**Домашняя работа №11**

Создать таблицу с количеством использования функций для дз 3. Используя data\_for\_merge.csv, заполнить ее через **оператор merge.** Итоговая таблица с результатами должна иметь вид

|  |  |
| --- | --- |
| Function\_name | Function\_count |
| CASE | 4 |
| CAST | 2 |
| CONCAT | 7 |

1. **ВАРИАНТ РЕШЕНИЯ**

--1. Создаём таблицу в БД, с временной так не получится

CREATE TABLE [dbo].[data\_for\_merge]

(

[IrinaPashkovich] NVARCHAR (MAX) NULL,

[Darya] NVARCHAR (MAX) NULL,

[Ekaterina] NVARCHAR (MAX) NULL,

[IrinaSelyutina] NVARCHAR (MAX) NULL,

[Viktor] NVARCHAR (MAX) NULL,

[Dmitriy] NVARCHAR (MAX) NULL,

[Svetlana] NVARCHAR (MAX) NULL,

[VikaVoronchuk] NVARCHAR (MAX) NULL,

[VikaOstopchuk] NVARCHAR (MAX) NULL

)

--2. Вставляем данные

BULK INSERT [dbo].[data\_for\_merge] --DROP TABLE #TableCSV3

FROM 'C:\1\data\_for\_merge.csv'

WITH

( FIRSTROW = 2, FIELDTERMINATOR =';' , ROWTERMINATOR ='\n', KEEPNULLS);

--3.Временная таблица для MERGE

--target

CREATE TABLE #TableInput --DROP TABLE #TableCSV

(

[Function\_name] NVARCHAR (MAX) NULL,

[Function\_count] INT

)

--source

CREATE TABLE #TempColumn

(

[Function\_name] NVARCHAR (MAX) NULL

)

--4. Необходимые переменные

DECLARE @Table NVARCHAR(20) = N'data\_for\_merge' --таблица csv

DECLARE @Column nvarchar(100) = N'' --переменная для хранения имени столбца

DECLARE @number INT -- счётчик

SET @number = (SELECT COUNT(COLUMN\_NAME) FROM INFORMATION\_SCHEMA.COLUMNS

WHERE (TABLE\_NAME = @Table)) -- считаем кол-во столбцов в таблице

--5. Запускаем цикл

WHILE @number > 0

BEGIN

-- отределяем поле таблицы csv для сравнения

SET @Column = (SELECT COLUMN\_NAME FROM INFORMATION\_SCHEMA.COLUMNS

WHERE (TABLE\_NAME = @Table)

AND ORDINAL\_POSITION = @number) --1

-- вставляем данные во временную таблицу

INSERT INTO #TempColumn

EXEC (N'SELECT ' + @Column + ' FROM ' + @Table)

-- Merge

MERGE #TableInput target

USING #TempColumn source ON target.[Function\_name]=source.[Function\_name]

--если значения совпадают с целевой таблицей плюсуем 1

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] =target.[Function\_count] + 1

--если не совпадает с целевой таблицей, и не NULL, то вставляем название функции и значение присваиваем 1

WHEN NOT MATCHED BY TARGET AND source.[Function\_name] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[Function\_name], 1);

