

SQL PROJECT

- **CREATING DATA - BASE :**
CREATE DATABASE ORG;
- **USE DATA - BASE :**
USE ORG;
- Creating 3 tables = 1. **Worker** ,
2. **bonus** ,
3. **Title**.
- INSERTING VALUES IN ABOVE 3 TABLES.

TASK 1 : CREATING 3 TABLES & INSERTING VALUES IN TO 3 TABLES

TABLE 1 -> **WORKER** :

```
CREATE TABLE Worker (  
WORKER_ID INT NOT NULL PRIMARY KEY AUTO_INCREMENT ,  
FIRST_NAME VARCHAR(25),  
LAST_NAME VARCHAR(25),  
SALARY INT,  
JOINING_DATE DATETIME,  
DEPARTMENT VARCHAR(25)  
);
```

INSERTING VALUES :

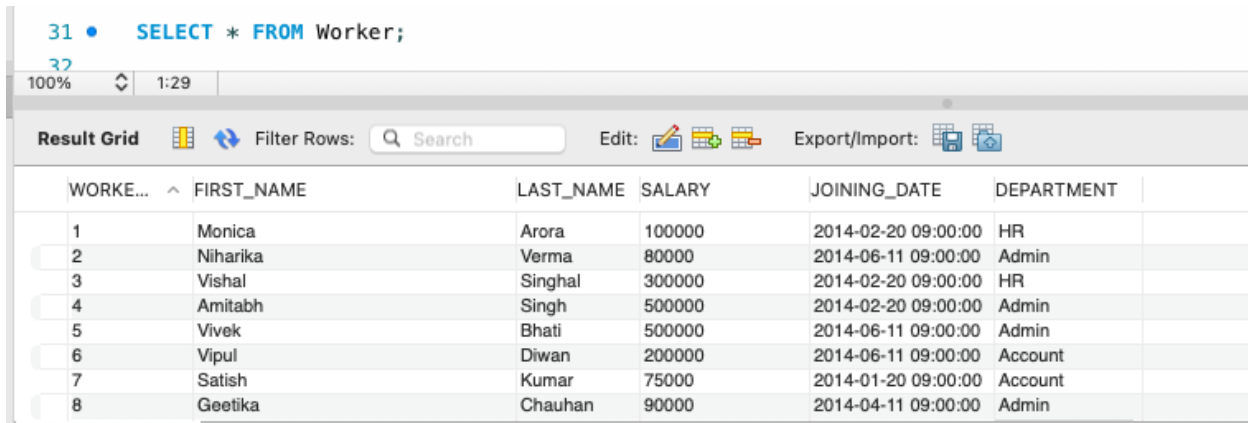
```
INSERT INTO Worker  
(WORKER_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)  
VALUES (001, 'Monica', 'Arora', 100000, '14-02-20 09:00:00', 'HR'),  
(002, 'Niharika', 'Verma', 80000, '14-06-11 09:00:00', 'Admin'),  
(003, 'Vishal', 'Singhal', 300000, '2014-02-20 09:00:00', 'HR'),  
(004, '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', 'Account'),  
(008, 'Geetika', 'Chauhan', 90000, '2014-04-11 09:00:00', 'Admin');
```

TO CHECK THE TABLE :

OUTPUT =>

```
SELECT * FROM TABLE_NAME ;
```

```
SELECT * FROM Worker ;
```



31 • SELECT * FROM Worker;

32

100% 1:29

Result Grid Filter Rows: Search Edit: Export/Import:

WORKER...	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin

TABLE 2 - **Bonus**

```
CREATE TABLE Bonus (  
WORKER_REF_ID INT,  
BONUS_DATE DATETIME,  
BONUS_AMOUNT INT,  
FOREIGN KEY(WORKER_REF_ID)  
REFERENCES Worker(WORKER_ID)  
ON DELETE CASCADE  
);
```

