

**Тема 2**

**Проектирование и  
информационное наполнение  
БД в СУБД MS Access**

# План

- 1. Проектирование таблиц**
- 2. Создание связей**
- 3. Создание форм**

# 1. Проектирование таблиц

Перед созданием любой базы данных следует провести предварительный анализ задачи и определить задействованные поля, чтобы в дальнейшем избежать лишних затрат памяти и использовать преимущества реляционной модели. При этом необходимо соблюдать следующие правила.

## **Правило 1: 1-Я НОРМАЛЬНАЯ ФОРМА**

***Согласно 1-й нормальной форме, в каждой ячейке располагается минимальная единица информации.***

Нельзя записывать несколько значений через запятую. Если по логике вещей такое решение напрашивается, значит объектов на самом деле несколько и для каждого из них по правилам следует заводить отдельную таблицу.

**Нормальная форма** — свойство отношения в реляционной модели данных, характеризующее его с точки зрения избыточности, потенциально приводящей к логически ошибочным результатам выборки или изменения данных.

Нормальная форма определяется как совокупность требований, которым должно удовлетворять отношение.

**Процесс преобразования отношений базы данных (БД) к виду, отвечающему нормальным формам, называется нормализацией.**

Нормализация предназначена для приведения структуры БД к виду, обеспечивающему минимальную логическую избыточность, и не имеет целью уменьшение или увеличение производительности работы или же уменьшение или увеличение физического объёма базы данных. Конечной целью нормализации является уменьшение потенциальной противоречивости хранимой в

**К. Дейт сформулировал общее назначение процесса нормализации:**

- исключение некоторых типов избыточности;**
- устранение некоторых аномалий обновления;**
- разработка проекта базы данных, который является достаточно «качественным» представлением реального мира, интуитивно понятен и может служить хорошей основой для последующего расширения;**
- упрощение процедуры применения необходимых ограничений целостности.**

ПРАВИЛЬНО

LastName	FirstName	MiddleName
Пушкин	Александр	Сергеевич

НЕПРАВИЛЬНО

Name
Пушкин Александр Сергеевич

ПРАВИЛЬНО

OrderDate	ReturnDate
13.02.05	28.02.05

НЕПРАВИЛЬНО

Dates
13.02.05-28.02.05

ПРАВИЛЬНО

City	Address
Москва	ул. Лесная, 23, кв.50

НЕПРАВИЛЬНО

Address
Москва, ул. Лесная, 23, кв.50

**Правило 2: Каждое поле любой таблицы должно быть уникальным.**

*Если значения в каком-то поле регулярно повторяются, их следует выносить в отдельную таблицу, а в поле ставить на них ссылку.*

Первые два правила определяют третье: **одна сущность (объект) – одна таблица.**

Сущность — объект предметной области, имеющий атрибуты (свойства).

**Основные элементы ER-моделей (моделей сущность – связь, англ. “Entity-Relationship model”, реализует который реляционная структура БД):**

- **объекты (сущности);**
- **атрибуты объектов (свойства);**
- **связи между объектами.**

