# Триггер 1. INSERT

При добавлении записи в таблицу информации о проживании, проверить на существование указанный ID бронирования, и если он найден, то изменить поле с информацией о реализации брони на 'Реализовано'. Если ID бронирования не найден и он не равен NULL, вывести ошибку.

### Код.

CREATE TRIGGER CheckBookingInfo

ON AccommodationInfo

**INSTEAD OF INSERT** 

AS

**BEGIN** 

IF (EXISTS (SELECT ID FROM BookingInfo WHERE ID = (SELECT BookingID FROM inserted)))

**BEGIN** 

INSERT INTO AccommodationInfo

VALUES ((SELECT ClientID FROM inserted), (SELECT RoomID FROM inserted), (SELECT BookingID FROM inserted),

(SELECT ArrivalDate from inserted), (SELECT DepartureDate from inserted), (SELECT Cost from inserted))

**UPDATE** BookingInfo

SET RealizationInfo = 'Реализовано'

WHERE ID = (SELECT BookingID FROM inserted)

END

ELSE IF NOT ((SELECT BookingID FROM inserted) = NULL) RAISERROR('Неверно введен BookingID', 16, 1)

ELSE INSERT INTO AccommodationInfo

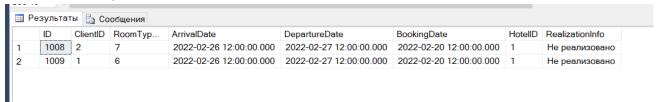
VALUES ((SELECT ClientID FROM inserted), (SELECT RoomID FROM inserted), (SELECT BookingID FROM inserted),

(SELECT ArrivalDate from inserted), (SELECT DepartureDate from inserted), (SELECT Cost from inserted))

**END** 

### Пример 1.

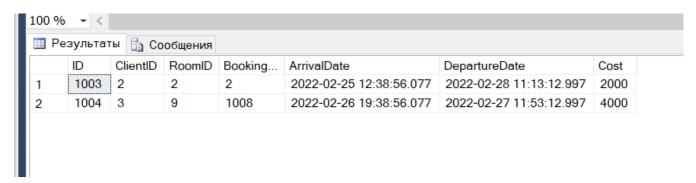
Таблица BookingInfo(ID > 1000) до добавления записи в AccommodationInfo



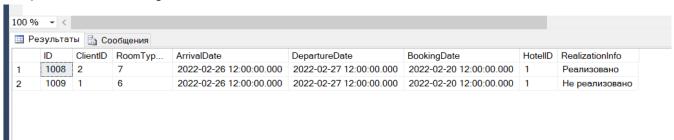
#### Запрос INSERT:

INSERT INTO AccommodationInfo VALUES (3, 9, 1008, CAST('2022-02-26T19:38:56.078' AS datetime), CONVERT(datetime, '2022-02-27T11:53:12.997', 126), 4000)

Добавили в AccommodationInfo запись:



Теперь таблица BookingInfo выглядит так:



### Пример 2.

Введем несуществующий BookingID

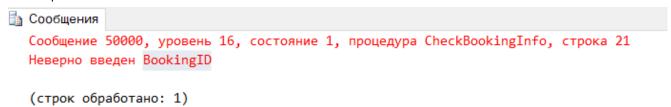
Запрос:

INSERT INTO AccommodationInfo

VALUES (1001, 1, 1010, CAST('2022-02-26T19:38:56.078' AS datetime),

CONVERT(datetime, '2022-02-27T11:53:12.997', 126), 1500)

Сообщение об ошибке:



# Триггер 2. INSERT

При добавлении клиента паспортные данные проверяются на корректность ввода. Пример правильного ввода: 6329 669704 (то есть 4 цифры - пробел - 6 цифр). Если ввод неправильный, отменяем его.

