

Знакомство с Access



Проверьте, как это работает!

Базы данных и веб-приложения могут принести большую пользу вашему бизнесу. Проектирование базы данных играет важнейшую роль в достижении ваших целей независимо от того, что вам нужно: управлять сведениями о сотрудниках, предоставлять еженедельные отчеты по данным или отслеживать заказы клиентов. Уделив время изучению всех нюансов проектирования баз данных, вы сможете создавать базы, которые будут не только отвечать вашим текущим требованиям, но и адаптироваться к изменениям.

Понятия и термины

Для начала рассмотрим основные термины и понятия. Чтобы спроектировать полезную базу данных, необходимо создать таблицы с данными по одному объекту. В таблицах можно собрать все данные по этому объекту и отобразить их полях, которые содержат наименьшую единицу данных.

Реляционные базы данных

База данных, в которой данные разделены на *таблицы* по типу электронных. Каждая таблица включает данные по одному объекту, например по клиентам (одна таблица) или товарам (другая таблица).

Записи и поля

Области хранения отдельных данных в таблице. В каждой строке (*или записи*) хранится уникальный элемент данных, например имя клиента. Столбцы (*или поля*) содержат сведения по каждой точке данных в виде наименьшей единицы: имя может находиться в одном столбце, а фамилия — в другом.

Первичный ключ

Значение, которое обеспечивает уникальность каждой записи. Допустим, есть два клиента с одинаковыми именами, например Юрий Богданов. Но у одного из них первичный ключ записей — 12, а у другого — 58.

Иерархические отношения

Общие связи между таблицами. Например, один клиент может иметь несколько заказов. У родительских таблиц есть первичные ключи. Дочерние таблицы содержат *внешние ключи*, которые являются значениями первичного ключа, которые показывают, как записи подчиненной таблицы связаны с родительской таблицей. Эти ключи связаны *связью*.

Что понимать под хорошим проектированием базы данных?

В основе проектирования хорошей базы данных лежат два принципа:

- **Избегайте повторяющихся сведений** (избыточных данных). Они занимают много места на диске и повышают вероятность ошибок.
- **Следите за правильностью и полнотой данных.** Неполные или неправильные сведения отображаются в запросах и отчетах, что в конечном итоге может привести к принятию ошибочных решений.

Чтобы избежать этих проблем:

- Разделяйте информацию в базе данных по таблицам для отдельных объектов. Избегайте повторения информации в нескольких таблицах. (Например, имена клиентов должны находиться только в одной таблице.)
- Объединяйте таблицы с помощью ключей, а не путем дублирования данных.
- Используйте процессы, которые обеспечивают точность и целостность информации в базе данных.
- Проектируйте базу данных с учетом своих требований к обработке данных и созданию отчетов по ним.

Чтобы повысить пользу баз данных в долгосрочной перспективе, выполните следующие пять шагов по проектированию:

Что понимать под хорошим проектированием базы данных?

Шаг 1. Определение назначения базы данных

Прежде чем начать, сформулируйте цель базы данных.

Шаг 2. Поиск и упорядочение необходимых сведений

Соберите все типы данных, которые необходимо записывать, например названия товаров и номера заказов.

Шаг 3. Разделение данных по таблицам

Разделите элементы данных на основные объекты, например товары, клиенты или заказы. Каждый объект выносится в таблицу.

Шаг 4. Превращение элементов данных в столбцы

Определите, какие данные необходимо хранить в каждой таблице. Эти отдельные элементы данных становятся полями в таблице. Например, таблица "Сотрудники" может содержать такие поля, как "Фамилия", "Имя" и "Дата приема на работу".

Что понимать под хорошим проектированием базы данных?

Шаг 5. Задание первичных ключей

Выберите первичный ключ для каждой таблицы. Первичный ключ, например код товара или код заказа, является уникальным идентификатором каждой записи. Если у вас нет явного уникального идентификатора, его можно создать с помощью Access.