1. Задание 07

[Кейс 1 «Дизайн-студии и заказы» 6](#_Toc183773932)

[Кейс 2 «ЧП «Компьютерная помощь»» 7](#_Toc183773933)

[Кейс 3 «Администрирование компьютерных классов» 8](#_Toc183773934)

[Кейс 4 «Результаты тестов абитуриентов» 9](#_Toc183773935)

[Кейс 5 «Репетиторы и обучающиеся» 10](#_Toc183773936)

*Кейсы:*

Руководству сети дизайн-студий требуется хранить информацию об отдельных студиях и их клиентах. Каждый клиент заказывает продукцию у одной дизайн-студии. Каждая студия всегда выполняет заказы нескольких клиентов. Студии всегда выполняют заказы. Клиенты, которые не заказывают услуги, не представляют интерес для руководства.

ЧП «Компьютерная помощь» необходимо знать, какой мастер обслуживает каких клиентов. Каждый мастер обслуживает несколько клиентов. Каждый клиент может обслуживаться несколькими мастерами. Некоторые мастера могут быть временно без работы. Данные о клиентах, которые не обслуживаются, интерес для ЧП представляют.

Заведующему лабораториями кафедры ИКГ необходимо знать, кто из администраторов обслуживает тот или иной компьютерный класс. Каждый класс может обслуживаться несколькими администраторами. Один администратор может обслуживать несколько классов. Классов, которые не обслуживаются администраторами, не бывает. Учитываются только администраторы, которые обслуживают хотя бы один класс.

Заведующему кафедрой ИКГ необходимо знать, какие у абитуриентов результаты вступительного теста по компьютерной графике. Каждый абитуриент сдает тест 1 раз.

Сообществу независимых репетиров необходимо хранить информацию о репетиторах и об их учащихся. Каждый репетитор может обучать несколько учащихся. Каждый учащийся обучается одним репетитором. Учитывается информация обо всех репетиторах (даже если они временно ни с кем не занимаются). Информация об учащихся, с которыми не проводятся индивидуальные занятия, также хранится в БД.

*Для каждого кейса:*

* определите сущности и связи;
* перечислите предположения для СС и КП;
* нарисуйте диаграммы ER-экземпляров;
* нарисуйте диаграммы ER-типа;
* укажите подходящее правило;
* составьте отношения.

ПРАВИЛО 1. Если степень бинарной связи равна 1:1 и класс принадлежности обеих сущностей является обязательным, то требуется только одно отношение. Первичным ключом этого отношения может быть ключ любой из двух сущностей.

ПРАВИЛО 2. Если степень бинарной связи равна 1:1 и класс принадлежности одной сущности является обязательным, а другой - необязательным, то необходимо построение двух отношений. Под каждую сущность необходимо выделение одного отношения, при этом ключ сущности должен служить первичным ключом для соответствующего отношения. Кроме того, ключ сущности, для которого класс принадлежности является необязательным, добавляется в качестве атрибута в отношение, выделенное для сущности с обязательным классом принадлежности.

ПРАВИЛО 3. Если степень бинарной связи равна 1:1 и класс принадлежности ни одной сущности не является обязательным, то необходимо использовать три отношения: по одному для каждой сущности, ключи которых служат в качестве первичных в соответствующих отношениях, и одного для связи. Среди своих атрибутов отношение, выделяемое связи, будет иметь по одному ключу сущности от каждой сущности.

ПРАВИЛО 4. Если степень бинарной связи равна 1:n и класс принадлежности n-связной сущности является обязательным, то достаточным является использование двух отношений, по одному на каждую сущность, при условии, что ключ сущности каждой сущности служит в качестве первичного ключа для соответствующего отношения. Дополнительно ключ 1-связной сущности должен быть добавлен как атрибут в отношение, отводимое n-связной сущности.

ПРАВИЛО 5. Если степень бинарной связи равна 1:n и класс принадлежности n-связной сущности является необязательным, то необходимо формирование трех отношений: по одному для каждой сущности, причем ключ каждой сущности служит первичным ключом соответствующего отношения, и одного отношения для связи. Связь должна иметь среди своих атрибутов ключ сущности от каждой сущности.

