

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ «ЛЭТИ»**

ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

197376, Санкт-Петербург, ул. проф. Попова, 5.

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

«З А Ч Т Е Н О»

_____ О.А. Жирнова

“ ” _____ 2021 г.

**ОТЧЁТ
по дисциплине «Базы данных»**

**Лабораторная работа № 9
«Создание UDF»**

Студент группы 9308 _____

Н.В. Яловега

Санкт Петербург 2021

Цель работы: научиться писать и применять функции, определяемые пользователем

Используемая база данных (БД): AdventureWorks

Порядок выполнения

Упражнение 1 – Создание скалярной функции.

Запрос 1. Создание скалярной функции, определяемой пользователем, Sales.GetMaximumDiscountForCategory, которая находит максимальный процент скидки (поле DiscountPct), доступный на данный момент для конкретной категории. Создали параметр @Category nvarchar(50) для ограничения результатов на основе категории и использовали функцию GETDATE() для ограничения строк на основе доступности скидки на данный момент в диапазоне StartDate и EndDate. Результат на рисунке 1.

```
IF OBJECT_ID ('Sales.GetMaximumDiscountForCategory', 'FN') IS NOT NULL
DROP FUNCTION Sales.GetMaximumDiscountForCategory
GO
CREATE FUNCTION Sales.GetMaximumDiscountForCategory(@Category
nvarchar(50) NULL)
RETURNS int
AS
BEGIN
RETURN (SELECT MAX(DiscountPct)
FROM Sales.SpecialOffer
WHERE Category = @Category AND GETDATE() >= StartDate AND GETDATE()
<= EndDate)
END
```

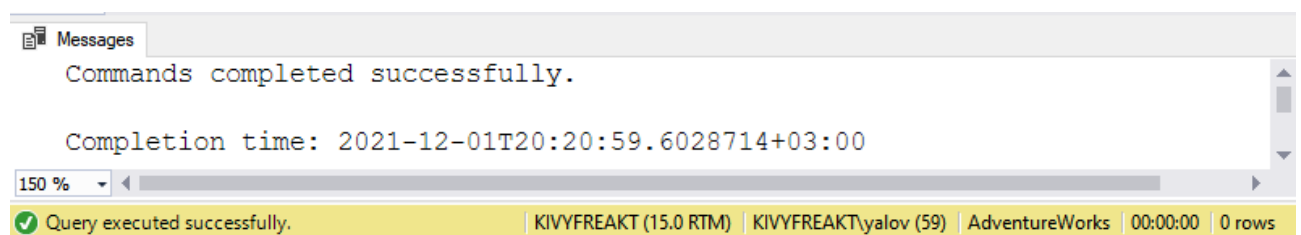
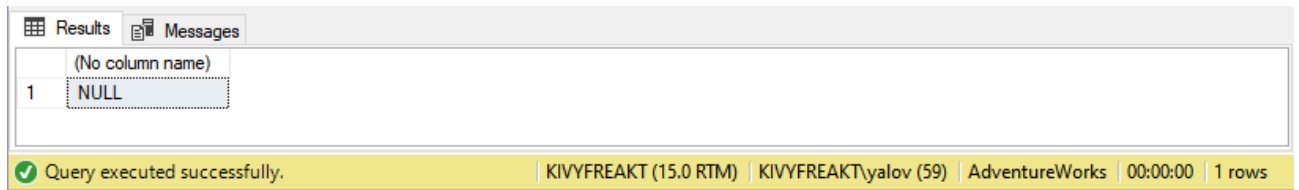


Рисунок 1

Запрос 2. Проверка функции. Результат на рисунке 2

```
SELECT Sales.GetMaximumDiscountForCategory('Reseller')
```



The screenshot shows the SQL Server Enterprise Manager interface. At the top, there are tabs for 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with one row and one column. The column header is '(No column name)' and the row value is 'NULL'. Below the table, a status bar indicates 'Query executed successfully.' followed by server and session information: 'KIVYFREACT (15.0 RTM) | KIVYFREACT\ygalov (59) | AdventureWorks | 00:00:00 | 1 rows'.

	(No column name)
1	NULL

Query executed successfully. | KIVYFREACT (15.0 RTM) | KIVYFREACT\ygalov (59) | AdventureWorks | 00:00:00 | 1 rows

Рисунок 2

Упражнение 2 – Создание функции, возвращающей табличное значение.

