

# Advanced Database - TP 2

## Advanced SQL

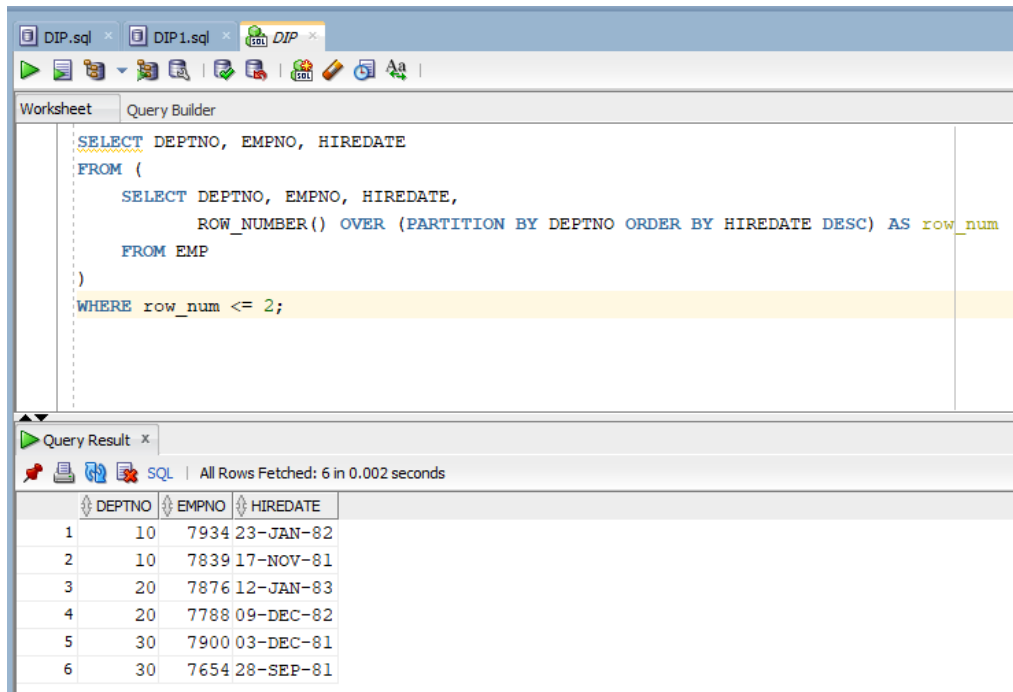
Name: Sayed Mujtaba

Lastname: Ahmady

STDNO: 62707

### Answer 1:

1. Gets the 2 persons per department, who have arrived the latest in the company.



The screenshot shows a SQL query editor with the following query:

```
SELECT DEPTNO, EMPNO, HIREDATE
FROM (
    SELECT DEPTNO, EMPNO, HIREDATE,
           ROW_NUMBER() OVER (PARTITION BY DEPTNO ORDER BY HIREDATE DESC) AS row_num
    FROM EMP
)
WHERE row_num <= 2;
```

Below the query editor, the 'Query Result' pane shows the results of the query. It indicates that all 6 rows were fetched in 0.002 seconds. The results are displayed in a table with columns DEPTNO, EMPNO, and HIREDATE.

	DEPTNO	EMPNO	HIREDATE
1	10	7934	23-JAN-82
2	10	7839	17-NOV-81
3	20	7876	12-JAN-83
4	20	7788	09-DEC-82
5	30	7900	03-DEC-81
6	30	7654	28-SEP-81

### Answer 2:

2. Show your analytical Skill and Invents an interesting query using Windows Functions (i.e.: a SELECT query on EMP table): The query should include the usage of "ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING".



Worksheet Query Builder

```

CREATE TABLE EMP_MEDIUM_TABLE
(EMPNO NUMBER(10),
MANAGER_ID NUMBER(10),
DEPTID VARCHAR2(10),
GENDER VARCHAR2(2) not null,
NAME VARCHAR2(1000));

INSERT INTO EMP_MEDIUM_TABLE
SELECT      LEVEL                                empl_id,
           TRUNC (DBMS_RANDOM.VALUE (1, 100), 0) manager_id,

```

Query Result x Script Output x

Task completed in 5.676 seconds

Table PROJECT\_MEDIUM\_TABLE created.

50,000 rows inserted.

Table PROJECT\_EMP\_MEDIUM\_TABLE created.

50,000 rows inserted.

## Answer 4:

The goal of this question is to check the response time of query in the time that we ran the query we can see that the response time is 0.023.

Worksheet Query Builder

```

SELECT gender, count(*) from EMP_MEDIUM_TABLE where MANAGER_ID = 7 group by gender;

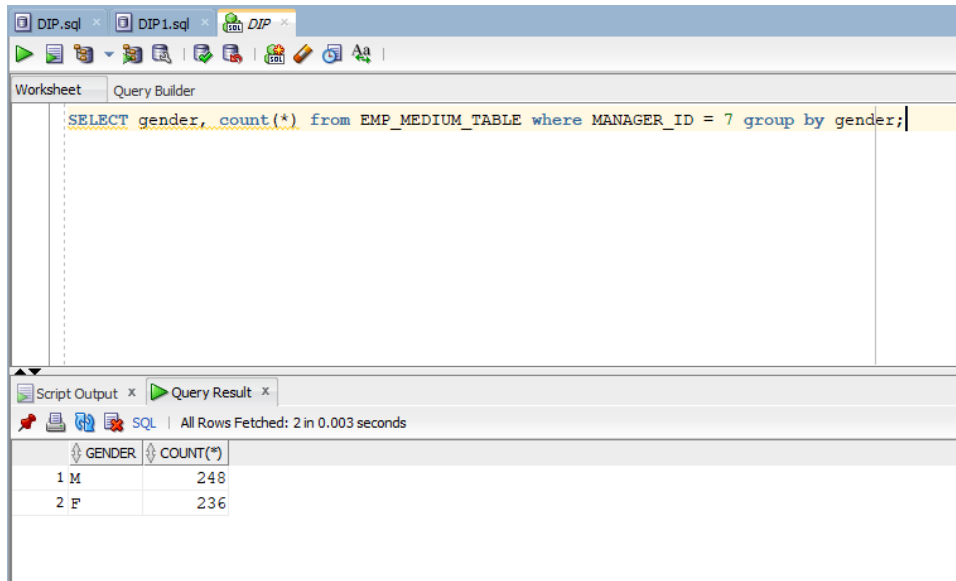
```

Script Output x Query Result x

All Rows Fetched: 2 in 0.023 seconds

GENDER	COUNT(*)
1 M	248
2 F	236

But second time that we ran the query we can see that the response time is much faster than the previous time because the data saved in cache of the system and the response time is much faster.

