

Лабораторная работа № 5.

Разработка запросов с подзапросами, Case-выражения, табличные выражения

Подготовительные действия

1. Запустите утилиту SQL Server Management Studio, в окне Обозреватель объектов (*Object Explorer*) в древовидной структуре раскройте папку Базы данных (*Databases*).

2. С помощью команды меню *File - Open - File* загрузите сценарий из файла D:\FIO\TRADE_XXX\CreateDB_Insert_Data.sql в Окно запросов (*Query*).

3. Выполните сценарий, нажав на панели инструментов кнопку Выполнить (*Execute*) (или клавишу F5). В результате будет создана база данных TRADE_XXX.

4. Обновите данные в окне Обозреватель объектов (*Object Explorer*). Для этого используйте команду Обновить (*Refresh*) в контекстном меню папки Базы данных (*Databases*) или соответствующую кнопку в верхней части окна. В результате база данных TRADE_XXX станет видимой в окне Обозреватель объектов (*Object Explorer*).

5. Закройте Окно запросов (*Query*), содержащее сценарий CreateDB_Insert_Data.sql. Затем на панели инструментов нажмите кнопку Новый запрос (*New Query*), и откройте новое пустое Окно запросов (*Query*), предназначенное для формирования нового сценария SubQuery.sql. Сделайте активной созданную базу данных TRADE_XXX:

```
USE TRADE_XXX
```

```
GO
```

Практические задания

Задание 1. Разработка запросов с подзапросами

1. Выведите названия всех предприятий, расположенных в том же регионе, что и предприятие господина Васько Г.Т. (Рисунок 1Рисунок 72).

```

SELECT Clients.ClientName
FROM Clients
WHERE Clients.Location =
(SELECT Clients.Location FROM Clients
WHERE Clients.HeadFullName =
'Васько Григорий Терентьевич')

```

	ClientName
1	ИП "Темп"
2	ИП "Темпера"
3	ОАО "Старт"

Рисунок 72. Запрос с однострочным подзапросом.

2. Выведите коды заказов и наименование товаров, количество которых в заказе больше среднего.

```

SELECT Orders.OrderID ,Products.ProductName
FROM Orders INNER JOIN Products
ON Products.ProductID = Orders.Product
WHERE Quantity >
(SELECT AVG(Quantity) FROM Orders)

```

	OrderID	ProductName
1	3	Винчестер HDD 1100GB
2	4	Монитор 24"
3	9	Монитор 24"

Рисунок 73. Запрос с групповой функцией в подзапросе.

3. Выведите названия и ФИО руководителей всех предприятий, расположенных в стране Беларусь (Рисунок 74).

```

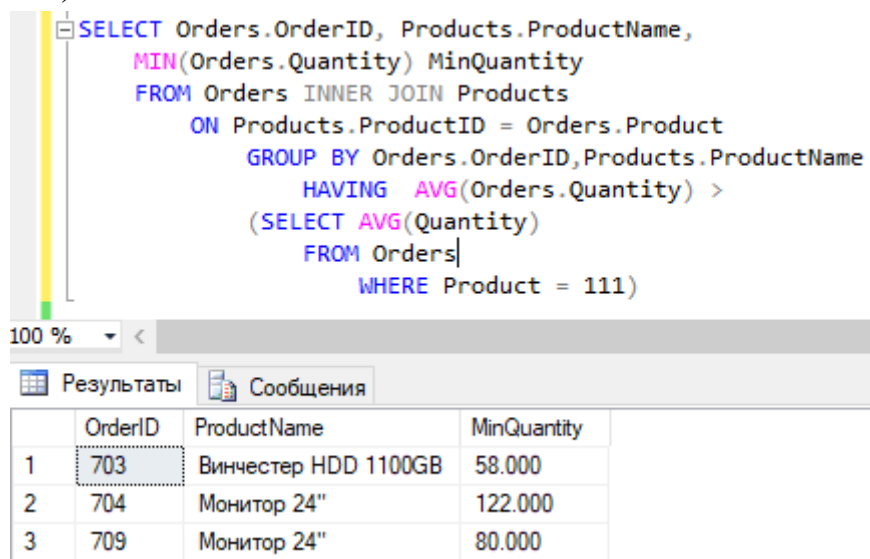
SELECT Clients.ClientName, Clients.HeadFullName
FROM Clients
WHERE Clients.Location IN
(SELECT Locations.LocationID
FROM Locations
WHERE Country = 'Беларусь')

