

Задача 1 Создание представлений

Отчеты по предметной области, которые было предложено сделать в задании 1 выполнить в форме представлений. Всего должно быть не менее 3 представлений.

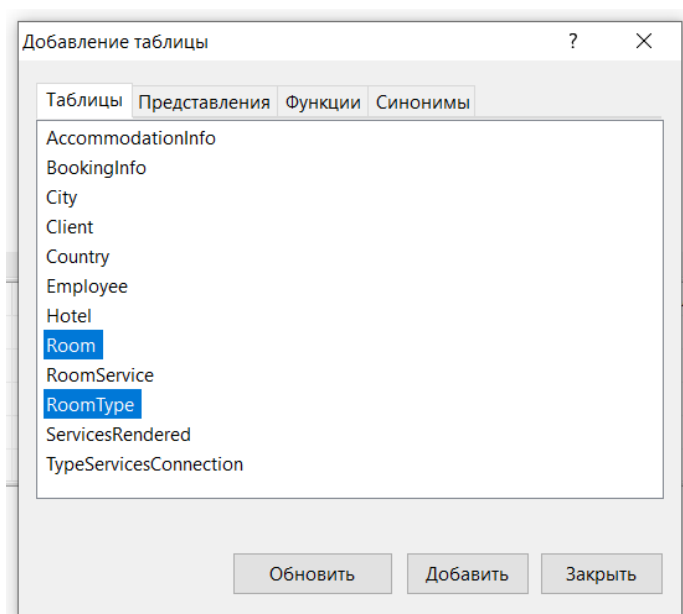
Представление 1. Список номеров

Список номеров всех отелей с указанием их типов.

Порядок выполнения:

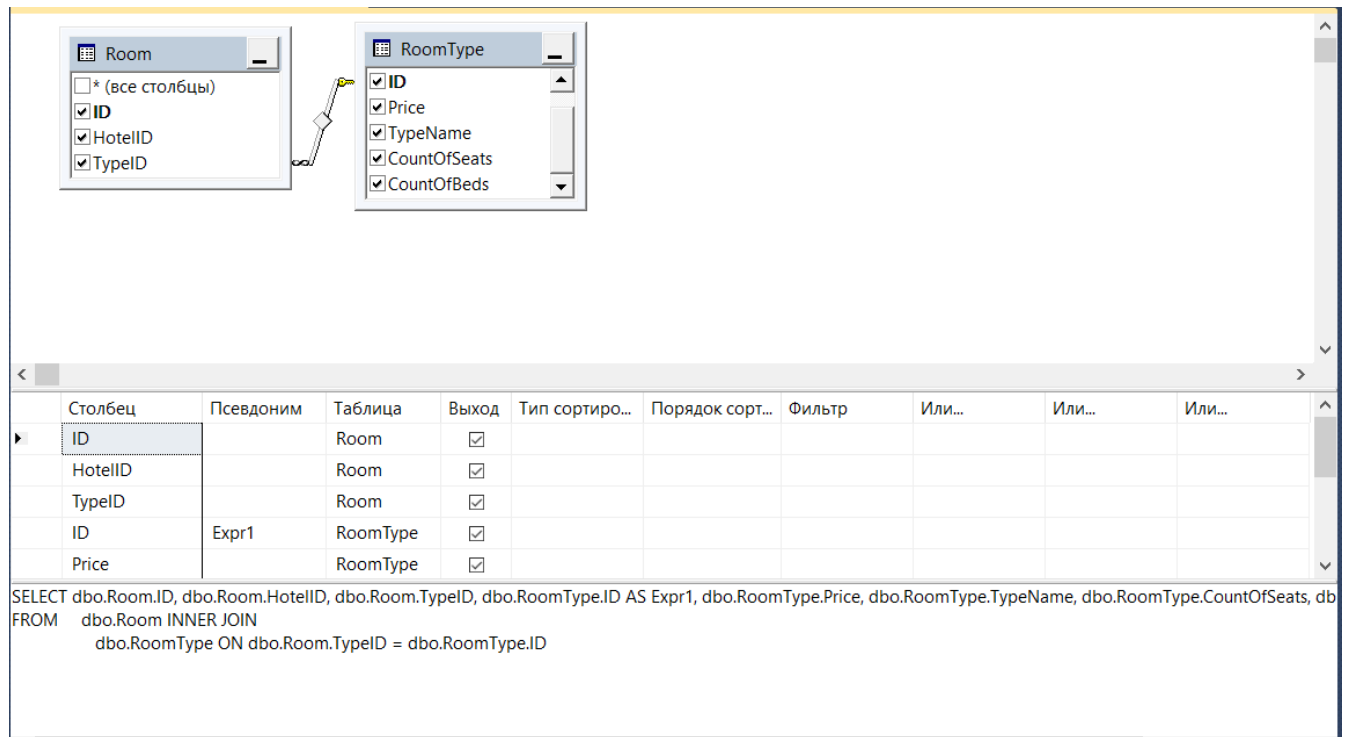
1. Чтобы создать представление выполните следующие действия:

- Раскройте узел базы данных , затем узел Представления (Views); отобразится список текущих представлений БД.
- Чтобы создать новое представление, в контекстном меню узла Представления (Views) выберите команду Создать представление (New View). Отобразится диалоговое окно Добавить таблицу (Add Table).
- Выберите таблицы, которые следует добавить в представление, и щелкните кнопку Добавить (Add). Выбранный объект будет отображен в области диаграммы окна конструктора запросов в виде плавающей панели со списком столбцов объекта.



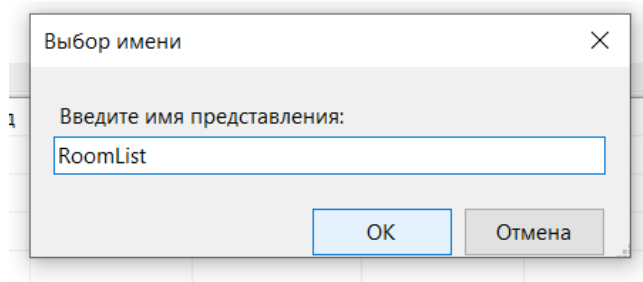
- По окончании работы с диалоговым окном Добавить таблицу (Add Table) щелкните кнопку Заккрыть (Close).

- Предоставленные списки столбцов используйте для выбора тех столбцов, которые следует использовать в представлении. Таким образом, создается инструкция SELECT, определяющая представление.



- В поле Filter можете ввести информацию ограничивающую представление (например, месяц или уровень зарплаты и т.д.):

3. Сохраните представление под именем.



4. Создайте запрос SELECT для проверки того, что представление возвращает только требуемые данные.

```
SELECT * FROM RoomList
```

Результаты								
Сообщения								
План выполнения								
	ID	HotelID	Type...	Expr1	Price	TypeName	CountOfSe...	CountOfBe...
1	1	1	1	1	3500	Стандарт 2/1	2	1
2	2	1	1	1	3500	Стандарт 2/1	2	1
3	3	1	2	2	3500	Стандарт 2/2	2	2
4	4	1	2	2	3500	Стандарт 2/2	2	2
5	5	1	4	4	4000	Комфорт	2	1
6	6	1	4	4	4000	Комфорт	2	1
7	7	1	5	5	2000	Бюджет 1	1	1
8	8	1	6	6	2500	Бюджет 2	2	2
9	9	1	6	6	2500	Бюджет 2	2	2
10	10	1	7	7	6000	Люкс	2	1
11	11	2	1	1	3500	Стандарт 2/1	2	1
12	12	2	2	2	3500	Стандарт 2/2	2	2
13	13	2	4	4	4000	Комфорт	2	1
14	14	2	4	4	4000	Комфорт	2	1
15	15	2	5	5	2000	Бюджет 1	1	1

