

Name: Barbara Jennina B. Perez

1) Select the employee in department 30.

```
SELECT * FROM employees WHERE department_id = 30;
```

2) List the names, numbers and department of all clerks.

```
SELECT first_name, last_name, phone_number, department_id FROM employees  
WHERE job_id IN ('PU_CLERK','SH_CLERK','ST_CLERK');
```

```
SELECT first_name, last_name, phone_number, department_id FROM employees  
WHERE job_id='PU_CLERK' or job_id='SH_CLERK' or job_id='ST_CLERK';
```

```
SELECT first_name, last_name, phone_number, department_id FROM employees  
WHERE job_id LIKE '%CLERK';
```

```
SELECT first_name, last_name, phone_number, department_id FROM employees  
WHERE job_id LIKE '%K';
```

3) Find the department numbers and the name of employee of all dept with Deptno greater or equal to 20.

```
SELECT department_id, first_name, last_name FROM employees WHERE department_id >= 20;
```

4) Find the employees whose commission is greater than their salary.

```
SELECT * FROM employees WHERE (commission_pct*salary) > salary;
```

5) Find the employees whose commission is greater than 60 percent of their salary.

```
SELECT * FROM employees WHERE (commission_pct*salary) > (salary*0.6);
```

6) Find the employee whose commission is greater than 50 percent of their salary.

```
SELECT * FROM employees WHERE (commission_pct*salary) > (salary*0.5);
```

7) List the name, job and salary of all employees in dept 20 who earn more than 2000.

```
SELECT first_name, last_name, job_id, salary FROM employees WHERE department_id=20 and  
salary>2000;
```

8) Find all salesmen in dept 30 whose salary is greater than or equal to Rs. 1500.

```
SELECT * FROM employees WHERE job_id='SA_REP' AND department_id=30 AND salary>=1500;
```

```
SELECT * FROM employees WHERE job_id LIKE 'SA%' AND department_id=30 AND salary>=  
1500;
```

(not sure if SA_MAN is considered a salesman)

9) Find all the employees whose job is either a president or manager.

```
SELECT * FROM employees
WHERE job_id='AD_PRES' or job_id LIKE '%MGR' or job_id LIKE '%MAN';
```

10) Find all managers who are not in dept 30.

```
SELECT * FROM employees
WHERE (job_id LIKE '%MGR' or job_id LIKE '%MAN') AND department_id <> 30;
```

11) Find the details of all managers and clerks in dept 10.

```
SELECT * FROM employees
WHERE (job_id LIKE '%MGR' or job_id LIKE '%MAN' or job_id LIKE '%CLERK') and
department_id=10;
```

12) Find the details of all manager (in any dept) and all clerks in dept 10

```
SELECT * FROM employees
WHERE (job_id LIKE '%MGR' or job_id LIKE '%MAN') OR (job_id LIKE '%CLERK' AND
department_id=10);
```

13) Find the details of all managers in dept 10 and all clerks in dept 20.

```
SELECT * FROM employees
WHERE ((job_id LIKE '%MGR' OR job_id LIKE '%MAN') AND department_id=10)
OR (job_id LIKE '%CLERK' AND department_id=20);
```

14) Find the details of all the manager in dept 10, all clerk in dept 20

```
SELECT * FROM employees
WHERE ((job_id LIKE '%MGR' OR job_id LIKE '%MAN') AND department_id=10)
OR (job_id LIKE '%CLERK' AND department_id=20);
```

15) And all employees who are neither clerks nor manager but whose salary is greater than or equal to Rs. 2000.

```
SELECT * FROM employees
WHERE NOT(job_id LIKE '%MGR' OR job_id LIKE '%MAN' OR job_id LIKE '%CLERK') AND salary>=
2000;
```

16) Find the names of everyone in deptno 20 who is neither a clerk nor a Manager.

```
SELECT first_name, last_name FROM employees
WHERE NOT(job_id LIKE '%MGR' OR job_id LIKE '%MAN' OR job_id LIKE '%CLERK') AND
department_id=20;
```

17) Find the employees who earns between Rs. 1200 and Rs.1400.

```
SELECT * FROM employees  
WHERE salary BETWEEN 1200 AND 1400;
```

```
SELECT * FROM employees  
WHERE salary >= 1200 AND salary <= 1400;
```