```

	ClientName	HeadFullName
1	ИП "Темп"	Васько Григорий Терентьевич
2	УП "Вера"	Прокуш Станислав Игоревич
3	ИП "Темпера"	Вась Григорий Терентьевич
4	ОАО "Старт"	Кулагин Василий Петрович

Рисунок 74. Запрос с многострочным подзапросом.

4. Выведите код, наименование товара и минимальное количество для товаров, в которых среднее количество в заказе больше, чем у товара с кодом 111 (Рисунок 75).



```

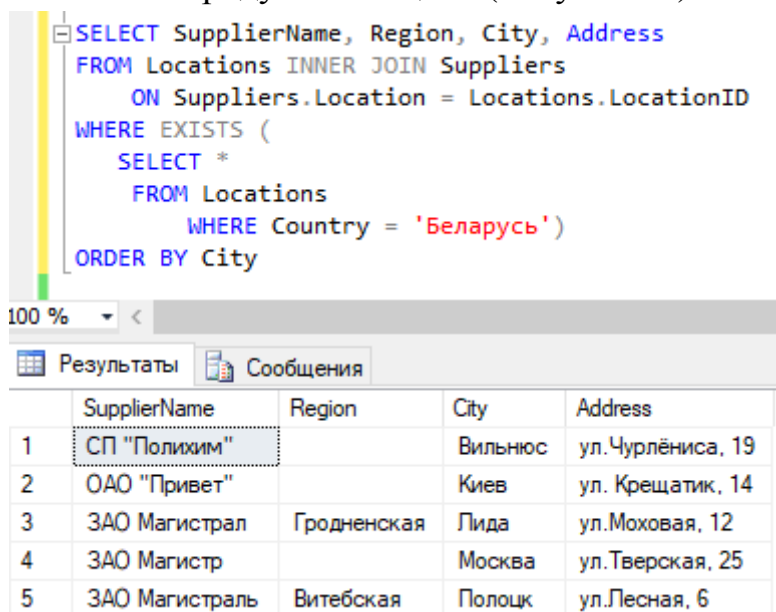
SELECT Orders.OrderID, Products.ProductName,
       MIN(Orders.Quantity) MinQuantity
FROM Orders INNER JOIN Products
      ON Products.ProductID = Orders.Product
GROUP BY Orders.OrderID, Products.ProductName
HAVING AVG(Orders.Quantity) >
       (SELECT AVG(Quantity)
        FROM Orders
        WHERE Product = 111)

```

	OrderID	ProductName	MinQuantity
1	703	Винчестер HDD 1100GB	58.000
2	704	Монитор 24"	122.000
3	709	Монитор 24"	80.000

Рисунок 75. Предложение HAVING в запросах с подзапросами.

5. Выведите поля имя поставщика, его регион, адрес, если в таблице Locations существуют записи, в которых название страны Беларусь. Отсортировать записи по городу поставщика (Рисунок 76).



```

SELECT SupplierName, Region, City, Address
FROM Locations INNER JOIN Suppliers
      ON Suppliers.Location = Locations.LocationID
WHERE EXISTS (
  SELECT *
  FROM Locations
  WHERE Country = 'Беларусь')
ORDER BY City

```

	SupplierName	Region	City	Address
1	СП "Полихим"		Вильнюс	ул.Чурлёниса, 19
2	ОАО "Привет"		Киев	ул. Крещатик, 14
3	ЗАО Магистрал	Гродненская	Лида	ул.Моховая, 12
4	ЗАО Магистр		Москва	ул.Тверская, 25
5	ЗАО Магистраль	Витебская	Полоцк	ул.Лесная, 6

Рисунок 76. Запрос с ключевым словом EXISTS.

