

1. ТРЕБОВАНИЯ К БАЗАМ ДАННЫХ

Структура базы данных == Схема базы данных

- Что хранится
- Как хранится
- Права доступа

1.1. Типы данных.

- Простые (числа, строки)
- Структурированные (адрес, телефон)
- Сущность (студент, группа) - любой различимый объект, который может быть представлен в базе данных. Между сущностями могут быть связи (студент учится в группе).

Сущности отличаются от структурированных данных тем, что равные сущности не обязательно указывают на одно и то же (два студента с одинаковыми инициалами).

1.2. Ограничение целостности. Ограничения целостности представляют собой ограничения, налагаемые на значения, которые разрешено принимать некоторой переменной, или комбинации переменных.

- На отдельные значение (возраст не может быть равен -3)
- На записи (если запись состоит из начала и конца, хотим, чтобы конец всегда был больше начала)
- На наборы записей (номера паспортов)
- На связи данных (студент учится в одной группе)

2. РАЗВИТИЕ БАЗ ДАННЫХ

2.1. Простые и Структурированные файлы.

Модель данных	Структура	Замечания
простой файл	Заголовок (названия столбцов) Данные (просто текст)	Чтобы что-то найти, придется прочитать все данные.
структур. файл	Заголовок (названия столбцов, типы и длины) Данные (записи одинаковой структуры)	Теперь зная номер записи, мы можем к ней перейти.

Достоинства

- Простота чтения

Недостатки

- Сложность поиска (особенно, если нужно найти строки, удовлетворяющему какому-то нетривиальному критерию (к примеру, на несколько элементов данных))
- Сложность обработки
- Сложность изменения
- Сложность хранения данных разных типов
- Нет проверки целостности

2.2. Более интересные модели.

Модель	Представление данных	Замечания
Файловые системы	Файл – одна запись Каталоги – подчиненные записи Пример: Иванов И.И./Оценки/Java – 4	Достоинства: - Структурирование данных - Простота реализации Недостатки: - Сложно извлекать требуемые данные - Нет проверки целостности - Большое количество файлов
Иерархические базы данных	Храним дерево записей, для каждой записи знаем ее тип и структуру (т.е. можем это проверять) Отношения родитель – ребенок	Недалеко ушло от файловой системы, но хранится в меньшем числе файлов и следит за корректной структурой всех записей. Достоинства: - Проверка целостности записей и отношений - Последовательное расположение записей - Эффективность реализации Недостатки: - Представление только древовидных данных - Нет отношения многие-ко-многим
Сетевые базы данных	Ориентированный граф записей Отношения владлец – запись У записи может быть более чем один владлец (к примеру, у оценки может быть владлец предмет и студент)	Достоинства - Представление всех типов связей - Возможность описания структуры - Эффективность реализации Недостатки - Сложность реализации - Жесткое ограничение структуры
Реляционные базы данных	Данные хранятся в таблицах Проверка целостности заданных связей Связи задаются в запросах	Достоинства - Представление всех типов связей - Гибкая структура данных - Математическая модель Недостатки - Сложность реализации - Сложность представления иерархических данных - Сложность составления эффективных запросов
Объектные базы данных	Сущность – объект Связь – поле Ограничения целостности – определение объекта	Достоинства - Простота представления объектов - Гибкая структура данных - «Логичное» направление ссылок Недостатки - Сложность реализации - Сложность миграции схемы - Малая распространенность