SET @number = @number - 1

DELETE FROM #TempColumn

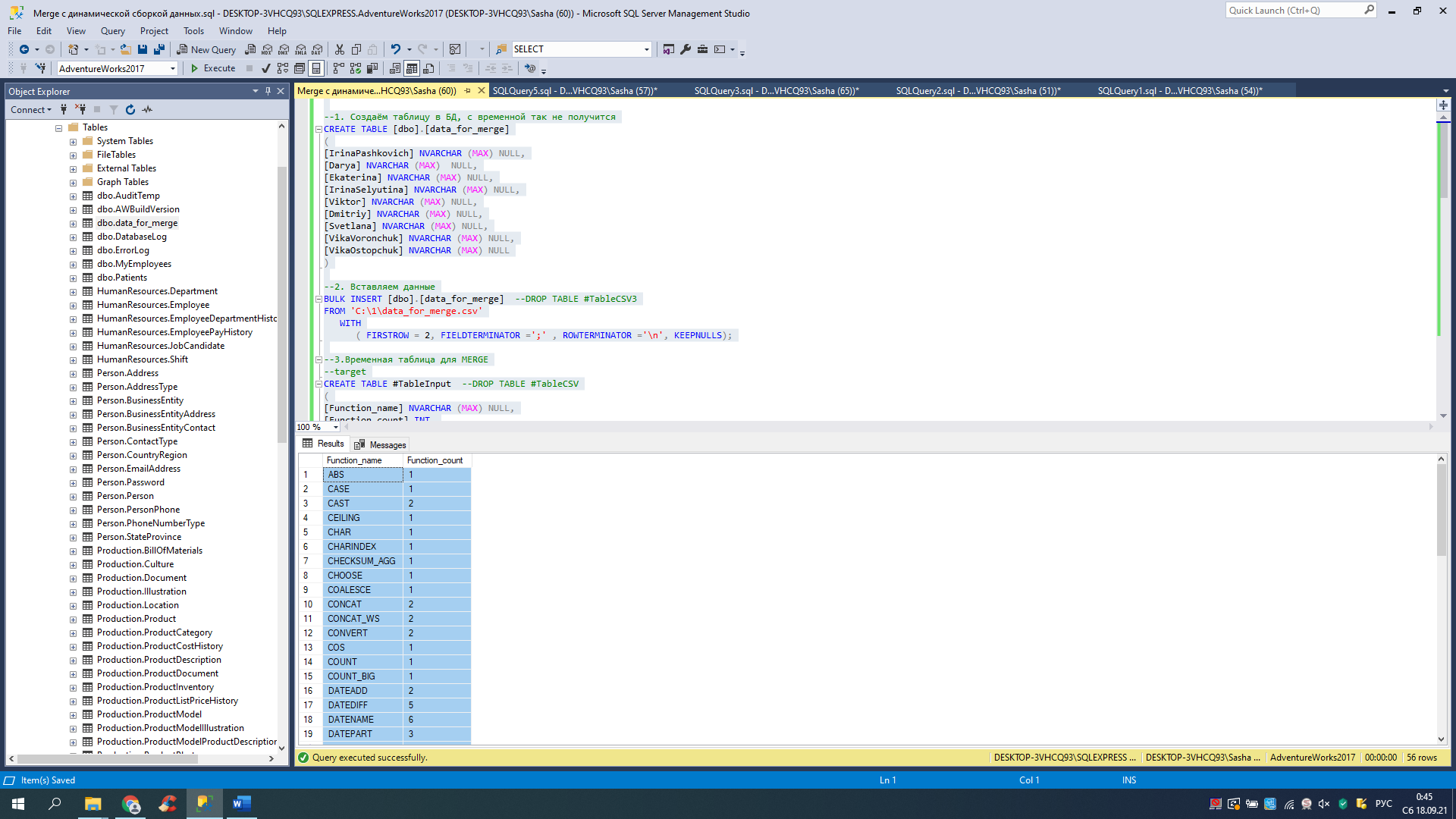
END;

-- Т.о. прогоняем 9 столбцов на сравнение

SELECT \* FROM #TableInput

ORDER BY [Function\_name]

РЕЗУЛЬТАТ



|  |  |
| --- | --- |
| Function\_name | Function\_count |
| ABS | 1 |
| CASE | 1 |
| CAST | 2 |
| CEILING | 1 |
| CHAR | 1 |
| CHARINDEX | 1 |
| CHECKSUM\_AGG | 1 |
| CHOOSE | 1 |
| COALESCE | 1 |
| CONCAT | 2 |
| CONCAT\_WS | 2 |
| CONVERT | 2 |
| COS | 1 |
| COUNT | 1 |
| COUNT\_BIG | 1 |
| DATEADD | 2 |
| DATEDIFF | 5 |
| DATENAME | 6 |
| DATEPART | 3 |
| DAY | 3 |
| EOMONTH | 2 |
| FLOOR | 2 |
| FORMAT | 1 |
| GETDATE | 2 |
| GROUPING | 1 |
| IIF | 4 |
| ISDATE | 1 |
| LEFT | 2 |
| LEN | 3 |
| LOG | 1 |
| LOWER | 1 |
| MAX | 2 |
| MIN | 2 |
| MONTH | 3 |
| PATINDEX | 1 |
| POWER | 1 |
| REPLACE | 2 |
| REPLICATE | 1 |
| REVERSE | 2 |
| ROUND | 4 |
| SIN | 1 |
| SOUNDEX | 1 |
| SPACE | 1 |
| SQRT | 1 |
| SQUARE | 1 |
| STDEV | 1 |
| STRING\_AGG | 1 |
| SUBSTRING | 3 |
| SYSDATETIME | 1 |
| SYSDATETIMEOFFSET | 1 |
| SYSUTCDATETIME | 1 |
| TAN | 1 |
| TRIM | 1 |
| UPPER | 2 |
| VAR | 1 |
| YEAR | 4 |

