Задание 1.

Код запроса/создания функции:

USE [stack]

GO

--задание 1

IF OBJECT\_ID ('select\_orders\_by\_item\_name') IS NOT NULL

DROP FUNCTION select\_orders\_by\_item\_name;

GO

CREATE FUNCTION select\_orders\_by\_item\_name(@ItemName nvarchar(max))

RETURNS Table AS RETURN

(

SELECT Orders.row\_id as order\_id, Customers.name as customer, COUNT(Orders.row\_id) as items\_count

FROM Orders, OrderItems, Customers

WHERE Orders.customer\_id = Customers.row\_id AND OrderItems.order\_id = Orders.row\_id AND OrderItems.name = @ItemName

GROUP BY Orders.row\_id, Customers.name, OrderItems.name

);

GO

-- примеры задания 1

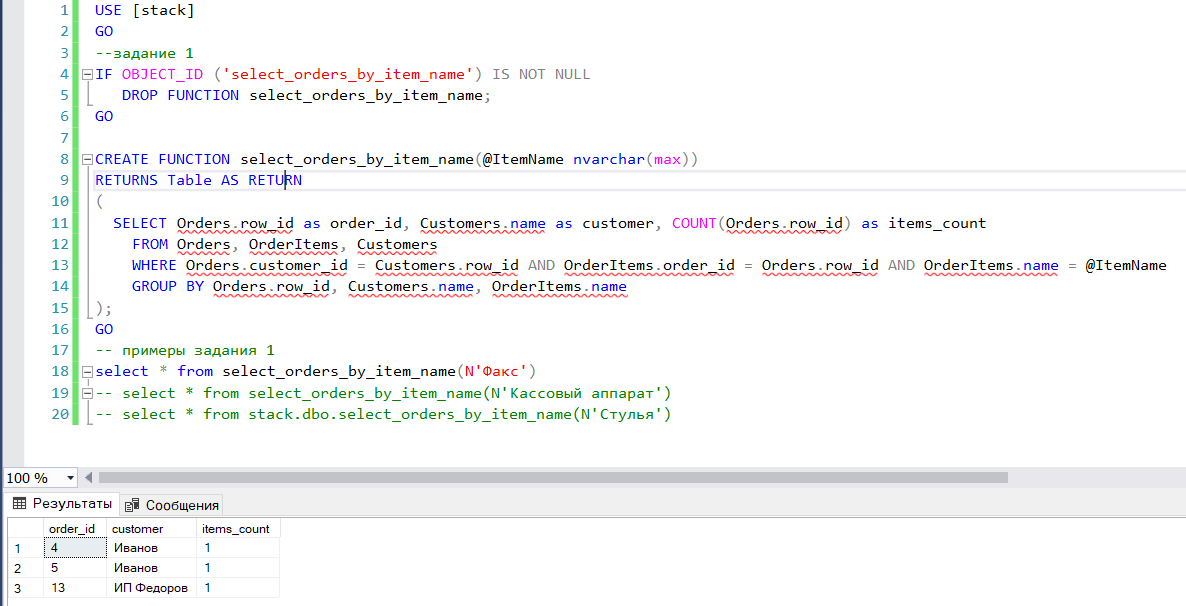
-- select \* from select\_orders\_by\_item\_name(N'Факс')

-- select \* from select\_orders\_by\_item\_name(N'Кассовый аппарат')

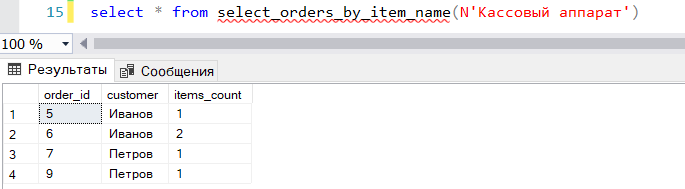
-- select \* from stack.dbo.select\_orders\_by\_item\_name(N'Стулья')

Отчёт о работе функции:

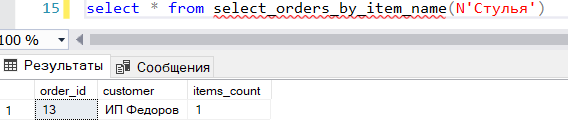
select \* from select\_orders\_by\_item\_name(N'Факс')



select \* from select\_orders\_by\_item\_name(N'Кассовый аппарат')



select \* from stack.dbo.select\_orders\_by\_item\_name(N'Стулья')



Задание 2.

Код запроса/создания функции:

IF OBJECT\_ID ('is\_root') IS NOT NULL

DROP FUNCTION is\_root;

GO

CREATE FUNCTION is\_root(@RowId int)

RETURNS BIT AS BEGIN

IF (SELECT TOP 1 group\_name

FROM Orders

WHERE row\_id = @RowId) IS NOT NULL

RETURN 1

RETURN 0

END;

GO

IF OBJECT\_ID ('calculate\_total\_price\_for\_orders\_group') IS NOT NULL

DROP FUNCTION calculate\_total\_price\_for\_orders\_group;

GO

CREATE FUNCTION calculate\_total\_price\_for\_orders\_group(@RowId int)

RETURNS INT AS BEGIN

DECLARE @total\_sum INT;

SET @total\_sum = 0;

IF dbo.is\_root(@RowId) = 1

SELECT @total\_sum = @total\_sum + dbo.calculate\_total\_price\_for\_orders\_group(o.row\_id)

FROM Orders AS o

WHERE o.parent\_id = @RowId

ELSE

SELECT @total\_sum = sum(i.price)

FROM Orders AS o inner join OrderItems AS i ON o.row\_id = i.order\_id

WHERE o.row\_id = @RowId

RETURN @total\_sum

END;

GO

-- примеры задания 2

--select dbo.calculate\_total\_price\_for\_orders\_group(1) as total\_price1 -- 703, все заказы

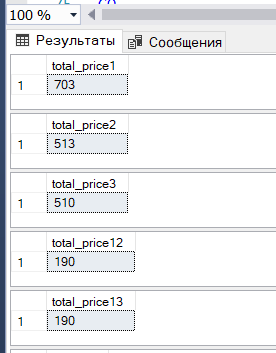
--select dbo.calculate\_total\_price\_for\_orders\_group(2) as total\_price2 -- 513, группа 'Частные лица'

--select dbo.calculate\_total\_price\_for\_orders\_group(3) as total\_price3 -- 510, группа 'Оргтехника'

--select dbo.calculate\_total\_price\_for\_orders\_group(12) as total\_price12 -- 190, группа 'Юридические лица'

--select dbo.calculate\_total\_price\_for\_orders\_group(13) as total\_price13 -- 190, заказ 'ИП Федоров'

Отчёт о работе функции:



Задание 3.

Код запроса:

SELECT Customers.name

FROM

Orders, OrderItems, Customers

WHERE

Orders.customer\_id = Customers.row\_id

AND OrderItems.order\_id = Orders.row\_id

AND Orders.registered\_at > '2019-12-31'

AND Orders.registered\_at < '2021-01-01'

GROUP BY Customers.name

HAVING COUNT(DISTINCT OrderItems.order\_id)

=

COUNT(DISTINCT

CASE WHEN OrderItems.name = N'Кассовый аппарат'

THEN OrderItems.order\_id END)

;

Отчёт о работе функции:

