

Министерство образования Республики Беларусь

Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Факультет компьютерного проектирования

Кафедра экономической информатики

ОТЧЁТ

Лабораторная работа №2 – 4

«Дополнение к Лаб.раб.2-4, к УСРС1 по индивидуальному заданию
студентов. Работа с БД, спроектированной ранее.»

Проверил: Мельников Дмитрий Васильевич

Выполнил: Ткачук Алексей Иванович

Студентка группы № 014301

Минск 2023

1. Предметная область:

Ремонт бытовой техники

Сервисный центр осуществляет ремонт и гарантийное обслуживание бытовой техники фирм-производителей Philips, Brown, Bosh.

Клиенты осуществляют заказы на ремонт товаров по гарантии и без нее. Сотрудники центра специализируются на ремонте и обслуживании отдельных товаров и выполняют соответствующие заказы.

В день исполнения заказа сервисный центр сообщает об этом клиенту. Срок бесплатного хранения отремонтированного изделия в сервисном центре составляет один месяц. После его истечения клиент лишается права бесплатного гарантийного ремонта изделия и оплачивает затраты центра на хранение данного товара (5 % стоимости ремонта за каждый дополнительный день).

Необходимо спроектировать БД Ремонт бытовой техники, информация которой будет использоваться для учета услуг по ремонту и гарантийному обслуживанию товаров, анализа сроков исполнения заказов, видов неисправностей и др.

В БД должна храниться информация: о товарах; сотрудниках; заказах; исполнениях заказов.

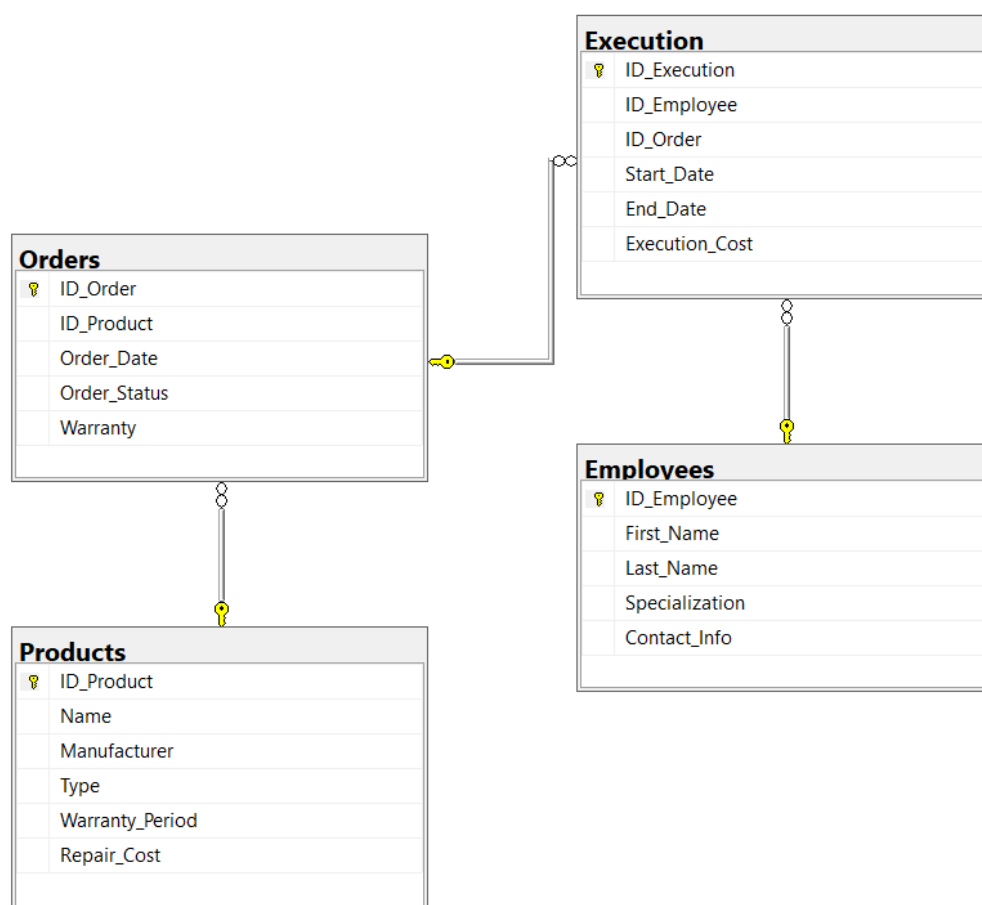
При проектировании БД необходимо учитывать следующее:

- товар может быть отремонтирован несколькими сотрудниками. Сотрудник может выполнять ремонт нескольких товаров;
- товар может быть связан с несколькими заказами. Заказ связан с одним товаром;
- заказ соотносится с одним исполнением. Исполнение связано с одним заказом;
- сотрудник осуществляет несколько исполнений заказов. Исполнение заказа связано с одним сотрудником.