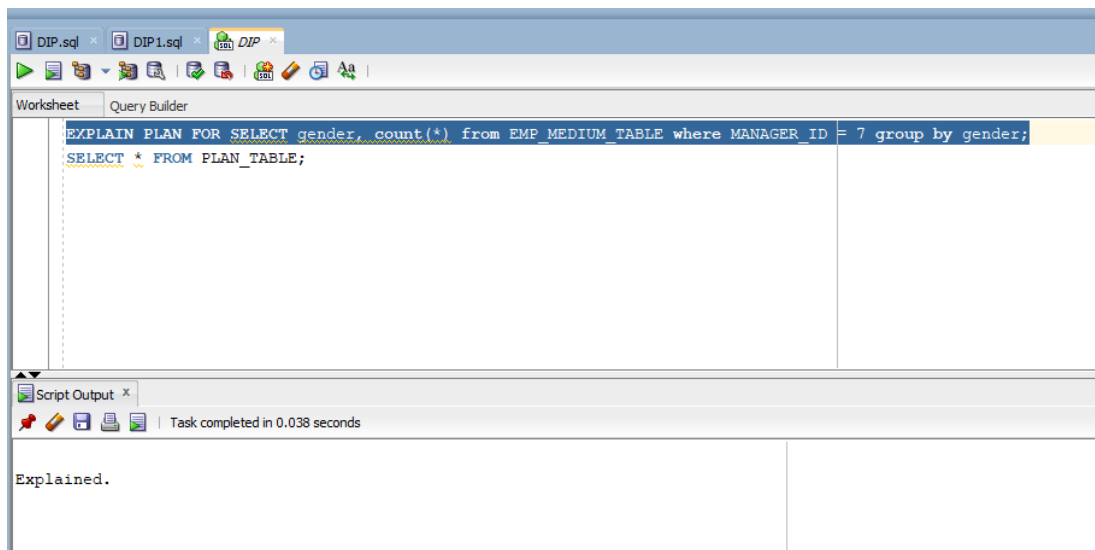


The screenshot shows the SQL Developer interface. The top pane displays the query: `SELECT gender, count(*) from EMP_MEDIUM_TABLE where MANAGER_ID = 7 group by gender;`. The bottom pane shows the 'Query Result' tab with the following data:

GENDER	COUNT(*)
1 M	248
2 F	236

Below the table, it indicates 'All Rows Fetched: 2 in 0.003 seconds'.

## Answer 5:



The screenshot shows the SQL Developer interface. The top pane displays the query: `EXPLAIN PLAN FOR SELECT gender, count(*) from EMP_MEDIUM_TABLE where MANAGER_ID = 7 group by gender;` followed by `SELECT * FROM PLAN_TABLE;`. The bottom pane shows the 'Script Output' tab with the text: 'Explained.'

The explained query

The screenshot shows a SQL Developer window with a query in the Worksheet tab. The query is: `EXPLAIN PLAN FOR SELECT gender, count(*) from EMP_MEDIUM_TABLE where MANAGER_ID = 7 group by gender; SELECT * FROM PLAN_TABLE;`. The Query Result tab displays the execution plan for this query.

STATEMENT_ID	PLAN_ID	TIMESTAMP	REMARKS	OPERATION	OPTIONS	OBJECT_NODE	OBJECT_OWNER	OBJECT_NAME	OBJECT_ALIAS
1 (null)		114-MAR-23 (null)	SELECT STATEMENT	(null)	(null)	(null)	(null)	(null)	(null)
2 (null)		114-MAR-23 (null)	HASH	GROUP BY	(null)	(null)	(null)	(null)	(null)
3 (null)		114-MAR-23 (null)	TABLE ACCESS	FULL	(null)	DIP	EMP_MEDIUM_TABLE	"EMP_MEDIUM_TABLE"@SEL\$	
4 (null)		214-MAR-23 (null)	SELECT STATEMENT	(null)	(null)	(null)	(null)	(null)	(null)
5 (null)		214-MAR-23 (null)	HASH	GROUP BY	(null)	(null)	(null)	(null)	(null)
6 (null)		214-MAR-23 (null)	TABLE ACCESS	FULL	(null)	DIP	EMP_MEDIUM_TABLE	"EMP_MEDIUM_TABLE"@SEL\$	
7 (null)		314-MAR-23 (null)	SELECT STATEMENT	(null)	(null)	(null)	(null)	(null)	(null)
8 (null)		314-MAR-23 (null)	HASH	GROUP BY	(null)	(null)	(null)	(null)	(null)
9 (null)		314-MAR-23 (null)	TABLE ACCESS	FULL	(null)	DIP	EMP_MEDIUM_TABLE	"EMP_MEDIUM_TABLE"@SEL\$	

### 3.2

Using the following query we created an index for the columns of Manager\_id and Gender on the table EMP\_MEDIUM\_TABLE.

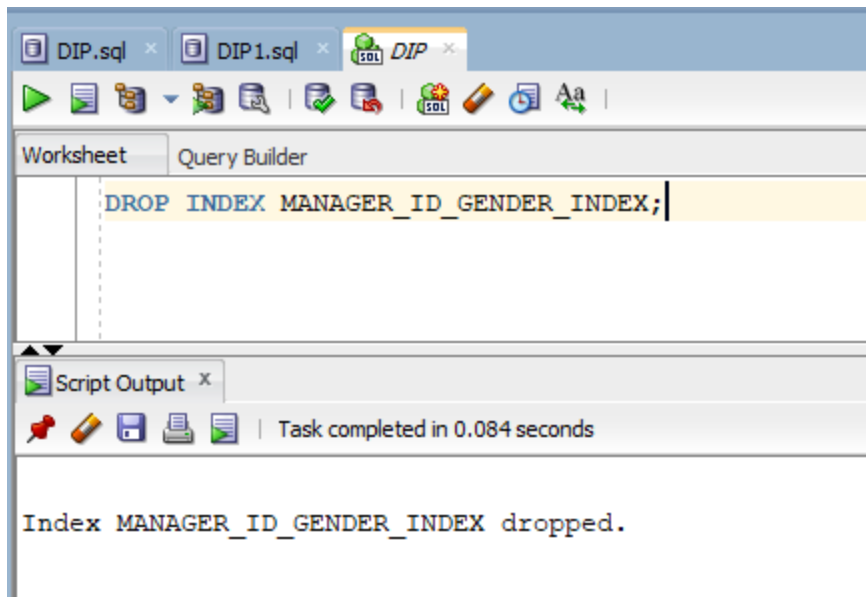
The screenshot shows a SQL Developer window with a query in the Worksheet tab. The query is: `CREATE INDEX MANAGER_ID_GENDER_INDEX ON EMP_MEDIUM_TABLE (MANAGER_ID, GENDER);`. The Query Result tab displays the execution result of this query.

STATEMENT_ID	PLAN_ID	TIMESTAMP	REMARKS	OPERATION	OPTIONS	OBJECT_NODE	OBJECT_OWNER	OBJECT_NAME	OBJECT_ALIAS
1 (null)		114-MAR-23 (null)	SELECT STATEMENT	(null)	(null)	(null)	(null)	(null)	(null)
2 (null)		114-MAR-23 (null)	HASH	GROUP BY	(null)	(null)	(null)	(null)	(null)
3 (null)		114-MAR-23 (null)	TABLE ACCESS	FULL	(null)	DIP	EMP_MEDIUM_TABLE	"EMP_MEDIUM_TABLE"@SEL\$	
4 (null)		214-MAR-23 (null)	SELECT STATEMENT	(null)	(null)	(null)	(null)	(null)	(null)
5 (null)		214-MAR-23 (null)	HASH	GROUP BY	(null)	(null)	(null)	(null)	(null)
6 (null)		214-MAR-23 (null)	TABLE ACCESS	FULL	(null)	DIP	EMP_MEDIUM_TABLE	"EMP_MEDIUM_TABLE"@SEL\$	
7 (null)		314-MAR-23 (null)	SELECT STATEMENT	(null)	(null)	(null)	(null)	(null)	(null)
8 (null)		314-MAR-23 (null)	HASH	GROUP BY	(null)	(null)	(null)	(null)	(null)
9 (null)		314-MAR-23 (null)	TABLE ACCESS	FULL	(null)	DIP	EMP_MEDIUM_TABLE	"EMP_MEDIUM_TABLE"@SEL\$	

