

Язык запросов SQL

Базы данных

База данных - это контейнер (обычно файл или группа файлов) для хранения упорядоченных данных.

Базы данных

Классификации СУБД по модели данных:

- Иерархические
- Сетевые
- Реляционные
- Объектно-ориентированные
- Объектно-реляционные

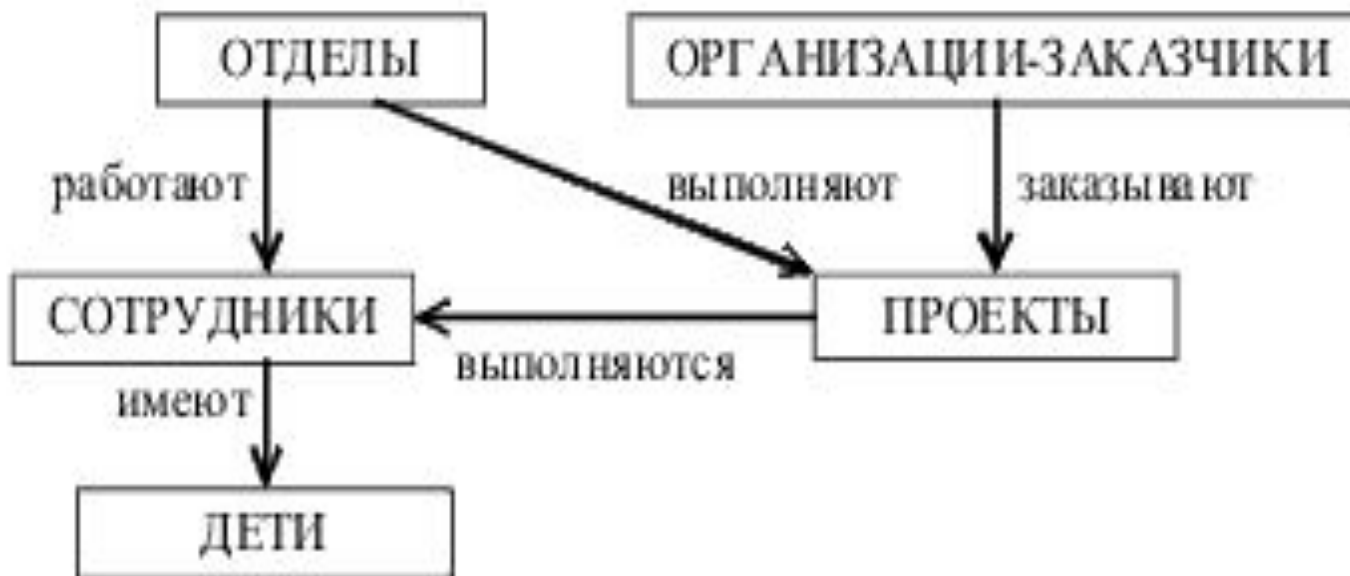
Базы данных

Иерархическая структура представляет совокупность элементов, связанных между собой по определенным правилам.



Базы данных

Сетевые базы данных (навигационные) - представление данных в виде произвольного графа позволяет представлять структуру данных любой сложности.



Базы данных

1961-64 гг. - первая система управления базами данных IDS - Integrated Data Store (Чарльз Бахман).

СУБД это - компьютеризованная система ведения записей. Она позволяет сохранять информацию и обеспечивает доступ к ней.

Базы данных

Начало 80-х г. - Объектно-ориентированная модель данных

Объектно-ориентированная модель данных

Базы данных

Основные функции СУБД

- управление данными во внешней памяти (на дисках);
- управление данными в оперативной памяти с использованием дискового кэша;
- журнализация изменений, резервное копирование и восстановление базы данных после сбоев;
- поддержка языков БД (язык определения данных, язык манипулирования данными).

