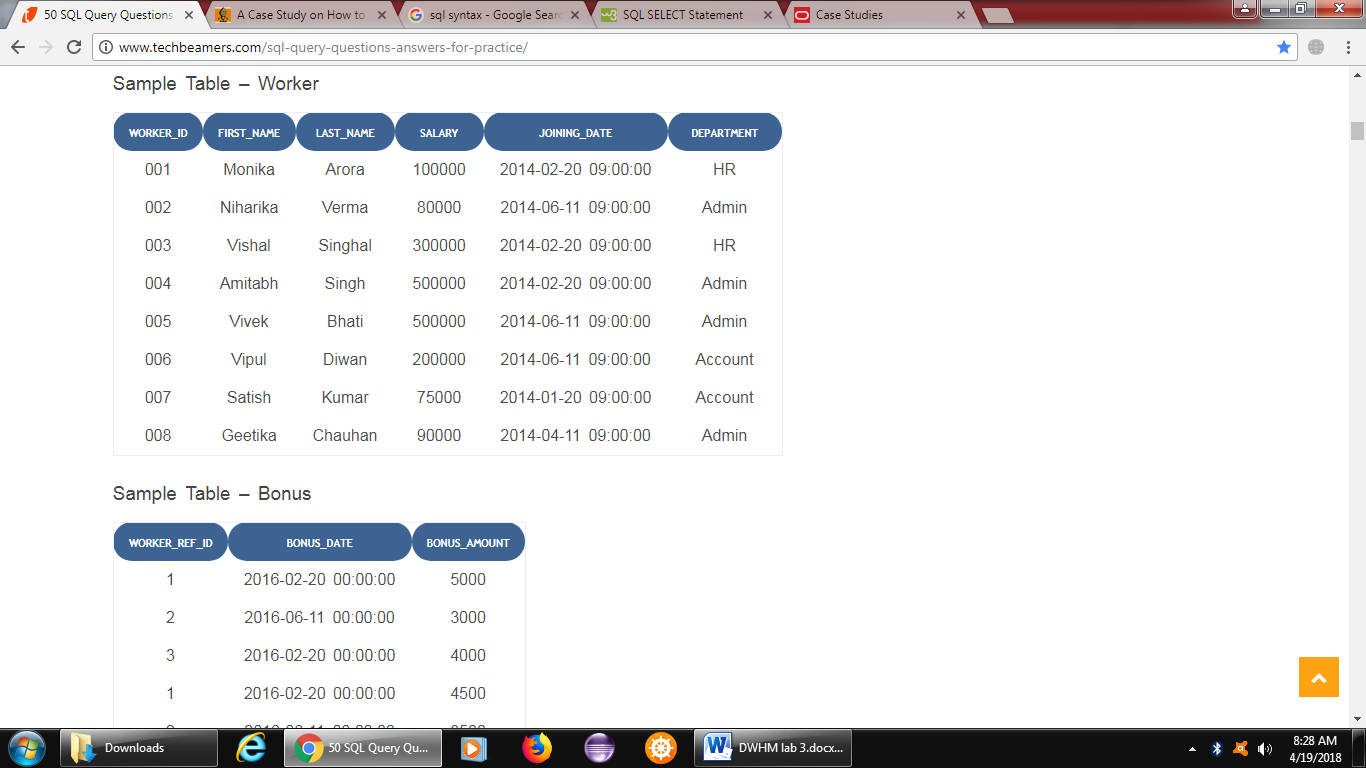
Lab 3





**Sample data can be created using the following sql script:**

CREATE DATABASE ORG;

SHOW DATABASES;

USE ORG;

CREATE TABLE Worker (

WORKER\_ID INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

FIRST\_NAME CHAR(25),

LAST\_NAME CHAR(25),

SALARY INT(15),

JOINING\_DATE DATETIME,

DEPARTMENT CHAR(25)

);

INSERT INTO Worker

(WORKER\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) VALUES

(001, 'Monika', 'Arora', 100000, '14-02-20 09.00.00', 'HR'),

(002, 'Niharika', 'Verma', 80000, '14-06-11 09.00.00', 'Admin'),

(003, 'Vishal', 'Singhal', 300000, '14-02-20 09.00.00', 'HR'),

(004, 'Amitabh', 'Singh', 500000, '14-02-20 09.00.00', 'Admin'),

(005, 'Vivek', 'Bhati', 500000, '14-06-11 09.00.00', 'Admin'),

(006, 'Vipul', 'Diwan', 200000, '14-06-11 09.00.00', 'Account'),

(007, 'Satish', 'Kumar', 75000, '14-01-20 09.00.00', 'Account'),

(008, 'Geetika', 'Chauhan', 90000, '14-04-11 09.00.00', 'Admin');

