create table worker

(

worker\_id int primary key identity(1,1),

first\_name varchar(15),

last\_name varchar(15),

salary money,

joining\_date date,

dept varchar(10)

);

insert into worker values

('monika','arora',100000,'2014-02-09','HR'),

('niharika','verma',80000,'2014-06-11','admin'),

('vishal','singhal',300000,'2014-02-20','HR'),

('Amitabh','Singh',500000,'2014-02-20','Admin'),

('Vivek','Bhati',500000,'2014-06-11','Admin'),

('Vipul','Diwan',200000,'2014-06-11','Account'),

('Satish','Kumar',75000,'2014-01-20','Account'),

('Geetika','Chauhan',90000,'2014-04-11','Admin');

select \* from worker;

create table bonus

(

worker\_ref\_id int,

bonus\_date date,

bonus\_amount money

);

insert into bonus values

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

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

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

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

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

select \* from bonus;

create table title

(

worker\_ref\_id int,

worker\_title varchar(15),

affected\_from date

);

insert into title values

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

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

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

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

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

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

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

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

select \* from worker;

select \* from bonus;

select \* from title;

--Q1 Write an SQL query to fetch “FIRST\_NAME” from Worker table using the alias name as <WORKER\_NAME>.

select first\_name [Worker\_Name] from worker;

--Q2 Write an SQL query to fetch “FIRST\_NAME” from Worker table in upper case.

select UPPER(first\_name) [Worker\_Name] from worker;

--Q3 Write an SQL query to fetch unique values of DEPARTMENT from Worker table.

select \* from worker;

select distinct dept from worker;

--Q4 Write an SQL query to print the first three characters of FIRST\_NAME from Worker table.

select SUBSTRING(first\_name,1,3) from worker;

select LEFT(first\_name,3) from worker;

select right(first\_name,2) from worker;

--Q5 Write an SQL query to find the position of the alphabet (‘a’) in the first name column ‘Amitabh’ from Worker table.

select \* from worker;

select CHARINDEX('b',first\_name) from worker where first\_name='Amitabh';

-- Q6 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;

--Q7 Write an SQL query to print the DEPARTMENT from Worker table after removing white spaces from the left side.

select LTRIM(dept) from worker;

--Q8 Write an SQL query that fetches the unique values of DEPARTMENT from Worker table and prints its length.

select distinct dept [department], LEN(dept) [Length] from worker;

--Q9 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;

--Q10 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) [Complete\_Name] from worker;

--Q11 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;

--Q12 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 asc;

select \* from worker order by dept desc;

select \* from worker order by first\_name asc, dept desc;

--Q13 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');

--Q14 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');

--Q15 Write an SQL query to print details of Workers with DEPARTMENT name as “Admin”.

select \* from worker where dept='Admin';

--Q16 Write an SQL query to print details of the Workers whose FIRST\_NAME contains ‘a’.

select \* from worker where first\_name like '%a%';

--Q17 Write an SQL query to print details of the Workers whose FIRST\_NAME ends with ‘a’.

select \* from worker where first\_name like '%a';

--Q18 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 LEN(first\_name)=6;

--Q19 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;

--Q20 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;

--Q21 Write an SQL query to fetch the count of employees working in the department ‘Admin’.

select dept, count(worker\_id) [Count] from worker where dept='Admin' group by dept;

select DAY(joining\_date) from worker;

--Q22 Write an SQL query to fetch worker names with salaries >= 50000 and <= 100000.

select first\_name, last\_name,salary from worker where salary between 50000 and 100000 order by salary desc;

--Q23 Write an SQL query to fetch the no. of workers for each department in the descending order.

select dept, count(worker\_id) [Count] from worker group by dept order by count(worker\_id) desc;

--Q24 Write an SQL query to print details of the Workers who are also Managers.

select \* from worker;

select \* from title;

select \* from worker w, title t where w.worker\_id=t.worker\_ref\_id and t.worker\_title='Manager';

--Q25 Write an SQL query to fetch duplicate records from title table having matching data in some fields of a table.

select worker\_title, affected\_from, count(\*)[Count] from title group by worker\_title, affected\_from having count(\*)>1 order by count(\*);

--Q26 Write an SQL query to show only odd rows from a table.

select \* from worker where worker\_id % 2=1;

--Q27 Write an SQL query to show only even rows from a table.

select \* from worker where worker\_id % 2=0;

--Q28 Write an SQL query to clone a new table from another table.

select \* into worker\_clone from worker where 1=1;

select \* from worker;

select \* from worker\_clone;

--Q29 Write an SQL query to fetch intersecting records of two tables.