Запрос успешно выполнен. | LAPTOP-DJ70UDCI

5. Посмотрите код созданного представления с помощью процедуры sp_helptext

EXEC sp_helptext RoomList

Результаты	
Сообщения	
План выполнения	
	Text
1	CREATE VIEW RoomList
2	AS
3	SELECT dbo.Room.ID, dbo.Room.HotelID, dbo.Room.TypeID, dbo.RoomType.ID AS Expr1, dbo.RoomType.Price, dbo.RoomType.TypeName, dbo.RoomTyp...
4	FROM dbo.Room INNER JOIN
5	dbo.RoomType ON dbo.Room.TypeID = dbo.RoomType.ID

6. Защита кода – шифрование представлений. Примените оператор WITH ENCRYPTION к Вашему представлению. После выполнения этого оператора предпримем попытку ознакомиться с определением представления с помощью процедуры sp_helptext. Какой результат вы получили?

Код:

ALTER VIEW RoomList

WITH ENCRYPTION

AS

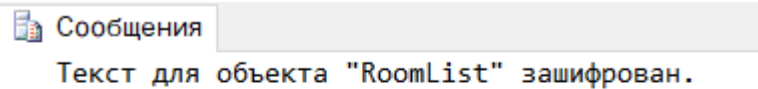
SELECT dbo.Room.ID, dbo.Room.HotelID, dbo.Room.TypeID, dbo.RoomType.ID AS Expr1,
dbo.RoomType.Price, dbo.RoomType.TypeName, dbo.RoomType.CountOfSeats,
dbo.RoomType.CountOfBeds

FROM dbo.Room INNER JOIN

 dbo.RoomType ON dbo.Room.TypeID = dbo.RoomType.ID

EXEC sp_helptext RoomList

Результат:



Представление 2. Прейскурант услуг

В этом представлении объединяются таблицы RoomService, TypeServicesConnection и RoomType. То есть список услуг будет отображаться с типами номеров, для которых они доступны, и названиями этих типов.

```
CREATE VIEW dbo.ServiceList
AS
SELECT dbo.RoomService.ID, dbo.RoomService.ServiceName,
dbo.RoomService.ServicePrice, dbo.RoomType.TypeName,
dbo.TypeServicesConnection.TypeID
FROM   dbo.RoomService INNER JOIN
        dbo.TypeServicesConnection ON dbo.RoomService.ID =
dbo.TypeServicesConnection.ServiceID INNER JOIN
        dbo.RoomType ON dbo.TypeServicesConnection.TypeID = dbo.RoomType.ID
```

	ID	ServiceName	ServicePri...	TypeName	TypeID
1	1	Бесплатный WI-FI	0	Стандарт 2/1	1
2	1	Бесплатный WI-FI	0	Стандарт 2/2	2
3	1	Бесплатный WI-FI	0	Стандарт 1/1	3
4	1	Бесплатный WI-FI	0	Комфорт	4
5	1	Бесплатный WI-FI	0	Бюджет 1	5
6	1	Бесплатный WI-FI	0	Бюджет 2	6
7	1	Бесплатный WI-FI	0	Люкс	7
8	2	Бесплатный завтрак	0	Стандарт 2/1	1
9	2	Бесплатный завтрак	0	Стандарт 2/2	2
10	2	Бесплатный завтрак	0	Стандарт 1/1	3
11	2	Бесплатный завтрак	0	Комфорт	4
12	3	Платный завтрак	300	Бюджет 1	5
13	3	Платный завтрак	300	Бюджет 2	6
14	2	Бесплатный завтрак	0	Люкс	7
15	1	Бесплатный WI-FI	0	Стандарт 2/1	1

Здесь мы выполнили запрос:

```
SELECT * FROM ServiceList
```

Если выполнить следующий запрос, то получим прейскурант услуг для конкретного типа номера(здесь TypeID = 5):

```
SELECT * FROM ServiceList
WHERE TypeID = 5
```