CREATE TABLE Bonus (

WORKER\_REF\_ID INT,

BONUS\_AMOUNT INT(10),

BONUS\_DATE DATETIME,

FOREIGN KEY (WORKER\_REF\_ID)

REFERENCES Worker(WORKER\_ID)

ON DELETE CASCADE

);

INSERT INTO Bonus

(WORKER\_REF\_ID, BONUS\_AMOUNT, BONUS\_DATE) VALUES

(001, 5000, '16-02-20'),

(002, 3000, '16-06-11'),

(003, 4000, '16-02-20'),

(001, 4500, '16-02-20'),

(002, 3500, '16-06-11');

CREATE TABLE Title (

WORKER\_REF\_ID INT,

WORKER\_TITLE CHAR(25),

AFFECTED\_FROM DATETIME,

FOREIGN KEY (WORKER\_REF\_ID)

REFERENCES Worker(WORKER\_ID)

ON DELETE CASCADE

);

INSERT INTO Title

(WORKER\_REF\_ID, WORKER\_TITLE, AFFECTED\_FROM) VALUES

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

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

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

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

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

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

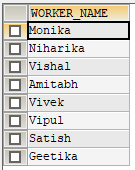
(006, 'Lead', '2016-06-11 00:00:00'),

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

**SQL QUERIES:**

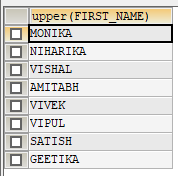
1. Write An SQL Query To Fetch “FIRST\_NAME” From Worker Table Using The Alias Name As <WORKER\_NAME>.

**Solution:** Select FIRST\_NAME AS WORKER\_NAME from Worker;



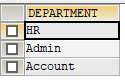
1. Write An SQL Query To Fetch “FIRST\_NAME” From Worker Table In Upper Case.

**Solution:** SELECT UPPER(FIRST\_NAME) FROM worker;



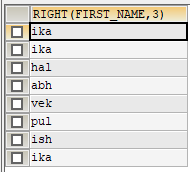
1. Write An SQL Query To Fetch Unique Values Of DEPARTMENT From Worker Table.

**Solution:** SELECT DISTINCT DEPARTMENT FROM worker;



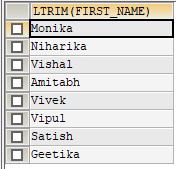
1. Write An SQL Query To Print Last Three Characters Of  FIRST\_NAME From Worker Table.

**Solution:** SELECT RIGHT(FIRST\_NAME,3) FROM worker;



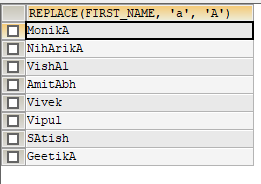
1. Write An SQL Query To Print The FIRST\_NAME From Worker Table After Removing White Spaces From The Left Side.

**Solution:** SELECT LTRIM(FIRST\_NAME) FROM worker;



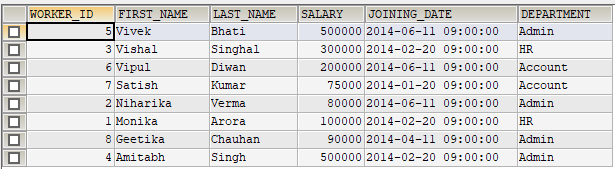
1. Write An SQL Query To Print The FIRST\_NAME From Worker Table After Replacing ‘a’ With ‘A’.

**Solution:** SELECT REPLACE(FIRST\_NAME, 'a', 'A') FROM worker;



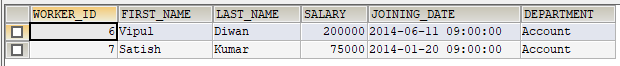
1. Write An SQL Query To Print All Worker Details From The Worker Table Order By FIRST\_NAME Descending.

**Solution:** SELECT \* FROM worker ORDER BY FIRST\_NAME DESC;



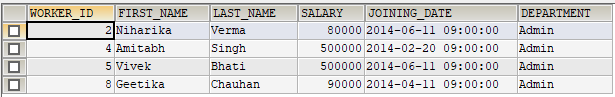
1. Write An SQL Query To Print Details For Workers With The First Name As “Vipul” And “Satish” From Worker Table.

**Solution:** SELECT \* FROM worker WHERE FIRST\_NAME IN('Vipul','Satish');



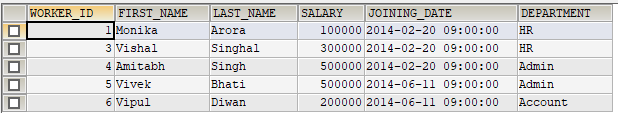
1. Write An SQL Query To Print Details Of Workers With DEPARTMENT Name As “Admin”.