(select \* from worker)

intersect

(select \* from worker\_clone);

--Q30 Write an SQL query to show records from one table that another table does not have.

select \* from worker where worker\_id not in (select worker\_id from worker\_clone);

--Q31 Write an SQL query to show the current date and time.

select GETDATE();

--Q32 Write an SQL query to show the top n (say 10) records of a table.

select top 10 \* from worker;

--Q33 Write an SQL query to determine the nth (say n=5) highest salary from a table.

select \* from worker order by salary desc;

with cte as

(

select worker\_id, first\_name, last\_name, salary, rownum=ROW\_NUMBER() over(order by salary desc) from worker

)

select \* from cte where rownum=5;

with cte as

(

select worker\_id, first\_name, last\_name, salary, ranknum=dense\_Rank() over(order by salary desc) from worker

)

select \* from cte where ranknum=5;

--Q34 Write an SQL query to determine the 5th highest salary without using TOP or limit method.

with cte as

(

select worker\_id, first\_name, last\_name, salary, ranknum=dense\_Rank() over(order by salary desc) from worker

)

select \* from cte where ranknum=5;

--Q35 Write an SQL query to fetch the list of employees with the same salary.

select \* from worker w1, worker w2 where w1.worker\_id<>w2.worker\_id and w1.salary=w2.salary;

--Q36 Write an SQL query to show the second highest salary from a table.

with cte as

(

select worker\_id, first\_name, last\_name, salary, ranknum=dense\_Rank() over(order by salary desc) from worker

)

select \* from cte where ranknum=2;

--Q37 Write an SQL query to show one row twice in results from a table.

select first\_name, dept from worker w1 where dept='HR'

union all

select first\_name, dept from worker w2 where dept='HR';

select \* from worker

except

select \* from worker\_clone;

select \* from worker\_clone

except

select \* from worker;

--Q38 Write an SQL query to fetch intersecting records of two tables.

delete from worker\_clone where worker\_id in (6,7,8);

select \* from worker

intersect

select \* from worker\_clone;

--Q39 Write an SQL query to fetch the first 50% records from a table.

select \* from worker where worker\_id in (select worker\_id/2 from worker);

select top 50 percent \* from worker;

--select \* from worker where worker\_id exists (select worker\_id from worker\_clone);

--Q40 Write an SQL query to fetch the departments that have less than five people in it.

select dept, count(worker\_id) from worker group by dept having count(worker\_id)<5 order by count(worker\_id);

--Q41 Write an SQL query to show all departments along with the number of people in there.

select dept, count(worker\_id) from worker group by dept;

--Q42 Write an SQL query to show the last record from a table.

select top 1 \* from worker order by worker\_id desc;

select \* from worker where worker\_id=(select max(worker\_id) from worker);

--Q43 Write an SQL query to fetch the first row of a table.

select \* from worker where worker\_id=(select min(worker\_id) from worker);

select top 1 \* from title;

--select \* from title;

--Q44 Write an SQL query to fetch the last five records from a table.

select \* from worker where worker\_id>(select count(worker\_id) from worker)-5;

--Q45 Write an SQL query to print the name of employees having the highest salary in each department.

select \* from worker;

with cte as

(select \*, ranknum=DENSE\_RANK() over (partition by dept order by salary desc) from worker)

select \* from cte where ranknum=1;

-- Write an SQL query to print the name of employees having the lowest salary in each department.

with cte as

(select \*, ranknum=DENSE\_RANK() over (partition by dept order by salary asc) from worker)

select \* from cte where ranknum=1;

--Q46 Write an SQL query to fetch three max salaries from a table.

with cte as

(

select worker\_id, first\_name, last\_name, salary, rownum=dense\_rank() over(order by salary desc) from worker

)

select \* from cte where rownum<=3;

--Q47 Write an SQL query to fetch three min salaries from a table.

with cte as

(

select worker\_id, first\_name, last\_name, salary, rownum=ROW\_NUMBER() over(order by salary ASC) from worker

)

select \* from cte where rownum<=3;

--Q48 Write an SQL query to fetch nth max salaries from a table.

with cte as

(

select worker\_id, first\_name, last\_name, salary, rownum=dense\_rank() over(order by salary desc) from worker

)

select \* from cte where rownum=5;

--Q49 Write an SQL query to fetch departments along with the total salaries paid for each of them.

select dept, sum(salary) [Total\_Salary] from worker group by dept;

--Q50 Write an SQL query to fetch the names of workers who earn the highest salary.

select first\_name, dept, salary from worker where salary=(select max(salary) from worker);

with cte as

(

select \*, ranknum=dense\_rank() over (order by salary desc) from worker

)

select \* from cte where ranknum=1;