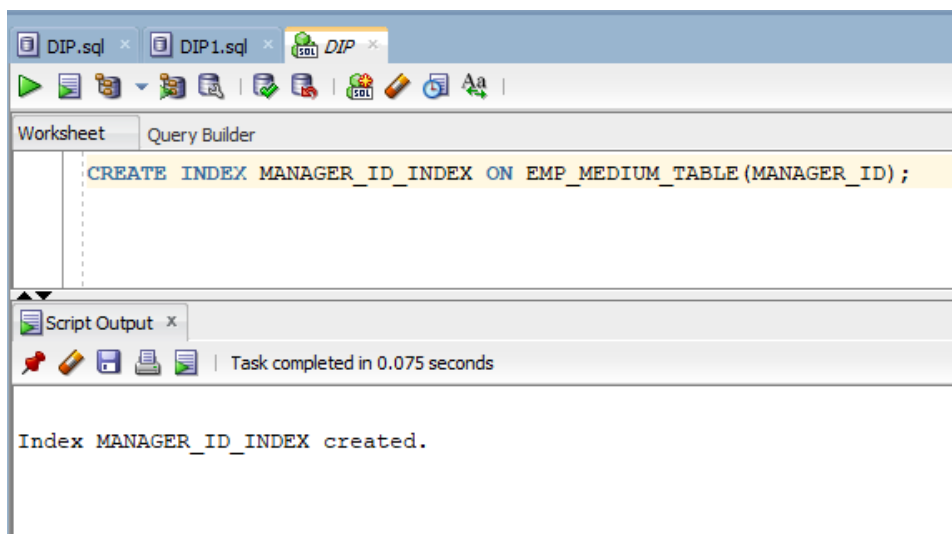
Explained.

Index MANAGER\_ID\_GENDER\_INDEX created.

Using the following query we delete the created index.



Using following query we created the index again.



If you use SqlPlus as a client (the display is easier with Sqlplus for Autrace). Run this:

```
C:\WINDOWS\system32\CMD.exe - sqlplus DIP/DIP
Microsoft Windows [Version 10.0.19045.2486]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hacker>sqlplus DIP/DIP

SQL*Plus: Release 21.0.0.0.0 - Production on Tue Mar 14 13:49:54 2023
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle. All rights reserved.

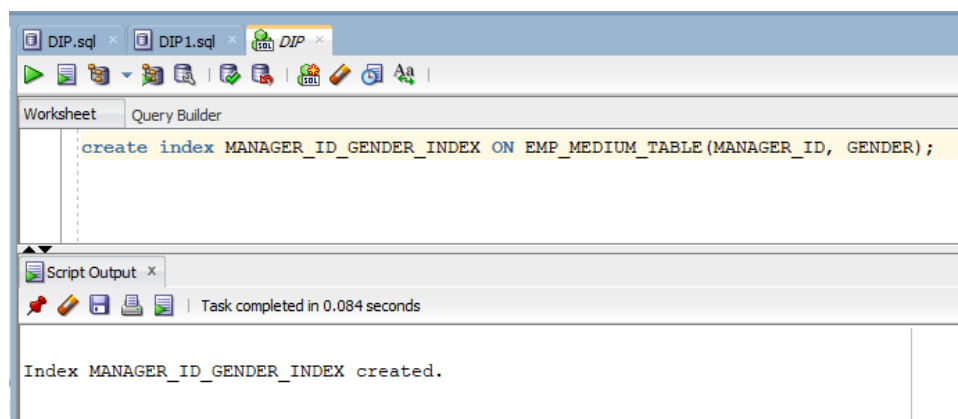
Last Successful login time: Tue Mar 14 2023 11:50:06 +01:00

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> set autotrace traceonly timing on;
SQL> _
```

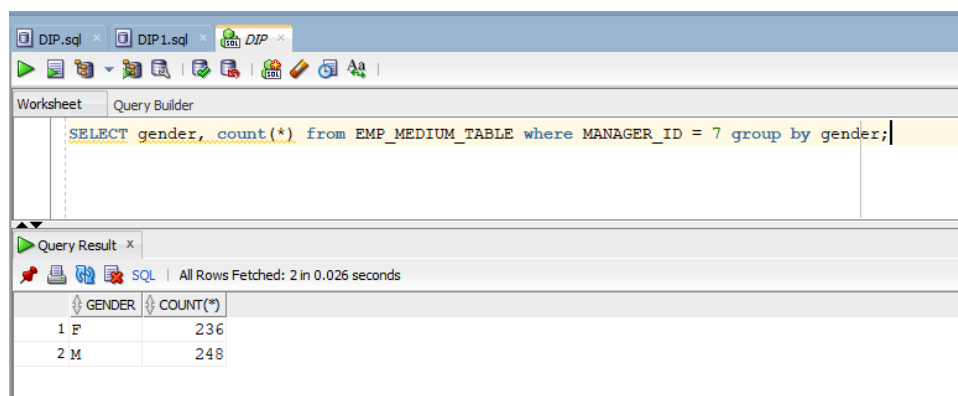
4.

Add a **covering index** on both columns fetched:



The screenshot shows the SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL statement: `create index MANAGER_ID_GENDER_INDEX ON EMP_MEDIUM_TABLE(MANAGER_ID, GENDER);`. The 'Script Output' tab below shows the execution result: 'Index MANAGER\_ID\_GENDER\_INDEX created.' and 'Task completed in 0.084 seconds'.

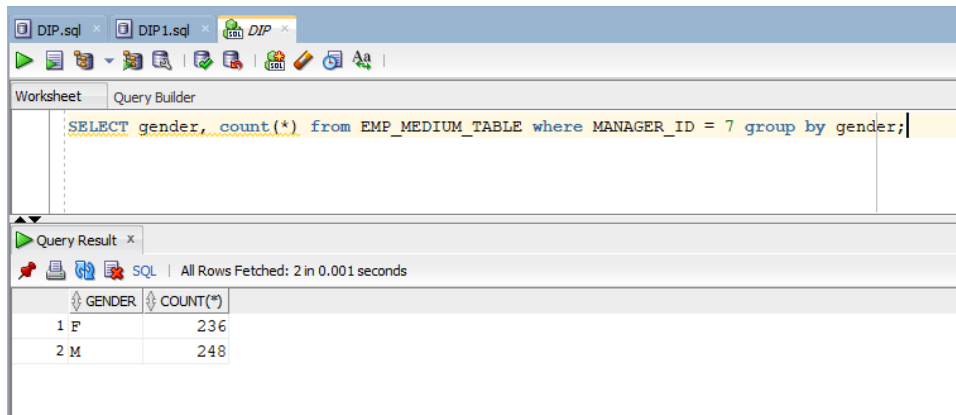
5.



The screenshot shows the SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL statement: `SELECT gender, count(*) from EMP_MEDIUM_TABLE where MANAGER_ID = 7 group by gender;`. The 'Query Result' tab below shows the execution result: 'All Rows Fetched: 2 in 0.026 seconds'. The result is displayed in a table with two columns: GENDER and COUNT(\*).

GENDER	COUNT(*)
1 F	236
2 M	248

RESPONSE TIME IS 0.026



The screenshot shows the SQL Developer interface. The top pane displays a query: `SELECT gender, count(*) from EMP_MEDIUM_TABLE where MANAGER_ID = 7 group by gender;`. The bottom pane shows the query result with two columns: GENDER and COUNT(\*). The results are as follows:

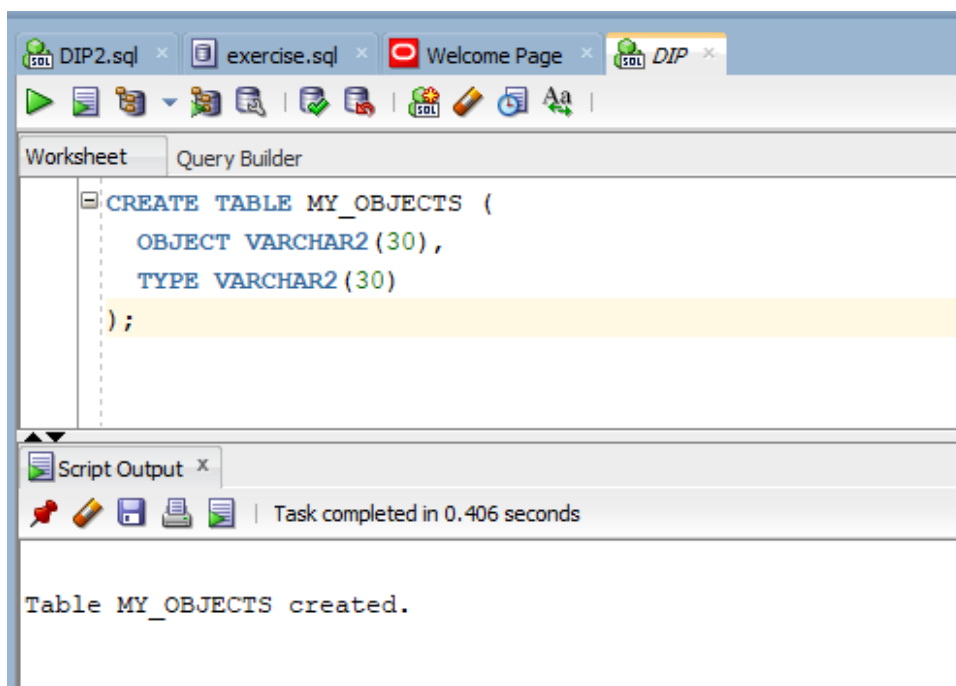
GENDER	COUNT(*)
1 F	236
2 M	248

SECOND TIME WHEN I RUN THE CODE THE RESPONSE TIME IS 0.001 IT MEANS MUCH MORE LESS THAN THE FIRST RESPONSE TIME BECAUSE OUR QUERY WAS SAVED IN THE CACHE AND IT WAS WHY IN SECOND TIME THE RESPONSE TIME IS MUCH MORE FASTER.

### Exercise 3. Data Dictionary

This script creates a table with two columns, OBJECT and TYPE, and then uses the INSERT INTO statement to populate the table with data from the USER\_OBJECTS view. The OBJECT\_NAME and OBJECT\_TYPE columns from USER\_OBJECTS are mapped to the OBJECT and TYPE columns in MY\_OBJECTS, respectively.

You can modify this script to include only the views that you are interested in, and to filter the results based on your criteria. For example, to show only tables and their associated constraints, you can modify the script as follows:

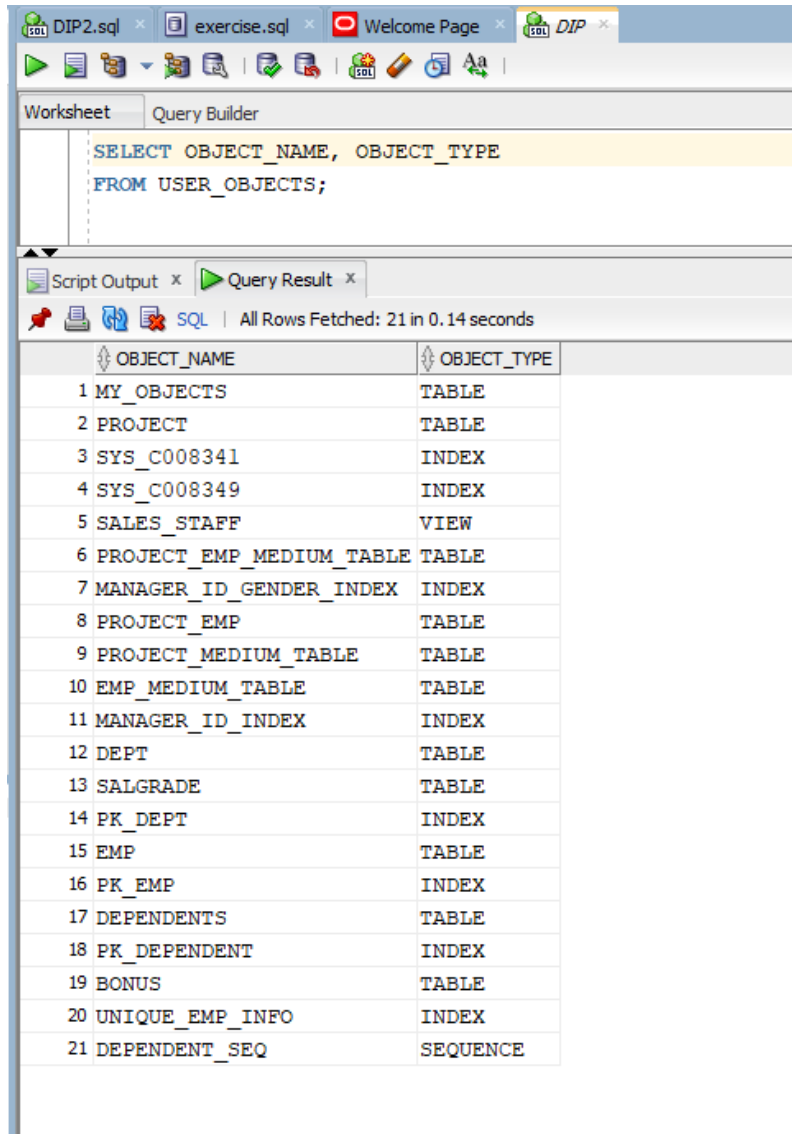


The screenshot shows the SQL Developer interface with a script editor. The script contains the following SQL code:

```
CREATE TABLE MY_OBJECTS (  
  OBJECT VARCHAR2 (30) ,  
  TYPE VARCHAR2 (30)  
);
```

The bottom pane shows the script output: "Table MY\_OBJECTS created." The status bar indicates "Task completed in 0.406 seconds".

This modified script creates a table with two columns, OBJECT and TYPE, and then uses the INSERT INTO statement to populate the table with data from two views: USER\_TABLES (for tables) and USER\_CONSTRAINTS (for constraints). The SELECT statement for each view selects the relevant columns and adds a string literal for the type column. The UNION ALL operator combines the results from both queries into a single result set.



The screenshot shows an SQL IDE interface. The top toolbar includes icons for running queries, saving, and other database functions. The 'Query Builder' tab is active, displaying the following SQL query:

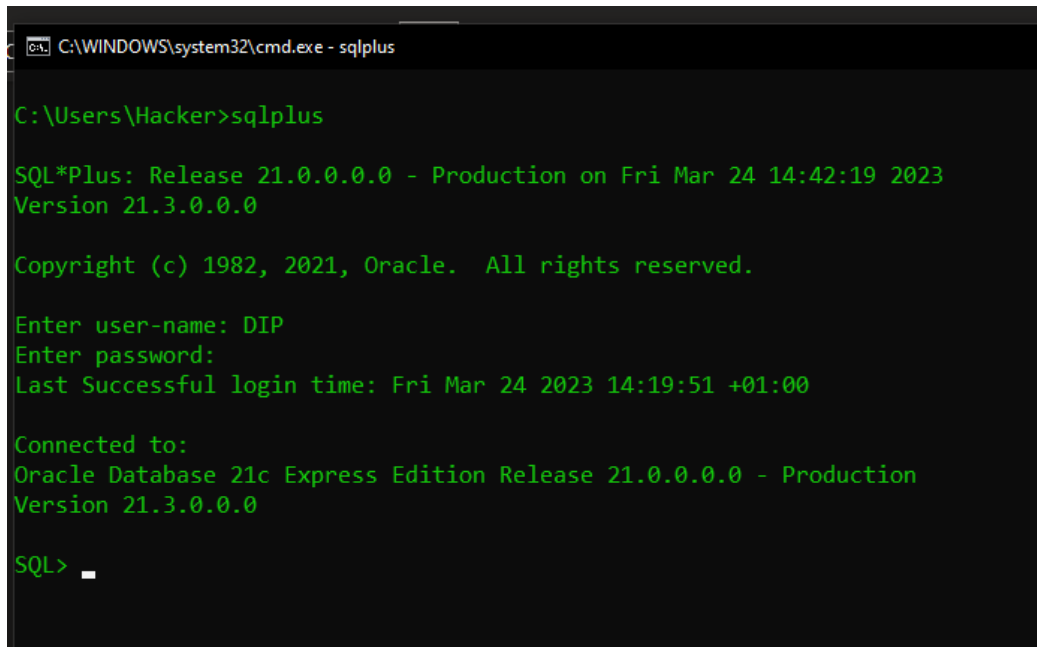
```
SELECT OBJECT_NAME, OBJECT_TYPE  
FROM USER_OBJECTS;
```

Below the query editor, the 'Query Result' tab is active, showing the results of the query. The status bar indicates 'All Rows Fetched: 21 in 0.14 seconds'. The results are displayed in a table with two columns: OBJECT\_NAME and OBJECT\_TYPE.

	OBJECT_NAME	OBJECT_TYPE
1	MY_OBJECTS	TABLE
2	PROJECT	TABLE
3	SYS_C008341	INDEX
4	SYS_C008349	INDEX
5	SALES_STAFF	VIEW
6	PROJECT_EMP_MEDIUM_TABLE	TABLE
7	MANAGER_ID_GENDER_INDEX	INDEX
8	PROJECT_EMP	TABLE
9	PROJECT_MEDIUM_TABLE	TABLE
10	EMP_MEDIUM_TABLE	TABLE
11	MANAGER_ID_INDEX	INDEX
12	DEPT	TABLE
13	SALGRADE	TABLE
14	PK_DEPT	INDEX
15	EMP	TABLE
16	PK_EMP	INDEX
17	DEPENDENTS	TABLE
18	PK_DEPENDENT	INDEX
19	BONUS	TABLE
20	UNIQUE_EMP_INFO	INDEX
21	DEPENDENT_SEQ	SEQUENCE

## Exercise 4. Use SqlPlus

I use windows system and instead of Docker I use Oracle Express So Based on that when I installed Oracle Express by default it has the SQL plus and using the following syntax I can login to my existing users,



```
C:\WINDOWS\system32\cmd.exe - sqlplus

C:\Users\Hacker>sqlplus

SQL*Plus: Release 21.0.0.0.0 - Production on Fri Mar 24 14:42:19 2023
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle. All rights reserved.

Enter user-name: DIP
Enter password:
Last Successful login time: Fri Mar 24 2023 14:19:51 +01:00

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> _
```

## 2

Listing All employees using the sql plus it has the same syntax as Sql Developer no different just sql plus is CLI interface and the sql developer is GUI interface.

```
SQL> SELECT * FROM EMP;
```

EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL
COMM	TEL	DEPTNO	MOBILE_NUM			
7369	SMITH 0149545243	JOHN	CLERK 20 0645818841	7902	17-DEC-80	800
7499	ALLEN 300 0149547243	BOB	SALESMAN 30 0645818851	7698	20-FEB-81	1600
7521	WARD 500 0149545247	PETER	SALESMAN 30 0645814841	7698	22-FEB-81	1250
7566	JONES 0149545456	JOHN	MANAGER 20 0635818841	7839	02-APR-81	2975
7654	MARTIN 1400 0149545784	JOE	SALESMAN 30 0645818811	7698	28-SEP-81	1250
7698	BLAKE 0149545254	BOB	MANAGER 30 0645812241	7839	01-MAY-81	2850
7782	CLARK 0149545245	JOHN	MANAGER 10 0645877841	7839	09-JUN-81	2450
7788	SCOTT 0149545249	GUY	ANALYST 20 0645988841	7566	09-DEC-82	3000
7839	KING 0149545241	GUY	PRESIDENT 10 0645528841	3563	17-NOV-81	5000
7844	TURNER 0 0149548243	PETER	SALESMAN 30 0619818841	7698	08-SEP-81	1500
7876	ADAMS 0149565243	JOSEPH	CLERK 20 0645810241	7788	12-JAN-83	1100
7900	JAMES 0149545564	ALAN	CLERK 30 0645818800	7698	03-DEC-81	950
7902	FORD 0149785243	MARIA	ANALYST 20 0645658841	7566	03-DEC-81	3000
7934	MILLER 0199545243	ALICE	CLERK 10 0645818541	7782	23-JAN-82	1300

14 rows selected.

## Exercise 5. Transaction Part 1 – Beginner

### First Client

```
C:\WINDOWS\system32\cmd.exe - sqlplus

C:\Users\Hacker>sqlplus

SQL*Plus: Release 21.0.0.0.0 - Production on Fri Mar 24 14:50:41 2023
Version 21.3.0.0.0

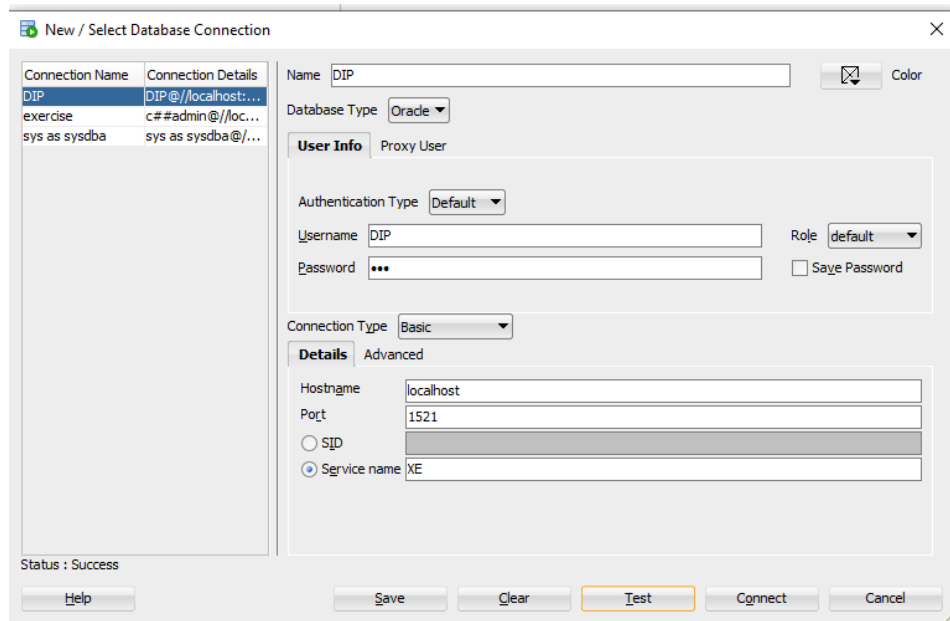
Copyright (c) 1982, 2021, Oracle. All rights reserved.

Enter user-name: DIP
Enter password:
Last Successful login time: Fri Mar 24 2023 14:42:23 +01:00

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> s_
```

