ASSIGNMENT-1

* create database cognidemo;
* use cognidemo;
* show tables;
* select \* from worker;
* insert into worker values(001,'Monika','Arora',100000,'2014-02-20 09:00:00','HR');
* insert into worker (worker\_id,first\_name,last\_name,salary,joining\_date,department)
* values(002,'Niharika','Verma',80000,'2014-06-11 09:00:00','Admin'),

(003,'Vishal','Singhal',300000,'2014-02-20 09:00:00','HR'),

(004,'Amitabh','Singh',500000,'2014-02-20 09:00:00','Admin'),

(005,'Vivek','Bhati',500000,'2014-06-11 09:00:00','Admin'),

(006,'Vipul','Diwan',200000,'2014-06-11 09:00:00','Account'),

(007,'Satish','Kumar',75000,'2014-01-20 09:00:00','Accouaant'),

(008,'Geetika','Chauhan',90000,'2014-04-11 09:00:00','Admin');

* select \* from worker;
* select \* from bonus;
* insert into bonus(worker\_ref\_id,bonus\_date,bonus\_amount)
* values

(1,'2016-02-20 00:00:00',5000),

(2,'2016-06-11 00:00:00',3000),

(3,'2016-02-20 00:00:00',4000),

(1,'2016-02-20 00:00:00',4500),

(2,'2016-06-11 00:00:00',3500);

* select \* from bonus;
* select \* from title;
* insert into title(worker\_ref\_id,worker\_title,affected\_from)
* values

(1,'Manager','2016-02-20 00:00:00'),

(2,'Executive','2016-06-11 00:00:00'),

(8,'Executive','2016-06-11 00:00:00'),

(5,'Manager','2016-06-11 00:00:00'),

(4,'Asst.Manager','2016-06-11 00:00:00'),

(7,'Executive','2016-06-11 00:00:00'),

(6,'Lead','2016-06-11 00:00:00'),

(3,'Lead','2016-06-11 00:00:00');

* select \* from title;

**-- Q-1. Write an SQL query to fetch “FIRST\_NAME” from Worker table using the alias name as <WORKER\_NAME>.**

select first\_name

from worker as worker\_name;

**-- Q-2. Write an SQL query to fetch “FIRST\_NAME” from Worker table in upper case.**

update worker

set first\_name=upper(first\_name);

select first\_name

from worker as worker\_name;

update worker set first\_name=lcase(first\_name);

**-- 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 (‘a’) in the first name column ‘Amitabh’ from Worker table.**

select first\_name, last\_name, position("a" in "Amitabh")

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 first\_name, 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 department, ltrim(department)

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) as 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’.**

update worker

set first\_name=lcase(first\_name);

select first\_name, replace(first\_name,"a","A") as replaced\_atoA

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.**

update worker set first\_name=ucase(first\_name), last\_name=ucase(last\_name);

select first\_name, last\_name, 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 asc;

**-- 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 department desc, first\_name asc;

**-- 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="Vipul" or first\_name="Satish";

# OR

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='Admin';

# OR

select \*

from worker

where department in ('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' and length(first\_name)=6;

**-- 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;

# OR

select \*

from worker

where salary>=100000 and salary<=500000;

**-- Q-20. Write an SQL query to print details of the Workers who have joined in Feb’2014.**

select \*

from worker

where month(joining\_date)=2 and year(joining\_date)=2014;

**-- Q-21. Write an SQL query to fetch the count of employees working in the department ‘Admin’.**

select count(department) as department\_count\_of\_admin

from worker

where department='Admin';

# OR

select count(\*)

from worker

where department='Admin';

**-- Q-22. Write an SQL query to fetch worker names with salaries >= 50000 and <= 100000.**

select first\_name, last\_name, salary

from worker

where salary>=50000 and salary<= 100000;

# OR

select first\_name, last\_name, salary

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\_workers

from worker

group by department

order by no\_of\_workers desc;

**-- Q-24. Write an SQL query to print details of the Workers who are also Managers.**

select distinct \*

from worker as w, title as t

where w.worker\_id=t.worker\_ref\_id and t.worker\_title='Manager';

**-- Q-25. Write an SQL query to fetch duplicate records having matching data in some fields of a table.**

select \* from worker;

select \*, count(worker\_id) from worker

group by first\_name,last\_name,salary,joining\_date,department

having count(worker\_id)>1;

# OR

select worker\_title, affected\_from, count(\*)

from title

group by worker\_title, affected\_from

having count(\*)>1;

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

select distinct\*

from worker

where mod(worker\_id,2) <> 0;

# OR

SELECT \*

FROM Worker

WHERE MOD (WORKER\_ID, 2) <> 0;

# OR

SELECT \*

FROM worker

WHERE worker\_id IN(SELECT worker\_id FROM worker WHERE worker\_id%2 = 1);

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

select distinct\*

from worker

where mod(worker\_id,2) = 0;

# OR

SELECT \*

FROM Worker WHERE MOD (WORKER\_ID, 2) = 0;

# OR

SELECT \*

FROM worker

WHERE worker\_id IN

(SELECT worker\_id

FROM worker

WHERE worker\_id%2 = 0);

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

create table new\_worker\_table as select \*

from worker;

select \*

from new\_worker\_table;

# OR

create table new\_worker\_table\_2 like worker; #only table structure created, no values input yet

select \*

from new\_worker\_table\_2;

insert into new\_worker\_table\_2

select \*

from worker; #values cloned into new table from old table

select \*

from new\_worker\_table\_2;

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

select distinct worker\_id as id,first\_name, last\_name,salary,joining\_date,department,worker\_title,affected\_from

from worker as w join title as t on w.worker\_id=t.worker\_ref\_id;

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

select distinct t.worker\_ref\_id as id,worker\_title, affected\_from, bonus\_date, bonus\_amount

from title as t join bonus b on t.worker\_ref\_id=b.worker\_ref\_id;