(not sure if 1200 and 1400 are inclusive values)

18) Find the employees who are clerks, analysts or salesman.

```
SELECT * FROM employees  
WHERE job_id LIKE '%CLERK' or job_id IN ('SA_REP', 'IT_PROG');
```

19) Find the employees who are not clerks, analyst or salesman.

```
SELECT * FROM employees  
WHERE NOT(job_id LIKE '%CLERK' or job_id IN ('SA_REP', 'IT_PROG'));
```

20) Find the employees who do not receive a commission.

```
SELECT * FROM employees  
WHERE commission_pct IS NULL;
```

21) Find the employee whose commission is Rs. 0.

```
SELECT * FROM employees  
WHERE commission_pct=0;
```

22) Find the different jobs of the employees receiving commission.

```
SELECT job_id FROM employees  
WHERE commission_pct IS NOT NULL;
```

```
SELECT DISTINCT job_id FROM employees  
WHERE commission_pct IS NOT NULL;
```

23) Find all employees who do not receive a commission or whose Commission is less than 0.1 .
If all employees not receiving commission are entitled to Rs. 250, Show the net earnings of all employees.

```
SELECT * FROM employees  
WHERE commission_pct IS NULL or commission_pct < 0.1;
```

24) Find all employees whose total earnings are greater than Rs. 2000.

(what is the definition of total earnings?)

25) Find all employees whose names begin with m.

```
SELECT * FROM employees  
WHERE first_name LIKE 'm%';
```

26) Find all employees whose names end with m.

```
SELECT * FROM employees  
WHERE last_name LIKE '%m';
```

27) Find all employees whose names contain the letter m in any case.

```
SELECT * FROM employees  
WHERE lower(first_name) LIKE '%m%' OR lower(last_name) LIKE '%m%';
```

if first_name only needed:

```
SELECT * FROM employees  
WHERE lower(first_name) LIKE '%m%'
```

28) Find the employees whose names are 5 characters long and end with n.

```
SELECT * FROM employees  
WHERE first_name LIKE '____n';
```

29) Find the employees who have the letter r as the third letter in their name.

```
SELECT * FROM employees  
WHERE first_name LIKE '__r%';
```

30) Find all employees hired in month of February (of any year).

```
SELECT * FROM employees  
WHERE EXTRACT(MONTH FROM hire_date)=2;
```

31) Find all employees who were hired on the last day of the month.

```
SELECT * FROM employees  
WHERE LAST_DAY(hire_date)=hire_date;
```

32) Find the employees who were hired more than 12 years ago.

```
SELECT * FROM employees  
WHERE MONTHS_BETWEEN(hire_date, SYSDATE)>144;
```

33) Find the managers hired in the year 1981.

```
SELECT * FROM employees
WHERE EXTRACT(YEAR FROM hire_date)=1981 AND (job_id LIKE '%MGR' OR job_id LIKE '%
MAN');
```

34) Display the names and the jobs of all employees, separated by a','.

```
SELECT first_name || ',' || last_name || ',' || job_id FROM employees;
```

35) Display the names of all employees with the initial letter only in capitals.

```
SELECT initcap(first_name) FROM employees;
```

36) Display the length of the name of all employees.

```
SELECT LENGTH(first_name) AS "Length of First Name" FROM employees;
```

37) Show the first three characters of the names of all employees.

```
SELECT SUBSTR(first_name, 1, 3) AS "First 3 Letters of First Name" FROM employees;
```

38) Show the last three characters of the names of all employees.

```
SELECT SUBSTR(first_name, -3, LENGTH(first_name)) AS "Last 3 Letters of First Name" FROM
employees;
```

39) Display the names of all employees with any 'a'.

```
SELECT first_name, last_name FROM employees
WHERE lower(first_name) LIKE '%a%' OR lower(last_name) LIKE '%a%';
```

if first_name only needed:

```
SELECT first_name FROM employees
WHERE lower(first_name) LIKE '%a%'
```

40) Display the names of all employees and the position at which the string 'ar' occurs in the name.

```
SELECT first_name, INSTR(first_name, 'ar') FROM employees
WHERE first_name LIKE '%ar%';
```

