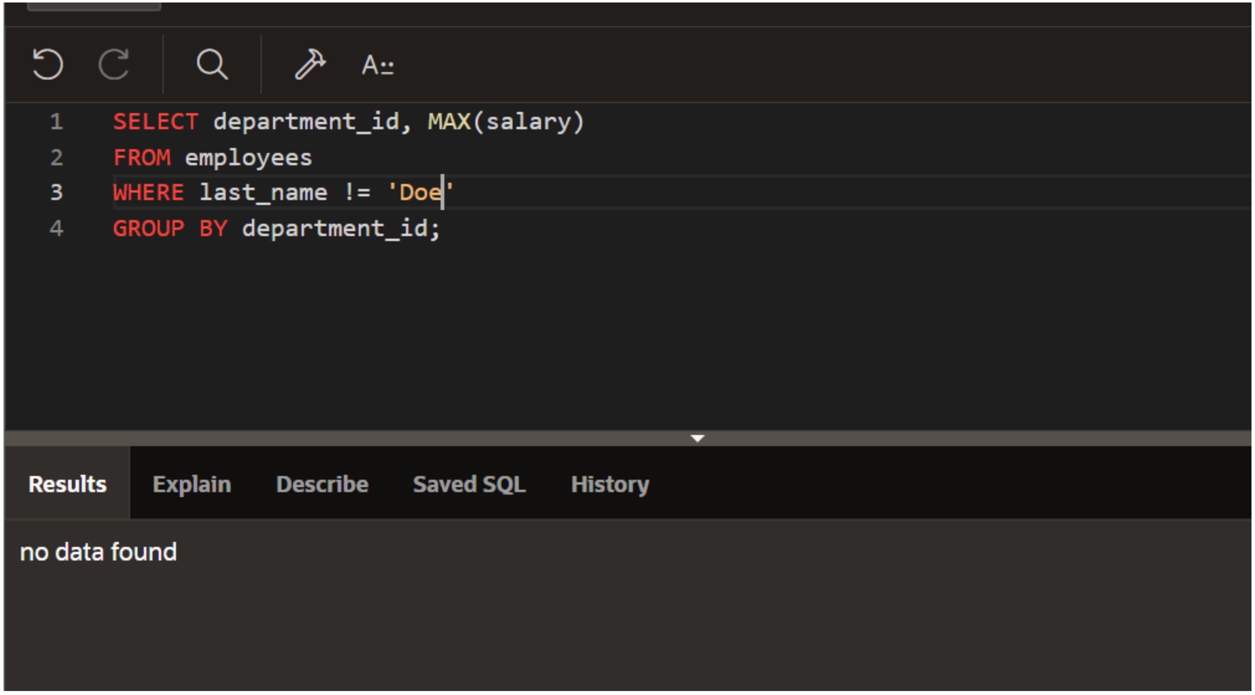
3 rows returned in 0.00 seconds Download



```sql
SELECT COUNT(*),department_id
FROM Employes
GROUP BY department_id
ORDER BY department_id;
```

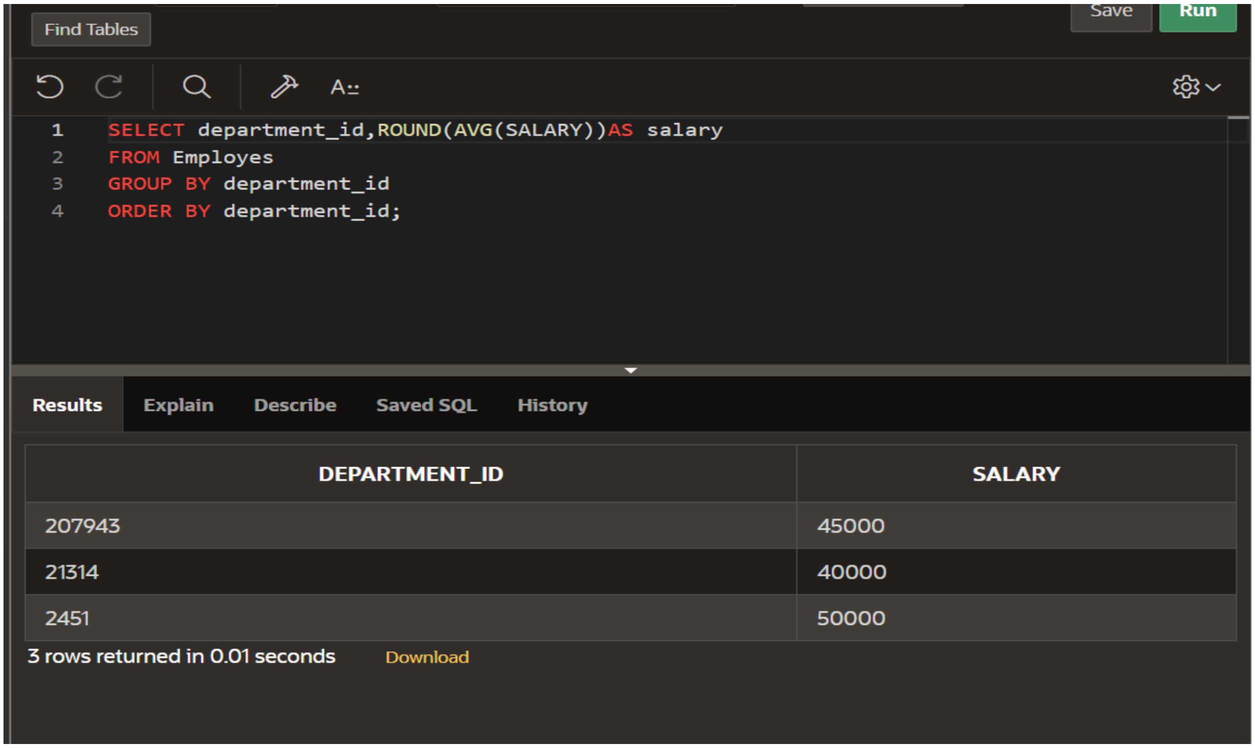
COUNT(*)	DEPARTMENT_ID
1	207943
1	21314
1	2451

3 rows returned in 0.00 seconds



```sql
SELECT department_id, MAX(salary)
FROM employees
WHERE last_name != 'Doe'
GROUP BY department_id;
```

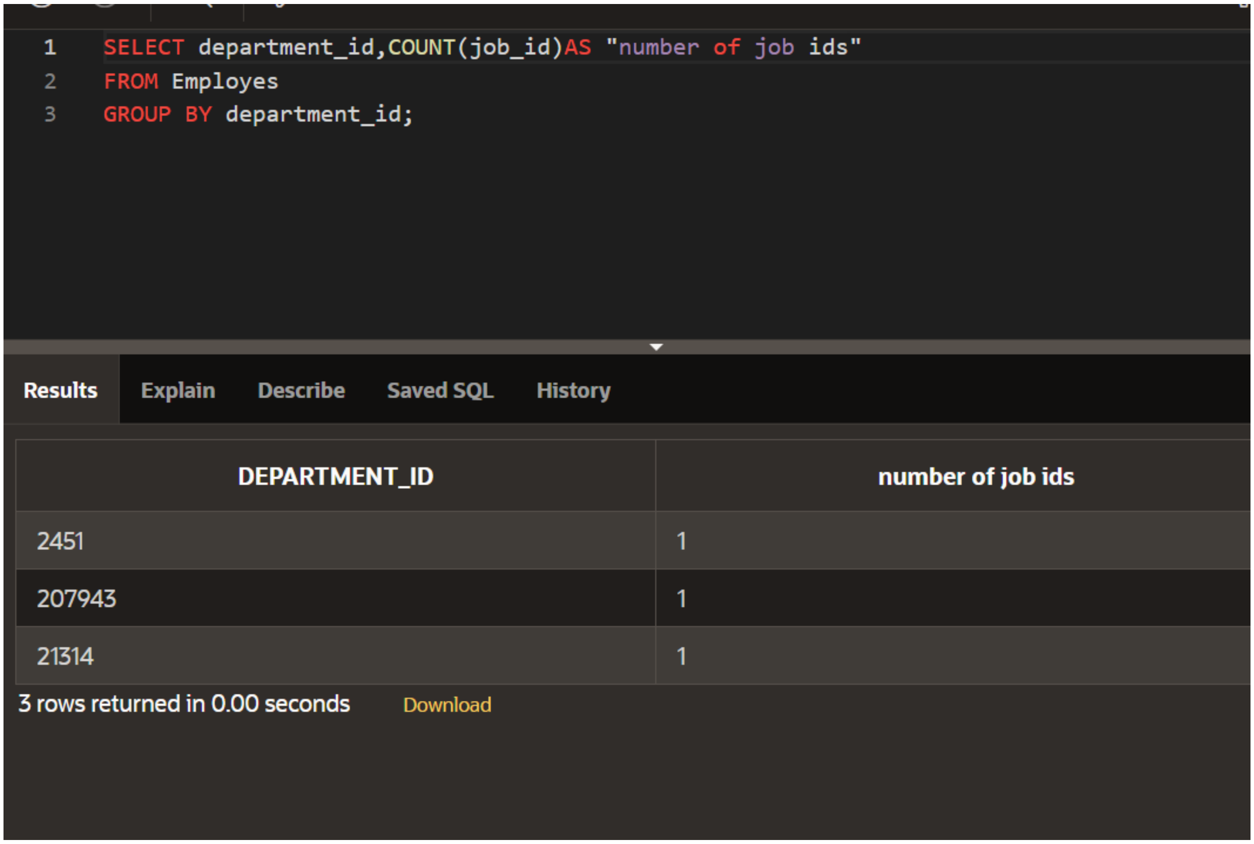
no data found



```
SELECT department_id,ROUND(AVG(SALARY))AS salary
FROM Employes
GROUP BY department_id
ORDER BY department_id;
```

DEPARTMENT_ID	SALARY
207943	45000
21314	40000
2451	50000

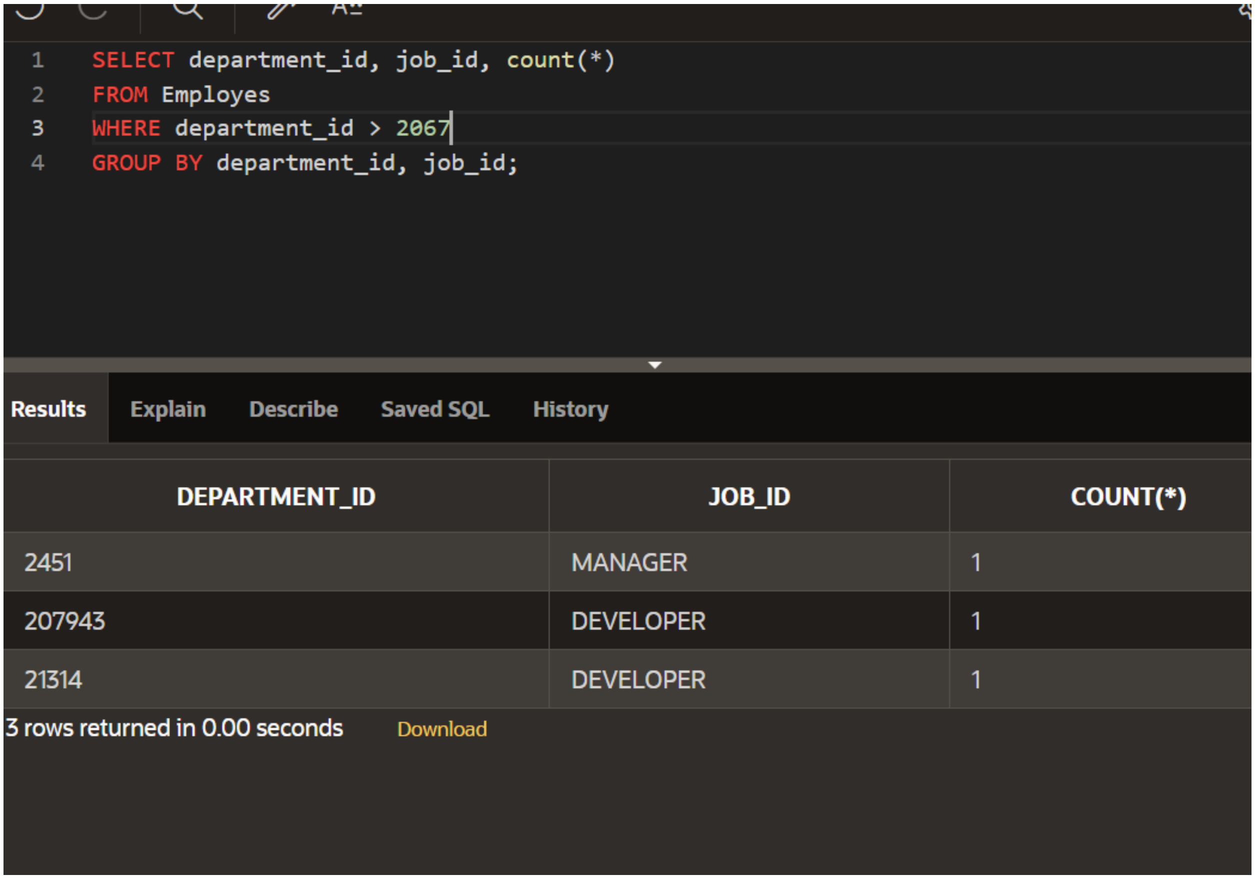
3 rows returned in 0.01 seconds



```
SELECT department_id,COUNT(job_id)AS "number of job ids"
FROM Employes
GROUP BY department_id;
```

DEPARTMENT_ID	number of job ids
2451	1
207943	1
21314	1

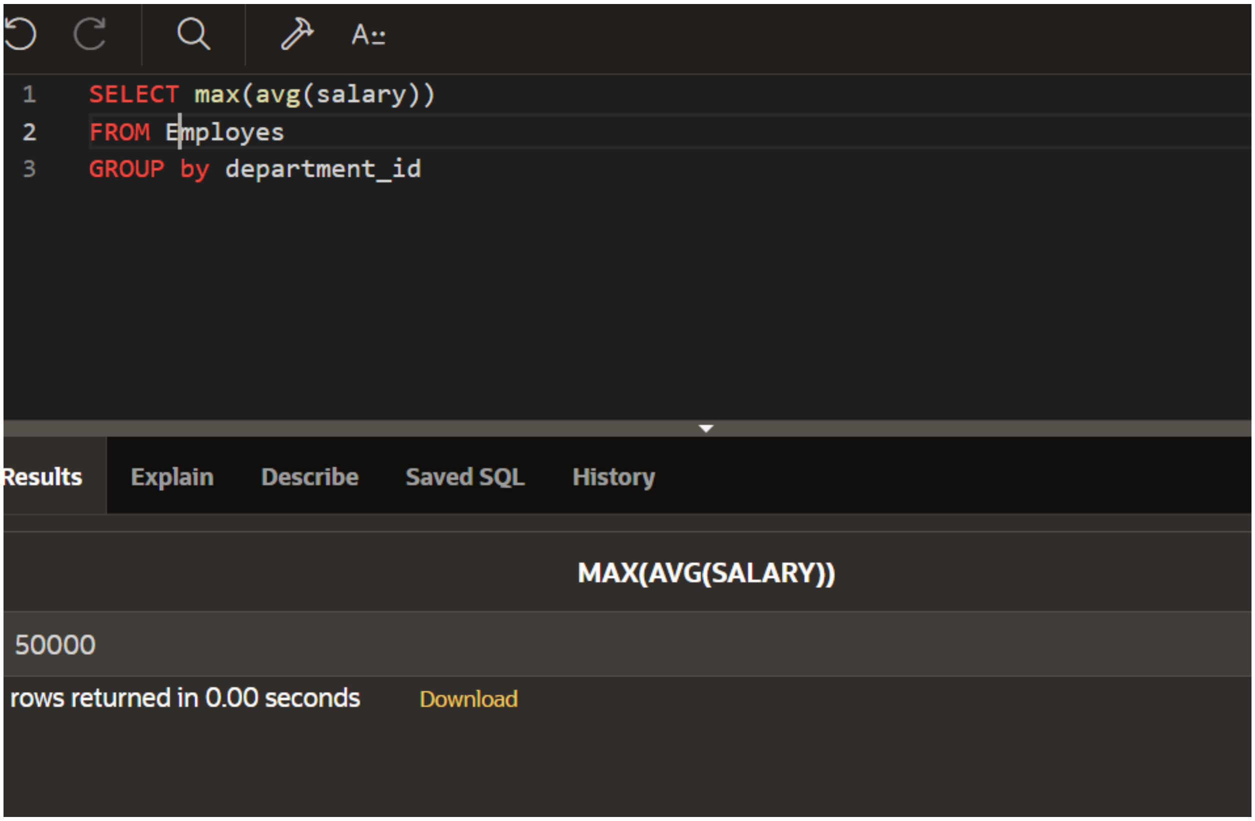
3 rows returned in 0.00 seconds



```sql
SELECT department_id, job_id, count(*)
FROM Employes
WHERE department_id > 2067
GROUP BY department_id, job_id;
```

DEPARTMENT_ID	JOB_ID	COUNT(*)
2451	MANAGER	1
207943	DEVELOPER	1
21314	DEVELOPER	1

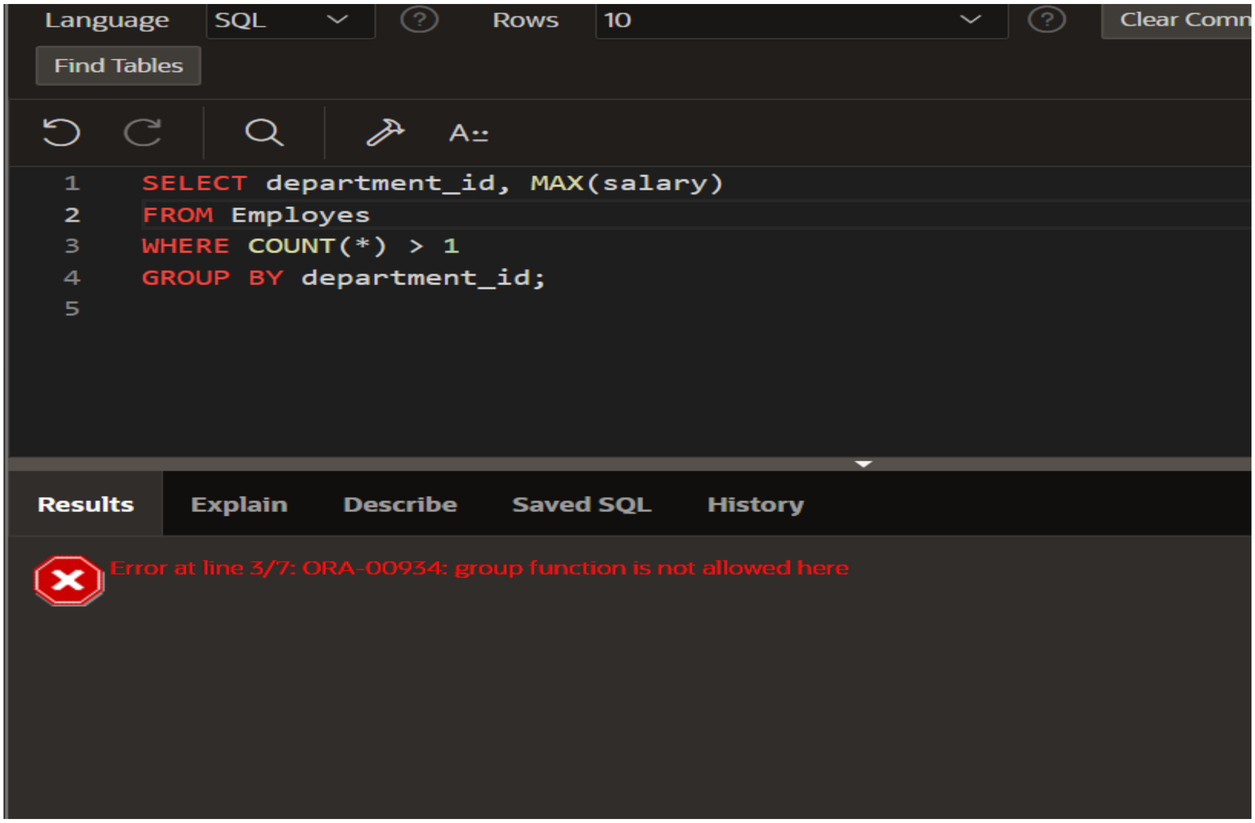
3 rows returned in 0.00 seconds



```sql
SELECT max(avg(salary))
FROM Employes
GROUP by department_id
```

