


## Лабораторная работа № 6


### Задания:

1. Определите потенциальные ключи для каждого из отношений. Если для каких-то отношений таких ключей окажется несколько, выберите один из них на роль первичного (РК), а остальные пометьте альтернативными (АК).

**Решение:** Таблицы с начальными данными были разбавлены после прочтения описания к заданию

 Table Name:  Schema: **mydb**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
🔑 Название	VARCHAR(255)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
🔑 Производитель	VARCHAR(255)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
🔍 Мин_игроков	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
🔍 Макс_игроков	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
🔍 Мин_возраст	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

 Table Name:  Schema: **mydb**

Index Name	Type
PRIMARY	PRIMARY
ak_название_произв...	UNIQUE

Index Columns

Column	#	Order
<input checked="" type="checkbox"/> Название	1	ASC
<input checked="" type="checkbox"/> Производитель	2	ASC
<input type="checkbox"/> Мин_игроков		ASC
<input type="checkbox"/> Макс_игроков		ASC
<input type="checkbox"/> Мин_возраст		ASC

Index Options

Storage Type:

Key Block Size:

Parser:

Visible: ☒

Index Comment:



Table Name:  Schema: **mydb**

Index Name	Type
PRIMARY	PRIMARY
ak_почта	UNIQUE

Column	#	Order
<input type="checkbox"/> Ид_клиента		ASC
<input type="checkbox"/> Фамилия		ASC
<input type="checkbox"/> Имя		ASC
<input type="checkbox"/> Дата_рождения		ASC
<input type="checkbox"/> Телефон		ASC
<input checked="" type="checkbox"/> Почта	1	ASC
<input type="checkbox"/> Адрес		ASC

Index Options

Storage Type:

Key Block Size:

Parser:

Visible: ☐

Index Comment:

Table Name:  Schema: **mydb**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
⚡ Таб.Номер	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
💎 Фамилия	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
💎 Имя	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

2. Определите, какие из атрибутов отношений являются обязательными, т.е. не допускают наличие неизвестных значений. Пометьте такие атрибуты как NOT NULL.

**Решение:** Это показано в номере 1

3. Определите условия проверки значений для атрибутов (где возможно и, на ваш взгляд, имеет смысл). Условия требуется записать в виде логического выражения (предиката), в левой части которого указано имя атрибута, а в правой – значения, с которыми требуется произвести

сравнение. Допускается использование реляционных (>, =, <=, =, !=) и логических (AND, OR) операторов, а также оператора LIKE в том синтаксисе, с которым вы познакомились при выполнении лабораторной работы второй недели.

Для таблицы «ИГРА»:

```
1 • SELECT *
2 FROM ИГРА
3 WHERE (Название LIKE 'Диксит' OR Название LIKE 'Монополия' OR Название LIKE 'Jungle')
4       AND (Производитель = 'Libellud' OR Производитель = 'Asmodee' OR Производитель = 'Hasbro Inc')
5       AND Мин_игроков >= 1
6       AND Макс_игроков >= Мин_игроков
7       AND Мин_возраст >= 0;
8
```

Для таблицы «КЛИЕНТ»:

```
SELECT *
FROM КЛИЕНТ
WHERE (Фамилия LIKE 'Петров' OR Фамилия LIKE 'Швыркин')
      AND (Имя LIKE 'Петр' OR Имя LIKE 'Василий')
      AND Дата_рождения <= '2005-01-01'
      AND Телефон LIKE '8-%'
      AND (Почта LIKE '%@gmail.com' OR Почта LIKE '%@mail.ru');
```

Для таблицы «ЗАКАЗ»:

```
1 • SELECT *
2 FROM ЗАКАЗ
3 WHERE Номер >= 1
4       AND (Получение = 'Самовывоз' OR Получение = 'Доставка')
5       AND Создан >= '2017-09-12';
6
```

Для таблицы «СОТРУДНИК»:

```

1  •  SELECT *
2  FROM СОТРУДНИК
3  WHERE Таб.Номер LIKE 'C0%'
4      AND (Фамилия LIKE 'Жуков' OR Фамилия LIKE 'Василий')
5      AND (Имя LIKE 'Дмитрий' OR Имя LIKE 'Сергей');
6

```

4. Определите связи между отношениями.

