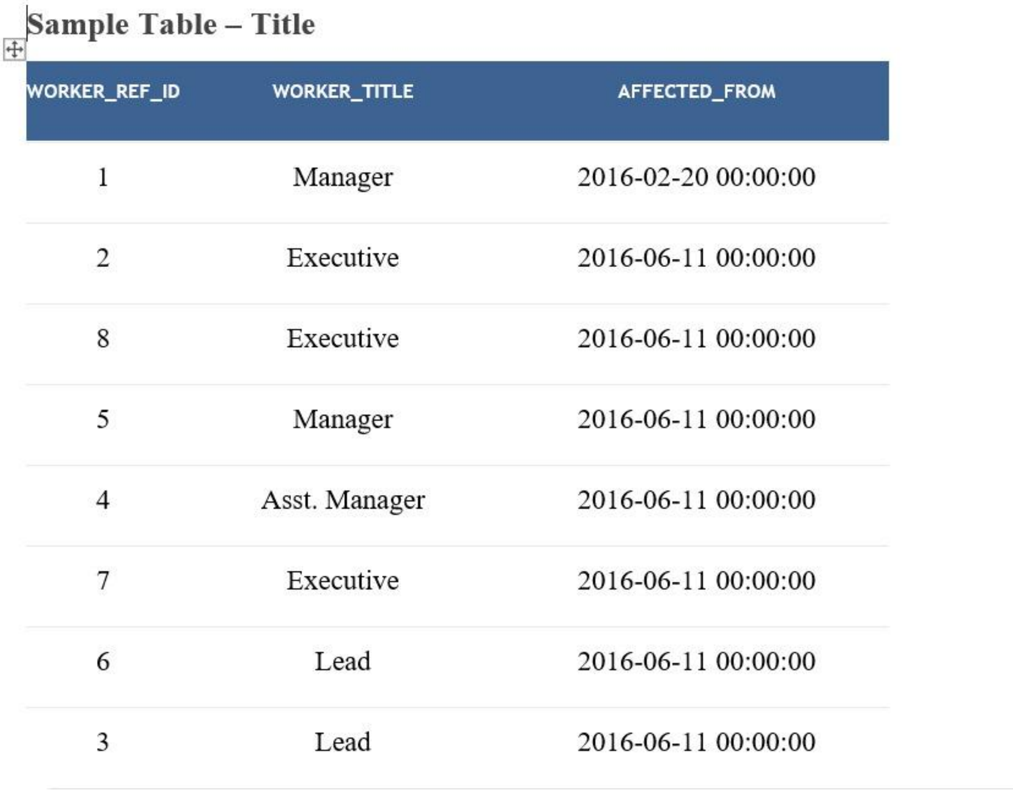
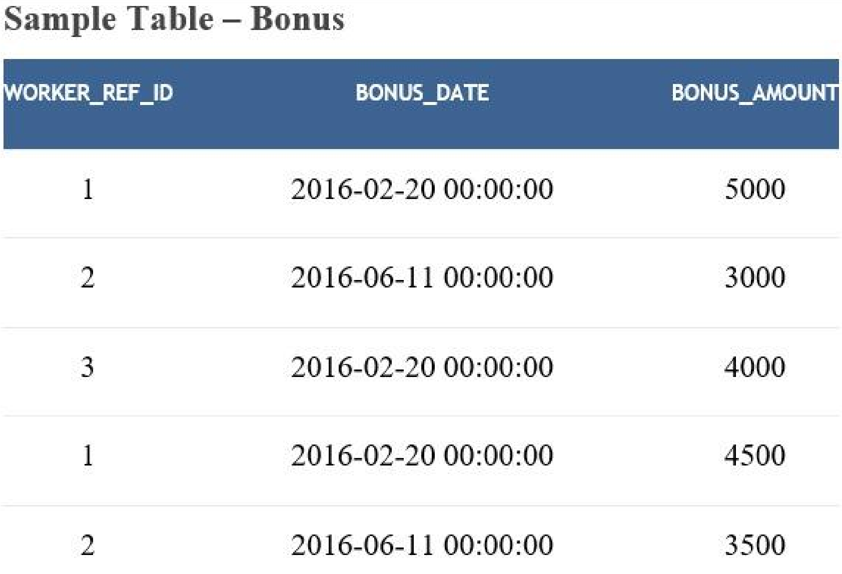
Mongo DB – Face Prep