INSERTING VALUES :

```
INSERT INTO Bonus (WORKER_REF_ID, BONUS_DATE, BONUS_AMOUNT)  
VALUES (001, '2016-02-20 00:00:00', 5000),  
(002, '2016-06-11 00:00:00', 3000),  
(003, '2016-02-20 00:00:00', 4000),  
(001, '2016-02-20 00:00:00', 4500),  
(002, '2016-06-11 00:00:00', 3500);
```

TO CHECK THE TABLE :
OUTPUT =>

```
SELECT * FROM TABLE_NAME ;
```

SELECT * FROM Bonus;




Result Grid   Filter Rows: <input type="text" value="Search"/> Export: 			
	WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT
▶	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

TABLE 3 = **Title**

```
CREATE TABLE Title (  
  WORKER_REF_ID INT,  
  WORKER_TITLE VARCHAR(25),  
  AFFECTED_FROM DATETIME,  
  FOREIGN KEY(WORKER_REF_ID)  
  REFERENCES Worker(WORKER_ID)  
  ON DELETE CASCADE  
);
```




INSERTING VALUES :

```
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');
```

TO CHECK THE TABLE :
OUTPUT =>

```
SELECT * FROM TABLE_NAME ;
```

SELECT * FROM Title ;

Result Grid   Filter Rows: <input type="text" value="Search"/> Export: 				
	WORKER_REF_ID	WORKER_TITLE	AFFECTED_FROM	
▶	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	

- LET'S SOLVE THE QUESTIONS

SELECT * FROM Worker;




SELECT * FROM Bonus;

SELECT * FROM Title;

Q-1. Write an SQL query to fetch “FIRST_NAME” from Worker table using the alias name as <WORKER_NAME>.

Query :

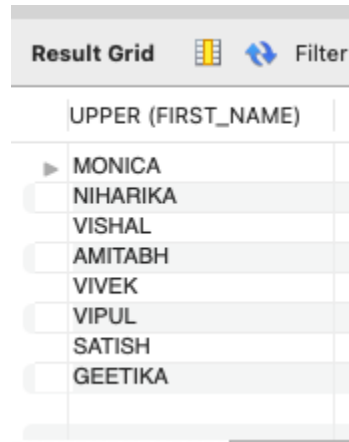
SELECT FIRST_NAME AS WORKER_NAME
FROM Worker;

Result Grid   Filter Rows: <input type="text" value="C"/>		
	WORKER_NAME	
▶	Monica	
◀	Niharika	
	Vishal	
◀	Amitabh	
	Vivek	
◀	Vipul	
	Satish	
◀	Geetika	
	Worker 153	
Action Output 		

Q-2. Write an SQL query to fetch “FIRST_NAME” from Worker table in upper case.

Query :

```
SELECT UPPER (FIRST_NAME)
FROM Worker;
```

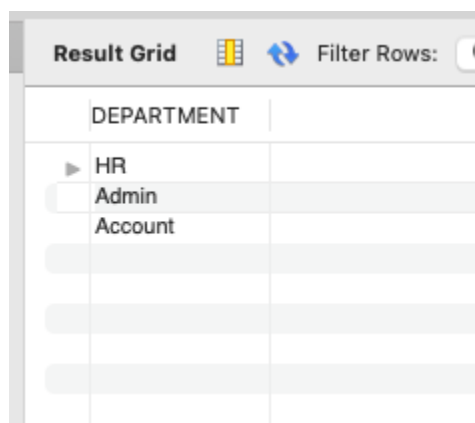


UPPER (FIRST_NAME)
MONICA
NIHARIKA
VISHAL
AMITABH
VIVEK
VIPUL
SATISH
GEETIKA

Q-3. Write an SQL query to fetch unique values of DEPARTMENT from Worker table.

