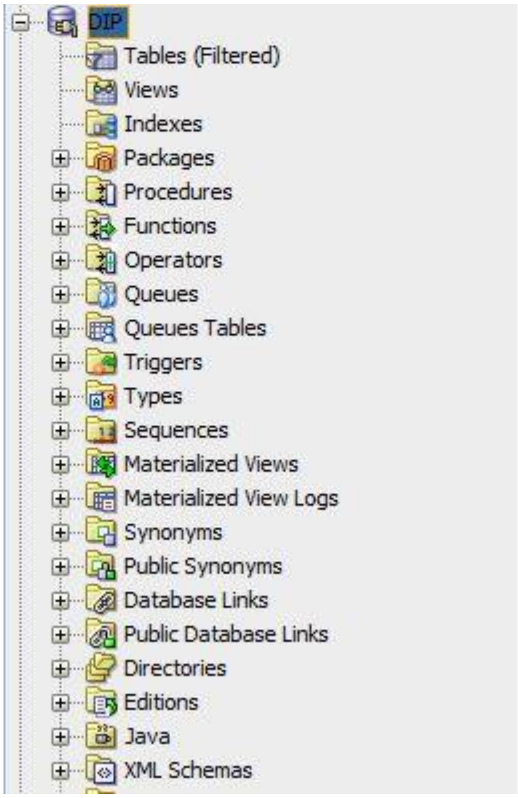
NAME : SAYED MUJTABA LASTNAME : AHMADY STNO : 62707

Section 1

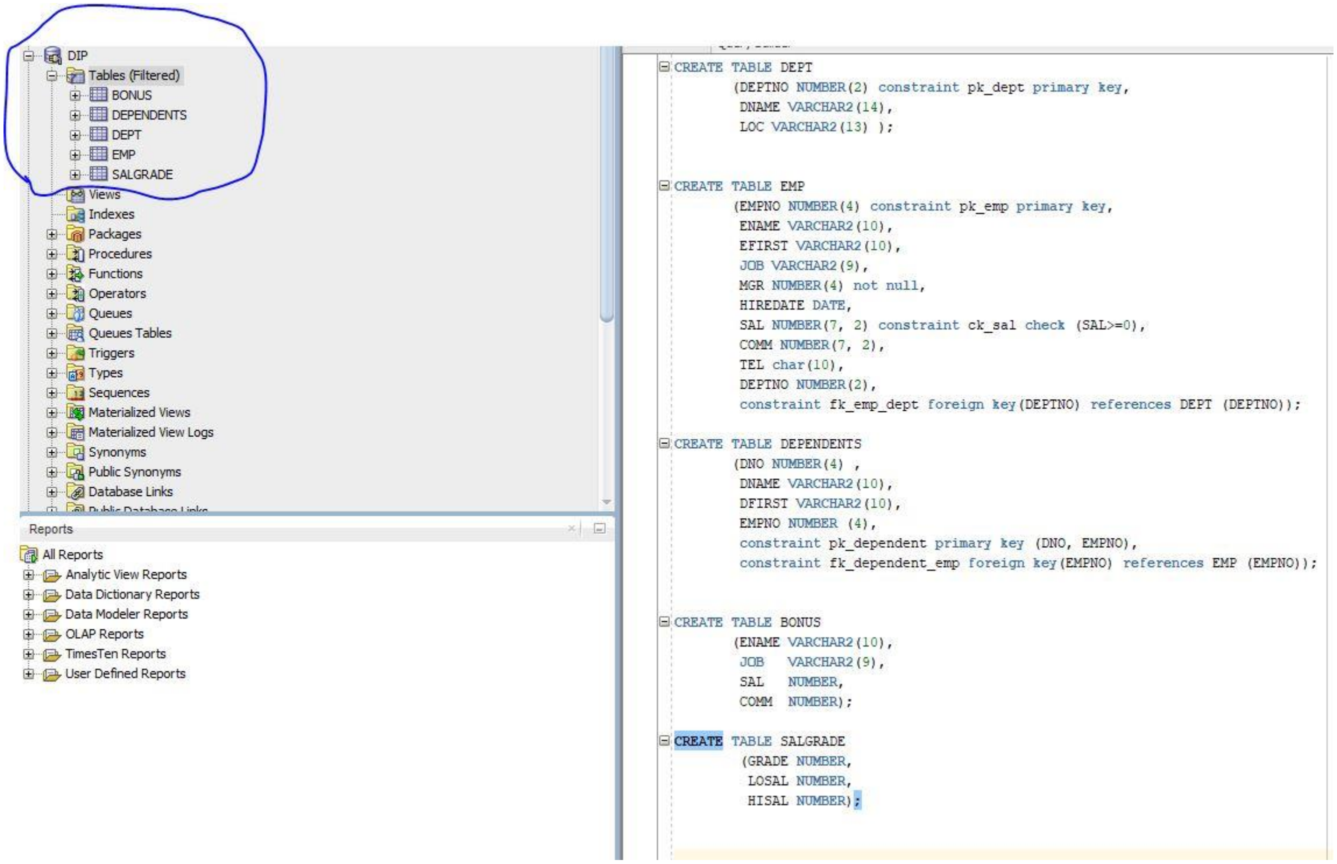
Answer 1:

After Downloading all the necessary Application In the SQL Developer I unlocked another User with empty Data like Table, View, and Indexes to Avoid confusion so right now I have an empty user after downloading the creation.sql I uploaded and I successfully inserted the data and I can see all the tables.

1: We can see that there is no preinserted data in our new user.



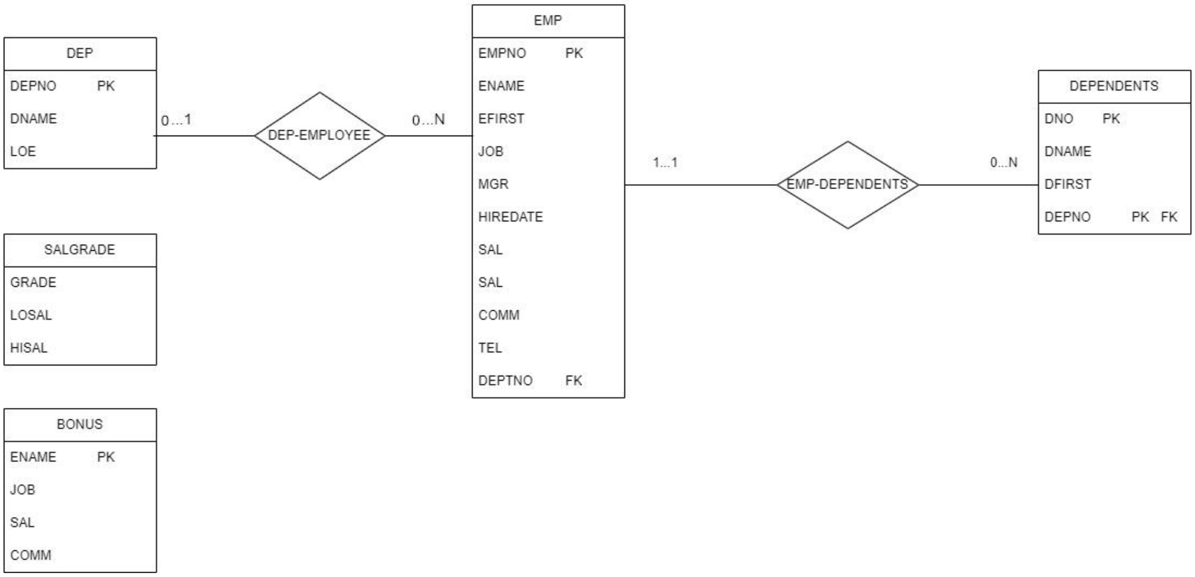
2: After Runing the Creation.sql database successfully we can see the our tables successfully created.



```
CREATE TABLE DEPT
        (DEPTNO NUMBER(2) constraint pk_dept primary key,
         DNAME VARCHAR2(14),
         LOC VARCHAR2(13) );


CREATE TABLE EMP
        (EMPNO NUMBER(4) constraint pk_emp primary key,
         ENAME VARCHAR2(10),
         EFIRST VARCHAR2(10),
         JOB VARCHAR2(9),
         MGR NUMBER(4) not null,
         HIREDATE DATE,
         SAL NUMBER(7, 2) constraint ck_sal check (SAL>=0),
         COMM NUMBER(7, 2),
         TEL char(10),
         DEPTNO NUMBER(2),
         constraint fk_emp_dept foreign key(DEPTNO) references DEPT (DEPTNO));

CREATE TABLE DEPENDENTS
        (DNO NUMBER(4) ,
         DNAME VARCHAR2(10),
         DFIRST VARCHAR2(10),
         EMPNO NUMBER (4),
         constraint pk_dependent primary key (DNO, EMPNO),
         constraint fk_dependent_emp foreign key(EMPNO) references EMP (EMPNO));


CREATE TABLE BONUS
        (ENAME VARCHAR2(10),
         JOB   VARCHAR2(9),
         SAL   NUMBER,
         COMM  NUMBER);

CREATE TABLE SALGRADE
          (GRADE NUMBER,
           LOSAL NUMBER,
           HISAL NUMBER);
```

Answer 2:

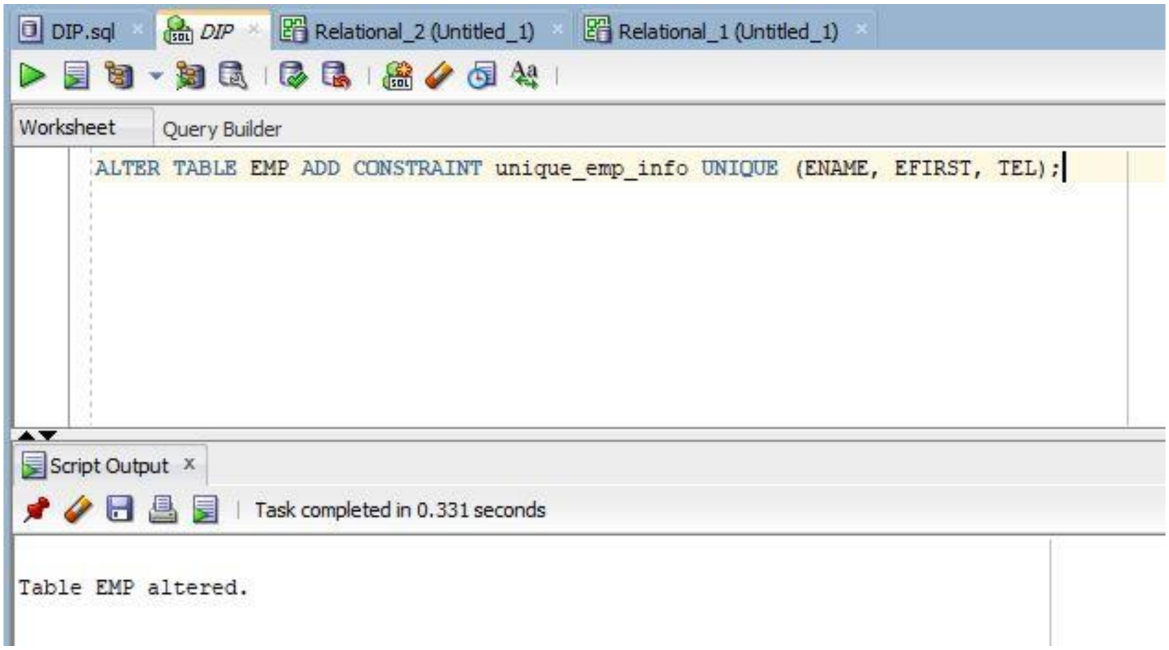
The ER diagram of this database consists of one strong relationship and one weak relationship so the relationships are one-to-many to many and many-to-one.



Answer 3:

Since we want to add some restrictions condition and we want to add an integrity constraint we need to alter our table not create a new table or only write add constraint because it will not work don't forget to add the alter table at beginning of table, we can see in the bellow picture that our table successfully altered.

Don't forget to add the names of columns inside curly brackets otherwise, it will not work properly, or you will face to error.

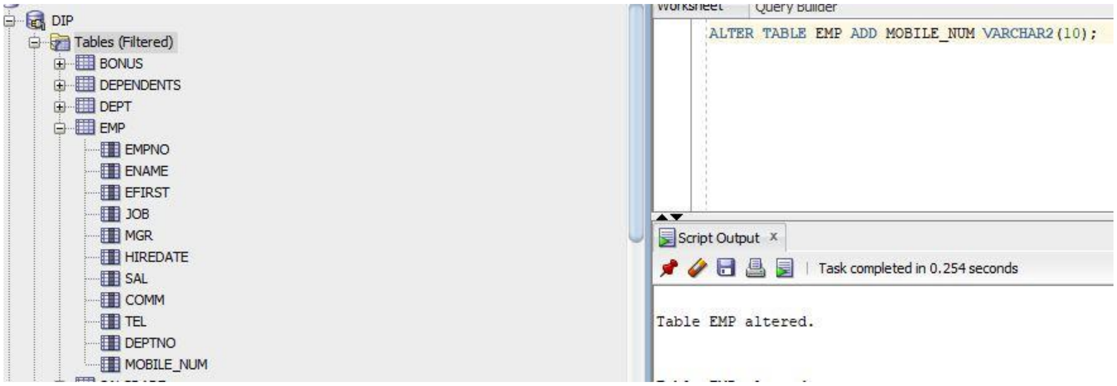

DIP.sql
DIP
Relational_2 (Untitled_1)
Relational_1 (Untitled_1)
Worksheet
Query Builder
Script Output
Task completed in 0.331 seconds

```
ALTER TABLE EMP ADD CONSTRAINT unique_emp_info UNIQUE (ENAME, EFIRST, TEL);
```

```
Table EMP altered.
```

Answer 4:

First of all we need to add a new column because we already have the TEL column so if the user has two type of numbers one telephone number and one mobile phone number we need to add a new column for the mobile phone number and after that, we can add the integrity constraint on our mobile phone number and check of the number start with the 06 or not.

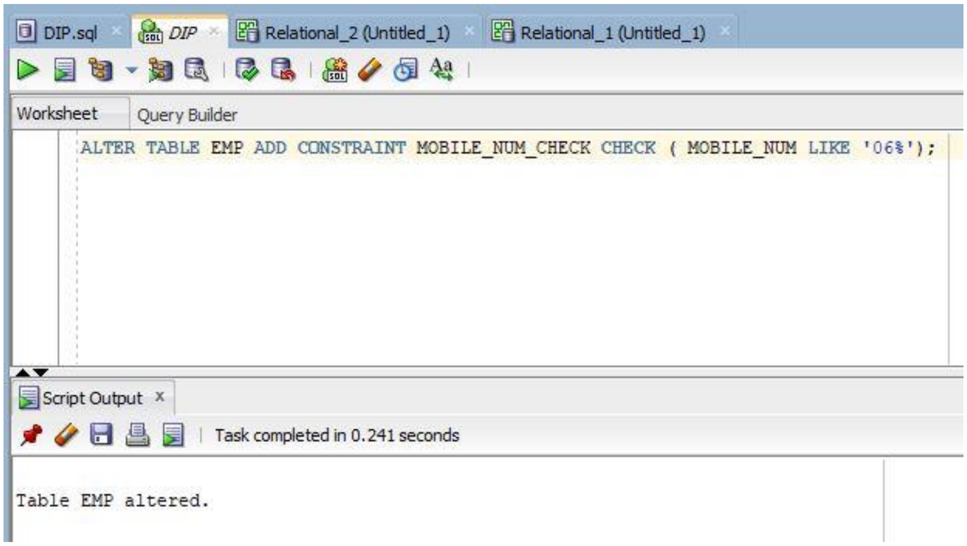


```
ALTER TABLE EMP ADD MOBILE_NUM VARCHAR2(10);
```

Task completed in 0.254 seconds

```
Table EMP altered.
```

So we Add our new column successfully and we can see in the employee table a new column add by the name of Mobile_num now it’s time to add our integrity constraint.



```
ALTER TABLE EMP ADD CONSTRAINT MOBILE_NUM_CHECK CHECK ( MOBILE_NUM LIKE '06%');
```

Task completed in 0.241 seconds

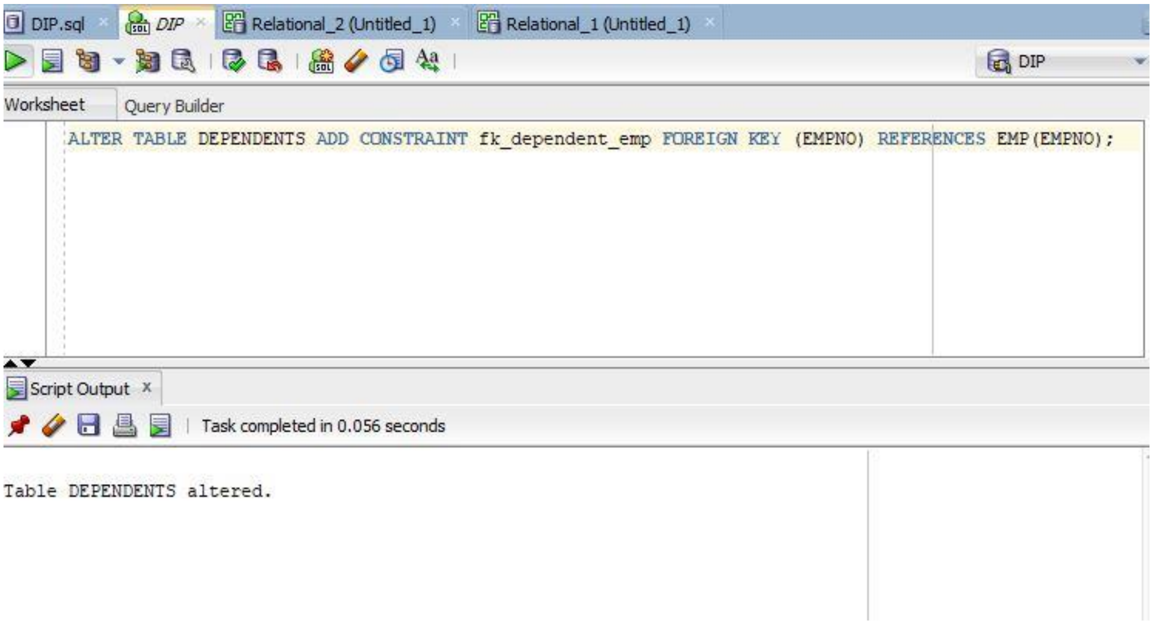
```
Table EMP altered.
```

Answer 5:

First we need to delete the existing foreign key constrain in the table of dependents because it is not possible to alter table to just add a new foreign key because we already have the foreign key constraint

After we alter the table and drop the foreign key constraint fk_ depenent_emp   after that we can again alter the table and add a foreign key constraint with the “On delete cascade” to dynamically apply the dynamically delete all the independents.

