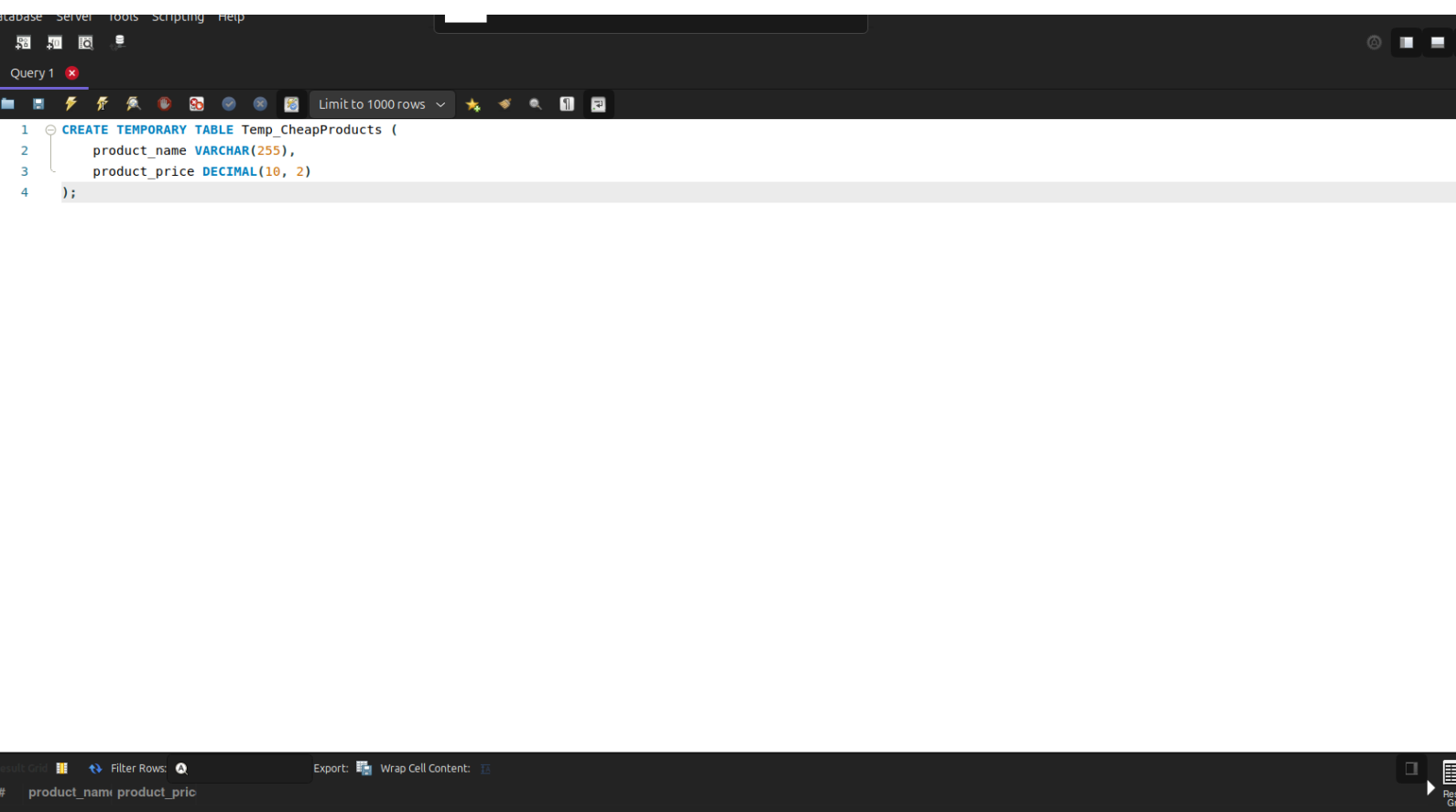
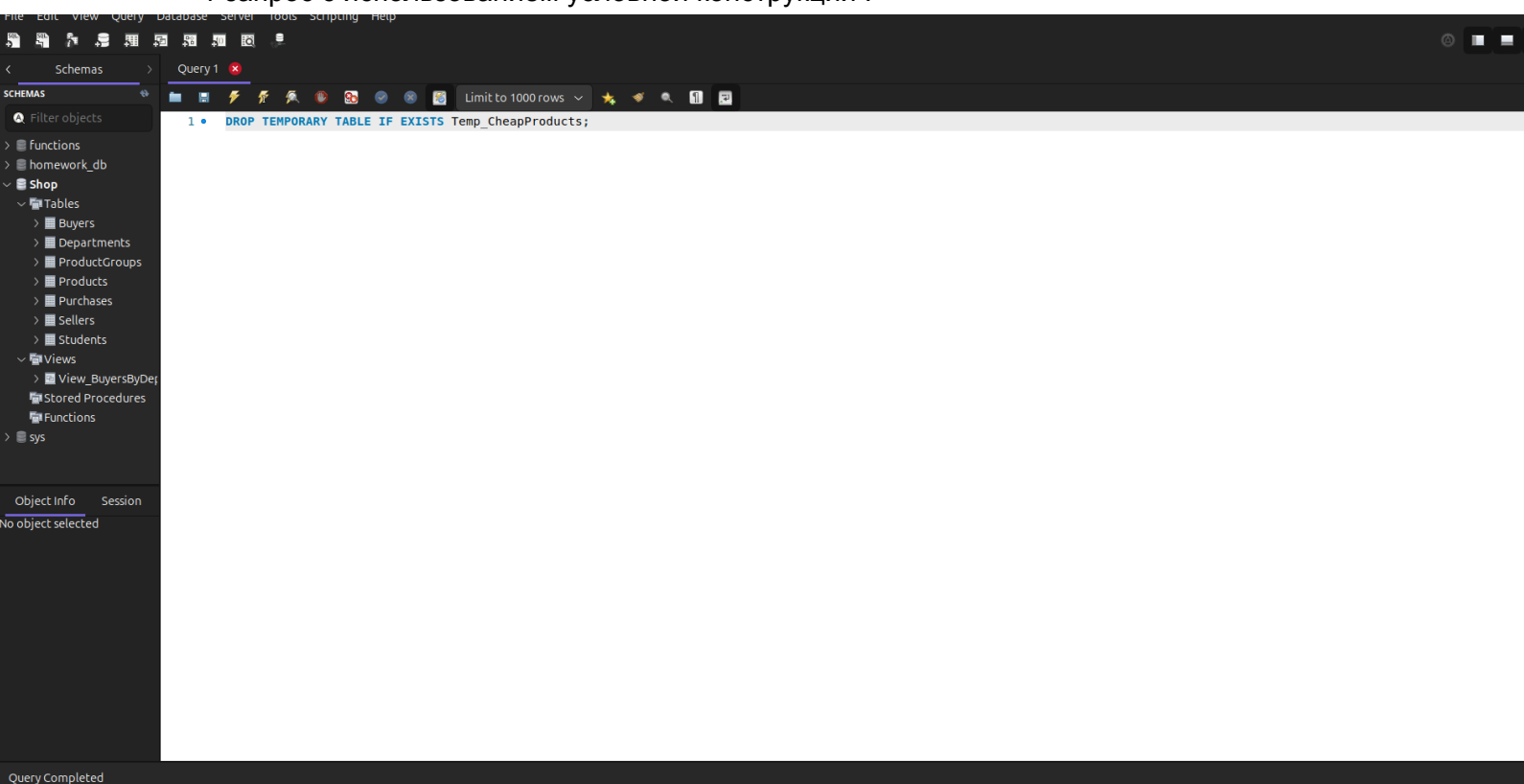