Таблица 1 -

Преподаватели

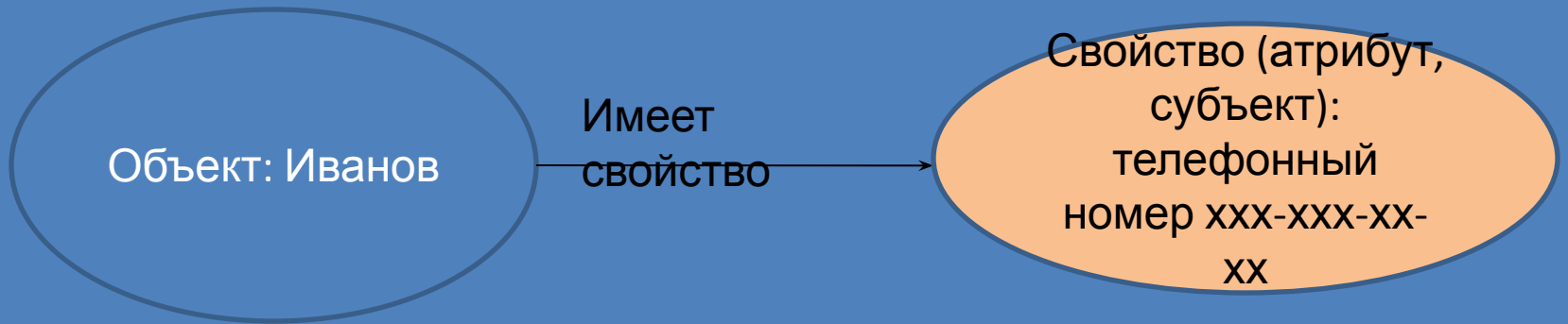
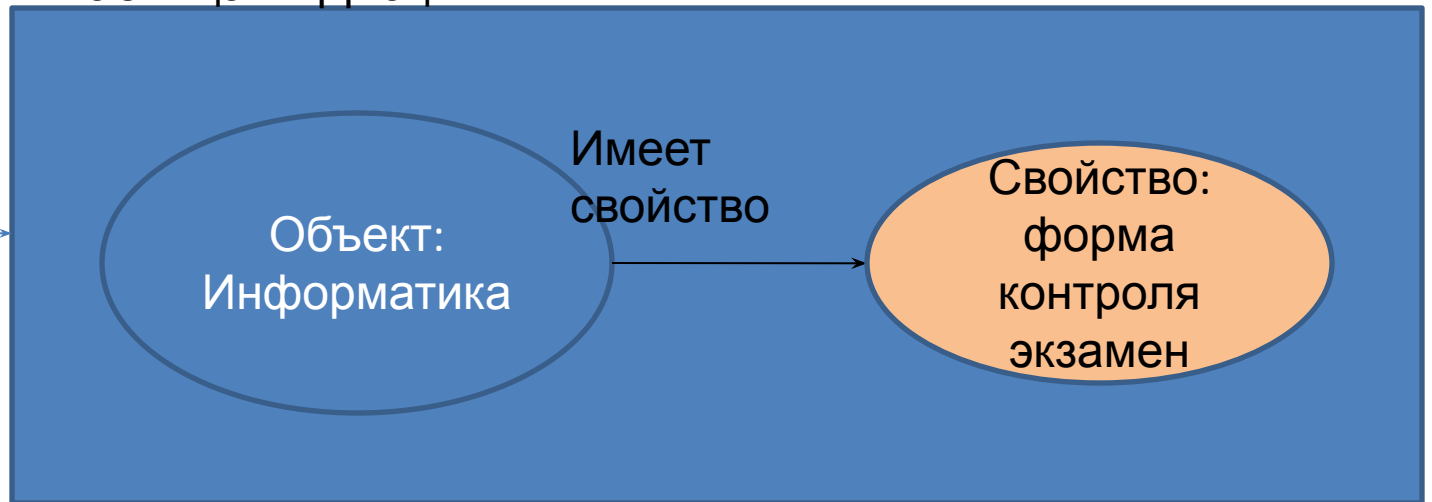


Таблица 2 - Дисциплины



СВЯЗЬ  
ОДИН –  
КО  
МНОГИМ

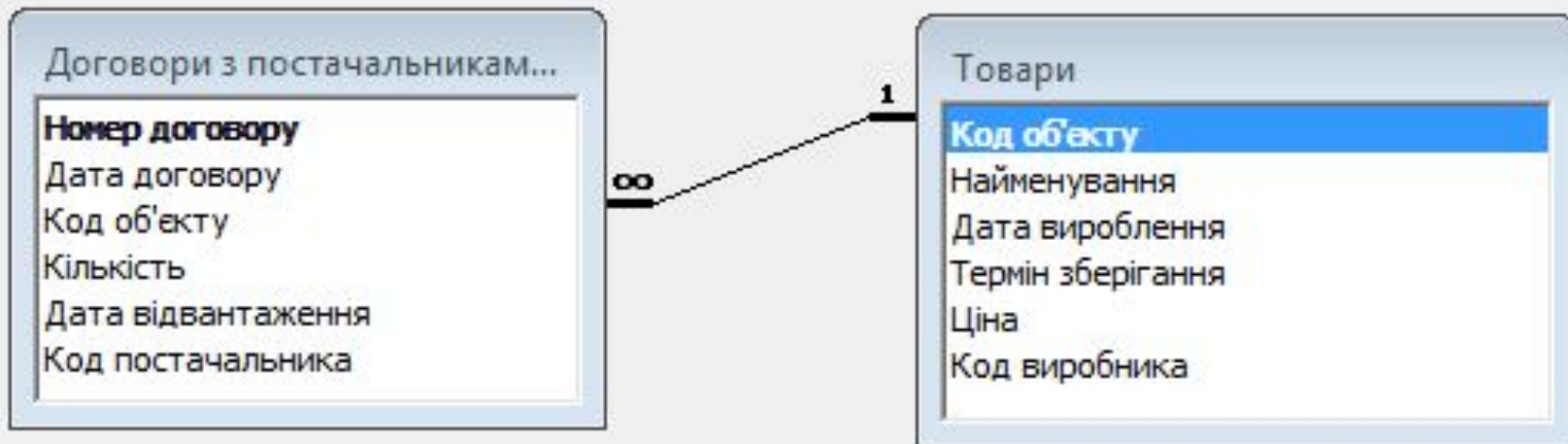


## 2. Создание связей

Создание модели данных реляционной структуры включает в себя создание связей между таблицами.

