

## 2. Создать 3 пользовательских функций:

a) Скалярная функция, подсчитывающая прибыль супермаркета за заданный период

```

CREATE FUNCTION Profit (@start_date DATE, @end_date DATE)
RETURNS DECIMAL(15,2)
AS
BEGIN
    DECLARE @profit DECIMAL(15,2);

    SELECT @profit =
        (SELECT SUM(final_amount) -- выручка
         FROM Sale
         WHERE CAST(sale_datetime AS DATE) BETWEEN @start_date AND @end_date)
        -
        (SELECT SUM(si.quantity * p.purchase_price) -- сумма закупки
         FROM SaleItem si
         JOIN Products p ON si.product_id = p.id
         JOIN Sale s ON si.sale_id = s.id
         WHERE CAST(s.sale_datetime AS DATE) BETWEEN @start_date AND @end_date)

    RETURN @profit;
END;
GO

SELECT dbo.Profit('2025-09-01', '2025-11-01') AS Прибыль;

```

Результаты	
	Прибыль
1	11180.20

b) Inline-функция, по заданной карте возвращающая список производителей товаров, которые когда-либо покупались владельцем карты

```

CREATE FUNCTION Manufacturers
(
    @card_id INT
)
RETURNS TABLE
AS
RETURN
(
    SELECT DISTINCT p.manufacturer AS Производитель
    FROM Sale s
    JOIN Customer c ON s.customer_id = c.id
    JOIN SaleItem si ON s.id = si.sale_id
    JOIN Products p ON si.product_id = p.id
    WHERE c.card_id = @card_id
);
GO

SELECT * FROM dbo.Manufacturers(1)

```

Производитель	
1	Greenfield
2	Макфа
3	Мираторг
4	Мондэлис
5	Петелинка
6	Простоквашино
7	Русская соль
8	Сырный двор
9	Ярмолпрод

c) Multi-statement-функция, выдающая список карт, суммарная стоимость покупок по которым превысила 10 000р., в виде: номер карты, дата выдачи, % скидки, общая сумма покупок по карте

```

CREATE FUNCTION ListCards()
RETURNS @result TABLE
(
    card_number INT,
    discount_rate DECIMAL(5,2),
    total_purchases DECIMAL(15,2)
)
AS
BEGIN
    INSERT INTO @result
    SELECT
        dc.id AS card_number,
        dc.discount_rate,
        SUM(s.final_amount) AS total_purchases
    FROM DiscountCard dc
    JOIN Customer c ON dc.id = c.card_id
    JOIN Sale s ON c.id = s.customer_id
    GROUP BY dc.id, dc.discount_rate
    HAVING SUM(s.final_amount) > 10000

    RETURN;
END;
GO

SELECT * FROM dbo.ListCards();

```

Результаты			
	card_number	discount_rate	total_purchases

Результат, если условие "Сумма покупок > 10000р."

Результаты			
	card_number	discount_rate	total_purchases
1	1	3.00	1794.50
2	2	5.00	3040.00
3	3	7.00	1391.60
4	4	10.00	2432.00
5	5	3.00	1629.60
6	6	4.00	2128.00
7	8	8.00	3249.00

Результат, если изменить условие на "Сумма покупок > 1000р."