Кроме того, следует учесть:

- каждый товар обязательно может быть отремонтирован сотрудниками. Каждый сотрудник обязательно выполняет ремонт товаров;
- товар не обязательно может быть связан с заказами. Каждый заказ обязательно связан с товаром;
- заказ не обязательно подлежит исполнению. Исполнение обязательно связано с заказом;
- сотрудник не обязательно осуществляет исполнения заказов. Каждое исполнение заказа обязательно связано с сотрудником.

2. Структуры таблиц, данные в таблицах, диаграмма баз данных:



DESKTOP-7L6U0EU\...ки - dbo.Products		DESKTOP-7L6U0EU\...ники - dbo.Orders	
	Column Name	Data Type	Allow Nulls
PK	ID_Product	int	<input type="checkbox"/>
	Name	nvarchar(100)	<input checked="" type="checkbox"/>
	Manufacturer	nvarchar(50)	<input checked="" type="checkbox"/>
	Type	nvarchar(50)	<input checked="" type="checkbox"/>
	Warranty_Period	int	<input checked="" type="checkbox"/>
	Repair_Cost	decimal(10, 2)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DESKTOP-7L6U0EU\...ники - dbo.Orders		DESKTOP-7L6U0EU\...и - dbo.Execution	
	Column Name	Data Type	Allow Nulls
PK	ID_Order	int	<input type="checkbox"/>
	ID_Product	int	<input checked="" type="checkbox"/>
	Order_Date	date	<input checked="" type="checkbox"/>
	Order_Status	nvarchar(50)	<input checked="" type="checkbox"/>
	Warranty	bit	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DESKTOP-7L6U0EU\...и - dbo.Execution		DESKTOP-7L6U0EU\...- dbo.Employees	
	Column Name	Data Type	Allow Nulls
PK	ID_Execution	int	<input type="checkbox"/>
	ID_Employee	int	<input checked="" type="checkbox"/>
	ID_Order	int	<input checked="" type="checkbox"/>
	Start_Date	date	<input checked="" type="checkbox"/>
	End_Date	date	<input checked="" type="checkbox"/>
	Execution_Cost	decimal(10, 2)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DESKTOP-7L6U0EU\...- dbo.Employees		DESKTOP-7L6U0EU\...ники - Diagram_0*	
	Column Name	Data Type	Allow Nulls
PK	ID_Employee	int	<input type="checkbox"/>
	First_Name	nvarchar(50)	<input checked="" type="checkbox"/>
	Last_Name	nvarchar(50)	<input checked="" type="checkbox"/>
	Specialization	nvarchar(100)	<input checked="" type="checkbox"/>
	Contact_Info	nvarchar(100)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Results Messages						
	ID_Employee	First_Name	Last_Name	Specialization	Contact_Info	
1	1	Иван	Иванов	Техник	ivan.ivanov@example.com	
2	2	Ольга	Смирнова	Служба поддержки	olga.smirnova@example.com	
3	3	Михаил	Петров	Инженер	mikhail.petrov@example.com	
4	4	Елена	Сидорова	Техник	elena.sidorova@example.com	
5	5	Андрей	Козлов	Техник	andrey.kozlov@example.com	
	ID_Execution	ID_Employee	ID_Order	Start_Date	End_Date	Execution_Cost
1	1001	1	101	2023-10-25	2023-10-26	100.00
2	1002	2	102	2023-10-26	2023-10-27	80.00
3	1003	3	103	2023-10-27	2023-10-28	120.00
4	1004	4	104	2023-10-28	2023-10-29	130.00
5	1005	5	105	2023-10-29	2023-10-30	90.00
	ID_Order	ID_Product	Order_Date	Order_Status	Warranty	
1	101	1	2023-10-25	Ожидает выполнения	1	
2	102	2	2023-10-26	В процессе	0	
3	103	3	2023-10-27	Завершен	1	
4	104	1	2023-10-28	Ожидает выполнения	1	
5	105	2	2023-10-29	В процессе	0	
	ID_Product	Name	Manufacturer	Type	Warranty_Period	Repair_Cost
1	1	Стиральная машина	Philips	Бытовая техника	12	150.00
2	2	Кофеварка	Bosch	Бытовая техника	24	80.00
3	3	Тостер	Brown	Бытовая техника	12	30.00
4	4	Холодильник	Samsung	Бытовая техника	36	250.00
5	5	Микроволновая пе...	Panasonic	Бытовая техника	24	120.00

