14|8|25

Task 4: Developing queries with DML Multi-row fu
and operators

Perform the advanced query processing an
its heuristics using the designing of opt
correlated & nested subqueries such as
Summary statistics.

Consider the Schema for

EMPLOYES (emp-no, emp-name, department,
salary, age)

Orders (emp-no Order id, Prince, qty-ordigt

Item file (item id, itemname, qty-ord, qty
hand, item rate

Queries using UNION, INTESECT, MINUS

Union: The union Operator returns all distin
rows select by two or more queries

SQL> select emp-no from employees;

Output:

SQL> select emp-no from orders;

Output:

Union ~~All~~:

SQL> select emp-no from employees
union select emp-no from orders.

Intersect:

SQL> select emp-no from employee
intersect select emp-no from order

output

Minus:

SQL> select emp-no from employees min
select emp-no from orders

output:

Practice questions:

. find the emp-no of employees whose name starts with 's' & end with 'H'.

. Find the names of the employees whose age is btw 20 & 40

. Display all the names of the employee beginning with 'r'.

. Display the sorted list of employees Name.

Syntax:

select column(s) from <Table names> where (conditions s) (order by coloumn name> [ase|] desc];

SQL > select empno _ ename, salary from employee order by salary;

output:

SQL > select emp no . emp-Name salary from employee order by salary desc;

output:

SQL> select salary + comm from emp-master. salary + comm.

output:

SQL > select 12* (salary + comm) annual-netsal
from emp-master;
output:

subqueries:

SQL> select * from employees
SQL> Insert into employees select * from employees
where emp-id (select emp-id from employees);

SQL> update employees set salary = salary * 10 where
department in (select department from employee where
department = ' sales' ';

Delete from employee where department in
(select department from employee wher =' sales');

IN:
Query: SELECT * from employee WHERE department IN
('sales', 'marketing');
output:

Not In:
Query: SELECT * from employee where department NOT
IN ('sales', 'marketing'):
Output:

Exist: select * from employee where EXISTS (select *
from order where emp-no = (link) ename);
output:

ALL:

Query: SELECT * from employee where salary > All
(select salary from employee where department = 'sales
output:

Any:

SELECT * from order_master where order_no = (select
order_no from order where order_no = 'sales')

output:

SQL > select * from order_master where order_no
= (select order_no for orders where order_no = '0001')
output:

SQL > select * from order_master where order_no = any
(select order_n from order_detail);
output:

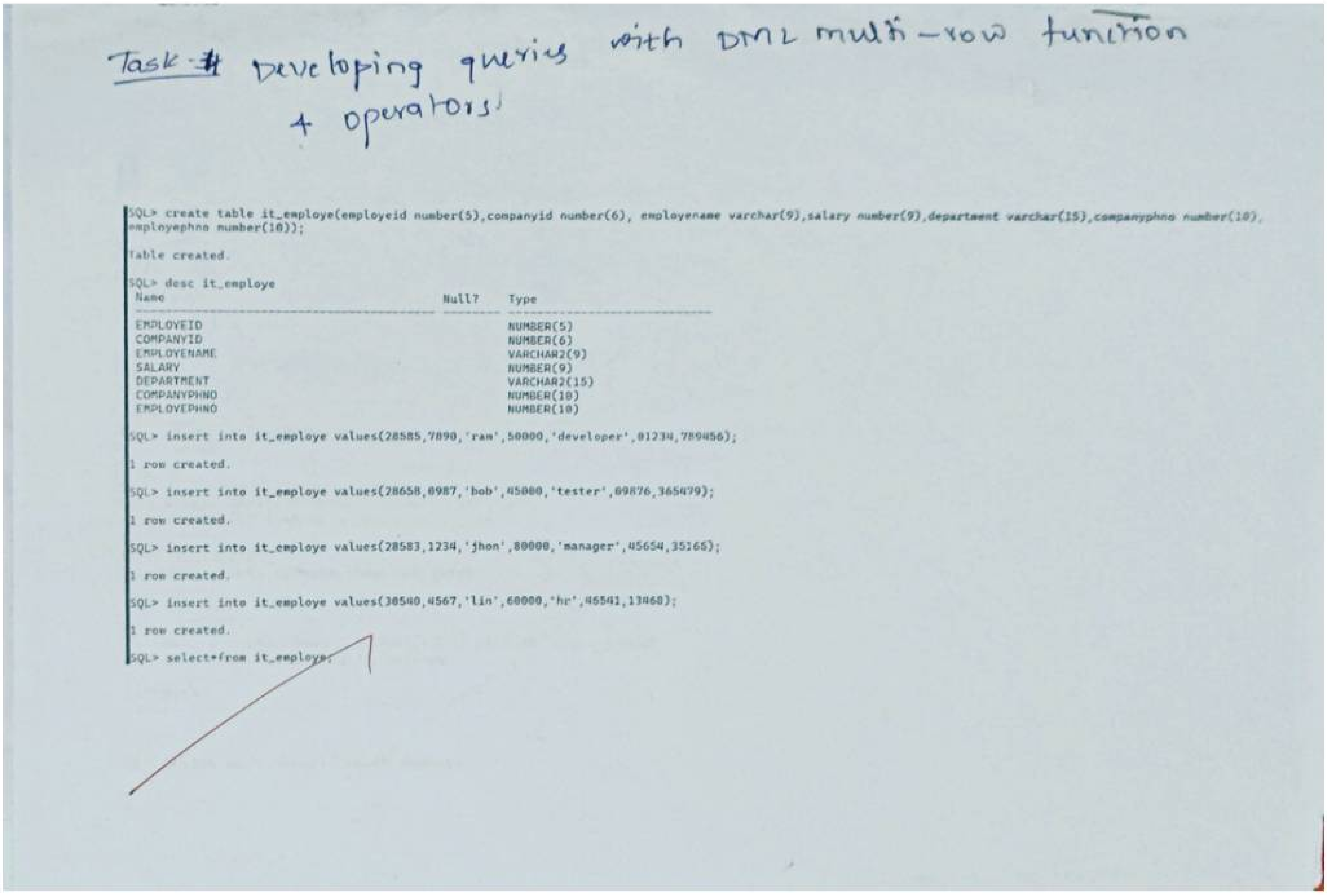
SQL > select * from order master where order_no in
(select order_no from order_detail)..
output:

SQL > select * from order_detail where qty-ord
<= all (select qty-ord [illegible] where it
output:

VELTECH
EX.No. 8
PERFORMANCE (5) 4
RESULT AND ANALYSIS (5) 5
VIVA VOCE (5)
RECORD (4)
TOTAL (15)
SIGN WITH DATE
21/8/25

21/8/25

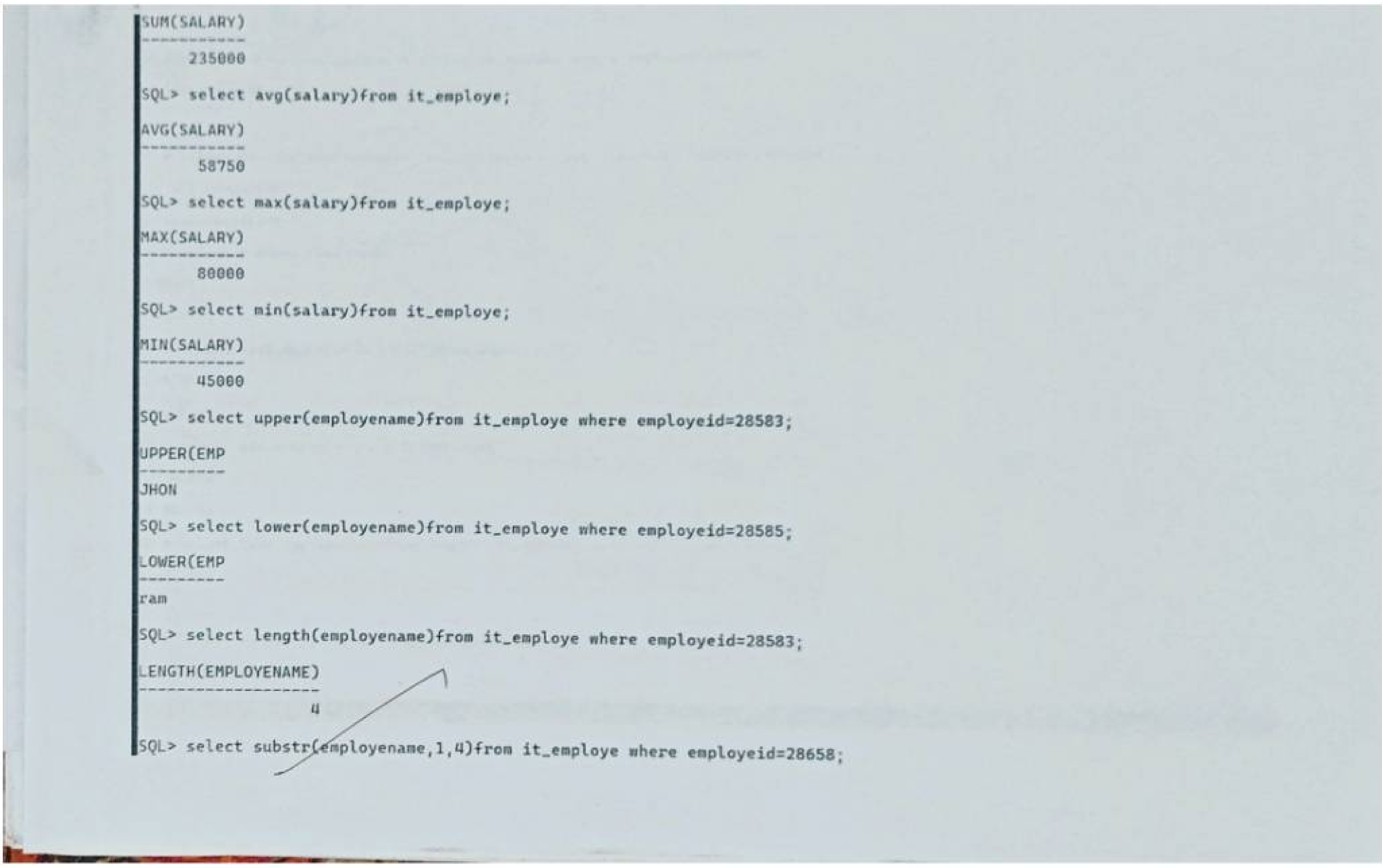
Result: Thus, the developing querying with DML multi-row function has done successfully.



