SQL Server

Test of Sql Server

Jyot Hathi

2021

Index

[**A.** **Given Reference Table Creation and Data Insertion** 3](#_Toc64718258)

[**1.** **Employee Table:** 3](#_Toc64718259)

[**2.** **Department Table (To Normalize Employee Table):** 4](#_Toc64718260)

[**3.** **Incentive Table:** 4](#_Toc64718261)

[**B.** **Execution of Give Queries:** 5](#_Toc64718262)

[**1.** **Get First\_Name from employee table using alias name “Employee Name”** 5](#_Toc64718263)

[**2.** **Get position of 'o' in name 'John' from employee table** 5](#_Toc64718264)

[**3.** **Get FIRST\_NAME ,Joining year,Joining Month and Joining Date from employee table** 5](#_Toc64718265)

[**4.** **Get all employee details from the employee table order by First\_Name Ascending and Salary descending.** 6](#_Toc64718266)

[**5.** **Get employee details from employee table whose employee name are not “John” and “Roy”.** 6](#_Toc64718267)

[**6.** **Get employee details from employee table whose first name ends with 'n'.** 6](#_Toc64718268)

[**7.** **Get employee details from employee table whose first name ends with 'n' and name contains 4 letters.** 7](#_Toc64718269)

[**8.** **Get employee details from employee table whose Salary less than 800000** 7](#_Toc64718270)

[**9.** **Get employee details from employee table who joined before January 1st 2013** 7](#_Toc64718271)

[**10.** **Get difference between JOINING\_DATE and INCENTIVE\_DATE from employee and incentives table** 8](#_Toc64718272)

[**11.** **Print database date.** 8](#_Toc64718273)

[**12.** **Get department,total salary with respect to a department from employee table.** 8](#_Toc64718274)

[**13.** **Get department,no of employees in a department,total salary with respect to a department from employee table order by total salary descending .** 9](#_Toc64718275)

[**14.** **Select no of employees joined with respect to year and month from employee table.** 9](#_Toc64718276)

[**15.** **Update incentive table with employee's Incentive\_amount as '12000' where employee name is 'John'** 9](#_Toc64718277)

[**16.** **Select TOP 2 salary from employee table** 10](#_Toc64718278)

[**17.** **Select 2nd Highest salary from employee table** 10](#_Toc64718279)

[**18.** **Write. What is the difference between UNION and UNION ALL?** 10](#_Toc64718280)

[**19.** **Write a syntax for CREATE Employee Table.** 10](#_Toc64718281)

[**20.** **Write a syntax for truncate all data from Emplyee Table.** 11](#_Toc64718282)

[**21.** **Write a syntax for CREATE Procedure to display the Employee details by passing the “Employee Id” in the procedure.** 11](#_Toc64718283)

[**22.** **Write a syntax for CREATE SQL function, which accept three number as argument and return the highest number.** 11](#_Toc64718284)

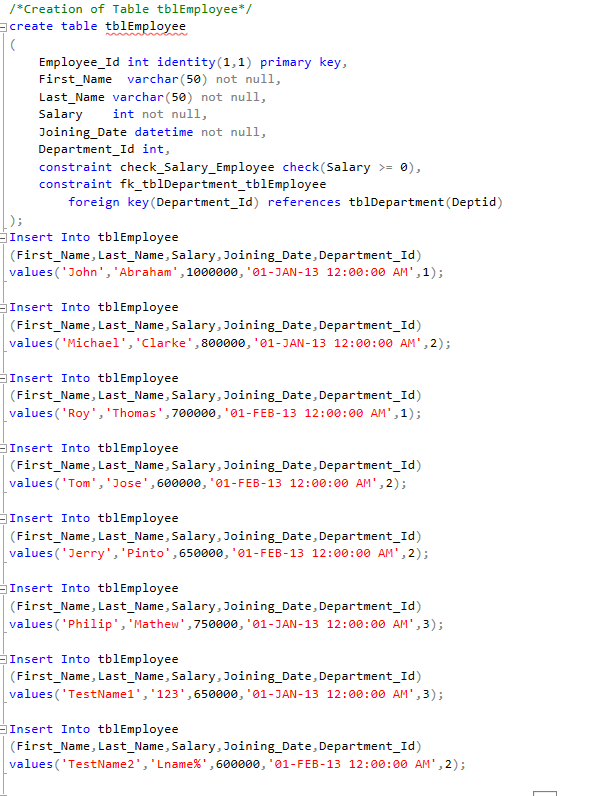
[**23.** **Write a syntax for Update the Employee's salary whose department is “Insurance”.** 12](#_Toc64718285)

