**Лекция 1. Тема: Базы данных.**

База данных (БД) — именованная совокупность данных, отражающая состояние объектов и их отношений в рассматриваемой предметной области, или иначе БД — это совокупность взаимосвязанных данных при такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений в определенной предметной области. БД состоит из множества связанных файлов.



Система управления базами данных (СУБД) — совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями.

СУБД – Система управления базами данных. Это инструмент, которым удобно работать с самой концепцией. СУБД – инструмент!

Это всё равно учить математику, через калькулятор. Потому что всё равно все наши знания будут зависеть от калькулятора и как мы выполняем те или иные действия на нём.

Здесь мы поговорим о языках и программных средствах, которые будут выполнять те или иные действия. Но основное наше время будет уделено концепции, почему же БД работает именно так и что-же происходит под «капотом». И что бы было если бы у нас не было СУБД.

Стержневые идеи современных информационных технологий базируются на концепции баз данных.

Согласно этой концепции, основой информационных технологий являются данные, которые должны быть организованы в базы данных в целях адекватного отображения изменяющегося реального мира и удовлетворения информационных потребностей пользователей.

Одним из важнейших понятий в теории баз данных является понятие информации. Под информацией понимаются любые сведения о каком-либо событии, процессе, объекте.

Данные — это информация, представленная в определенном виде, позволяющем автоматизировать ее сбор, хранение и дальнейшую обработку человеком или информационным средством. Для компьютерных технологий данные — это информация в дискретном, фиксированном виде, удобная для хранения, обработки на ЭВМ, а также для передачи по каналам связи.

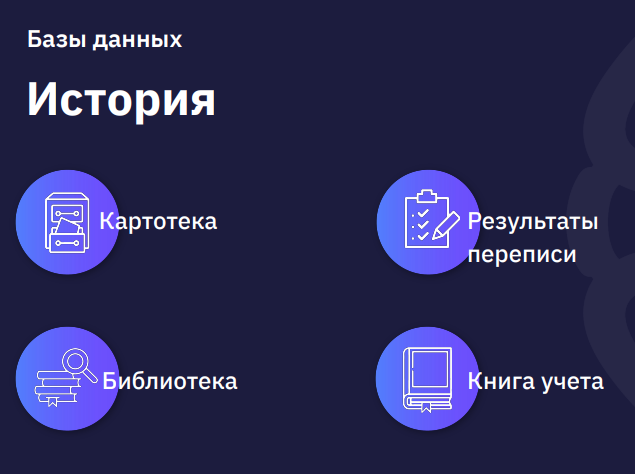
Автоматизированная информационная система (АИС) — это система, реализующая автоматизированный сбор, обработку, манипулирование данными, функционирующая на основе ЭВМ и других технических средств и включающая соответствующее программное обеспечение (ПО) и персонал. В дальнейшем в этом качестве будет использоваться термин информационная система (ИС), который подразумевает понятие автоматизированная.

Каждая ИС в зависимости от ее назначения имеет дело с той или иной частью реального мира, которую принято называть предметной областью (ПрО) системы. Выявление ПрО — это необходимый начальный этап разработки любой ИС. Именно на этом этапе определяются информационные потребности всей совокупности пользователей будущей системы, которые, в свою очередь, предопределяют содержание ее базы данных.

Банк данных (БнД) является разновидностью ИС. БнД — это система специальным образом организованных данных: баз данных, программных, технических, языковых, организационно-методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования данных.

Под задачами обработки данных обычно понимается специальный класс решаемых на ЭВМ задач, связанных с видом, хранением, сортировкой, отбором по заданному условию и группировкой записей однородной структуры.

Отдельные программы или комплекс программ, реализующие автоматизацию решения прикладных задач обработки данных, называются приложениями. Приложения, созданные средствами СУБД, относят к приложениям СУБД. Приложения, созданные вне среды СУБД с помощью систем программирования, использующих средства доступа к БД, к примеру, Delphi или Visual Studio, называют внешними приложениями.



Раньше была БД, но удобной и точной настройки инструментов не было, как и компьютеров. И соответственно всё это происходило вручную. Например, результаты переписи в России в 1897 г., поэтому результаты переписи обработали к 1905 г. (8 лет обработки!!!)

**Основные функции БД:**

1. Для хранения.
2. Обработки.
3. Быстрого извлечения необходимой информации.

Таким образом, БД может быть определена как:

Совокупность предназначенных для машинной обработки интегрированных данных, служащая для удовлетворения нужд многих пользователей.