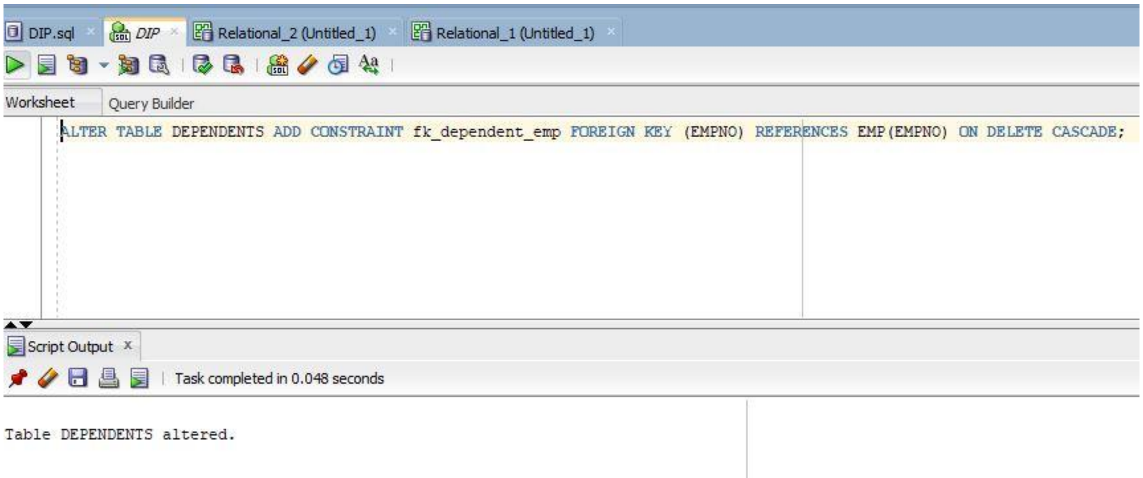
With this query we can delete the key constraint.


DIP.sql
DIP
Relational_2 (Untitled_1)
Relational_1 (Untitled_1)
DIP
Worksheet
Query Builder
Script Output
Task completed in 0.056 seconds

```
ALTER TABLE DEPENDENTS ADD CONSTRAINT fk_dependent_emp FOREIGN KEY (EMPNO) REFERENCES EMP(EMPNO);
```

```
Table DEPENDENTS altered.
```

After we delete the constraint we can add new updates that we want using the following query.


DIP.sql
DIP
Relational_2 (Untitled_1)
Relational_1 (Untitled_1)
Worksheet
Query Builder
Script Output
Task completed in 0.048 seconds

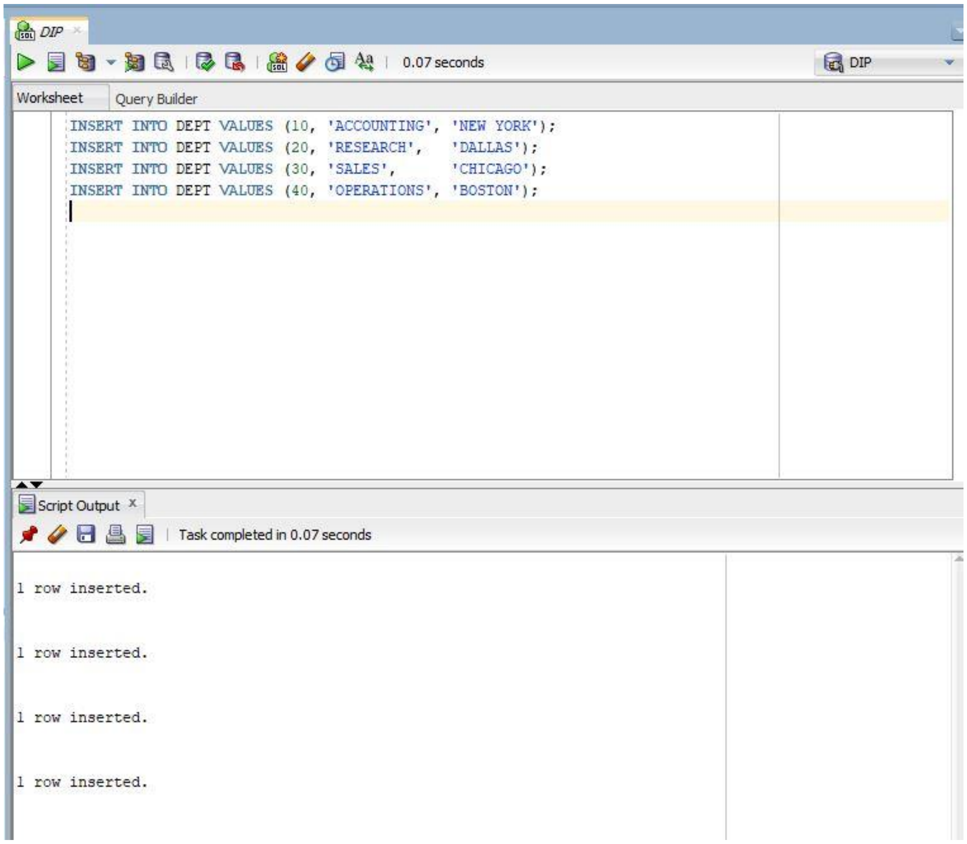
```
ALTER TABLE DEPENDENTS ADD CONSTRAINT fk_dependent_emp FOREIGN KEY (EMPNO) REFERENCES EMP(EMPNO) ON DELETE CASCADE;
```

```
Table DEPENDENTS altered.
```

Answer 6:

After the inserting the data we will face to two different errors.

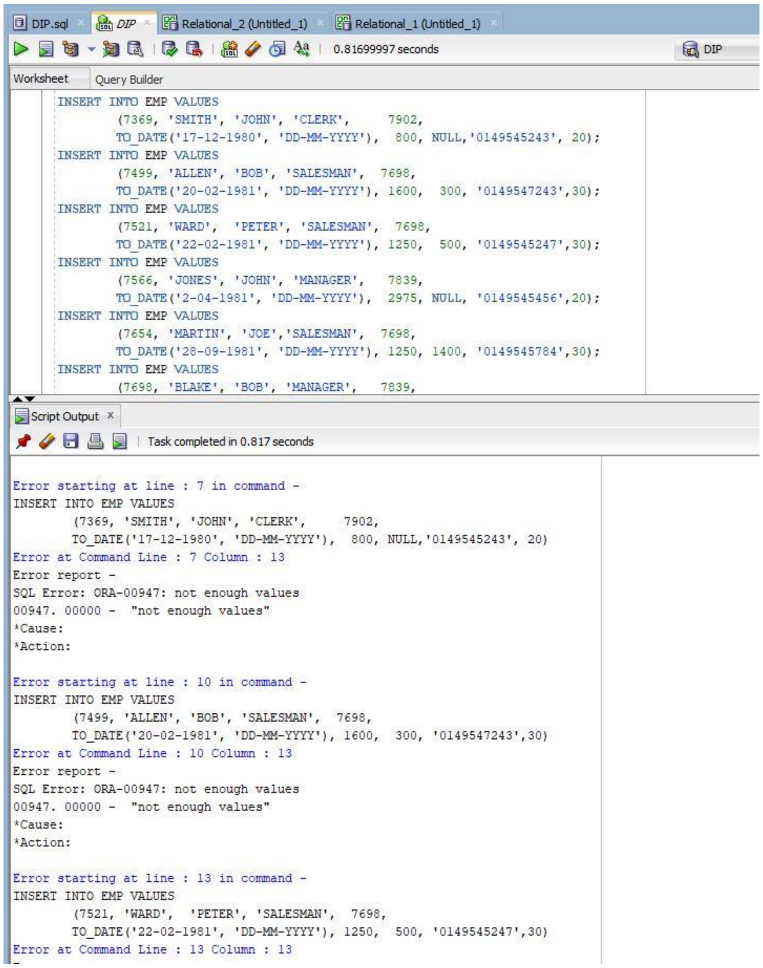
Data for the DEPT table will insert successfully because there is no problem with.



```
INSERT INTO DEPT VALUES (10, 'ACCOUNTING', 'NEW YORK');
INSERT INTO DEPT VALUES (20, 'RESEARCH',   'DALLAS');
INSERT INTO DEPT VALUES (30, 'SALES',      'CHICAGO');
INSERT INTO DEPT VALUES (40, 'OPERATIONS', 'BOSTON');
```

```
1 row inserted.


1 row inserted.


1 row inserted.


1 row inserted.
```

In second place we will face to the “not enough value” because at the end we add a new column in our table by the name of the mobile_num and also we checked a condition that the numbers must start with the 06 so we will add some new value for the mobile_num in order to bypass the errors.



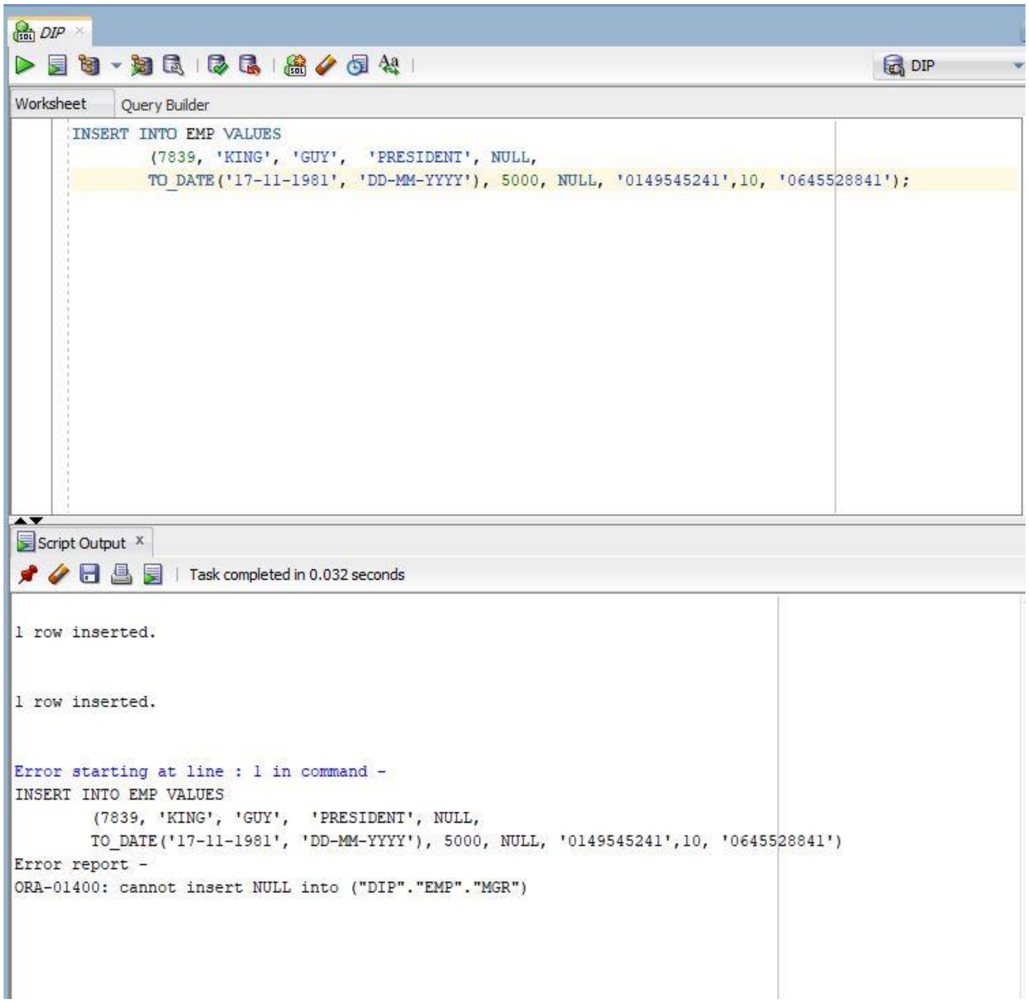
DIP.sql
DIP
Relational_2 (Untitled_1)
Relational_1 (Untitled_1)
0.81699997 seconds
DIP
Worksheet
Query Builder
Script Output
Task completed in 0.817 seconds

```
INSERT INTO EMP VALUES
        (7369, 'SMITH', 'JOHN', 'CLERK',     7902,
        TO_DATE('17-12-1980', 'DD-MM-YYYY'),  800, NULL,'0149545243', 20);
INSERT INTO EMP VALUES
        (7499, 'ALLEN', 'BOB', 'SALESMAN',  7698,
        TO_DATE('20-02-1981', 'DD-MM-YYYY'), 1600,  300, '0149547243',30);
INSERT INTO EMP VALUES
        (7521, 'WARD',  'PETER', 'SALESMAN',  7698,
        TO_DATE('22-02-1981', 'DD-MM-YYYY'), 1250,  500, '0149545247',30);
INSERT INTO EMP VALUES
        (7566, 'JONES', 'JOHN', 'MANAGER',   7839,
        TO_DATE('2-04-1981', 'DD-MM-YYYY'),  2975, NULL, '0149545456',20);
INSERT INTO EMP VALUES
        (7654, 'MARTIN', 'JOE','SALESMAN',  7698,
        TO_DATE('28-09-1981', 'DD-MM-YYYY'), 1250, 1400, '0149545784',30);
INSERT INTO EMP VALUES
        (7698, 'BLAKE', 'BOB', 'MANAGER',   7839,
```

```

Error starting at line : 7 in command -
INSERT INTO EMP VALUES
        (7369, 'SMITH', 'JOHN', 'CLERK',     7902,
        TO_DATE('17-12-1980', 'DD-MM-YYYY'),  800, NULL,'0149545243', 20)
Error at Command Line : 7 Column : 13
Error report -
SQL Error: ORA-00947: not enough values
00947. 00000 -  "not enough values"
*Cause:
*Action:

Error starting at line : 10 in command -
INSERT INTO EMP VALUES
        (7499, 'ALLEN', 'BOB', 'SALESMAN',  7698,
        TO_DATE('20-02-1981', 'DD-MM-YYYY'), 1600,  300, '0149547243',30)
Error at Command Line : 10 Column : 13
Error report -
SQL Error: ORA-00947: not enough values
00947. 00000 -  "not enough values"
*Cause:
*Action:

Error starting at line : 13 in command -
INSERT INTO EMP VALUES
        (7521, 'WARD',  'PETER', 'SALESMAN',  7698,
        TO_DATE('22-02-1981', 'DD-MM-YYYY'), 1250,  500, '0149545247',30)
Error at Command Line : 13 Column : 13
```

So after the adding data for new column it will run successfully.

The Last error is about “Null value” sice the MGR table is config as that it can not be null but in once of the queries the value for the MGT table is null so we need to change null value to a real number value.



```
INSERT INTO EMP VALUES
        (7839, 'KING', 'GUY',  'PRESIDENT', NULL,
        TO_DATE('17-11-1981', 'DD-MM-YYYY'), 5000, NULL, '0149545241',10, '0645528841');
```

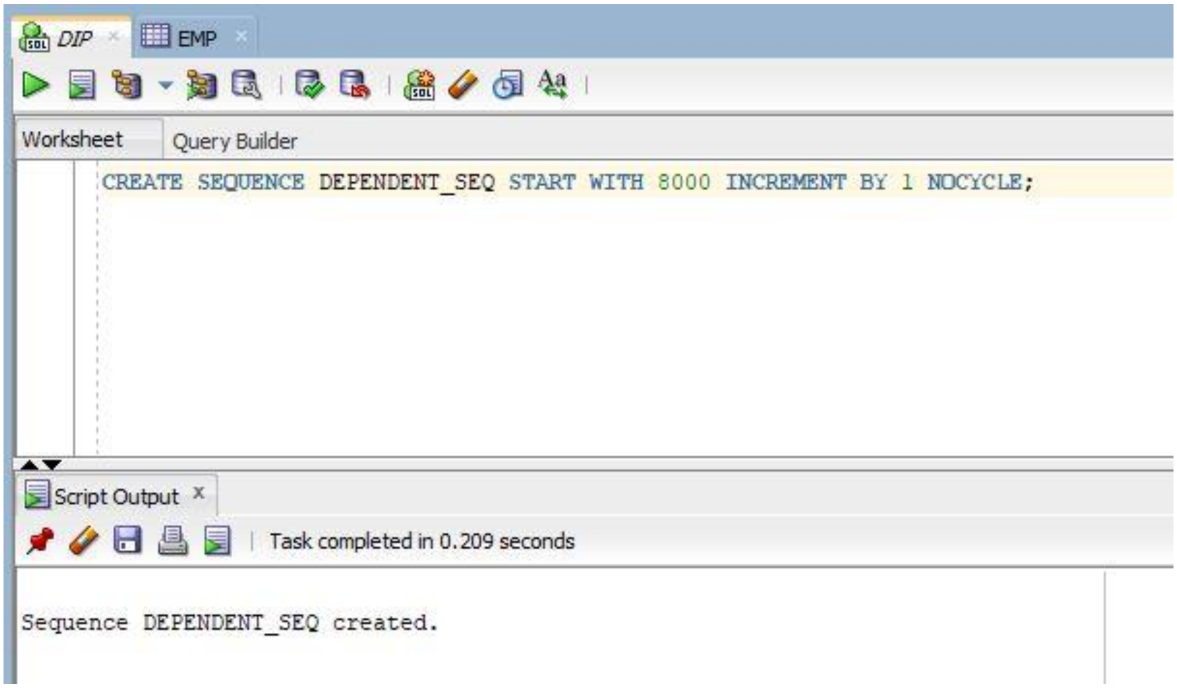
```
1 row inserted.


1 row inserted.


Error starting at line : 1 in command -
INSERT INTO EMP VALUES
        (7839, 'KING', 'GUY',  'PRESIDENT', NULL,
        TO_DATE('17-11-1981', 'DD-MM-YYYY'), 5000, NULL, '0149545241',10, '0645528841')
Error report -
ORA-01400: cannot insert NULL into ("DIP"."EMP"."MGR")
```

We can see that the MGR column value is NULL value so we need to change it after that it will successfully run. That I did.

And the last data for the SALGRADE table will run successfully because everything is fine.