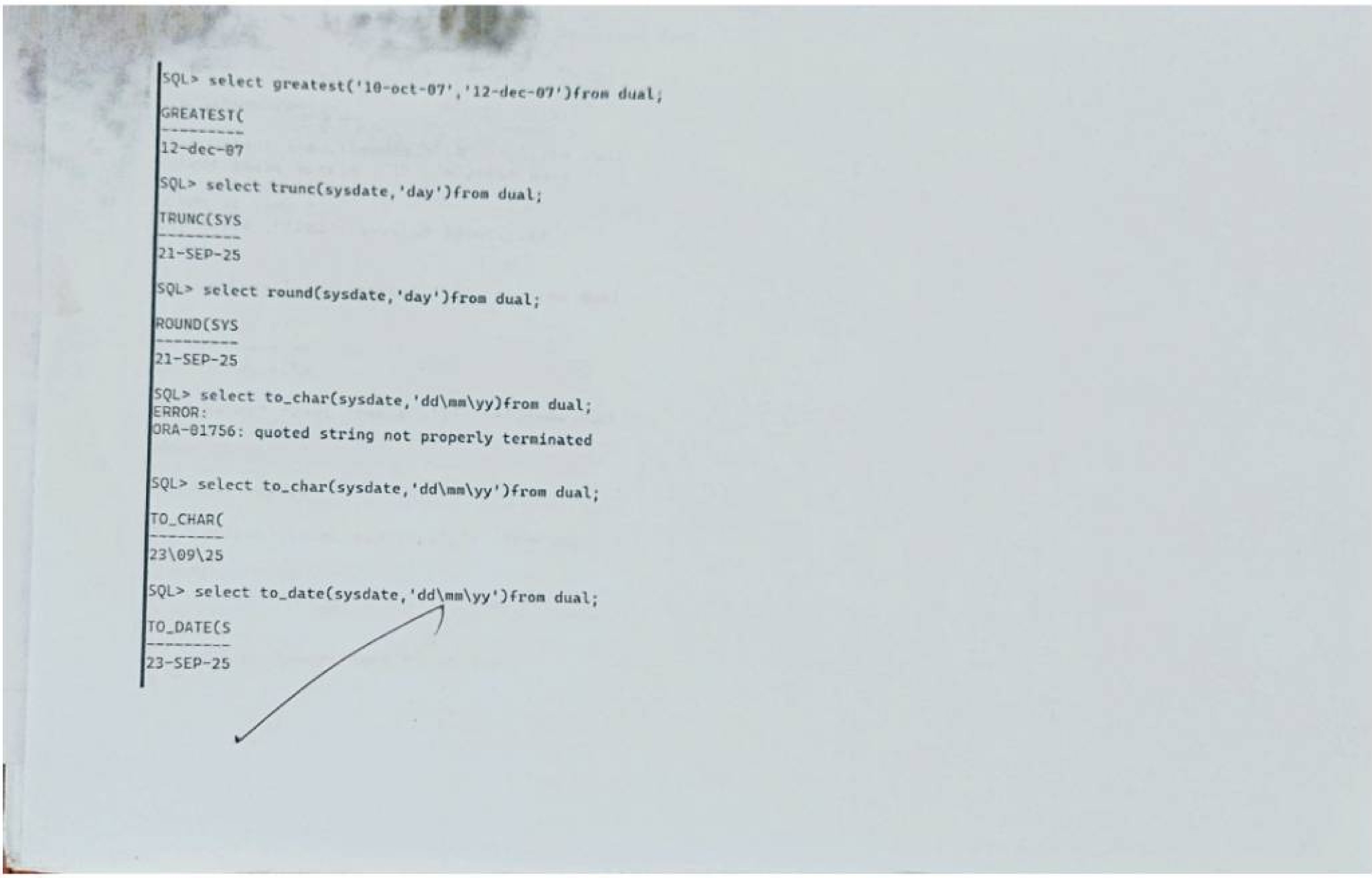
Task-# Developing queries with DML multi-row function + operators.

```
SQL> create table it_employe(employeid number(5),companyid number(6), employename varchar(9),salary number(9),department varchar(15),companyphno number(10),
employephno number(10));

Table created.

SQL> desc it_employe
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 EMPLOYEID                                          NUMBER(5)
 COMPANYID                                          NUMBER(6)
 EMPLOYENAME                                        VARCHAR2(9)
 SALARY                                             NUMBER(9)
 DEPARTMENT                                         VARCHAR2(15)
 COMPANYPHNO                                        NUMBER(10)
 EMPLOYEPHNO                                        NUMBER(10)

SQL> insert into it_employe values(28585,7890,'ram',50000,'developer',01234,789456);

1 row created.

SQL> insert into it_employe values(28658,0987,'bob',45000,'tester',09876,365479);

1 row created.

SQL> insert into it_employe values(28583,1234,'jhon',80000,'manager',45654,35165);

1 row created.

SQL> insert into it_employe values(30540,4567,'lin',60000,'hr',45541,13468);

1 row created.

SQL> select*from it_employe;
```

