

```

-- Q-1. Write an SQL query to fetch "FIRST_NAME" from Worker table using the alias
name as <WORKER_NAME>.
select first_name AS WORKER_NAME from worker;

-- Q-2. Write an SQL query to fetch "FIRST_NAME" from Worker table in upper case.
select UPPER(first_name) from worker;

-- Q-3. Write an SQL query to fetch unique values of DEPARTMENT from Worker table.
SELECT distinct department from worker;

-- Q-4. Write an SQL query to print the first three characters of FIRST_NAME from
Worker table.
select substring(first_name, 1, 3) from worker;

-- Q-5. Write an SQL query to find the position of the alphabet ('b') in the first
name column 'Amitabh' from Worker table.
select INSTR(first_name, 'B') from worker where first_name = 'Amitabh';

-- Q-6. Write an SQL query to print the FIRST_NAME from Worker table after
removing white spaces from the right side.
select RTRIM(first_name) from worker;

-- Q-7. Write an SQL query to print the DEPARTMENT from Worker table after
removing white spaces from the left side.
select LTRIM(first_name) from worker;

-- Q-8. Write an SQL query that fetches the unique values of DEPARTMENT from
Worker table and prints its length.
select distinct department, LENGTH(department) from worker;

-- Q-9. Write an SQL query to print the FIRST_NAME from Worker table after
replacing 'a' with 'A'.
select REPLACE(first_name, 'a', 'A') from worker;

-- Q-10. Write an SQL query to print the FIRST_NAME and LAST_NAME from Worker
table into a single column COMPLETE_NAME.
-- A space char should separate them.
select CONCAT(first_name, ' ', last_name) AS COMPLETE_NAME from worker;

-- Q-11. Write an SQL query to print all Worker details from the Worker table
order by FIRST_NAME Ascending.
select * from worker ORDER by first_name;

-- Q-12. Write an SQL query to print all Worker details from the Worker table
order by
-- FIRST_NAME Ascending and DEPARTMENT Descending.
select * from worker order by first_name, department DESC;

-- Q-13. Write an SQL query to print details for Workers with the first name as
"Vipul" and "Satish" from Worker table.

```

```

select * from worker where first_name IN ('Vipul', 'Satish');

-- Q-14. Write an SQL query to print details of workers excluding first names,
"Vipul" and "Satish" from Worker table.
select * from worker where first_name NOT IN ('Vipul', 'Satish');

-- Q-15. Write an SQL query to print details of Workers with DEPARTMENT name as
"Admin*".
select * from worker where department LIKE 'Admin%';

-- Q-16. Write an SQL query to print details of the Workers whose FIRST_NAME
contains 'a'.
select * from worker where first_name LIKE '%a%';

-- Q-17. Write an SQL query to print details of the Workers whose FIRST_NAME ends
with 'a'.
select * from worker where first_name LIKE '%a';

-- Q-18. Write an SQL query to print details of the Workers whose FIRST_NAME ends
with 'h' and contains six alphabets.
select * from worker where first_name LIKE '_____h';

-- Q-19. Write an SQL query to print details of the Workers whose SALARY lies
between 100000 and 500000.
select * from worker where salary between 100000 AND 500000;

-- Q-20. Write an SQL query to print details of the Workers who have joined in
Feb'2014.
select * from worker where YEAR(joining_date) = 2014 AND MONTH(joining_date) = 02;

-- Q-21. Write an SQL query to fetch the count of employees working in the
department 'Admin'.
select department, count(*) from worker where department = 'Admin';

-- Q-22. Write an SQL query to fetch worker full names with salaries >= 50000 and
<= 100000.
select concat(first_name, ' ', last_name) from worker
where salary between 50000 and 100000;

-- Q-23. Write an SQL query to fetch the no. of workers for each department in the
descending order.
select department, count(worker_id) AS no_of_worker from worker group by
department
ORDER BY no_of_worker desc;

-- Q-24. Write an SQL query to print details of the Workers who are also Managers.
select w.* from worker as w inner join title as t on w.worker_id = t.worker_ref_id
where t.worker_title = 'Manager';

-- Q-25. Write an SQL query to fetch number (more than 1) of same titles in the

```

ORG of different types.

```
select worker_title, count(*) as count from title group by worker_title having  
count > 1;
```

-- Q-26. Write an SQL query to show only odd rows from a table.

```
-- select * from worker where MOD (WORKER_ID, 2) != 0;  
select * from worker where MOD (WORKER_ID, 2) <> 0;
```

-- Q-27. Write an SQL query to show only even rows from a table.

```
select * from worker where MOD (WORKER_ID, 2) = 0;
```

-- Q-28. Write an SQL query to clone a new table from another table.

```
CREATE TABLE worker_clone LIKE worker;  
INSERT INTO worker_clone select * from worker;  
select * from worker_clone;
```

-- Q-29. Write an SQL query to fetch intersecting records of two tables.

```
select worker.* from worker inner join worker_clone using(worker_id);
```

-- Q-30. Write an SQL query to show records from one table that another table does not have.