ПРАВИЛО 6. Если степень бинарной связи равна m:n, то для хранения данных необходимо три отношения: по одному для каждой сущности, причем ключ каждой сущности используется в качестве первичного ключа соответствующего отношения, и одного отношения для связи. Последнее отношение должно иметь в числе своих атрибутов ключ сущности каждой сущности.

ПРАВИЛО 7. В случае трехсторонней связи необходимо использовать четыре предварительных отношения, по одному для каждой сущности, причем ключ каждой сущности должен служить в качестве первичного ключа для соответствующего отношения, и одно для связи. Отношение, порождаемое связью, будет иметь среди своих атрибутов ключи сущности от каждой сущности. (Аналогично, когда связь n-сторонняя, требуется n+1 предва-рительное отношение.)

ПРАВИЛО 8. Исходная сущность служит источником генерации одного отношения, причем ключ сущности служит в качестве ключа отношения. Ролевые элементы и связи, их соединяющие, порождают такое число отношений, которое определяется ранее описанными правилами, причем каждая роль трактуется как обычная сущность.

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Автоматически созданное описание

## «Дизайн-студии и заказы»

Руководству сети дизайн-студий требуется хранить информацию об отдельных студиях и их клиентах. Каждый клиент заказывает продукцию у одной дизайн-студии. Каждая студия всегда выполняет заказы нескольких клиентов. Студии всегда выполняют заказы. Клиенты, которые не заказывают услуги, не представляют интерес для руководства.

### Сущности и связи

*Сущности*

Студия, клиент, заказ

*Связи*

| Сущность 1 | Сущность 2 | Связь | Описание |
| --- | --- | --- | --- |
| Студия | Клиент | 1:М | Каждый клиент делает заказ только в одной студии, у каждой студии множество заказов |
| Студия | Заказ | 1:М | Студия выполняет множество заказов |
| Клиент | Заказ | 1:М | Заказ ассоциирован с одним клиентом, у клиента может быть множество заказов |

### Предположения для СС и КП

| Сущность | КП | Внешние ключи | Пояснение |
| --- | --- | --- | --- |
| Студия | Studio\_id | — |  |
| Клиент | Client\_id | Studio\_id, Order\_id | Клиент делает заказ **в студии** |
| Заказ | Order\_id | Client\_id | Каждый заказ выполняется **для клиента** |

### Диаграммы ER-экземпляров

### Диаграмма ER

### Правила

ПРАВИЛО 4. Если степень бинарной связи равна 1:n и класс принадлежности n-связной сущности является обязательным, то достаточным является использование двух отношений, по одному на каждую сущность, при условии, что ключ сущности каждой сущности служит в качестве первичного ключа для соответствующего отношения. Дополнительно ключ 1-связной сущности должен быть добавлен как атрибут в отношение, отводимое n-связной сущности.

### Отношения

| Сущность | Отношение |
| --- | --- |
| Студия | **Studio\_id**, address, phone |
| Клиент | **Client\_id**, name, phone, **Order\_id** |
| Заказ | **Order\_id**, order\_date, amount, client\_id, studio\_id |

### Схема БД

## «ЧП «Компьютерная помощь»»

ЧП «Компьютерная помощь» необходимо знать, какой мастер обслуживает каких клиентов. Каждый мастер обслуживает несколько клиентов. Каждый клиент может обслуживаться несколькими мастерами. Некоторые мастера могут быть временно без работы. Данные о клиентах, которые не обслуживаются, интерес для ЧП представляют.