Query :

```
SELECT DISTINCT DEPARTMENT
FROM Worker;
```

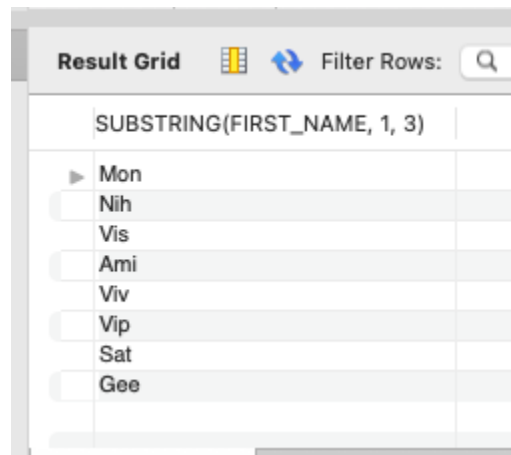


DEPARTMENT
HR
Admin
Account

Q-4. Write an SQL query to print the first three characters of FIRST_NAME from Worker table

Query:

```
SELECT SUBSTRING(FIRST_NAME, 1, 3)
FROM Worker;
```

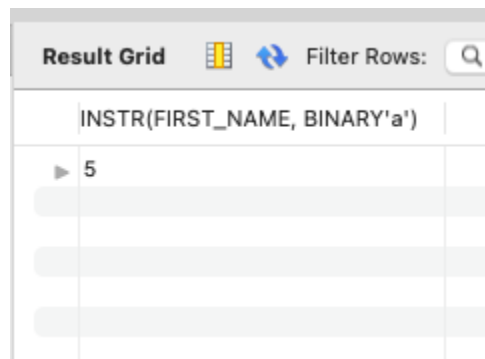


SUBSTRING(FIRST_NAME, 1, 3)
Mon
Nih
Vis
Ami
Viv
Vip
Sat
Gee

Q-5. Write an SQL query to find the position of the alphabet ('a') in the first name column 'Amitabh' from Worker table.

Query:

```
Select INSTR(FIRST_NAME, BINARY'a')
FROM Worker WHERE FIRST_NAME = 'Amitabh';
```



INSTR(FIRST_NAME, BINARY'a')
5

Q-6. Write an SQL query to print the FIRST_NAME from Worker table after removing white spaces from the right side.

Query :

```
SELECT RTRIM(FIRST_NAME)
FROM Worker;
```

Result Grid	
RTRIM(FIRST_NAME)	
▶	Monica
◀	Niharika
	Vishal
◀	Amitabh
	Vivek
◀	Vipul
	Satish
◀	Geetika

Q-7. Write an SQL query to print the DEPARTMENT from Worker table after removing white spaces from the left side.

Query :

```
SELECT LTRIM(DEPARTMENT)
FROM Worker;
```

LTRIM(DEPARTMENT)	
▶	HR
◀	Admin
	HR
◀	Admin
	Admin
◀	Account
	Account
◀	Admin

Q-8. Write an SQL query that fetches the unique values of DEPARTMENT from Worker table and prints its length.

Query :

```
SELECT DISTINCT(LENGTH(DEPARTMENT))  
FROM Worker;
```

(LENGTH(DEPARTMENT))	
▶ 2	
5	
7	

Q-9. Write an SQL query to print the FIRST_NAME from Worker table after replacing 'a' with 'A'.

Query :

```
SELECT REPLACE(FIRST_NAME, 'a', 'A')  
FROM Worker;
```

REPLACE(FIRST_NAME, 'a', 'A')	
▶ Monica	
NihArikA	
VishAl	
AmitAbh	
Vivek	
Vipul	
SAtish	
GeetikA	

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.

Query :

```
SELECT concat(FIRST_NAME, ' ', LAST_NAME) AS COMPLETE_NAME
FROM Worker;
```

COMPLETE_NAME
Monica_Arora
Niharika_Verma
Vishal_Singhal
Amitabh_Singh
Vivek_Bhati
Vipul_Diwan
Satish_Kumar
Geetika_Chauhan

Q-11. Write an SQL query to print all Worker details from the Worker table order by FIRST_NAME Ascending.