### Код.

CREATE TRIGGER CheckPassport

**ON Client** 

FOR INSERT

AS

**BEGIN** 

DECLARE @Passport varchar(255) = (SELECT Passport FROM inserted)

IF NOT(@Passport LIKE '[0-9][0-9][0-9][0-9] [0-9][0-9][0-9][0-9][0-9])

**BEGIN** 

**ROLLBACK TRAN** 

RAISERROR('Паспортные данные введены некорректно', 16, 1)

**END** 

**END** 

## Пример 1.

Добавим клиента с корректными данными:

### Таблица до добавления:

Результ	таты 🛅 С	ообщения		
	ID	FirstName	LastName	Passport
986	351	Аввакум	Шувалов	0022 154299
987	370	Сталина	Шувалов	4561 492898
988	425	Глория	Шувалов	5386 589138
989	557	Тала	Шувалов	7779 720327
990	609	Леонтий	Шувалов	6528 542705
991	989	Фредер	Шувалов	5119 894244
992	86	Феофан	Шульгин	4636 686486
993	897	Матео	Шульгин	1928 881026
994	75	Ярослава	Щеглов	1809 137219
995	252	Акилина	Щербаков	5236 260665
996	339	Таисия	Щукин	3243 925705
997	589	Раймонд	Юдин	9512 689824
998	781	Евгений	Юдин	1867 925000
999	957	Малика	Юдин	5716 337128
1000	1009	Валерия	Зайцева	6879 945858

Запрос:

**INSERT INTO CLient** 

VALUES('Мария', 'Зайцева', '6860 445008')

Таблица после добавления:

<u> </u>	езультаты	🔓 Сообщени	19	
	ID	FirstName	LastName	Passport
987	370	Сталина	Шувалов	4561 492898
988	425	Глория	Шувалов	5386 589138
989	557	Тала	Шувалов	7779 720327
990	609	Леонтий	Шувалов	6528 542705
991	989	Фредер	Шувалов	5119 894244
992	86	Феофан	Шульгин	4636 686486
993	897	Матео	Шульгин	1928 881026
994	75	Ярослава	Щеглов	1809 137219
995	252	Акилина	Щербаков	5236 260665
996	339	Таисия	Щукин	3243 925705
997	589	Раймонд	Юдин	9512 689824
998	781	Евгений	Юдин	1867 925000
999	957	Малика	Юдин	5716 337128
1	1009	Валерия	Зайцева	6879 945858
1	1011	Мария	Зайцева	6860 445008

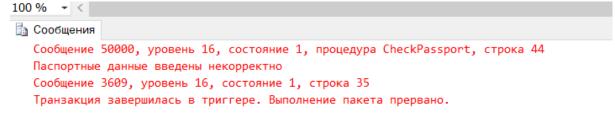
## Пример 2.

Уберем из паспортных данных запроса пробел и последнюю цифру:

**INSERT INTO CLient** 

VALUES('Мария', 'Зайцева', '686044500')

Получим сообщение об ошибке:



# Триггер 3. DELETE

При удалении клиента удаляются все его данные о проживании и бронировании. Если клиента не существует, выводится ошибка.

### Код.

CREATE TRIGGER DeleteClient
ON Client
INSTEAD OF DELETE
AS
BEGIN
DECLARE @ClientID INT = (SELECT ID FROM deleted)

IF (EXISTS (SELECT ID FROM Client WHERE ID = @ClientID))
BEGIN
DELETE FROM AccommodationInfo WHERE ClientID = @ClientID
DELETE FROM BookingInfo WHERE ClientID = @ClientID
DELETE FROM Client WHERE ID = @ClientID
END
ELSE RAISERROR('Клиента с таким ID не существует', 16, 1)
END

## Пример 1.

### Удалим клиента с ID = 1011:

### Таблица Client до удаления:

	-			
	ID	FirstName	LastName	Passport
987	370	Сталина	Шувалов	4561 492898
988	425	Глория	Шувалов	5386 589138
989	557	Тала	Шувалов	7779 720327
990	609	Леонтий	Шувалов	6528 542705
991	989	Фредер	Шувалов	5119 894244
992	86	Феофан	Шульгин	4636 686486
993	897	Матео	Шульгин	1928 881026
994	75	Ярослава	Щеглов	1809 137219
995	252	Акилина	Щербаков	5236 260665
996	339	Таисия	Щукин	3243 925705
997	589	Раймонд	Юдин	9512 689824
998	781	Евгений	Юдин	1867 925000
999	957	Малика	Юдин	5716 337128
10	1009	Валерия	Зайцева	6879 945858
10	1011	Мария	Зайцева	6860 445008