MAX(AVG(SALARY))
50000

rows returned in 0.00 seconds

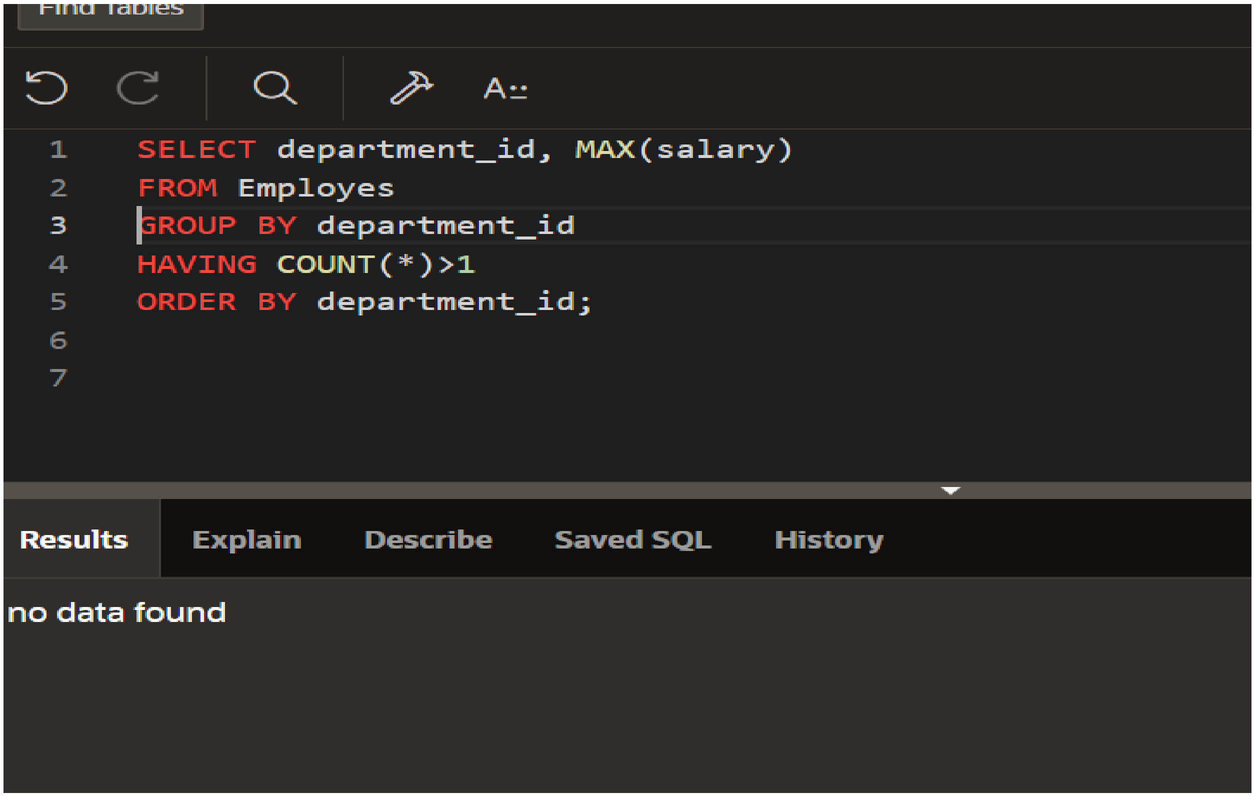


Language SQL

Rows 10

```
SELECT department_id, MAX(salary)
FROM Employes
WHERE COUNT(*) > 1
GROUP BY department_id;
```

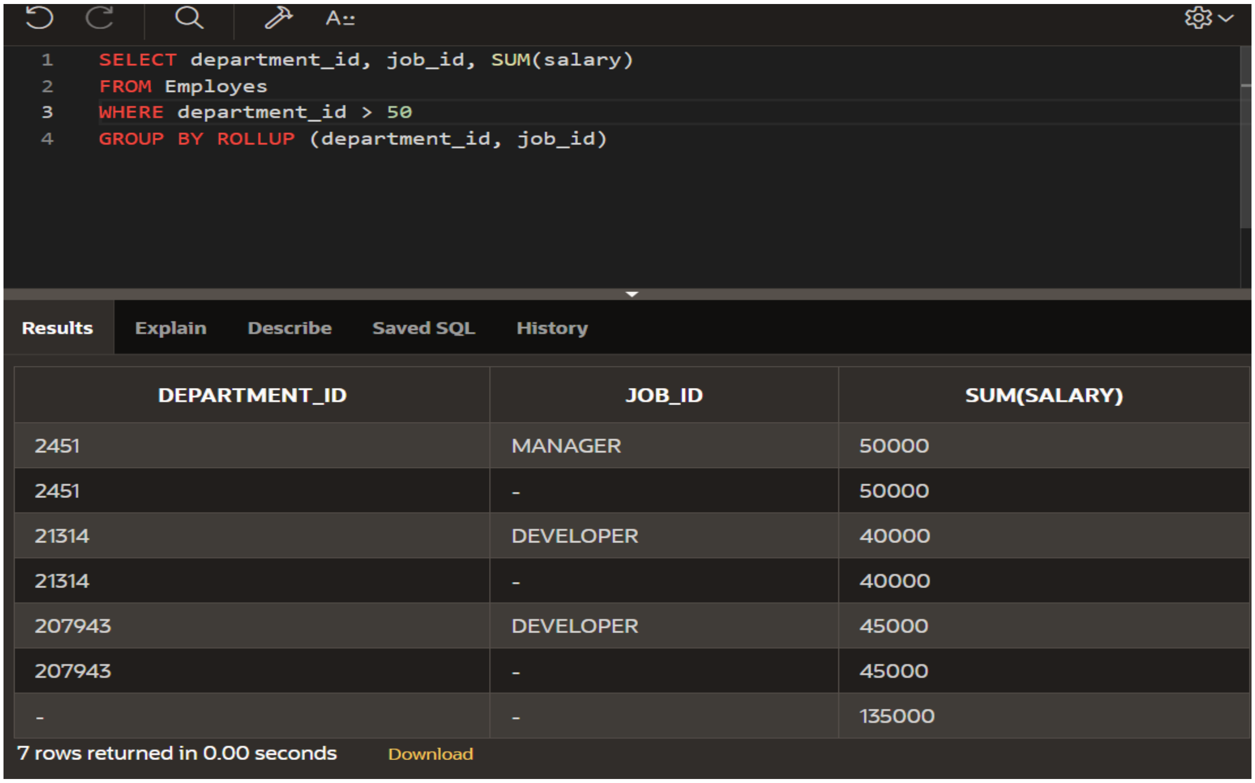
```
Error at line 3/7: ORA-00934: group function is not allowed here
```



```
SELECT department_id, MAX(salary)
FROM Employes
GROUP BY department_id
HAVING COUNT(*)>1
ORDER BY department_id;
```

```
no data found
```

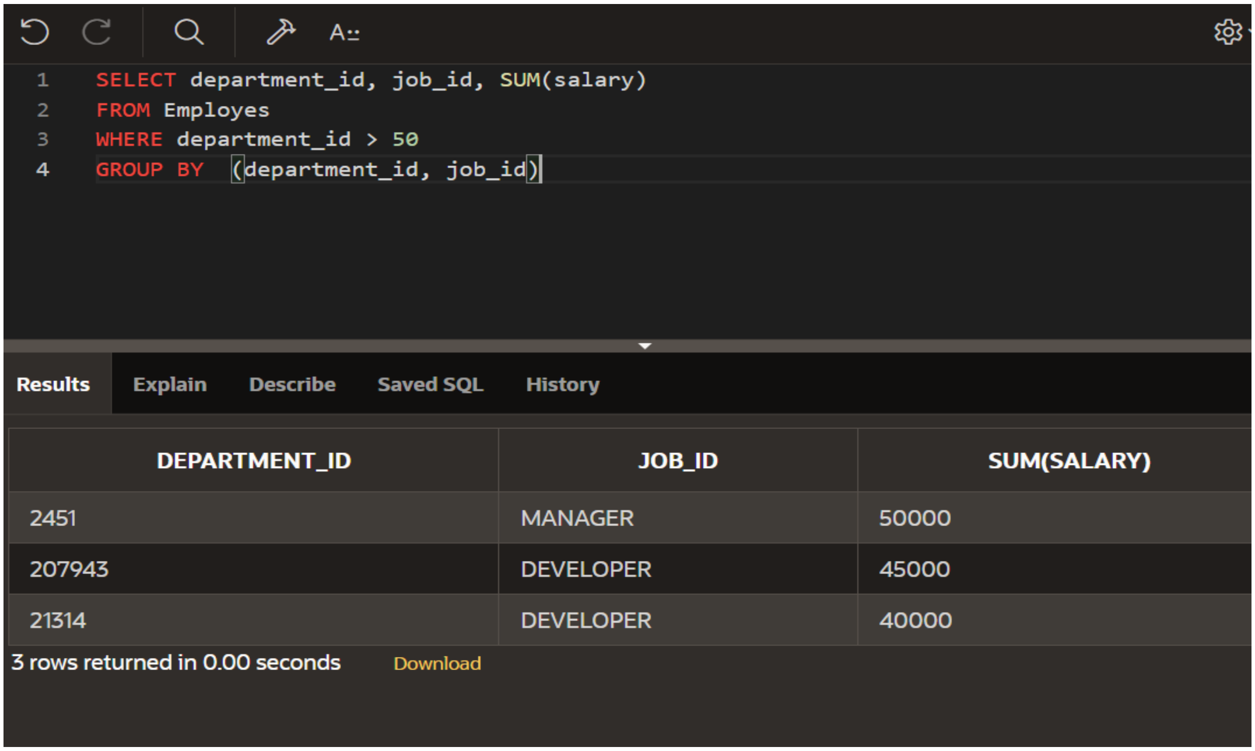
DP 9.2:



```
SELECT department_id, job_id, SUM(salary)
FROM Employes
WHERE department_id > 50
GROUP BY ROLLUP (department_id, job_id)
```

DEPARTMENT_ID	JOB_ID	SUM(SALARY)
2451	MANAGER	50000
2451	-	50000
21314	DEVELOPER	40000
21314	-	40000
207943	DEVELOPER	45000
207943	-	45000
-	-	135000

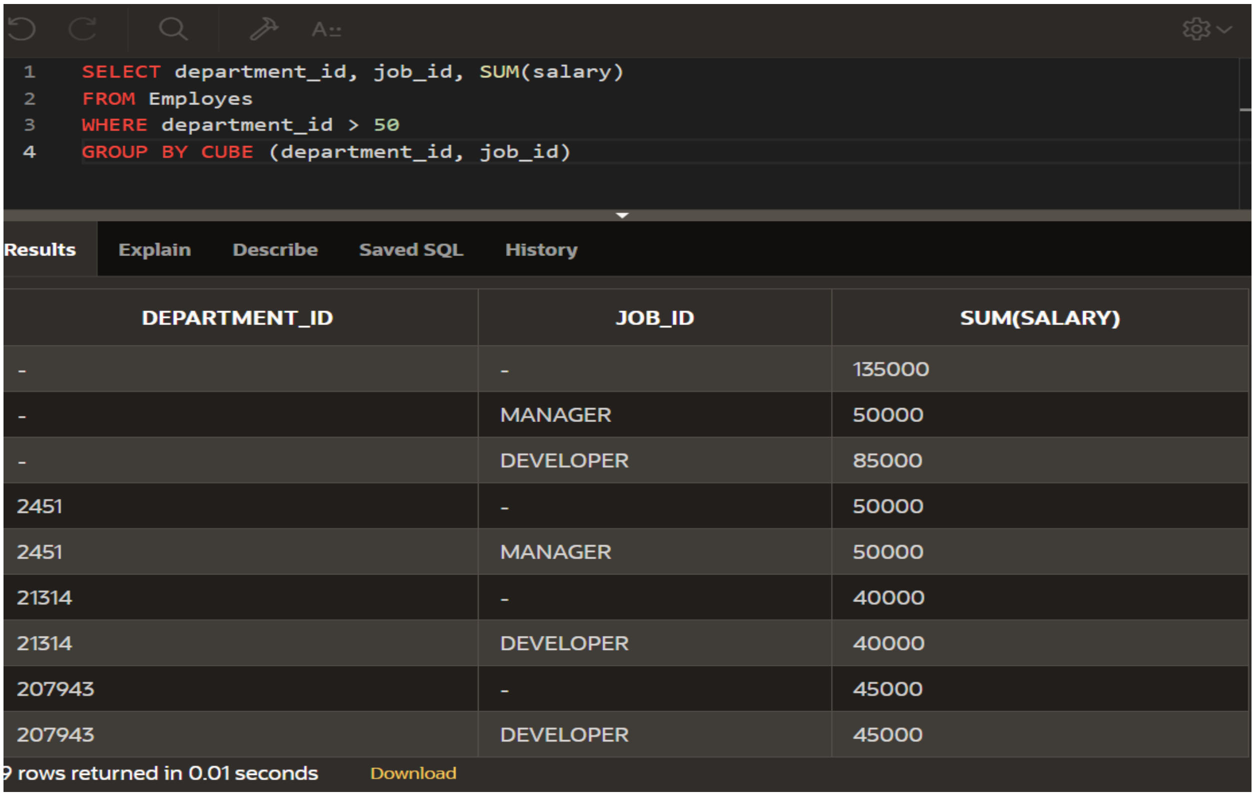
7 rows returned in 0.00 seconds



```
SELECT department_id, job_id, SUM(salary)
FROM Employes
WHERE department_id > 50
GROUP BY  (department_id, job_id)
```

DEPARTMENT_ID	JOB_ID	SUM(SALARY)
2451	MANAGER	50000
207943	DEVELOPER	45000
21314	DEVELOPER	40000

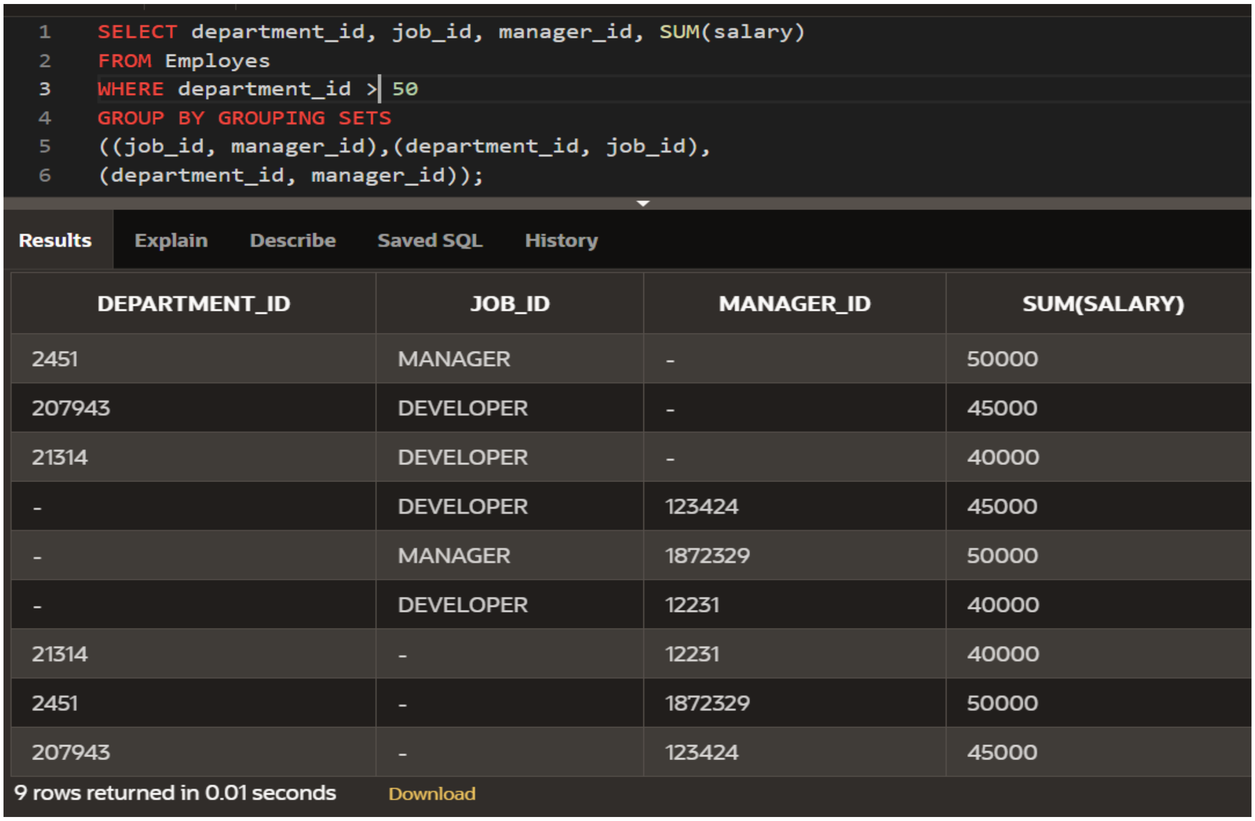
3 rows returned in 0.00 seconds



```
SELECT department_id, job_id, SUM(salary)
FROM Employes
WHERE department_id > 50
GROUP BY CUBE (department_id, job_id)
```

DEPARTMENT_ID	JOB_ID	SUM(SALARY)
-	-	135000
-	MANAGER	50000
-	DEVELOPER	85000
2451	-	50000
2451	MANAGER	50000
21314	-	40000
21314	DEVELOPER	40000
207943	-	45000
207943	DEVELOPER	45000

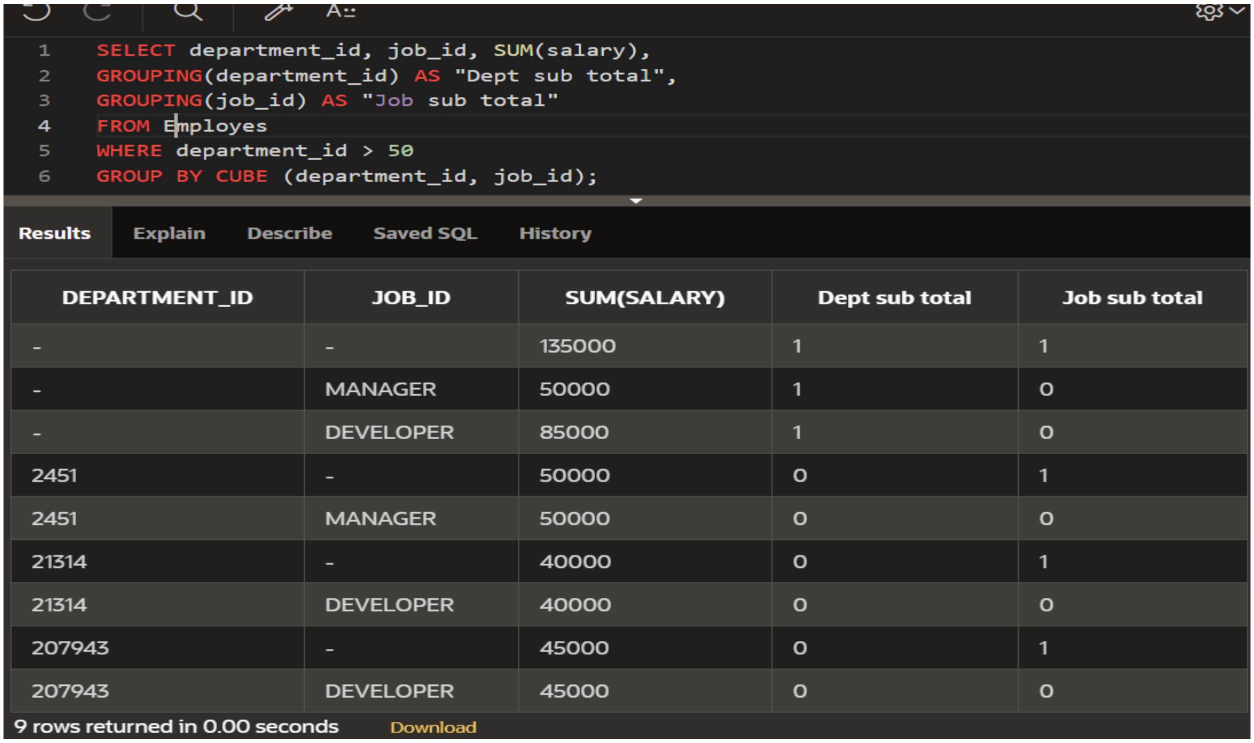
9 rows returned in 0.01 seconds



```
SELECT department_id, job_id, manager_id, SUM(salary)
FROM Employes
WHERE department_id > 50
GROUP BY GROUPING SETS
((job_id, manager_id),(department_id, job_id),
(department_id, manager_id));
```

DEPARTMENT_ID	JOB_ID	MANAGER_ID	SUM(SALARY)
2451	MANAGER	-	50000
207943	DEVELOPER	-	45000
21314	DEVELOPER	-	40000
-	DEVELOPER	123424	45000
-	MANAGER	1872329	50000
-	DEVELOPER	12231	40000
21314	-	12231	40000
2451	-	1872329	50000
207943	-	123424	45000

9 rows returned in 0.01 seconds

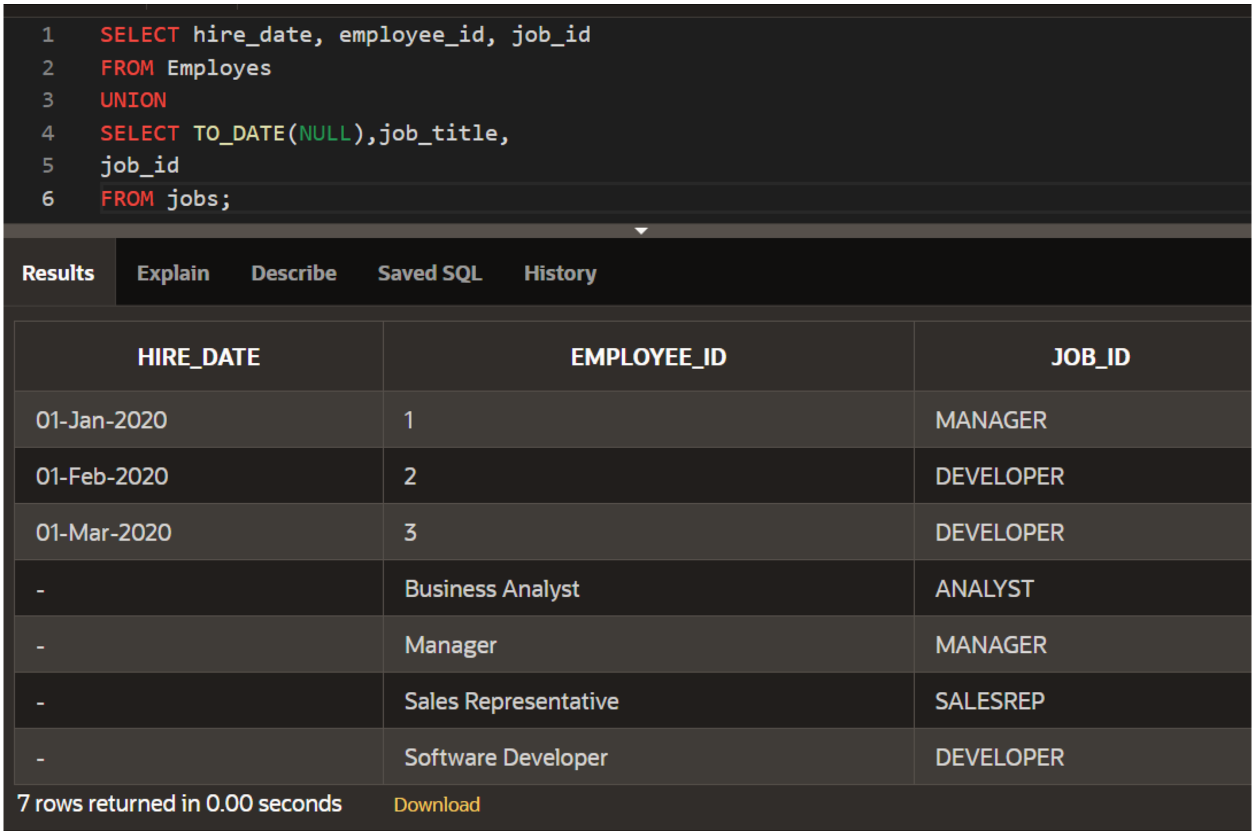


```sql
SELECT department_id, job_id, SUM(salary),
GROUPING(department_id) AS "Dept sub total",
GROUPING(job_id) AS "Job sub total"
FROM Employes
WHERE department_id > 50
GROUP BY CUBE (department_id, job_id);
```

DEPARTMENT_ID	JOB_ID	SUM(SALARY)	Dept sub total	Job sub total
-	-	135000	1	1
-	MANAGER	50000	1	0
-	DEVELOPER	85000	1	0
2451	-	50000	0	1
2451	MANAGER	50000	0	0
21314	-	40000	0	1
21314	DEVELOPER	40000	0	0
207943	-	45000	0	1
207943	DEVELOPER	45000	0	0

9 rows returned in 0.00 seconds

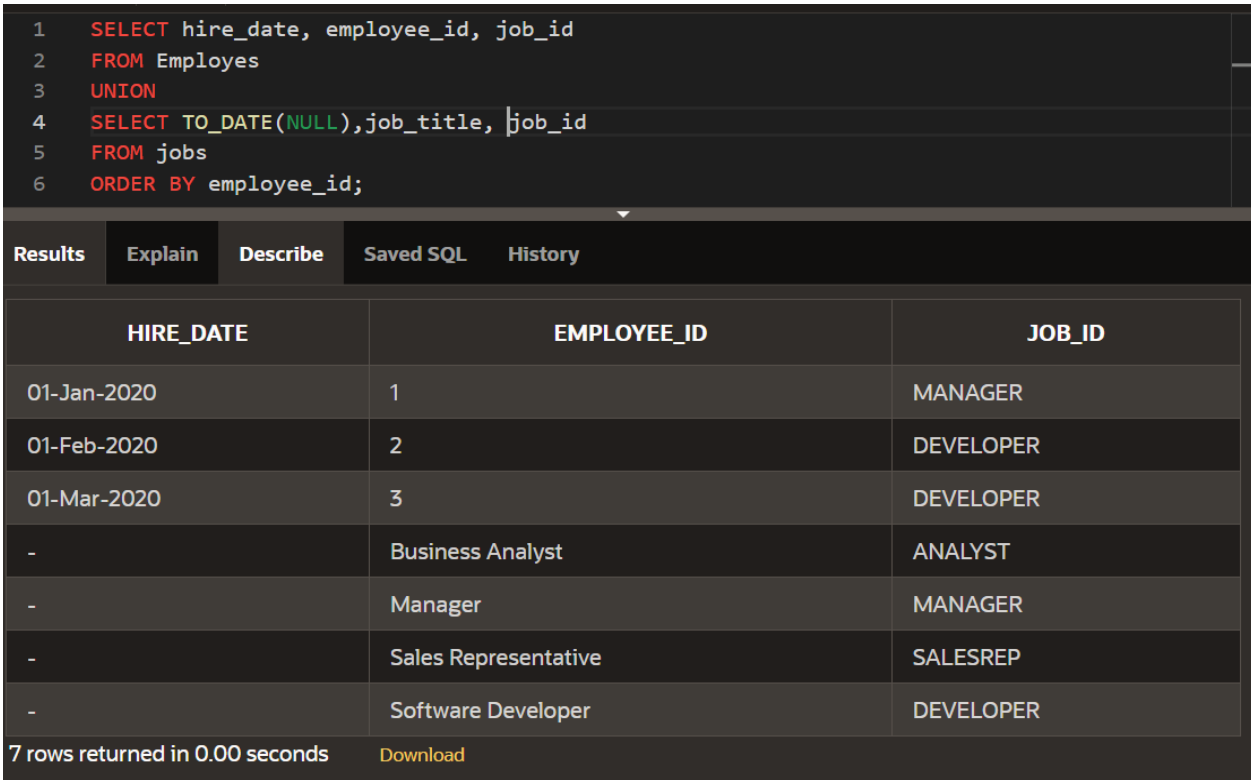
DP 9.3:



```sql
SELECT hire_date, employee_id, job_id
FROM Employes
UNION
SELECT TO_DATE(NULL),job_title,
job_id
FROM jobs;
```

HIRE_DATE	EMPLOYEE_ID	JOB_ID
01-Jan-2020	1	MANAGER
01-Feb-2020	2	DEVELOPER
01-Mar-2020	3	DEVELOPER
-	Business Analyst	ANALYST
-	Manager	MANAGER
-	Sales Representative	SALESREP
-	Software Developer	DEVELOPER

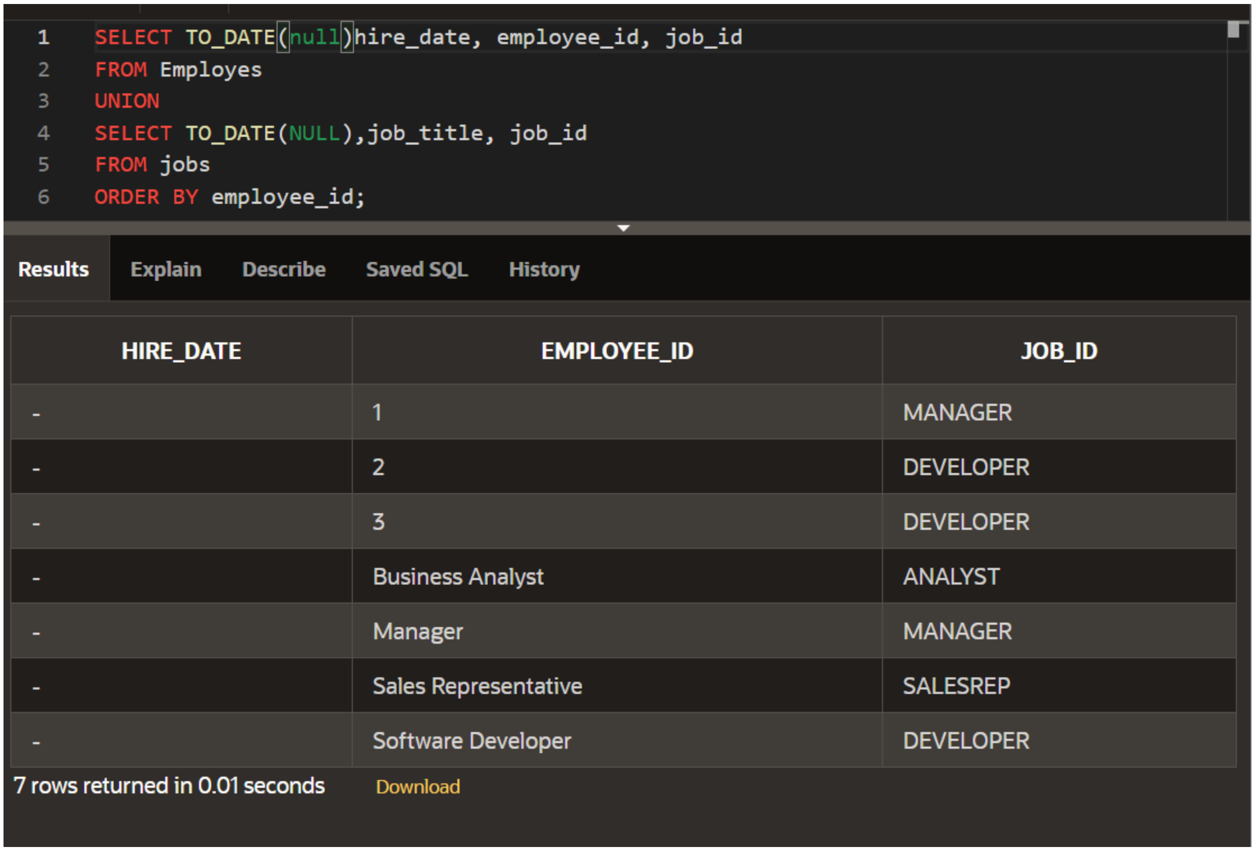
7 rows returned in 0.00 seconds



```
SELECT hire_date, employee_id, job_id
FROM Employes
UNION
SELECT TO_DATE(NULL),job_title, job_id
FROM jobs
ORDER BY employee_id;
```

HIRE_DATE	EMPLOYEE_ID	JOB_ID
01-Jan-2020	1	MANAGER
01-Feb-2020	2	DEVELOPER
01-Mar-2020	3	DEVELOPER
-	Business Analyst	ANALYST
-	Manager	MANAGER
-	Sales Representative	SALESREP
-	Software Developer	DEVELOPER

7 rows returned in 0.00 seconds Download



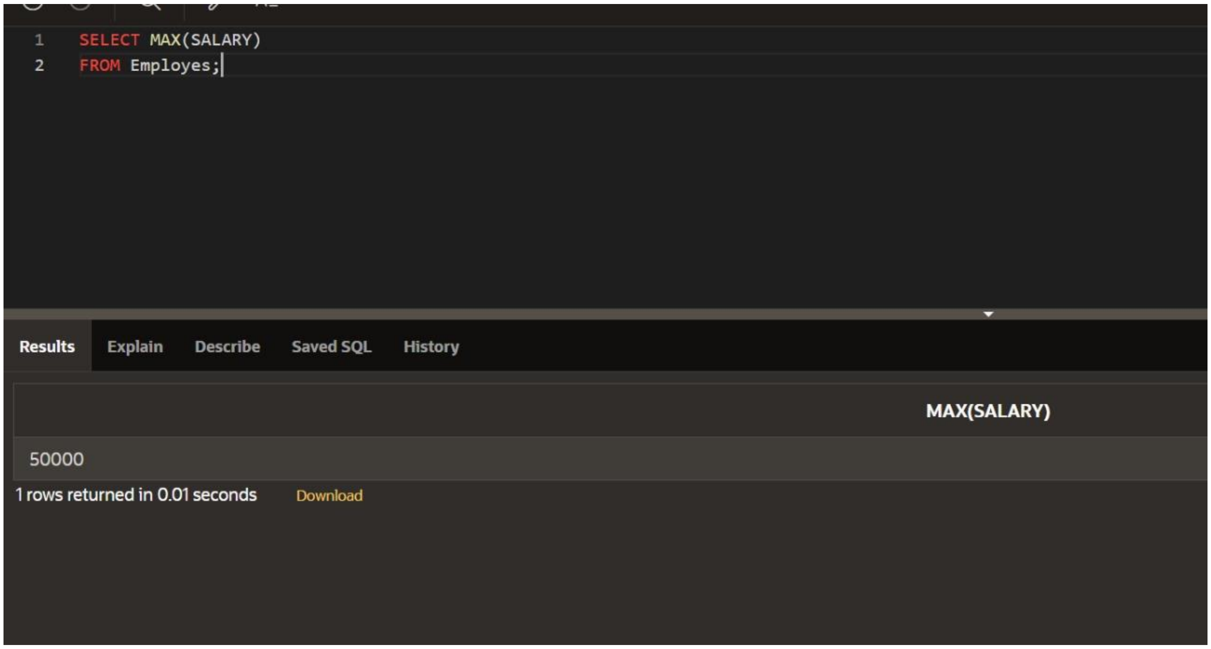
```
SELECT TO_DATE(null)hire_date, employee_id, job_id
FROM Employes
UNION
SELECT TO_DATE(NULL),job_title, job_id
FROM jobs
ORDER BY employee_id;
```

HIRE_DATE	EMPLOYEE_ID	JOB_ID
-	1	MANAGER
-	2	DEVELOPER
-	3	DEVELOPER
-	Business Analyst	ANALYST
-	Manager	MANAGER
-	Sales Representative	SALESREP
-	Software Developer	DEVELOPER

7 rows returned in 0.01 seconds Download

SECTION 8

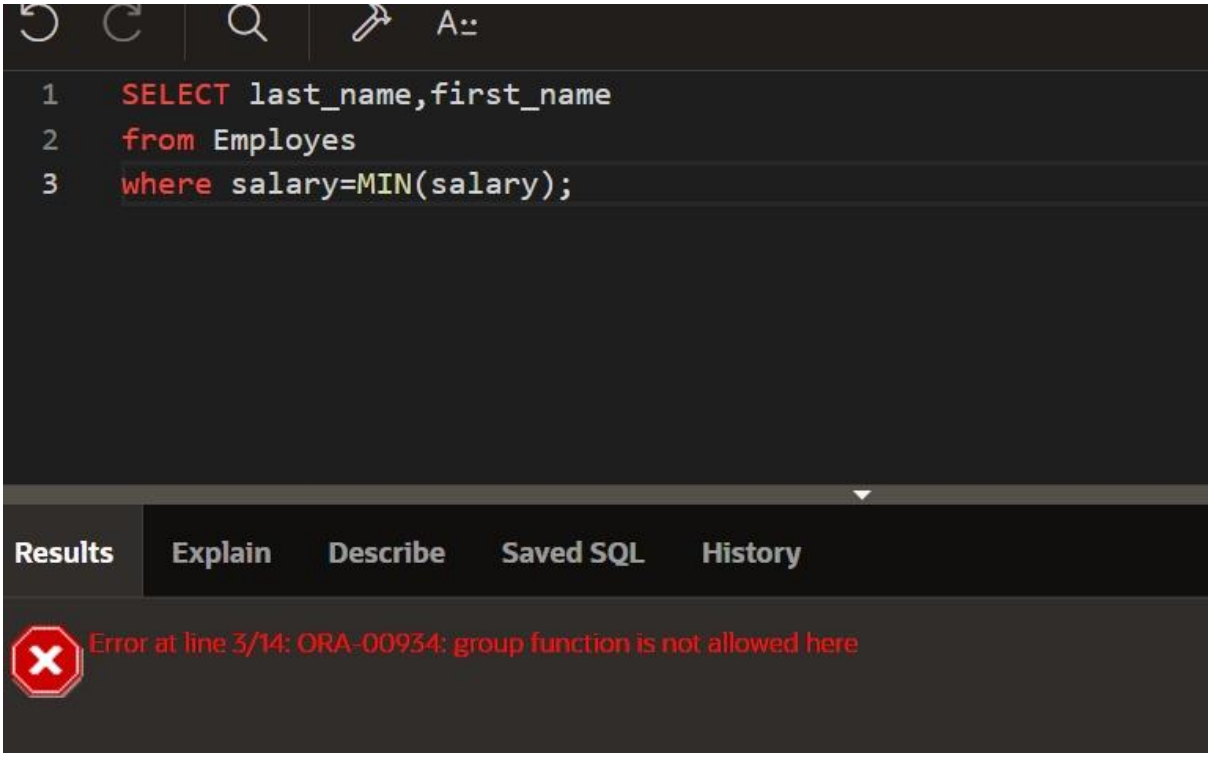
DP_8.1:



```sql
SELECT MAX(SALARY)
FROM Employes;
```

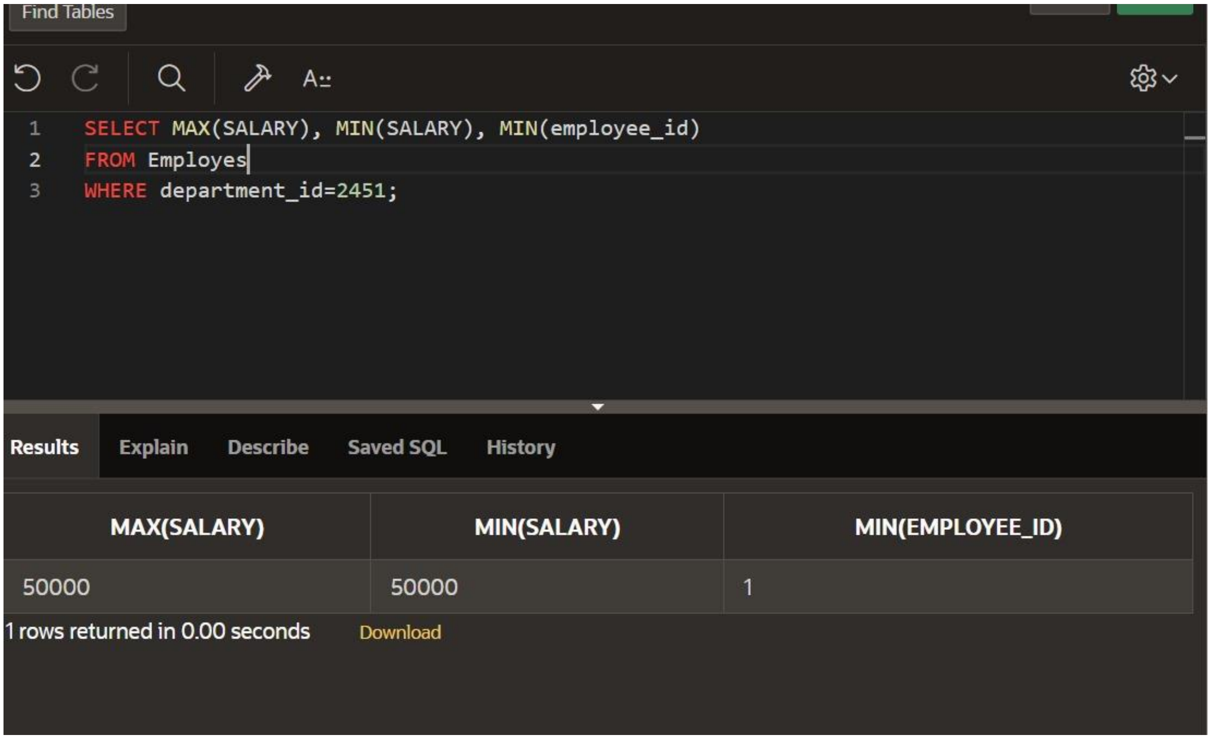
```
MAX(SALARY)
50000
```

1 rows returned in 0.01 seconds



```sql
SELECT last_name,first_name
from Employes
where salary=MIN(salary);
```

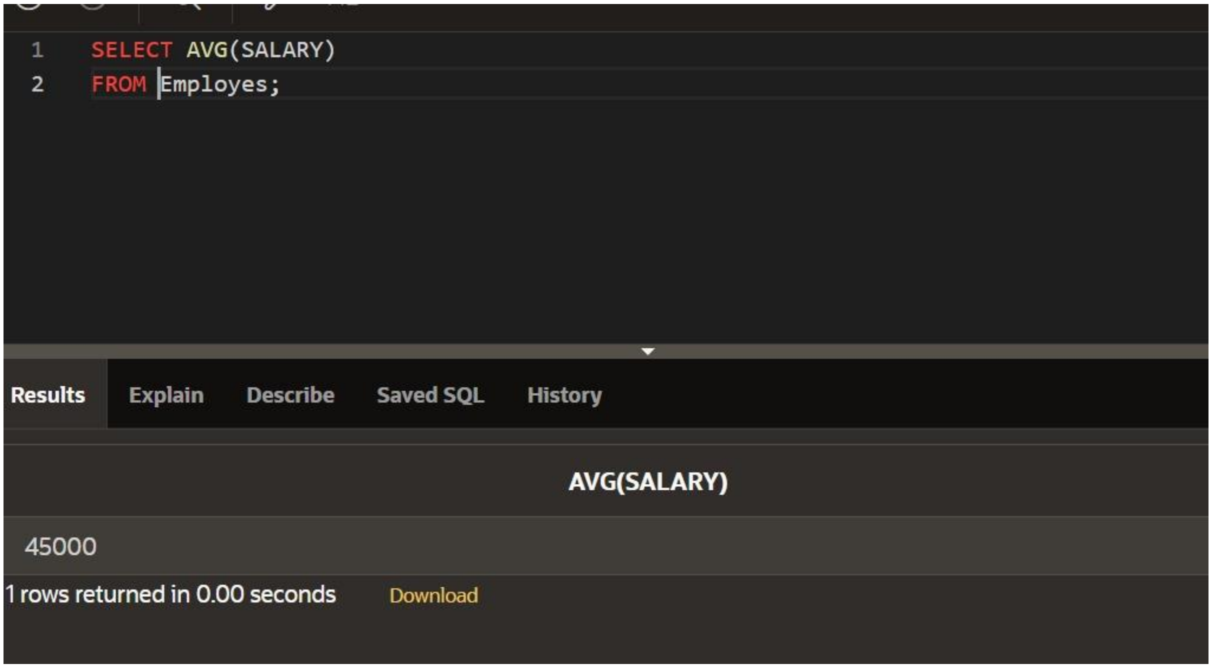
Error at line 3/14: ORA-00934: group function is not allowed here



```
SELECT MAX(SALARY), MIN(SALARY), MIN(employee_id)
FROM Employes
WHERE department_id=2451;
```

