

```

/*
a) Создайте scalar-valued функцию, которая будет принимать в качестве входного параметра
id заказа (Sales.SalesOrderHeader.SalesOrderID) и
возвращать максимальную цену продукта из заказа (Sales.SalesOrderDetail.UnitPrice).
*/
CREATE FUNCTION Sales.getMaxUnitPrice
(
    @id INT
)
RETURNS MONEY
AS
BEGIN
    RETURN
    (
        SELECT
            MAX(Detail.UnitPrice) AS MaxUnitPrice
        FROM Sales.SalesOrderDetail AS Detail
        WHERE Detail.SalesOrderID = @id
        GROUP BY Detail.SalesOrderID
    );
END;
GO

```

Рисунок 1 – Задание а (Решение)

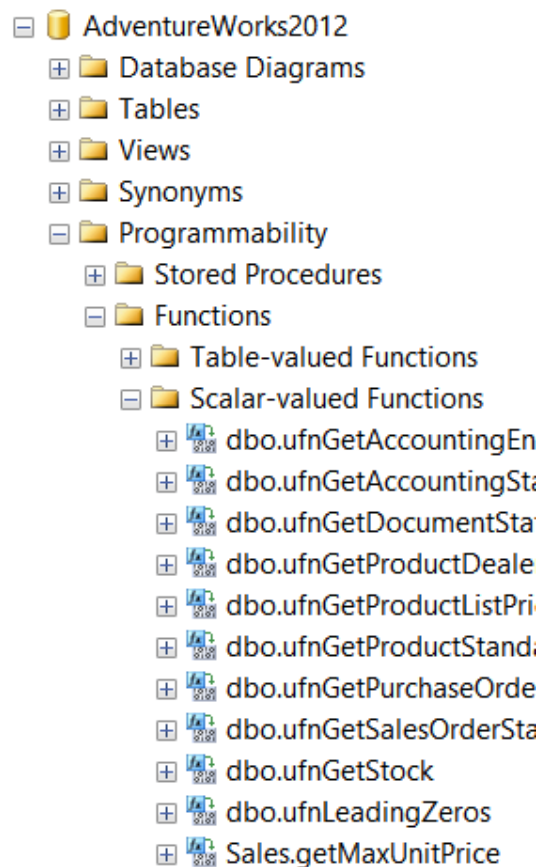


Рисунок 2 – Задание а (Результат)

```

/*
b) Создайте inline table-valued функцию, которая будет принимать в качестве входных параметров
id продукта (Production.Product.ProductID) и количество строк, которые необходимо вывести.
Функция должна возвращать определенное количество инвентаризационных записей о продукте
с наибольшим его количеством (по Quantity) из Production.ProductInventory.
Функция должна возвращать только продукты, хранящиеся в отделе A (Production.ProductInventory.Shelf).
*/
CREATE FUNCTION Production.getInventoryRecords
(
    @id INT,
    @count INT
)
RETURNS TABLE
AS
RETURN
(
    SELECT TOP(@count)
        *
    FROM Production.ProductInventory AS Inventory
    WHERE
        Inventory.Shelf = 'A' AND
        Inventory.ProductID = @id
    ORDER BY
        Quantity DESC
);
GO

```

Рисунок 3 – Задание b (Решение)

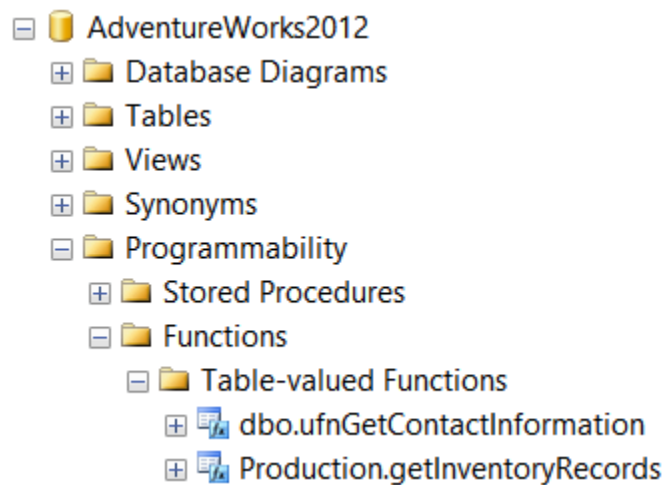


Рисунок 4 – Задание b (Результат)

```

/*
    c) Вызовите функцию для каждого продукта, применив оператор CROSS APPLY.
*/
SELECT
    Records.*
FROM Production.Product AS Prod
    CROSS APPLY Production.getInventoryRecords(Prod.ProductID, 1) AS Records
ORDER BY Prod.ProductID;

```

Рисунок 5 – Задание с (Решение)