В связи с концепцией баз данных еще раз подчеркнем два ключевых момента: 1. Информация уже не скрыта в сочетании “файл-программа”; она хранится явным образом в БД. БД ориентирована на интегрированные запросы, а не на одну задачу. 2. Возможность выделения по запросу из всех данных, хранящихся в БД, только необходимых и в требуемой форме (структуре и форматах). Наиболее широко БД используются в управленческой деятельности благодаря следующим свойствам:

• Скорость. Возможный доступ к информации за требуемое время.

• Полная доступность. Вся информация, содержащаяся в БД, доступна для использования (с учетом, конечно, необходимости засекречивания и защиты).

• Гибкость. Легко вносимые изменения и дополнения в БД позволяют получать ответы на вопросы, которые ранее оставались без ответа.

• Целостность. Уменьшилась степень дублирования данных и ликвидирована их противоречивость; упорядочился процесс обновления и восстановления БД после сбоев; появилась возможность управления параллельного общения с БД нескольких прикладных программ. Другими словами, под базой данных понимается некоторая унифицированная совокупность данных, совместно используемая персоналом/населением группы, предприятия, региона, страны, мира... Задача базы данных состоит в хранении всех представляющих интерес данных в одном или нескольких местах, причем таким способом, который заведомо исключает ненужную избыточность. В хорошо спроектированной базе данных избыточность данных исключается, и вероятность сохранения противоречивых данных минимизируется. Таким образом, создание баз данных преследует две основные цели: понизить избыточность данных и повысить их надежность.

Замечание. Таким образом, СУБД решают множество проблем, которые затруднительно или вообще невозможно решить при использовании файловых систем. При этом существуют приложения, для которых вполне достаточно файлов, и приложения, для которых безусловно нужны базы данных. Другими словами, современные системы управления файлами и управления базами данных представляют собой весьма совершенные инструменты, каждый из которых может быть успешно применен в соответствующей области деятельности. Но всегда необходимо помнить, что каждый инструмент приносит максимальную пользу именно в той области, для которой он создан.

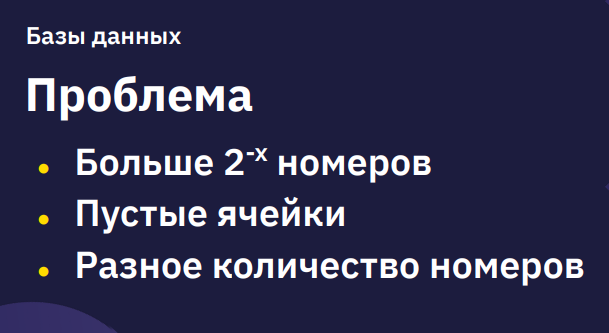
Существует иерархическая модель хранения данных. Это модель реальности, которая нас полностью не удовлетворяет. Есть ограничения! Есть картины, которые писал не один художник, а также несколько направлений (жанров), написание портретов. Эта иерархия удобна только для определённого количества задач. Если же нам предстоит решить другую задачу, то эта иерархия будет сложной, для решения наших задач и есть некоторые проблемы, которых мы не можем избежать. Но можно поменять эту иерархическую модель, чтобы наши все вопросы в неё помещались. И для этого у нас есть решение. Когда у нас не было СУБД компьютеров и всё было в виде картотек, книг учёта и т.д. Это всё можно было решить с помощью предметного указателя. В чём проблема иерархической модели, она очень негибкая, она сразу определяет сценарий использования этой БД, как мы можем их получать и обрабатывать. Попробуем другую модель на основе примера, она стала доступной уже с возникновением компьютеров. И так, необходимо создать свой собственный телефонный справочник, для этого воспользуемся таблицей Excel.

|  |  |
| --- | --- |
| ФИО | Номер телефона |
| Иванов Иван Иванович | 123456 |
| Сидоров Семён Петрович | 654321 |
| Васильев Василий Васильевич | 456789 |

БД – это некоторая модель реальности и реальность может далеко не всегда укладываться в ту модель, которую мы создали. У многих сейчас в реальном мире не один номер телефона, может быть один номер телефона, два, три и т.д. и не понятно, как мы сюда можем это поместить. Но вроде как мы можем выйти из этого положения добавив ещё один столбик в нашей таблице.

|  |  |  |
| --- | --- | --- |
| ФИО | Номер телефона | Дополнительный номер телефона |
| Иванов Иван Иванович | 123456 | 123 |
| Сидоров Семён Петрович | 654321 | 312 |
| Васильев Василий Васильевич | 456789 | 456 |

Если номер действительно только один, мы можем оставить пустую ячейку или добавить в ней прочерк.