6. Определите поставщиков, которые поставляют более одного наименования товара (Рисунок 77).

```

SELECT DISTINCT Supplier, SupplierName
FROM Orders Or1 --alias
INNER JOIN Suppliers
ON Suppliers.SupplierID = Or1.Supplier
WHERE EXISTS (
    SELECT *
    FROM Orders Or2 --alias
    WHERE Or1.Supplier = Or2.Supplier
    AND Or1.Product <> Or2.Product
)

```

	Supplier	SupplierName
1	500	ОАО "Привет"
2	567	СП "Полихим"

Рисунок 77. Запрос со сложным условием в предложении WHERE.

7. Определить клиентов, имеющих поставщиков в своем регионе (Рисунок 78).

```

SELECT *
FROM Clients
WHERE Location = ANY (
    SELECT Location
    FROM Suppliers)

```

	ClientID	ClientName	HeadFullName	Location
1	1	ГП "Верас"	Прокушев Станислав Игоревич	202
2	4	ИП "Темпера"	Вась Григорий Терентьевич	104
3	5	ОАО "Старт"	Кулагин Василий Петрович	103

Рисунок 78. Запрос с ключевым словом ANY.

8. Определить номера заказов и наименование товаров, у которых количество товара больше или равно, чем количество товара в любом заказе поставщика с номером 567 (Рисунок 79).

```

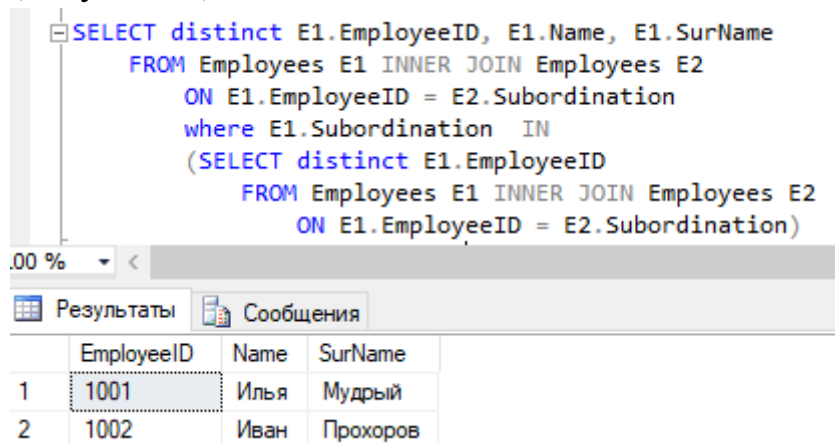
SELECT Orders.OrderID, Products.ProductName, Orders.Quantity
FROM Orders INNER JOIN Products
ON Orders.Product = Products.ProductID
WHERE Quantity >= ALL(
    SELECT Quantity
    FROM Orders
    WHERE Orders.Supplier = 567)

```

	OrderID	ProductName	Quantity
1	703	Винчестер HDD 1100GB	58.000
2	704	Монитор 24"	122.000
3	709	Монитор 24"	80.000

Рисунок 79. Запрос с ключевым словом ANY.

9. Вывести из таблицы Employees: всех руководителей, которые сами являются подчиненными (Рисунок 80) и всех сотрудников, у которых нет подчиненных (Рисунок 81).

