

# Использование связей



## Проверьте, как это работает!

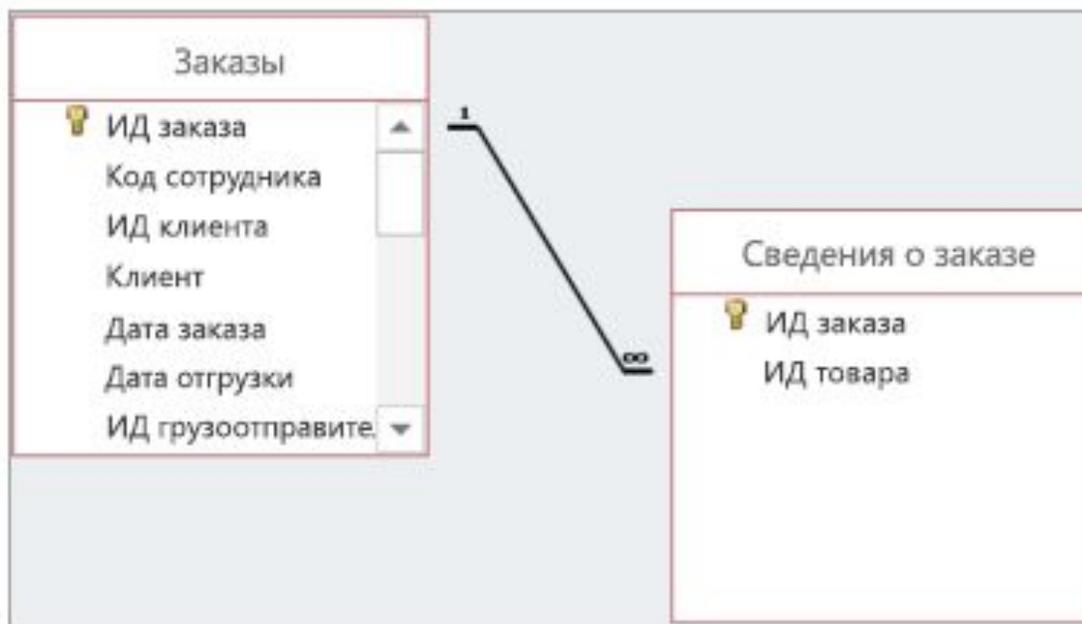
---

Главное достоинство реляционных баз данных заключается в возможности использовать информацию из разных таблиц. Для этого сначала требуется создать между ними связи. После этого вы сможете объединять эти данные в запросах, формах и отчетах.

Чтобы увидеть все связи в базе данных, откройте шаблон Access, а затем на вкладке **Работа с базами данных** нажмите кнопку **Схема данных**.

## Описание области "Схема данных"

Линии в представлении связей указывают соединения между таблицами. На приведенном ниже изображении Таблица слева — это **родительская** таблица. Таблица справа — это дочерняя таблица. Линии между ними соединяют поля (в этом случае — идентификатор заказа и код продукта), которые используются для сопоставления данных.



По линиям и символам можно определить параметры связи.

## Описание области "Схема данных"

---

- Толстая соединительная линия означает, что включено обеспечение целостности данных. Это хорошо. Данные будут синхронизироваться.
- На приведенном изображении цифра 1 означает, что в таблице слева может быть только одна связанная запись. В таблице "Заказы" каждому заказу может соответствовать только одна запись.
- Значок " $\infty$ " означает, что в нескольких записях может быть указан одинаковый номер или код. Заказ из таблицы слева, определяемый номером заказа, может быть указан в таблице "Сведения о заказах" несколько раз, поскольку в одном заказе может быть несколько продуктов.

# Виды связей между таблицами

---

- **Один-к-одному.** Каждый элемент используется в каждой таблице только один раз. Например, каждый сотрудник может использовать только один служебный автомобиль. Дополнительные сведения см. в статье [Создание связей типа "один-к-одному"](#).
- **Один-ко-многим.** Для одного элемента из первой таблицы можно создать связь с несколькими элементами из второй таблицы. Например, в каждой накладной может быть указано несколько продуктов.
- **Многие-ко-многим.** Для одного или нескольких элементов из первой таблицы можно создать связь с одним или несколькими элементами из второй таблицы. Например, в каждый заказ может входить несколько продуктов, и каждый продукт может быть указан в нескольких заказах. Дополнительные сведения см. в статье [Создание связей типа "многие-ко-многим"](#).

# Виды связей между таблицами

---

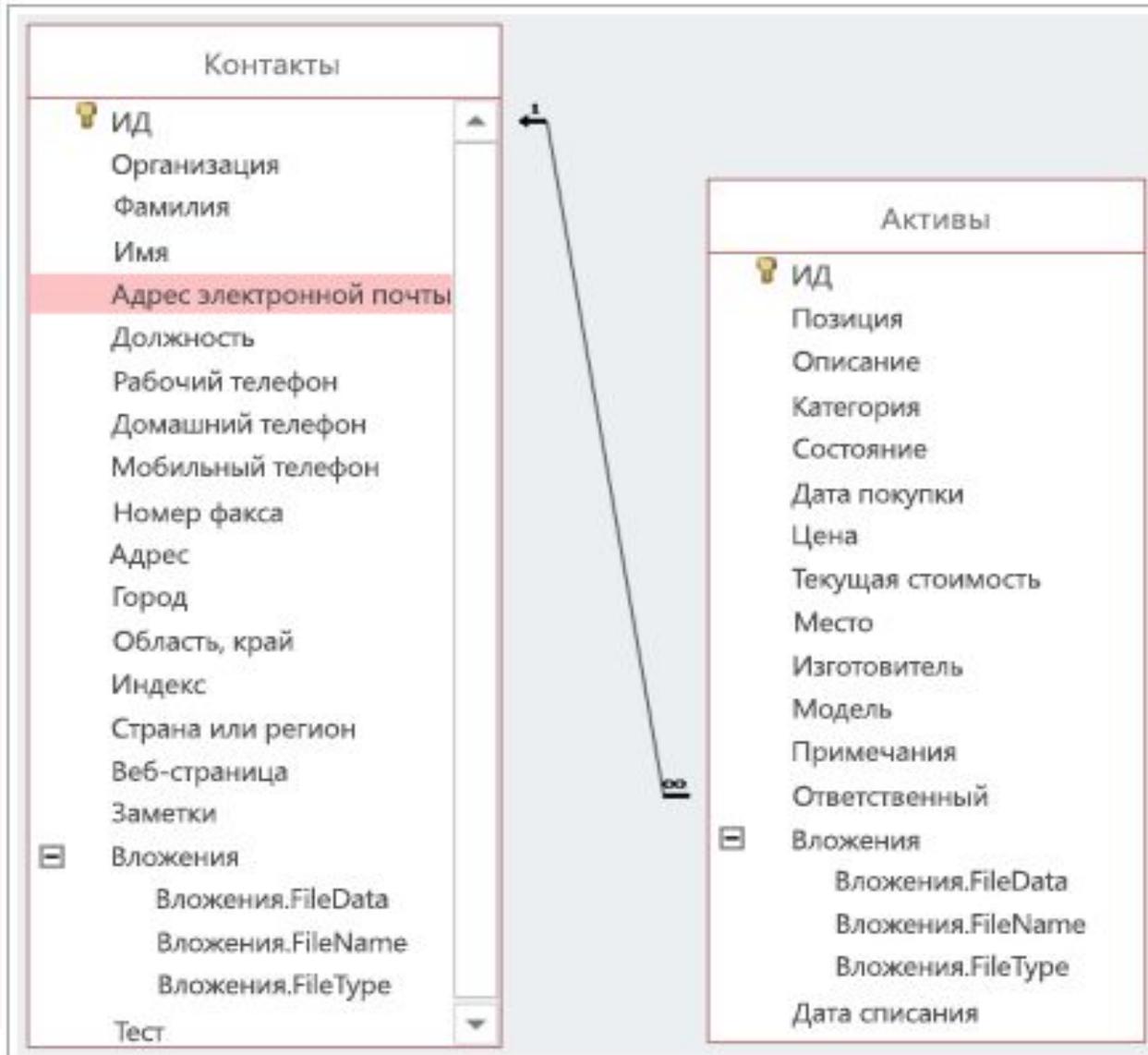
## Связи типа «один ко многим»

Связи типа "один-ко-многим" — одни из наиболее распространенных в хорошо структурированных базах данных.

Связи между таблицами обычно определяются первичным ключом в одной из них. Первичный ключ служит уникальным идентификатором каждой записи (часто числовым). Чтобы показать, что сведения в двух таблицах связаны, с помощью первичного ключа обычно создается связь, исходящая из одной таблицы.

В показанной ниже связи у каждого человека из таблицы "Контакты" есть идентификатор, представляющий собой первичный ключ (он отмечен значком ключа). Этот идентификатор также используется в поле "Владелец" в таблице "Активы". Чтобы написать электронное письмо человеку, связанному с активом, следует использовать значение поля "Адрес электронной почты". Для этого необходимо узнать значение поля "Владелец" из таблицы "Активы", а затем найти этот идентификатор в таблице "Контакты". Число **1** на одном конце соединительной линии и знак " $\infty$ " на другом означают, что это связь типа "один-ко-многим", поэтому один контакт может быть связан с несколькими активами.

# Виды связей между таблицами



# Изменение связи

Если вы работаете с существующей базой данных или создали базу из шаблона, вы можете изменить связь нужным образом.

1. Выберите **Работа с базами данных > Схема данных**.
2. Выберите линию, соединяющую две связанные таблицы.
3. На вкладке **Конструктор** нажмите кнопку **Изменить связи**.

Изменение связей

Таблица/запрос	Связанная таблица/запрос
Клиенты	Заказы
ИД	ИД клиента

Обеспечение целостности данных  
 каскадное обновление связанных полей  
 каскадное удаление связанных записей

Тип отношения: один-ко-многим

OK  
Отмена  
Объединение...  
Новое..

## Изменение связи

---

**Таблица/запрос** — это родительская таблица, указанная слева (в рассмотренном примере — «Клиенты»).

**Связанная таблица/запрос** — это дочерняя таблица (в рассмотренном примере — «Заказы»).

Даже если в области "Схема данных" таблицы расположены в другом порядке, их размещение в показанном диалоговом окне определяет направление соединительной линии между ними и, соответственно, направление связи. Это важно, например, для связей типа "один-ко-многим", потому что в таблице слева (родительской) в таком случае может быть связано только одно поле, а в таблице справа (дочерней) — несколько.

4. Чтобы изменить соединенные поля, выберите другое поле под каждой отображаемой таблицей. В рассмотренном примере поле "Код" из таблицы "Клиенты" соединяется с полем "Код клиента" из таблицы "Заказы".
5. Настройте синхронизацию данных между таблицами.

# Изменение связи

---

## **Обеспечение целостности данных**

Выберите этот пункт, чтобы избежать ошибок в данных и поддерживать синхронизацию сведений с помощью связи.

Например, предположим, что у вас есть связь типа "один-к-одному" между таблицами "Сотрудники" и "Льготы сотрудников". Если сотрудник уволится и вы удалите его из таблицы "Сотрудники", соответствующая запись в таблице "Льготы сотрудников" тоже удалится.

Иногда не имеет смысла принудительно обеспечивать целостность данных. Например, предположим, что у вас есть связь "один-ко-многим" между грузоотправителями и заказами. Вы удаляете грузоотправитель, и этот грузоотправитель соответствует заказам в таблице "заказы". Эти заказы становятся потерянными, то есть они по-прежнему содержат идентификационный номер, но этот идентификатор больше не является действительным, так как запись, на которую она ссылается, больше не существует.

## **Каскадное обновление связанных полей**

Установите этот флажок, чтобы данные в связанных полях обновлялись во всех связанных таблицах.

Предположим, вам нужно просто изменить код поставщика. Если у вас установлен этот флажок, код поставщика обновится не только в таблице "Поставщики", но и в других связанных с ней таблицах, в которых также используется это значение кода (например, в таблице "Заказы").

## **Каскадное удаление**

Установите этот флажок, если при удалении записей вам требуется удалять и связанные записи из других таблиц.

## Изменение связи

---

### **Каскадное удаление связанных записей**

Установите этот флажок, если при удалении записей вам требуется удалять и связанные записи из других таблиц.

Например, предположим, что вы удалили грузоотправитель. Если выбран этот параметр, Access удаляет все записи во всех таблицах, которые ссылаются на идентификационный код поставщика, включая все заказы (в таблице "заказы"), которые были отгружены этим грузоотправителем. Этот параметр можно выбрать только в том случае, если вы уверены, что хотите удалить историю заказов.

# Удаление отношения между таблицами

---

## Удаление связи между таблицами

1. Выберите **Работа с базами данных > Схема данных**.
2. Выберите линию, соединяющую две связанные таблицы.
3. Нажмите клавишу DELETE. Если потребуется подтвердить удаление, нажмите кнопку **Да**.