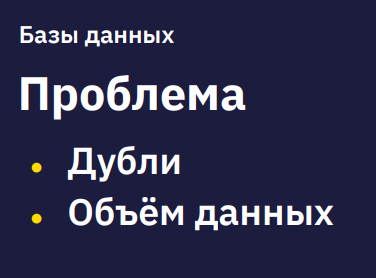
Можно также указать в столбце «Примечания» для каких целей эти телефоны. В данном случае, это как с массивами, если мы заранее не знаем количество элементов, то нам тяжело определить массив для хранения нашей нужной информации. Попробуем решить данную задачу добавив в таблицу до 5 номеров.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ФИО | Телефон 1 | Телефон 2 | Телефон 3 | Телефон 4 | Телефон 5 |
| Иванов Иван Иванович | 123456 | 123 | 879 | 876 | 578 |
| Сидоров Семён Петрович | 654321 | 312 | 231 | - | - |
| Васильев Василий Васильевич | 456789 | 456 | - | - | - |

Да, но осталась проблема, что у кого-то ещё больше 5 номеров телефона. Но и образовалась ещё одна проблема, у кого-то 5 номеров, у кого-то 3, а у кого-то всего только 2. Получается, что из-за одного только (с 5 номерами) возникает огромное количество незаполненных ячеек в нашей теперь достаточно большой таблице. То есть образовалось в таблице много места без какой-то информации. Какая ещё проблема: у Иванова 5 номеров, а какой из них личный, домашний, рабочий и т.д. мы не знаем. Давайте взглянем одно из подобных решений проблемы с этой таблицей.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ФИО | Телефон | Комментарий |  | ФИО | Телефон | Комментарий |
| Иванов Иван Иванович | 123456 | Личный |  | Сидоров Семён Петрович | 123 | Личный |
| Иванов Иван Иванович | 123 | Рабочий |  | Сидоров Семён Петрович | 312 | Старый, не работает, возможно |
| Иванов Иван Иванович | 879 | Для поездок |  | Сидоров Семён Петрович | 231 | Рабочий |
| Иванов Иван Иванович | 876 | Для интернета |  | Васильев Василий Васильевич | 456789 | Личный |
| Иванов Иван Иванович | 578 | Для знакомств |  | Васильев Василий Васильевич | 456 | рабочий |

В этом случае у Иванова может быть хоть 300 номеров телефона, и я к каждому могу добавить свой комментарий. То есть в данном случае таблица становится не в ширину, а в длину. Получается это как вариант для неограниченного хранения телефонов. Но некоторые наши проблемы остались.

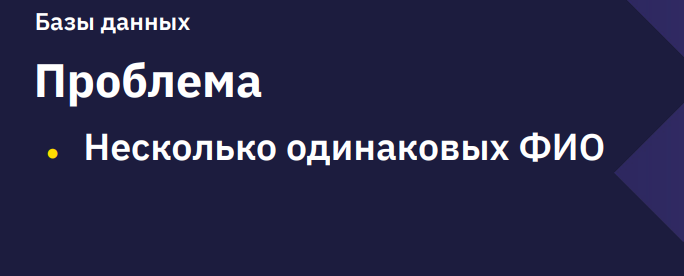


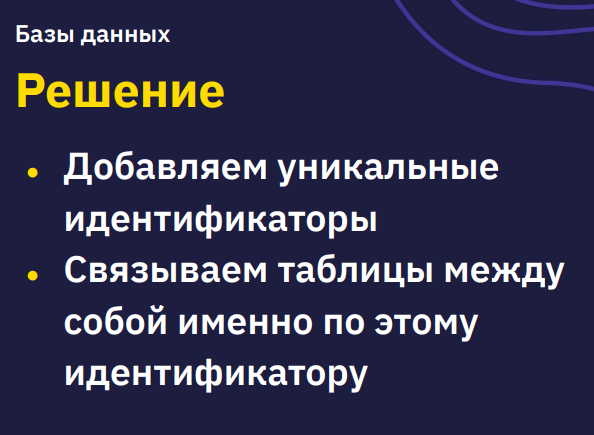
Например.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ФИО** | **Тел** | **Коммент** | **Адрес** | **Д/р** | **Статус** |
| Иванов Иван Иванович | 123456 | Личный | Казань | 20.01.1970 | Женат |
| Иванов Иван Иванович | 123 | Рабочий | Казань | 20.01.1970 | Женат |
| Иванов Иван Иванович | 879 | Для поездок | Казань | 20.01.1970 | Женат |
| Иванов Иван Иванович | 876 | Для интернета | Казань | 20.01.1970 | Женат |
| Иванов Иван Иванович | 578 | Для знакомств | Казань | 20.01.1970 | Женат |