	ID	ServiceName	ServicePri...	TypeNa...	TypeID
1	1	Бесплатный WI-FI	0	Бюджет 1	5
2	3	Платный завтрак	300	Бюджет 1	5
3	1	Бесплатный WI-FI	0	Бюджет 1	5
4	4	Бесплатное хранение багажа	0	Бюджет 1	5
5	5	Сейф	500	Бюджет 1	5
6	6	Парковка	50	Бюджет 1	5
7	7	Сауна 2 часа	1000	Бюджет 1	5
8	8	Бассейн 2 часа	200	Бюджет 1	5
9	9	Прачечная	200	Бюджет 1	5

Представление 3. Список нереализованных броней

```
CREATE VIEW BookingNotRealizedList
AS
SELECT ID, ClientID, RoomTypeID, ArrivalDate, DepartureDate, BookingDate, HotelID,
RealizationInfo
FROM    dbo.BookingInfo
WHERE   (RealizationInfo = 'Не реализовано')

SELECT * FROM BookingNotRealizedList
```

	ID	ClientID	RoomTyp...	ArrivalDate	DepartureDate	BookingDate	HotelID	RealizationInfo
1	5	5	2	2022-03-15 12:00:00.000	2022-03-27 12:00:00.000	2022-02-28 12:56:09.560	1	Не реализовано
2	7	7	3	2022-04-25 12:00:00.000	2022-04-30 12:00:00.000	2022-04-10 12:56:09.560	1	Не реализовано
3	8	8	6	2022-04-27 12:00:00.000	2022-05-05 12:00:00.000	2022-04-15 12:56:09.560	3	Не реализовано

Задача 2: Создание обновляемого представления

Представление 3 создадим с конструкцией WITH CHECK OPTION.

```
CREATE VIEW BookingNotRealizedList
AS
SELECT ID, ClientID, RoomTypeID, ArrivalDate, DepartureDate, BookingDate, HotelID,
RealizationInfo
```

```
FROM    dbo.BookingInfo
WHERE (RealizationInfo = 'Не реализовано')
WITH CHECK OPTION
```

обновление или вставка данных с помощью представления, в котором задана конструкция WITH CHECK OPTION, осуществляется, только если строка, полученная в результате применения этой операции обновления или вставки, соответствует критериям, допускающим ее включение в это представление.

После создания представления выполним вставку в него и посмотрим результаты.

Представление до вставки:

100 %									
		Результаты		Сообщения					
	ID	ClientID	RoomTyp...	ArrivalDate	DepartureDate	BookingDate	HotelID	RealizationInfo	
1	5	5	2	2022-03-15 12:00:00.000	2022-03-27 12:00:00.000	2022-02-28 12:56:09.560	1	Не реализовано	
2	7	7	3	2022-04-25 12:00:00.000	2022-04-30 12:00:00.000	2022-04-10 12:56:09.560	1	Не реализовано	
3	8	8	6	2022-04-27 12:00:00.000	2022-05-05 12:00:00.000	2022-04-15 12:56:09.560	3	Не реализовано	

Попытка вставить неподходящую строку:

```
INSERT INTO BookingNotRealizedList
VALUES (1, 3, CAST('2022-04-25T12:00:00.000' AS datetime),
CAST('2022-04-28T12:00:00.000' AS datetime),
CAST('2022-04-20T12:56:09.560' AS datetime), 1, 'Реализовано');
```

Сообщения

Сообщение 550, уровень 16, состояние 1, строка 32

Ошибка при попытке вставки или обновления, поскольку целевое представление либо указывает WITH CHECK OPTION, либо охватывает представлени

Выполнение данной инструкции было прервано.

Вставка подходящей строки:

```
INSERT INTO BookingNotRealizedList
VALUES (1, 3, CAST('2022-04-25T12:00:00.000' AS datetime),
CAST('2022-04-28T12:00:00.000' AS datetime),
CAST('2022-04-20T12:56:09.560' AS datetime), 1, 'Не реализовано');
```