Answer 7:

According to the syntax that was in our tp question I create the Dependent_seq and it starts from 8000 and every time increment by 1 **and** The "NO CYCLE" option means that the sequence will not wrap around to the starting value once it reaches the maximum value.

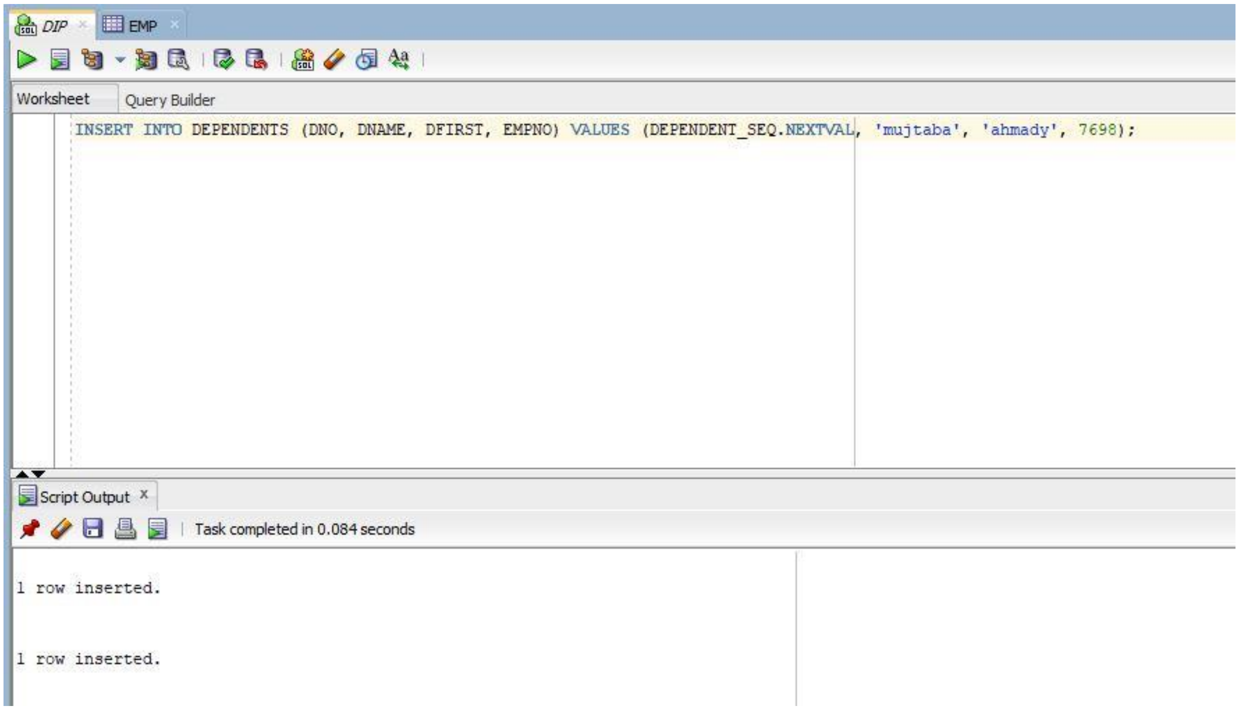


```
CREATE SEQUENCE DEPENDENT_SEQ START WITH 8000 INCREMENT BY 1 NOCYCLE;
```

```
Sequence DEPENDENT_SEQ created.
```

Answer 8:

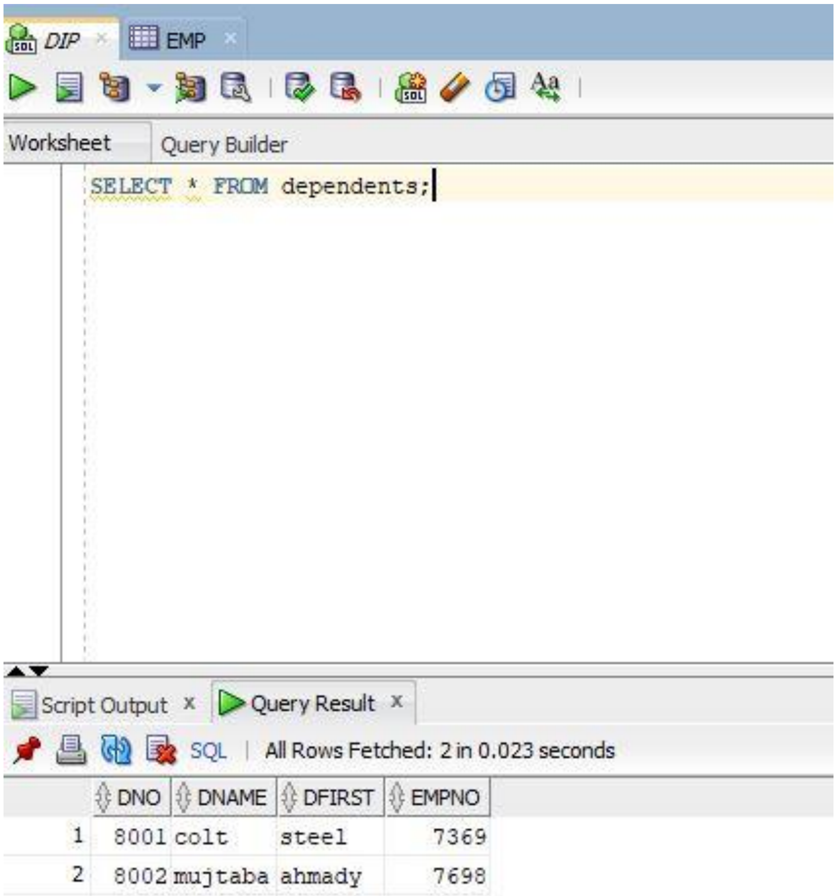
After creating the Dependent_seq using it I could successfully could add some data and it works successfully and using the NEXVAL we specifi that every time the query runs add should generate a new unique base of the sequence that we identified before.



```
INSERT INTO DEPENDENTS (DNO, DNAME, DFIRST, EMPNO) VALUES (DEPENDENT_SEQ.NEXTVAL, 'mujtaba', 'ahmady', 7698);
```

```
1 row inserted.


1 row inserted.
```

We can see that two row with different DNO added successfully.



```
SELECT * FROM dependents;
```

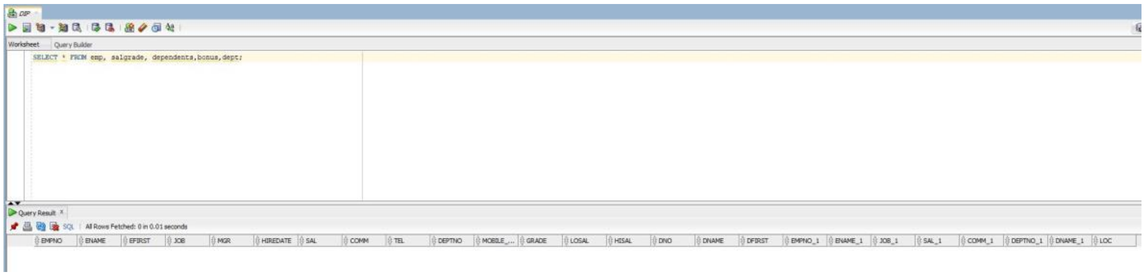
All Rows Fetched: 2 in 0.023 seconds

	DNO	DNAME	DFIRST	EMPNO
1	8001	colt	steel	7369
2	8002	mujtaba	ahmady	7698

Section 2

Answer 1:

Using the "SELECT *" we can list all the content of a table also we can see the attributes of the tables so we can use it on a single table or on all tables but if you use on all tables at once you are not able to see the contents of the tables.

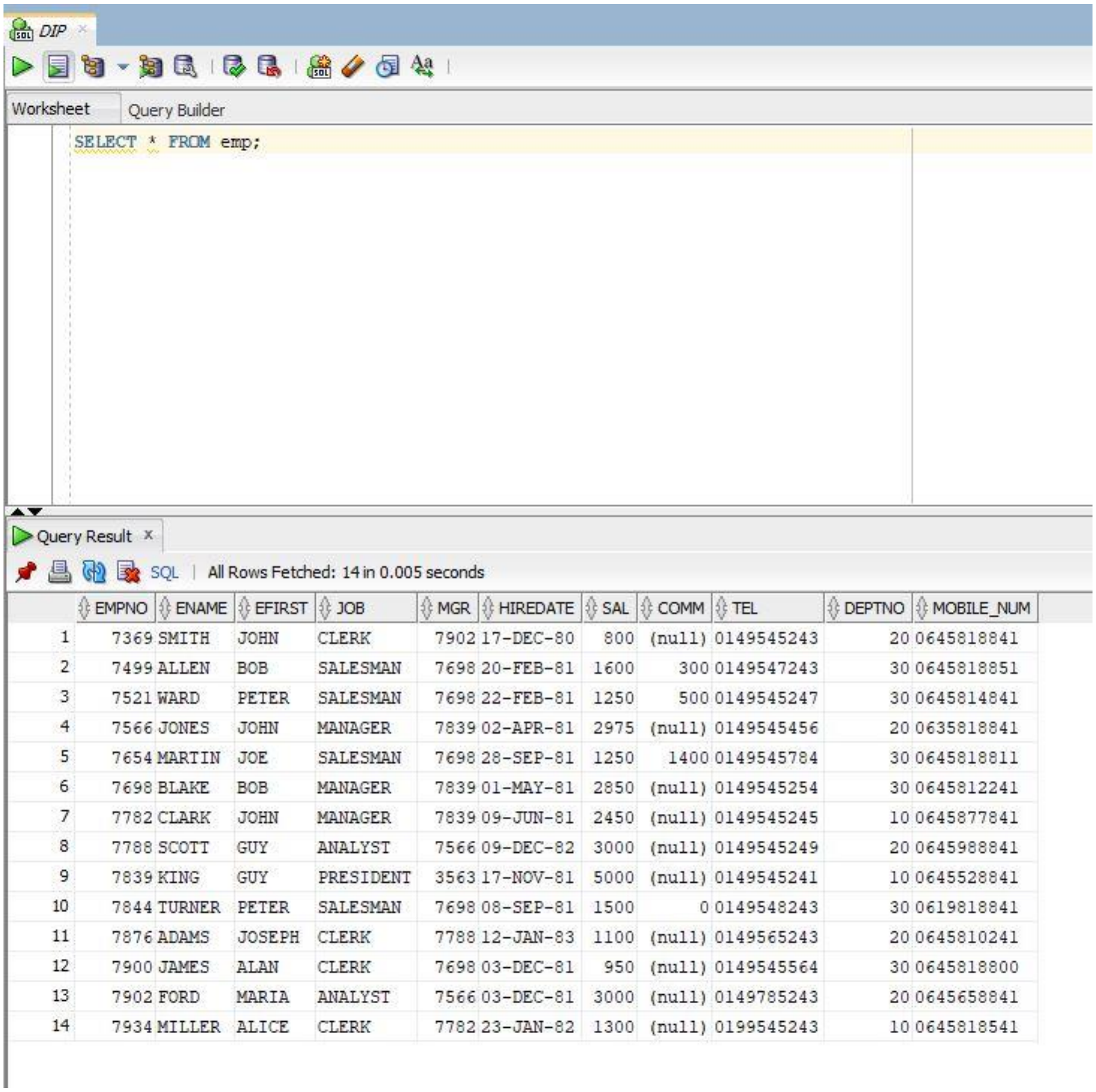


```
SELECT * FROM emp, salgrade, dependents,bonus,dept;
```

All Rows Fetched: 0 in 0.01 seconds

EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_...	GRADE	LOSAL	HISAL	DNO	DNAME	DFIRST	EMPNO_1	ENAME_1	JOB_1	SAL_1	COMM_1	DEPTNO_1	DNAME_1	LOC

So If you want to list all the content of a table with their headers and attributes you need to use the same query but just use one single table.



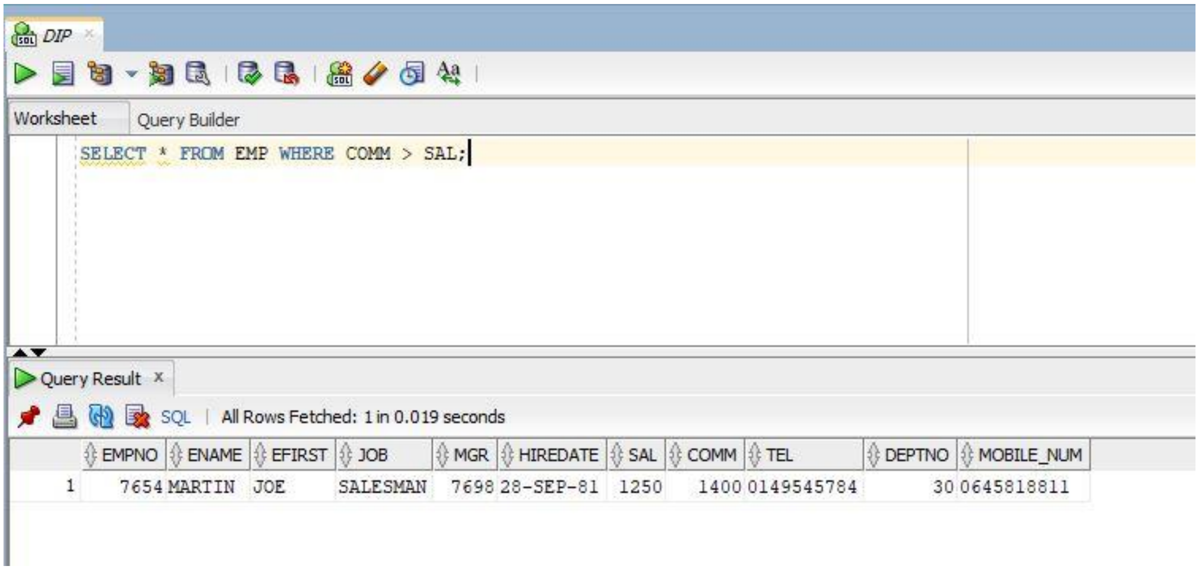
```
SELECT * FROM emp;
```

All Rows Fetched: 14 in 0.005 seconds

	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

Answer 2:

OK since we want those employees that their commission is bigger that their salary so we need to use comparative operators since we want to list those employees that their commission is bigger than their salary we use bigger sign.



```
SELECT * FROM EMP WHERE COMM > SAL;
```

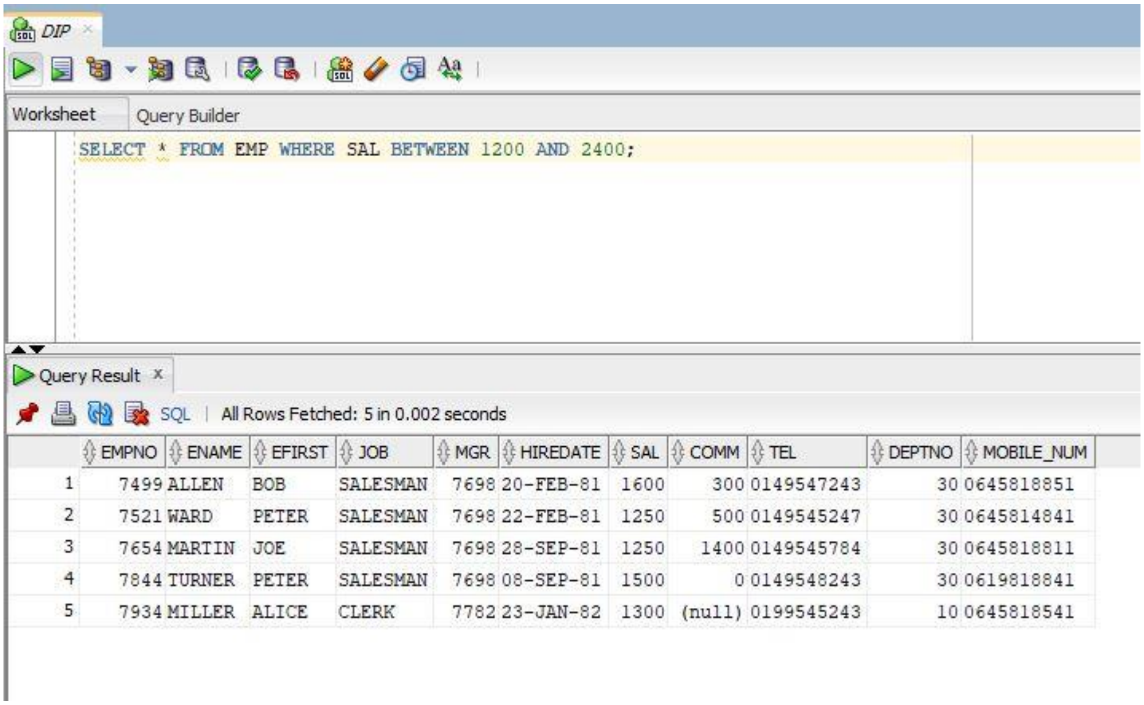
All Rows Fetched: 1 in 0.019 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7654	MARTIN	JOE	SALESMAN	7698	28-SEP-81	1250	1400	0149545784	30	0645818811

Answer 3:

So when we say earning it depends on that only they asked as about salary or commission or both so according to that you can see that we can write different type of query.

If they mean only the salary so we have only 5 Employees that their income is between 1200 and 2400.