[**24. State the difference between varchar and nvarchar.** 12](#_Toc64718286)

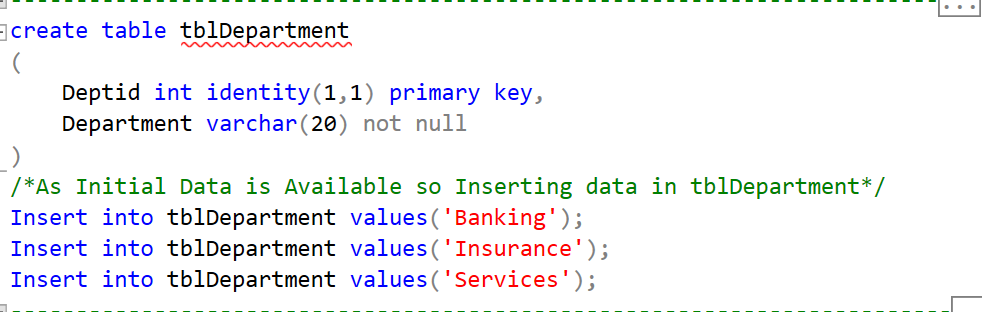
[**25. Write a query that insert the data into Employee table, data as mentioned. {First name : 'Critiano' , Last name : 'Ronaldo' , Salary : '30000' , Joining Date : '01-FEB-13 12.00.00 AM' , Department : 'Banking' }** 12](#_Toc64718287)