Результаты		Сообщения		План выполнения					
	ID	ClientID	RoomTyp...	ArrivalDate	DepartureDate	BookingDate	HotelID	RealizationInfo	
1	5	5	2	2022-03-15 12:00:00.000	2022-03-27 12:00:00.000	2022-02-28 12:56:09.560	1	Не реализовано	
2	7	7	3	2022-04-25 12:00:00.000	2022-04-30 12:00:00.000	2022-04-10 12:56:09.560	1	Не реализовано	
3	8	8	6	2022-04-27 12:00:00.000	2022-05-05 12:00:00.000	2022-04-15 12:56:09.560	3	Не реализовано	
4	11	1	3	2022-04-25 12:00:00.000	2022-04-28 12:00:00.000	2022-04-20 12:56:09.560	1	Не реализовано	

Значение также добавилось в таблицу BookingInfo:

	ID	ClientID	RoomTyp...	ArrivalDate	DepartureDate	BookingDate	HotelID	RealizationInfo
1	1	1	3	2022-02-25 12:00:00.000	2022-02-28 12:00:00.000	2022-02-20 12:56:09.560	1	Реализовано
2	2	2	7	2022-02-27 12:00:00.000	2022-03-03 12:00:00.000	2022-02-23 12:56:09.560	4	Реализовано
3	3	3	4	2022-03-02 12:00:00.000	2022-03-14 12:00:00.000	2022-02-28 12:56:09.560	1	Реализовано
4	4	4	1	2022-03-05 12:00:00.000	2022-03-12 12:00:00.000	2022-03-01 12:56:09.560	2	Реализовано
5	5	5	2	2022-03-15 12:00:00.000	2022-03-27 12:00:00.000	2022-02-28 12:56:09.560	1	Не реализовано
6	6	6	4	2022-04-05 12:00:00.000	2022-04-13 12:00:00.000	2022-03-20 12:56:09.560	1	Реализовано
7	7	7	3	2022-04-25 12:00:00.000	2022-04-30 12:00:00.000	2022-04-10 12:56:09.560	1	Не реализовано
8	8	8	6	2022-04-27 12:00:00.000	2022-05-05 12:00:00.000	2022-04-15 12:56:09.560	3	Не реализовано
9	11	1	3	2022-04-25 12:00:00.000	2022-04-28 12:00:00.000	2022-04-20 12:56:09.560	1	Не реализовано

Задача 3: Создание индексированного представления

- Установить параметры для поддержки индексированных представлений

```
SET NUMERIC_ROUNDABORT OFF;
```

```
SET ANSI_PADDING, ANSI_WARNINGS, CONCAT_NULL_YIELDS_NULL,
```

```
ARITHABORT, QUOTED_IDENTIFIER, ANSI_NULLS ON;
```

```
GO
```

- Создать представления с привязкой к схеме (используя опцию WITH SCHEMABINDING)

Представление, в котором отображается список броней, в которых длительность проживания меньше 10 дней.

```
CREATE VIEW BookingListLengthLess10
```

```
WITH SCHEMABINDING
```

```
AS
```

```
SELECT ID, ClientID, RoomTypeID, ArrivalDate, DepartureDate, BookingDate, HotelID,
RealizationInfo
```

```
FROM    dbo.BookingInfo
```

```
WHERE DATEDIFF(d, ArrivalDate, DepartureDate) < 10
```

```
WITH CHECK OPTION
```

- Создать кластерного индекса для представления

```
CREATE UNIQUE CLUSTERED INDEX CL_BookingListID
```

```
ON BookingListLengthLess10(ID, ArrivalDate)
```

Создадим такой же для таблицы:

```
CREATE UNIQUE CLUSTERED INDEX CL_BookingListID
```

```
ON BookingInfo(ID, ArrivalDate)
```