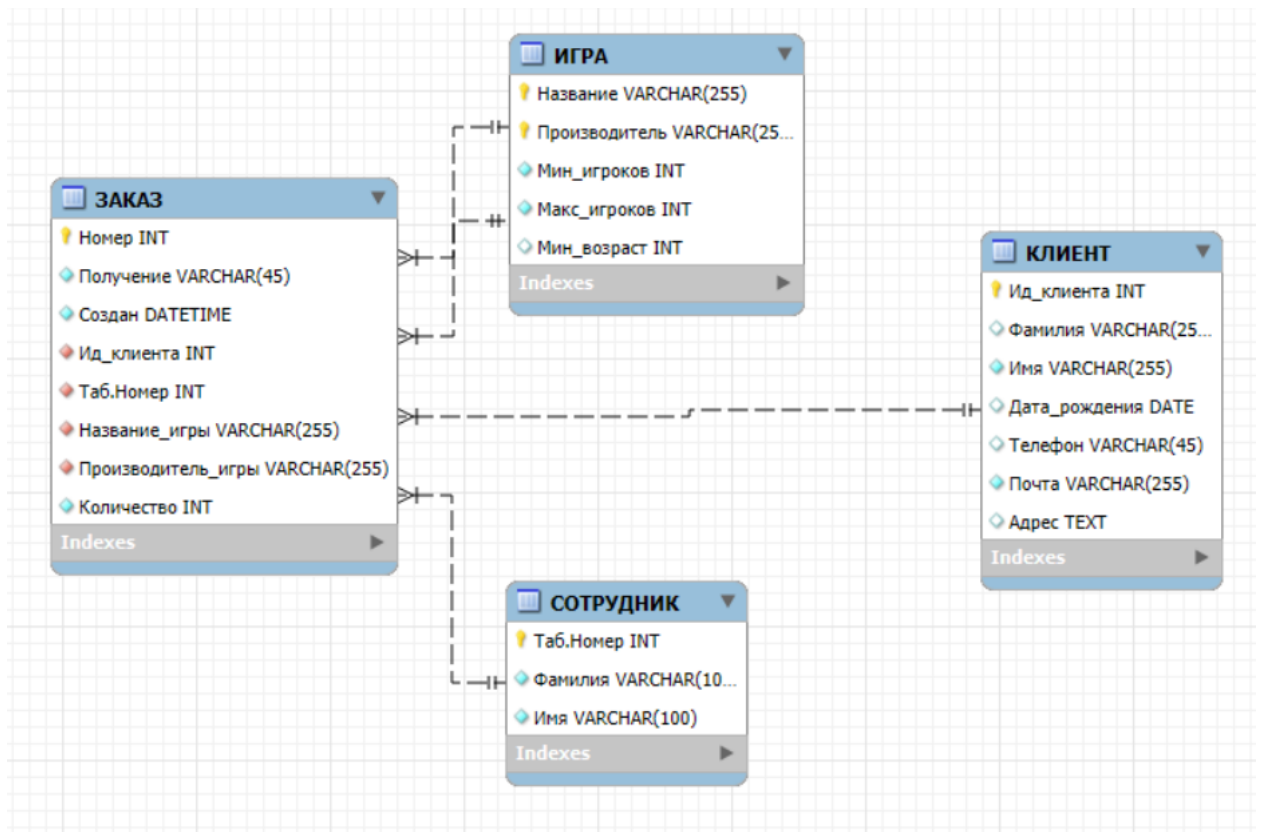
**Решение:**

- ИГРА (1) → ЗАКАЗ (N)
- КЛИЕНТ (1) → ЗАКАЗ (N)
- СОТРУДНИК (1) → ЗАКАЗ (N)

5. Установите связь между отношениями, добавив атрибуты первичного ключа главного отношения (того, к которому относится слово ОДИН) в состав дочернего (того, к которому относится слово МНОГО) и отметив их как внешний ключ (FK).