41) Show the salary of all employees rounding it to the nearest Rs. 1000.

```
SELECT ROUND(salary, -3) FROM employees;
```

42) Show the salary of all employees ignoring fractions ,less than Rs.1000.

```
SELECT TRUNC(salary) FROM employees WHERE salary < 1000;
```

43) Display the details of all employees, sorted on the names.

```
SELECT * FROM employees
ORDER BY first_name, last_name;
```

44) Display the name of all employees, based on their tenure, with the oldest employee coming first.

```
SELECT first_name, last_name FROM employees
ORDER BY hire_date;
```

45) Display the names, job and salary of all employees sorted on jobs and Salary.

```
SELECT first_name, last_name, job_id, salary FROM employees
ORDER BY job_id, salary;
```

46) Display the names, job and salary of all employees, sorted on jobs and within job, sorted on the descending order of salary.

```
SELECT first_name, last_name, job_id, salary FROM employees
ORDER BY job_id ASC, salary DESC;
```

SUBQUERIES

47) write a SQL query to find those employees who get higher salary than the employee whose ID is 163. Return first name, last name.

```
select first_name, last_name from employees
where salary > (select salary from employees where
employee_id=163);
```

48) Display the name, salary, department id, job id for those employees who works in the same designation as the employee works whose id is 169

```
select first_name, salary, department_id, job_id from employees
where job_id=(select job_id from employees where
employee_id=169);
```

49) Display the name, salary, department id for those employees who earn such amount of salary which is the smallest salary of any of the departments

```
select first_name, salary, department_id from employees
```

```
where salary = any(select min(salary) from employees
group by department_id);
```

```
select first_name, salary, department_id from employees
where salary in(select min(salary) from employees
group by department_id);
```

50) Display the employee id, employee name for all employees who earn more than the average salary

```
select employee_id, first_name from employees
where salary > all(select avg(salary) from employees);
```

```
select employee_id, first_name from employees
where salary > (select avg(salary) from employees);
```

51) Display the employee name, employee id and salary of all employees who report to John

```
select first_name, employee_id, salary from employees
where manager_id=any(select employee_id from employees
where first_name='John');
```

52) SQL query to find all those employees who work in the HR department. Return department ID, name (first name), job ID and department name.

```
select departments.department_id, first_name, job_id, department_name
from employees join departments
on(employees.department_id=departments.department_id)
where employees.department_id=(select department_id
from departments where department_name='Human Resources');
```

--w/o subquery:

```
select e.department_id, first_name, job_id, department_name
from employees e, departments d
where e.department_id = d.department_id
AND d.department_name='Human Resources';
```

53) write a SQL query to find those employees whose ID matches any of the number 134, 159

and 183. Return all the fields.

```
select * from employees
where employee_id in(134, 159, 183);
```

```
select * from employees
where employee_id = any(134, 159, 183);
```

TABLES:

1.

```
create table salesman_master(
salesman_no varchar(6) primary key
constraint start_s check(salesman_no like 'S%'),
salesman_name varchar(20) not null,
Address1 varchar(30) not null,
Address2 varchar(30),
city varchar(20),
pincode number(6),
"state" varchar(20),
sal_amt number(8,2) not null
constraint salchecknotzero check(sal_amt!=0),
tgt_to_get number(6,2) not null
constraint tgtchecknotzero check(tgt_to_get!=0),
ytd_dales number(6,2) not null,
remarks varchar(60));
```

2.

```
create table sales_order(
s_order_no varchar(6) primary key
constraint start_o check(s_order_no like 'O%'),
s_order_date date,
client_no varchar(6) references client_master(client_no),
dely_addr varchar(25),
salesman_no varchar(6) references salesman_master(salesman_no),
dely_type char(1) default 'F',
billed_yn char(1) default 'N',
```



```
dely_date date,
constraint check_dely_date check(dely_date >= s_order_date),
order_status varchar(10),
constraint check_order_status check(order_status in('in process', 'Fulfilled', 'BackOrder',
'Canceled')));
```

3.

```
create table sales_order_details(
s_order_no varchar(6) references sales_order,
product_no varchar(6) references product_master,
qty_ordered number(8),
qty_disp number(8),
product_rate number(10,2));
```