Second client



Answer 1

```
SQL>
SQL> set autocommit off;
SQL> _
```

2

```
SQL> UPDATE EMP SET SAL = 5000 WHERE EMPNO = 7369;

1 row updated.

SQL> _
```

3

```
C:\WINDOWS\system32\cmd.exe - sqlplus
SQL> SELECT * FROM EMP;
```

EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL
7369	SMITH	JOHN	CLERK	7902	17-DEC-80	5000
7499	ALLEN	BOB	SALESMAN	7698	20-FEB-81	1600
7521	WARD	PETER	SALESMAN	7698	22-FEB-81	1250
7566	JONES	JOHN	MANAGER	7839	02-APR-81	2975
7654	MARTIN	JOE	SALESMAN	7698	28-SEP-81	1250
7698	BLAKE	BOB	MANAGER	7839	01-MAY-81	2850
7782	CLARK	JOHN	MANAGER	7839	09-JUN-81	2450
7788	SCOTT	GUY	ANALYST	7566	09-DEC-82	3000
7839	KING	GUY	PRESIDENT	3563	17-NOV-81	5000
7844	TURNER	PETER	SALESMAN	7698	08-SEP-81	1500
7876	ADAMS	JOSEPH	CLERK	7788	12-JAN-83	1100
7900	JAMES	ALAN	CLERK	7698	03-DEC-81	950
7902	FORD	MARIA	ANALYST	7566	03-DEC-81	3000
7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300

```
14 rows selected.
```

YES

4

The screenshot shows a SQL client window with a 'Query Result' tab. The query executed is 'SELECT \* FROM EMP;'. The result is a table with 14 rows and 12 columns: EMPNO, ENAME, EFIRST, JOB, MGR, HIREDATE, SAL, COMM, TEL, DEPTNO, and MOBILE\_NUM. The data represents the 'EMP' table from the Oracle database.

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7369	SMITH	JOHN	CLERK	7902	17-DEC-80	800	(null)	0149545243	20	0645818841
2	7499	ALLEN	BOB	SALESMAN	7698	20-FEB-81	1600	300	0149547243	30	0645818851
3	7521	WARD	PETER	SALESMAN	7698	22-FEB-81	1250	500	0149545247	30	0645814841
4	7566	JONES	JOHN	MANAGER	7839	02-APR-81	2975	(null)	0149545456	20	0635818841
5	7654	MARTIN	JOE	SALESMAN	7698	28-SEP-81	1250	1400	0149545784	30	0645818811
6	7698	BLAKE	BOB	MANAGER	7839	01-MAY-81	2850	(null)	0149545254	30	0645812241
7	7782	CLARK	JOHN	MANAGER	7839	09-JUN-81	2450	(null)	0149545245	10	0645877841
8	7788	SCOTT	GUY	ANALYST	7566	09-DEC-82	3000	(null)	0149545249	20	0645988841
9	7839	KING	GUY	PRESIDENT	3563	17-NOV-81	5000	(null)	0149545241	10	0645528841
10	7844	TURNER	PETER	SALESMAN	7698	08-SEP-81	1500	0	0149548243	30	0619818841
11	7876	ADAMS	JOSEPH	CLERK	7788	12-JAN-83	1100	(null)	0149565243	20	0645810241
12	7900	JAMES	ALAN	CLERK	7698	03-DEC-81	950	(null)	0149545564	30	0645818800
13	7902	FORD	MARIA	ANALYST	7566	03-DEC-81	3000	(null)	0149785243	20	0645658841
14	7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300	(null)	0199545243	10	0645818541

No In the second client we can't see the updated salary because we turned off the auto commit so we need to commit it manually.

```
SQL> commit;

Commit complete.

SQL> _
```

After successful commit we can see the updated salary

Worksheet Query Builder

`SELECT * FROM EMP;`

Query Result x

All Rows Fetched: 14 in 0.007 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7369	SMITH	JOHN	CLERK		7902 17-DEC-80	5000	(null)	0149545243	20	0645818841
2	7499	ALLEN	BOB	SALESMAN		7698 20-FEB-81	1600	300	0149547243	30	0645818851
3	7521	WARD	PETER	SALESMAN		7698 22-FEB-81	1250	500	0149545247	30	0645814841
4	7566	JONES	JOHN	MANAGER		7839 02-APR-81	2975	(null)	0149545456	20	0635818841
5	7654	MARTIN	JOE	SALESMAN		7698 28-SEP-81	1250	1400	0149545784	30	0645818811
6	7698	BLAKE	BOB	MANAGER		7839 01-MAY-81	2850	(null)	0149545254	30	0645812241
7	7782	CLARK	JOHN	MANAGER		7839 09-JUN-81	2450	(null)	0149545245	10	0645877841
8	7788	SCOTT	GUY	ANALYST		7566 09-DEC-82	3000	(null)	0149545249	20	0645988841
9	7839	KING	GUY	PRESIDENT		3563 17-NOV-81	5000	(null)	0149545241	10	0645528841
10	7844	TURNER	PETER	SALESMAN		7698 08-SEP-81	1500	0	0149548243	30	0619818841
11	7876	ADAMS	JOSEPH	CLERK		7788 12-JAN-83	1100	(null)	0149565243	20	0645810241
12	7900	JAMES	ALAN	CLERK		7698 03-DEC-81	950	(null)	0149545564	30	0645818800
13	7902	FORD	MARIA	ANALYST		7566 03-DEC-81	3000	(null)	0149785243	20	0645658841
14	7934	MILLER	ALICE	CLERK		7782 23-JAN-82	1300	(null)	0199545243	10	0645818541

## Exercise 6. Transaction Part 2 && GRANT rights – Advanced

0

Worksheet Query Builder

`CREATE USER user2 IDENTIFIED BY mypass;`

Script Output x

Task completed in 0.216 seconds

User USER2 created.