**Решение:**

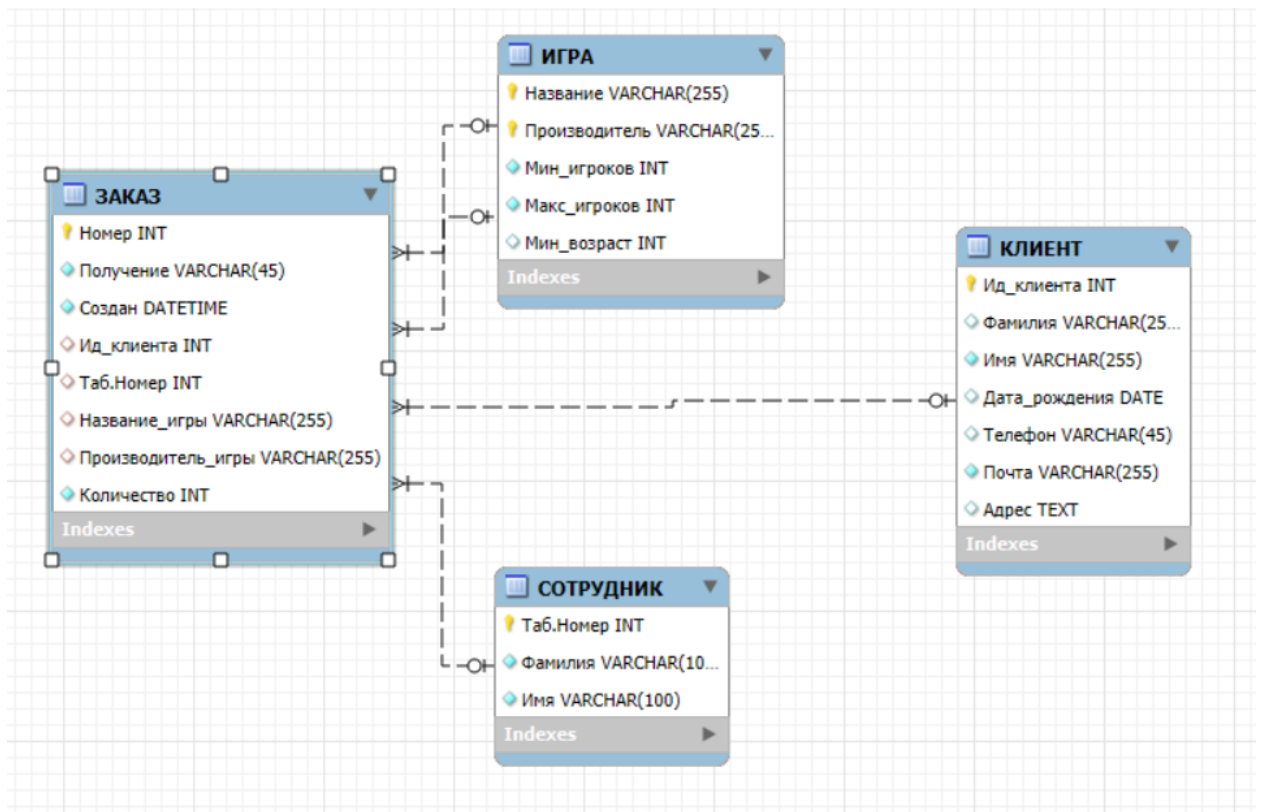
Table Name: ЗАКАЗ		Schema: mydb		
Foreign Key Name	Referenced Table	Column	Referenced Column	Foreign Ke
Название_игры	`mydb`.`ИГРА`	<input type="checkbox"/> Номер		On Up
Производитель_игры	`mydb`.`ИГРА`	<input type="checkbox"/> Получение		On D
Ид_клиента	`mydb`.`КЛИЕНТ`	<input type="checkbox"/> Создан		
Таб_номер	`mydb`.`СОТРУДНИК`	<input type="checkbox"/> Ид_клиента		
		<input checked="" type="checkbox"/> Таб.Номер	Таб.Номер	[
		<input type="checkbox"/> Название_игры		
		<input type="checkbox"/> Производитель...		
		<input type="checkbox"/> Количество		



6. Проверьте добавленные при установке связей атрибуты на обязательность, пометьте их как NOT NULL, если это требуется. Проверьте, не появились ли новые потенциальные ключи в результате добавления новых атрибутов.

### Решение:

Если изменить их на необязательные



Никаких новых потенциальных ключей не появилось

7. Ответьте на вопрос (письменно): Допустим, что каждый сотрудник может обработать только один заказ (после чего его увольняют ☹). В этом случае, связь между отношениями ЗАКАЗ и СОТРУДНИК трансформируется в один-к-одному. Что нужно будет сделать дополнительно для поддержки этой связи? Что может помешать указать одного и того же сотрудника в отношении ЗАКАЗ, ответственным за обработку нескольких разных заказов?

**Решение:**

Для поддержки связи один-к-одному между отношениями ЗАКАЗ и СОТРУДНИК, где каждый сотрудник может обработать только один заказ, необходимо выполнить несколько шагов:

- В таблице ЗАКАЗ должно быть установлено уникальное ограничение на поле, которое ссылается на Таб\_номер. Это гарантирует, что каждый сотрудник может быть назначен только на один заказ.
- Необходимо внедрить логику, которая автоматически увольняет сотрудника после завершения обработки заказа. Это может быть реализовано через триггеры в базе данных или через код приложения, который будет следить за статусом заказа и увольнять сотрудника по его завершении.
- Важно также обеспечить правильное обновление статусов заказов и сотрудников. После обработки заказа статус заказа должен измениться, а статус сотрудника — на "уволен".

Что может помешать указать одного и того же сотрудника ответственным за обработку нескольких заказов:

- Если не будет установлено уникальное ограничение, то система не сможет предотвратить назначение одного и того же сотрудника на несколько заказов, что нарушит логику один-к-одному.
- Если логика увольнения или обновления статусов реализована неверно, это может привести к ситуации, когда сотрудник остается активным и может быть назначен на новый заказ, несмотря на то, что он уже должен был быть уволен.
- Если несколько процессов одновременно пытаются назначить одного и того же сотрудника на разные заказы, это может привести к конфликтам и ошибкам в данных.