	Product...	Locatio...	Sh...	Bin	Quant...	rowguid	ModifiedDate
2	2	1	A	2	427	F407C07A-CA14-4684-A02C-608BD00C2233	2008-09-08 00:00:00.000
3	3	1	A	7	585	E18A519B-FB5E-4051-874C-58CD58436C95	2002-05-02 00:00:00.000
4	4	1	A	6	512	6BEAF0A0-971A-4CE1-96FE-692807D5DC00	2008-09-08 00:00:00.000
5	316	5	A	11	532	1EE3DBD3-2A7E-47DC-AF99-1B585575EFB9	2002-05-02 00:00:00.000
6	317	5	A	1	158	83332A73-48A9-401D-95F4-385C944D716F	2008-09-09 00:00:00.000
7	318	5	A	2	171	B62232E8-90B5-4DA1-BFE1-453AA1917EFC	2008-09-09 00:00:00.000
8	319	50	A	23	305	FEBBCC76-2764-48A3-A086-77D1E883137C	2008-09-09 00:00:00.000
9	320	5	A	4	372	DEEDBA07-171B-4038-88A5-A57166E8F446	2008-09-09 00:00:00.000
10	321	50	A	26	641	B291EE86-CB6E-4D74-B47D-B8B0794CA9C4	2008-09-09 00:00:00.000
11	322	5	A	7	587	04DA1BB9-9625-4E71-B861-93F64A3A53DC	2008-09-09 00:00:00.000
12	323	5	A	12	568	BD637BC5-EB67-4424-8F92-DCA208276E6F	2008-09-09 00:00:00.000
13	324	5	A	13	568	AEC9BF74-0E06-4181-B6C7-C8FA41126A4E	2002-05-02 00:00:00.000
14	325	60	A	1	641	D42750B2-2C8A-4457-9DDA-4B43CBDC4594	2008-09-12 00:00:00.000
15	326	60	A	2	475	EADAA7F8-E962-46C4-977F-05538C494BCD	2008-09-12 00:00:00.000
16	328	5	A	8	568	BF52885D-74A3-422B-8C75-8A46350D9A6D	2002-05-02 00:00:00.000
17	329	5	A	9	558	7A1417A3-21EA-4DA1-B3D1-E85BAB7D8A...	2002-05-02 00:00:00.000

Рисунок 6 – Задание с (Результат)

```

/*
    d) Вызовите функцию для каждого продукта, применив оператор OUTER APPLY.
*/
SELECT
    Records.*
FROM Production.Product AS Prod
    OUTER APPLY Production.getInventoryRecords(Prod.ProductID, 1) AS Records
ORDER BY Prod.ProductID;
GO

```

Рисунок 7 – Задание d (Решение)

	Product...	Locatio...	Shelf	Bin	Quant...	rowguid	ModifiedDate
1	1	1	A	1	408	47A24246-6C43-48EB-968F-025738A8A410	2008-09-08 00:00:00.000
2	2	1	A	2	427	F407C07A-CA14-4684-A02C-608BD00C2233	2008-09-08 00:00:00.000
3	3	1	A	7	585	E18A519B-FB5E-4051-874C-58CD58436C95	2002-05-02 00:00:00.000
4	4	1	A	6	512	6BEAF0A0-971A-4CE1-96FE-692807D5DC00	2008-09-08 00:00:00.000
5	316	5	A	11	532	1EE3DBD3-2A7E-47DC-AF99-1B585575EFB9	2002-05-02 00:00:00.000
6	317	5	A	1	158	83332A73-48A9-401D-95F4-385C944D716F	2008-09-09 00:00:00.000
7	318	5	A	2	171	B62232E8-90B5-4DA1-BFE1-453AA1917EFC	2008-09-09 00:00:00.000
8	319	50	A	23	305	FEBBCC76-2764-48A3-A086-77D1E883137C	2008-09-09 00:00:00.000
9	320	5	A	4	372	DEEDBA07-171B-4038-88A5-A57166E8F446	2008-09-09 00:00:00.000
10	321	50	A	26	641	B291EE86-CB6E-4D74-B47D-B8B0794CA9C4	2008-09-09 00:00:00.000
11	322	5	A	7	587	04DA1BB9-9625-4E71-B861-93F64A3A53DC	2008-09-09 00:00:00.000
12	323	5	A	12	568	BD637BC5-EB67-4424-8F92-DCA208276E6F	2008-09-09 00:00:00.000
13	324	5	A	13	568	AEC9BF74-0E06-4181-B6C7-C8FA41126A4E	2002-05-02 00:00:00.000
14	325	60	A	1	641	D42750B2-2C8A-4457-9DDA-4B43CBDC4594	2008-09-12 00:00:00.000
15	326	60	A	2	475	EADAA7F8-E962-46C4-977F-05538C494BCD	2008-09-12 00:00:00.000
16	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 8 – Задание d (Результат)

```

/*
e) Измените созданную inline table-valued функцию, сделав ее multistatement table-valued
(предварительно сохранив для проверки код создания inline table-valued функции).
*/
DROP FUNCTION Production.getInventoryRecords;
GO

CREATE FUNCTION Production.getInventoryRecords
(
    @id INT,
    @count INT
)
RETURNS @inventoryRecords TABLE
(
    ProductID INT,
    LocationID SMALLINT,
    Shelf NVARCHAR(10),
    Bin TINYINT,
    Quantity SMALLINT,
    rowguid UNIQUEIDENTIFIER,
    ModifiedDate DATETIME
)
AS
BEGIN
    INSERT INTO @inventoryRecords
    SELECT TOP(@count)
        *
    FROM Production.ProductInventory AS Inventory
    WHERE
        Inventory.Shelf = 'A' AND
        Inventory.ProductID = @id
    ORDER BY
        Quantity DESC
    RETURN
END;
GO