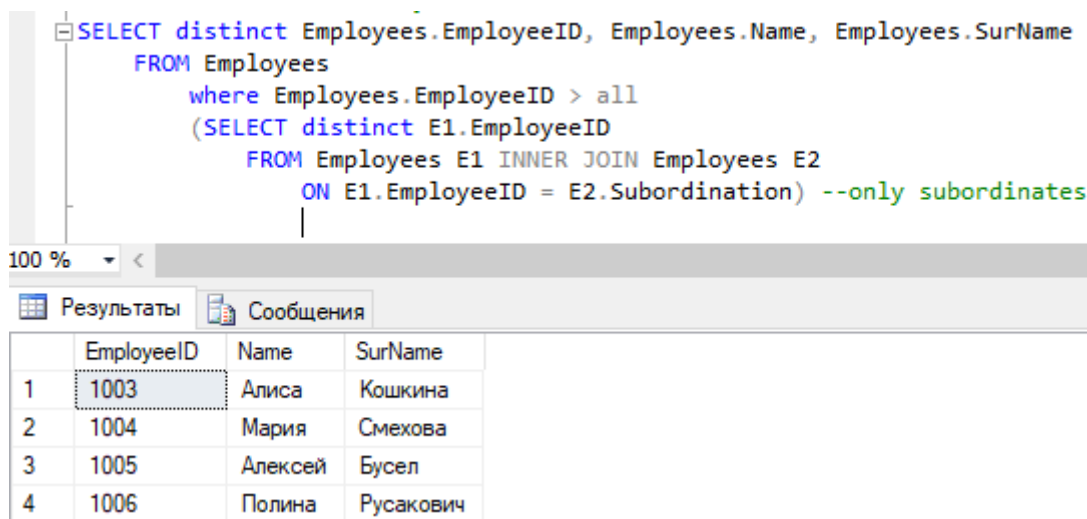


```

SELECT distinct E1.EmployeeID, E1.Name, E1.SurName
FROM Employees E1 INNER JOIN Employees E2
ON E1.EmployeeID = E2.Subordination
where E1.Subordination IN
(SELECT distinct E1.EmployeeID
FROM Employees E1 INNER JOIN Employees E2
ON E1.EmployeeID = E2.Subordination)
  
```

	EmployeeID	Name	SurName
1	1001	Илья	Мудрый
2	1002	Иван	Прохоров

Рисунок 80. Запрос с самосоединением.



```

SELECT distinct Employees.EmployeeID, Employees.Name, Employees.SurName
FROM Employees
where Employees.EmployeeID > all
(SELECT distinct E1.EmployeeID
FROM Employees E1 INNER JOIN Employees E2
ON E1.EmployeeID = E2.Subordination) --only subordinates
  
```

	EmployeeID	Name	SurName
1	1003	Алиса	Кошкина
2	1004	Мария	Сметова
3	1005	Алексей	Бусел
4	1006	Полина	Русакович

Рисунок 81. Запрос с самосоединением.

Задание 2. Разработка запросов с выражением CASE

1. Определить категорию заказа в зависимости от его стоимости (Рисунок 82):

- 1, если стоимость заказа < 2500 руб.
- 2, если стоимость заказа >= 2500 руб. и < 5000 руб.
- 3, если стоимость заказа >= 5000 руб. и < 7500 руб.
- 4, если стоимость заказа >= 7500 руб.

Используется представление OrdersCOST, для него введен псевдоним ОС. Для вывода двух знаков после запятой для значения стоимости заказов применяется функция CAST. Сортировка выполняется по полю номер 3 в списке вывода.

```

SELECT OC.ProductName, OC.OrderDate, CAST(OC.COST as numeric(9, 2)) OrderCOST,
CASE
    WHEN OC.COST < 2500 THEN 1
    WHEN OC.COST < 5000 THEN 2
    WHEN OC.COST < 7500 THEN 3
    ELSE 4
END OrderCategory
FROM OrdersCost OC
ORDER BY 3

```

	ProductName	OrderDate	OrderCOST	OrderCategory
1	Монитор 24"	2017-09-06	1990.00	1
2	Монитор 24"	2017-04-12	2786.00	2
3	Винчестер HDD 1100GB	2017-05-30	2999.75	2
4	Винчестер HDD 1100GB	2017-04-19	6959.42	3
5	Монитор 24"	2017-04-24	7363.00	3
6	Винчестер HDD 4 TB	2017-04-04	7497.50	3
7	Винчестер HDD 4 TB	2017-05-09	8097.30	4
8	Монитор 24"	2017-04-04	15920.00	4
9	Монитор 24"	2017-04-04	24278.00	4

Рисунок 82. Запрос с выражением CASE.