1

DIP is user one

Worksheet Query Builder

```
SELECT * FROM DIP.EMP;
```

Query Result x

SQL | All Rows Fetched: 14 in 0.01 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7369	SMITH	JOHN	CLERK	7902	17-DEC-80	5000	(null)	0149545243	20	0645818841
2	7499	ALLEN	BOB	SALESMAN	7698	20-FEB-81	1600	300	0149547243	30	0645818851
3	7521	WARD	PETER	SALESMAN	7698	22-FEB-81	1250	500	0149545247	30	0645814841
4	7566	JONES	JOHN	MANAGER	7839	02-APR-81	2975	(null)	0149545456	20	0635818841
5	7654	MARTIN	JOE	SALESMAN	7698	28-SEP-81	1250	1400	0149545784	30	0645818811
6	7698	BLAKE	BOB	MANAGER	7839	01-MAY-81	2850	(null)	0149545254	30	0645812241
7	7782	CLARK	JOHN	MANAGER	7839	09-JUN-81	2450	(null)	0149545245	10	0645877841
8	7788	SCOTT	GUY	ANALYST	7566	09-DEC-82	3000	(null)	0149545249	20	0645988841
9	7839	KING	GUY	PRESIDENT	3563	17-NOV-81	5000	(null)	0149545241	10	0645528841
10	7844	TURNER	PETER	SALESMAN	7698	08-SEP-81	1500	0	0149548243	30	0619818841
11	7876	ADAMS	JOSEPH	CLERK	7788	12-JAN-83	1100	(null)	0149565243	20	0645810241
12	7900	JAMES	ALAN	CLERK	7698	03-DEC-81	950	(null)	0149545564	30	0645818800
13	7902	FORD	MARIA	ANALYST	7566	03-DEC-81	3000	(null)	0149785243	20	0645658841
14	7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300	(null)	0199545243	10	0645818541

3

Worksheet Query Builder

```
GRANT SELECT ON DIP.EMP TO user2;
GRANT SELECT ON DIP.DEPT TO user2;
```

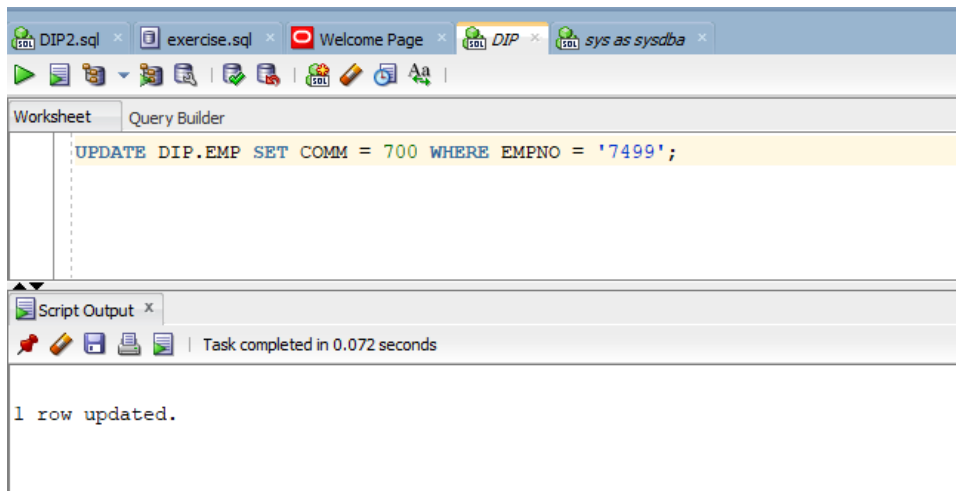
Script Output x

Task completed in 0.106 seconds

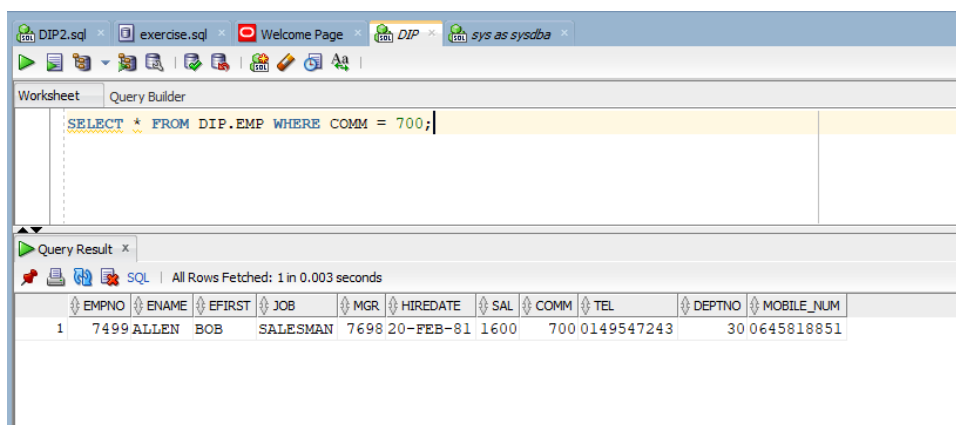
```
Grant succeeded.

Grant succeeded.
```

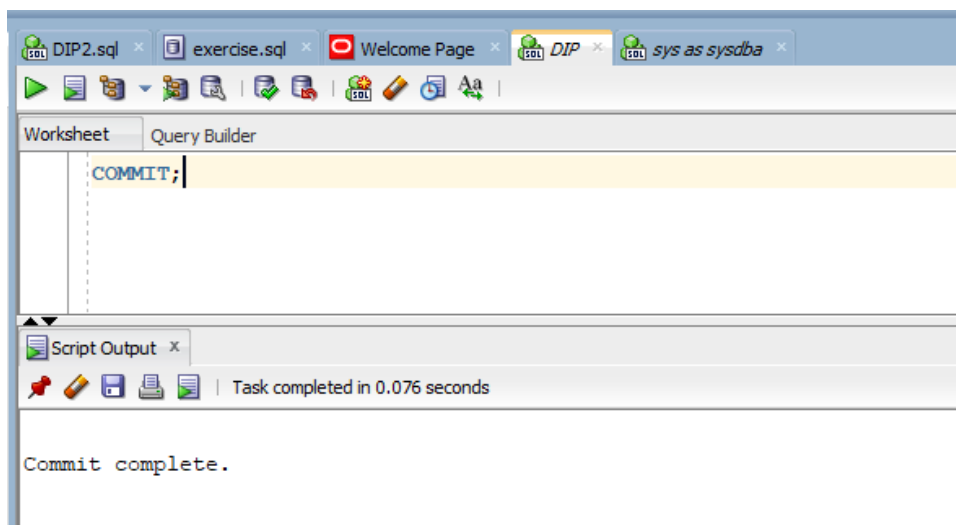
4



5

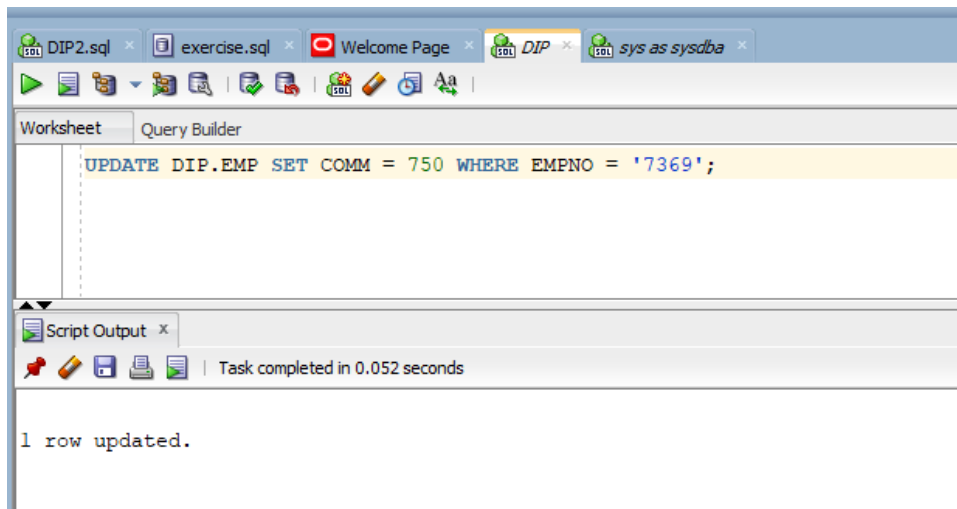


After changing the data we commit the database in order to the U2 be able to see also the changes