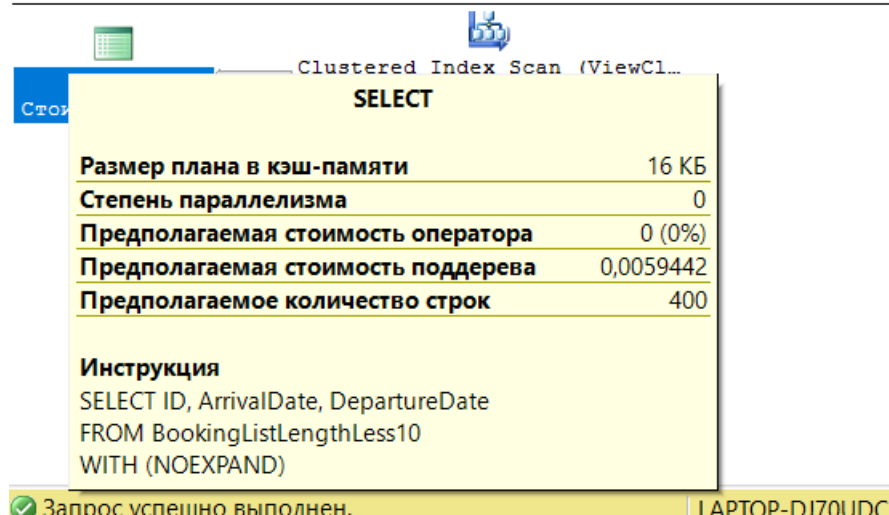
- Сравнить производительность индексируемого представления с производительностью запроса SELECT, на котором оно основано, для этого можно использовать табличную подсказку `NOEXPAND`

```
SELECT ID, ArrivalDate, DepartureDate
```

```
FROM BookingListLengthLess10
```

```
WITH (NOEXPAND)
```

Запрос 1: стоимость запроса (по отношению к пакету):
`SELECT ID, ArrivalDate, DepartureDate FROM BookingListLengthLess10`



SELECT	
Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0059442
Предполагаемое количество строк	400
Инструкция	
SELECT ID, ArrivalDate, DepartureDate FROM BookingListLengthLess10 WITH (NOEXPAND)	

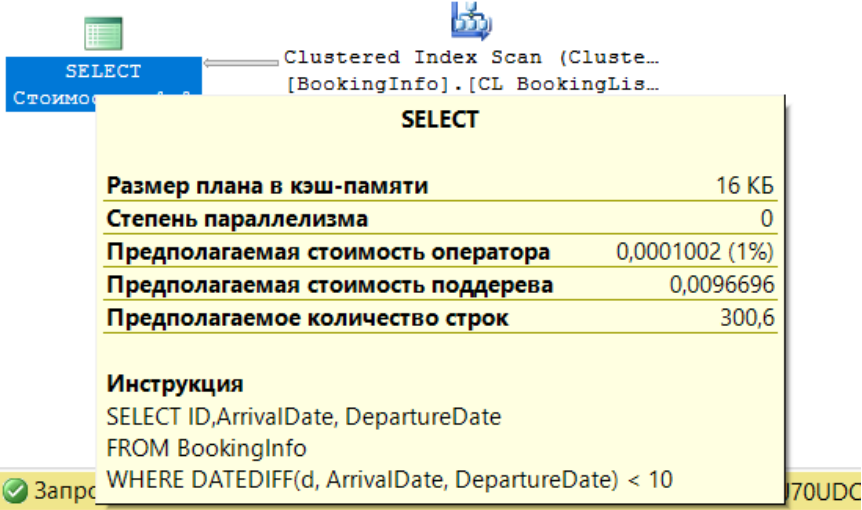
Запрос успешно выполнен.

```
SELECT ID,ArrivalDate, DepartureDate
```

```
FROM BookingInfo
```

```
WHERE DATEDIFF(d, ArrivalDate, DepartureDate) < 10
```


Запрос 1: стоимость запроса (по отношению к пакету):
SELECT ID,ArrivalDate, DepartureDate FROM BookingInfo



SELECT
Стоимость

Clustered Index Scan (Cluste...
[BookingInfo].[CL BookingLis...

SELECT

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0,0001002 (1%)
Предполагаемая стоимость поддерева	0,0096696
Предполагаемое количество строк	300,6

Инструкция
SELECT ID,ArrivalDate, DepartureDate
FROM BookingInfo
WHERE DATEDIFF(d, ArrivalDate, DepartureDate) < 10

Запрос 1: стоимость запроса (по отношению к пакету): 70UDC

Представление оказалось быстрее почти в 2 раза.