56 строк

2 ВАРИАНТ РЕШЕНИЯ

-- 1. Создаём временную таблицу для данных

CREATE TABLE #TableCSV

(

[IrinaPashkovich] NVARCHAR (MAX) NULL,

[Darya] NVARCHAR (MAX) NULL,

[Ekaterina] NVARCHAR (MAX) NULL,

[IrinaSelyutina] NVARCHAR (MAX) NULL,

[Viktor] NVARCHAR (MAX) NULL,

[Dmitriy] NVARCHAR (MAX) NULL,

[Svetlana] NVARCHAR (MAX) NULL,

[VikaVoronchuk] NVARCHAR (MAX) NULL,

[VikaOstopchuk] NVARCHAR (MAX) NULL

)

-- 2. Вставляем данные из CSV

BULK INSERT #TableCSV

FROM 'C:\1\data\_for\_merge.csv'

WITH

( FIRSTROW = 2, FIELDTERMINATOR =';' , ROWTERMINATOR ='\n', KEEPNULLS);

-- 3. Временная таблица для результата

CREATE TABLE #TableInput --DROP TABLE #TableCSV

(

[Function\_name] NVARCHAR (MAX) NULL,

[Function\_count] INT

)

-- 4. Выполняем Merge со всеми полями

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[IrinaPashkovich]

--если значения совпадают с целевой таблицей плюсуем 1

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

--если не совпадает с целевой таблицей, и не NULL, то вставляем название функции и значение присваиваем 1

WHEN NOT MATCHED BY TARGET AND source.[IrinaPashkovich] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[IrinaPashkovich], 1);

-- и так для каждого поля

--[Darya]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[Darya]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[Darya] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[Darya], 1);

--[Ekaterina]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[Ekaterina]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[Ekaterina] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[Ekaterina], 1);

--[IrinaSelyutina]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[IrinaSelyutina]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[IrinaSelyutina] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[IrinaSelyutina], 1);

--[Viktor]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[Viktor]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[Viktor] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[Viktor], 1);

--[Dmitriy]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[Dmitriy]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[Dmitriy] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[Dmitriy], 1);

--[Svetlana]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[Svetlana]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[Svetlana] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[Svetlana], 1);

--[VikaVoronchuk]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[VikaVoronchuk]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[VikaVoronchuk] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[VikaVoronchuk], 1);

--[VikaOstopchuk]

MERGE #TableInput target

USING #TableCSV source ON target.[Function\_name]=source.[VikaOstopchuk]

WHEN MATCHED

THEN UPDATE SET target.[Function\_count] = target.[Function\_count] + 1

WHEN NOT MATCHED BY TARGET AND source.[VikaOstopchuk] IS NOT NULL

THEN INSERT ([Function\_name], [Function\_count])

VALUES (source.[VikaOstopchuk], 1);

SELECT \* FROM #TableInput

ORDER BY [Function\_name]

|  |  |
| --- | --- |
| Function\_name | Function\_count |
| ABS | 1 |
| CASE | 1 |
| CAST | 2 |
| CEILING | 1 |
| CHAR | 1 |
| CHARINDEX | 1 |
| CHECKSUM\_AGG | 1 |
| CHOOSE | 1 |
| COALESCE | 1 |
| CONCAT | 2 |
| CONCAT\_WS | 2 |
| CONVERT | 2 |
| COS | 1 |
| COUNT | 1 |
| COUNT\_BIG | 1 |
| DATEADD | 2 |
| DATEDIFF | 5 |
| DATENAME | 6 |
| DATEPART | 3 |
| DAY | 3 |
| EOMONTH | 2 |
| FLOOR | 2 |
| FORMAT | 1 |
| GETDATE | 2 |
| GROUPING | 1 |
| IIF | 4 |
| ISDATE | 1 |
| LEFT | 2 |
| LEN | 3 |
| LOG | 1 |
| LOWER | 1 |
| MAX | 2 |
| MIN | 2 |
| MONTH | 3 |
| PATINDEX | 1 |
| POWER | 1 |
| REPLACE | 2 |
| REPLICATE | 1 |
| REVERSE | 2 |
| ROUND | 4 |
| SIN | 1 |
| SOUNDEX | 1 |
| SPACE | 1 |
| SQRT | 1 |
| SQUARE | 1 |
| STDEV | 1 |
| STRING\_AGG | 1 |
| SUBSTRING | 3 |
| SYSDATETIME | 1 |
| SYSDATETIMEOFFSET | 1 |
| SYSUTCDATETIME | 1 |
| TAN | 1 |
| TRIM | 1 |
| UPPER | 2 |
| VAR | 1 |
| YEAR | 4 |

56 строк