#### Таблица BookingInfo до удаления:

		■ Результаты										
ID	)	ClientlD	RoomTyp	ArrivalDate	DepartureDate	BookingDate	HotellD	RealizationInfo				
1 10	011	1011	6	2022-02-26 12:00:00.000	2022-02-27 12:00:00.000	2022-02-20 12:00:00.000	1	Реализовано				
2 10	800	2	7	2022-02-25 12:00:00.000	2022-02-27 12:00:00.000	2022-02-20 12:00:00.000	1	Реализовано				
3 10	009	1	6	2022-02-26 12:00:00.000	2022-02-27 12:00:00.000	2022-02-20 12:00:00.000	1	Не реализовано				

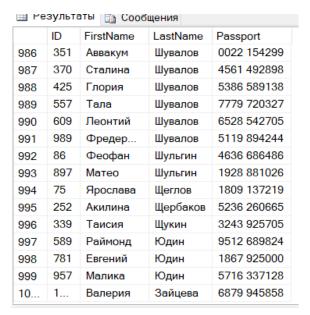
#### Таблица AccommodationInfo до удаления:

<u>■</u> P	🖽 Результаты 🛅 Сообщения											
	ID	ClientID	RoomID	Booking	ArrivalDate	DepartureDate	Cost					
1	1003	2	2	2	2022-02-25 12:38:56.077	2022-02-28 11:13:12.997	2000					
2	1008	1011	1	1011	2022-02-26 19:38:56.077	2022-02-27 11:53:12.997	1500					

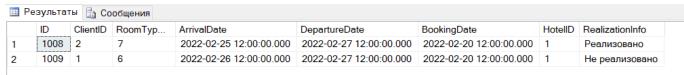
Запрос удаления: DELETE FROM Client

WHERE ID = 1011

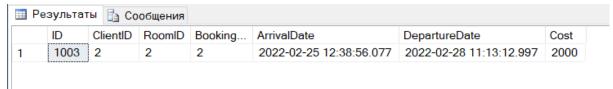
Таблица Client после удаления:



#### Таблица BookingInfo после удаления:



#### Таблица AccommodationInfo после удаления:



### Пример 2.

Попробуем ещё раз удалить клиента с тем же ID:

**DELETE FROM Client** 

WHERE ID = 1011

Получим сообщение об ошибке:

```
Сообщения

Сообщение 50000, уровень 16, состояние 1, процедура DeleteClient, строка 43

Клиента с таким ID не существует

(строк обработано: 0)
```

# Триггер 4. DELETE

При попытке удаления города, в котором есть отели, выводится ошибка.

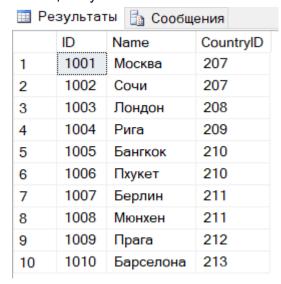
### Код.

CREATE TRIGGER DeleteCity
ON City
INSTEAD OF DELETE
AS
BEGIN

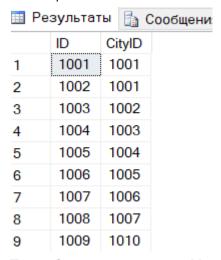
IF (EXISTS (SELECT \* FROM Hotel WHERE CityID = (SELECT ID FROM deleted))) RAISERROR('Нельзя удалить этот город, потому что в нем есть отели.', 16, 1) END

## Пример 1.

#### Таблица City



#### Таблица Hotel



Попробуем удалить город Москва с ID = 1001:

DELETE City WHERE Name = 'Москва'