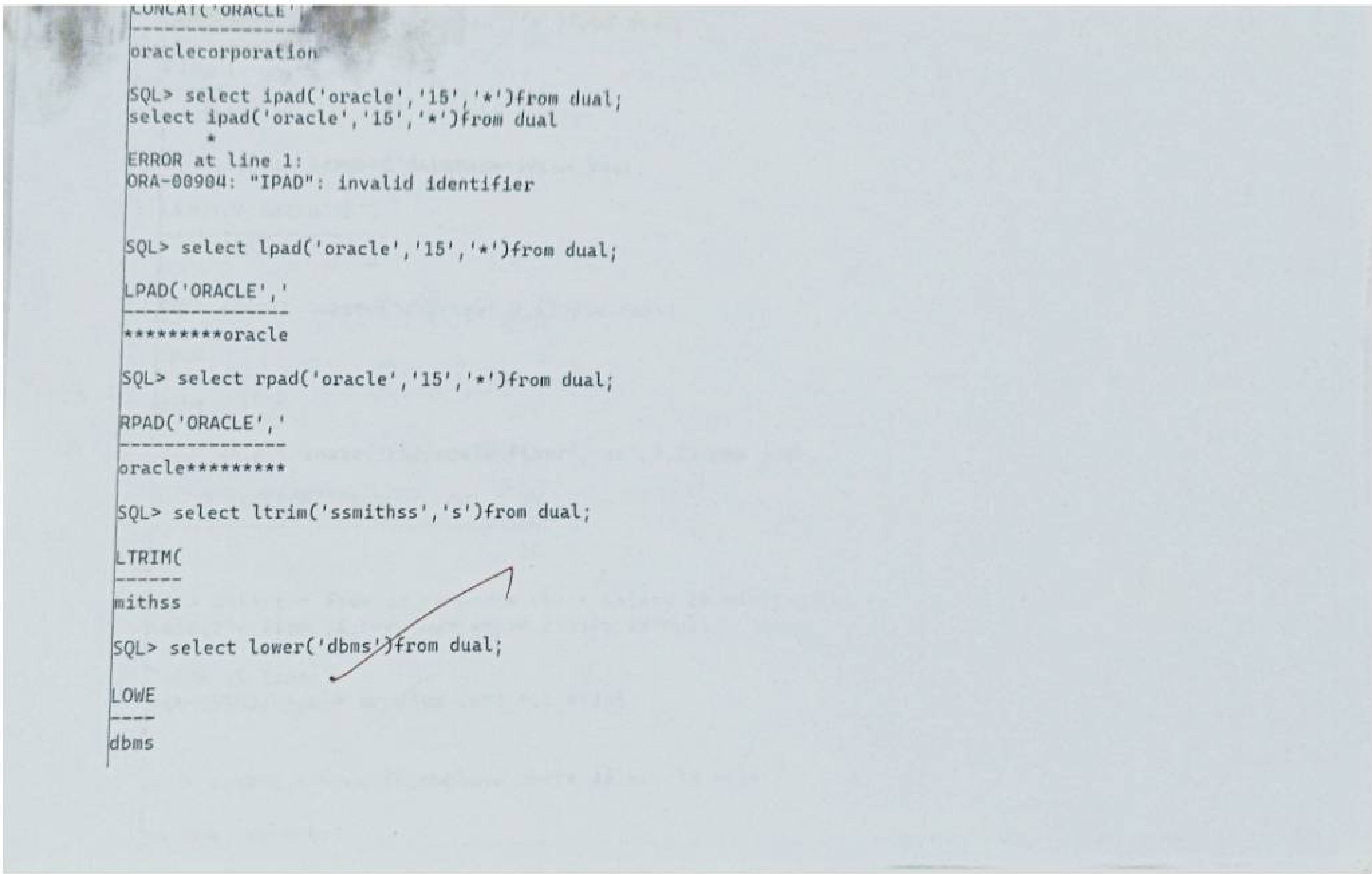
```
EMPLOYEID  COMPANYID EMPLOYENA      SALARY DEPARTMENT      COMPANYPHNO
---------- ---------- ---------- ---------- --------------- -----------
MPLOYEPHNO
-----------
     28585       7890 ram             50000 developer              1234
    789456

     28658        987 bob             45000 tester                 9876
    365479

     28583       1234 jhon            80000 manager               45654
     35165

EMPLOYEID  COMPANYID EMPLOYENA      SALARY DEPARTMENT      COMPANYPHNO
---------- ---------- ---------- ---------- --------------- -----------
MPLOYEPHNO
-----------
     30540       4567 lin             60000 hr                    46541
     13468

QL> select count(*) from it_company_employees;
elect count(*) from it_company_employees
                    *
RROR at line 1:
RA-00942: table or view does not exist


QL>   COUNT(*)
P2-0042: unknown command "COUNT(*)" - rest of line ignored.
QL> select count(*) from it_employe;

  COUNT(*)
----------
         4

QL> select sum(salary)from it_employe;
```