1 запрос для создания временной таблицы через переменную типа TABLE



- 1 запрос с использованием условной конструкции IF



- 1 запрос с использованием цикла WHILE;;

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'SCHEMAS' tree with 'Shop' expanded, showing tables like Buyers, Departments, ProductGroups, Products, Purchases, Sellers, and Students. The main query window, titled 'Query 1', contains the following SQL code:

```
1
2 DELIMITER //
3
4 • CREATE PROCEDURE SimpleWhile()
5 • BEGIN
6 •     DECLARE i INT DEFAULT 1;
7 •
8 •     WHILE i <= 3 DO
9 •         SELECT i AS 'Число';
10 •        SET i = i + 1;
11 •     END WHILE;
12 •
13 • END //
14
15 DELIMITER ;
16 • CALL SimpleWhile();
17 • DROP PROCEDURE SimpleWhile;
```

The bottom status bar indicates 'Query Completed'. Below the query window, a small table preview shows a single row with the value '3' under the column header 'Число'.

- 1 запрос для создания скалярной функции;

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'SCHEMAS' tree with 'Shop' expanded. The main query window, titled 'Query 1', contains the following SQL code:

```
1 DELIMITER //
2
3 • CREATE FUNCTION GetPriceWithTax(product_id INT)
4 RETURNS DECIMAL(10,2)
5 DETERMINISTIC
6 BEGIN
7     DECLARE final_price DECIMAL(10,2);
8
9     -- 10% налога
10    SELECT Price * 1.10 INTO final_price
11    FROM Products
12    WHERE idProduct = product_id;
13
14    RETURN final_price;
15 END //
16
17
18 DELIMITER ;
```

The bottom status bar indicates 'Query interrupted'.