Базы данных

По степени распределенности

- Локальные СУБД (все части размещаются на одном компьютере)
- Распределённые СУБД (части СУБД могут размещаться на двух и более компьютерах).

Базы данных

По способу доступа к БД

- Файл-серверные

Microsoft Access, Paradox, dBase, FoxPro, Visual FoxPro

- Клиент-серверные

Oracle, Firebird, Interbase, IBM DB2, Informix, MS SQL Server, Sybase Adaptive Server Enterprise, PostgreSQL, MySQL

- Встраиваемые

OpenEdge, SQLite, BerkeleyDB, Firebird Embedded, Microsoft SQL Server Compact, ЛИНТЕР

Базы данных

Начало реляционной теории баз
1969 г. - Эдгар Фред Кодд -
возможность управления данными с
использованием математической
теории отношений

Отношение (таблица) является
множеством кортежей (строк, записей),
состоящих из атрибутов (столбцов,
полей). Для каждого атрибута
определен тип данных.

Реляционная база данных

таблица **Students**:

Код_студента	Фамилия_студента	Институт	Курс
116	Петров	ИМИ	4
673	Иванов	ФТИ	2
1150	Сидоров	ИМИ	3

Реляционная база данных

Первичный ключ таблицы - это уникальный идентификатор строк.

Реляционная база данных

Виды отношений в

РБД:

- **‘один-к-одному’** (записывается как ‘1:1’)
- **‘один-ко-многим’** (записывается как ‘1:n’)
- **‘мноغو-к-одному’** (записывается как ‘n:1’)
- **‘мноغو-ко-многим** (записывается как ‘m:n’)

Реляционная база данных

таблица **Progress** :

Код_предмета	Код_студента	Оценка
01	673	3
07	116	4
02	1150	5
01	1150	4

Типы данных

Строковые типы данных:

Типы данных	Описание
CHAR(size)	Строки фиксированной длиной (могут содержать буквы, цифры и специальные символы). Фиксированный размер указан в скобках. Не более 255 символов
VARCHAR(size)	Может хранить не более 255 символов.
TINYTEXT	Может хранить не более 255 символов.
TEXT	Может хранить не более 65 535 символов.

Типы данных

Типы с плавающей точкой:

Типы данных	Описание
TINYINT(size)	Может хранить числа от -128 до 127
SMALLINT(size)	Диапазон от -32 768 до 32 767
MEDIUMINT(size)	Диапазон от -8 388 608 до 8 388 607
INT(size)	Диапазон от -2 147 483 648 до 2 147 483 647
BIGINT(size)	Диапазон от -9 223 372 036 854 775 808 до 9 223 372 036 854 775 807
FLOAT(size,d)	Число с плавающей точкой небольшой точности.
DOUBLE(size,d)	Число с плавающей точкой двойной точности.

Типы данных

Типы с плавающей точкой:

Типы данных	Описание
<code>DATE()</code>	Дата в формате ГГГГ-ММ-ДД
<code>DATETIME()</code>	Дата и время в формате ГГГГ-ММ-ДД ЧЧ:ММ:СС

Типы данных

Пример:

‘1’ не равно 1

‘1’ + ‘1’ не равно 2

‘1’ не равно ‘1 ’

‘01’ не равно ‘1’

Реляционная база данных

Правила для таблиц РБД

- каждая таблица должна иметь уникальное имя
- названия полей в одной таблице не должны совпадать
- каждое значение поля должно соответствовать своему определенному типу.
- каждая таблица должна описывать только один вид данных.

Введение в SQL

SQL - эс-кью-эль (Structured Query Language - структурированный язык запросов), универсальный язык, применяемый для создания, модификации и управления данными в реляционных базах данных.