Запрос 1. Создание функции Sales.GetDiscountsForDate, у которой входной параметр @DateToCheck datetime. Он используется для фильтрации скидок на основе введенной даты. Это позволяет компании Adventure Works проверить, какие скидки будут доступны на указанную дату. Результат на рисунке 3.

```
IF OBJECT_ID ('Sales.GetDiscountsForDate', 'IF') IS NOT NULL
DROP FUNCTION Sales.GetDiscountsForDate
GO
CREATE FUNCTION Sales.GetDiscountsForDate(@DateToCheck datetime)
RETURNS TABLE
AS
RETURN (SELECT Description, DiscountPct, Type, Category, StartDate, EndDate,
MinQty, MaxQty
FROM Sales.SpecialOffer
WHERE StartDate <= @DateToCheck and EndDate >= @DateToCheck
ORDER BY StartDate, EndDate ASC
OFFSET 0 ROW)
```

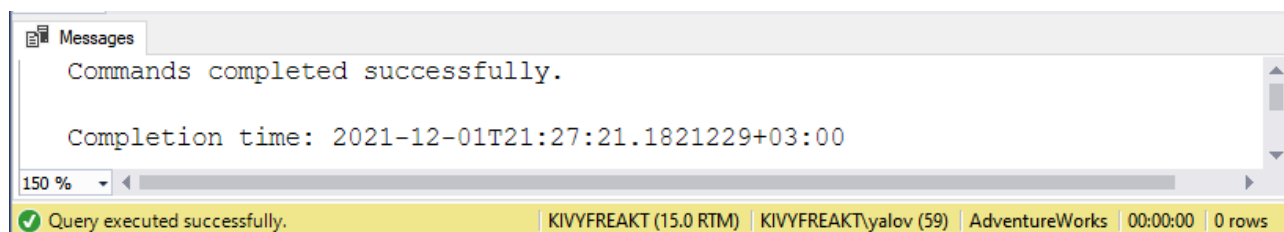


Рисунок 3

Запрос 2. проверка функции. Результат на рисунке 4.

```
SELECT * FROM Sales.GetDiscountsForDate(GETDATE()) ORDER BY DiscountPct DESC
```

The screenshot shows the 'Results' window in SQL Server Enterprise Manager. It displays a table with 9 columns: Description, DiscountPct, Type, Category, StartDate, EndDate, MinQty, and MaxQty. There is one row of data.

	Description	DiscountPct	Type	Category	StartDate	EndDate	MinQty	MaxQty
1	Half price off everything	0,50	Seasonal Discount	Customer	2021-11-18 22:06:13.157	2021-12-18 22:06:13.157	0	20

The status bar at the bottom indicates 'Query executed successfully.' and '1 rows'.

Рисунок 4

Запрос 1. Создание функции, возвращающей табличное значение (Multy-Statement Table-valued UDF). Создали функцию GetDiscountedProducts в схеме Sales. В этой функции был написан запрос для поиска продуктов, имеющих скидку. Для формирования этого запроса соединены таблицы: Sales.SpecialOfferProduct, Sales.SpecialOffer и Production.Product. Запрос выводит следующие данные: столбцы ProductID, Name, ListPrice из таблицы Production.Product, столбцы Description и DiscountPct из таблицы Sales.SpecialOffer, а также два вычисляемых столбца. Первый вычисляемый столбец получается в результате произведения значений из поля ListPrice на DiscountPct; второй – в результате вычитания из ListPrice произведения значений поля ListPrice на DiscountPct. У функции есть параметр @IncludeHistory bit, который применяется для фильтрации возвращенной таблицы на основе того, требуются ли сведения об истории скидок или необходимы только текущие сведения. Результат на рисунке 5.

```
IF OBJECT_ID ('Sales.GetDiscountedProducts', 'TF') IS NOT NULL
DROP FUNCTION Sales.GetDiscountedProducts
GO
CREATE FUNCTION Sales.GetDiscountedProducts(@IncludeHistory bit)
RETURNS @retVal TABLE (
ProductID int NOT NULL,
Name nvarchar(50),
ListPrice money,
DiscountDescription nvarchar(255),
DiscountPercentage smallmoney,
DiscountAmount money,
DiscountedPrice money
)
AS BEGIN
INSERT @retVal SELECT pp.ProductID,
pp.Name,
pp.ListPrice,
so.Description,
so.DiscountPct,
pp.ListPrice * so.DiscountPct as DiscountAmt,
ListPrice - pp.ListPrice * so.DiscountPct as DiscountPrice
FROM Sales.SpecialOffer AS so
INNER JOIN Sales.SpecialOfferProduct AS sop ON so.SpecialOfferID =
sop.SpecialOfferID
INNER JOIN Production.Product AS pp ON sop.ProductID = pp.ProductID
WHERE (@IncludeHistory = 1)
```

```

OR (@IncludeHistory = 0 AND so.EndDate >= GETDATE() AND so.StartDate <=
GETDATE())
ORDER BY pp.ProductID ASC
RETURN
END;