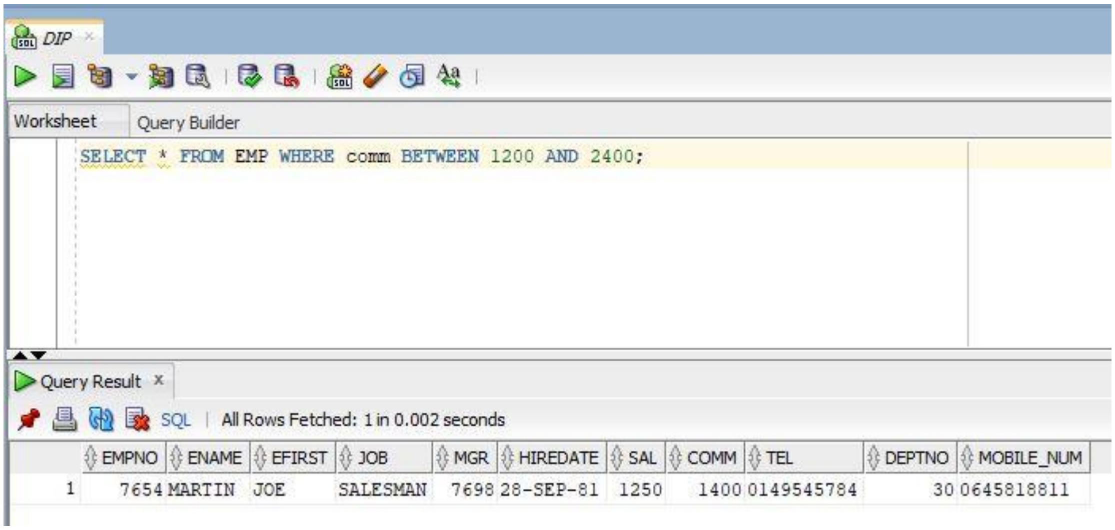


```
SELECT * FROM EMP WHERE SAL BETWEEN 1200 AND 2400;
```

All Rows Fetched: 5 in 0.002 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7499	ALLEN	BOB	SALESMAN	7698	20-FEB-81	1600	300	0149547243	30	0645818851
2	7521	WARD	PETER	SALESMAN	7698	22-FEB-81	1250	500	0149545247	30	0645814841
3	7654	MARTIN	JOE	SALESMAN	7698	28-SEP-81	1250	1400	0149545784	30	0645818811
4	7844	TURNER	PETER	SALESMAN	7698	08-SEP-81	1500	0	0149548243	30	0619818841
5	7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300	(null)	0199545243	10	0645818541

If they mean only commission so we have only 1 Employee that his income is between 1200 and 2400.

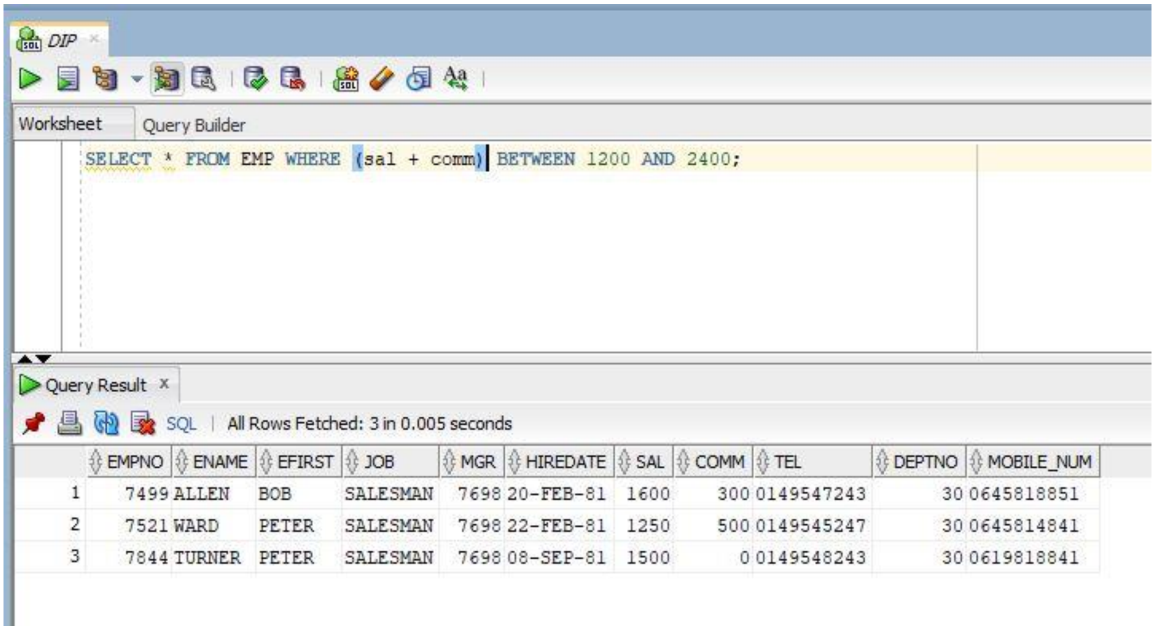


```
SELECT * FROM EMP WHERE comm BETWEEN 1200 AND 2400;
```

All Rows Fetched: 1 in 0.002 seconds

```
     EMPNO ENAME   EFIRST JOB       MGR  HIREDATE  SAL  COMM TEL        DEPTNO MOBILE_NUM
1     7654 MARTIN  JOE    SALESMAN  7698 28-SEP-81 1250 1400 0149545784     30 0645818811
```

So finally, if they mean the sum of salary and commission by the earing we can use the below query to check it and we can see that only we have 3 Employees.



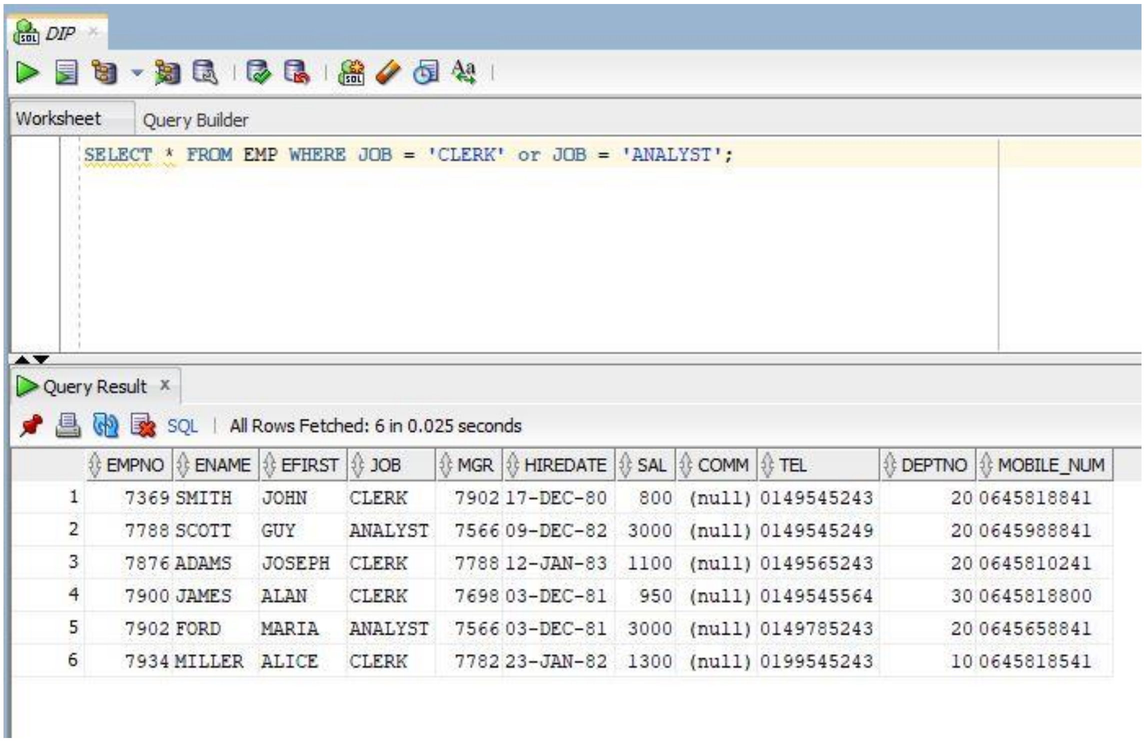
```
SELECT * FROM EMP WHERE (sal + comm) BETWEEN 1200 AND 2400;
```

All Rows Fetched: 3 in 0.005 seconds

```
     EMPNO ENAME   EFIRST JOB       MGR  HIREDATE  SAL  COMM TEL        DEPTNO MOBILE_NUM
1     7499 ALLEN   BOB    SALESMAN  7698 20-FEB-81 1600  300 0149547243     30 0645818851
2     7521 WARD    PETER  SALESMAN  7698 22-FEB-81 1250  500 0149545247     30 0645814841
3     7844 TURNER  PETER  SALESMAN  7698 08-SEP-81 1500    0 0149548243     30 0619818841
```

Answer 4:

if you want to list those employees that their job is clerk or analyst we need to check using = comparative operator so we can see that we have 6 employees with the mentioned job.



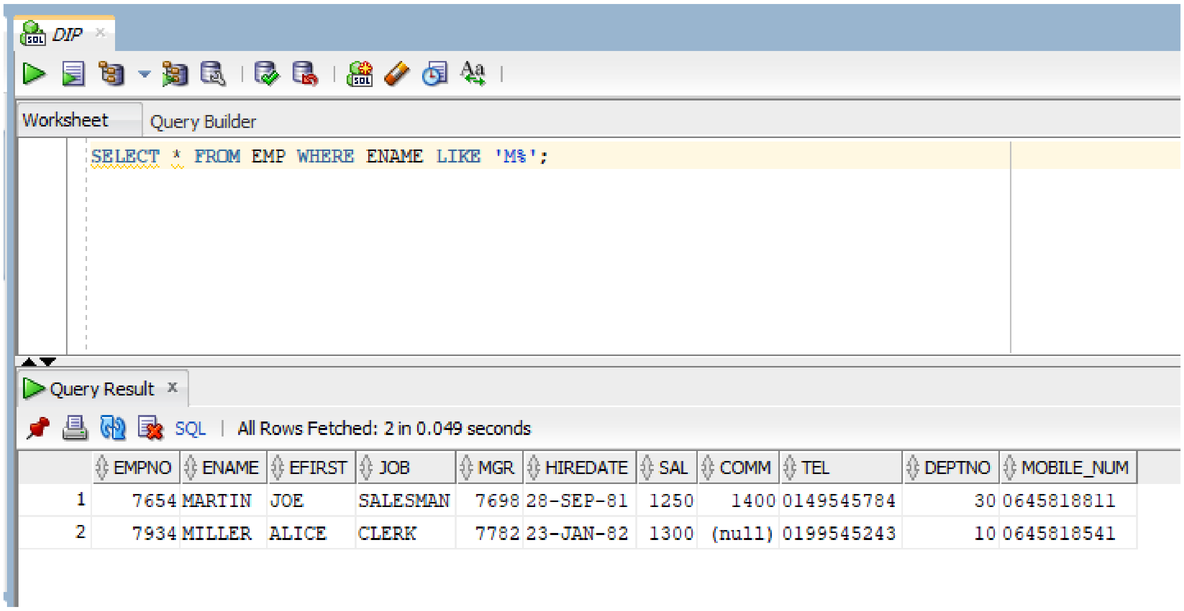
```sql
SELECT * FROM EMP WHERE JOB = 'CLERK' or JOB = 'ANALYST';
```

All Rows Fetched: 6 in 0.025 seconds

	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	7788	SCOTT	GUY	ANALYST	7566	09-DEC-82	3000	(null)	0149545249	20	0645988841
3	7876	ADAMS	JOSEPH	CLERK	7788	12-JAN-83	1100	(null)	0149565243	20	0645810241
4	7900	JAMES	ALAN	CLERK	7698	03-DEC-81	950	(null)	0149545564	30	0645818800
5	7902	FORD	MARIA	ANALYST	7566	03-DEC-81	3000	(null)	0149785243	20	0645658841
6	7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300	(null)	0199545243	10	0645818541

Answer 5:

For listing those employees that their name start with M we need to use LIKE function with the below pattern and we can see that we have only 2 employees with that their names start with M.



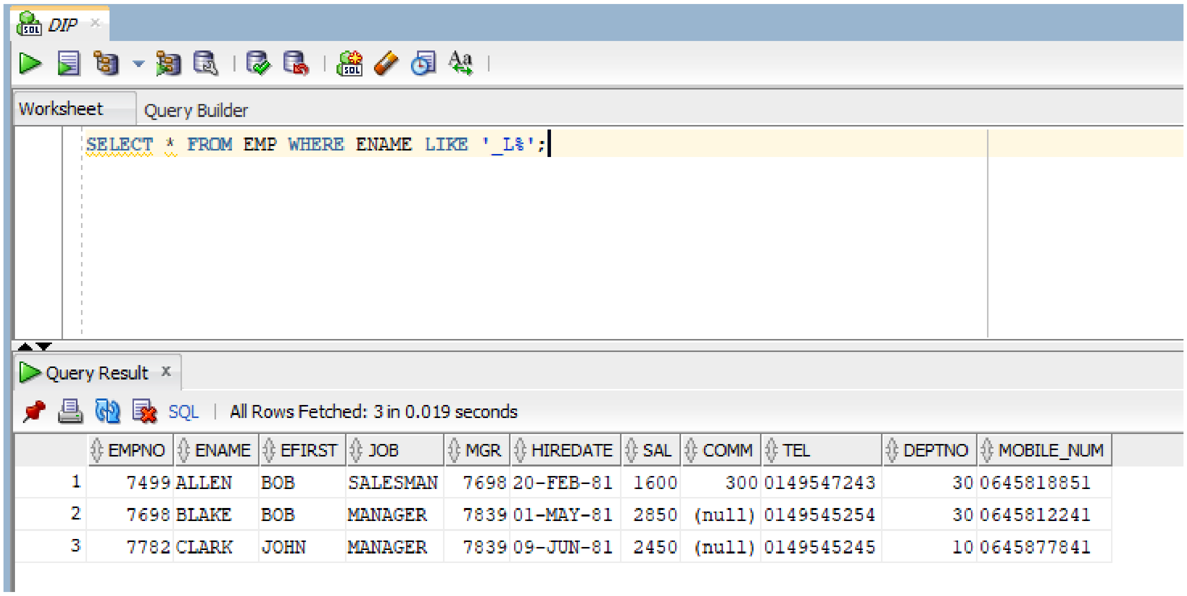
```sql
SELECT * FROM EMP WHERE ENAME LIKE 'M%';
```

All Rows Fetched: 2 in 0.049 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7654	MARTIN	JOE	SALESMAN	7698	28-SEP-81	1250	1400	0149545784	30	0645818811
2	7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300	(null)	0199545243	10	0645818541

Answer 6:

If you want to list those employees that the second character name of their names start with the L you can use the following patter and we can see that we only have 3 user with the same condition.



```
SELECT * FROM EMP WHERE ENAME LIKE '_L%';
```

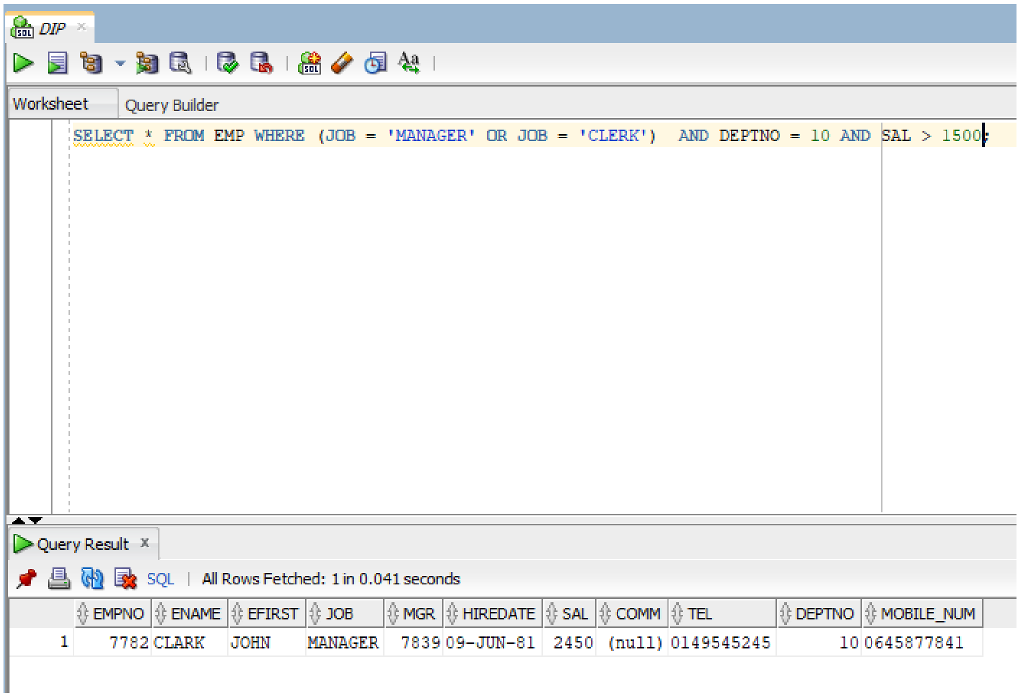
All Rows Fetched: 3 in 0.019 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7499	ALLEN	BOB	SALESMAN	7698	20-FEB-81	1600	300	0149547243	30	0645818851
2	7698	BLAKE	BOB	MANAGER	7839	01-MAY-81	2850	(null)	0149545254	30	0645812241
3	7782	CLARK	JOHN	MANAGER	7839	09-JUN-81	2450	(null)	0149545245	10	0645877841

Answer 7:

For this query we can a simple traditional query to check and also we can use join you can see that both of them have the same out put.