# **Given Reference Table Creation and Data Insertion**

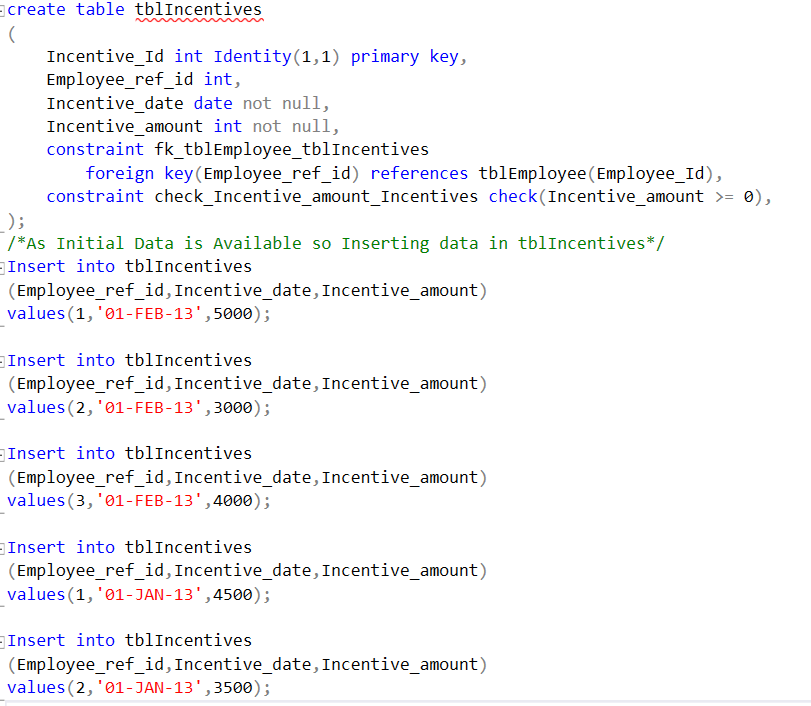
## **Employee Table:**



## **Department Table (To Normalize Employee Table):**



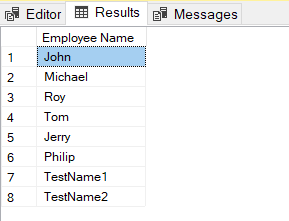
## **Incentive Table:**



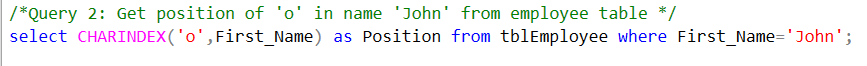
# **Execution of Give Queries:**

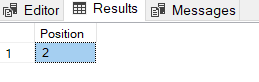
## **Get First\_Name from employee table using alias name “Employee Name”**



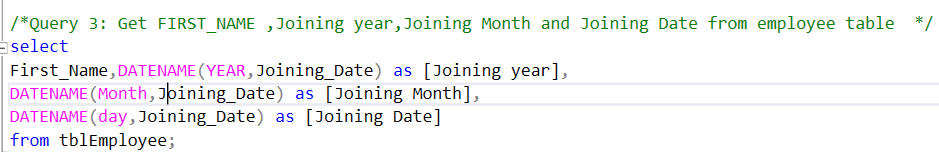


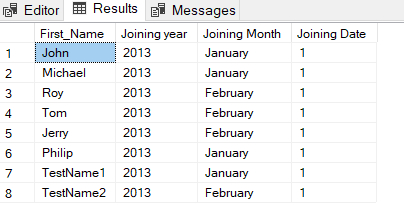
## **Get position of 'o' in name 'John' from employee table**



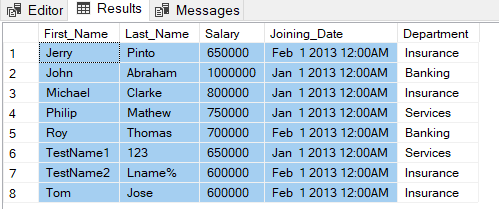
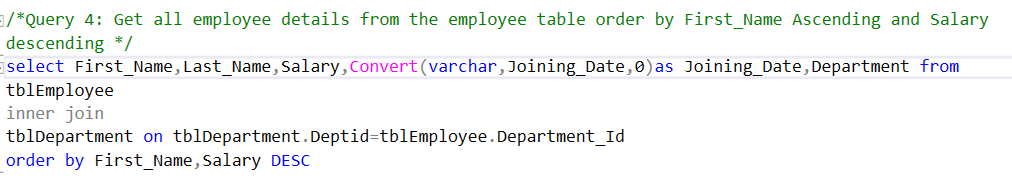


## **Get FIRST\_NAME ,Joining year,Joining Month and Joining Date from employee table**

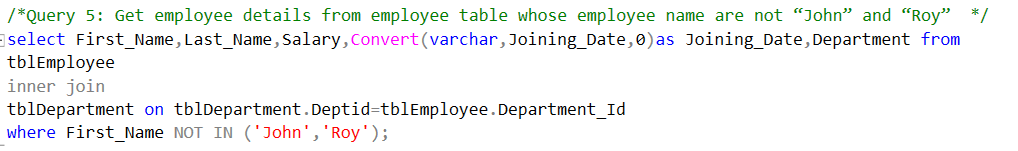


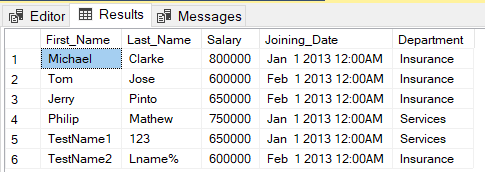


## **Get all employee details from the employee table order by First\_Name Ascending and Salary descending.**

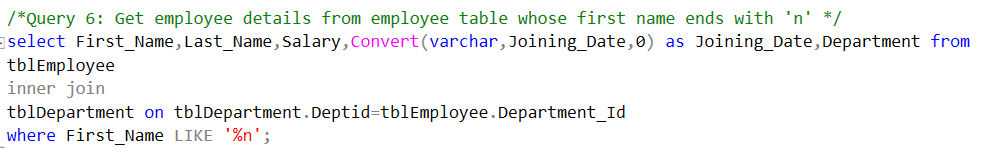


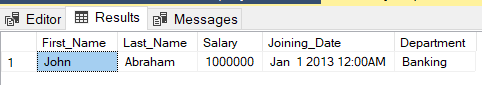
## **Get employee details from employee table whose employee name are not “John” and “Roy”.**



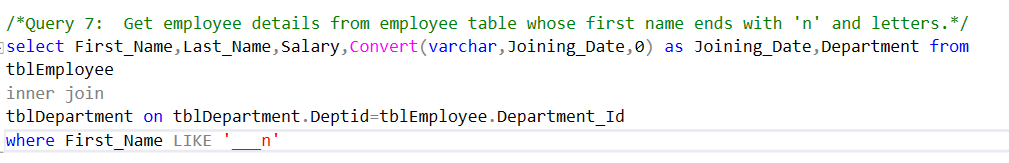


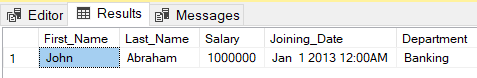
## **Get employee details from employee table whose first name ends with 'n'.**



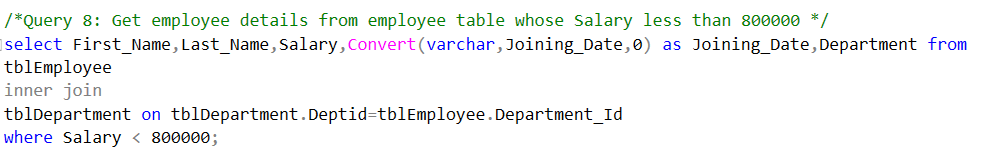


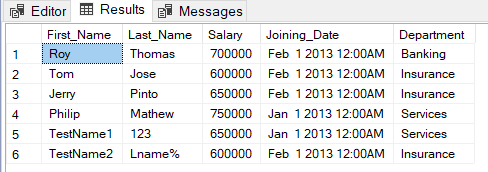
## **Get employee details from employee table whose first name ends with 'n' and name contains 4 letters.**



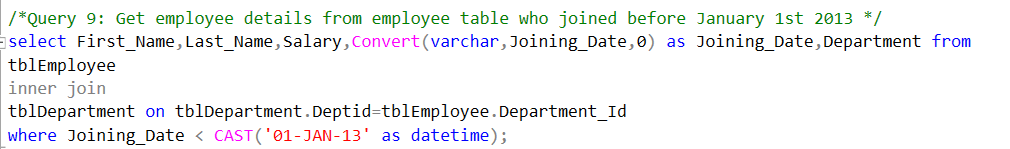


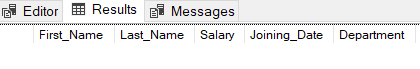
## **Get employee details from employee table whose Salary less than 800000**



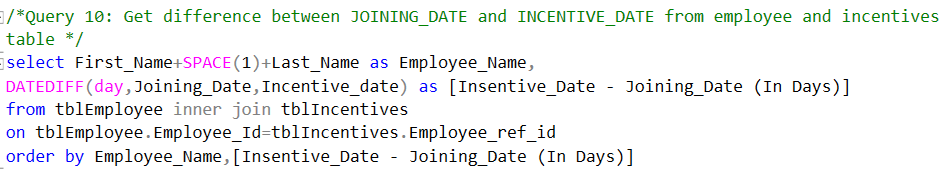


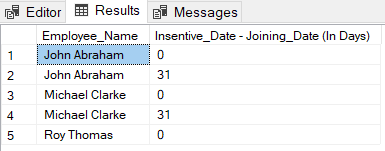
## **Get employee details from employee table who joined before January 1st 2013**