Преимущества SQL:

- SQL поддерживают все современные СУБД
- прост в изучении
- позволяет выполнять очень сложные операции с базами данных

Введение в SQL

- **SQL** позволяет создавать таблицы данных.
- **SQL** позволяет хранить данные.
- **SQL** позволяет получать данные.
- **SQL** позволяет изменять данные.
- **SQL** позволяет изменять структуру таблиц.
- **SQL** позволяет объединять данные.
- **SQL** позволяет выполнять вычисления.
- **SQL** позволяет обеспечивать защиту данных.

Основные реализации SQL

- MySQL
- Oracle Database
- PostgreSQL
- Microsoft SQL Server
- Borland Interbase Server
- Lotus Notes

Два вида SQL

- Интерактивный SQL
- Вложенный SQL

Подразделы SQL

- DDL - Язык Определения Данных
- DML - Язык Манипулирования Данными
- DCD - Язык Управления Данными

Команды SQL

- **SELECT** - выборка строк из таблиц
- **CREATE TABLE** - создание таблицы
- **INSERT** - ввод строки в таблицу

Команда SELECT

SELECT *поле1, поле2*
FROM *имя_таблицы;*

Команда SELECT

```
SELECT *  
FROM имя_таблицы;
```

Команда SELECT

SELECT *поле3, поле1, поле5*
FROM *имя_таблицы;*

Команда SELECT

```
SELECT поле3, поле1  
INTO новая_табл  
FROM имя_таблицы;
```

Команда SELECT

SELECT *поле3, поле1* --комментарий1
FROM *имя_таблицы;* --- комментарий2

Создание и удаление таблиц

```
CREATE TABLE [имя_таблицы](  
поле1(тип, NULL / NOT NULL),  
поле2(тип, NULL / NOT NULL));
```

```
DROP TABLE [имя_таблицы];
```


Ввод строк в таблицу

INSERT INTO *Students* **VALUES** (
1453, 'Смирнов', 'ФЭИ', 5)

INSERT INTO *Students* **VALUES** (
5432, 'Кузьмин', 'ФТИ', 3)

INSERT INTO *Students* **VALUES** (
876, 'Попов', 'ИМИ', 2);

Ввод строк в таблицу

Неправильная команда:

```
INSERT INTO Students VALUES (  
'ИМИ', 'Попов', 876, 2);
```

Следующая команда не выдаст ошибки:

```
INSERT INTO Students VALUES (  
876, 'ИМИ', 'Попов', 2);
```

Литература

1. Баканов В.М. Введение в язык SQL запросов к базам данных
2. Дж.Грофф, П.Вайнберг. SQL: Полное руководство
3. Семенова И.И. SQL стандарт в СУБД MS SQL Server, Oracle, VFP и Access
4. Форта Бен. Освой самостоятельно SQL. 10 минут на урок

1. Создать таблицу **Workers**, содержащую сведения о работниках предприятия. Таблица должна иметь следующие поля: **snum** - номер работника, **worker** - ФИО работника, **bonus** - премия, **qualification** - квалификация.

2. Занести в таблицу следующие данные:

snum	worker	bonus	qualification
1	Ковалев Павел Иванович	10000	2 разряд
2	Сидоров Антон Николаевич	20000	1 разряд
3	Москвитина Жанна Дмитриевна	18000	2 разряд
4	Протодряконова Любовь Аркадьевна	10000	2 разряд
5	Синицын Николай Петрович	5000	3 разряд
6	Томский Геннадий Егорович	25000	1 разряд
7	Жуков Петр Дмитриевич	11000	2 разряд

3. Какое поле в таблице **Workers** будет первичным ключом? Может ли это поле содержать пустые значения? Почему?

4. С помощью команды **SELECT** выбрать из таблицы **Workers** все поля.

5. Выбрать из таблицы **Workers** поля в следующем порядке: номер, ФИО работника, бонус.

6. На основе таблицы **Workers** создать новую таблицу **Workers2** со следующими полями: работник, бонус, квалификация.

7. Может ли таблица **Workers2** содержать одинаковые записи и почему?

8. Удалите таблицу **Workers2**.