Simple Query

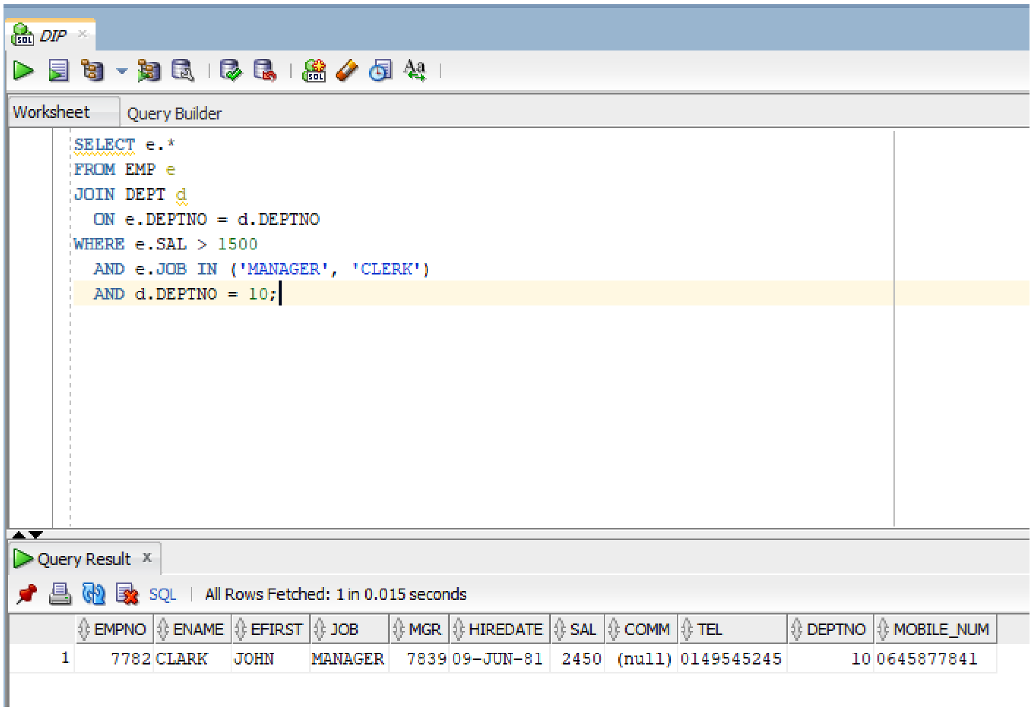


```
SELECT * FROM EMP WHERE (JOB = 'MANAGER' OR JOB = 'CLERK')  AND DEPTNO = 10 AND SAL > 1500
```

All Rows Fetched: 1 in 0.041 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7782	CLARK	JOHN	MANAGER	7839	09-JUN-81	2450	(null)	0149545245	10	0645877841

Join Query



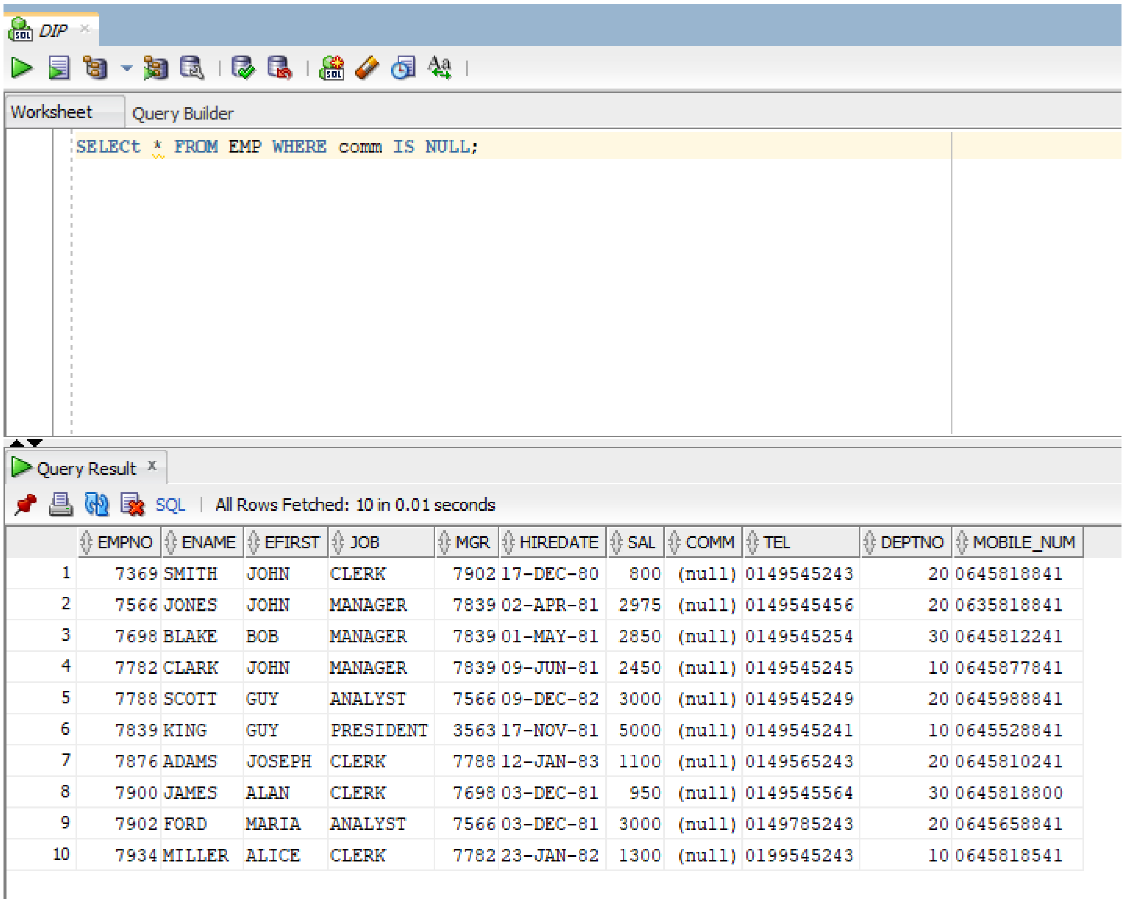
```
SELECT e.*
FROM EMP e
JOIN DEPT d
  ON e.DEPTNO = d.DEPTNO
WHERE e.SAL > 1500
  AND e.JOB IN ('MANAGER', 'CLERK')
  AND d.DEPTNO = 10;
```

All Rows Fetched: 1 in 0.015 seconds

```
    EMPNO ENAME EFIRST JOB     MGR  HIREDATE   SAL  COMM   TEL        DEPTNO MOBILE_NUM
  1  7782 CLARK JOHN   MANAGER 7839 09-JUN-81 2450 (null) 0149545245     10 0645877841
```

Answer 8:

If you want to select null values in a column, we need to use the is null key work so we can see we have 10 employees that their commission is null.



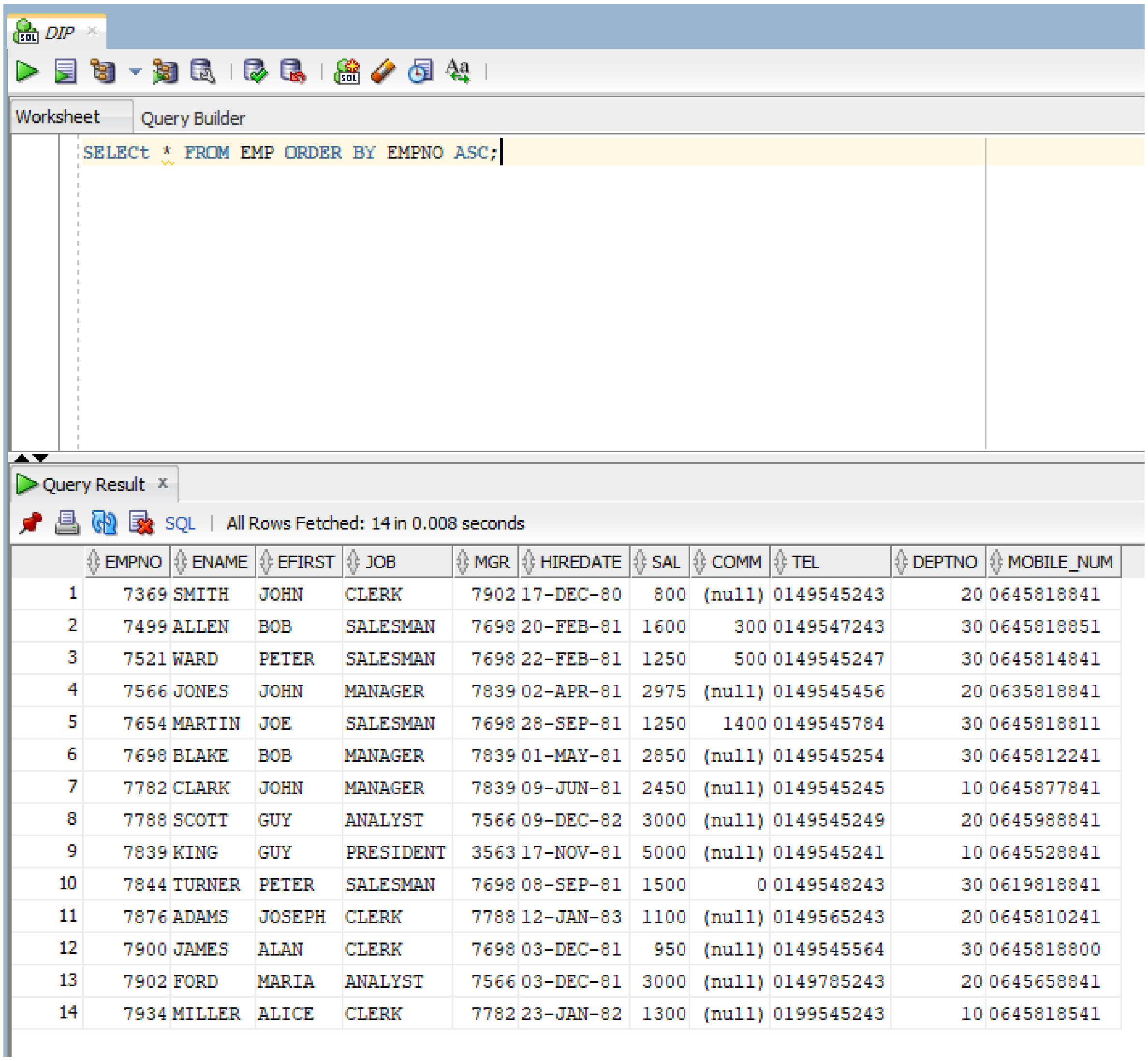
```
SELECt * FROM EMP WHERE comm IS NULL;
```

SQL | All Rows Fetched: 10 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	800	(null)	0149545243	20	0645818841
2	7566	JONES	JOHN	MANAGER	7839	02-APR-81	2975	(null)	0149545456	20	0635818841
3	7698	BLAKE	BOB	MANAGER	7839	01-MAY-81	2850	(null)	0149545254	30	0645812241
4	7782	CLARK	JOHN	MANAGER	7839	09-JUN-81	2450	(null)	0149545245	10	0645877841
5	7788	SCOTT	GUY	ANALYST	7566	09-DEC-82	3000	(null)	0149545249	20	0645988841
6	7839	KING	GUY	PRESIDENT	3563	17-NOV-81	5000	(null)	0149545241	10	0645528841
7	7876	ADAMS	JOSEPH	CLERK	7788	12-JAN-83	1100	(null)	0149565243	20	0645810241
8	7900	JAMES	ALAN	CLERK	7698	03-DEC-81	950	(null)	0149545564	30	0645818800
9	7902	FORD	MARIA	ANALYST	7566	03-DEC-81	3000	(null)	0149785243	20	0645658841
10	7934	MILLER	ALICE	CLERK	7782	23-JAN-82	1300	(null)	0199545243	10	0645818541

Answer 9:

For ordering all employees in Ascending we can use ORDER BY name of column and ASC key work.



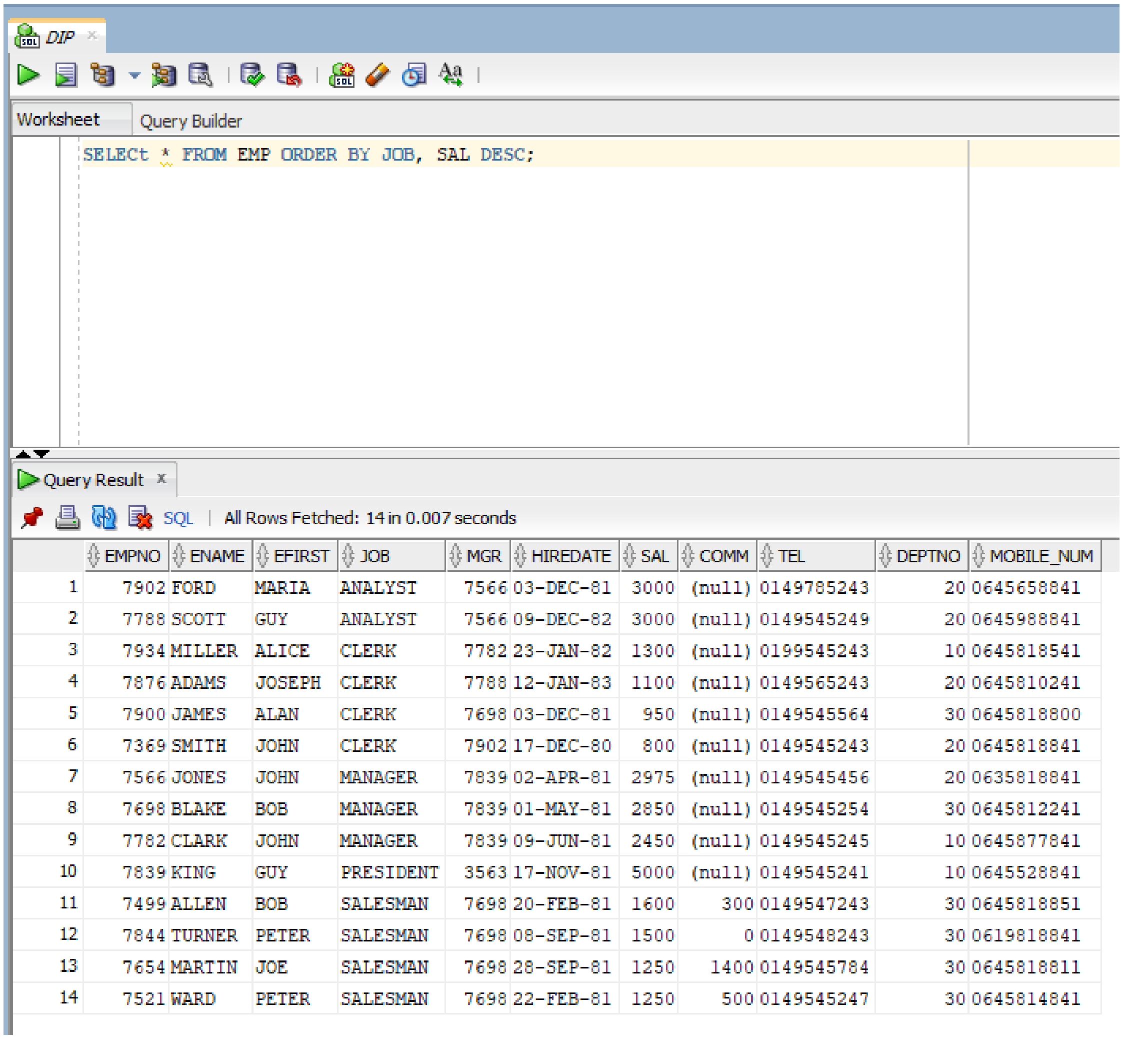
```
SELECt * FROM EMP ORDER BY EMPNO ASC;
```

All Rows Fetched: 14 in 0.008 seconds

	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

Answer 10:

Since we have more than one employee in one position and the salary in the same position is different for different employees we ordered by descending to list employees in the same position from biggest to smallest.



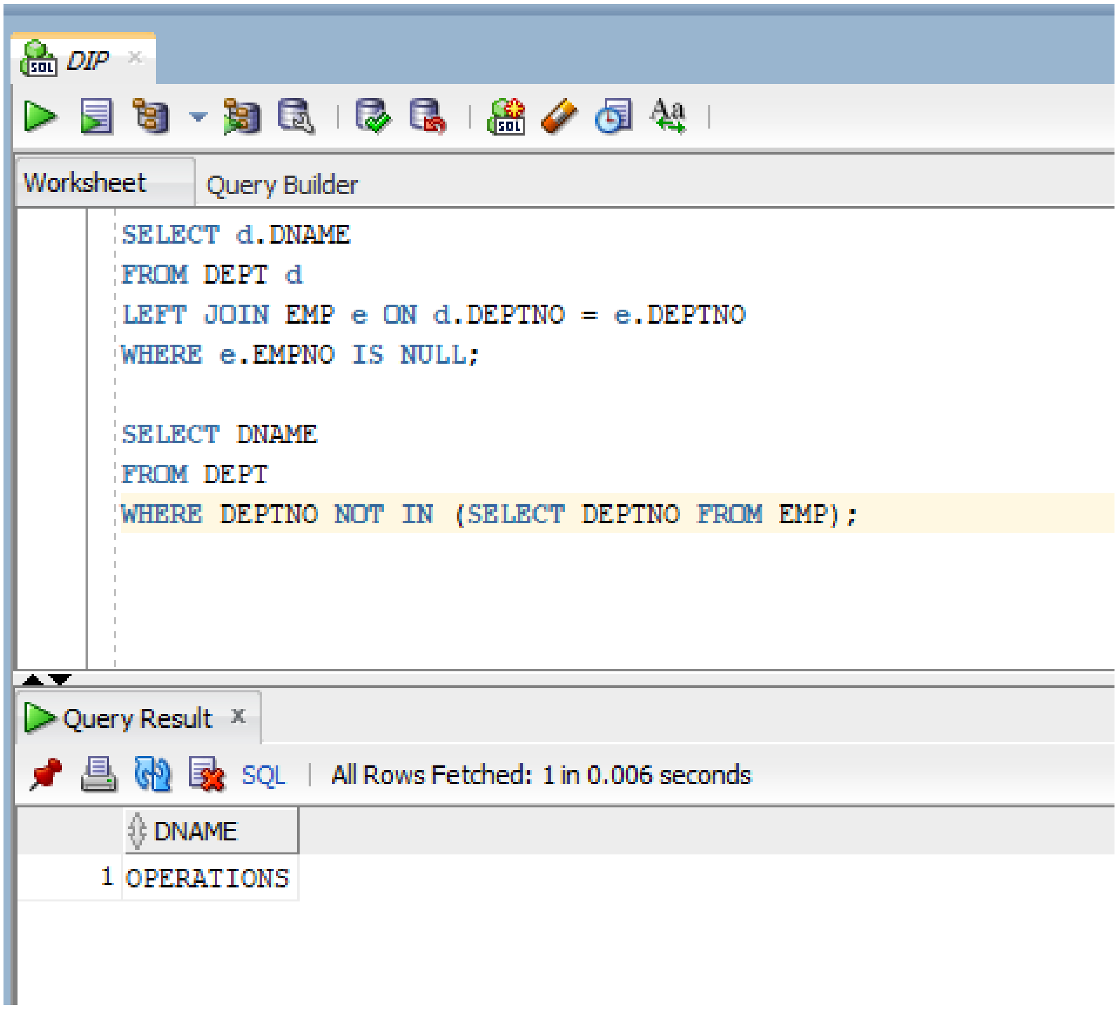
```
SELECt * FROM EMP ORDER BY JOB, SAL DESC;
```

All Rows Fetched: 14 in 0.007 seconds

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

Answer 11:

For this question we can write two query to check those departments without employees we can LEFT Join also we can Use NOT IN both have the same out put.