4.

```
INSERT INTO salesman_master
```

```
VALUES(
```

```
    'S00001',
```

```
    'Kiran',
```

```
    'A/14',
```

```
    'Worli',
```

```
    'Bombay',
```

```
    400002,
```

```
    'MAH',
```

```
    3000,
```

```
    100,
```

```
    50,
```

```
    'Good'
```

```
);
```

```
INSERT INTO salesman_master
```

```
VALUES(
```

```
    'S00002',
```

```
    'Manish',
```

```
    '65',
```

```
    'Nariman',
```

```
    'Bombay',
```

```
    400001,
```

```
    'MAH',
```

```
        3000,  
        200,  
        100,  
        'Good'  
    );
```

```
INSERT INTO salesman_master  
VALUES(  
    'S00003',  
    'Ravi',  
    'P-7',  
    'Bandra',  
    'Bombay',  
    400032,  
    'MAH',  
    3000,  
    200,  
    100,  
    'Good'  
);
```

```
INSERT INTO salesman_master  
VALUES(  
    'S00004',  
    'Ashish',  
    'A/5',  
    'Juhu',  
    'Bombay',  
    400044,  
    'MAH',  
    3500,  
    200,  
    150,  
    'Good'  
);
```

5.

```
INSERT INTO sales_order  
(s_order_no, s_order_date, client_no, dely_addr,  
salesman_no, dely_type, billed_yn, dely_date, order_status)
```

```
VALUES(
    'O19001',
    '12-JAN-21',
    'C00001',
    NULL,
    'S00001',
    'F',
    'N',
    '20-JAN-21',
    'in process'
);
```

```
INSERT INTO sales_order
(s_order_no, s_order_date, client_no, dely_addr,
salesman_no, dely_type, billed_yn, dely_date, order_status)
VALUES(
    'O19002',
    '25-JAN-21',
    'C00002',
    NULL,
    'S00002',
    'P',
    'N',
    '27-JAN-21',
    'Canceled'
);
```

```
INSERT INTO sales_order
(s_order_no, s_order_date, client_no, dely_addr,
salesman_no, dely_type, billed_yn, dely_date, order_status)
VALUES(
    'O46865',
    '18-FEB-21',
    'C00003',
    NULL,
    'S00003',
    'F',
    'Y',
    '20-FEB-21',
    'Fulfilled'
);
```

```
);
```

```
INSERT INTO sales_order
(s_order_no, s_order_date, client_no, dely_addr,
salesman_no, dely_type, billed_yn, dely_date, order_status)
VALUES(
    'O19003',
    '03-APR-21',
    'C00001',
    NULL,
    'S00001',
    'F',
    'Y',
    '07-APR-21',
    'Fulfilled'
);
```

```
INSERT INTO sales_order
(s_order_no, s_order_date, client_no, dely_addr,
salesman_no, dely_type, billed_yn, dely_date, order_status)
VALUES(
    'O46866',
    '20-MAY-21',
    'C00004',
    NULL,
    'S00002',
    'P',
    'N',
    '22-MAY-21',
    'Canceled'
);
```

```
INSERT INTO sales_order
(s_order_no, s_order_date, client_no, dely_addr,
salesman_no, dely_type, billed_yn, dely_date, order_status)
VALUES(
    'O10008',
    '24-MAY-21',
    'C00005',
    NULL,
```

```
'S00004',  
'F',  
'N',  
'26-MAY-21',  
'in process'  
);
```

6.

```
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O19001','P00001',4,4,525);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O19001','P07965',2,1,8400);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O19001','P07885',2,1,5250);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O19002','P00001',10,0,525);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O46865','P07868',3,3,3150);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O46865','P07885',3,1,5250);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O46865','P00001',10,10,525);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O46865','P03453',4,4,1050);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O19003','P03453',2,2,1050);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O19003','P06734',1,1,12000);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O46866','P07965',1,0,8400);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O46866','P07975',1,0,1050);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O10008','P00001',10,5,525);  
INSERT INTO SALES_ORDER_DETAILS  
VALUES ('O10008','P07975',5,3,1050);
```