MAX(SALARY)	MIN(SALARY)	MIN(EMPLOYEE_ID)
50000	50000	1

1 rows returned in 0.00 seconds

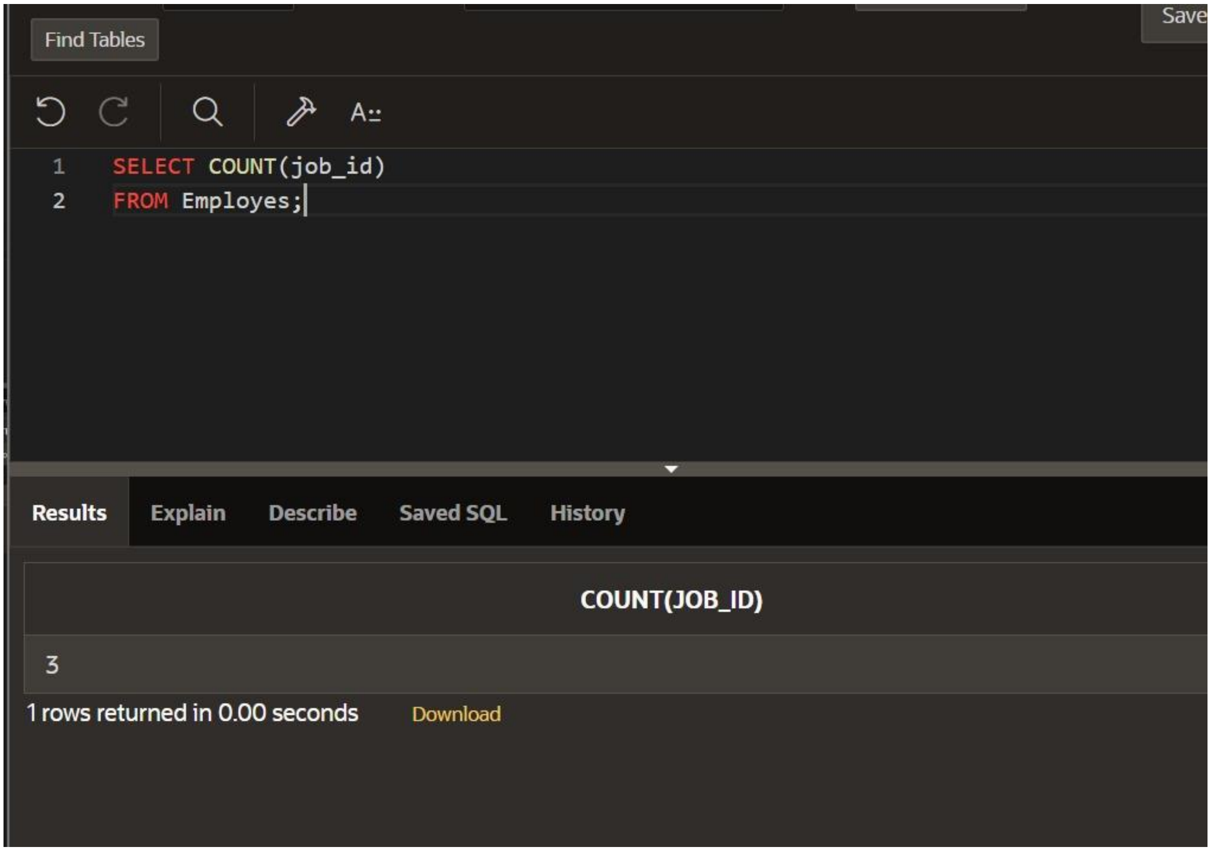


```
SELECT AVG(SALARY)
FROM Employes;
```

AVG(SALARY)
45000

1 rows returned in 0.00 seconds

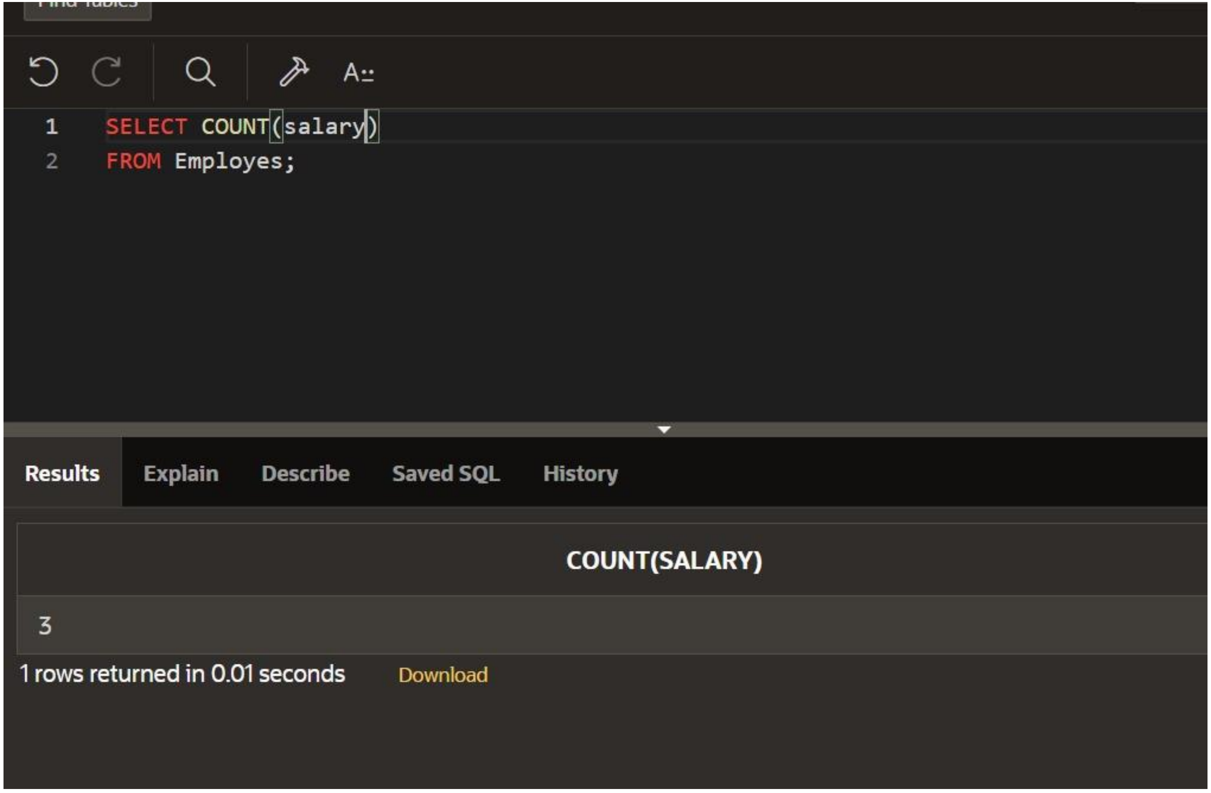
DP_8.2:



```sql
SELECT COUNT(job_id)
FROM Employes;
```

COUNT(JOB_ID)
3

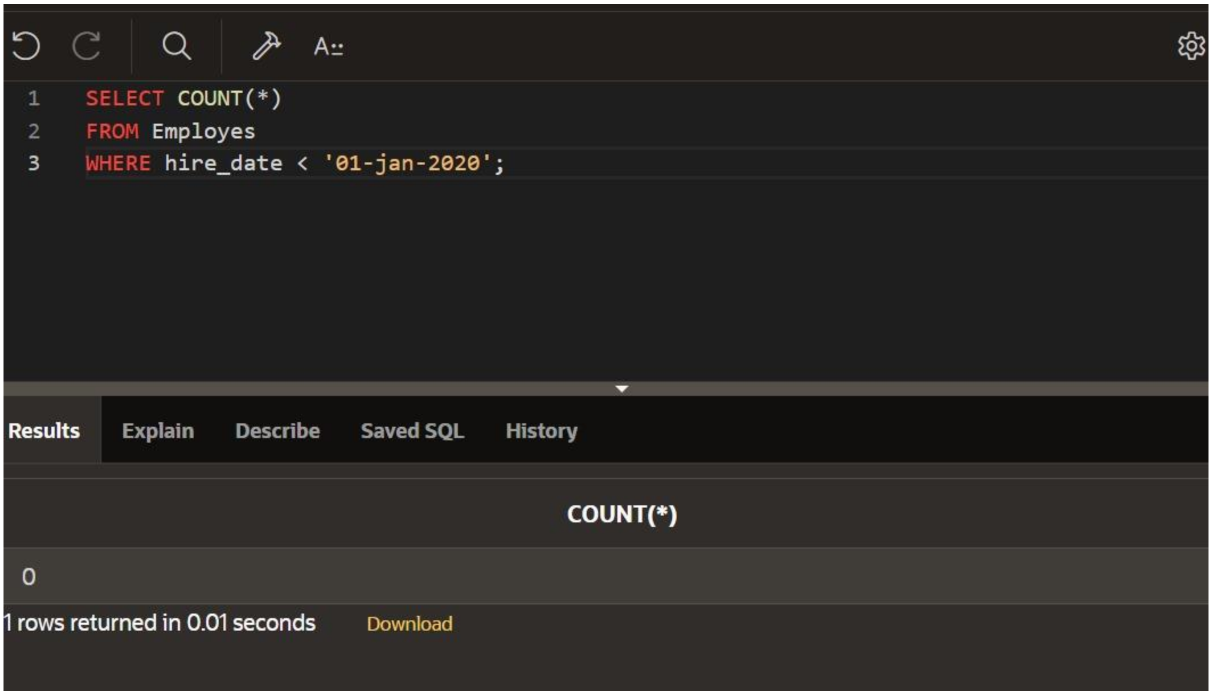
1 rows returned in 0.00 seconds



```sql
SELECT COUNT(salary)
FROM Employes;
```

COUNT(SALARY)
3

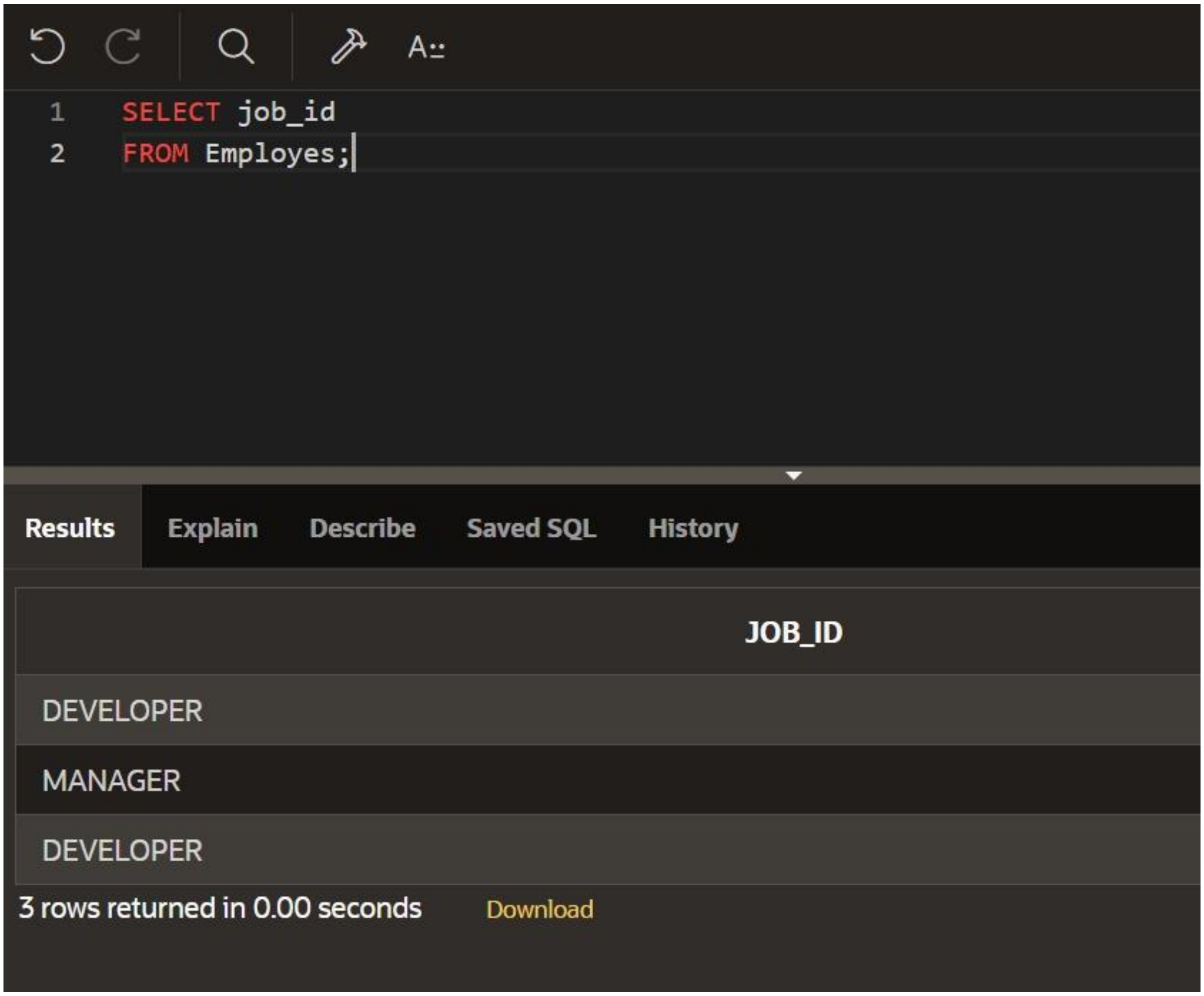
1 rows returned in 0.01 seconds



```sql
SELECT COUNT(*)
FROM Employes
WHERE hire_date < '01-jan-2020';
```

```
COUNT(*)
0
```

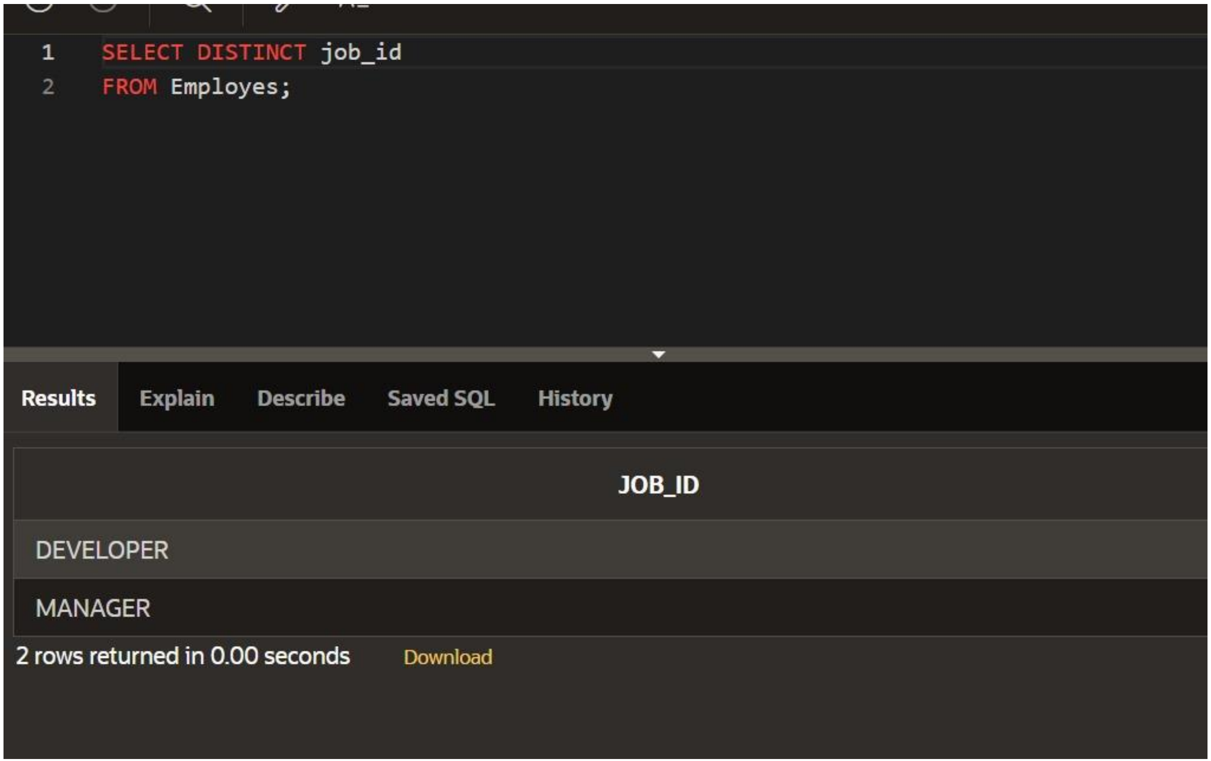
1 rows returned in 0.01 seconds



```sql
SELECT job_id
FROM Employes;
```

```
JOB_ID
DEVELOPER
MANAGER
DEVELOPER
```

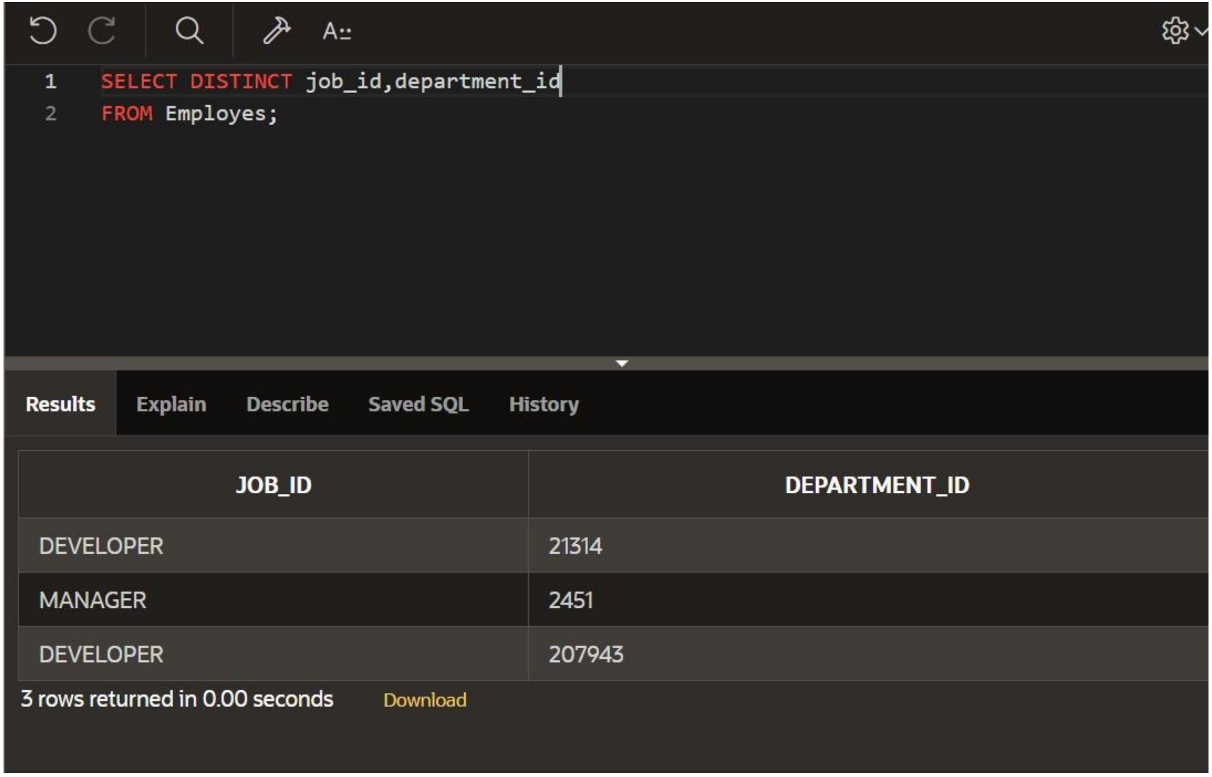
3 rows returned in 0.00 seconds



```sql
SELECT DISTINCT job_id
FROM Employes;
```