```

Рисунок 9 – Задание е (Решение)

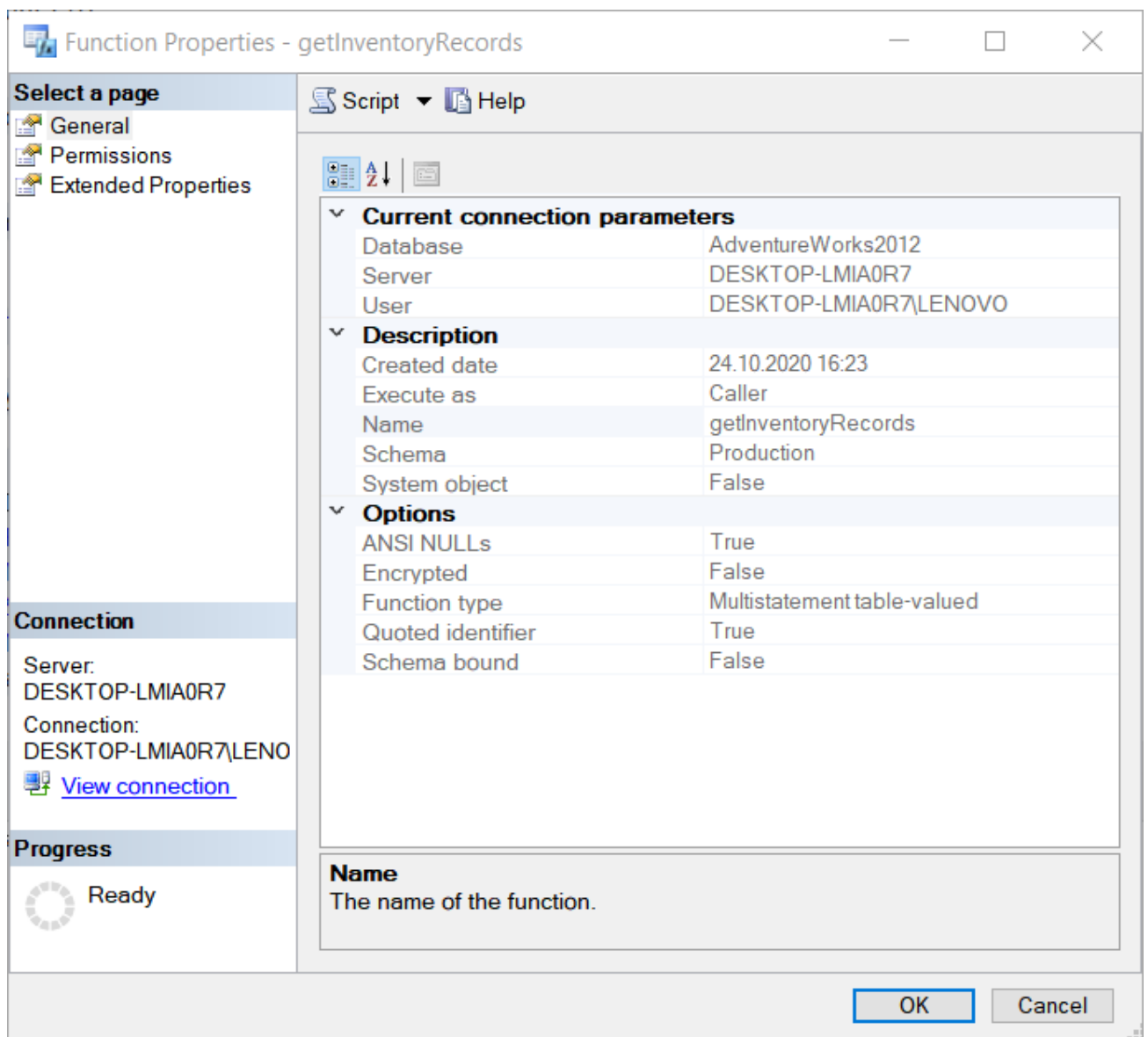


Рисунок 10 – Задание е (Результат)