Мы добавили в таблицу некоторую актуальную информацию, адрес, день рождения, статус. Которые должны храниться о каждом человеке, мало того мы взяли эту информацию только по Иванову, другие у нас просто не поместились в данной таблице. Такая форма хранения информации хранит нам много сложностей в управлении ей и по мимо этого нам нужно её дублировать много раз. Если у человека 5 номеров, то мы её дублируем 5 раз, а если 300 номеров, то дублировать 300 раз. По мимо того может поменяться адрес, статус и комментарий, а также нам нужно гораздо больше информации о человеке (семья, дети, родственники, друзья). И в итоге мы придём к тому, что наша таблица вырастет в ширину и следить за ней становится всё более сложно. Для решения этой проблемы нам понадобится доп. таблица.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Чей тел | Тел | Коммент |  | Чей тел | Тел | Коммент |
| Иванов Иван Иванович | 123456 | Личный |  | Сидоров Семён Петрович | 123 | Личный |
| Иванов Иван Иванович | 123 | Рабочий |  | Сидоров Семён Петрович | 312 | Старый, не работает, возможно |
| Иванов Иван Иванович | 879 | Для поездок |  | Сидоров Семён Петрович | 231 | Рабочий |
| Иванов Иван Иванович | 876 | Для интернета |  | Васильев Василий Васильевич | 456789 | Личный |
| Иванов Иван Иванович | 578 | Для знакомств |  | Васильев Василий Васильевич | 456 | рабочий |



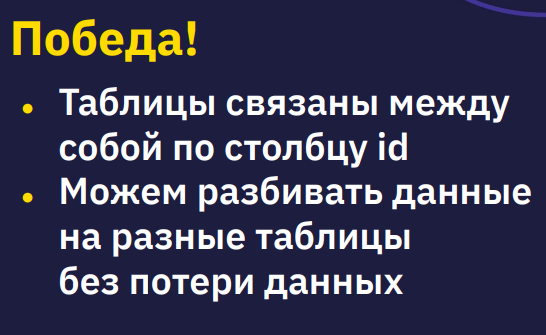


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | ФИО | Адрес | Д/Р | Статус |
| 1 | Иванов И.И. | Казань | 20.01.1970 | Женат |
| 2 | Иванов И.И. | Санкт-Петербург | 20.01.1990 | Холост |
| 3 | Сидоров С.П. | Москва | 23.04.1983 | Женат |
| 4 | Васильев В. В. | Белгород | 05.10.1989 | Холост |

Мы добавили столбец идентификатор и добавили ещё одного Иванова И.И., так как мы добавили идентификатор, то нас уже не интересует Иванов из Казани и Иванов из Санкт-Петербурга. Так как у них есть свой идентификатор, по которому мы и будем ориентироваться в таблице.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Чей тел | Тел | Коммент |  | 2 | 644 | Личный |
| 1 | 123456 | Личный |  | 3 | 123 | Личный |
| 1 | 123 | Рабочий |  | 3 | 312 | Старый, не работает, возможно |
| 1 | 879 | Для поездок |  | 3 | 231 | Рабочий |
| 1 | 876 | Для интернета |  | 4 | 456789 | Личный |
| 1 | 578 | Для знакомств |  | 4 | 456 | Рабочий |

Таким образом, с помощью идентификатора мы связали две таблицы. Иначе столбец id, называется первичный ключ. Первичный ключ, это уникальная информация, которая позволяет нам идентифицировать каждую конкретную запись, которая есть в нашей таблице. В последней таблицу роль идентификатора выполняет столбец «Чей тел» и тут он уже называется «Внешний ключ». Главное, чтобы мы поняли, для чего нам нужен первичный и внешний ключ, а роль идентификатора – это взаимосвязь двух таблиц и более. При этом мы получили огромную пользу, что разбили две таблицы, но связь с ними поддерживается идентификатором.



Мы с вами создали связи между таблицами и в некотором роде это наши реляционные БД (или таблицы), в которых данные распределены по отдельно связанным таблицам. У нас есть несколько таблиц, которые нас связывают, это и есть реляционная БД.

