

SQL. Вводный курс

Лекция №1: Базы данных

Что такое SQL и для чего он нужен?

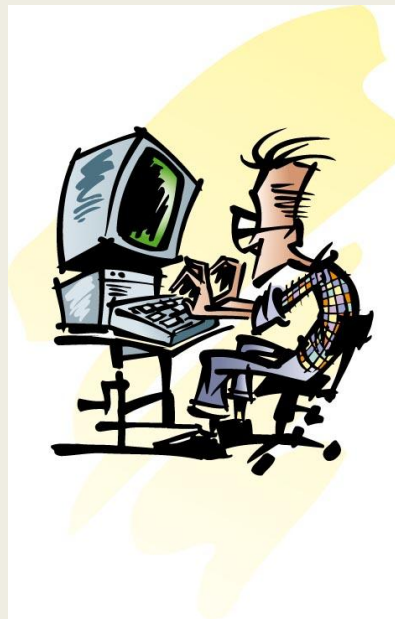
S STRUCTURED

Q QUERY

L LANGUAGE

← "язык структурированных запросов"

SQL - это универсальное средство - **язык** - для общения с различными видами Баз Данных



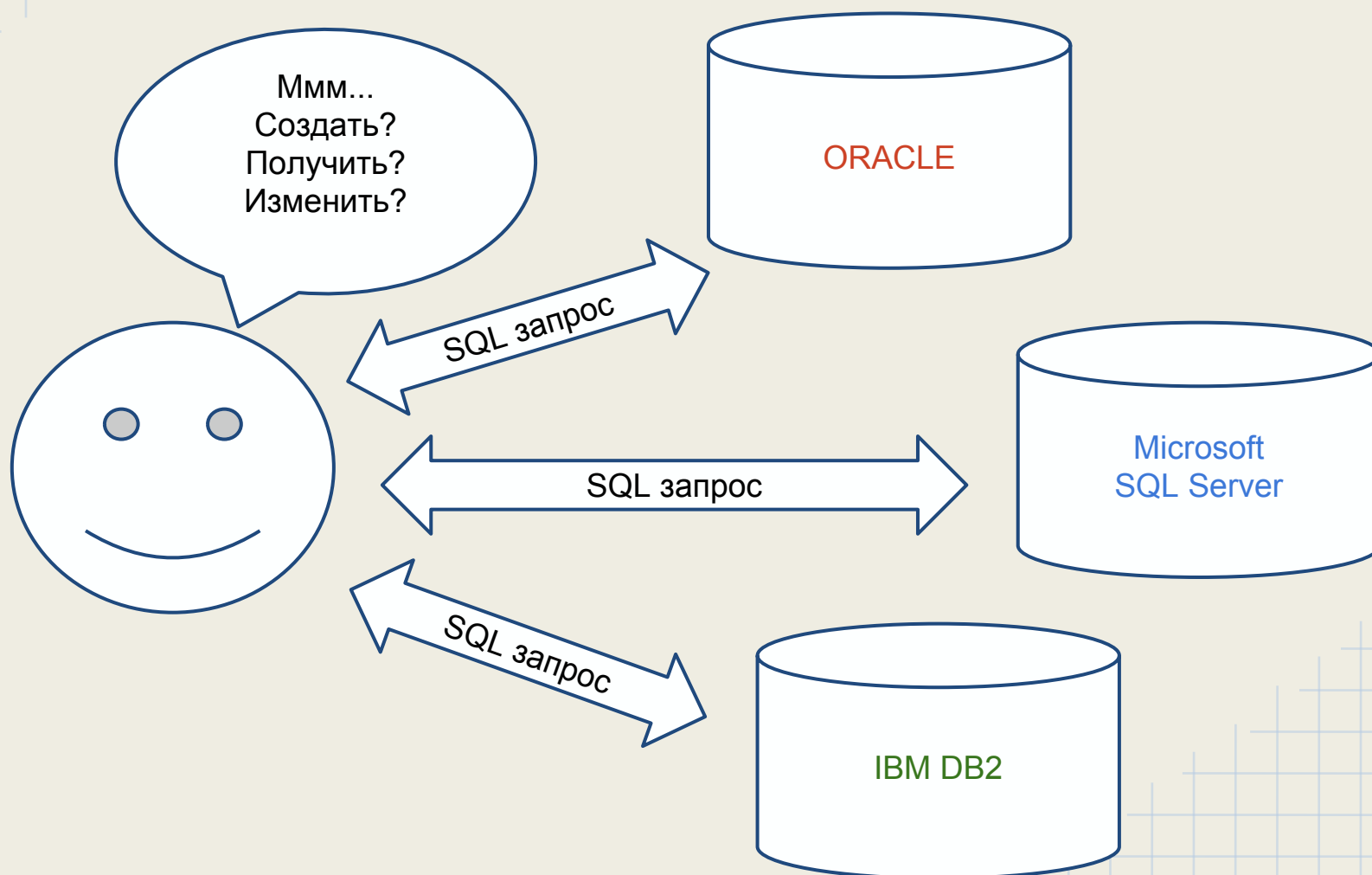
ХОЧУ...

