

Aayush Shah

Batch D1

19BCE245

27 February 2021

Practical 4

Do as directed.

- 1) List the name of employees whose name is having five characters in total.

```
SQL> ed
Wrote file afiedt.buf

  1 * select first_name from employees where length(first_name) = 5
SQL> /

FIRST_NAME
-----
Neena
Bruce
David
Valli
Karen

SQL> spool off;
```

- 2) List the names of employees whose name starts with 'K' and having five characters in total.

```
SQL> ed
Wrote file afiedt.buf

  1 * select first_name from employees where first_name like 'K____'
SQL> /

FIRST_NAME
-----
Karen

SQL> spool off;
```

- 3) List the names of employees whose name is of five character and fourth character is either 'i' or 'e'.

```
SQL> ed
Wrote file afiedt.buf

  1 * select first_name from employees where first_name like '___i_' or first_name like
'___e_'

SQL> /

FIRST_NAME
-----
David
Karen

SQL> spool off;
```

- 4) List the names of employee whose name is of five character and which starts with 'D' and ends with 'd'.

```
SQL> ed
Wrote file afiedt.buf

  1 * select first_name from employees where first_name like 'D___d'

SQL> /

FIRST_NAME
-----
David

SQL> spool off;
```

- 5) List the employee whose Salary in four digits ending with Zero.

```
SQL> ed
Wrote file afiedt.buf

  1 * select first_name,salary from employees where to_char(salary) like '___0'

SQL> /

FIRST_NAME      SALARY
-----
Alexander      9000
Bruce           6000
David           4800
Valli           4800
Karen           2500
William         8300
```

6 rows selected.

SQL> spool off;

6) List the employee whose names having a character set 'll' together.

SQL> ed

Wrote file afiedt.buf

```
1 * select first_name from employees where instr(first_name,'ll')>0
SQL> /
```

FIRST_NAME

Valli

William

SQL> spool off;

7) List the details of employees whose Salaries is more than the salary of 'William'.

SQL> ed

Wrote file afiedt.buf

```
1 * select * from employees where salary > (select salary from employees where
first_name = 'William')
```

SQL> /

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL
PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY
COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	
100	Steven	King	515.123.4567
PRESIDENT		24000	17-06-00
		103	90
101	Neena	Kochhar	515.123.4568
18-06-12	VICE PRESIDENT	17000	0 100 90
102	Lex	De Haan	123.515.4569
19-06-17	VICE PRESIDENT	17000	100 90
103	Alexander	Hunold	590.423.4567
20-05-16	IT PROG	9000	102 60
114	Den	Raphaely	515.127.4561
01-09-90	SALES CLERK	11000	0 100 30

SQL> spool off;

8) List the employees whose jobs are same as of 'David'.

SQL> ;

```
1 * select first_name,job_id from employees where job_id = (select job_id from
employees where first_name = 'David') and first_name != 'David'
```

SQL> ed

Wrote file afiedt.buf

```
1 * select first_name,job_id from employees where job_id = (select job_id from
employees where first_name = 'David') and first_name != 'David'
```

SQL> /

FIRST_NAME	JOB_ID
Alexander	IT PROG

SQL> spool off;

9) /SKIP/ List the Employees who are senior to their own Managers.

[skipped]

```
SQL> select emp2.first_name from employees emp1,employees emp2 where
emp1.employee_id=emp2.manager_id and emp1.hire_date>emp2.hire_date;
```

FIRST_NAME

Steven

Alexander

Bruce

David

Valli Den Karen

10) List the employees, whose jobs are same as 'David' or Salary is higher than 'Alexander'.

SQL> ed

Wrote file afiedt.buf

```
1 * select first_name,job_id,salary from employees where job_id = (select job_id from
employees where first_name = 'David') or salary > (select salary from employees where
first_name = 'Alexander')
```

SQL> /

FIRST_NAME	JOB_ID	SALARY
Steven	PRESIDENT	24000
Neena	VICE PRESIDENT	17000
Lex	VICE PRESIDENT	17000

Alexander	IT PROG	9000
David	IT PROG	4800
Den	SALES CLERK	11000

6 rows selected.

SQL> spool off;

11) List all jobs of deptno 90 that are not found in deptno 60.

SQL> ed

Wrote file afiedt.buf

```
1 * select distinct job_id from employees where department_id = 90 and job_id NOT IN
(select job_id from employees where department_id = 60)
```

SQL> /

JOB_ID

VICE PRESIDENT

PRESIDENT

SQL> spool off;

12) List the employees whose salary is more than 10000 after giving 20% increment.

SQL> ed

Wrote file afiedt.buf

```
1 * select first_name,salary,salary * 1.2 from employees where salary * 1.2 > 10000
```

SQL> /

FIRST_NAME	SALARY	SALARY*1.2
Steven	24000	28800
Neena	17000	20400
Lex	17000	20400
Alexander	9000	10800
Den	11000	13200

SQL> spool off;

13) List the employees who are not working as 'CLERK'.

SQL> ed

Wrote file afiedt.buf

```
1 * select first_name,job_id from employees where instr(job_id,'CLERK') = 0
SQL> /
```

FIRST_NAME	JOB_ID
Steven	PRESIDENT
Neena	VICE PRESIDENT
Lex	VICE PRESIDENT
Alexander	IT PROG
Bruce	FINANCIAL MANAGER
David	IT PROG
Valli	FINANCE ACCOUNTANT
William	ACCOUNTANT

8 rows selected.

```
SQL> spool off;
```

14) [SKIP] List the employees whose salary is greater than his managers' salary.

```
SQL> select emp2.first_name from employees emp1,employees emp2 where
emp1.employee_id=emp2.manager_id and emp1.salary>emp2.salary;
```

FIRST_NAME

Neena

Lex

Alexander Bruce David Valli
Den Karen

15) List the employees whose annual salary ranging from 70,000 and 90,000.

```
SQL> ed
```

Wrote file afiedt.buf

```
1 * select first_name,salary,salary * 12 from employees where salary * 12 between
70000 and 90000
```

```
SQL> /
```

FIRST_NAME	SALARY	SALARY*12
Bruce	6000	72000

```
SQL> spool off;
```

- 16) Display the employee_id, first_name, salary, job_id, deptno of all employees working as IT PROG with annual salary > 80,000, print the information in asc order of Deptno.