JOB_ID
DEVELOPER
MANAGER

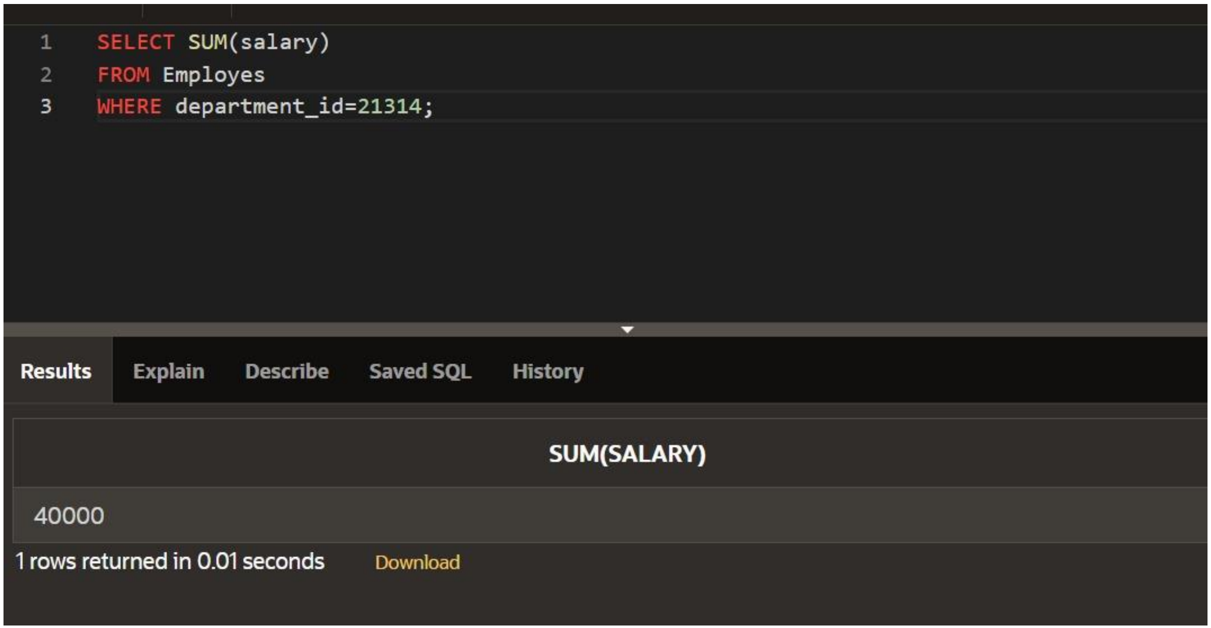
2 rows returned in 0.00 seconds Download



```sql
SELECT DISTINCT job_id,department_id
FROM Employes;
```

JOB_ID	DEPARTMENT_ID
DEVELOPER	21314
MANAGER	2451
DEVELOPER	207943

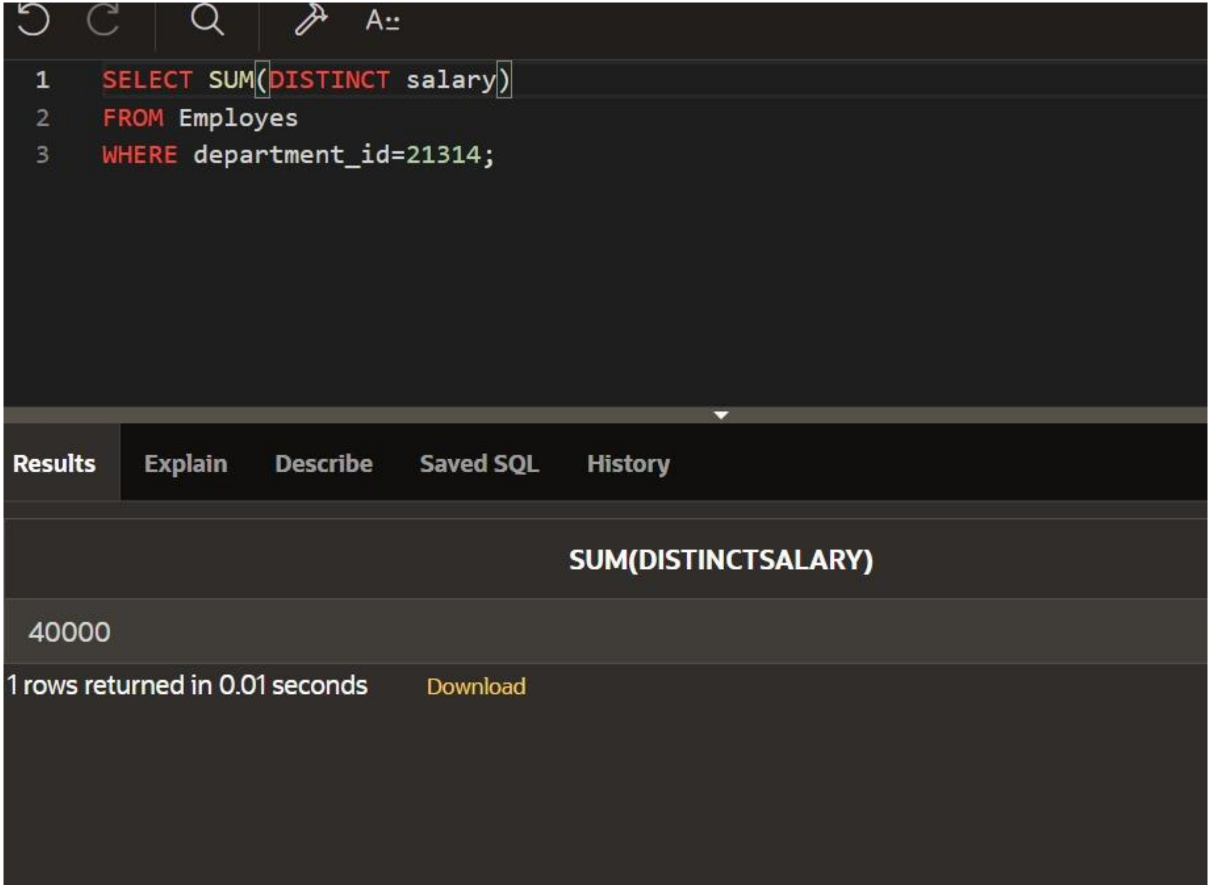
3 rows returned in 0.00 seconds Download



```sql
SELECT SUM(salary)
FROM Employes
WHERE department_id=21314;
```

SUM(SALARY)
40000

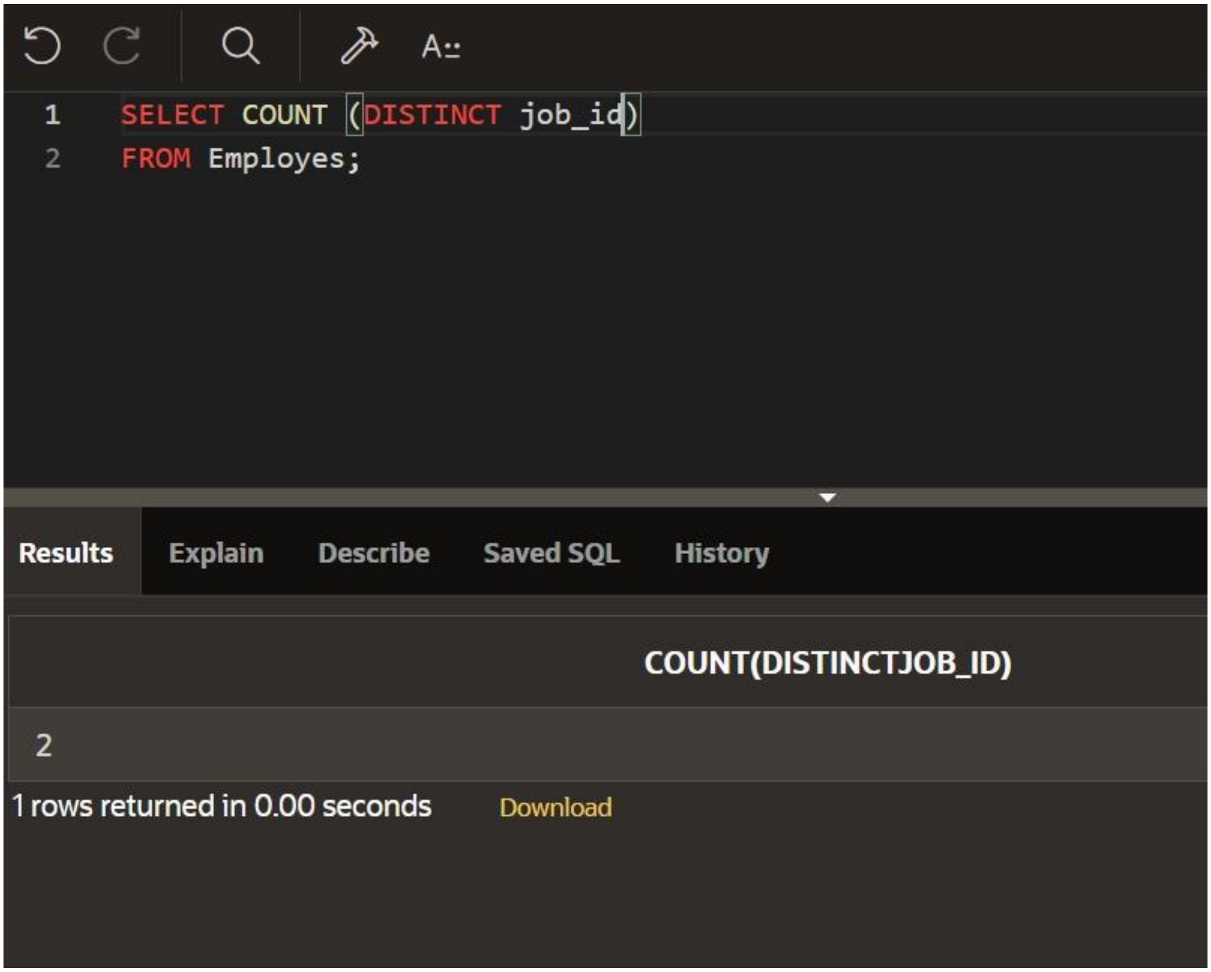
1 rows returned in 0.01 seconds



```sql
SELECT SUM(DISTINCT salary)
FROM Employes
WHERE department_id=21314;
```

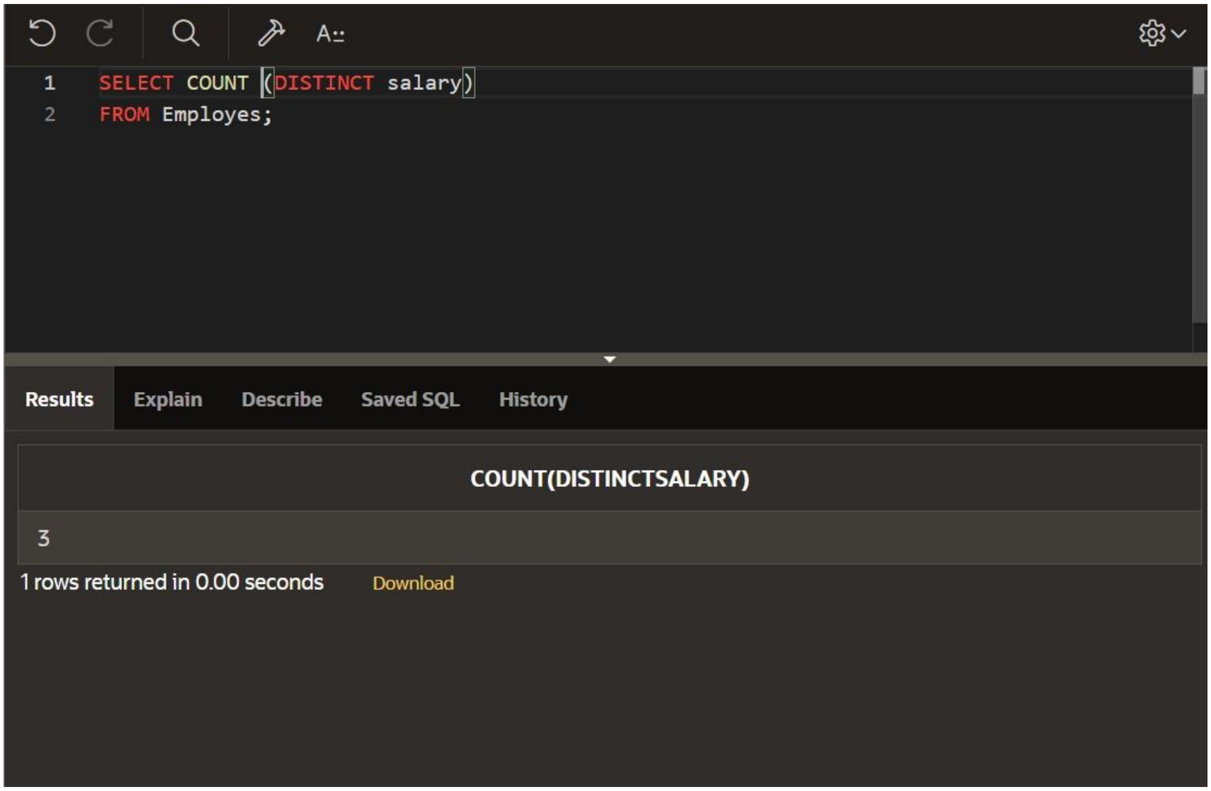
SUM(DISTINCTSALARY)
40000

1 rows returned in 0.01 seconds



```
SELECT COUNT (DISTINCT job_id)
FROM Employes;
```

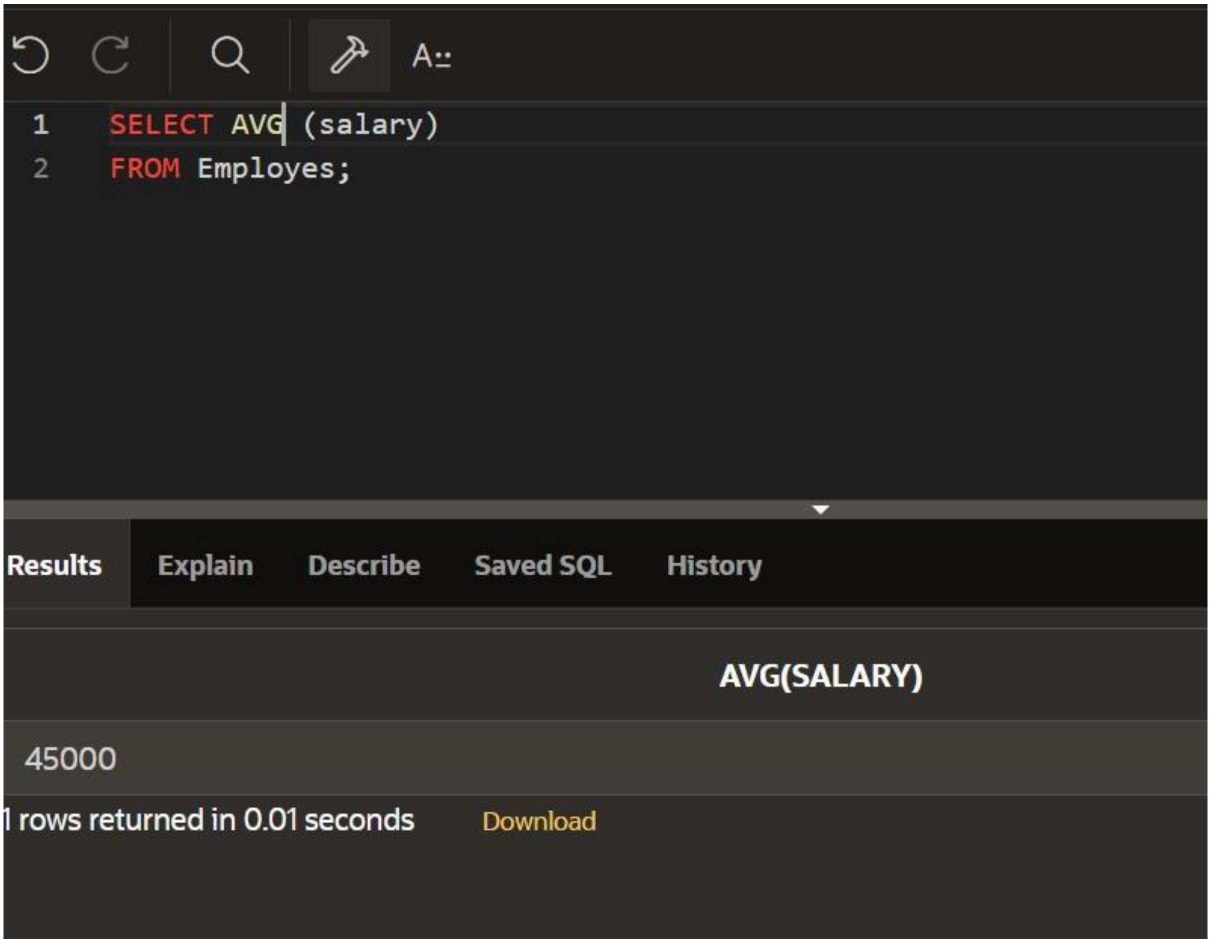
```
COUNT(DISTINCTJOB_ID)
2
1 rows returned in 0.00 seconds
```



```sql
SELECT COUNT (DISTINCT salary)
FROM Employes;
```

COUNT(DISTINCTSALARY)
3

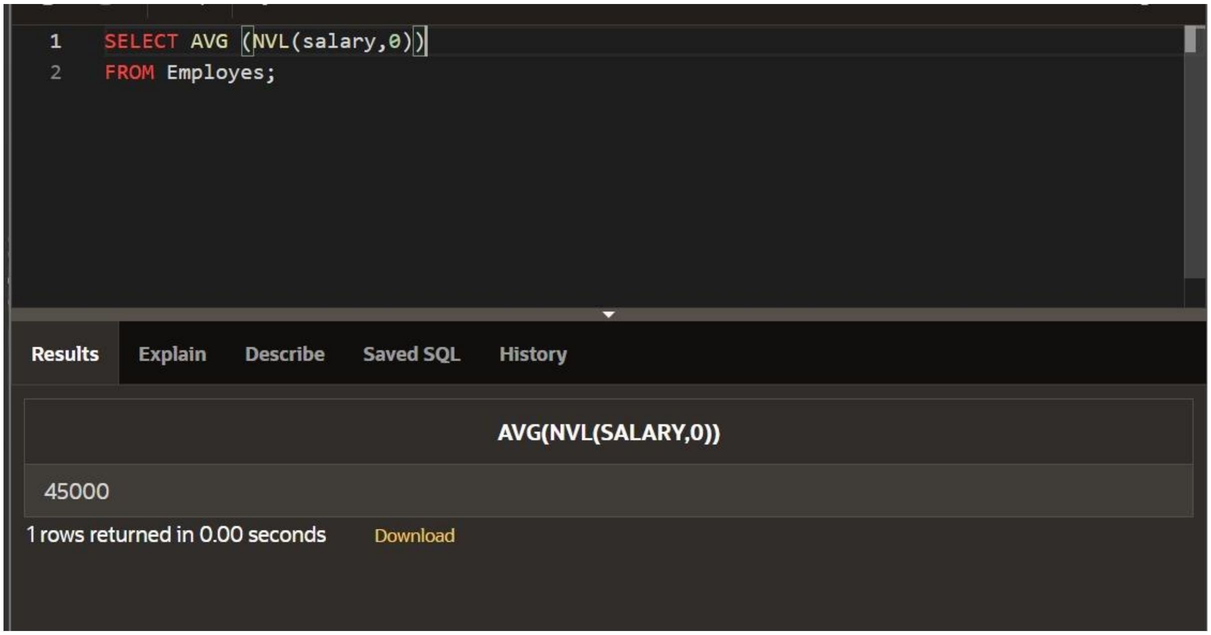
1 rows returned in 0.00 seconds   Download



```sql
SELECT AVG (salary)
FROM Employes;
```

AVG(SALARY)
45000

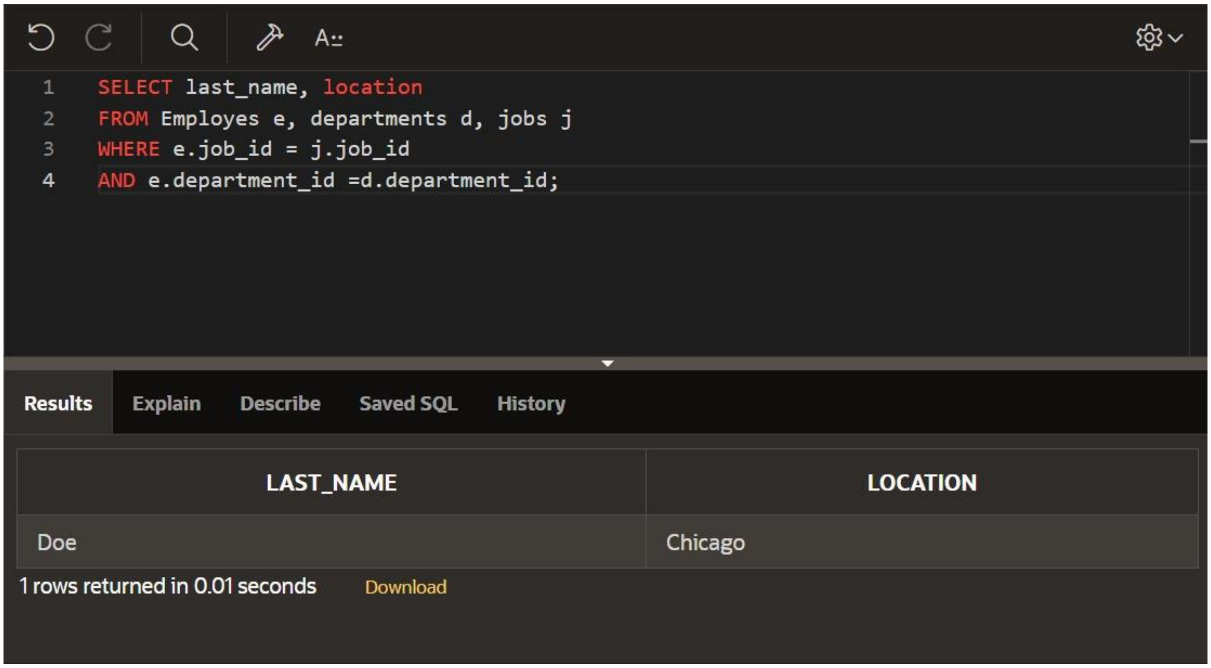
1 rows returned in 0.01 seconds   Download



```
SELECT AVG (NVL(salary,0))
FROM Employes;
```

AVG(NVL(SALARY,0))
45000

1 rows returned in 0.00 seconds



```
SELECT last_name, location
FROM Employes e, departments d, jobs j
WHERE e.job_id = j.job_id
AND e.department_id =d.department_id;
```