2. Определить для каждого заказа: его стоимость больше, равна или меньше средней стоимости всех заказов.

```

SELECT OC1.ProductName, OC1.OrderDate, CAST(OC1.COST as numeric(9, 2)) OrderCOST,
CASE
    WHEN OC1.COST < (SELECT AVG(OC2.COST) FROM OrdersCost OC2)
    THEN 'below average'
    WHEN OC1.COST = (SELECT AVG(OC2.COST) FROM OrdersCost OC2)
    THEN 'on average'
    ELSE 'above average'
END OrderCategory
FROM OrdersCost OC1
ORDER BY 3

```

	ProductName	OrderDate	OrderCOST	OrderCategory
1	Монитор 24"	2017-09-06	1990.00	below average
2	Монитор 24"	2017-04-12	2786.00	below average
3	Винчестер HDD 1100GB	2017-05-30	2999.75	below average
4	Винчестер HDD 1100GB	2017-04-19	6959.42	below average
5	Монитор 24"	2017-04-24	7363.00	below average
6	Винчестер HDD 4 TB	2017-04-04	7497.50	below average
7	Винчестер HDD 4 TB	2017-05-09	8097.30	below average
8	Монитор 24"	2017-04-04	15920.00	above average
9	Монитор 24"	2017-04-04	24278.00	above average

Рисунок 83. Применение выражения CASE с вложенными запросами.

3. Уменьшить стоимость заказа в зависимости от его стоимости:

- на 5 %, если стоимость заказа ≥ 5000 руб. и ≤ 7500 руб.
- На 10 %, если стоимость заказа ≥ 7500 руб.

Создадим новую таблицу T_OrderCOST, в которую поместим все записи из представления OrderCOST.

```

SELECT * INTO T_OrderCOST FROM OrderCOST

SELECT * FROM T_OrderCOST
-----
UPDATE T_OrderCOST
SET T_OrderCOST.COST = CASE
    WHEN COST >= 5000 AND COST < 7500 THEN COST*0.95
    WHEN COST > 7500 THEN COST*0.90
    ELSE COST
END
-----
SELECT * FROM T_OrderCOST
-----

```

00 % <

Результаты Сообщения

	ClientName	ProductName	OrderDate	DeliveryDate	COST
1	ГП "Верас"	Монитор 24"	2017-04-12	2017-05-03	2786.0000000
2	ИП "Темп"	Винчестер HDD 4 TB	2017-05-09	2017-05-25	8097.3000000
3	УП "Вера"	Винчестер HDD 1100GB	2017-04-19	2017-05-03	6959.4200000
4	ИП "Темп"	Монитор 24"	2017-04-04	2017-05-03	24278.0000000
5	ИП "Темпера"	Винчестер HDD 4 TB	2017-04-04	2017-05-10	7497.5000000
6	УП "Вера"	Монитор 24"	2017-04-24	2017-05-10	7363.0000000
7	ИП "Темп"	Монитор 24"	2017-09-06	2017-09-25	1990.0000000
8	ОАО "Рога и копыта"	Винчестер HDD 1100GB	2017-05-30	2017-05-15	2999.7500000
9	ИП "Темп"	Монитор 24"	2017-04-04	2017-04-14	15920.0000000

	ClientName	ProductName	OrderDate	DeliveryDate	COST
1	ГП "Верас"	Монитор 24"	2017-04-12	2017-05-03	2786.0000000
2	ИП "Темп"	Винчестер HDD 4 TB	2017-05-09	2017-05-25	7287.5700000
3	УП "Вера"	Винчестер HDD 1100GB	2017-04-19	2017-05-03	6611.4490000
4	ИП "Темп"	Монитор 24"	2017-04-04	2017-05-03	21850.2000000
5	ИП "Темпера"	Винчестер HDD 4 TB	2017-04-04	2017-05-10	7122.6250000
6	УП "Вера"	Монитор 24"	2017-04-24	2017-05-10	6994.8500000
7	ИП "Темп"	Монитор 24"	2017-09-06	2017-09-25	1990.0000000
8	ОАО "Рога и копыта"	Винчестер HDD 1100GB	2017-05-30	2017-05-15	2999.7500000
9	ИП "Темп"	Монитор 24"	2017-04-04	2017-04-14	14328.0000000

Рисунок 84. Использование выражения CASE в инструкции UPDATE.