SQL> ed

Wrote file afiedt.buf

```
1 * select employee_id,first_name,salary,job_id,department_id from employees where
job_id = 'IT PROG' and salary * 12 > 80000 order by department_id ASC
```

SQL> /

EMPLOYEE_ID	FIRST_NAME	SALARY	JOB_ID	DEPARTMENT_ID
-------------	------------	--------	--------	---------------

103	Alexander	9000	IT PROG	60
-----	-----------	------	---------	----

SQL> spool off;

- 17) List employee_id, first_name, salary, job_id, deptno, experience and Annual Salary of employees working for Dept 90 or 60.

SQL> ed

Wrote file afiedt.buf

```
1 select employee_id,first_name,salary,job_id,department_id,
2 ROUND((SYSDATE - TO_DATE(hire_date))/365.25, 1) as EXPERIENCE,
3 salary * 12 as ANNUAL_SALARY
4 * from employees where department_id in (60,90)
```

SQL> /

EMPLOYEE_ID	FIRST_NAME	SALARY	JOB_ID	DEPARTMENT_ID	EXPERIENCE	ANNUAL_SALARY
-------------	------------	--------	--------	---------------	------------	---------------

100	Steven	24000	PRESIDENT	90	20.7	288000
101	Neena	17000	VICE PRESIDENT	90	8.7	204000
102	Lex	17000	VICE PRESIDENT	90	3.7	204000
103	Alexander	9000	IT PROG	60	4.8	108000
104	Bruce	6000	FINANCIAL MANAGER	60	6.8	72000
105	David	4800	IT PROG	60	19.4	57600

106 Valli 57600	4800 FINANCE ACCOUNTANT	60	19.4
--------------------	-------------------------	----	------

7 rows selected.

SQL> spool off;

18) [SKIP] Find total number of orders placed by each salesman.

SQL> select salesmanNo,count(*) No_Of_Order from sales_order GROUP BY salesmanNo;

SALESM	NO_OF_ORDER	-----
S01	2	
S02	2	
S03	1	
S04	1	

19) How many customers have no balance due?

SQL> ed

Wrote file afiedt.buf

1 * select count(*) from client where baldue=0 group by baldue
SQL> /

COUNT(*)	-----
2	

SQL> spool off;

20) Which product is most expensive? (highest sell price).

SQL> ed

Wrote file afiedt.buf

1 * select productno, description,sellprice from (select * from product order by sellprice desc) where rownum = 1

SQL> /

PRODUC	DESCRIPTION	SELLPRICE

P03453	Monitors	12000

SQL> spool off;

21) Which product has highest profit per unit? (Product no with highest (sell_price-cost_price)/cost_price))

SQL> ed

Wrote file afiedt.buf

```
1 * select productno,(sellprice-costprice)/costprice Highest_Profit_per_unit from
product where ((sellprice- costprice)/costprice)=(select max((sellprice-costprice)/
costprice) from product)
```

SQL> /

PRODUC HIGHEST_PROFIT_PER_UNIT

```
-----
PO6734          1.1
```

SQL> spool off;

22) Find product name of product has highest profit per unit.

SQL> ed

Wrote file afiedt.buf

```
1 * select productno,description,sellprice,(sellprice- costprice)/costprice
Highest_Profit_per_unit from product where ((sellprice-costprice)/costprice)=(select
max((sellprice- costprice)/costprice) from product)
```

SQL> /

PRODUC DESCRIPTION SELLPRICE HIGHEST_PROFIT_PER_UNIT

```
-----
PO6734 Mouse          1050          1.1
```

SQL> spool off;

23) [SKIP] Find total amount of order placed for each product.(sum of ProductRate*QtyOrdered)

SQL> ed

Wrote file afiedt.buf

```
1 * select s1.salesmanno,sum(s2.productrate*s2.qtyordered) Total_Amount from
sales_order s1 join sales_order_details s2 on s1.orderno=s2.orderno group by
s1.salesmanno
```

SQL> /

SALESM TOTAL_AMOUNT

```
-----
S01      29400
S02      5250
S03      9450
```

```
SQL> spool off;
```

24) [SKIP] Display year-wise amount of order placed by each salesman.

```
SQL> select extract(year from s1.delaydate)
Order_Year,sum(s2.productrate*s2.qtyordered) Total_Amount from sales_order s1 join
sales_order_details s2 on s1.orderno=s2.orderno group by extract(year from
s1.delaydate);
```

```
ORDER_YEAR TOTAL_AMOUNT
```

```
-----
```

```
2017 14700
```

```
2016 29400
```

25) [SKIP] Identify salesman who placed any order of more than Rs.10000.

```
select s1.salesmanno,s2.orderno,s2.productrate*s2.qtyordered Order_Amount from
sales_order s1 join sales_order_details s2 on s1.orderno=s2.orderno where
s2.productrate*qtyordered>10000;
```

```
SALESM ORDERN ORDER_AMOUNT
```

```
-----
```

```
S01 019001 10500
```

```
S01 019001 16800
```