3. T-SQL запросы:
- 1) Вывести все данные таблиц:

```
1 -- 3.1. Вывести все данные из таблиц
2 SELECT * FROM Employees;
3 SELECT * FROM Orders;
4 SELECT * FROM Products;
5 SELECT * FROM Execution;
```

81 %

Results Messages

	ID_Employee	First_Name	Last_Name	Specialization	Contact_Info
1	1	Иван	Иванов	Техник	ivan.ivanov@example.com
2	2	Ольга	Смирнова	Служба поддержки	olga.smirnova@example.com
3	3	Михаил	Петров	Инженер	mikhail.petrov@example.com
4	4	Елена	Сидорова	Техник	elena.sidorova@example.com
5	5	Андрей	Козлов	Техник	andrey.kozlov@example.com

	ID_Order	ID_Product	Order_Date	Order_Status	Warranty
1	101	1	2023-10-25	Ожидает выполнения	1
2	102	2	2023-10-26	В процессе	0
3	103	3	2023-10-27	Завершен	1
4	104	1	2023-10-28	Ожидает выполнения	1
5	105	2	2023-10-29	В процессе	0

	ID_Product	Name	Manufacturer	Type	Warranty_Period	Repair_Cost
1	1	Стиральная машина	Philips	Бытовая техника	12	150.00
2	2	Кофеварка	Bosch	Бытовая техника	24	80.00
3	3	Тостер	Brown	Бытовая техника	12	30.00
4	4	Холодильник	Samsung	Бытовая техника	36	250.00
5	5	Микроволновая пе...	Panasonic	Бытовая техника	24	120.00

	ID_Execution	ID_Employee	ID_Order	Start_Date	End_Date	Execution_Cost
1	1001	1	101	2023-10-25	2023-10-26	100.00
2	1002	2	102	2023-10-26	2023-10-27	80.00
3	1003	3	103	2023-10-27	2023-10-28	120.00
4	1004	4	104	2023-10-28	2023-10-29	130.00
5	1005	5	105	2023-10-29	2023-10-30	90.00

- 2) Вывести определенные данные, отсортированные в алфавитном порядке по определенному полю

```
7 -- 3.2. Вывести определенные данные, отсортированные в алфавитном порядке по определенному полю
8 -- (Вывести сотрудников отсортированных по имени)
9 SELECT * FROM Employees ORDER BY First_Name;
10
```

81 %

Results Messages

	ID_Employee	First_Name	Last_Name	Specialization	Contact_Info
1	5	Андрей	Козлов	Техник	andrey.kozlov@example.com
2	4	Елена	Сидорова	Техник	elena.sidorova@example.com
3	1	Иван	Иванов	Техник	ivan.ivanov@example.com
4	3	Михаил	Петров	Инженер	mikhail.petrov@example.com
5	2	Ольга	Смирнова	Служба поддержки	olga.smirnova@example.com

3) Вывести определенные данные, отсортированные в обратном алфавитном порядке по двум полям

```
11 -- 3.3. Вывести определенные данные, отсортированные в обратном алфавитном порядке по двум полям
12 -- (Вывести заказы отсортированные по дате заказа в обратном порядке и по статусу заказа в обратном порядке)
13 SELECT * FROM Orders ORDER BY Order_Date DESC, Order_Status DESC;
14
```

81 %

	ID_Order	ID_Product	Order_Date	Order_Status	Warranty
1	105	2	2023-10-29	В процессе	0
2	104	1	2023-10-28	Ожидает выполнения	1
3	103	3	2023-10-27	Завершен	1
4	102	2	2023-10-26	В процессе	0
5	101	1	2023-10-25	Ожидает выполнения	1

