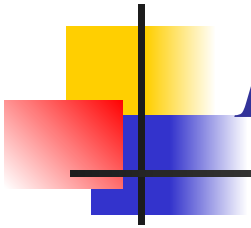
Министерство образования и науки Республики Казахстан
Семипалатинский Государственный Педагогический Институт
Физико-математический факультет
Кафедра информатики и информационных систем

Дисциплина : «Базы данных и информационных систем»
Специальность: 5В001110 – информатика

Тема: Язык SQL

Преподаватель: Батырока К.А.
Выпролинили: Шаяхметова А. Нурланкызы А.



План:

1. Введение

2. SQL

3. Функции языка SQL

4. Стандарты SQL

5. SQL в компьютерной сети

6. Элементы языка SQL

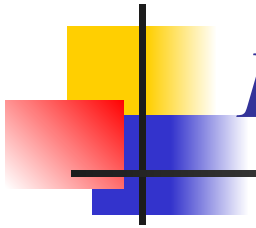
7. Ключевые слова. Имена. Константы

8. Типы данных. Выражения.

9. Встроенные функции

10. Чтение данных.

Оператор Select. Предложение Select



Введение

Большинство современных СУБД построено на реляционной модели данных. Для получения информации из отношений (таблиц) базы данных в качестве языка манипулирования данными в теоретическом плане используется язык SQL

***SQL** – структурированный язык запросов, предназначенный для работы с БД реляционного типа.*

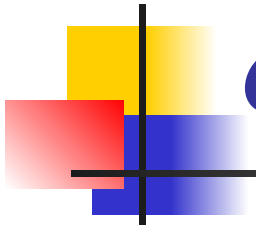
***SQL** является интерактивным языком запросов, который обеспечивает пользователю быстрый доступ к данным.*

***SQL** является также языком программирования баз данных. Программисты могут вставить SQL-запросы в свои программы, чтобы получить доступ к базам данных*

***SQL** – язык распределения базы данных, служит для распределения данных взаимодействующих систем, для распределенной обработки баз данных*

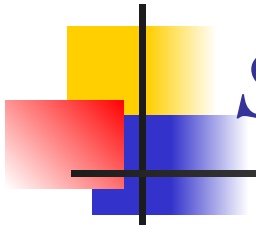
Функции языка SQL:

- ***Организация данных*** *– создание и изменение структуры баз данных*
- ***Чтение данных***
- ***Обработка данных*** *– удаление, добавление и корректировка данных*
- ***Управление доступа к данным*** *– предоставление привилегий (ограничение возможностей) пользователю для чтения и изменения данных*
- ***Совместное использование данных-*** *координация общего пользования данных многими пользователями*
- ***Целостность данных*** *– защита данных от разрушения при сбое системы или других обстоятельствах*



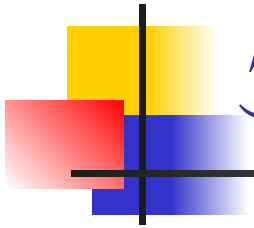
Стандарты SQL

Разработка SQL началась в 1982 году Американским институтом национальных стандартов ANSI (American National Standards Institute). В 1986 SQL был официально утвержден как стандарт ANSI, ф в 1987 году – в качестве стандарта ISO (International Standards Organization) – международной организации по стандартизации.



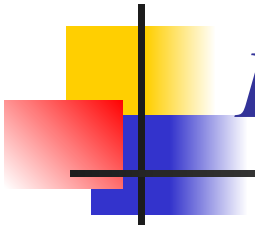
SQL в компьютерной сети

Сервер базы данных выполняет SQL – запрос и возвращает пользователю только ту информацию из базы данных, которая соответствует этому SQL - запросу



Элементы языка SQL

- *Ключевые слова*
- *Имена*
- *Константы*
- *Типы данных*
- *Встроенные функции*
- *выражения*

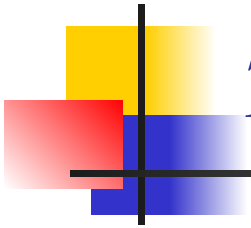


Ключевые слова. Имена. Константы

Ключевые слова *– это фиксированный набор английских слов, которые определяют тип запроса и необходимую информацию для выполнения этого запроса*

Имена *используются для обозначения (присвоения имени) таблиц, столбцов в таблице, а также владельцев таблиц (баз данных)*

Константы *служат для явного указания величин – чисел, строк, дату и время – в командах SQL*



Типы данных. Выражения.

Типы данных служат для представления информации в базах данных. В SQL определен набор типов данных (char, varchar , integer, smallint…)

Выражения в SQL представляют собой имена, константы, встроенные функции, связанные между собой знаками арифметических операций. В сложных выражениях для изменения порядка вычислений применяются круглые скобки.