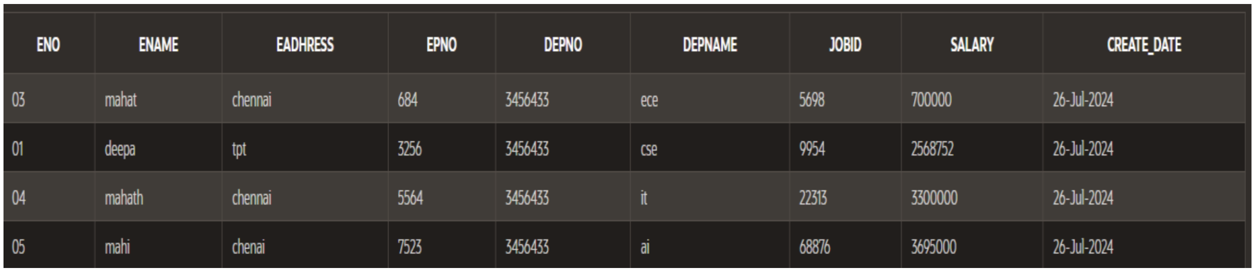
LAST_NAME	LOCATION
Doe	Chicago

1 rows returned in 0.01 seconds

DATABASE PROGRAMMING WITH SQL

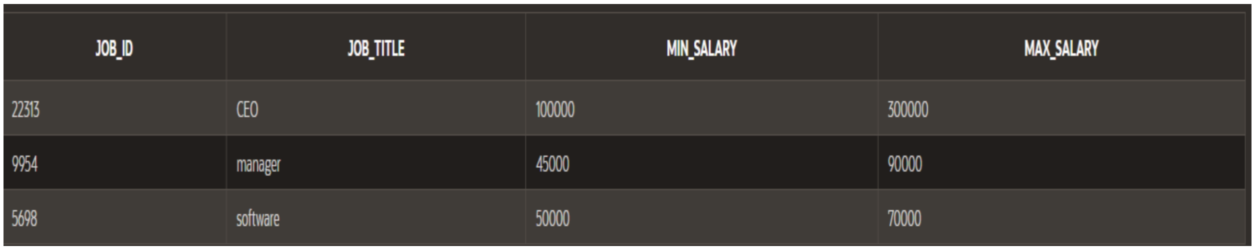
7.1 Oracle Equijoin and Cartesian Product

```
CREATE TABLE employ(
  eno VARCHAR(14),
  ename VARCHAR(14),
  eadhress VARCHAR(15),
  epno VARCHAR(15),
  depno VARCHAR(14),
  depname VARCHAR(14),
  jobid VARCHAR(10),
  salary VARCHAR(10),
  create_date DATE DEFAULT SYSDATE);
```



ENO	ENAME	EADHRESS	EPNO	DEPNO	DEPNAME	JOBID	SALARY	CREATE_DATE
03	mahat	chennai	684	3456433	ece	5698	700000	26-Jul-2024
01	deepa	tpt	3256	3456433	cse	9954	2568752	26-Jul-2024
04	mahath	chennai	5564	3456433	it	22313	3300000	26-Jul-2024
05	mahi	chenai	7523	3456433	ai	68876	3695000	26-Jul-2024

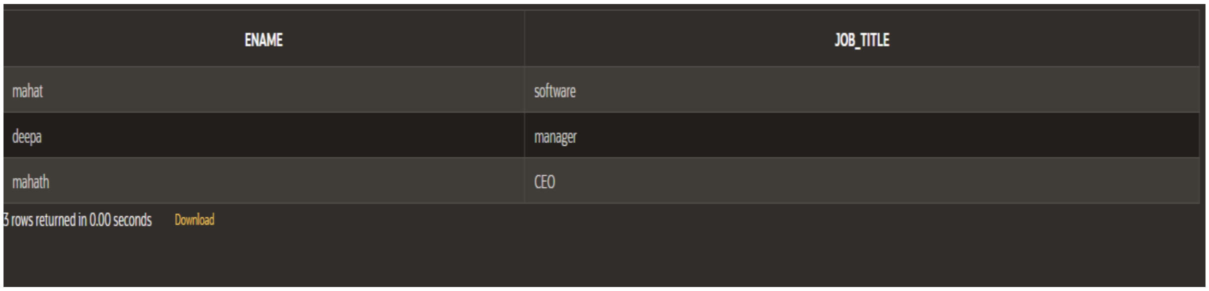
```
CREATE TABLE jobs (
   job_id VARCHAR(10) PRIMARY KEY,
   job_title VARCHAR(50) NOT NULL,
   min_salary DECIMAL(8, 2),
   max_salary DECIMAL(8, 2)
   );
```



JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
22313	CEO	100000	300000
9954	manager	45000	90000
5698	software	50000	70000

PROPRIETARY JOINS:

```
SELECT employ.ename, jobs.job_title
FROM employ,jobs
WHERE employ.jobid=jobs.job_id;
```



ENAME	JOB_TITLE
mahat	software
deepa	manager
mahath	CEO

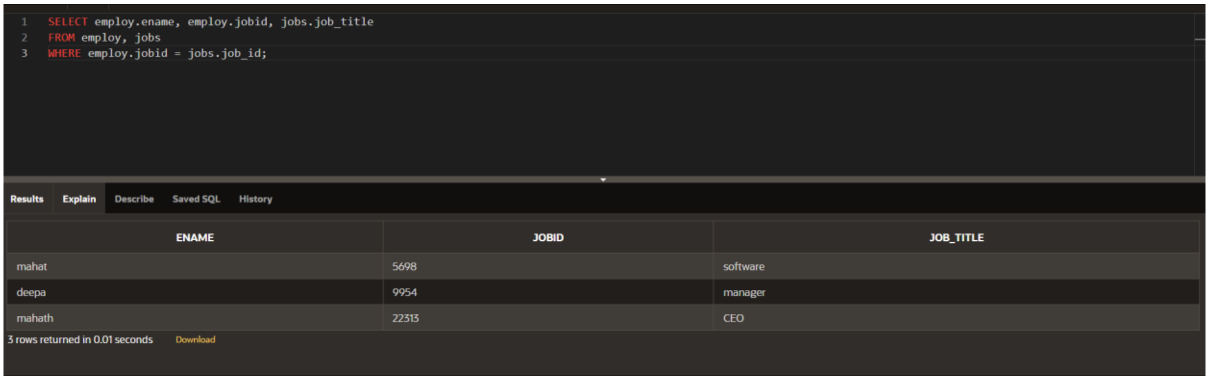
3 rows returned in 0.00 seconds

EQUIJOIN:

SELECT employ.ename, employ.jobid, jobs.job_title

FROM employ, jobs

WHERE employ.jobid = jobs.job_id;



```
SELECT employ.ename, employ.jobid, jobs.job_title
FROM employ, jobs
WHERE employ.jobid = jobs.job_id;
```

ENAME	JOBID	JOB_TITLE
mahat	5698	software
deepa	9954	manager
mahath	22313	CEO

3 rows returned in 0.01 seconds

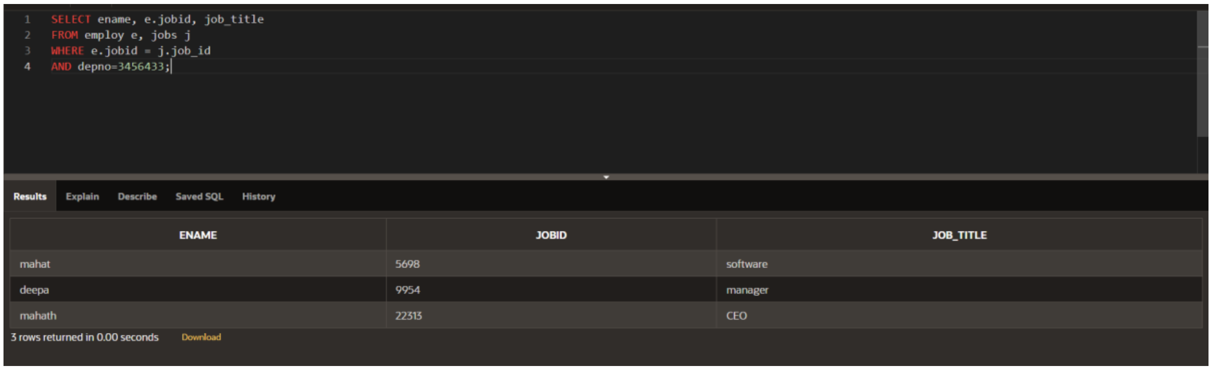
ALIASES:

SELECT ename, e.jobid, job_title

FROM employ e, jobs j

WHERE e.jobid = j.job_id

AND depno=3456433;



```
SELECT ename, e.jobid, job_title
FROM employ e, jobs j
WHERE e.jobid = j.job_id
AND depno=3456433;
```

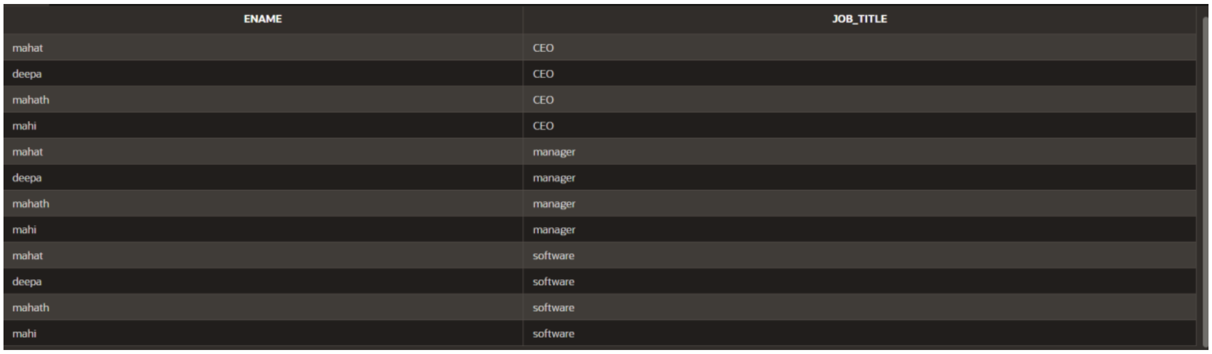
ENAME	JOBID	JOB_TITLE
mahat	5698	software
deepa	9954	manager
mahath	22313	CEO

3 rows returned in 0.00 seconds

CARTESIAN PRODUCT JOIN:

SELECT employ.ename,jobs.job_title

FROM employ,jobs;



ENAME	JOB_TITLE
mahat	CEO
deepa	CEO
mahath	CEO
mahi	CEO
mahat	manager
deepa	manager
mahath	manager
mahi	manager
mahat	software
deepa	software
mahath	software
mahi	software

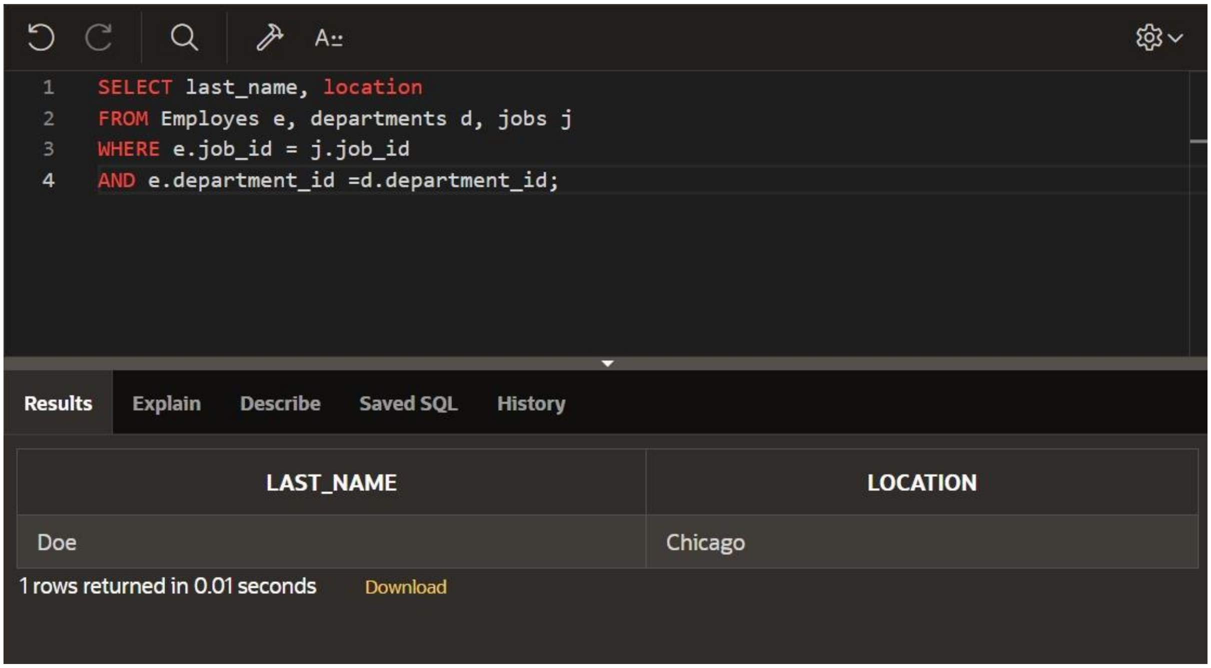
JOIN:

SELECT last_name, location

FROM Employes e, departments d, jobs j

WHERE e.job_id = j.job_id

AND e.department_id =d.department_id;



```
SELECT last_name, location
FROM Employes e, departments d, jobs j
WHERE e.job_id = j.job_id
AND e.department_id =d.department_id;
```

LAST_NAME	LOCATION
Doe	Chicago

1 rows returned in 0.01 seconds Download

DP_7.2:

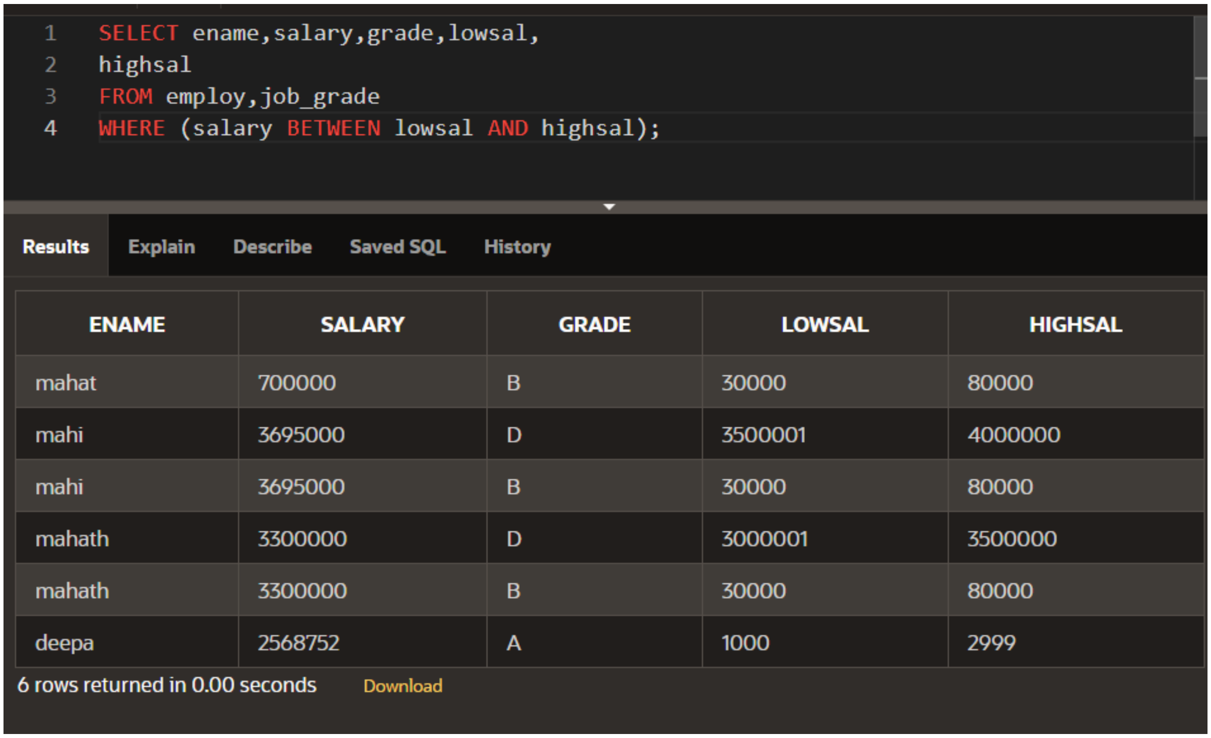
Nonequijoin:

SELECT ename,salary,grade,lowsal,

highsal

FROM employ,job_grade

WHERE (salary BETWEEN lowsal AND highsal);



```
SELECT ename,salary,grade,lowsal,
highsal
FROM employ,job_grade
WHERE (salary BETWEEN lowsal AND highsal);
```

ENAME	SALARY	GRADE	LOWSAL	HIGHSAL
mahat	700000	B	30000	80000
mahi	3695000	D	3500001	4000000
mahi	3695000	B	30000	80000
mahath	3300000	D	3000001	3500000
mahath	3300000	B	30000	80000
deepa	2568752	A	1000	2999

6 rows returned in 0.00 seconds

Section 6

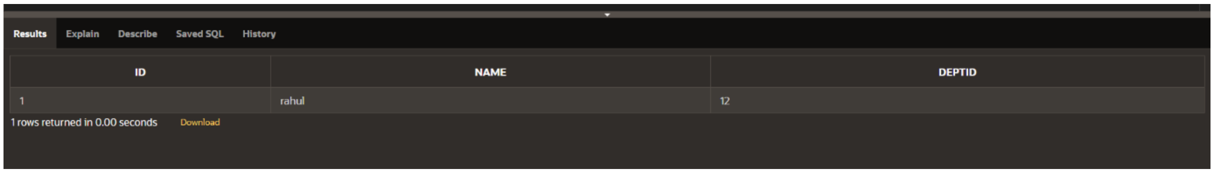
HEARIACHEY LEVEL

select level id,name,deptid

from emp

start with id=100

connect by prior id=deptid;



ID	NAME	DEPTID
1	rahul	12

1 rows returned in 0.00 seconds

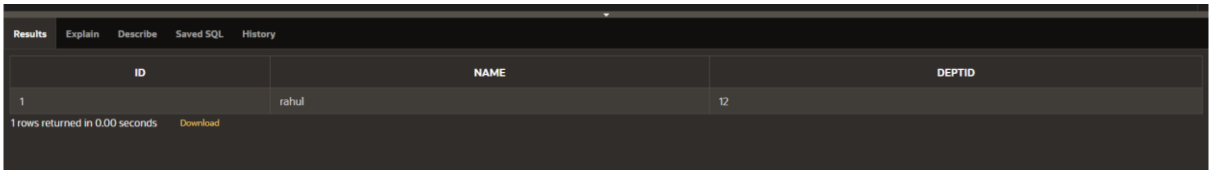
HIERACHIAL USING START WITH KEYWORD

select id,name,deptid

from emp

start with id=100

connect by prior id=deptid;



ID	NAME	DEPTID
1	rahul	12

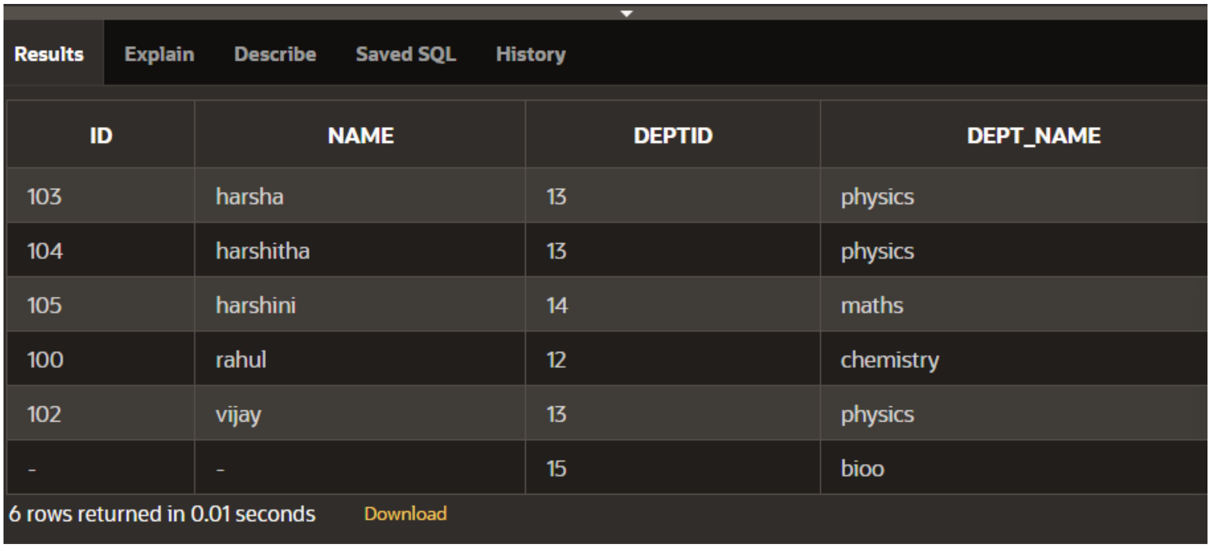
1 rows returned in 0.00 seconds

FULL OUTER JOIN

select e.id,e.name,d.deptid,d.dept_name

from emp e full outer join dept d

on (e.deptid=d.deptid);