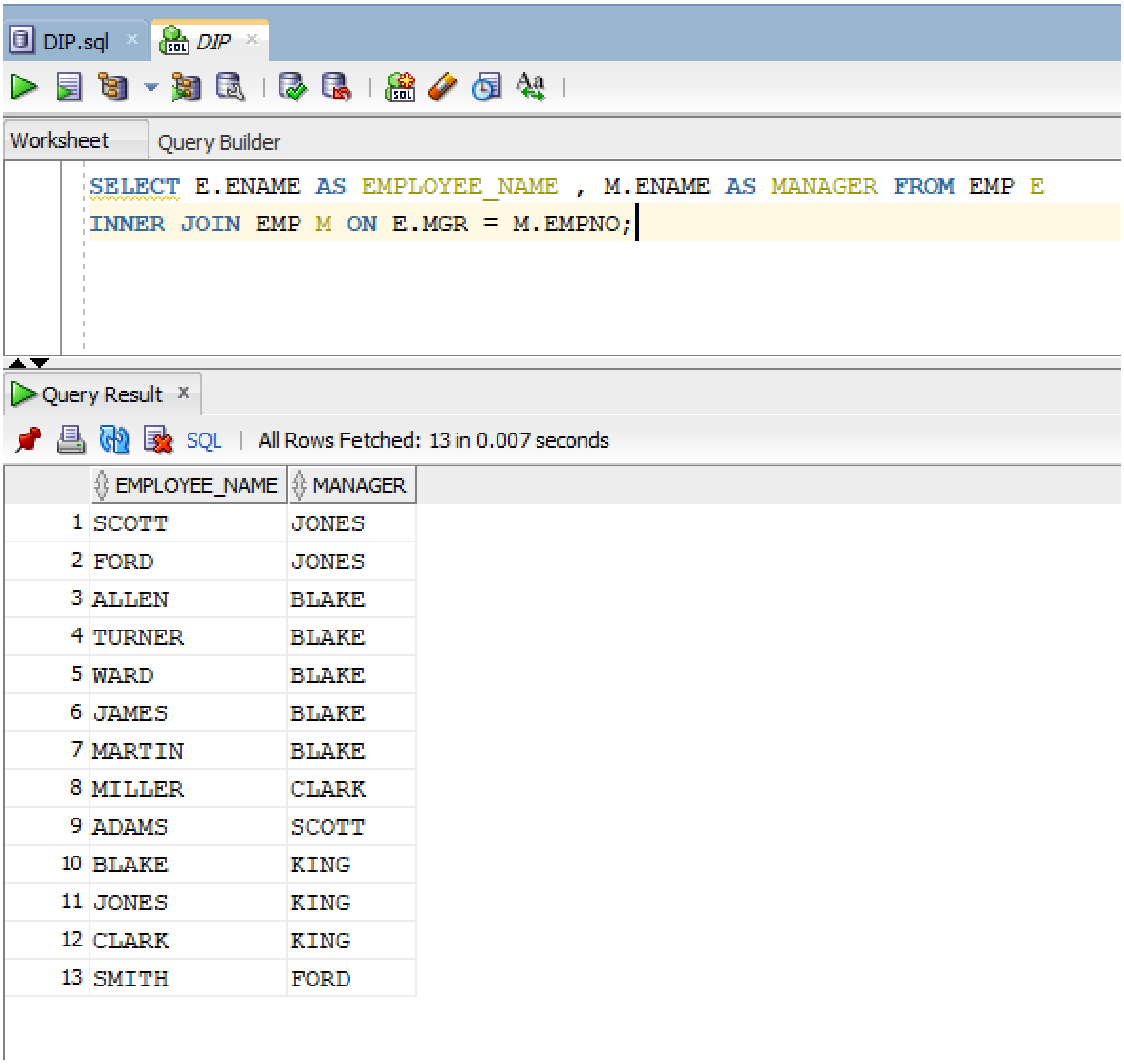
```
SELECT d.DNAME
FROM DEPT d
LEFT JOIN EMP e ON d.DEPTNO = e.DEPTNO
WHERE e.EMPNO IS NULL;

SELECT DNAME
FROM DEPT
WHERE DEPTNO NOT IN (SELECT DEPTNO FROM EMP);
```

All Rows Fetched: 1 in 0.006 seconds

	DNAME
1	OPERATIONS

Answer 12:

So since we don’t have two table to select data from it. We have only one table of employees which inside that we have the employee and manager both so we will use inner join and create to alias and join the table with it self.



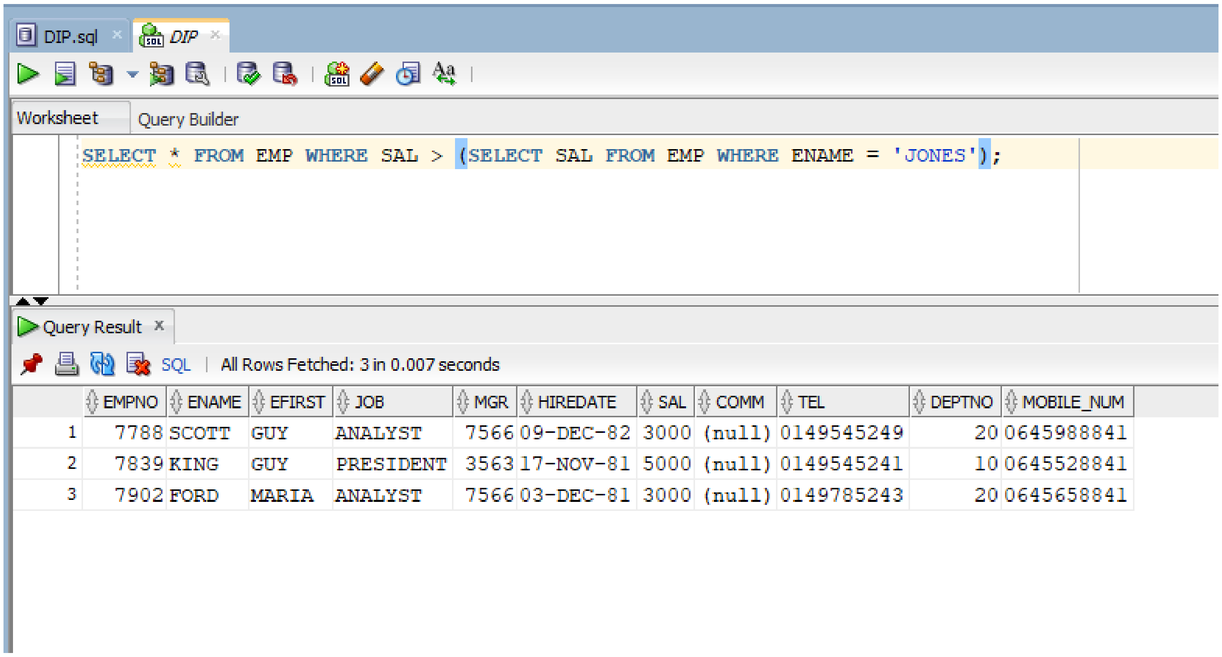
```
SELECT E.ENAME AS EMPLOYEE_NAME , M.ENAME AS MANAGER FROM EMP E
INNER JOIN EMP M ON E.MGR = M.EMPNO;
```

All Rows Fetched: 13 in 0.007 seconds

	EMPLOYEE_NAME	MANAGER
1	SCOTT	JONES
2	FORD	JONES
3	ALLEN	BLAKE
4	TURNER	BLAKE
5	WARD	BLAKE
6	JAMES	BLAKE
7	MARTIN	BLAKE
8	MILLER	CLARK
9	ADAMS	SCOTT
10	BLAKE	KING
11	JONES	KING
12	CLARK	KING
13	SMITH	FORD

Answer 13:

So for this question we need to use sub query. One Query to check the salary od JONES and second query to check the employees that there salary is bigger than JONES.



```
SELECT * FROM EMP WHERE SAL > (SELECT SAL FROM EMP WHERE ENAME = 'JONES');
```

All Rows Fetched: 3 in 0.007 seconds

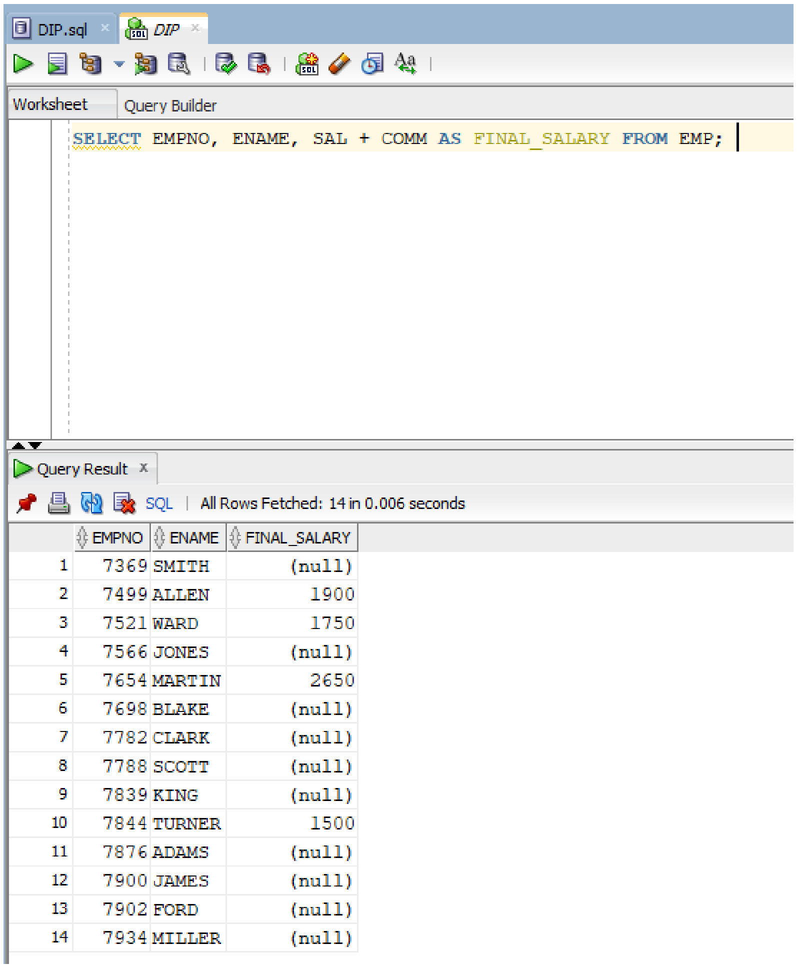
	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7788	SCOTT	GUY	ANALYST	7566	09-DEC-82	3000	(null)	0149545249	20	0645988841
2	7839	KING	GUY	PRESIDENT	3563	17-NOV-81	5000	(null)	0149545241	10	0645528841
3	7902	FORD	MARIA	ANALYST	7566	03-DEC-81	3000	(null)	0149785243	20	0645658841

Answer 14:

So for this we can write two different type of query one simple query and second with condition

If we write a simple query we will face to some problems because every employee have salary but not commission so those employees that don't have commissions they will display null. But with the check condition we can check if the commission was null just show us the salary.

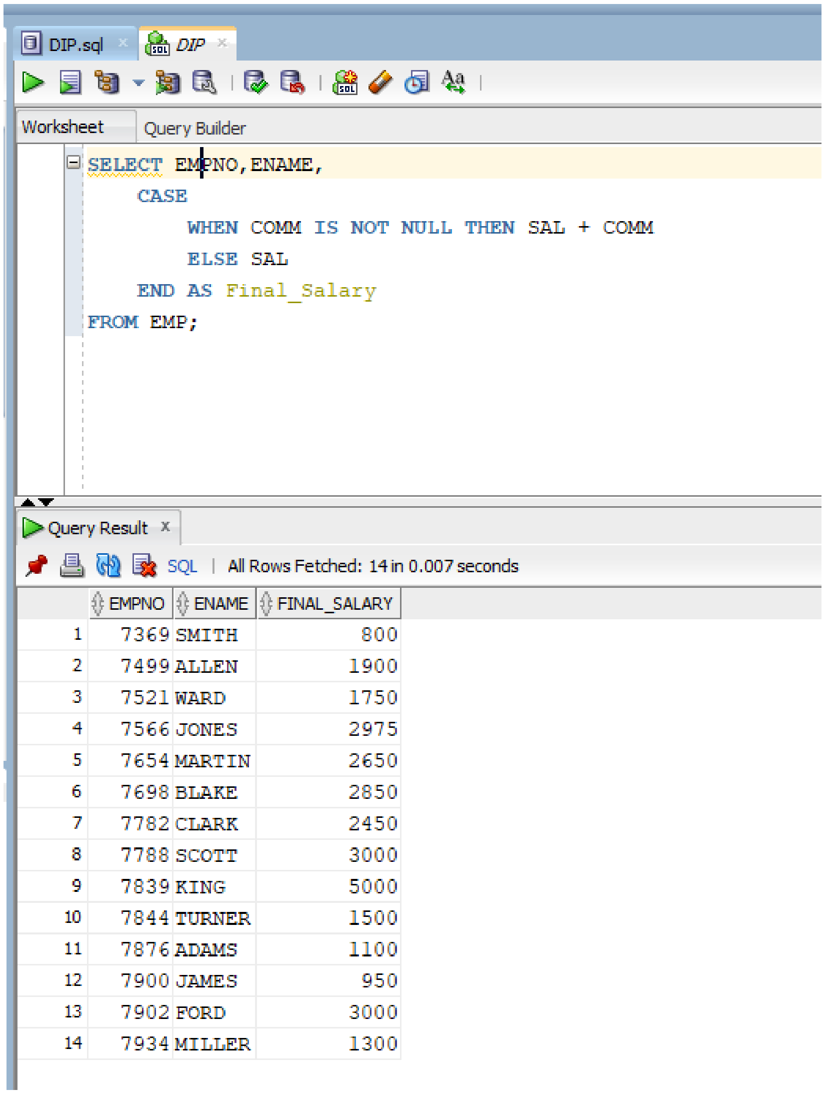
Simple query.



```
SELECT EMPNO, ENAME, SAL + COMM AS FINAL_SALARY FROM EMP;
```

	EMPNO	ENAME	FINAL_SALARY
1	7369	SMITH	(null)
2	7499	ALLEN	1900
3	7521	WARD	1750
4	7566	JONES	(null)
5	7654	MARTIN	2650
6	7698	BLAKE	(null)
7	7782	CLARK	(null)
8	7788	SCOTT	(null)
9	7839	KING	(null)
10	7844	TURNER	1500
11	7876	ADAMS	(null)
12	7900	JAMES	(null)
13	7902	FORD	(null)
14	7934	MILLER	(null)

Condition Query.



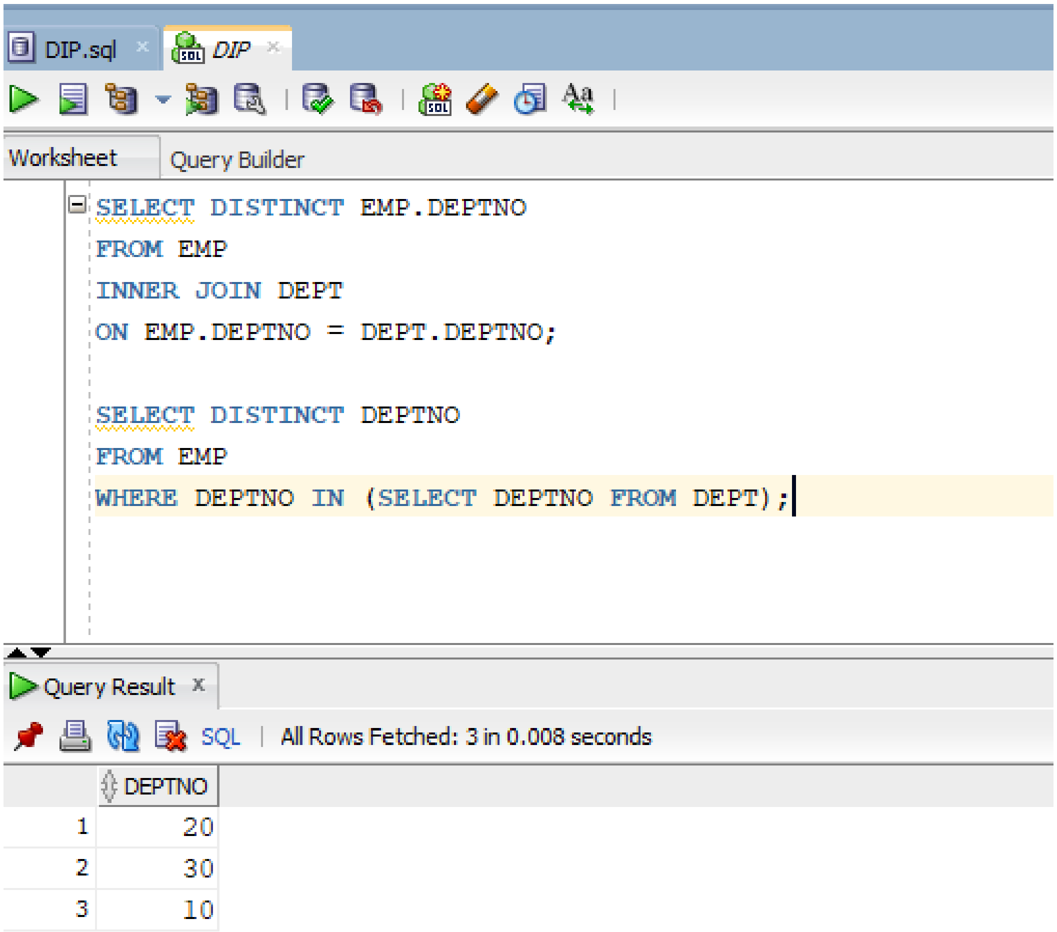
```
SELECT EMPNO,ENAME,
    CASE
        WHEN COMM IS NOT NULL THEN SAL + COMM
        ELSE SAL
    END AS Final_Salary
FROM EMP;
```

All Rows Fetched: 14 in 0.007 seconds

	EMPNO	ENAME	FINAL_SALARY
1	7369	SMITH	800
2	7499	ALLEN	1900
3	7521	WARD	1750
4	7566	JONES	2975
5	7654	MARTIN	2650
6	7698	BLAKE	2850
7	7782	CLARK	2450
8	7788	SCOTT	3000
9	7839	KING	5000
10	7844	TURNER	1500
11	7876	ADAMS	1100
12	7900	JAMES	950
13	7902	FORD	3000
14	7934	MILLER	1300

Answer 15:

So for this we can write also two queries one simple and one with join.