Связи в реляционных базах данных определяются по совпадению значений полей в разных таблицах. Например, объекты и заказ объектов связаны отношением "**один-ко-многим**", поскольку одной записи в таблице, содержащей сведения об объектах, может соответствовать несколько записей в таблице заказов этих объектов.

Одна из таблиц, где ключ первичный и совпадения в нем не допускаются, **главная**, а вторая, где ключ вторичный и совпадения в нем допускаются, **подчиненная**. Иначе говорят, что первичный ключ в родительской таблице соответствует множеству внешних ключей в дочерней таблице. Это отношение наиболее часто встречается в базах данных.



**Тип зв'язи один – ко – многим, или один к бесконечности**

Если рассмотреть отношения между преподавателями и курсами лекций, которые они читают, это будет отношение "многие-ко-многим", поскольку один преподаватель может читать несколько курсов, но и один курс может читаться несколькими преподавателями. В базе данных это отношение требует создания третьей таблицы, называемой связующей, или сводной таблицей. Эта таблица содержит в качестве внешних (вторичных) ключей первичные ключи обеих таблиц, для создания перекрестных ссылок.



Третий тип связей между таблицами - это отношение "один-к-одному". Такой тип отношений встречается гораздо реже. Как правило, это бывает в двух случаях: запись имеет большое количество полей, и тогда данные об одном типе объектов разносятся по двух связанных таблицах, или нужно определить дополнительные атрибуты для некоторого количества записей в таблице, тогда создается отдельная таблица для этих дополнительных атрибутов, которая связывается отношением "один-к-одному" с основной таблицей.



Для обеспечения целостности, работа с данными должна проводиться с учетом нижеперечисленных правил.

- Невозможно ввести в связанное поле подчиненной таблицы значение, отсутствующее в связанном поле главной таблицы. Однако можно ввести пустое значение, показывающее, что для данной записи связь отсутствует.
- Не допускается удаление записи из главной таблицы, если существуют связанные с ней записи в подчиненной таблице.
- Невозможно изменить значение ключевого поля в главной таблице, если существуют записи, связанные с данной таблицей.

Чтобы эти правила контролировались для конкретной связи, при ее создании следует установить флажок «Обеспечение целостности данных» (Обеспечение целостности данных, Enforce Referential Integrity). Тогда любая попытка выполнить действие, нарушающее одно из перечисленных выше правил, приведет к выводу на экран предупреждения, а само действие выполнено не будет.

Чтобы преодолеть ограничения на удаление или изменение связанных записей, сохраняя при этом целостность данных, следует установить флажок "Каскадное обновление связанных полей" (Каскадное обновление связанных полей, Cascade Update Related Fields) и "Каскадное удаление связанных записей" (Каскадное удаление связанных записей, Cascade Delete Related Records). Если установлен флажок "Каскадное обновление связанных полей", то при изменении ключевого поля главной таблицы автоматически будут изменены и соответствующие значения поля связанных записей. Если установлен флажок "Каскадное удаление связанных записей", то при удалении записи в главной таблице удаляются и все связанные записи в подчиненной таблице.

# Изменение связей



Таблица/запрос:

Связанная таблица/запрос:

Товари



Договоры з поставчальни



Код об'екту



Код об'екту



Обеспечение целостности данных

каскадное обновление связанных полей

каскадное удаление связанных записей

Тип отношения:

один-ко-многим

OK

Отмена

Объединение...

Новое..

### 3. Создание форм

Формы предназначены для быстрого и удобного заполнения таблиц, причем существует как вариант одна форма - одна таблица, так и вариант составная форма - две таблицы. Также благодаря формам можно просматривать, редактировать, осуществлять быстрый поиск необходимых записей.

Пример. На вкладке «Формы» выберите "Создание формы с помощью мастера". В качестве источника данных формы выберите таблицу «Справочник сотрудников».



## Создание форм



Выберите поля для формы.

Допускается выбор нескольких таблиц или запросов.

Таблицы и запросы

Таблица: Довідник співробітн

Доступные поля:

Empty box for available fields.