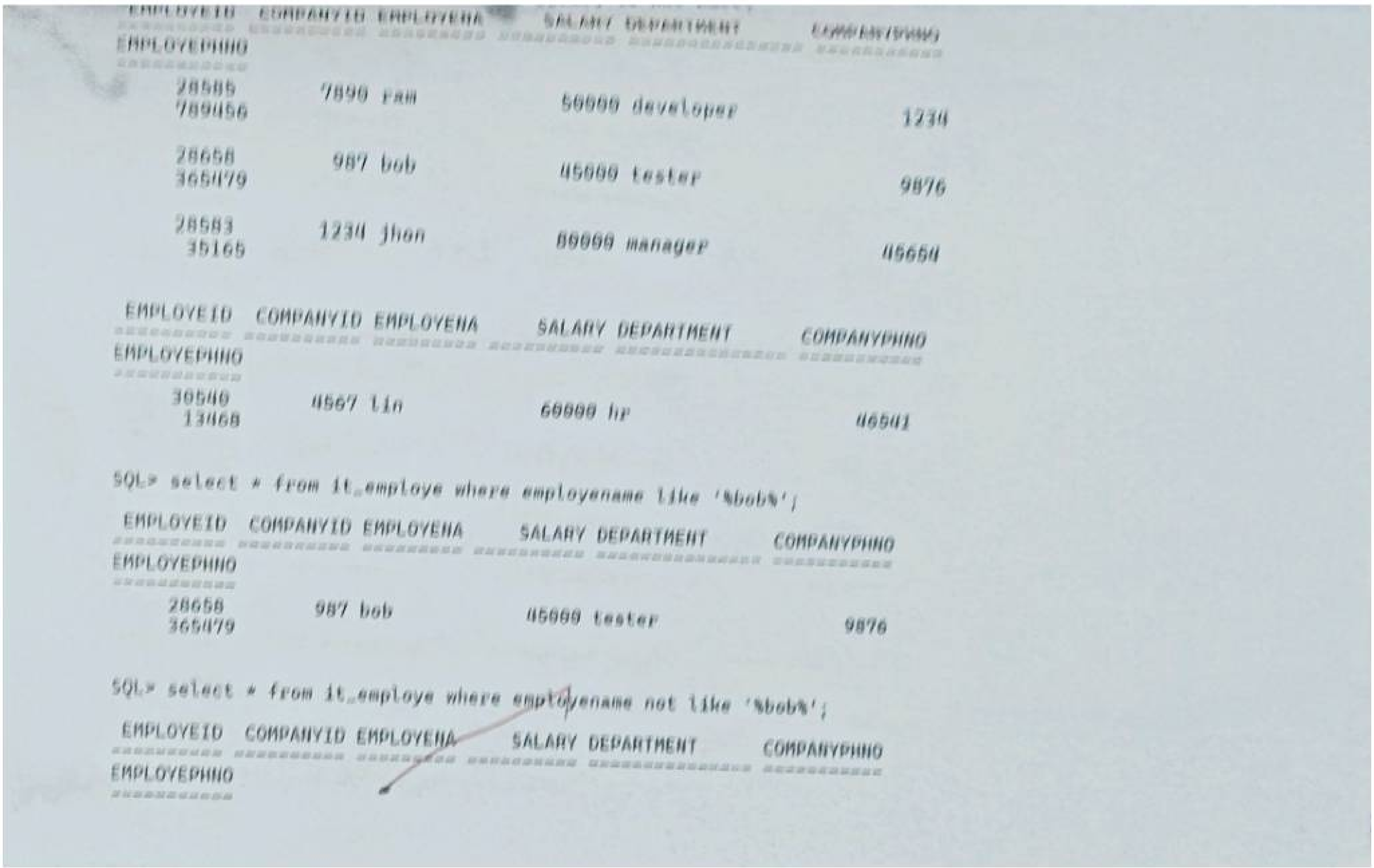
```
SUM(SALARY)
-----------
     235000

SQL> select avg(salary)from it_employe;

AVG(SALARY)
-----------
      58750

SQL> select max(salary)from it_employe;

MAX(SALARY)
-----------
      80000

SQL> select min(salary)from it_employe;

MIN(SALARY)
-----------
      45000

SQL> select upper(employename)from it_employe where employeid=28583;

UPPER(EMP
---------
JHON

SQL> select lower(employename)from it_employe where employeid=28585;

LOWER(EMP
---------
ram

SQL> select length(employename)from it_employe where employeid=28583;

LENGTH(EMPLOYENAME)
-------------------
                  4

SQL> select substr(employename,1,4)from it_employe where employeid=28658;
```