### Сущности и связи

*Сущности*

Мастер, клиент

*Связи*

| Сущность 1 | Сущность 2 | Связь | Описание |
| --- | --- | --- | --- |
| Мастер | Клиент | М:М | Мастер может обслуживать множество клиентов, клиент может обслуживаться несколькими мастерами |

### Предположения для СС и КП

| Сущность | КП | Внешние ключи | Пояснение |
| --- | --- | --- | --- |
| Мастер | Master\_id | Client\_id | Мастер обслуживает клиентов, но клиентов в текущий момент может и не быть |
| Клиент | Client\_id | Master\_id | Каждый клиент обслуживается мастерами, клиент не может не обслуживаться мастерами — внешний ключ обязателен |

### Диаграммы ER-экземпляров

### Диаграмма ER

### Правила

### Отношения

| Сущность | Отношение |
| --- | --- |
| Мастер | **Master\_id**, name, specialization, phone |
| Клиент | **Client\_id,** name, contact, **Master\_id** |

### Схема БД

## «Администрирование компьютерных классов»

Заведующему лабораториями кафедры ИКГ необходимо знать, кто из администраторов обслуживает тот или иной компьютерный класс. Каждый класс может обслуживаться несколькими администраторами. Один администратор может обслуживать несколько классов. Классов, которые не обслуживаются администраторами, не бывает. Учитываются только администраторы, которые обслуживают хотя бы один класс.

### Сущности и связи

*Сущности*

Администратор, класс

*Связи*

| Сущность 1 | Сущность 2 | Связь | Описание |
| --- | --- | --- | --- |
| Администратор | Класс | М:М | Несколько администраторов могут обслуживать один класс, администратор может обслуживать несколько классов |

### Предположения для СС и КП

| Сущность | КП | Внешние ключи | Пояснение |
| --- | --- | --- | --- |
| Администратор | Admin\_id | Room\_numder | Каждый администратор обслуживает хотя бы один класс. |
| Класс | Room\_numder | Admin\_id | У каждого класса есть номер. Каждый класс обслуживается хотя бы одним администратором. |

### Диаграммы ER-экземпляров

### Диаграмма ER

### Правила

### Отношения

| Сущность | Отношение |
| --- | --- |
|  |  |

### Схема БД

## «Результаты тестов абитуриентов»

Заведующему кафедрой ИКГ необходимо знать, какие у абитуриентов результаты вступительного теста по компьютерной графике. Каждый абитуриент сдает тест 1 раз.

### Сущности и связи

*Сущности*

Абитуриент, результат

*Связи*

| Сущность 1 | Сущность 2 | Связь | Описание |
| --- | --- | --- | --- |
| Абитуриент | Результат | 1:1 | Каждый абитуриент сдает тест 1 раз. |

### Предположения для СС и КП

| Сущность | КП | Внешние ключи | Пояснение |
| --- | --- | --- | --- |
| Абитуриент | Abit\_id | — | Абитуриент есть. |
| Результат | Test\_id | Abit\_id | Каждый результат принадлежит какому-либо абитуриенту |

### Диаграммы ER-экземпляров

### Диаграмма ER

### Правила

### Отношения

| Сущность | Отношение |
| --- | --- |
|  |  |

### Схема БД

## «Репетиторы и обучающиеся»

Сообществу независимых репетиров необходимо хранить информацию о репетиторах и об их учащихся. Каждый репетитор может обучать несколько учащихся. Каждый учащийся обучается одним репетитором. Учитывается информация обо всех репетиторах (даже если они временно ни с кем не занимаются). Информация об учащихся, с которыми не проводятся индивидуальные занятия, также хранится в БД.

### Сущности и связи

*Сущности*

Репетитор, учащийся

*Связи*

| Сущность 1 | Сущность 2 | Связь | Описание |
| --- | --- | --- | --- |
| Репетитор | Учащийся | 1:М | У каждого репетитора может быть несколько учащихся, но обучающийся учится только у одного репетитора |

### Предположения для СС и КП

| Сущность | КП | Внешние ключи | Пояснение |
| --- | --- | --- | --- |
| Репетитор | Teacher\_id | Student\_id | Внешние ключи неявляются обязательными. У каждого репетитора может быть несколько учащихся, но обучающийся учится только у одного репетитора |
| Учащийся | Student\_id | Teacher\_id |

### Диаграммы ER-экземпляров

### Диаграмма ER

### Правила

### Отношения

| Сущность | Отношение |
| --- | --- |
|  |  |

### Схема БД