4) Создать вычисляемое поле, и вывести данные, убрав дубликаты

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 SELECT DISTINCT
2     Specialization + ' (' + Contact_Info + ')' AS 'Сотрудник(контактный номер)'
3 FROM
4     Employees;
```

81 %

	Сотрудник(контактный номер)
1	Инженер (mikhail.petrov@example.com)
2	Служба поддержки (olga.smirnova@example.com)
3	Техник (andrey.kozlov@example.com)
4	Техник (elena.sidorova@example.com)
5	Техник (ivan.ivanov@example.com)

5) Вывести 30% строк

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 -- 3.5. Вывести 30% строк
2 -- 30% сотрудников
3 SELECT TOP 30 PERCENT * FROM Employees;
```

81 %

	ID_Employee	First_Name	Last_Name	Specialization	Contact_Info
1	1	Иван	Иванов	Техник	ivan.ivanov@example.com
2	2	Ольга	Смирнова	Служба поддержки	olga.smirnova@example.com
3	3	Михаил	Петров	Инженер	mikhail.petrov@example.com

6) Вывести первые 5 строк

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 -- 3.6. Вывести первые 5 строк
2 -- (первые 5 заказов)
3 SELECT TOP 5 * FROM Orders;
4
```

81 %

Results Messages

	ID_Order	ID_Product	Order_Date	Order_Status	Warranty
1	101	1	2023-10-25	Ожидает выполнения	1
2	102	2	2023-10-26	В процессе	0
3	103	3	2023-10-27	Завершен	1
4	104	1	2023-10-28	Ожидает выполнения	1
5	105	2	2023-10-29	В процессе	0

7) Вывести строки начиная с 4 строки по 7 строку, отсортированные в обратном порядке

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 -- 3.7. Вывести строки начиная с 4 строки по 7 строку, отсортированные в обратном порядке
2 -- исполнения с 4 по 7 отсортированные по ID исполнения
3 SELECT * FROM (SELECT *, ROW_NUMBER() OVER (ORDER BY ID_Execution DESC) AS RowNum FROM Execution) AS SubQuery
4 WHERE RowNum BETWEEN 4 AND 7;
```

81 %

Results Messages

	ID_Execution	ID_Employee	ID_Order	Start_Date	End_Date	Execution_Cost	RowNum
1	1007	7	107	2023-10-31	2023-11-01	80.00	4
2	1006	6	106	2023-10-30	2023-10-31	90.00	5
3	1005	5	105	2023-10-29	2023-10-30	90.00	6
4	1004	4	104	2023-10-28	2023-10-29	130.00	7

8) Отсортировать данные, используя в качестве критерия сортировки сложное выражение на основе столбцов

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 -- 3.8. Отсортировать данные, используя в качестве критерия сортировки сложное выражение на основе столбцов
2 -- заказы, отсортированные по стоимости ремонта (убывает) и по типу товара (возрастает)
3 SELECT * FROM Orders
4 INNER JOIN Products ON Orders.ID_Product = Products.ID_Product
5 ORDER BY Products.Repair_Cost DESC, Products.Type ASC;
6
```

81 %

Results Messages

	ID_Execution	ID_Employee	ID_Order	Start_Date	End_Date	Execution_Cost	RowNum
1	1007	7	107	2023-10-31	2023-11-01	80.00	4
2	1006	6	106	2023-10-30	2023-10-31	90.00	5
3	1005	5	105	2023-10-29	2023-10-30	90.00	6
4	1004	4	104	2023-10-28	2023-10-29	130.00	7

9) Запрос на выборку с добавлением SELECT INTO

```
1 SELECT ID_Product, Name, Repair_Cost INTO ExpensiveRepairs
2 FROM Products
3 WHERE Repair_Cost > 100.00;
```

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 SELECT * FROM ExpensiveRepairs
```

81 %

Results Messages

	ID_Product	Name	Repair_Cost
1	1	Стиральная машина	150.00
2	4	Холодильник	250.00
3	5	Микроволновая печь	120.00
4	9	Телевизор	600.00
5	10	Ноутбук	800.00

10) С помощью языка T-SQL написать и выполнить 7 запросов на фильтрацию данных с использованием в запросах операторов сравнения, логических операторов:

- **Товары samsung с гарантией более 12 дней:**

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 SELECT * FROM Products WHERE Warranty_Period > 12 AND Manufacturer = 'Samsung';
2
```

81 %

Results Messages

	ID_Product	Name	Manufacturer	Type	Warranty_Period	Repair_Cost
1	4	Холодильник	Samsung	Бытовая техника	36	250.00
2	9	Телевизор	Samsung	Электроника	36	600.00

- **Все исполнения заказов с датой начала после 2023-10-28 и стоимостью исполнения менее 120.00:**

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 SELECT * FROM Execution WHERE Start_Date > '2023-10-28' AND Execution_Cost < 120.00;
```

81 %

Results Messages

	ID_Execution	ID_Employee	ID_Order	Start_Date	End_Date	Execution_Cost
1	1005	5	105	2023-10-29	2023-10-30	90.00
2	1006	6	106	2023-10-30	2023-10-31	90.00
3	1007	7	107	2023-10-31	2023-11-01	80.00
4	1008	8	108	2023-11-01	2023-11-02	70.00
5	1010	10	110	2023-11-03	2023-11-04	100.00

- сотрудники, специализирующиеся как "Техник" или "Инженер":

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))

```
1 SELECT * FROM Employees WHERE Specialization IN ('Техник', 'Инженер');
```

81 %

Results Messages

	ID_Employee	First_Name	Last_Name	Specialization	Contact_Info
1	1	Иван	Иванов	Техник	ivan.ivanov@example.com
2	3	Михаил	Петров	Инженер	mikhail.petrov@example.com
3	4	Елена	Сидорова	Техник	elena.sidorova@example.com
4	5	Андрей	Козлов	Техник	andrey.kozlov@example.com
5	6	Мария	Смирнова	Инженер	maria.smirnova@example.com
6	7	Александр	Иванов	Техник	alexander.ivanov@example.com
7	8	Екатерина	Соколова	Техник	ekaterina.sokolova@example.com
8	10	Анна	Кузнецова	Инженер	anna.kuznetsova@example.com

- заказы, выполненные не в гарантийный период:

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 SELECT * FROM Orders WHERE Warranty = 0;
```

81 %

Results Messages

	ID_Order	ID_Product	Order_Date	Order_Status	Warranty
1	102	2	2023-10-26	В процессе	0
2	105	2	2023-10-29	В процессе	0
3	107	4	2023-10-31	Ожидает выполнения	0
4	110	2	2023-11-03	Ожидает выполнения	0

- товары с гарантийным периодом больше 12 месяцев и стоимостью ремонта не более 100.00:

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 SELECT * FROM Products WHERE Warranty_Period > 12 AND Repair_Cost <= 100.00;
```

81 %

Results Messages

	ID_Product	Name	Manufacturer	Type	Warranty_Period	Repair_Cost
1	2	Кофеварка	Bosch	Бытовая техника	24	80.00
2	7	Утюг	Bosch	Утюги	24	45.00

- **всех сотрудников с именем "Иван":**

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))*

```

1 SELECT * FROM Employees WHERE First_Name = 'Иван';
2

```

81 %

Results Messages

	ID_Employee	First_Name	Last_Name	Specialization	Contact_Info
1	1	Иван	Иванов	Техник	ivan.ivanov@example.com

- **все заказы, сделанные после 2023-10-27:**

DESKTOP-7L6U0EU\...ки - dbo.Products SQLQuery4.sql - DE...йТехники (sa (53))*

```

1 SELECT * FROM Orders WHERE Order_Date > '2023-10-27';
2

```

81 %

Results Messages

	ID_Order	ID_Product	Order_Date	Order_Status	Warranty
1	104	1	2023-10-28	Ожидает выполнения	1
2	105	2	2023-10-29	В процессе	0
3	106	3	2023-10-30	Завершен	1
4	107	4	2023-10-31	Ожидает выполнения	0
5	108	5	2023-11-01	В процессе	1
6	109	1	2023-11-02	Завершен	1
7	110	2	2023-11-03	Ожидает выполнения	0