avulapalli maneeshwar reddy



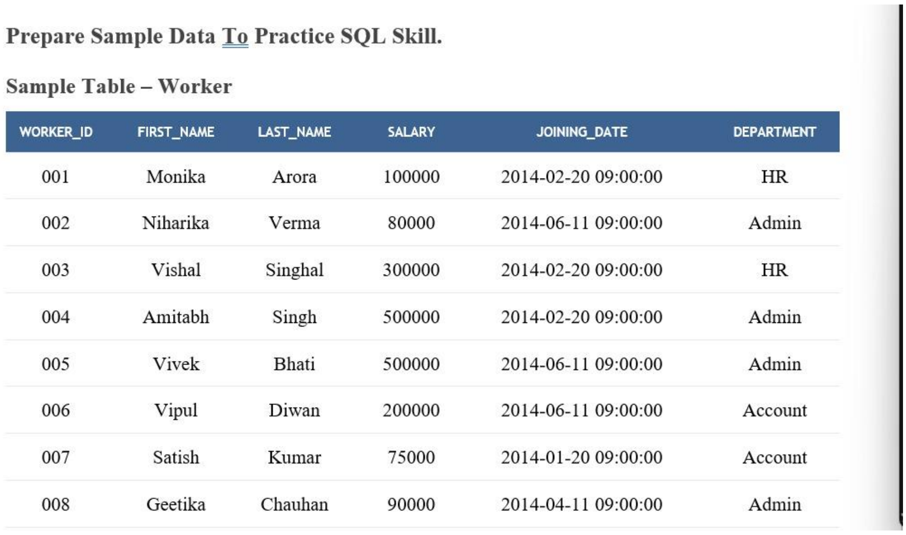
Sample Table – Title

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



Sample Table – Bonus

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



Prepare Sample Data To Practice SQL Skill.

Sample Table – Worker

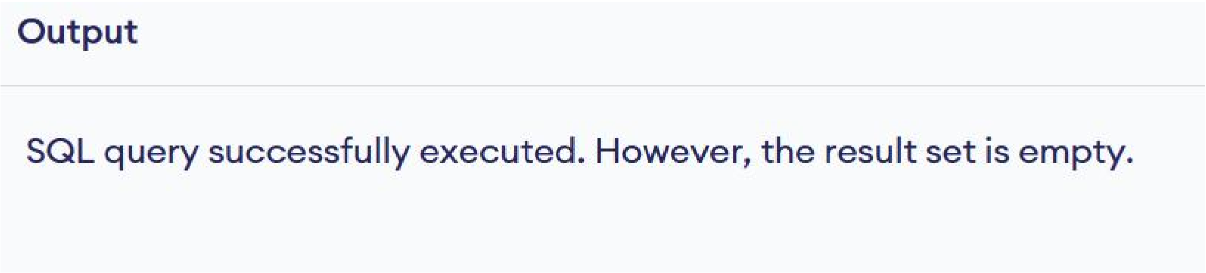
WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT
001	Monika	Arora	100000	2014-02-20 09:00:00	HR
002	Niharika	Verma	80000	2014-06-11 09:00:00	Admin
003	Vishal	Singhal	300000	2014-02-20 09:00:00	HR
004	Amitabh	Singh	500000	2014-02-20 09:00:00	Admin
005	Vivek	Bhati	500000	2014-06-11 09:00:00	Admin
006	Vipul	Diwan	200000	2014-06-11 09:00:00	Account
007	Satish	Kumar	75000	2014-01-20 09:00:00	Account
008	Geetika	Chauhan	90000	2014-04-11 09:00:00	Admin