ID	NAME	DEPTID	DEPT_NAME
103	harsha	13	physics
104	harshitha	13	physics
105	harshini	14	maths
100	rahul	12	chemistry
102	vijay	13	physics
-	-	15	bioo

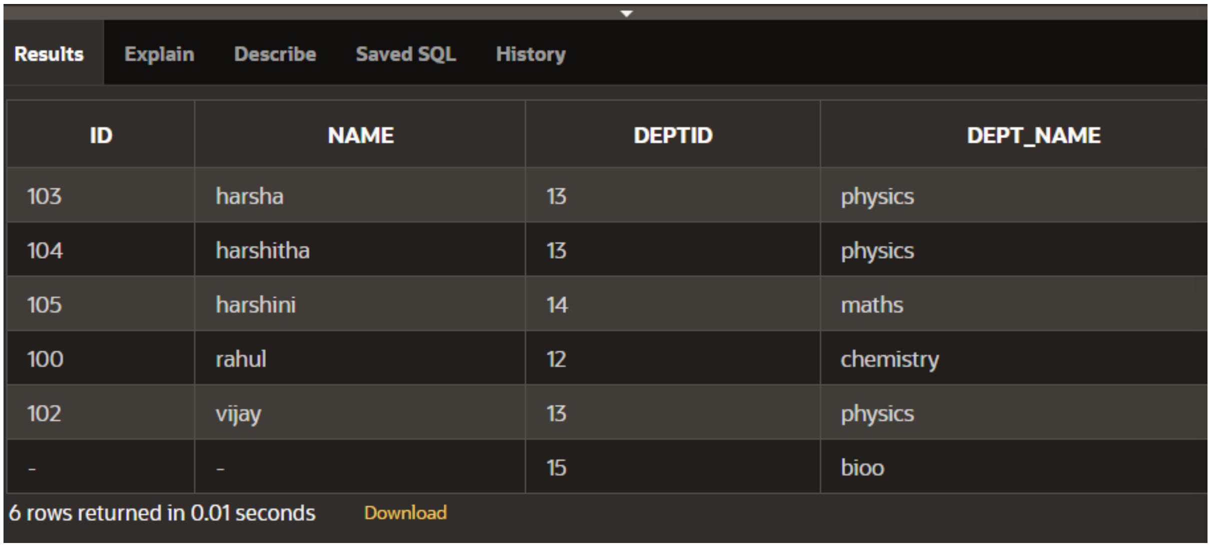
6 rows returned in 0.01 seconds

RIGHT OUTER JOIN

select e.id,e.name,d.deptid,d.dept_name

from emp e right outer join dept d

on (e.deptid=d.deptid);



ID	NAME	DEPTID	DEPT_NAME
103	harsha	13	physics
104	harshitha	13	physics
105	harshini	14	maths
100	rahul	12	chemistry
102	vijay	13	physics
-	-	15	bioo

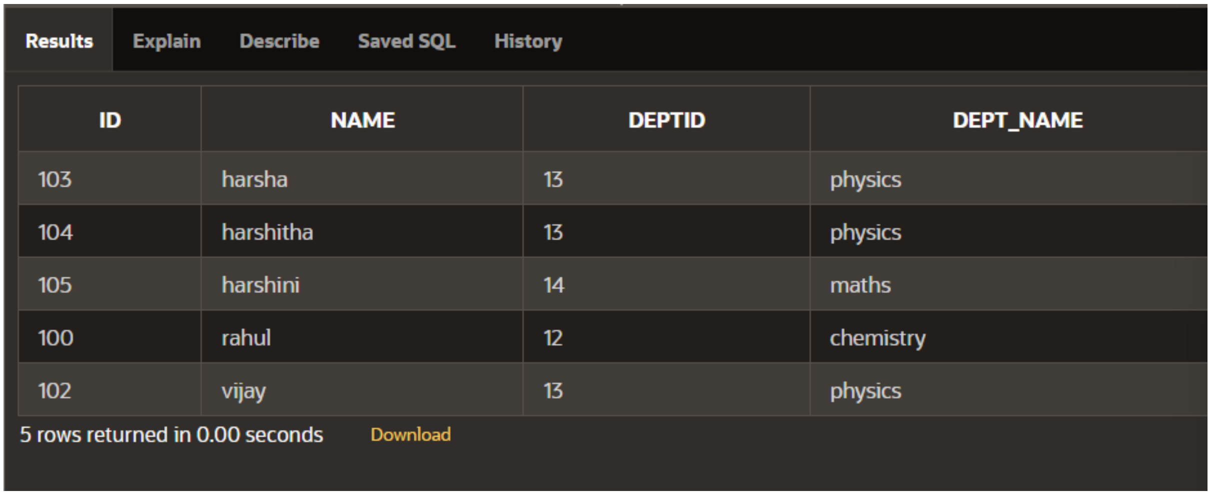
6 rows returned in 0.01 seconds

LEFT OUTER JOIN

select e.id,e.name,d.deptid,d.dept_name

from emp e left outer join dept d

on (e.deptid=d.deptid);



ID	NAME	DEPTID	DEPT_NAME
103	harsha	13	physics
104	harshitha	13	physics
105	harshini	14	maths
100	rahul	12	chemistry
102	vijay	13	physics

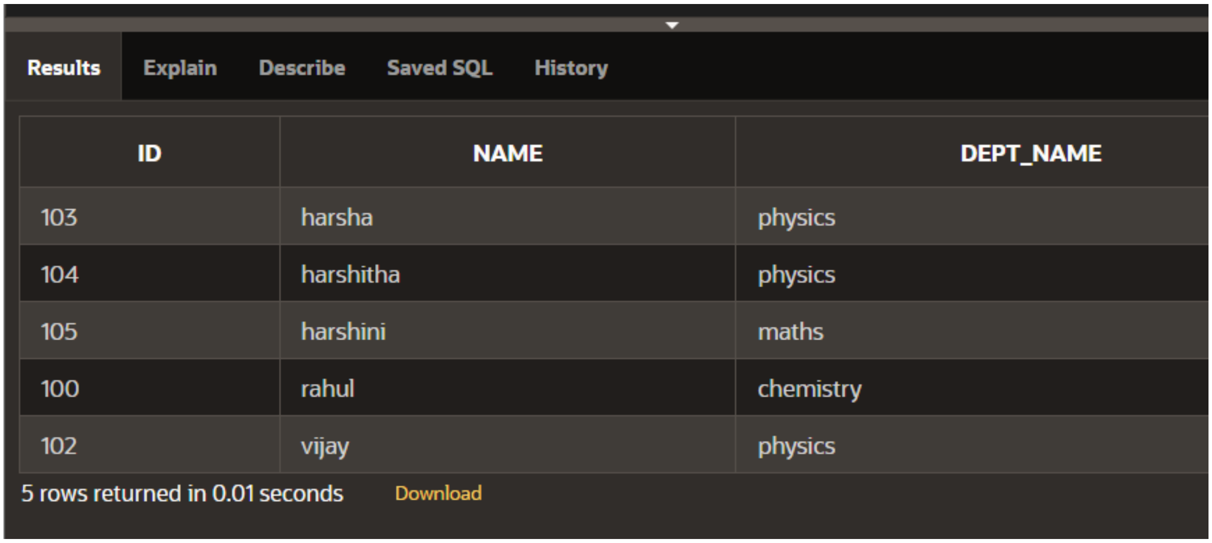
5 rows returned in 0.00 seconds

ON CLAUSE

select id,name,dept_name

from emp e join dept d

on(e.deptid=d.deptid);



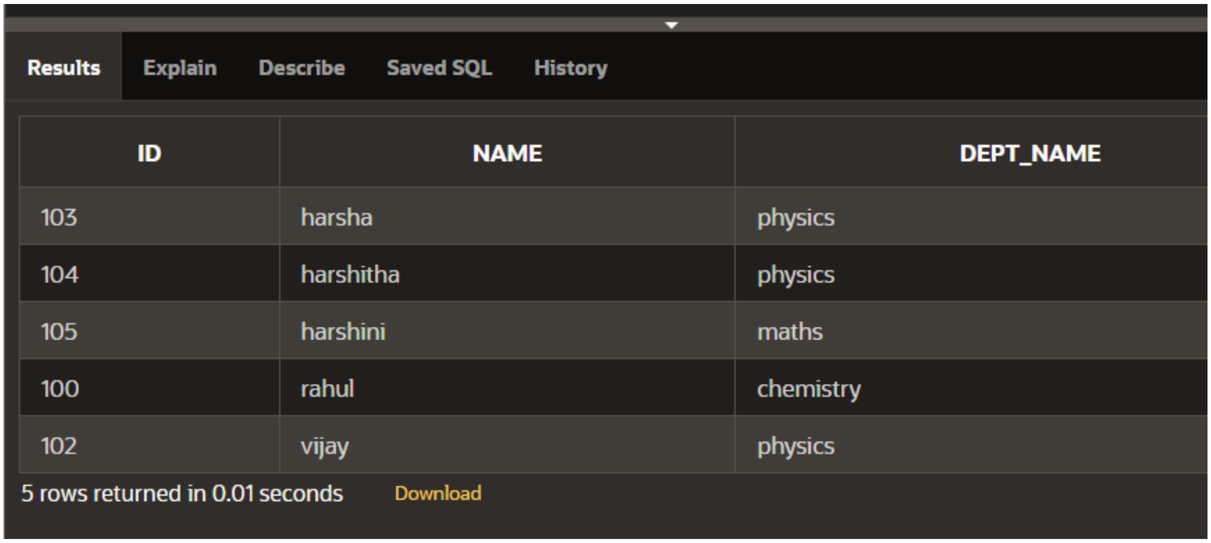
ID	NAME	DEPT_NAME
103	harsha	physics
104	harshitha	physics
105	harshini	maths
100	rahul	chemistry
102	vijay	physics

5 rows returned in 0.01 seconds

USING CLAUSE

select id,name,deptid,dept_name

from emp join dept using (deptid);



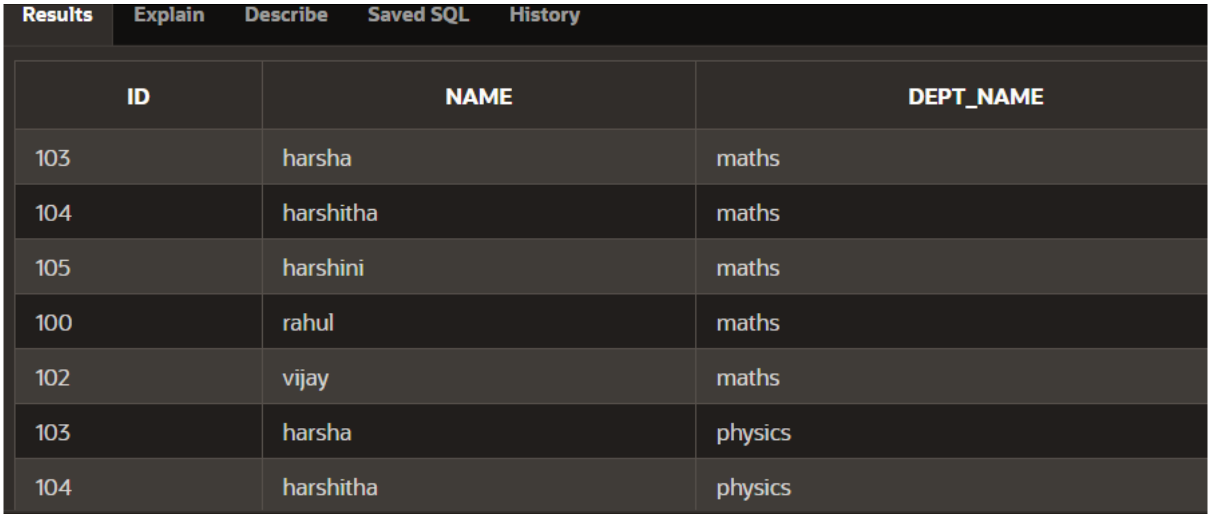
ID	NAME	DEPT_NAME
103	harsha	physics
104	harshitha	physics
105	harshini	maths
100	rahul	chemistry
102	vijay	physics

5 rows returned in 0.01 seconds

CROSS JOIN

select id,name,dept_name

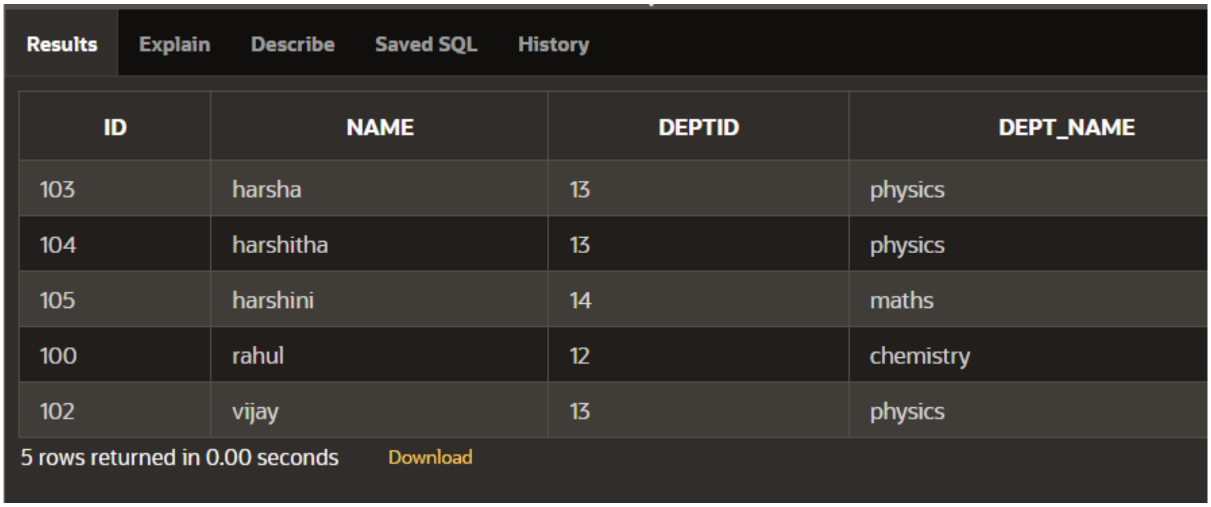
from emp cross join dept;



ID	NAME	DEPT_NAME
103	harsha	maths
104	harshitha	maths
105	harshini	maths
100	rahul	maths
102	vijay	maths
103	harsha	physics
104	harshitha	physics

NATURAL JOIN

```
select id,name,deptid,dept_name

from emp natural join dept;
```



ID	NAME	DEPTID	DEPT_NAME
103	harsha	13	physics
104	harshitha	13	physics
105	harshini	14	maths
100	rahul	12	chemistry
102	vijay	13	physics

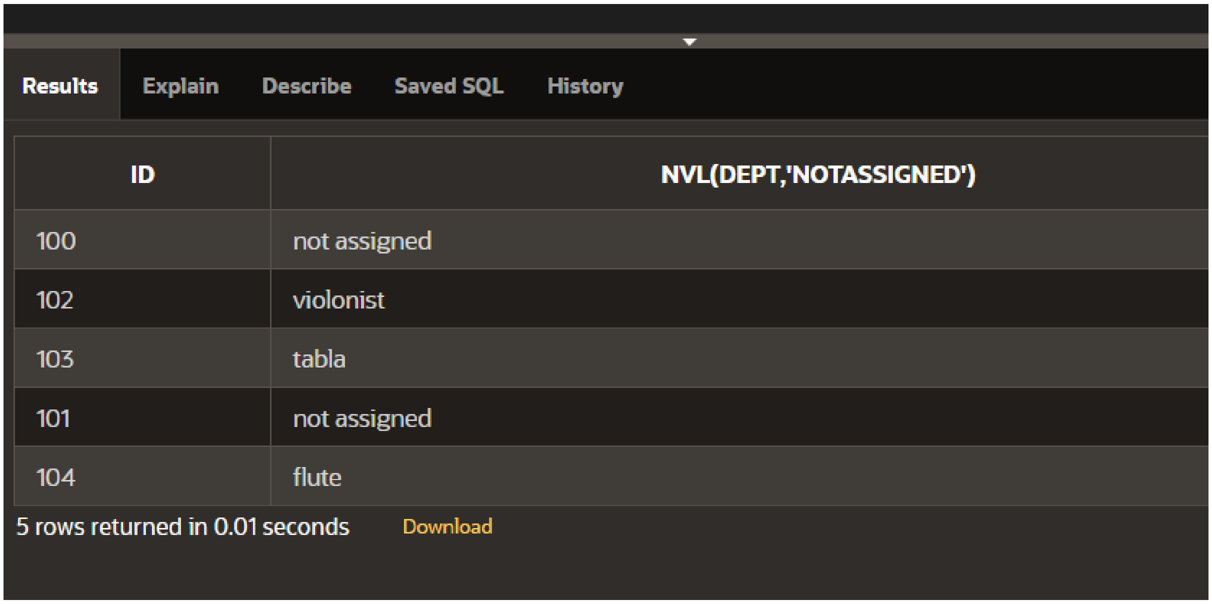
5 rows returned in 0.00 seconds

Section 5

NVL FUNCTION

select id,nvl(dept,'not assigned')

from singer;



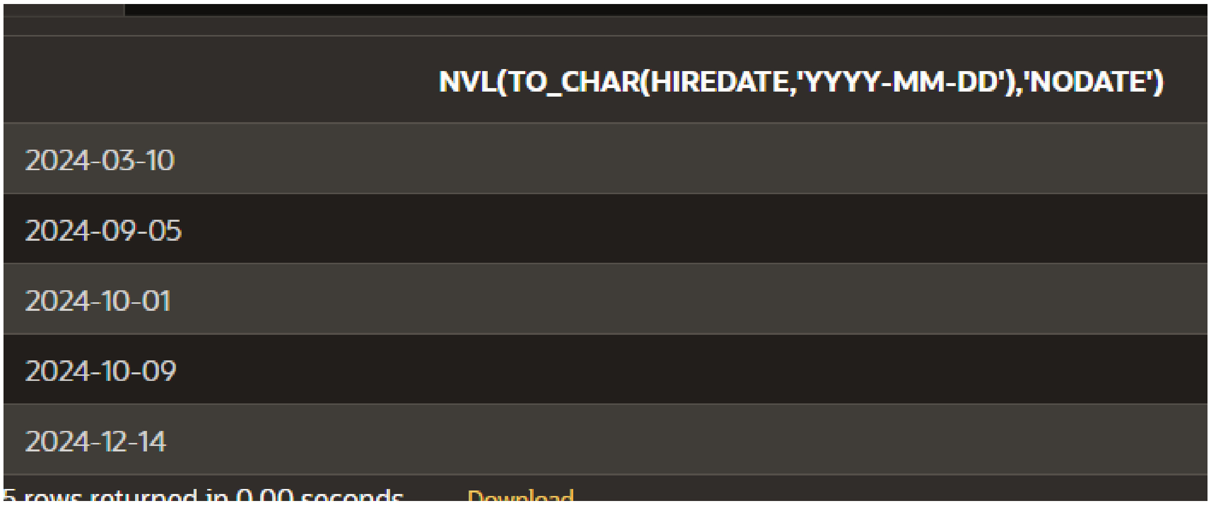
ID	NVL(DEPT,'NOTASSIGNED')
100	not assigned
102	violonist
103	tabla
101	not assigned
104	flute

5 rows returned in 0.01 seconds

NVL DATE

SELECT NVL(TO_CHAR(hiredate, 'YYYY-MM-DD'), 'no date')

FROM singer;

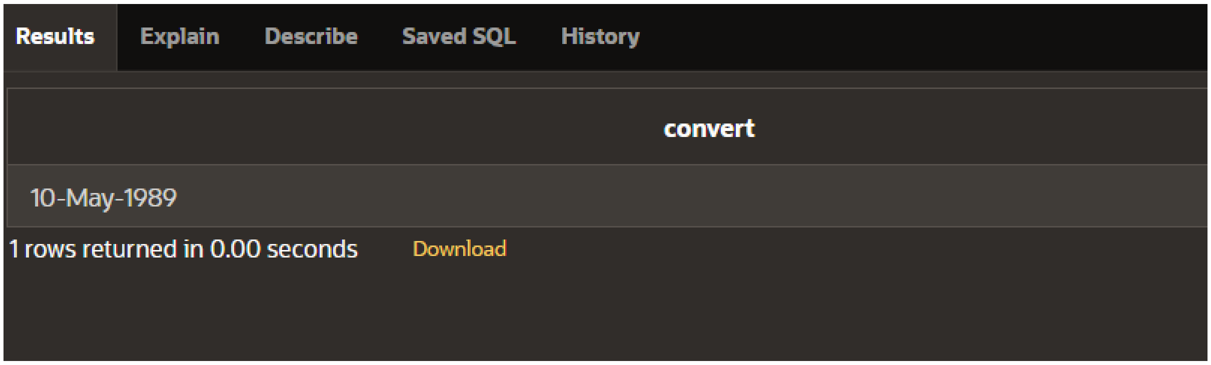


NVL(TO_CHAR(HIREDATE,'YYYY-MM-DD'),'NODATE')
2024-03-10
2024-09-05
2024-10-01
2024-10-09
2024-12-14

5 rows returned in 0.00 seconds

CHARACTER TO DATE

```
select to_date('may10,1989','fxmondd,yyyy') as "convert"

from dual;
```