Получим сообщение об ошибке:

```
100 % ▼ < Сообщения Сообщение 50000, уровень 16, состояние 1, процедура DeleteCity, строка 47 Нельзя удалить этот город, потому что в нем есть отели.

(строк обработано: 1)
```

# Триггер 5. UPDATE

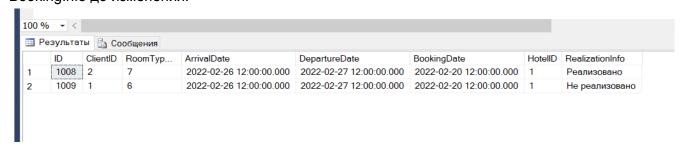
При изменении даты прибытия в информации о бронировании осуществляем проверку о том, что дата прибытия раньше (меньше) даты отбытия. Если нет, то предупреждать.

### Код.

CREATE TRIGGER UpdateDate
ON BookingInfo
INSTEAD OF UPDATE
AS
BEGIN
IF ((SELECT ArrivalDate from inserted) < (SELECT DepartureDate FROM inserted))
BEGIN
UPDATE BookingInfo
SET ArrivalDate = (SELECT ArrivalDate from inserted)
WHERE ID = (SELECT ID from inserted)
END
ELSE RAISERROR ('Дата прибытия не может быть позже даты отбытия', 16, 1)
END

### Пример 1.

BookingInfo до изменения:



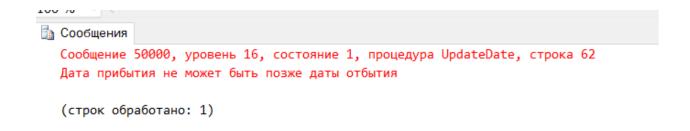
Запрос(дата прибытия теперь совпадает с датой отбытия):

**UPDATE** BookingInfo

SET ArrivalDate = CAST('2022-02-27T12:00:00.000' AS datetime)

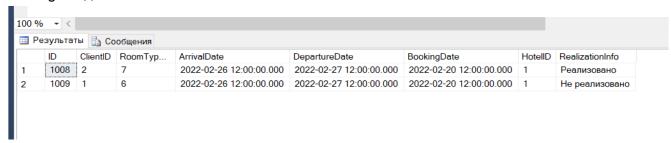
WHERE ID = 1008

Ошибка:



### Пример 2.

#### BookingInfo до изменения:



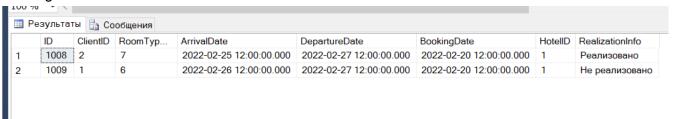
Запрос(корректная дата):

**UPDATE** BookingInfo

SET ArrivalDate = CAST('2022-02-25T12:00:00.000' AS datetime)

WHERE ID = 1008

BookingInfo после изменения:



# Триггер 6. UPDATE

При изменения RoomID(номера)(то есть когда клиента хочет переселиться в другой номер) в таблице информации о проживании проверять, не живет ли кто-нибудь сейчас в этом номере, то есть если максимальная дата отбытия из этого номера позднее даты прибытия клиента, желающего переселиться(то есть клиенты находятся в отеле в одно время), то предупреждать.

Также если цена номера, в который переселяются, не совпадает с ценой предыдущего номера, то изменить стоимость проживания.

(Не могут одновременно выполняться два условия: максимальная дата отбытия клиента из номера больше даты прибытия клиента, желающего переселиться, и клиенты не находятся в отеле в одно время, потому что тогда получается, что дата прибытия клиента из нужного номера позже даты отбытия текущего клиента и мы проверяем человека из будущего, что невозможно.)

