|  |  |
| --- | --- |
| **Cognizant Digital Nurture 4.0: Deep Skilling** | |
| Name: Rajatesh Paul | Superset ID: 6365356 |
| Week: 02 | Advanced SQL (Exercise: Functions) |

**Exercise 1:** Create a Scalar Function

Goal: Create a scalar function to calculate the annual salary of an employee.

Steps:

1. Define a scalar function named `fn\_CalculateAnnualSalary`.

2. The function should take `Salary` as input and return `Salary \* 12`.

3. Test the function by selecting the annual salary for each employee.

**Code:**

CREATE FUNCTION fn\_CalculateAnnualSalary (@Salary DECIMAL(10,2))

RETURNS DECIMAL(10,2)

AS

BEGIN

RETURN @Salary \* 12;

END;

GO -- Ends the batch

-- Test the function

SELECT

EmployeeID,

FirstName,

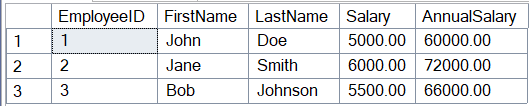
LastName,

Salary,

dbo.fn\_CalculateAnnualSalary(Salary) AS AnnualSalary

FROM Employees;

**Output:**

****

**Exercise 2:** Create a Table-Valued Function

Goal: Create a table-valued function to return employees in a specific department.

Steps:

1. Define a table-valued function named `fn\_GetEmployeesByDepartment`.

2. The function should take `DepartmentID` as input and return a table with employee

details.

3. Test the function by selecting employees from the IT department.

**Code:**

CREATE FUNCTION fn\_GetEmployeesByDepartment (@DeptID INT)

RETURNS TABLE

AS

RETURN

SELECT

E.EmployeeID,

E.FirstName,

E.LastName,

E.Salary,

E.JoinDate

FROM Employees E

WHERE E.DepartmentID = @DeptID;

GO -- Ends batch

SELECT \* FROM fn\_GetEmployeesByDepartment(2);

**Output:**

****

**Exercise 3:** Create a User-Defined Function

Goal: Create a user-defined function to calculate the bonus for an employee.

Steps:

1. Define a user-defined function named `fn\_CalculateBonus`.

2. The function should take `Salary` as input and return `Salary \* 0.10`.

3. Test the function by selecting the bonus for each employee.

**Code:**

CREATE FUNCTION fn\_CalculateBonus (@Salary DECIMAL(10,2))

RETURNS DECIMAL(10,2)

AS

BEGIN

RETURN @Salary \* 0.10;

END;

GO -- Ends the function creation batch

-- Test the function separately

SELECT

EmployeeID,

FirstName,

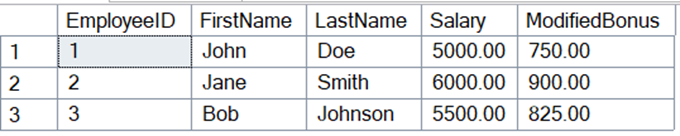
LastName,

Salary,

dbo.fn\_CalculateBonus(Salary) AS Bonus

FROM Employees;

**Output:**

****

**Exercise 4:** Modify a User-Defined Function

Goal: Modify the `fn\_CalculateBonus` function to return `Salary \* 0.15`.

Steps:

1. Alter the `fn\_CalculateBonus` function to return `Salary \* 0.15`.

2. Test the modified function by selecting the bonus for each employee.

DROP FUNCTION IF EXISTS fn\_CalculateBonus;

GO

CREATE FUNCTION fn\_CalculateBonus (@Salary DECIMAL(10,2))

RETURNS DECIMAL(10,2)

AS

BEGIN

RETURN @Salary \* 0.15;

END;

GO

-- Step 3: Test the modified function

SELECT

EmployeeID,

FirstName,

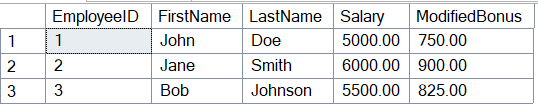
LastName,

Salary,

dbo.fn\_CalculateBonus(Salary) AS ModifiedBonus

FROM Employees;

**Output:**



**Exercise 5:** Delete a User-Defined Function

Goal: Delete the `fn\_CalculateBonus` function.

Steps:

1. Drop the `fn\_CalculateBonus` function.

2. Verify that the function has been deleted.Code:

**Code:**

DROP FUNCTION IF EXISTS fn\_CalculateBonus;

GO

CREATE FUNCTION fn\_CalculateBonus (@Salary DECIMAL(10,2))

RETURNS DECIMAL(10,2)

AS

BEGIN

RETURN @Salary \* 0.15;

END;

GO

SELECT

EmployeeID,

FirstName,

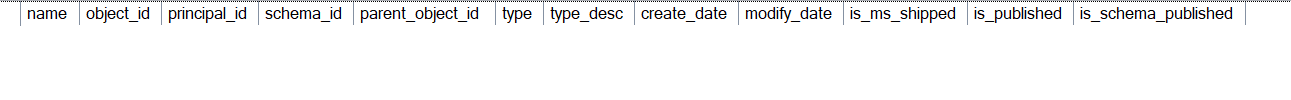
LastName,

Salary,

dbo.fn\_CalculateBonus(Salary) AS ModifiedBonus

FROM Employees;

**Output:**

****