Выбранные поля:

Код співробітника  
ПІБ співробітника  
Контактні дані (телефон, е-мейл  
Адреса  
Паспортні дані  
Посада

Отмена

< Назад

Далее >

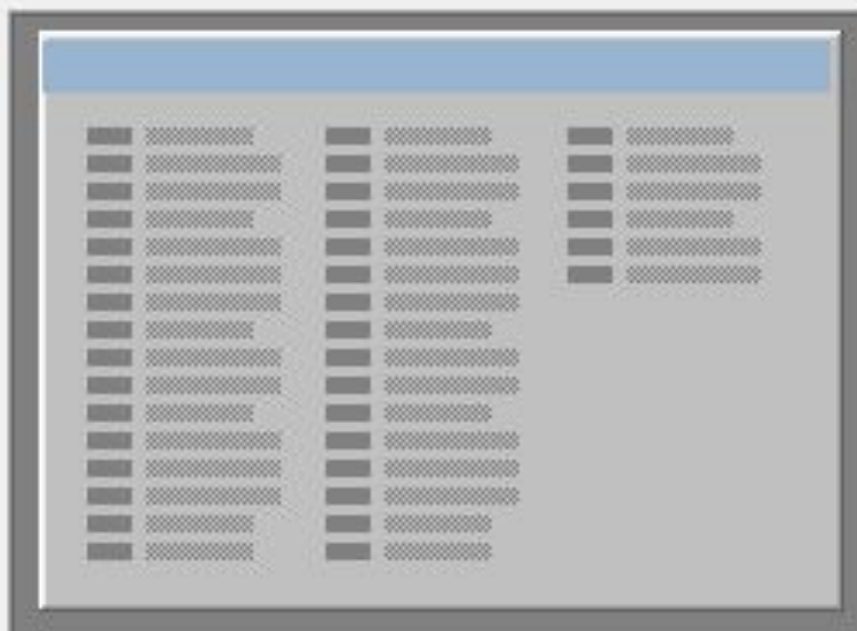
Готово

Вы должны пройти шаги мастера:

1. выберите нужные поля таблицы как поля формы. Нажмите кнопку «Далее»;
2. на следующем шаге мастера выберите внешний вид формы - в один столбец. Нажмите кнопку «Далее»;
3. выберите необходимый стиль формы. Нажмите кнопку «Далее»;
4. задайте имя формы - "Справочник сотрудников". Селектор "Дальнейшие действия" переключите на открытие формы для просмотра или ввода данных. Нажмите кнопку «Готово». Созданная форма открыта в режиме пользователя для ввода или просмотра данных связанной с ней таблицы.

## Создание форм

Выберите внешний вид формы:



- в один столбец
- ленточный
- табличный
- выровненный
- сводная таблица
- сводная диаграмма

Отмена

< Назад

Далее >

Готово

Код співробітника

Счетчик

ПІБ співробітника

Контактні дані (телефон)

Адреса

Паспортні дані

Посада

После открытия по умолчанию форма выводит на экран первую запись. Для просмотра других записей связанной таблицы существуют кнопки «Первая запись», «Предыдущая запись», «Следующая запись», «Последняя запись». Для создания новой записи предназначена кнопка «Новая запись». Между кнопками находится "Поле номера записи". Таким образом, через созданную форму можно работать с таблицей-источником. Перед введением новой записи через форму необходимо нажать кнопку «Новая запись». Для текущей записи возможно редактирование. Для отмены ввода или редактирования необходимо нажать клавишу Esc на клавиатуре.

Код співробітника

ПІБ співробітника

Контактні дані (телефон, е-мейл)

Адреса

Паспортні дані

Посада

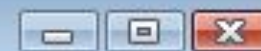


Аналогичную форму можно создать быстрее, если выбрать необходимую таблицу в списке таблиц на соответствующей вкладке и нажать на панели инструментов кнопку «Новый объект: автоформа». По умолчанию это будет форма в один столбец.

Если выбрать для быстрого создания таблицу которая является главной в связи "один ко многим" с другой таблицей (т.е. находится в связи на стороне "1"), то MS Access создаст составную форму, и через такую форму пользователь может заполнять одновременно две таблицы, которые находятся в связи. Это более удобно, рационально и быстро.

Подчиненная форма при этом – табличная.

Довідник клієнтів



Код клієнта

Найменування клієнта

Контактні дані (телефон, е-мейл)

Адреса

Реквізити

	Код замовлення	Дата замовлення	Код співробітника	Код товару	Кількість	Дата оплати
▶	(Счетчик)		▼		0	

Запись: 1 из 1

Запись: 1 из 1



Полученная форма в числе других полей имеет поле со списком. Это поле автоматически было создано, потому что таблица-источник уже имела соответствующее поле со списком. В противном случае его необходимо создавать в конструкторе форм.