Задание 3. Разработка запросов с табличными выражениями

1. Выбрать все группы месяцев в столбце DeliveryDate таблицы Orders (Рисунок 85).


```

SELECT DeliveryMonth
FROM (SELECT MONTH(DeliveryDate) DeliveryMonth
      FROM Orders) OM
GROUP BY DeliveryMonth

```

100 % <

Результаты Сообщения

	DeliveryMonth
1	4
2	5
3	9

Рисунок 85. Запрос с табличным выражением в предложении FROM.

2. Выбрать наименования товаров и название валюты, у которых код валюты равен EUR или BYN.

```

SELECT P.ProductName,
      (SELECT C.CurrencyName
       FROM Currency C
       WHERE C.CurrencyID = P.Currency) CurrencyName
FROM Products P
WHERE P.Currency IN ('EUR', 'BYN')

```

100 % <

Результаты Сообщения

	ProductName	CurrencyName
1	Кабель	Евро
2	Винчестер HDD 4 TB	БелРубНов
3	Винчестер HDD 1100GB	БелРубНов

Рисунок 86. Табличное выражение в списке выбора.

Задания для самостоятельного выполнения

Задание 4. Разработка запросов с подзапросами

1. Выведите названия всех поставщиков, расположенных в том же районе, что и предприятие господина Бендера О.
2. Выведите наименования тех поставщиков и их адреса, у которых стоимость товаров в заказе ниже средней.
3. Выведите наименования поставщиков и их адреса, расположенных в стране Россия и Беларусь.
4. Выведите фамилию, имя, должность сотрудников и среднее число товаров в заказе, у которых в среднее число товаров в заказе больше, чем у сотрудника с кодом 1005.
5. Выведите поля Country, Region, City из таблицы Locations, если в таблице Locations не существуют записи, в которых название страны Польша.

6. Определите поставщиков, которые поставляют только одно наименование товара.

7. Определите заказы, у которых количество товара МЕНЬШЕ, чем количество товара в любом заказе клиента с определенным номером.

8. Измените в таблице `Clients` поле `Location` на `NULL` для регионов страны Украина. Используйте запрос на обновление с подзапросом с многострочным подзапросом, который определяет коды регионов страны Украина.

Задание 5. Разработка запросов с выражением CASE

1. Определите категорию товара в зависимости от его цены. Диапазоны цен для категорий подберите самостоятельно.

2. Определите для каждого товара: его цена больше, равна или меньше средней цены всех товаров.

3. Увеличьте цену заказа в зависимости от его цены. Диапазоны цен для увеличения подберите самостоятельно.

Задание 6. Разработка запросов с табличными выражениями

1. Выберите все группы городов в столбце `City` таблицы `Locations`.

2. Выберите клиентов и название городов, которые находятся в регионах с кодами 101 и 102.

3. Выбрать из таблицы `Orders` поля `OrderID` и `OrderDate` для всех заказов, у которых дата заказа является минимальной. Используйте подзапрос в предложении `WHERE` запроса.

Сохраните все команды, разработанные на языке SQL, в файле сценария с именем `D:\FIO\TRADE_XXX\SubQuery.sql`.

Затем удалите базу данных `TRADE_XXX`. Для этого в ее контекстном меню выберите команду *Удалить (Delete)* и затем в появившемся окне *Удаление объекта (Delete Object)* установите флажок *Закреть существующие соединения (Close Existing Connections)*.