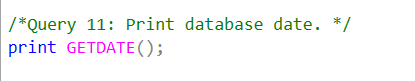


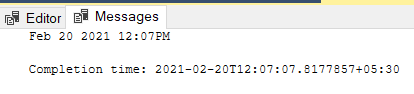
## **Get difference between JOINING\_DATE and INCENTIVE\_DATE from employee and incentives table**



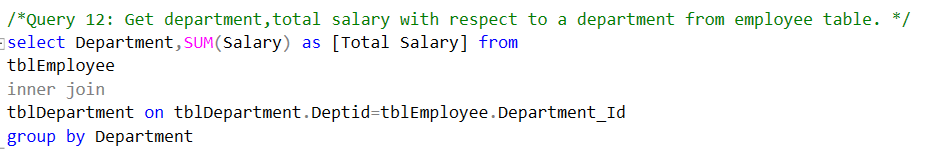


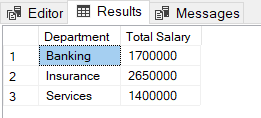
## **Print database date.**



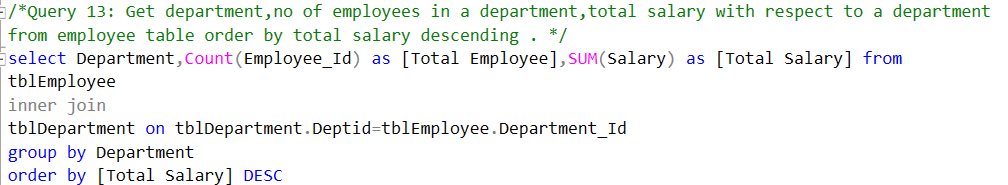


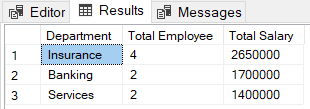
## **Get department,total salary with respect to a department from employee table.**



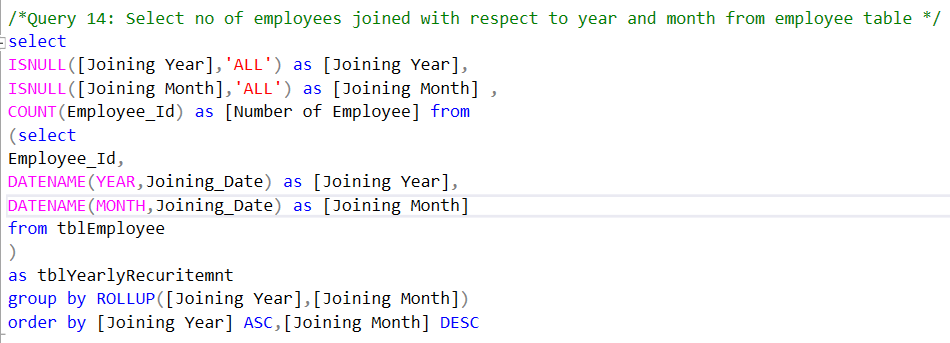


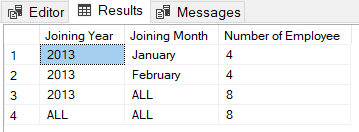
## **Get department,no of employees in a department,total salary with respect to a department from employee table order by total salary descending .**



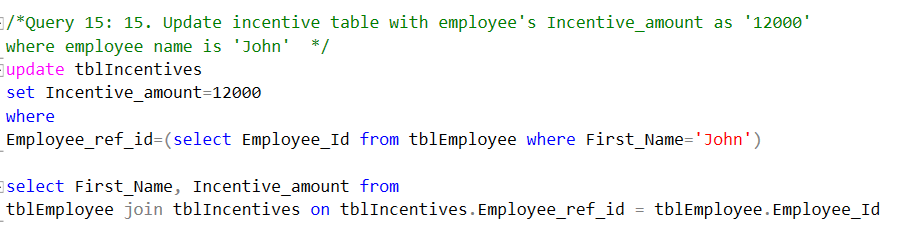


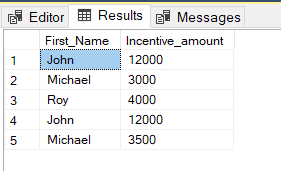
## **Select no of employees joined with respect to year and month from employee table.**