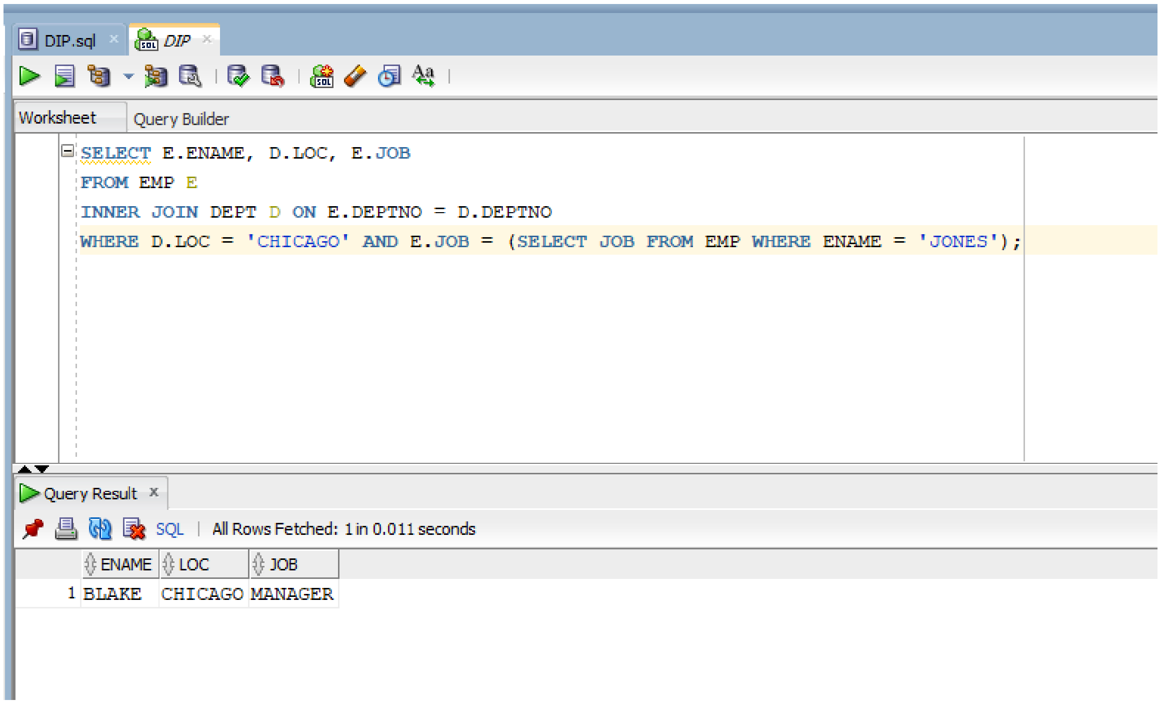
```
SELECT DISTINCT EMP.DEPTNO
FROM EMP
INNER JOIN DEPT
ON EMP.DEPTNO = DEPT.DEPTNO;

SELECT DISTINCT DEPTNO
FROM EMP
WHERE DEPTNO IN (SELECT DEPTNO FROM DEPT);
```

All Rows Fetched: 3 in 0.008 seconds

	DEPTNO
1	20
2	30
3	10

Answer 16:

For listing those employees that work in CHICAGO and have the same job the best way is to use join because it is much more easy that a simple query.



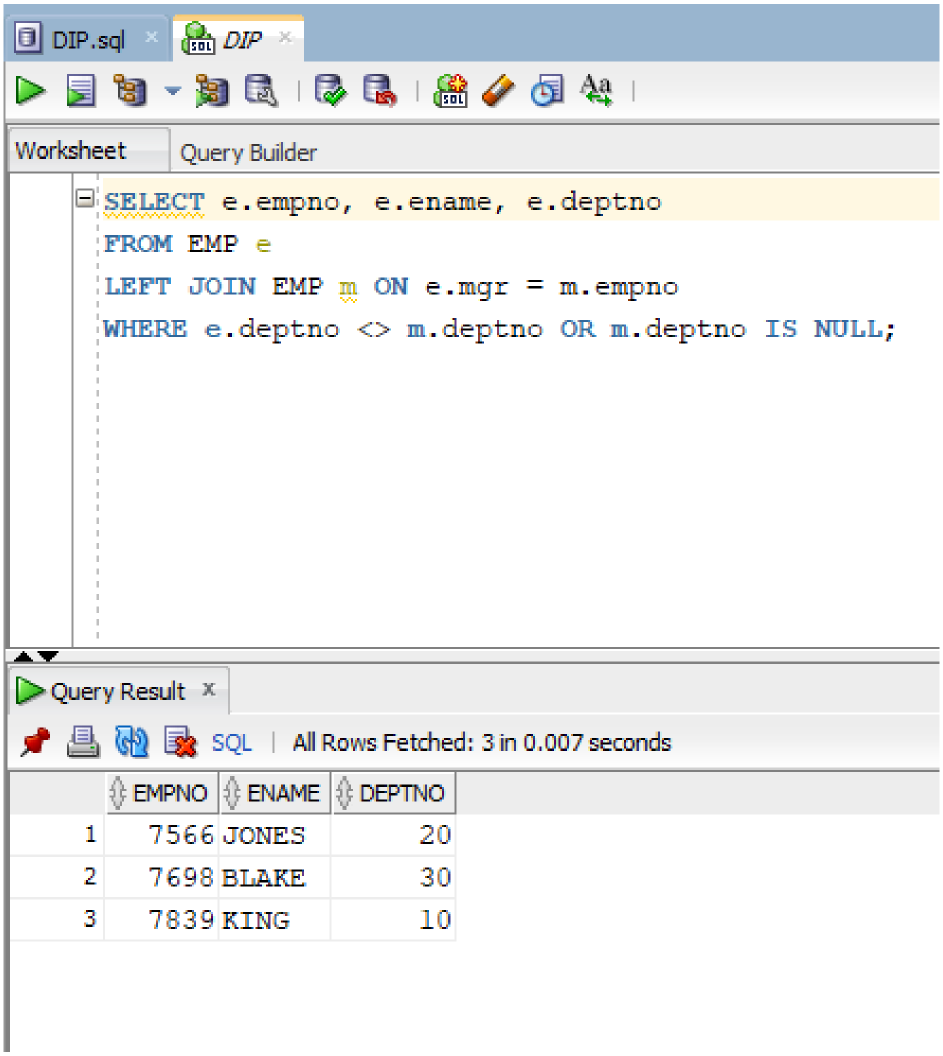
```
SELECT E.ENAME, D.LOC, E.JOB
FROM EMP E
INNER JOIN DEPT D ON E.DEPTNO = D.DEPTNO
WHERE D.LOC = 'CHICAGO' AND E.JOB = (SELECT JOB FROM EMP WHERE ENAME = 'JONES');
```

All Rows Fetched: 1 in 0.011 seconds

	ENAME	LOC	JOB
1	BLAKE	CHICAGO	MANAGER

Answer 17:

For This question since the Manager and employees are in the same table we can use a left join and join the table with it self.



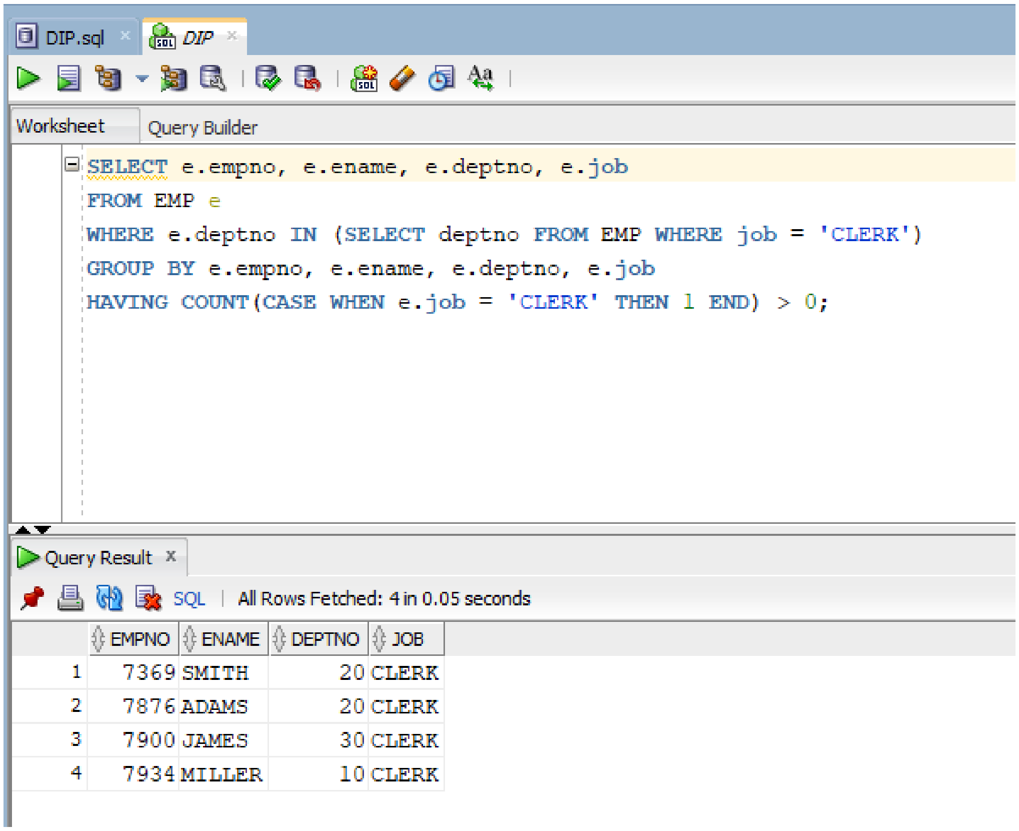
```sql
SELECT e.empno, e.ename, e.deptno
FROM EMP e
LEFT JOIN EMP m ON e.mgr = m.empno
WHERE e.deptno <> m.deptno OR m.deptno IS NULL;
```

All Rows Fetched: 3 in 0.007 seconds

	EMPNO	ENAME	DEPTNO
1	7566	JONES	20
2	7698	BLAKE	30
3	7839	KING	10

Answer 18:

So for this question we need to use a sub query to retrieve the job and having condition to check that the job clerk is at least one



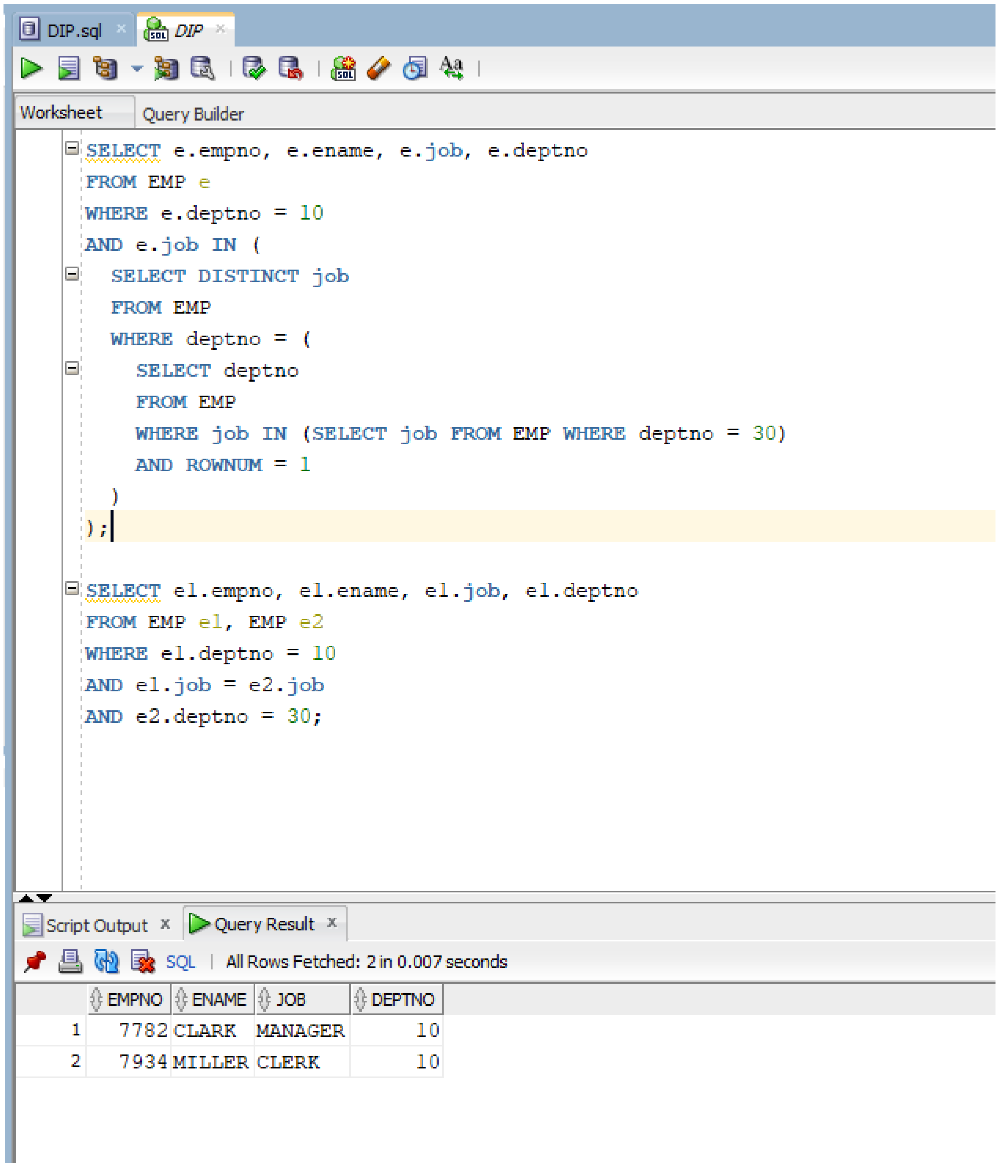
```
SELECT e.empno, e.ename, e.deptno, e.job
FROM EMP e
WHERE e.deptno IN (SELECT deptno FROM EMP WHERE job = 'CLERK')
GROUP BY e.empno, e.ename, e.deptno, e.job
HAVING COUNT(CASE WHEN e.job = 'CLERK' THEN 1 END) > 0;
```

All Rows Fetched: 4 in 0.05 seconds

	EMPNO	ENAME	DEPTNO	JOB
1	7369	SMITH	20	CLERK
2	7876	ADAMS	20	CLERK
3	7900	JAMES	30	CLERK
4	7934	MILLER	10	CLERK

Answer 19:

For this query we can write two different kind of query simple query or sub query both have the same out put.



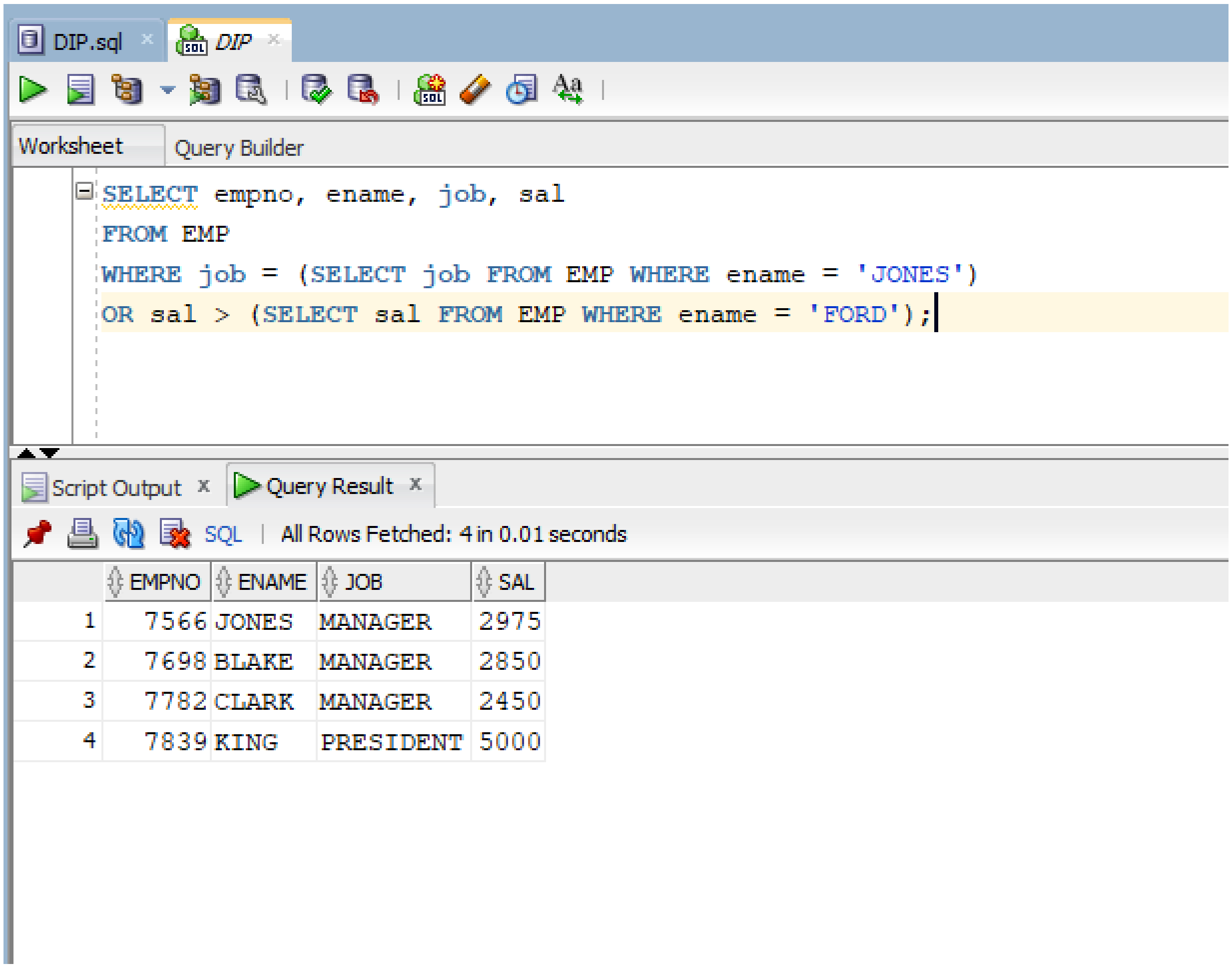
```sql
SELECT e.empno, e.ename, e.job, e.deptno
FROM EMP e
WHERE e.deptno = 10
AND e.job IN (
  SELECT DISTINCT job
  FROM EMP
  WHERE deptno = (
    SELECT deptno
    FROM EMP
    WHERE job IN (SELECT job FROM EMP WHERE deptno = 30)
    AND ROWNUM = 1
  )
);

SELECT e1.empno, e1.ename, e1.job, e1.deptno
FROM EMP e1, EMP e2
WHERE e1.deptno = 10
AND e1.job = e2.job
AND e2.deptno = 30;
```