```

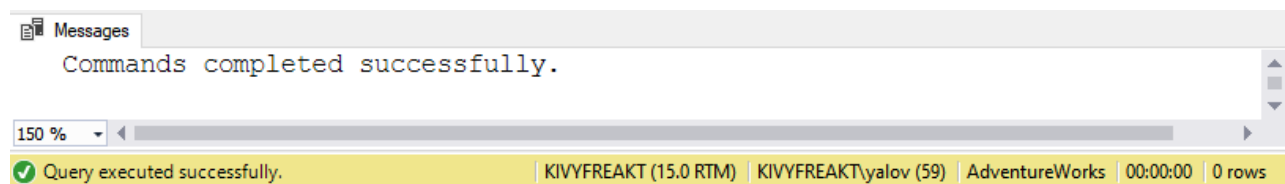


Рисунок 5

Запрос 2. Проверка функции. Результат на рисунке 6.

```
SELECT * FROM Sales.GetDiscountedProducts(0)
```

```
SELECT * FROM Sales.GetDiscountedProducts(1)
```

Результаты		Сообщения					
	ProductID	Name	ListPrice	DiscountDescription	DiscountPercentage	DiscountAmount	DiscountedPrice
1	680	HL Road Frame - Black, 58	1431.50	No Discount	0.00	0.00	1431.50
2	706	HL Road Frame - Red, 58	1431.50	No Discount	0.00	0.00	1431.50
3	707	Sport-100 Helmet, Red	34.99	No Discount	0.00	0.00	34.99
4	707	Sport-100 Helmet, Red	34.99	Volume Discount 15 to 24	0.05	1.7495	33.2405
5	707	Sport-100 Helmet, Red	34.99	Sport Helmet Discount-2003	0.15	5.2485	29.7415
6	708	Sport-100 Helmet, Black	34.99	Sport Helmet Discount-2003	0.15	5.2485	29.7415
7	708	Sport-100 Helmet, Black	34.99	Volume Discount 15 to 24	0.05	1.7495	33.2405
8	708	Sport-100 Helmet, Black	34.99	No Discount	0.00	0.00	34.99

	ProductID	Name	ListPrice	DiscountDescription	DiscountPercentage	DiscountAmount	DiscountedPrice
1	680	HL Road Frame - Black, 58	1431.50	No Discount	0.00	0.00	1431.50
2	706	HL Road Frame - Red, 58	1431.50	No Discount	0.00	0.00	1431.50
3	707	Sport-100 Helmet, Red	34.99	No Discount	0.00	0.00	34.99
4	707	Sport-100 Helmet, Red	34.99	Volume Discount ...	0.02	0.6998	34.2902
5	707	Sport-100 Helmet, Red	34.99	Volume Discount ...	0.05	1.7495	33.2405
6	707	Sport-100 Helmet, Red	34.99	Sport Helmet Disc...	0.15	5.2485	29.7415
7	707	Sport-100 Helmet, Red	34.99	Sport Helmet Disc...	0.10	3.499	31.491
8	708	Sport-100 Helmet, Black	34.99	Sport Helmet Disc...	0.10	3.499	31.491
9	708	Sport-100 Helmet, Black	34.99	Sport Helmet Disc...	0.15	5.2485	29.7415
10	708	Sport-100 Helmet, Black	34.99	Volume Discount ...	0.05	1.7495	33.2405
11	708	Sport-100 Helmet, Black	34.99	Volume Discount ...	0.02	0.6998	34.2902
12	708	Sport-100 Helmet, Black	34.99	No Discount	0.00	0.00	34.99
13	709	Mountain Bike Socks, M	9.50	No Discount	0.00	0.00	9.50
14	709	Mountain Bike Socks, M	9.50	Volume Discount ...	0.02	0.19	9.31
15	709	Mountain Bike Socks, M	9.50	Volume Discount ...	0.05	0.475	9.025

(загружено строк: 381)

(загружено строк: 538)

Рисунок 6

Выводы

Написаны и применены функции, определяемые пользователем. Созданы функции: определяемая пользователем скалярная функция Sales.GetMaximumDiscountForCategory, функция возвращающая табличное значение (In-Line Table-valued UDF) Sales.GetDiscountsForDate, функция возвращающая табличное значение (Multy-Statement Table-valued UDF) GetDiscountedProducts. Для создания функций, определяемых пользователем, используется команда CREATE FUNCTION.

Список использованных источников

1. Горячев А. В., Новакова Н. Е. Распределенные базы данных. Мет. указания к лаб. работам., СПб. Изд-во СПбГЭТУ «ЛЭТИ», 2008
2. Горячев А.В, Новакова Н.Е. Особенности разработки и администрирования приложений баз данных: учеб. пособие. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2016. 68 с.
3. Дейт К. Введение в системы баз данных. : Пер. с англ. – 6-е изд. - К.:Диалектика, 1998.