11) – 12) Переименовать одну из таблиц в базе данных (Product), добавить по 2 записи в каждую таблицу:

```
DESKTOP-7L6U0EU\...ки - dbo.Products | SQLQuery4.sql - DE...йТехники (sa (53)) *
1 use РемонтБытовойТехники
2 -- Переименование таблицы Products в ProductsNew
3 EXEC sp_rename 'Products', 'ProductsNew';
4
5 -- Добавление новых значений в таблицу ProductsNew
6 INSERT INTO ProductsNew (ID_Product, Name, Manufacturer, Type, Warranty_Period, Repair_Cost)
7 VALUES
8     (11, 'Холодильник', 'Samsung', 'Бытовая техника', 36, 600.00),
9     (12, 'Пылесос', 'Dyson', 'Уборочная техника', 24, 400.00);
10
11 -- Добавление новых значений в таблицу Employees
12 INSERT INTO Employees (ID_Employee, First_Name, Last_Name, Specialization, Contact_Info)
13 VALUES
14     (11, 'Николай', 'Смирнов', 'Техник', 'nikolay.smirnov@example.com'),
15     (12, 'Ольга', 'Иванова', 'Инженер', 'olga.ivanova@example.com');
16
17 -- Добавление новых значений в таблицу Orders
18 INSERT INTO Orders (ID_Order, ID_Product, Order_Date, Order_Status, Warranty)
19 VALUES
20     (111, 11, '2023-11-08', 'Ожидает выполнения', 1),
21     (112, 12, '2023-11-09', 'В процессе', 0);
22
23 -- Добавление новых значений в таблицу Execution
24 INSERT INTO Execution (ID_Execution, ID_Employee, ID_Order, Start_Date, End_Date, Execution_Cost)
25 VALUES
26     (1011, 11, 111, '2023-11-08', '2023-11-09', 90.00),
27     (1012, 12, 112, '2023-11-09', '2023-11-10', 80.00);
```

81 %

Messages

Caution: Changing any part of an object name could break scripts and stored procedures.

(2 rows affected)

(2 rows affected)

(2 rows affected)

(2 rows affected)

Completion time: 2023-11-04T02:52:11.0954741+03:00

13) Добавление столбца в таблице Orders:

```
DESKTOP-7L6U0EU\...ки - dbo.Products | SQLQuery4.sql - DE...йТехники
1 use РемонтБытовойТехники
2
3 -- Добавляем новый столбец Delivery_Date в таблицу Orders
4 ALTER TABLE Orders
5 ADD Delivery_Date DATE;
```

81 %

Messages

Commands completed successfully.

Completion time: 2023-11-04T02:56:56.4105198+03:00

14) Добавление ограничения с именем на новый введенный столбец

The screenshot shows the SQL Server Enterprise Manager interface. The SQL command window contains the following code:

```
1 use РемонтБытовойТехники
2
3 -- Устанавливаем ограничение с именем CK_Delivery_Date на столбец Delivery_Date
4 ALTER TABLE Orders
5 ADD CONSTRAINT CK_Delivery_Date CHECK (Delivery_Date >= Order_Date);
```

Below the command window, the Messages pane shows the message: "Commands completed successfully." and the completion time: "2023-11-04T03:01:41.0533774+03:00".

The Object Explorer on the left shows the database structure for "РемонтБытовойТехники". The "dbo.Orders" table is selected, and the "Constraints" folder is expanded, showing the newly added "CK_Delivery_Date" constraint.

15) 5 запросов, используя агрегатные функции:

1. Найти сумму стоимости всех заказов:

The screenshot shows the SQL Server Enterprise Manager interface. The SQL command window contains the following code:

```
1 use РемонтБытовойТехники
2
3 SELECT SUM(Execution_Cost) AS Total
4 FROM Execution;
5
```

Below the command window, the Results pane shows the results of the query:

	Total_Cost
1	1150.00

2. Найти максимальную стоимость ремонта среди всех товаров:

The screenshot shows the SQL Server Enterprise Manager interface. The SQL command window contains the following code:

```
1 use РемонтБытовойТехники
2
3 SELECT MAX(Repair_Cost) AS Max_Repair_Cost
4 FROM ProductsNew;
5
```

Below the command window, the Results pane shows the results of the query:

	Max_Repair_Cost
1	800.00

3. Найти минимальную длительность исполнения заказа:

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT MIN(DATEDIFF(day, Start_Date, End_Date)) AS Min_Execution_Duration
4 FROM Execution;
5
```

81 %

Results Messages

	Min_Execution_Duration
1	1

4. Найти среднюю стоимость ремонта для каждого типа товара:

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT Type, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY Type;
6
```