All Rows Fetched: 2 in 0.007 seconds

	EMPNO	ENAME	JOB	DEPTNO
1	7782	CLARK	MANAGER	10
2	7934	MILLER	CLERK	10

Answer 20:

For this query we can use left join and also it’s possible with a simple query I prefer to write with simple query



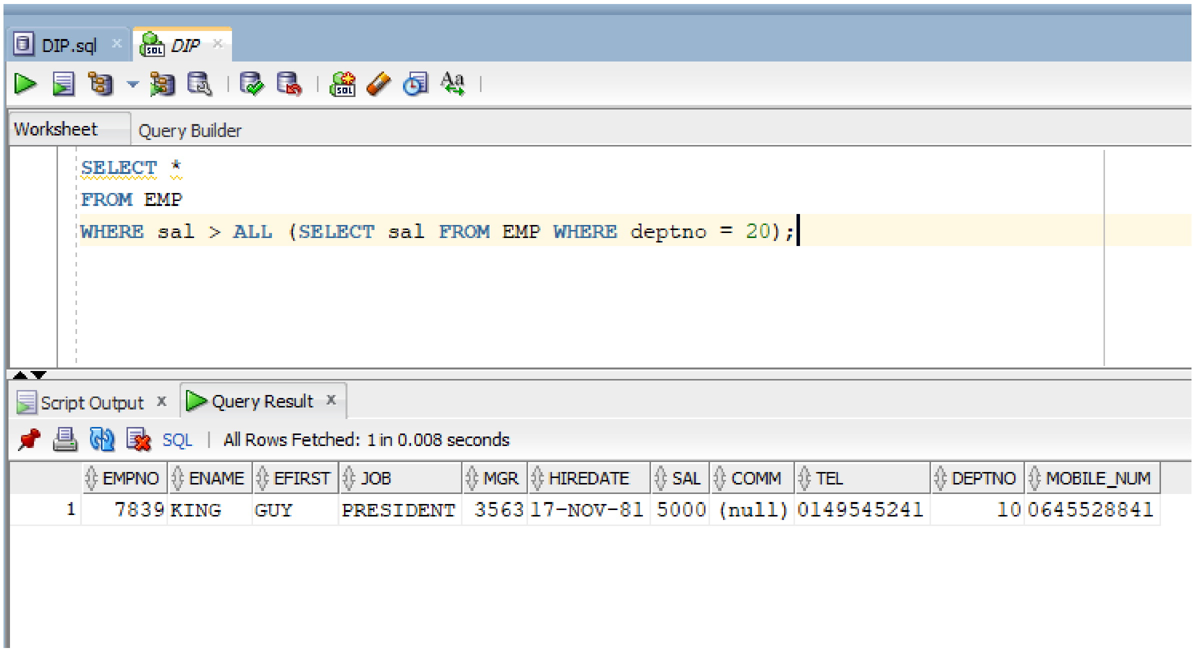
```
SELECT empno, ename, job, sal
FROM EMP
WHERE job = (SELECT job FROM EMP WHERE ename = 'JONES')
OR sal > (SELECT sal FROM EMP WHERE ename = 'FORD');
```

All Rows Fetched: 4 in 0.01 seconds

	EMPNO	ENAME	JOB	SAL
1	7566	JONES	MANAGER	2975
2	7698	BLAKE	MANAGER	2850
3	7782	CLARK	MANAGER	2450
4	7839	KING	PRESIDENT	5000

Answer 21:

For this question first we need to list all salaries then using a sub query we can check those employees that have bigger salary than department 20



```
SELECT *
FROM EMP
WHERE sal > ALL (SELECT sal FROM EMP WHERE deptno = 20);
```

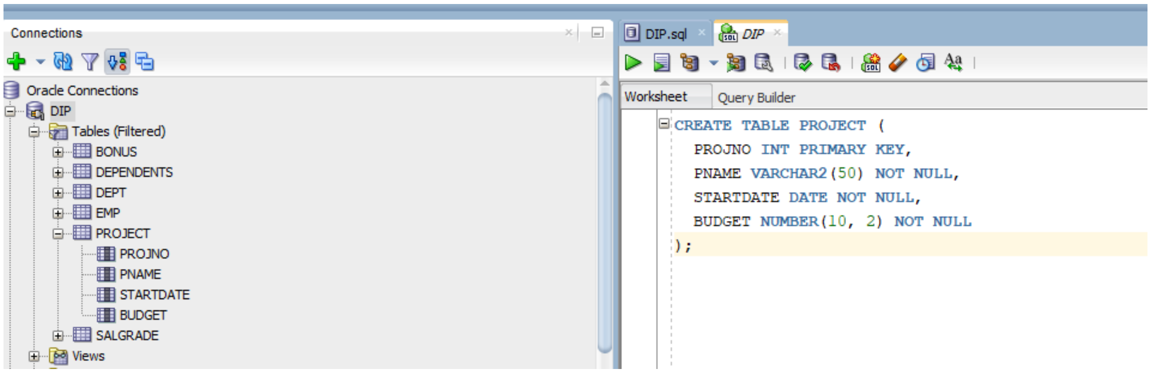
All Rows Fetched: 1 in 0.008 seconds

	EMPNO	ENAME	EFIRST	JOB	MGR	HIREDATE	SAL	COMM	TEL	DEPTNO	MOBILE_NUM
1	7839	KING	GUY	PRESIDENT	3563	17-NOV-81	5000	(null)	0149545241	10	0645528841

Section 3

Answer 1:

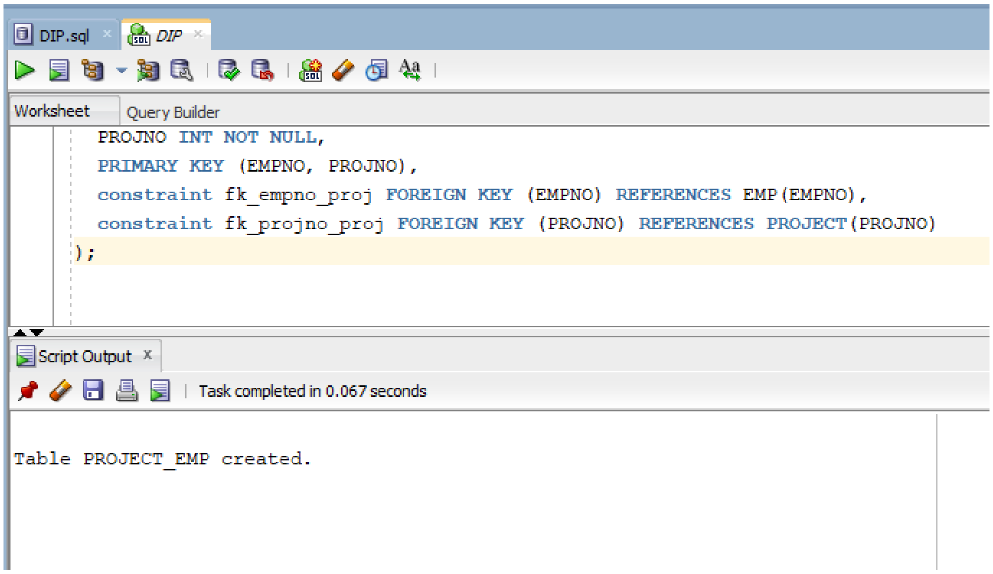
Using the following code we can create the table.



```
CREATE TABLE PROJECT (
  PROJNO INT PRIMARY KEY,
  PNAME VARCHAR2(50) NOT NULL,
  STARTDATE DATE NOT NULL,
  BUDGET NUMBER(10, 2) NOT NULL
);
```

Answer 2:

Using the following code we can create the table



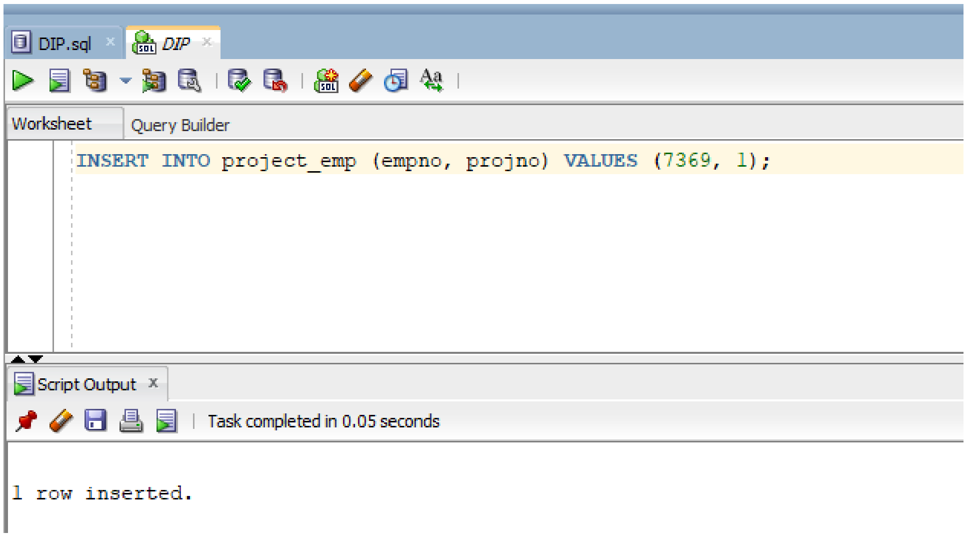
```
    PROJNO INT NOT NULL,
    PRIMARY KEY (EMPNO, PROJNO),
    constraint fk_empno_proj FOREIGN KEY (EMPNO) REFERENCES EMP(EMPNO),
    constraint fk_projno_proj FOREIGN KEY (PROJNO) REFERENCES PROJECT(PROJNO)
);
```

```
Table PROJECT_EMP created.
```

Using the following code we can insert some data.

So first of all we need to add some data to the project table because without valid id it is not possible to add data in child table we will face to this error

ORA-02291: integrity constraint (DIP.FK_PROJNO_PROJ) violated - parent key not found

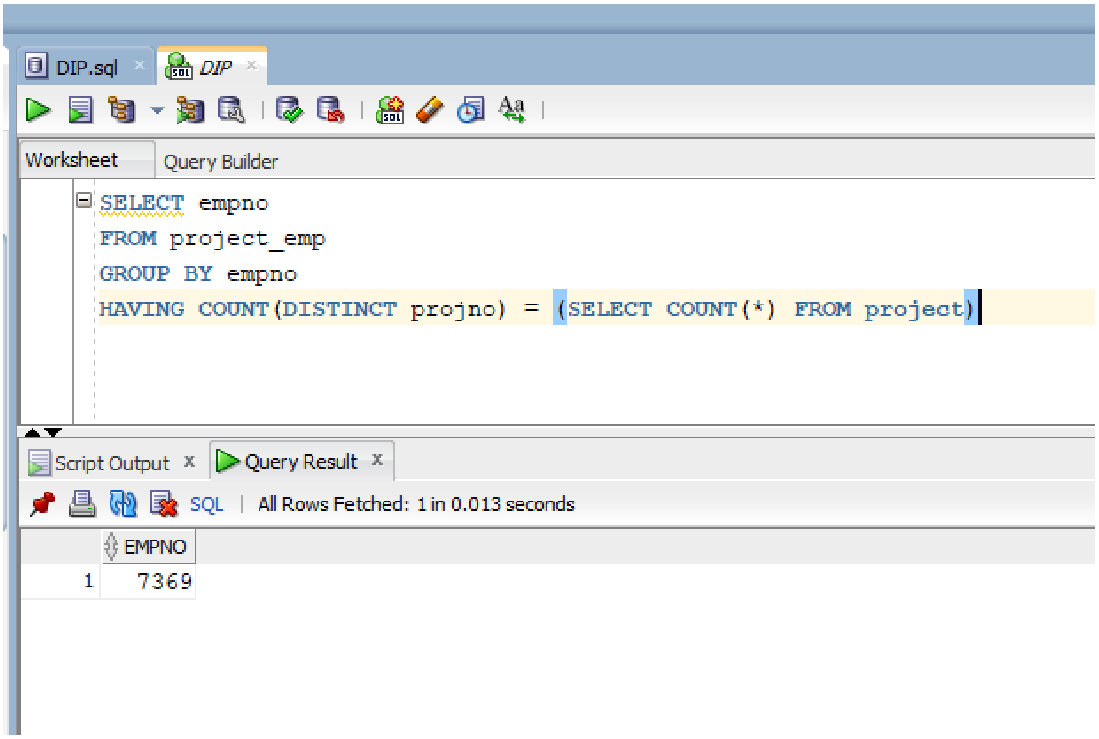


```
INSERT INTO project_emp (empno, projno) VALUES (7369, 1);
```

```
1 row inserted.
```

Answer 3:

So SIce I only Add one row of data it will only show me that data. But the query is like following.



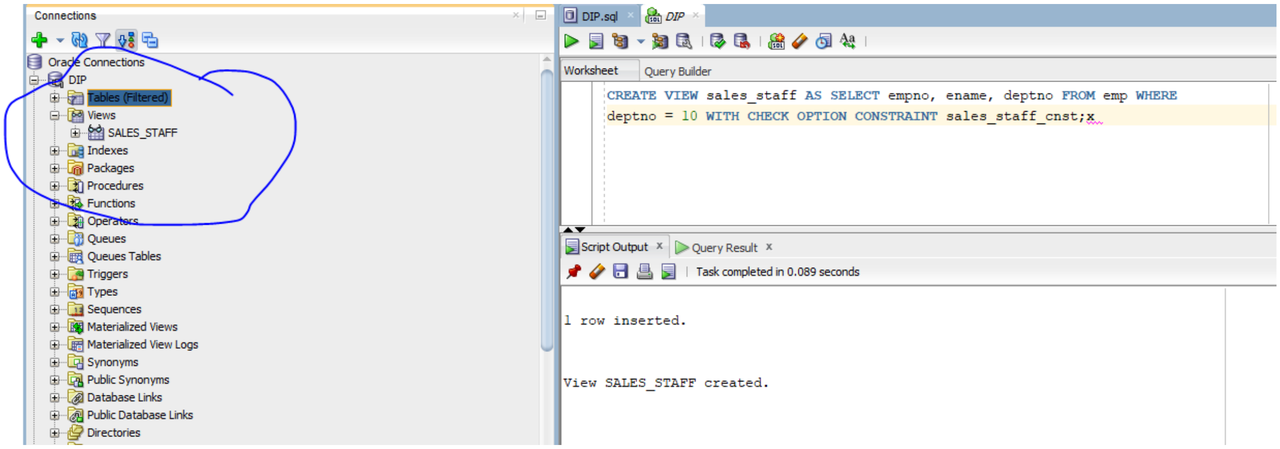
```
SELECT empno
FROM project_emp
GROUP BY empno
HAVING COUNT(DISTINCT projno) = (SELECT COUNT(*) FROM project)
```

All Rows Fetched: 1 in 0.013 seconds

	EMPNO
1	7369

Answer 4:

The CREATE VIEW statement with the WITH CHECK OPTION clause creates a view that restricts the data that can be modified or inserted into the view based on a specific condition.



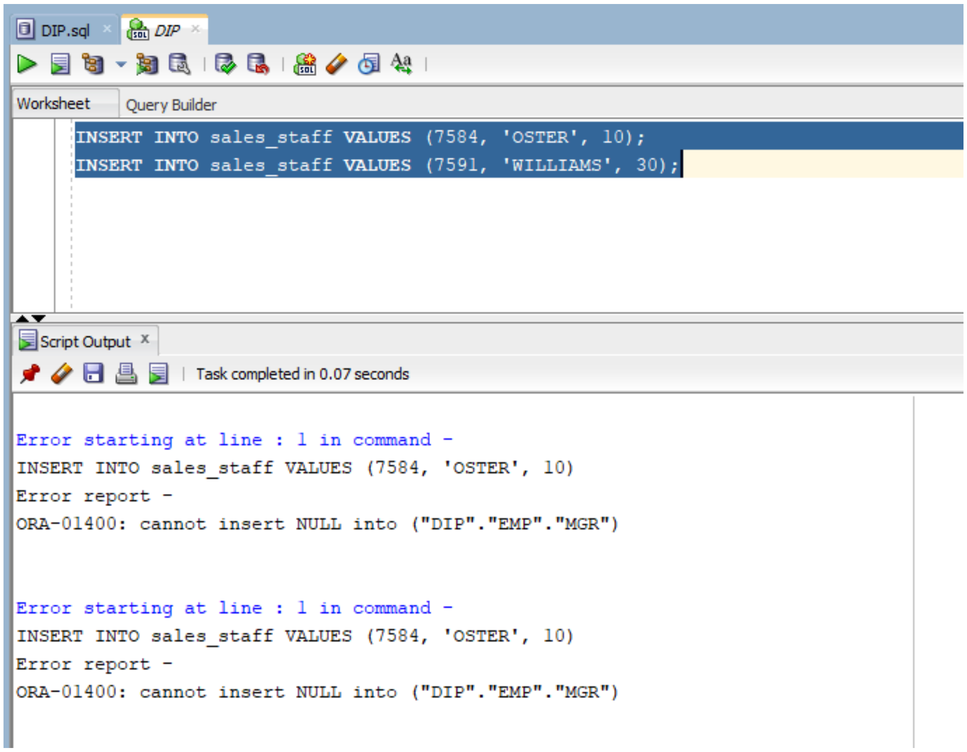
```
CREATE VIEW sales_staff AS SELECT empno, ename, deptno FROM emp WHERE
deptno = 10 WITH CHECK OPTION CONSTRAINT sales_staff_cnst;x
```

Task completed in 0.089 seconds

```
1 row inserted.


View SALES_STAFF created.
```

Answer 5:

We know that a view is just a table or a combination of some tables so we can do what ever operation that we want but we can insert data in a view because it already has the data of other tables.



```
INSERT INTO sales_staff VALUES (7584, 'OSTER', 10);
INSERT INTO sales_staff VALUES (7591, 'WILLIAMS', 30);
```

```
Error starting at line : 1 in command -
INSERT INTO sales_staff VALUES (7584, 'OSTER', 10)
Error report -
ORA-01400: cannot insert NULL into ("DIP"."EMP"."MGR")


Error starting at line : 1 in command -
INSERT INTO sales_staff VALUES (7584, 'OSTER', 10)
Error report -
ORA-01400: cannot insert NULL into ("DIP"."EMP"."MGR")
```