**Solution:** SELECT \* FROM worker WHERE DEPARTMENT = 'Admin';



1. Write An SQL Query To Print Details Of The Workers Whose SALARY Lies Between 100000 And 500000.

**Solution:** SELECT \* FROM worker WHERE SALARY BETWEEN 100000 AND 500000;



1. Write An SQL Query To Print Details Of The Workers Who Have Joined In Jan’2014.

**Solution:** SELECT \* FROM worker WHERE JOINING\_DATE LIKE '2014-01%';



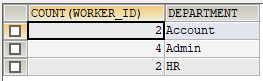
1. Write An SQL Query To Fetch The Count Of Employees Working In The Department ‘Admin’.

**Solution:** SELECT COUNT(WORKER\_ID)FROM worker WHERE DEPARTMENT = "Admin";



1. Write An SQL Query To Fetch The No. Of Workers For Each Department In The Ascending Order.

**Solution:** SELECT COUNT(WORKER\_ID), DEPARTMENT FROM worker GROUP BY DEPARTMENT ASC;



1. Write An SQL Query To Fetch The List Of Employees With The Same Salary.

**Solution:** SELECT w.FIRST\_NAME, w.SALARY FROM worker w INNER JOIN worker v ON v.salary=w.salary WHERE w.worker\_id!=v.worker\_id;



1. Write An SQL Query To Show The Second Highest Salary From A Table.

**Solution:** SELECT MAX(SALARY) FROM worker WHERE SALARY< ( SELECT MAX(SALARY) FROM worker);



|  |
| --- |
| Lab Tasks |
| Submission Date: 26-04-18 |

Execute OLTP queries for the following scenario:

*An organization has a main data table that lists events. Each event has a series of fields including duration, reason code and status.*

*The OLTP system has a table of statuses with attributes status code and Name and there is a common field between the main table (status code), which allows the organization to print the status name rather than the code.*

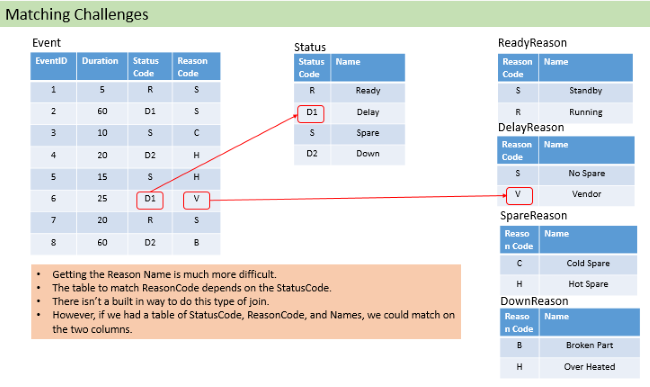
*There are four statuses (ready, delay, spare and down) and there are a range of reasons associated with each status with reason codes. It is possible for the same reason code to exist in two statuses (e.g. delay maintenance and down maintenance).*

*For some reason, the source database has four tables of reasons (one for each status). Each of these has a field which links to the main reason code.*

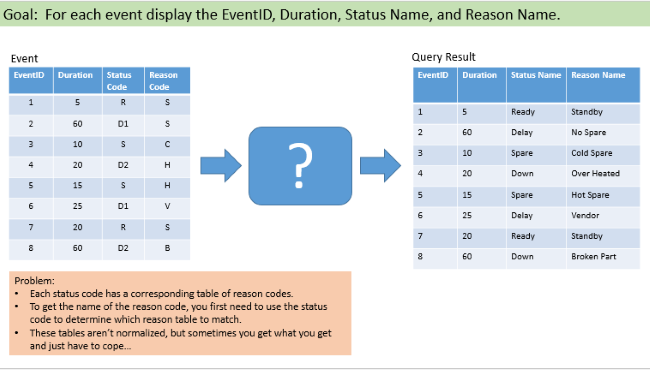
*The organization wishes to do the following:*

*When the main table record status is ‘delay’, then retrieve the name in the ‘delay’ table which corresponds to the main table delay code. When the status is ‘down’, do the same thing but look up the name in the ‘down’ table.*

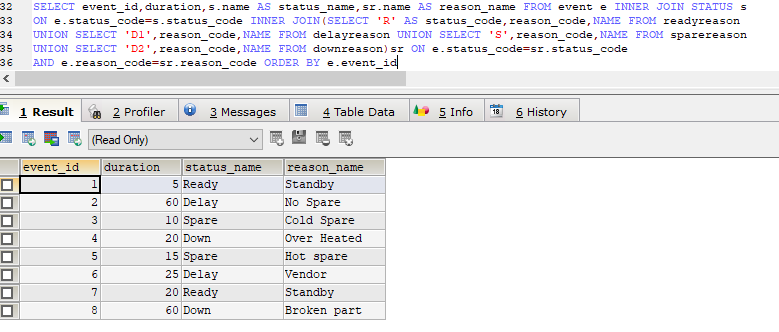
**Sample screenshots:**



Query output should be:



**Hint: Look regarding joins and union in sql.**

**Solution: **