81 %

Results Messages

	Type	Avg_Repair_Cost
1	Бытовая техника	205.000000
2	Кухонная техника	70.000000
3	Уборочная техника	400.000000
4	Утюги	45.000000
5	Уход за волосами	35.000000
6	Электроника	700.000000

5. Посчитать общее количество заказов для каждого сотрудника:

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT ID_Employee, COUNT(ID_Order) AS Total_Orders
4 FROM Execution
5 GROUP BY ID_Employee;
```

81 %

Results Messages

	ID_Employee	Total_Orders
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1
9	9	1
10	10	1
11	11	1
12	12	1

16) Запросы с оператором Group By:

1. Найти общее количество заказов для каждого сотрудника:

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT ID_Employee, COUNT(ID_Order) AS Total_Orders
4 FROM Execution
5 GROUP BY ID_Employee;
```

81 %

Results Messages

	ID_Employee	Total_Orders
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1
9	9	1
10	10	1
11	11	1
12	12	1

2. Найти среднюю стоимость ремонта для каждого типа товара:

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT Type, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY Type;
```

81 %

Results Messages

	Type	Avg_Repair_Cost
1	Бытовая техника	205.000000
2	Кухонная техника	70.000000
3	Уборочная техника	400.000000
4	Утюги	45.000000
5	Уход за волосами	35.000000
6	Электроника	700.000000

17) Запросы на фильтрацию групп, используя HAVING:

1. Найти среднюю стоимость ремонта для типов товаров, у которых средняя стоимость ремонта превышает 100:

```
3 SELECT Type, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY Type
6 HAVING AVG(Repair_Cost) > 100;
7
```

81 %

Results Messages

	Type	Avg_Repair_Cost
1	Бытовая техника	205.000000
2	Уборочная техника	400.000000
3	Электроника	700.000000

2. Найти среднюю стоимость ремонта для производителей, у которых есть хотя бы один продукт с гарантийным периодом больше 24 месяцев:

```
3 SELECT Manufacturer, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY Manufacturer
6 HAVING MAX(Warranty_Period) > 24;
7
```

81 %

Results Messages

	Manufacturer	Avg_Repair_Cost
1	Samsung	483.333333

18) Написать запросы на расширения SQL Server для группировки (ROLLUP, CUBE, GROUPING SETS, OVER)

1. ROLLUP

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT Type, Manufacturer, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY ROLLUP (Type, Manufacturer);
6
7
```

81 %

Results Messages

	Type	Manufacturer	Avg_Repair_Cost
1	Бытовая техника	Bosch	80.000000
2	Бытовая техника	Brown	30.000000
3	Бытовая техника	Panasonic	120.000000
4	Бытовая техника	Philips	150.000000
5	Бытовая техника	Samsung	425.000000
6	Бытовая техника	NULL	205.000000
7	Кухонная техника	Philips	70.000000
8	Кухонная техника	NULL	70.000000
9	Уборочная техника	Dyson	400.000000
10	Уборочная техника	NULL	400.000000
11	Утюги	Bosch	45.000000
12	Утюги	NULL	45.000000
13	Уход за волосами	Brown	35.000000
14	Уход за волосами	NULL	35.000000
15	Электроника	Panasonic	800.000000
16	Электроника	Samsung	600.000000
17	Электроника	NULL	700.000000
18	NULL	NULL	265.000000

2. CUBE

SQLQuery4.sql - DE...йТехники (sa (53))*

```

1 use РемонтБытовойТехники
2
3 SELECT Type, Manufacturer, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY CUBE (Type, Manufacturer);
6
7

```

81 %

Results Messages

	Type	Manufacturer	Avg_Repair_Cost
1	Бытовая техника	Bosch	80.000000
2	Утюги	Bosch	45.000000
3	NULL	Bosch	62.500000
4	Бытовая техника	Brown	30.000000
5	Уход за волосами	Brown	35.000000
6	NULL	Brown	32.500000
7	Уборочная техника	Dyson	400.000000
8	NULL	Dyson	400.000000
9	Бытовая техника	Panasonic	120.000000
10	Электроника	Panasonic	800.000000
11	NULL	Panasonic	460.000000
12	Бытовая техника	Philips	150.000000
13	Кухонная техника	Philips	70.000000
14	NULL	Philips	110.000000
15	Бытовая техника	Samsung	425.000000
16	Электроника	Samsung	600.000000
17	NULL	Samsung	483.333333
18	NULL	NULL	265.000000
19	Бытовая техника	NULL	205.000000
20	Кухонная техника	NULL	70.000000
21	Уборочная техника	NULL	400.000000
22	Утюги	NULL	45.000000
23	Уход за волосами	NULL	35.000000
24	Электроника	NULL	700.000000