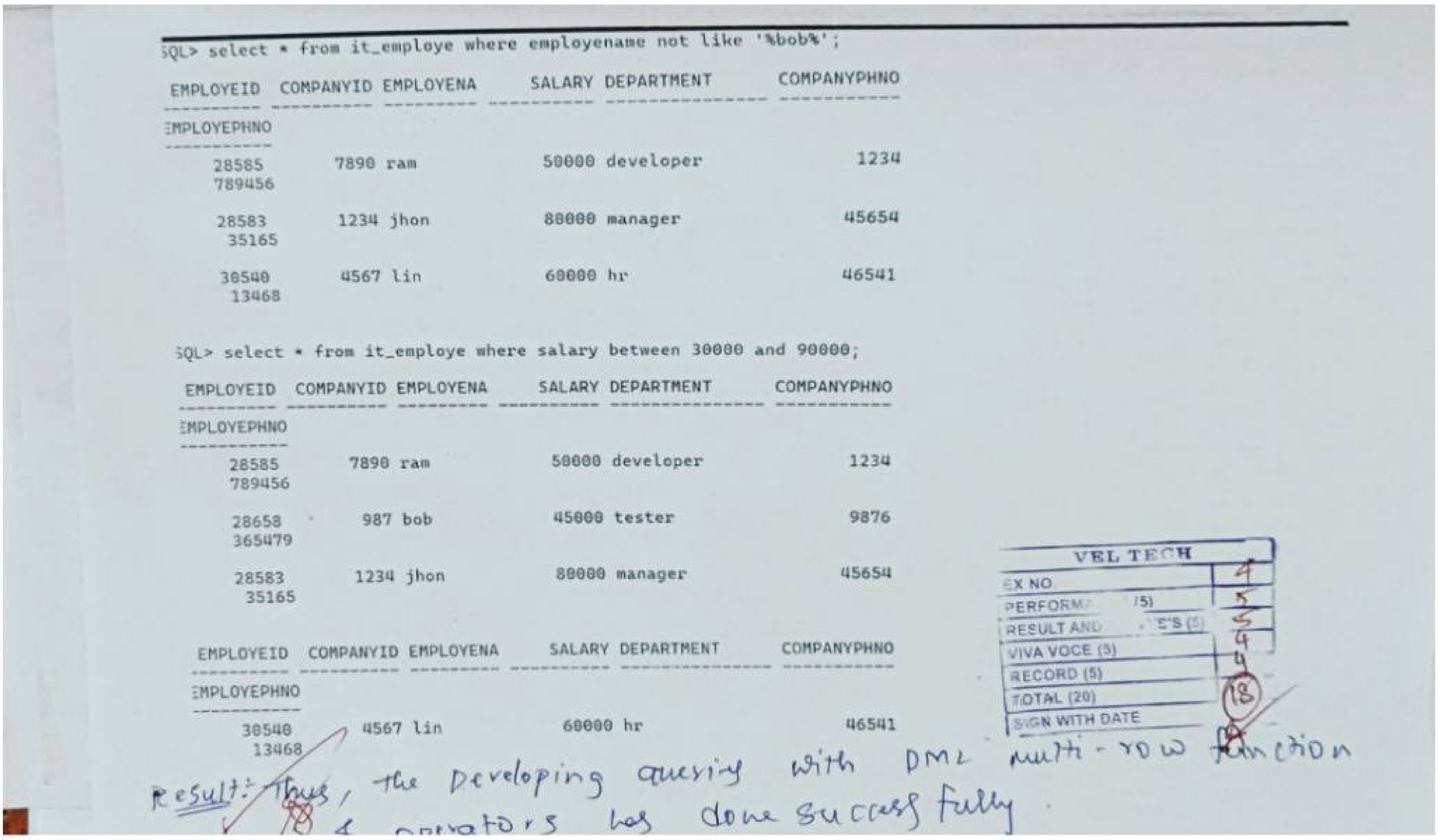
```
SQL> select greatest('10-oct-07','12-dec-07')from dual;

GREATEST(
---------
12-dec-07

SQL> select trunc(sysdate,'day')from dual;

TRUNC(SYS
---------
21-SEP-25

SQL> select round(sysdate,'day')from dual;

ROUND(SYS
---------
21-SEP-25

SQL> select to_char(sysdate,'dd\mm\yy)from dual;
ERROR:
ORA-01756: quoted string not properly terminated


SQL> select to_char(sysdate,'dd\mm\yy')from dual;

TO_CHAR(
--------
23\09\25

SQL> select to_date(sysdate,'dd\mm\yy')from dual;

TO_DATE(S
---------
23-SEP-25
```



```
CONCAT('ORACLE'
----------------
oraclecorporation

SQL> select ipad('oracle','15','*')from dual;
select ipad('oracle','15','*')from dual
       *
ERROR at line 1:
ORA-00904: "IPAD": invalid identifier

SQL> select lpad('oracle','15','*')from dual;

LPAD('ORACLE','
---------------
*********oracle

SQL> select rpad('oracle','15','*')from dual;

RPAD('ORACLE','
---------------
oracle*********

SQL> select ltrim('ssmithss','s')from dual;

LTRIM(
------
mithss

SQL> select lower('dbms')from dual;

LOWE
----
dbms
```



```
 EMPLOYEID  COMPANYID EMPLOYENA     SALARY DEPARTMENT       COMPANYPHNO
========== ========== ========= ========== =============== ===========
EMPLOYEPHNO
===========
     28585       7890 ram            50000 developer              1234
    789456

     28658        987 bob            45000 tester                 9876
    365479

     28583       1234 jhon           80000 manager               45654
     35165

 EMPLOYEID  COMPANYID EMPLOYENA     SALARY DEPARTMENT       COMPANYPHNO
========== ========== ========= ========== =============== ===========
EMPLOYEPHNO
===========
     30540       4567 lin            60000 hr                    46541
     13468

SQL> select * from it_employe where employename like '%bob%';

 EMPLOYEID  COMPANYID EMPLOYENA     SALARY DEPARTMENT       COMPANYPHNO