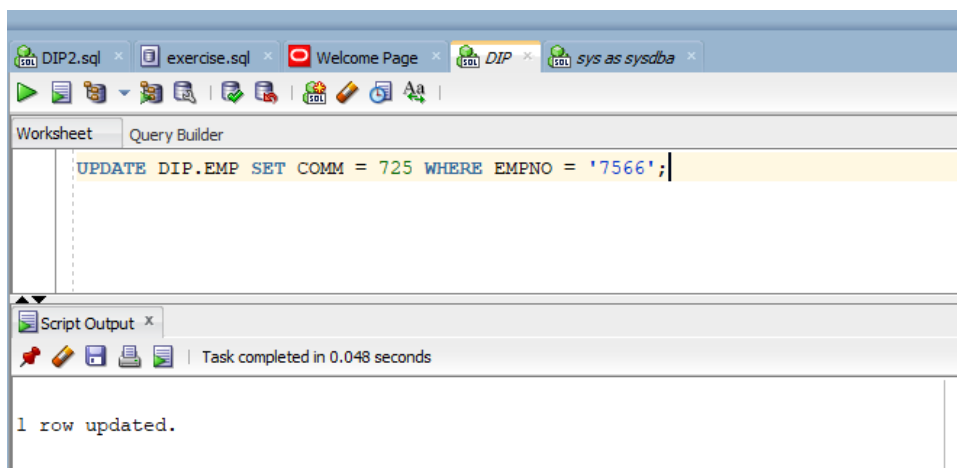


6

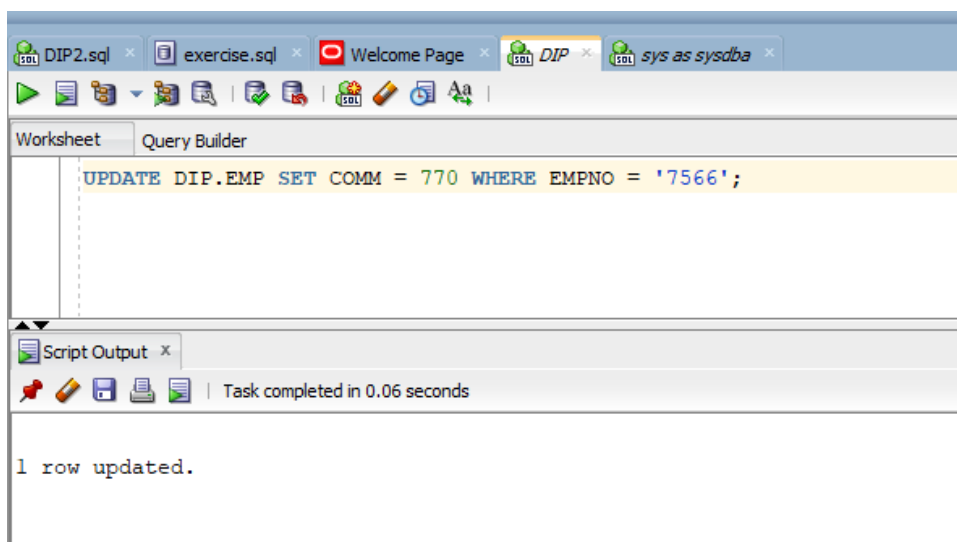
A



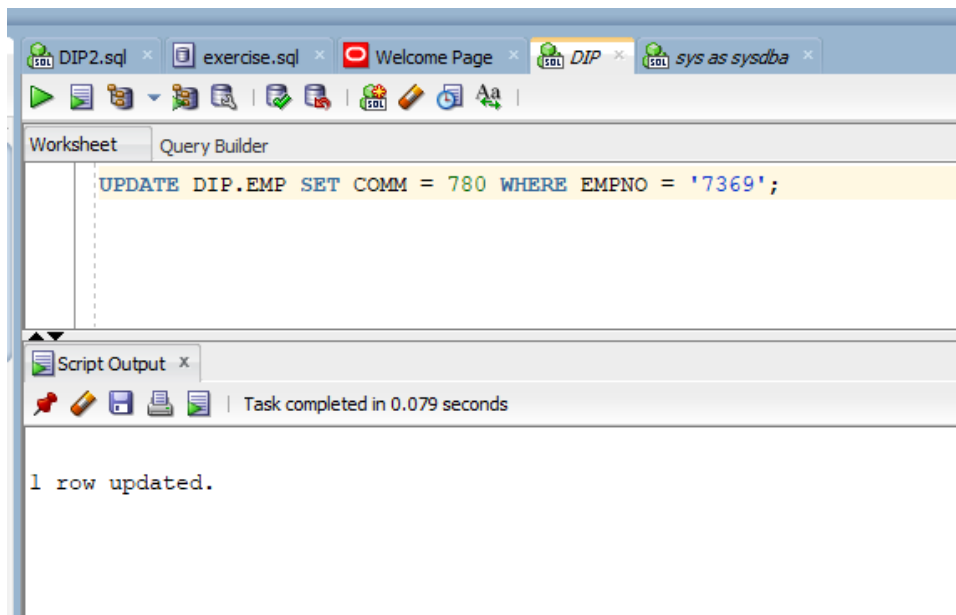
B



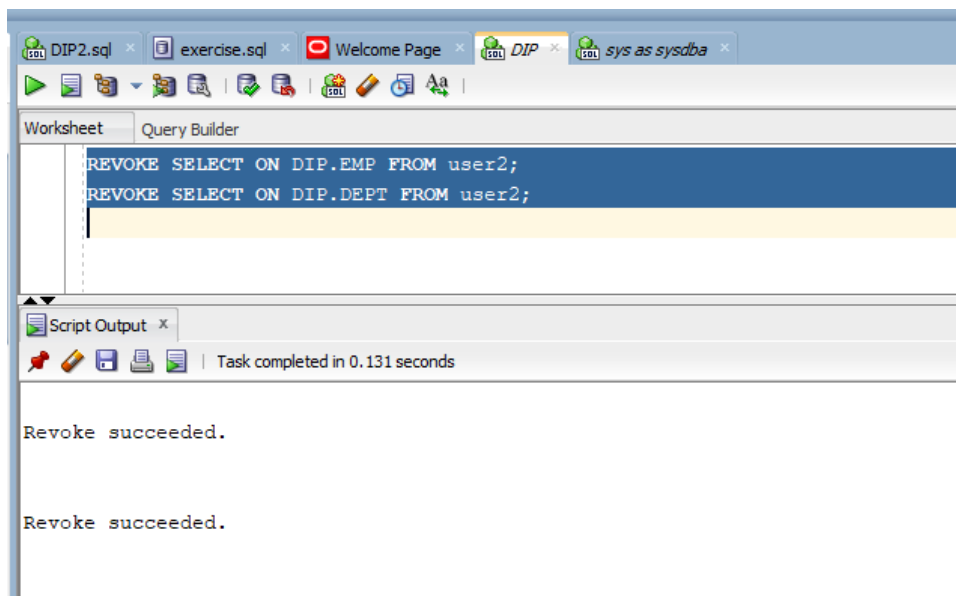
C



D



7



Privilege has been removed