### Код.

CREATE TRIGGER UpdateRoomID

ON AccommodationInfo

**INSTEAD OF UPDATE** 

AS

**BEGIN** 

DECLARE @ID INT, @RoomID INT, @ArrivalDate datetime, @DepartureDate datetime SELECT @ID = ID, @RoomID = RoomID, @ArrivalDate = ArrivalDate, @DepartureDate =

DepartureDate FROM inserted

IF (EXISTS (SELECT \* FROM AccommodationInfo WHERE RoomID = @RoomID))

IF((SELECT DepartureDate FROM AccommodationInfo WHERE RoomID = @RoomID) >= @ArrivalDate)

RAISERROR('Требуемый номер в данный момент занят', 16, 1)

**UPDATE** AccommodationInfo

SET RoomID = @RoomID

WHERE ID = @ID

 ${\sf IF\;NOT\;((SELECT\;Price\;FROM\;RoomType\;WHERE\;ID=(SELECT\;TypeID\;FROM\;RoomType;Colored and Colored and$ 

WHERE ID = @RoomID))

= (SELECT Cost FROM AccommodationInfo WHERE ID = @ID))

**UPDATE** AccommodationInfo

SET Cost = @RoomID

WHERE ID = @ID

**END** 

### Пример 1.

### AccommodationInfo до изменения:

Результаты      Сообщения										
	ID	ClientID	RoomID	Booking	ArrivalDate	DepartureDate	Cost			
1	1003	2	2	2	2022-02-25 12:38:56.077	2022-02-28 11:13:12.997	2000			
2	1009	1013	1	1012	2022-02-26 19:38:56.077	2022-02-27 11:53:12.997	1500			
3	1010	1009	4	NULL	2022-02-22 19:38:56.077	2022-02-24 11:53:12.997	2500			

#### Запрос:

UPDATE AccommodationInfo

SET RoomID = 1

WHERE ID = 1011

#### AccommodationInfo после изменения:

■ Pe	езультат	гы 🛅 Со	общения				
	ID	ClientID	RoomID	Booking	ArrivalDate	DepartureDate	Cost
1	1003	2	2	2	2022-02-25 12:38:56.077	2022-02-28 11:13:12.997	2000
2	1009	1013	1	1012	2022-02-26 19:38:56.077	2022-02-27 11:53:12.997	1500
3	1011	1009	1	NULL	2022-02-28 19:38:56.077	2022-03-02 11:53:12.997	1500

## Пример 2.

### AccommodationInfo до изменения:

□ Pe	езультат	гы 🛅 Со	общения				
	ID	ClientID	RoomID	Booking	ArrivalDate	DepartureDate	Cost
1	1003	2	2	2	2022-02-25 12:38:56.077	2022-02-28 11:13:12.997	2000
2	1009	1013	1	1012	2022-02-26 19:38:56.077	2022-02-27 11:53:12.997	1500
3	1011	1009	1	NULL	2022-02-28 19:38:56.077	2022-03-02 11:53:12.997	1500
4	1012	2	4	NULL	2022-02-25 19:38:56.077	2022-03-02 11:53:12.997	2500

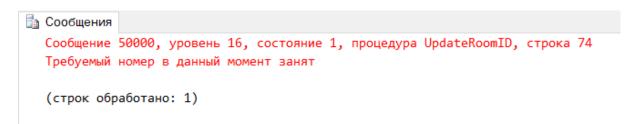
#### Запрос:

UPDATE AccommodationInfo

SET RoomID = 2

WHERE ID = 1012

Получим сообщение об ошибке, потому что дата отбытия у клиента с RoomID = 2 - это 28.02.2022, а дата прибытия у клиента, который хочет в этот номер переселиться, - 25 февраля.



#### Таблица не изменилась.

⊞ P€										
	ID	ClientID	RoomID	Booking	ArrivalDate	DepartureDate	Cost			
1	1003	2	2	2	2022-02-25 12:38:56.077	2022-02-28 11:13:12.997	2000			
2	1009	1013	1	1012	2022-02-26 19:38:56.077	2022-02-27 11:53:12.997	1500			
3	1011	1009	1	NULL	2022-02-28 19:38:56.077	2022-03-02 11:53:12.997	1500			
4	1012	2	4	NULL	2022-02-25 19:38:56.077	2022-03-02 11:53:12.997	2500			