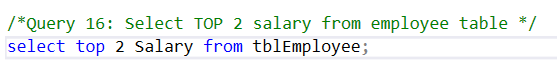


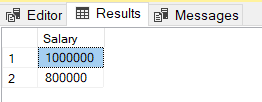
## **Update incentive table with employee's Incentive\_amount as '12000' where employee name is 'John'**



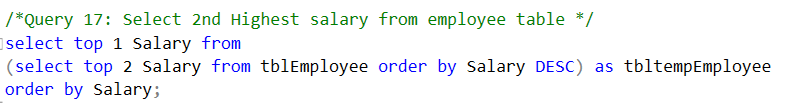


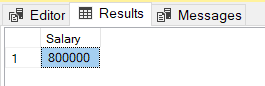
## **Select TOP 2 salary from employee table**





## **Select 2nd Highest salary from employee table**





## **Write. What is the difference between UNION and UNION ALL?**

Ans:

1. Union All Not Remove the Duplication of Records
2. Whereas UNION First Remove Duplication and Then Display Data.

## **Write a syntax for CREATE Employee Table.**

Ans:

Create table tblEmployee

(

EmployeeId int Identity(1,1) primary key,

First\_Name varchar(50) not null,

Last Name varchar(50) not null,

Salary int not null,

Joining\_Date datetime not null,

Department varchar(50),

Constraint ck\_Salary\_tblEmployee CHECK(Salary >=0)

);

**Note**: Here We Can Normalize the table By Create Department Table and then have to set reference of that to employee table instead of Department Manually.

## **Write a syntax for truncate all data from Emplyee Table.**

Ans.

TRUNCATE table tblEmployee;

## **Write a syntax for CREATE Procedure to display the Employee details by passing the “Employee Id” in the procedure.**

Ans.

Create proc sp\_tblEmployee

@EmployeeId int =0

As

Begin

Select \* from tblEmployee where [Employee Id] = @ EmployeeId;

End

## **Write a syntax for CREATE SQL function, which accept three number as argument and return the highest number.**

Ans.

Create function fc\_HighestNumber(@number1 int, @number2 int ,@number3 int)

Returns int

As

Declare @highest int;

Begin

If @number1 > @number2 AND @number1 > @number3

Set @ highest= @number1;

Else if @number2 > @number3

Set @ highest= @number2;

Else

Set @ highest= @number3;

Return @ highest;

End

## **Write a syntax for Update the Employee's salary whose department is “Insurance”.**

Ans:

1. If Employee Is Not Normalized by creating Separate Department table

Update tblEmployee

Set Salary=<new Salary>

Where Department = ‘Insurance’

1. If Employee Is Normalized by creating Separate Department table

Update tblEmployee

Set Salary=<new Salary>

Where DepartmentId = (select DeptId from tblDepartment where Department = ‘Insurance’);

## **State the difference between varchar and nvarchar.**

Ans:

1. varchar store data as 8-bit representation so it requires less size to store
2. nvarchar store data as Unicode Format (16-Bit Representation) so It Take More Amount of storage in compare of varchar.

## **25. Write a query that insert the data into Employee table, data as mentioned. {First name : 'Critiano' , Last name : 'Ronaldo' , Salary : '30000' , Joining Date : '01-FEB-13 12.00.00 AM' , Department : 'Banking' }**

Ans:

1. If Employee Is Not Normalized by creating Separate Department table

Insert into tblEmployee

(First\_Name,Last\_Name,Salary,Joining\_Date,Department)

Values('Critiano', 'Ronaldo', 30000, 01-FEB-13 12:00:00 AM, 'Banking');

1. If Employee Is Normalized by creating Separate Department table

Insert into tblEmployee

(First\_Name,Last\_Name,Salary,Joining\_Date,Department\_Id)

Values('Critiano', 'Ronaldo', 30000, 01-FEB-13 12:00:00 AM,1);

**Note**: Here 1 is Department id from tblDepartment whose Department Value is Banking