- 1 запрос для создания процедуры без параметров ;

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Schemas' tree with 'homework\_db' expanded, showing tables like 'Buyers', 'Departments', 'ProductGroups', 'Products', 'Purchases', and 'Sellers'. The 'Sellers' table structure is visible: idSeller (int, PK), Surname (varchar(50)), FirstName (varchar(50)), and Department\_id (int). The main query window, titled 'Query 4', contains the following T-SQL code:

```
1 DELIMITER //
2
3 CREATE PROCEDURE ShowAllSellers()
4 BEGIN
5
6     SELECT * FROM Sellers;
7
8 END //
9
10 DELIMITER ;
11 CALL ShowAllSellers();
```

The bottom pane shows the execution results for 'Result 11', displaying a single row of data from the 'Sellers' table:

#	idSeller	Surname	FirstName	Department_id
1	1	Иванов	Иван	1

The status bar at the bottom indicates 'Query Completed'.

- 1 запрос для создания процедуры с входным параметром;

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Schemas' tree with 'homework\_db' expanded, showing tables like 'Buyers', 'Departments', 'ProductGroups', 'Products', 'Purchases', and 'Sellers'. The 'Sellers' table structure is visible: idSeller (int, PK), Surname (varchar(50)), FirstName (varchar(50)), and Department\_id (int). The main query window, titled 'Query 4', contains the following T-SQL code:

```
1 DELIMITER //
2
3 CREATE PROCEDURE SearchSellersByLetter(IN first_letter VARCHAR(1))
4 BEGIN
5     SELECT * FROM Sellers
6     WHERE Surname LIKE CONCAT(first_letter, '%');
7
8 END //
9
10 DELIMITER ;
11
12 CALL SearchSellersByLetter('И');
```

The bottom pane shows the execution results for 'Result 6', displaying a single row of data from the 'Sellers' table:

#	idSeller	Surname	FirstName	Department_id
1	1	Иванов	Иван	1

The status bar at the bottom indicates 'Query Completed'.

- 1 запрос для создания процедуры с входными параметрами и RETURN;

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Schemas' tree with 'homework\_db' expanded, showing 'Tables' and 'Sellers'. The 'Sellers' table columns are listed: idSeller (int AI PK), Surname (varchar(50)), FirstName (varchar(50)), and Department\_id (int). The main pane shows a SQL query window with the following code:

```
1 DELIMITER //
2
3
4 CREATE PROCEDURE GetSellersCount(OUT result_val INT)
5 BEGIN
6
7     SELECT COUNT(*) INTO result_val FROM Sellers;
8 END //
9
10 DELIMITER ;
11
12
13 CALL GetSellersCount(@total);
14 SELECT @total AS 'Результат RETURN';
```

The bottom pane shows the execution results, indicating that the procedure was executed successfully and returned a result of 12.

- 1 запрос для создания процедуры обновления данных в таблице базы данных UPDATE;

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Schemas' tree with 'homework\_db' expanded, showing 'Tables' and 'Sellers'. The 'Sellers' table columns are listed: idSeller (int AI PK), Surname (varchar(50)), FirstName (varchar(50)), and Department\_id (int). The main pane shows a SQL query window with the following code:

```
1 DELIMITER //
2
3
4 CREATE PROCEDURE UpdateSellerFirstName(IN s_id INT, IN new_name VARCHAR(50))
5 BEGIN
6
7     UPDATE Sellers
8     SET FirstName = new_name
9     WHERE idSeller = s_id;
10 END //
11
12 DELIMITER ;
13
14 CALL UpdateSellerFirstName(1, 'Алешенька');
15 SELECT * FROM Sellers;
16
17
18
19
20
21
22 DROP PROCEDURE IF EXISTS UpdateSellerFirstName;
```

The bottom pane shows the execution results, indicating that the procedure was executed successfully and returned a result of 14.

1 запрос для создания процедуры извлечения данных из таблиц базы данных SELECT;

Local instance 3306 - Warning - not supported

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

Functions

homework\_db

Shop

Tables

Buyers

Departments

ProductGroups

Products

Purchases

Sellers

Columns

idSeller

Surname

FirstName

Department\_id

Indexes

Foreign Keys

Triggers

Object Info Session

Table: Sellers

Columns:

idSeller int AI PK

Surname varchar(50)

FirstName varchar(50)

Department\_id int

Query 4

```
1 DELIMITER //
2
3 CREATE PROCEDURE GetAllSellers()
4 BEGIN
5
6     SELECT * FROM Sellers;
7 END //
8
9 DELIMITER ;
10
11 CALL GetAllSellers();
```

Result Grid

Filter Rows

Export: Wrap Cell Content

#	idSeller	Surname	FirstName	Department_id
1	1	Иванов	Алеше...	1

Result 15

Read Only

Query Completed