-- MINUS

```
select worker.* from worker left join worker_clone using(worker_id) WHERE  
worker_clone.worker_id is NULL;
```

-- Q-31. Write an SQL query to show the current date and time.

-- DUAL

```
select curdate();  
select now();
```

-- Q-32. Write an SQL query to show the top n (say 5) records of a table order by descending salary.

```
select * from worker order by salary desc LIMIT 5;
```

-- Q-33. Write an SQL query to determine the nth (say n=5) highest salary from a table.

```
select * from worker order by salary desc LIMIT 4,1;
```

-- Q-34. Write an SQL query to determine the 5th highest salary without using LIMIT keyword.

```
select salary from worker w1  
WHERE 4 = (  
SELECT COUNT(DISTINCT (w2.salary))  
from worker w2  
where w2.salary >= w1.salary  
);
```

-- Q-35. Write an SQL query to fetch the list of employees with the same salary.

```
select w1.* from worker w1, worker w2 where w1.salary = w2.salary and w1.worker_id  
!= w2.worker_id;
```

```

-- Q-36. Write an SQL query to show the second highest salary from a table using
sub-query.
select max(salary) from worker
where salary not in (select max(salary) from worker);

-- Q-37. Write an SQL query to show one row twice in results from a table.
select * from worker
UNION ALL
select * from worker ORDER BY worker_id;

-- Q-38. Write an SQL query to list worker_id who does not get bonus.
select worker_id from worker where worker_id not in (select worker_ref_id from
bonus);

-- Q-39. Write an SQL query to fetch the first 50% records from a table.
select * from worker where worker_id <= ( select count(worker_id)/2 from worker);

-- Q-40. Write an SQL query to fetch the departments that have less than 4 people
in it.
select department, count(department) as depCount from worker group by department
having depCount < 4;

-- Q-41. Write an SQL query to show all departments along with the number of
people in there.
select department, count(department) as depCount from worker group by department;

-- Q-42. Write an SQL query to show the last record from a table.
select * from worker where worker_id = (select max(worker_id) from worker);

-- Q-43. Write an SQL query to fetch the first row of a table.
select * from worker where worker_id = (select min(worker_id) from worker);

-- Q-44. Write an SQL query to fetch the last five records from a table.
(select * from worker order by worker_id desc limit 5) order by worker_id;

-- Q-45. Write an SQL query to print the name of employees having the highest
salary in each department.
select w.department, w.first_name, w.salary from
(select max(salary) as maxsal, department from worker group by department) temp
inner join worker w on temp.department = w.department and temp.maxsal = w.salary;

-- Q-46. Write an SQL query to fetch three max salaries from a table using
co-related subquery
select distinct salary from worker w1
where 3 >= (select count(distinct salary) from worker w2 where w1.salary <=
w2.salary) order by w1.salary desc;
-- DRY RUN AFTER REVISING THE CORELATED SUBQUERY CONCEPT FROM LEC-9.
select distinct salary from worker order by salary desc limit 3;

```

-- Q-47. Write an SQL query to fetch three min salaries from a table using co-related subquery
select distinct salary from worker w1
where 3 >= (select count(distinct salary) from worker w2 where w1.salary >= w2.salary) order by w1.salary desc;

-- Q-48. Write an SQL query to fetch nth max salaries from a table.
select distinct salary from worker w1
where n >= (select count(distinct salary) from worker w2 where w1.salary <= w2.salary) order by w1.salary desc;

-- Q-49. Write an SQL query to fetch departments along with the total salaries paid for each of them.
select department , sum(salary) as depSal from worker group by department order by depSal desc;

-- Q-50. Write an SQL query to fetch the names of workers who earn the highest salary.
select first_name, salary from worker where salary = (select max(Salary) from worker);