Создать

Узнать

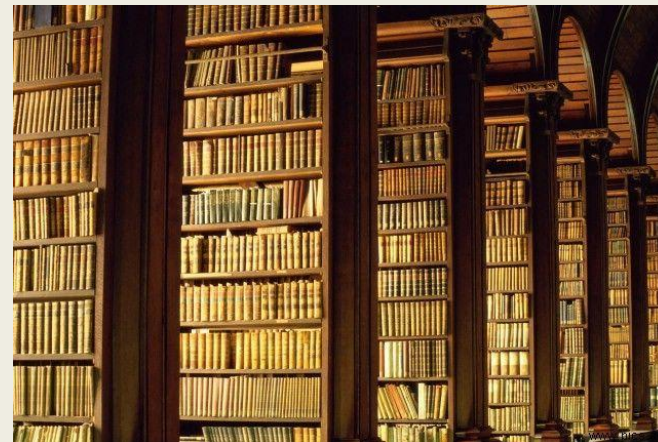
Изменить

Универсальный язык (стандарт)



Что такое БД?

Базы Данных -
это хранилище информации.



Любой набор информации может быть представлен как БД.
С точки зрения пользователя - важно представление,
в итоге получаем...

БД под носом!

Ежедневно мы сталкиваемся с Базами Данных и сервисом доступа к ним



Из чего же состоит База Данных?

Во-первых, таблиц

Таблица в свою очередь состоит из столбцов и строк.

Столбец - это поле, определенного типа, либо уникальных, либо повторяющихся значений.

Строке соответствует конкретная запись таблицы - это как некое предложение, состоящее из набора значений полей.

Во-вторых...

Поле-1 Н/П	Поле-2 ФИО	Поле-3 Курс	Поле-4 Стипендия
1	Кабанов	4	575.50
2	Павленко	1	625.35
3	Запорожец	3	400.00
4	Шуст	4	625.35
6	Чайко	5	810.50

Связи между таблицами

Между таблицами могут быть связи.

Существование связей между таблицами обеспечивается наличие так называемых ключей: первичный и вторичный.

Первичный (он же Primary key) - в таблице только один, более того он уникален, может быть составным

Что касается вторичных (они же Foreign keys), то их в таблице может присутствовать несколько.

Основная таблица
Экзамены

PK	FK1	...	FK2
7	1		C
13	3		C
19	1		A
22	2		B
27	1		A
56	1		B

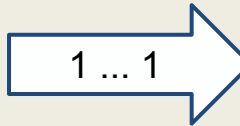
PK	Название
A	Математика
B	Физика
C	Английский

Таблицы
деталей

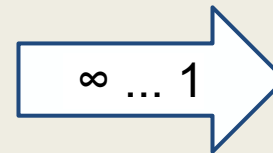
PK	ФИО
1	Кабанов
2	Запорожец
3	Шуст

Типы отношений между таблицами

1. Один к одному



2. Один ко многим



3. Много ко многим

Типы данных SQL

Текстовые

- CHARACTER(*длина*) - последовательность символов
длина описывает неизменное количество символов строки
- CHAR(*длина*) - сокращенная форма
- VARCHAR(*длина*) - отступление от стандарта, строка переменной длины, но не более указанной

Все текстовые данные заключаются в одинарные кавычки.