Query :

```
SELECT *
FROM Worker
ORDER BY FIRST_NAME ASC;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL
Worker 163					

Q-12. Write an SQL query to print all Worker details from the Worker table order by FIRST_NAME Ascending and DEPARTMENT Descending.

Query :

```
SELECT *
FROM Worker
ORDER BY FIRST_NAME ASC , DEPARTMENT DESC;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

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

Query :

```
SELECT *
FROM Worker
WHERE FIRST_NAME in ('Vipul','Satish');
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
NULL	NULL	NULL	NULL	NULL	NULL

Q-14. Write an SQL query to print details of workers excluding first names, “Vipul” and “Satish” from Worker table.

Query :

```
SELECT *  
FROM Worker  
WHERE NOT FIRST_NAME in ('Vipul', 'Satish');
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

Q-15. Write an SQL query to print details of Workers with DEPARTMENT name as “Admin”.

Query :

```
SELECT *  
FROM Worker  
WHERE DEPARTMENT = 'Admin';
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

Result Grid						
Filter Rows:		Search		Edit:		Export/Import:
WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT	
1	Monica	Arora	100000	2014-02-20 09:00:00	HR	
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin	
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin	
NULL	NULL	NULL	NULL	NULL	NULL	

Q-18. Write an SQL query to print details of the Workers whose FIRST_NAME ends with 'h' and contains six alphabets.

Query :

```
SELECT *  
FROM Worker  
WHERE LENGTH(FIRST_NAME) = 6  
and FIRST_NAME LIKE '%h';
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
NULL	NULL	NULL	NULL	NULL	NULL

Q-19. Write an SQL query to print details of the Workers whose SALARY lies between 100000 and 500000.

Query :

```
SELECT *  
FROM Worker  
WHERE SALARY BETWEEN 100000 and 500000;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
NULL	NULL	NULL	NULL	NULL	NULL

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

Query :

```
SELECT *  
FROM Worker  
WHERE JOINING_DATE = 2014-02-20 ;
```

Or

```
Select * from Worker where joined like '2014-02%';
```

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

Query:

```
SELECT count(DEPARTMENT)  
FROM Worker  
WHERE DEPARTMENT = 'Admin';
```

count(DEPARTMENT)
4

Q-22. Write an SQL query to fetch worker names with salaries ≥ 50000 and ≤ 100000 .

Query:

```
SELECT CONCAT(FIRST_NAME, '_', LAST_NAME) AS  
WORKER_NAMES, SALARY  
FROM Worker  
WHERE SALARY BETWEEN 50000 AND 100000;
```

	WORKER_NAMES	SALARY
▶	Monica_Arora	100000
	Niharika_Verma	80000
	Satish_Kumar	75000
	Geetika_Chauhan	90000

Q-23. Write an SQL query to fetch the no. of workers for each department in the descending order.

Query:

```
SELECT DEPARTMENT, COUNT(WORKER_ID)  
FROM Worker  
GROUP BY DEPARTMENT  
ORDER BY COUNT(DEPARTMENT) DESC;
```

	DEPARTMENT	COUNT(WORKER_ID)
▶	Admin	4
	HR	2
	Account	2

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

Query:

```
SELECT W.WORKER_ID,CONCAT(W.FIRST_NAME, ' ', W.LAST_NAME)
AS NAMES,W.SALARY, W.JOINING_DATE,W.DEPARTMENT,
T.WORKER_TITLE, T.AFFECTED_FROM
FROM Worker W
inner JOIN Title T
ON W.WORKER_ID = T.WORKER_REF_ID
WHERE WORKER_TITLE = 'Manager';
```

WORKER_ID	NAMES	SALARY	JOINING_DATE	DEPARTMENT	WORKER_TITLE	AFFECTED_FROM
1	Monica Arora	100000	2014-02-20 09:00:00	HR	Manager	2016-02-20 00:00:00
5	Vivek Bhati	500000	2014-06-11 09:00:00	Admin	Manager	2016-06-11 00:00:00

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

Query :

```
SELECT DEPARTMENT, JOINING_DATE, COUNT(*)
FROM Worker
GROUP BY DEPARTMENT, JOINING_DATE
HAVING COUNT(*) > 1;
```

-- OR.