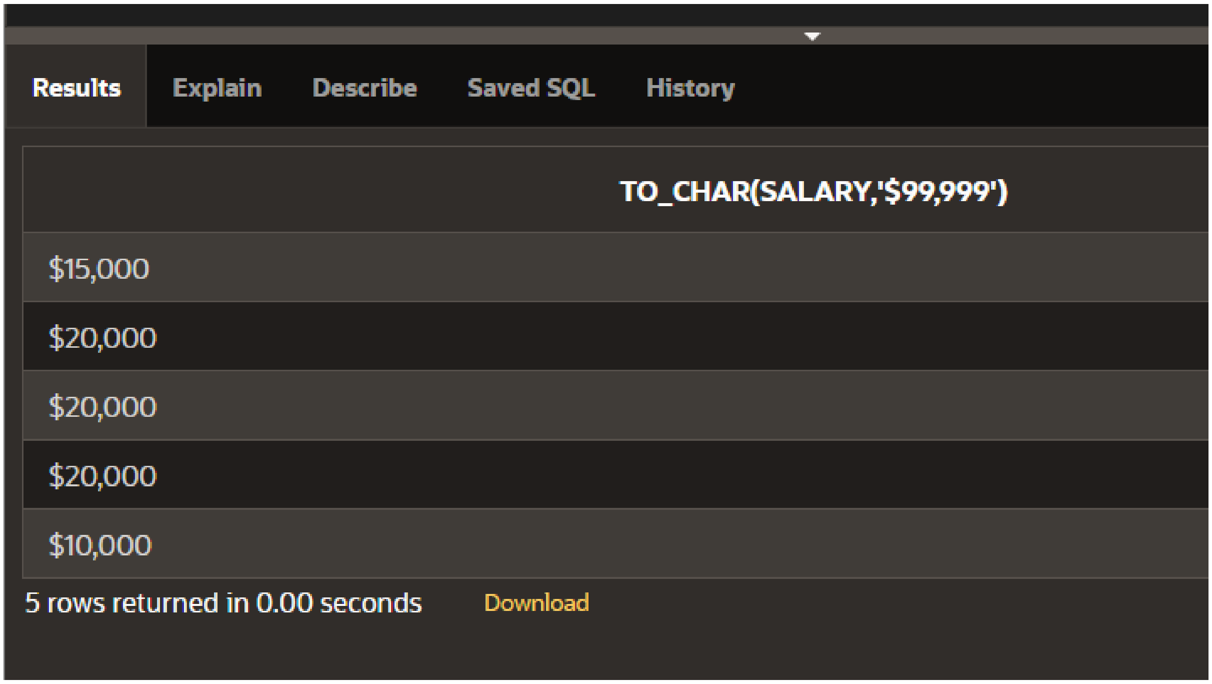


convert
10-May-1989

1 rows returned in 0.00 seconds

NUMBER TO CHARACTER

```
select to_char(salary,'$99,999')

from singer;
```



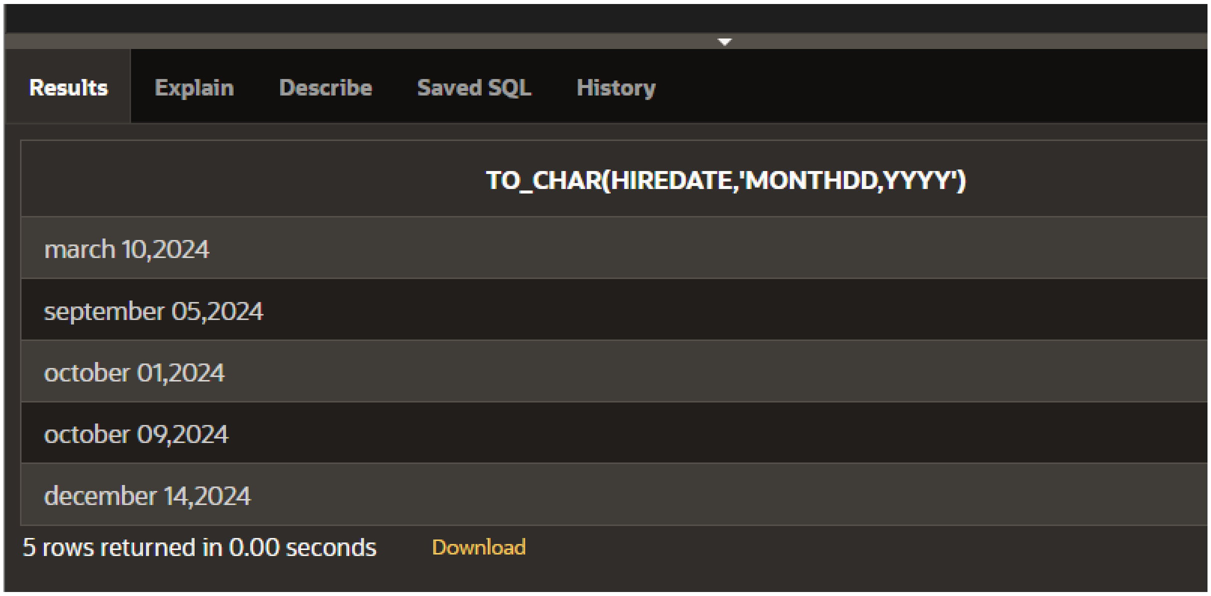
TO_CHAR(SALARY,'$99,999')
$15,000
$20,000
$20,000
$20,000
$10,000

5 rows returned in 0.00 seconds

DATE TO CHARACTER

```
select to_char(hiredate,'month dd,yyyy')
```

from singer;



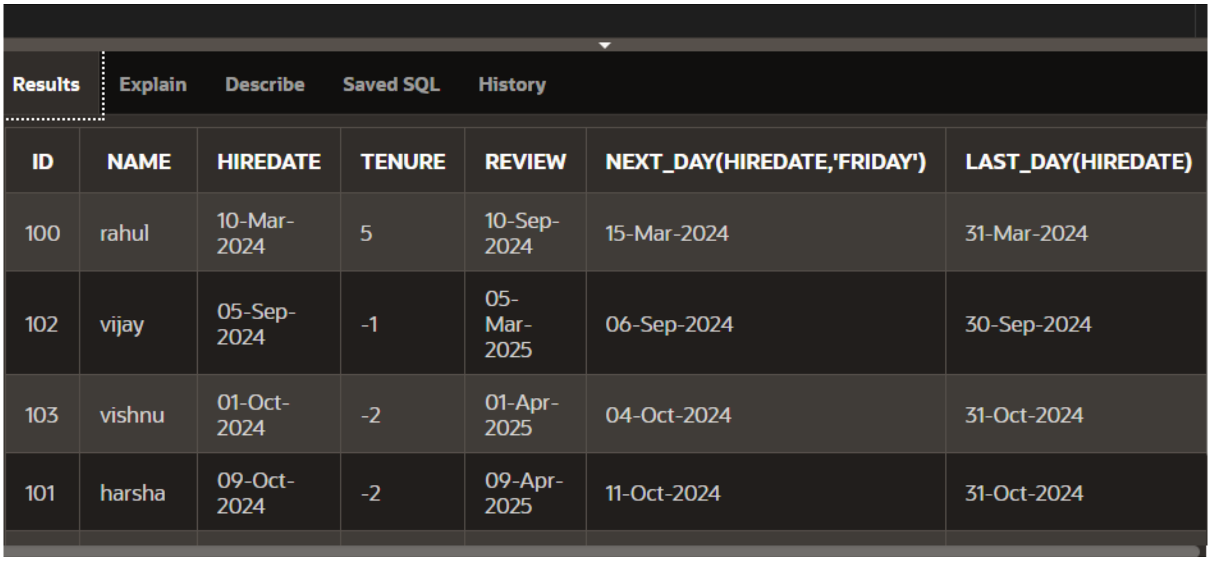
TO_CHAR(HIREDATE,'MONTHDD,YYYY')
march 10,2024
september 05,2024
october 01,2024
october 09,2024
december 14,2024

5 rows returned in 0.00 seconds

DATE FUNCTION

select id,name,hiredate,round(months_between(sysdate,hiredate)) as tenure, add_months(hiredate,6) as review, next_day(hiredate,'friday'),last_day(hiredate)

from singer;



ID	NAME	HIREDATE	TENURE	REVIEW	NEXT_DAY(HIREDATE,'FRIDAY')	LAST_DAY(HIREDATE)
100	rahul	10-Mar-2024	5	10-Sep-2024	15-Mar-2024	31-Mar-2024
102	vijay	05-Sep-2024	-1	05-Mar-2025	06-Sep-2024	30-Sep-2024
103	vishnu	01-Oct-2024	-2	01-Apr-2025	04-Oct-2024	31-Oct-2024
101	harsha	09-Oct-2024	-2	09-Apr-2025	11-Oct-2024	31-Oct-2024

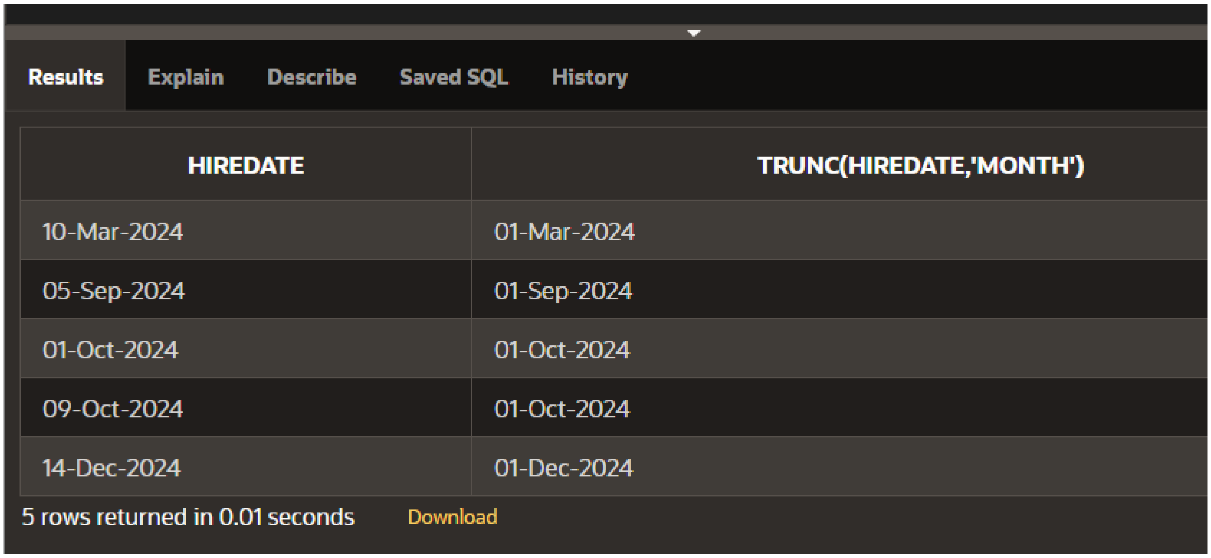
Section 4

TRUNC

select hiredate,

trunc(hiredate,'month')

from singer;



HIREDATE	TRUNC(HIREDATE,'MONTH')
10-Mar-2024	01-Mar-2024
05-Sep-2024	01-Sep-2024
01-Oct-2024	01-Oct-2024
09-Oct-2024	01-Oct-2024
14-Dec-2024	01-Dec-2024

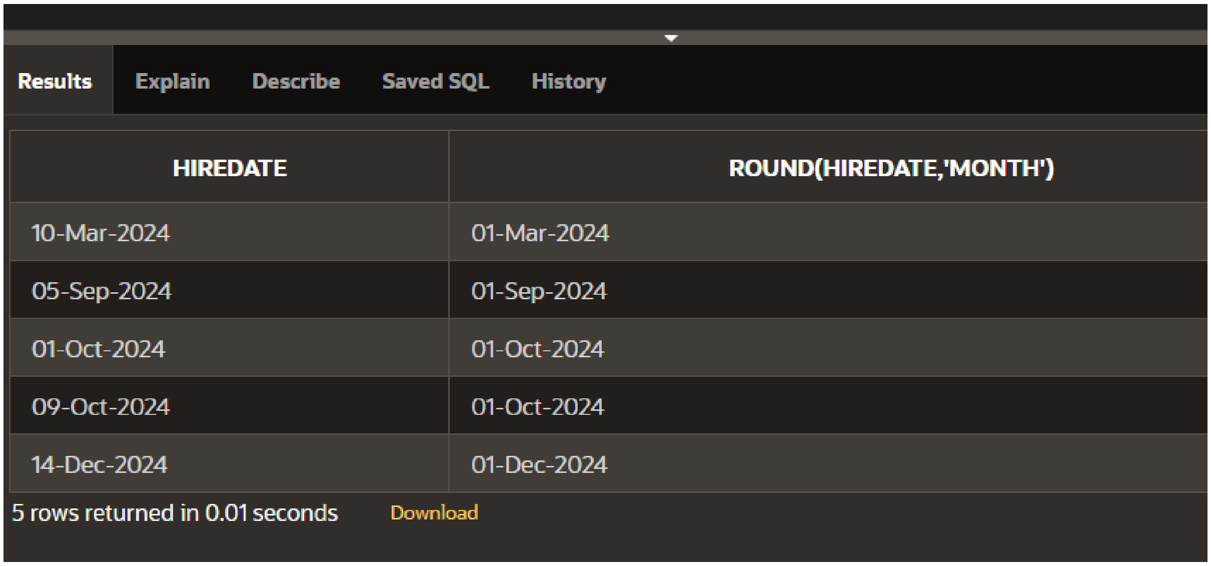
5 rows returned in 0.01 seconds

ROUND TO HIREDATE

select hiredate,

round(hiredate,'month')

from singer;



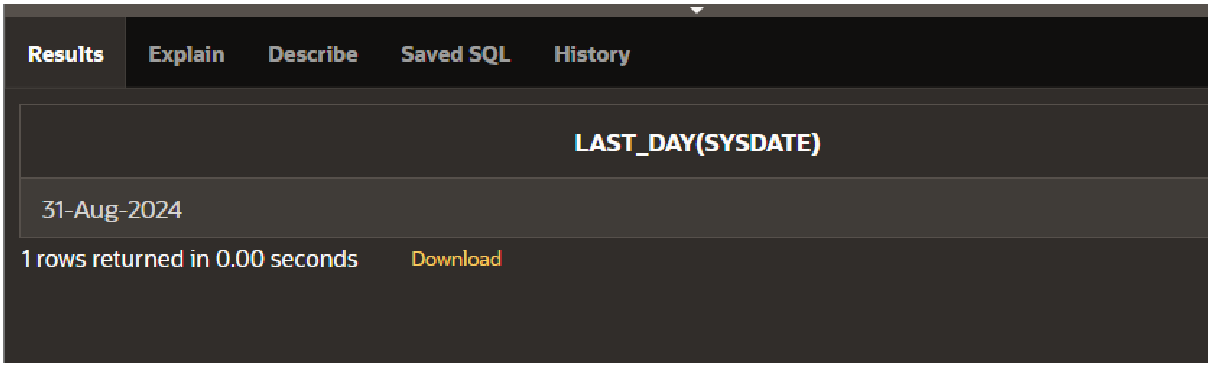
HIREDATE	ROUND(HIREDATE,'MONTH')
10-Mar-2024	01-Mar-2024
05-Sep-2024	01-Sep-2024
01-Oct-2024	01-Oct-2024
09-Oct-2024	01-Oct-2024
14-Dec-2024	01-Dec-2024

5 rows returned in 0.01 seconds

LAST DAY

select last_day(sysdate)

from dual;



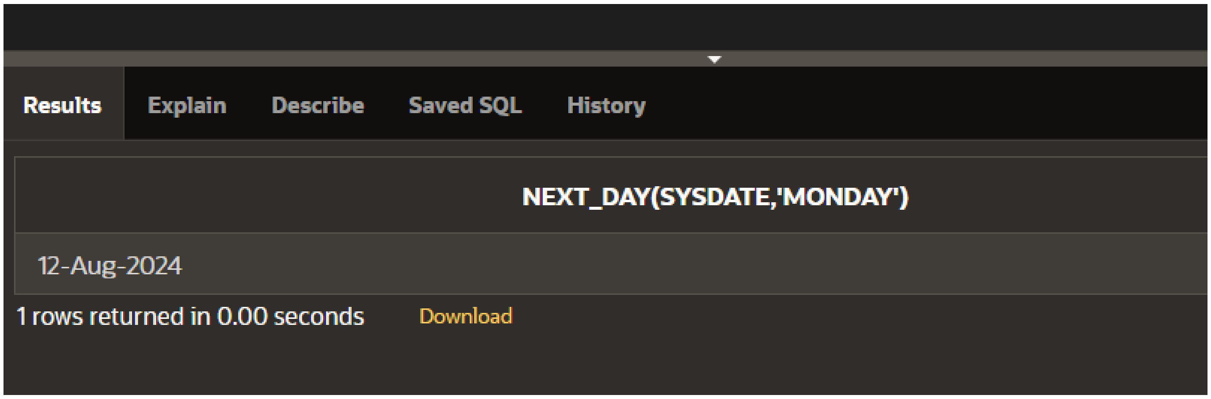
LAST_DAY(SYSDATE)
31-Aug-2024

1 rows returned in 0.00 seconds

NEXT DAY

select next_day(sysdate,'monday')

from dual;



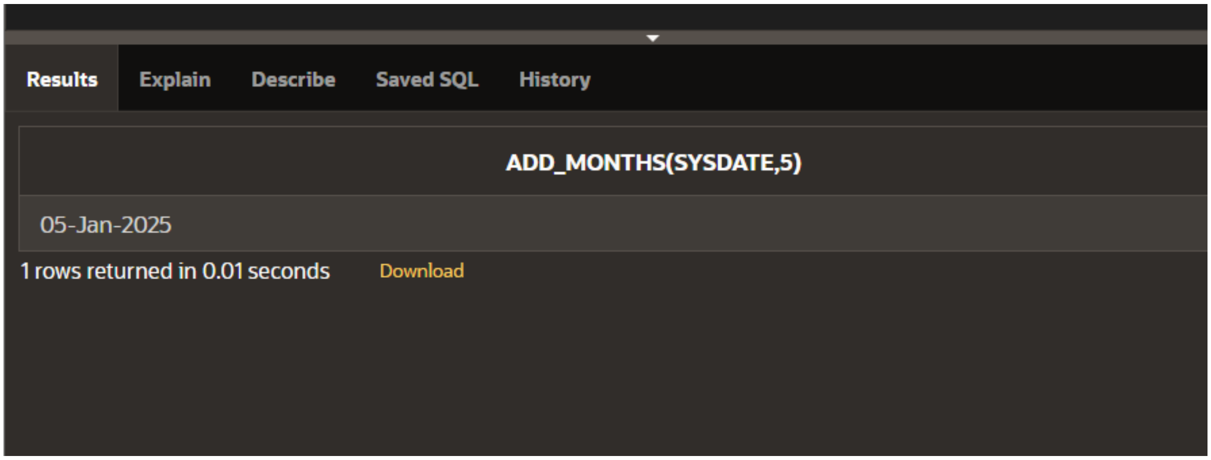
NEXT_DAY(SYSDATE,'MONDAY')
12-Aug-2024

1 rows returned in 0.00 seconds

ADD MONTHS

select add_months(sysdate,5)

from dual;



ADD_MONTHS(SYSDATE,5)
05-Jan-2025

1 rows returned in 0.01 seconds

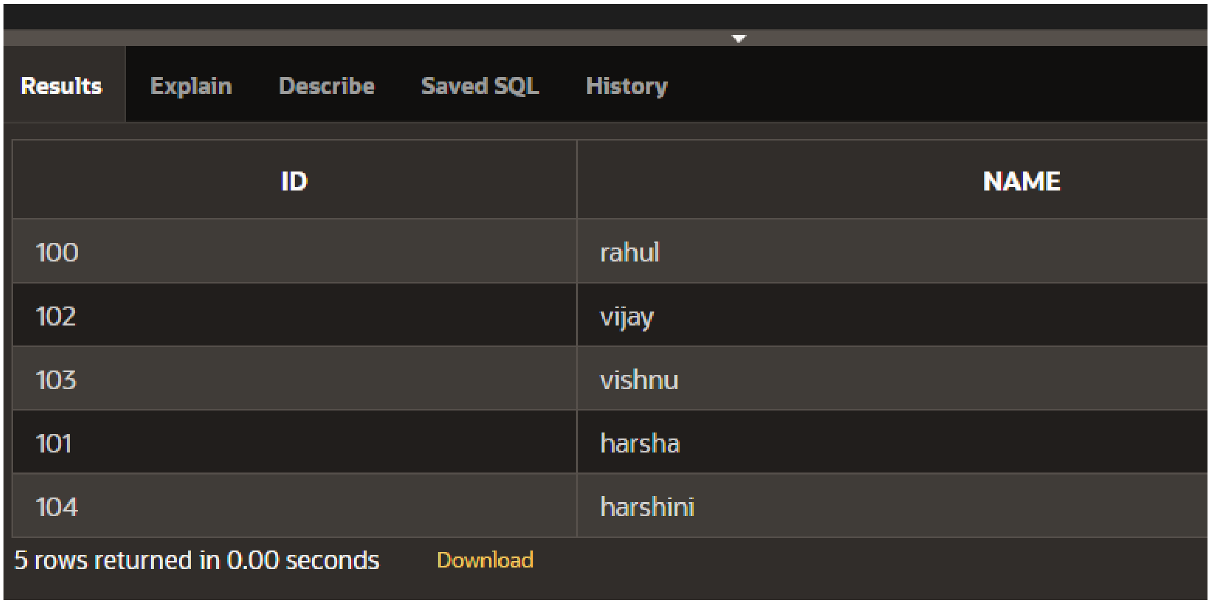
MONTHS BETWEEN

select id,name

from singer

where months_between

(sysdate,hiredate)<100;



ID	NAME
100	rahul
102	vijay
103	vishnu
101	harsha
104	harshini

5 rows returned in 0.00 seconds