Встроенные функции *в основном предназначены для преобразования типов данных и для обработки строк*

Некоторые встроенные функции

Current_date()- *возвращает текущую дату*

Current_time*(точность) - возвращает текущее время*

Char_length*(строка) – возвращает длину строки*

Extract *– возвращает значение части day, hour и т.д. даты*

Lower*(строка) - возвращает строку, преобразованную к нижнему регистру*

Upper *(строка) - возвращает строку, преобразованную к верхнему регистру*

Month*(дата) – возвращает значение месяца из указанной даты в виде целого числа*

Year*(дата) – возвращает значение года из указанной даты в виде целого числа*

Чтение данных.
Оператор Select. Предложение Select

Оператор Select *читает данные из базы данных и возвращает их в виде таблицы результата запроса*

Предложение Select*, с которого начинается оператор Select, содержит элементы данных, которые будут возвращены в виде результирующей таблицы. Указанные в предложении Select элементы данных будут составлять столбцы возвращаемой таблицы. В качестве возвращаемых столбцов могут быть указаны:*

1) *Имя столбца некоторой таблицы базы данных;*
2) *Константа, которая будет содержаться в соответствующем столбце возвращаемой таблицы*
3) *Выражение, которое будет вычисляться для каждой строки возвращаемой таблицы, и помещаться в соответствующем столбце этой таблицы*

Предложение FROM

Предложение FROM начинается с ключевого слова FROM, за которым следует в простом случае список спецификаций таблиц, разделенных запятыми. В общем случае за ключевым словом FROM указывается операция соединения (JOIN) исходных таблиц.

Спецификатор таблицы определяет таблицу, из которой запрашиваются исходные данные для формирования возвращаемой таблицы. Спецификатор таблицы представляет собой либо имя исходной таблицы, либо имя исходной таблицы вместе с псевдонимом, указываемым после имени через пробел. Синтаксическая диаграмма спецификатора таблицы следующая (рис 1).

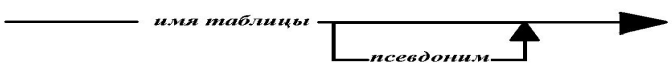


Рис 1 Синтаксическая диаграмма спецификатора таблицы.

Псевдоним может быть может быть использован в следующих предложениях оператора SELECT вместо имени.

Примеры

1. *SELECT Sname, City*
   FROM salespeople
2. *SELECT Sname, salespeople. City, Customers.**
   From Salespeople S, Customers

Предложение FROM

Операция соединения *определена в стандарте SQL2 и имеет следующую синтаксическую диаграмму (рис 2).*

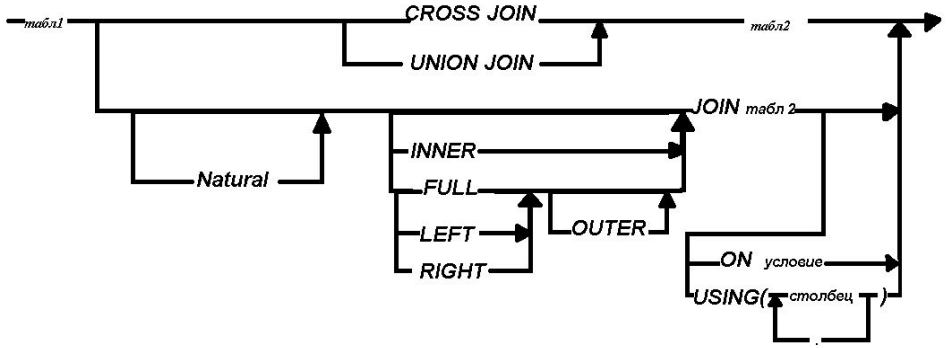


Рис 2 Синтаксическая диаграмма операции соединения

*Операция **CROSS JOIN** дает декартово произведение двух таблиц, **UNION JOIN** дает сложение двух таблиц. Ключевое слово **INNER** задает внутреннее объединение (по умолчанию), **OUTER** – внешнее объединение (левое, правое и полное) ; **NATURAL** – естественное объединение (равенство значений в одноименных столбцах, при этом предложения **ON** и **USING** не обязательны).*

Предложение WHERE

Предложение WHERE осуществляется отбор нужных строк из таблицы, получаемой в предложении FROM. Отбор строк производится в соответствии с условием поиска, которое указывается в предложении за ключевым словом WHERE. Условие поиска представляет собой выражение, которое вычисляется для каждой строки возвращаемой таблицы. При вычислении выражения данные соответствующей строки берутся из столбцов, указанных в выражении. Для каждой проверяемой строки условие поиска может иметь одно из трех значений – TRUE, FALSE и NULL. Отбираются только те строки, для которой условие поиска имеют значение TRUE. Условие поиска имеет следующую синтаксическую диаграмму (рис 3).