```
SELECT WORKER_TITLE, AFFECTED_FROM, COUNT(*)
FROM Title
GROUP BY WORKER_TITLE, AFFECTED_FROM
HAVING COUNT(*) > 1;
```

-- OR


```

SELECT WORKER_REF_ID, BONUS_DATE, COUNT(*)
FROM Bonus
GROUP BY WORKER_REF_ID, BONUS_DATE
HAVING COUNT(*) > 1;

```

	DEPARTMENT	JOINING_DATE	COUNT(*)
▶	HR	2014-02-20 09:00:00	2
	Admin	2014-06-11 09:00:00	2

OR

	WORKER_TITLE	AFFECTED_FROM	COUNT(*)
▶	Executive	2016-06-11 00:00:00	3
	Lead	2016-06-11 00:00:00	2

OR

	WORKER_REF_ID	BONUS_DATE	COUNT(*)
▶	1	2016-02-20 00:00:00	2
	2	2016-06-11 00:00:00	2

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

Query:

```

Select * from Worker where WORKER_ID % 2 != 0;

```

OR

```

Select * from Bonus where WORKER_REF_ID % 2 != 0;

```

OR

```

Select * from Title where WORKER_REF_ID % 2 != 0;

```

	WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
▶	1	Monica	Arora	100000	2014-02-20 09:00:00	HR
	3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
	5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
	7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
	NULL	NULL	NULL	NULL	NULL	NULL

OR

	WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT
	1	2016-02-20 00:00:00	5000
	3	2016-02-20 00:00:00	4000
	1	2016-02-20 00:00:00	4500

OR

	WORKER_REF_ID	WORKER_TITLE	AFFECTED_FROM
▶	1	Manager	2016-02-20 00:00:00
	5	Manager	2016-06-11 00:00:00
	7	Executive	2016-06-11 00:00:00
	3	Lead	2016-06-11 00:00:00

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

Query :

Select * from Worker where WORKER_ID % 2 = 0;

OR

Select * from Bonus where WORKER_REF_ID % 2 = 0;

OR

Select * from Title where WORKER_REF_ID % 2 = 0;

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
▶ 2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

OR

WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT
▶ 2	2016-06-11 00:00:00	3000
2	2016-06-11 00:00:00	3500

OR

WORKER_REF_ID	WORKER_TITLE	AFFECTED_FROM
▶ 2	Executive	2016-06-11 00:00:00
8	Executive	2016-06-11 00:00:00
4	Asst. Manager	2016-06-11 00:00:00
6	Lead	2016-06-11 00:00:00

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

Query :

```
CREATE TABLE CLONE_TABLE SELECT * FROM Worker;
```

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

Query:

```
SELECT distinct *
FROM Worker W
INNER JOIN Bonus B
ON W.WORKER_ID = B.WORKER_REF_ID;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT	WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT
1	Monica	Arora	100000	2014-02-20 09:00:00	HR	1	2016-02-20 00:00:00	5000
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin	2	2016-06-11 00:00:00	3000
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR	3	2016-02-20 00:00:00	4000
1	Monica	Arora	100000	2014-02-20 09:00:00	HR	1	2016-02-20 00:00:00	4500
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin	2	2016-06-11 00:00:00	3500

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

Query :

```
SELECT distinct *
FROM Worker W
left JOIN Bonus B
ON W.WORKER_ID = B.WORKER_REF_ID
where B.WORKER_REF_ID is NULL;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT	WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin	NULL	NULL	NULL
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin	NULL	NULL	NULL
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account	NULL	NULL	NULL
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account	NULL	NULL	NULL
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin	NULL	NULL	NULL

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

Query :