Sql Code:

```
DROP TABLE IF EXISTS Worker;
DROP TABLE IF EXISTS Title;
DROP TABLE IF EXISTS Bonus;

CREATE TABLE Worker (
    WORKER_ID INT,
    FIRST_NAME VARCHAR(50),
    LAST_NAME VARCHAR(50),
    SALARY INT,
    JOINING_DATE DATETIME,
    DEPARTMENT VARCHAR(50)
);

INSERT INTO Worker VALUES
(001, 'Monika', 'Arora', 100000, '2014-02-20 09:00:00', 'HR'),
(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', 'Account'),
(008, 'Geetika', 'Chauhan', 90000, '2014-04-11 09:00:00', 'Admin');


CREATE TABLE Title (
    WORKER_REF_ID INT,
    WORKER_TITLE VARCHAR(50),
    AFFECTED_FROM DATETIME
);


INSERT INTO Title 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');


CREATE TABLE Bonus (
    WORKER_REF_ID INT,
    BONUS_DATE DATETIME,
    BONUS_AMOUNT INT
);
```

```
INSERT INTO Bonus 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);
```

Output:



Output

SQL query successfully executed. However, the result set is empty.

1) Write an SQL query to fetch unique values of DEPARTMENT from Worker table.

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

2) 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, DEPARTMENT DESC;
```

3). Write an SQL query to print details of the Workers whose FIRST_NAME contains ‘a’

```
SELECT *
FROM Worker
WHERE FIRST_NAME LIKE '%a%';
```

4. 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';
```

5. 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;
```

6. 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;
```

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

```
SELECT COUNT(*) AS Admin_Employee_Count
FROM Worker
WHERE DEPARTMENT = 'Admin';
```

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

```
SELECT FIRST_NAME, LAST_NAME
FROM Worker
WHERE SALARY BETWEEN 50000 AND 100000;
```

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

```
SELECT DEPARTMENT, COUNT(*) AS Worker_Count
FROM Worker
GROUP BY DEPARTMENT
ORDER BY Worker Count DESC;
```

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

```
SELECT W.*
FROM Worker W
JOIN Title T ON W.WORKER_ID = T.WORKER_REF_ID
WHERE T.WORKER_TITLE = 'Manager';
```

11. Write an SQL query to determine the 2nd lowest salary without using TOP or limit method.

```
SELECT MIN(SALARY) AS Second_Lowest_Salary
FROM Worker
WHERE SALARY > (
  SELECT MIN(SALARY)
  FROM Worker
);
```

12) Write an SQL query to fetch the list of employees with the same salary

```
SELECT *
FROM Worker
WHERE SALARY IN (

   SELECT SALARY

   FROM Worker

   GROUP BY SALARY

   HAVING COUNT(*) > 1

);
```

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

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

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

```
SELECT *
FROM Worker
WHERE WORKER_ID = 1
UNION ALL

SELECT *
FROM Worker
WHERE WORKER_ID = 1;
```

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

```
WITH RankedWorkers AS (
  SELECT *,

       ROW_NUMBER() OVER (ORDER BY WORKER_ID) AS rn,

       COUNT(*) OVER () AS total_count

  FROM Worker

)

SELECT *
FROM RankedWorkers
WHERE rn <= total_count / 2;
```

16. Write an SQL query to fetch the departments that have less than three people in it.

```
SELECT DEPARTMENT
FROM Worker
GROUP BY DEPARTMENT
HAVING COUNT(*) < 3;
```

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

```
SELECT DEPARTMENT, COUNT(*) AS NumOfWorkers
FROM Worker
GROUP BY DEPARTMENT;
```

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

```
SELECT *
FROM (

  SELECT *

  FROM Worker

  ORDER BY WORKER_ID DESC

  LIMIT 5

) AS LastFive
ORDER BY WORKER_ID;
```

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

```
SELECT FIRST_NAME, LAST_NAME, DEPARTMENT, SALARY
FROM Worker
WHERE SALARY = (

  SELECT MAX(SALARY)

  FROM Worker AS W2

  WHERE W2.DEPARTMENT = Worker.DEPARTMENT

);
```

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

```
SELECT DISTINCT SALARY
FROM Worker
ORDER BY SALARY DESC
LIMIT 3;
```

21. Write an SQL query to print the name of employees having the lowest salary in accunt and admin department.

```
SELECT FIRST_NAME, LAST_NAME, DEPARTMENT, SALARY
FROM Worker
WHERE DEPARTMENT IN ('Account', 'Admin')

 AND SALARY = (

  SELECT MIN(SALARY)

  FROM Worker AS W2

  WHERE W2.DEPARTMENT = Worker.DEPARTMENT

);
```