========== ========== ========= ========== =============== ===========
EMPLOYEPHNO
===========
     28658        987 bob            45000 tester                 9876
    365479

SQL> select * from it_employe where employename not like '%bob%';

 EMPLOYEID  COMPANYID EMPLOYENA     SALARY DEPARTMENT       COMPANYPHNO
========== ========== ========= ========== =============== ===========
EMPLOYEPHNO
===========
```



```
SQL> select * from it_employe where employename not like '%bob%';

 EMPLOYEID  COMPANYID EMPLOYENA      SALARY DEPARTMENT        COMPANYPHNO
---------- ---------- --------- ---------- --------------- -----------
EMPLOYEPHNO
-----------
     28585       7890 ram             50000 developer              1234
    789456

     28583       1234 jhon            80000 manager               45654
     35165

     30540       4567 lin             60000 hr                    46541
     13468

SQL> select * from it_employe where salary between 30000 and 90000;

 EMPLOYEID  COMPANYID EMPLOYENA      SALARY DEPARTMENT        COMPANYPHNO
---------- ---------- --------- ---------- --------------- -----------
EMPLOYEPHNO
-----------
     28585       7890 ram             50000 developer              1234
    789456

     28658        987 bob             45000 tester                 9876
    365479

     28583       1234 jhon            80000 manager               45654
     35165

 EMPLOYEID  COMPANYID EMPLOYENA      SALARY DEPARTMENT        COMPANYPHNO
---------- ---------- --------- ---------- --------------- -----------
EMPLOYEPHNO
-----------
     30540       4567 lin             60000 hr                    46541
     13468
```

VEL TECH	
EX NO.	4
PERFORM... (5)	5
RESULT AND ...'S (5)	5
VIVA VOCE (5)	4
RECORD (5)	4
TOTAL (20)	18
SIGN WITH DATE	

Result:- Thus, the Developing quesiry with DML multi-row function & operators has done success fully.