```
SELECT CURDATE();
```

CURDATE()
2021-09-13

Q-32. Write an SQL query to show the top n (say 10) records of a table

Query :

```
SELECT * FROM Worker ORDER BY FIRST_NAME DESC LIMIT 10;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

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

Query :

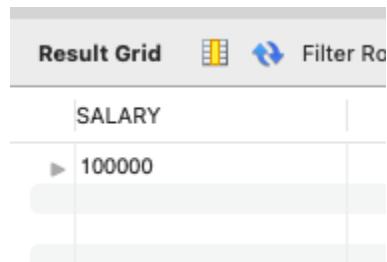
```
SELECT * FROM Worker ORDER BY SALARY DESC LIMIT 5;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
1	Monica	Arora	100000	2014-02-20 09:00:00	HR
NULL	NULL	NULL	NULL	NULL	NULL

Q-34. Write an SQL query to determine the 5th highest salary without using TOP or limit method.

Query :

```
SELECT SALARY
FROM Worker w1
where 4= (Select count(distinct(w2.salary))from Worker w2 where
w1.salary >=w2.salary);
```



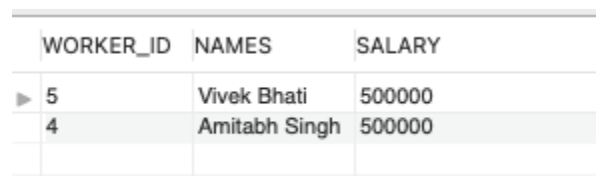
The screenshot shows a 'Result Grid' window with a 'Filter Row' button. The grid displays a single row with the column header 'SALARY' and the value '100000'.

SALARY
100000

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

Query :

```
Select DISTINCT W.WORKER_ID, CONCAT(W.FIRST_NAME, ' ',
W.LAST_NAME) AS NAMES, W.SALARY
from Worker W, Worker W1
where W.SALARY = W1.SALARY
and W.WORKER_ID != W1.WORKER_ID;
```



The screenshot shows a table with three columns: 'WORKER_ID', 'NAMES', and 'SALARY'. It contains two rows of data.

WORKER_ID	NAMES	SALARY
5	Vivek Bhati	500000
4	Amitabh Singh	500000

Q-36. Write an SQL query to show the second highest salary from a table.

Query :

```
SELECT MAX(SALARY) AS SALARY
FROM Worker
WHERE SALARY <> (SELECT MAX(SALARY)
FROM Worker);
```

SALARY
300000

Q-37. Write an SQL query to show one row twice in results from a table.

Query :

```
SELECT FIRST_NAME, SALARY
FROM Worker W
WHERE W.SALARY= 500000
UNION ALL
SELECT FIRST_NAME, SALARY
FROM Worker W1
WHERE W1.SALARY= 500000 ;
```

FIRST_NAME	SALARY
Amitabh	500000
Vivek	500000
Amitabh	500000
Vivek	500000

Q-38. Write an SQL query to fetch intersecting records of two tables

Query :

```
SELECT distinct *
FROM Worker W
INNER JOIN Bonus B
ON W.WORKER_ID = B.WORKER_REF_ID;
```

	WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT	WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT
▶	1	Monica	Arora	100000	2014-02-20 09:00:00	HR	1	2016-02-20 00:00:00	5000
	2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin	2	2016-06-11 00:00:00	3000
	3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR	3	2016-02-20 00:00:00	4000
	1	Monica	Arora	100000	2014-02-20 09:00:00	HR	1	2016-02-20 00:00:00	4500
	2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin	2	2016-06-11 00:00:00	3500

Q-39. Write an SQL query to fetch the first 50% records from a table

Query :

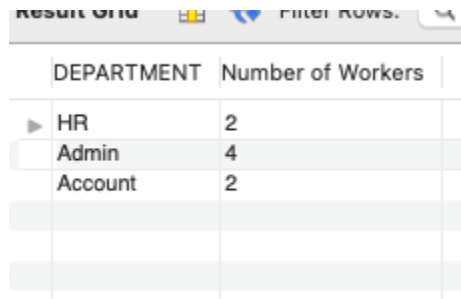
```
Select *
from worker
where worker_id <= (select round(count(worker_id)/2,0) from worker);
```

	WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
▶	1	Monica	Arora	100000	2014-02-20 09:00:00	HR
	2	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
	3	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
	4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
	NULL	NULL	NULL	NULL	NULL	NULL

Q-40. Write an SQL query to fetch the departments that have less than five people in it.

Query :