Пример:

```
'QA Light!!! Курс введение в SQL.' as CHARACTER(32)
'QA Light!!!' as CHARACTER(32)
'QA Light!!!' as VARCHAR(32)
'QA Light!!! Общение с БД на ее языке!' as CHAR(37)
'QA Light!!! Общение с БД на ее я' as VARCHAR(32)
```

Типы данных SQL

Числовые

- INTEGER (INT) - целочисленный, диапазон значений от -2^{31} до $+2^{31}$
- SMALLINT - целочисленный, диапазон значений от -2^{15} до $+2^{15}$
- DECIMAL(*точность*[,*масштаб*]) - десятичное число с фиксированной точкой, где *точность* - общее количество значащих цифр, *масштаб* - максимальное количество цифр после запятой
- NUMERIC(*точность*[,*масштаб*])
- FLOAT(*точность*) - число с плавающей точкой и указанной минимальной точностью
- REAL - число с плавающей точкой, зависящее от конкретной реализации SQL
- DOUBLE PRECISION - число такое же как REAL, но с два раза превышающей точностью

Типы данных SQL

Дата-время

Чаще всего записывается как DATE или DATETIME.
Также отдельный тип для времени TIME.

Значения записываются, как строковые, в одинарных кавычках в соответствии с форматом, определенным в системе или клиентском приложении:

- '2012-sep-14'
- '2012.09.14'
- '2012/09/14'
- '2012/09/14 12:05:44.123'
- '2012/09/14 00:00:00.000'

Типы данных SQL

Неопределенные или пропущенные данные определяются как *NULL*

Такое поле нельзя проверить на равенство

~~*X = NULL*~~

в таких случаях необходимо использовать

X IS NULL

X IS NOT NULL

В агрегирующих функциях значение такого поля игнорируется

В случае с *NULL* условные операторы, результатом которых может быть *true/false*, расширяются еще одним значением *unknown*

Свойства реляционной модели данных

Свойства таблиц реляционной базы данных:

- в таблицах не должно быть записей-дубликатов, следовательно
 - в таблице всегда имеется поле или их набор, позволяющих идентифицировать запись, иначе говоря, в таблице всегда есть первичный ключ
- записи таблицы не упорядочены, следовательно, в таблице не существует:
 - иного способа адресации записей, кроме адресации по ключу
 - первой записи, последней, предыдущей, следующей и т.д.
- столбцы таблицы также не упорядочены, следовательно:
 - единственным способом их адресации в запросах является использование наименования поля
 - их порядок, задается в момент создания таблицы

Индексы

Номера домов тоже зачем-то нумеруют!

Индексы непосредственно связаны с конкретными полями таблицы.

Их может быть сколько угодно и их различные комбинации.

Пример, как найти книгу в библиотеке:

1. тематический зал
2. первая буква фамилии автора
3. толщина?... нет, название! книги



Однако, не стоит сильно увлекаться индексацией, нужно помнить, что любое изменение информации в БД, приведет к перестройке индексов, что может тормозить работу базы

Что SQL может сделать с Базой Данных?

- создать новую таблицу CREATE TABLE
- добавить в таблицу новые записи INSERT
- изменить записи UPDATE
- удалить записи DELETE
- произвести выборку записей из одной или нескольких таблиц (по определенному условию) SELECT
- изменить структуру таблиц ALTER TABLE
- удалить таблицу DROP TABLE
- настроить ограничения для таблиц и их полей ALTER

Операторы SQL

Операторы определения данных (Data Definition Language, **DDL**)

- CREATE создает объект БД (саму базу, таблицу, представление, пользователя и т. д.)
- ALTER изменяет объект
- DROP удаляет объект

Операторы SQL

Операторы манипуляции данными (Data Manipulation Language, **DML**)

- SELECT выбирает данные, в соответствии с заданными критериями
- INSERT добавляет новые данные
- UPDATE изменяет существующие данные
- DELETE удаляет данные

Операторы SQL

Операторы определения доступа к данным (Data Control Language, **DCL**)

- GRANT предоставляет пользователю (группе) разрешения на определенные операции с объектом
- REVOKE отзывает ранее выданные разрешения
- DENY задает запрет, имеющий приоритет над разрешением

Операторы SQL

Операторы управления транзакциями (Transaction Control Language, **TCL**)

- COMMIT применяет транзакцию.
- ROLLBACK откатывает все изменения, сделанные в контексте текущей транзакции.