3. GROUPING SETS

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT Type, Manufacturer, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY GROUPING SETS ((Type), (Manufacturer), ());
6
7
```

81 %

Results Messages

	Type	Manufacturer	Avg_Repair_Cost
1	NULL	Bosch	62.500000
2	NULL	Brown	32.500000
3	NULL	Dyson	400.000000
4	NULL	Panasonic	460.000000
5	NULL	Philips	110.000000
6	NULL	Samsung	483.333333
7	NULL	NULL	265.000000
8	Бытовая техника	NULL	205.000000
9	Кухонная техника	NULL	70.000000
10	Уборочная техника	NULL	400.000000
11	Утюги	NULL	45.000000
12	Уход за волосами	NULL	35.000000
13	Электроника	NULL	700.000000

4. OVER

SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT Type, Manufacturer, AVG(Repair_Cost) AS Avg_Repair_Cost
4 FROM ProductsNew
5 GROUP BY GROUPING SETS ((Type), (Manufacturer), ());
6
7
```

81 %

Results

Messages

	Type	Manufacturer	Avg_Repair_Cost
1	NULL	Bosch	62.500000
2	NULL	Brown	32.500000
3	NULL	Dyson	400.000000
4	NULL	Panasonic	460.000000
5	NULL	Philips	110.000000
6	NULL	Samsung	483.333333
7	NULL	NULL	265.000000
8	Бытовая техника	NULL	205.000000
9	Кухонная техника	NULL	70.000000
10	Уборочная техника	NULL	400.000000
11	Утюги	NULL	45.000000
12	Уход за волосами	NULL	35.000000
13	Электроника	NULL	700.000000

19) Написать запрос на разворачивание данных (PIVOT), отмена разворачивания (UNPIVOT).

1) PIVOT

DESKTOP-7L6U0EU\...dbo.ProductsNew SQLQuery4.sql - DE...йТехники (sa (53))*

```
1 use РемонтБытовойТехники
2
3 SELECT *
4 FROM (
5     SELECT Manufacturer, Type, Repair_Cost
6     FROM ProductsNew
7 ) AS SourceTable
8 PIVOT (
9     AVG(Repair_Cost)
10    FOR Type IN ([Бытовая техника], [Электроника], [Утюги])
11 ) AS PivotTable;
```

81 %

Results Messages

	Manufacturer	Бытовая техника	Электроника	Утюги
1	Bosch	80.000000	NULL	45.000000
2	Brown	30.000000	NULL	NULL
3	Dyson	NULL	NULL	NULL
4	Panasonic	120.000000	800.000000	NULL
5	Philips	150.000000	NULL	NULL
6	Samsung	425.000000	600.000000	NULL

2) UNPIVOT

```

1 use РемонтБытовойТехники
2
3 SELECT ID_Product, Worker, KindOfDev
4 FROM (
5     SELECT ID_Product, Manufacturer, Type
6     FROM ProductsNew
7 ) AS SourceTable
8 UNPIVOT (
9     KindOfDev FOR Worker IN (Manufacturer, Type)
10 ) AS UnpivotedTable;
11

```

81 %

Results Messages

	ID_Product	Worker	KindOfDev
1	1	Manufacturer	Philips
2	1	Type	Бытовая техника
3	2	Manufacturer	Bosch
4	2	Type	Бытовая техника
5	3	Manufacturer	Brown
6	3	Type	Бытовая техника
7	4	Manufacturer	Samsung
8	4	Type	Бытовая техника
9	5	Manufacturer	Panasonic
10	5	Type	Бытовая техника
11	6	Manufacturer	Philips
12	6	Type	Кухонная техника
13	7	Manufacturer	Bosch
14	7	Type	Утюги
15	8	Manufacturer	Brown
16	8	Type	Уход за волосами
17	9	Manufacturer	Samsung
18	9	Type	Электроника
19	10	Manufacturer	Panasonic
20	10	Type	Электроника
21	11	Manufacturer	Samsung
22	11	Type	Бытовая техника
23	12	Manufacturer	Dyson
24	12	Type	Уборочная техника