```
SELECT DEPARTMENT, COUNT(WORKER_ID) as 'Number of Workers'
FROM Worker
GROUP BY DEPARTMENT
HAVING COUNT(WORKER_ID) < 5;
```

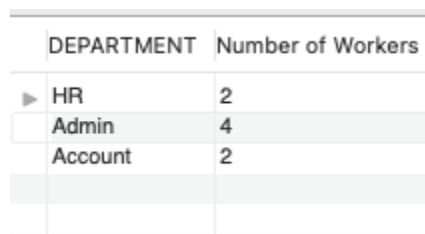


DEPARTMENT	Number of Workers
HR	2
Admin	4
Account	2

Q-41. Write an SQL query to show all departments along with the number of people in there.

Query :

```
SELECT DEPARTMENT, COUNT(WORKER_ID) as 'Number of Workers'
FROM Worker
GROUP BY DEPARTMENT
HAVING COUNT(WORKER_ID);
```



DEPARTMENT	Number of Workers
HR	2
Admin	4
Account	2

Q-42. Write an SQL query to show the last record from a table.

Query :

```
SELECT *
FROM Worker
ORDER BY WORKER_ID DESC LIMIT 1;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
▶ 8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

Q-43. Write an SQL query to fetch the first row of a table.

Query :

```
SELECT *
from Worker
ORDER BY WORKER_ID LIMIT 1;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
▶ 1	Monica	Arora	100000	2014-02-20 09:00:00	HR
NULL	NULL	NULL	NULL	NULL	NULL

Q-44. Write an SQL query to fetch the last five records from a table.

Query :

```
SELECT *
FROM Worker
ORDER BY DESC WORKER_ID LIMIT 5;
```

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
▶ 8	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin
7	Satish	Kumar	75000	2014-01-20 09:00:00	Account
6	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
5	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
4	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
NULL	NULL	NULL	NULL	NULL	NULL

Q-45. Write an SQL query to print the name of employees having the highest salary in each department.

Query :
?????

Q-46. Write an SQL query to fetch three max salaries from a table.

Query :

Select SALARY from Worker order by Salary desc limit 3;

SALARY
500000
500000
300000

Q-47. Write an SQL query to fetch three min salaries from a table.

Query :

SELECT SALARY FROM Worker ORDER BY SALARY LIMIT 3;

Result Grid		Filter Row
SALARY		
75000		
80000		
90000		

Q-48. Write an SQL query to fetch nth max salaries from a table.

Query :

```
SELECT distinct(SALARY)
FROM Worker
ORDER BY SALARY LIMIT 8;
```

SALARY
75000
80000
90000
100000
200000
300000
500000

Q-49. Write an SQL query to fetch departments along with the total salaries paid for each of them.

Query :

```
SELECT DEPARTMENT, SUM(SALARY)
FROM Worker
GROUP BY Department;
```

DEPARTMENT	SUM(SALARY)
HR	400000
Admin	1170000
Account	275000

Q-50. Write an SQL query to fetch the names of workers who earn the highest salary.

Query :

```
SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) AS NAMES ,SALARY  
FROM Worker  
where Salary=(select max(Salary) from Worker);
```

	NAMES	SALARY	
▶	Amitabh Singh	500000	
	Vivek Bhati	500000	