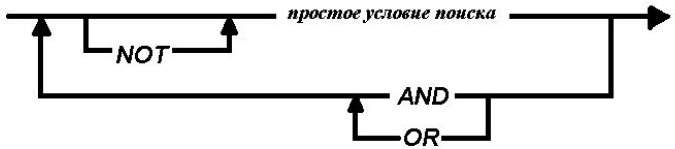


Рис 3 Синтаксическая диаграмма условии поиска

Из диаграммы видно, что условие поиска состоит из простых условий объединенных с помощью логических операций NOT, AND и OR.

Предложение WHERE

В следующей таблице приведены результаты логических операций над значениями TRUE, FALSE и NULL.

	OR			AND			NOT
	TRUE	FALSE	NULL	TRUE	FALSE	NULL	
TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	NULL	FALSE
FALSE		FALSE	NULL		FALSE	FALSE	TRUE
NULL			NULL			NULL	NULL

В SQL существует пять простых условий поиска:

- ✔*Сравнение;*
- ✔*Проверка на принадлежность диапазону;*
- ✔*Проверка на принадлежность множеству;*
- ✔*Проверка на соответствие шаблону;*
- ✔*Проверка на равенство значению NULL.*

Предложение WHERE

Сравнение. *При сравнении вычисляются и сравниваются значения двух выражений. Существует шесть операций сравнения, которые показаны на следующей синтаксической диаграмме (рис 4)*

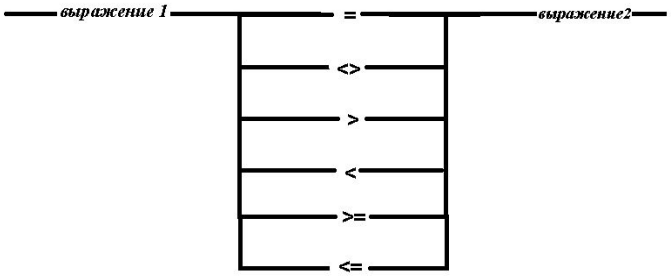


Рис 4 Синтаксическая диаграмма операций сравнения

Результатом сравнения будет значения TRUE , если значения выражений сравниваются и FALSE – если не сравниваются. В случае, если значение хотя бы одного выражения будет NULL, то результат сравнения будет NULL.

Предложение WHERE

Проверка на принадлежность диапазону. *Данная проверка определяет находится или нет значение проверяемого выражения в диапазоне, указанном за ключевым словом BETWEEN, как показано на следующей диаграмме (рис 5)*

Рис 5 Синтаксическая диаграмма проверки на принадлежность диапазону

Проверка на принадлежность диапазону эквивалента следующему логическому выражению (A>=B) AND (A<=C), где A – проверяемое выражение, а B и C – соответственно нижнее и верхнее границы диапазона.

Предложение WHERE

Проверка на принадлежность множеству. *Данной проверке соответствует следующая синтаксическая диаграмма (рис 6)*

Рис 6 Синтаксическая диаграмма проверки на принадлежность множеству

Проверка на принадлежность множеству определяет, равняется или нет значение проверяемого выражения одному из констант, указанных за ключевым словом IN. Проверка на принадлежность множеству эквивалентна следующему условию поиска сравнение

(A=C1) OR (A=C2)..(A=Cn) *, где A – проверяемое выражение, а Ci – константы.*

Предложение WHERE

Проверка на соответствие шаблону. *Данная проверка выполняется над текстовыми данными. Данной проверке соответствует следующая синтаксическая диаграмма (рис 7)*

Рис 7 Синтаксическая диаграмма проверки на соответствие шаблону

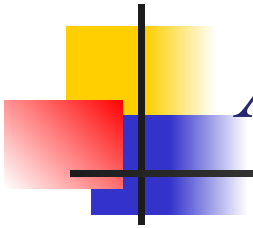
С помощью этой проверки отбираются те строки, в которых данные, соответствующие заданному столбцу, отвечают некоторому шаблону, следующему в предложении за ключевым словом LIKE.

Предложение WHERE

Проверка на равенство значению NULL. *Для проверки значения элемента столбца на равенство NULL в SQL предусмотрена специальная проверка, называемая проверкой на равенство NULL. Синтаксическая диаграмма этой проверки следующая (рис 8)*

Рис 8 Синтаксическая диаграмма проверки на равенство значению NULL

В соответствии с этой проверкой отбираются те строки, т.е. проверка имеет значение TRUE, у которых данные в указанном столбце равны NULL.



Агрегатные функции

В SQL предусмотрены специальные функции, которые называются агрегатными (статистическими) функциями. В SQL имеются следующие агрегатные функции:

SUM() – вычисляют сумму значений, содержащих в аргументе;

AVG() – вычисляет среднее значение;

MIN(), MAX() – находит минимальное, максимальное значения;

COUNT() – подсчитывает в столбце количество не NULL-значений;

COUNT(*) – подсчитывает количество строк в запрашиваемой таблице.

Выражение должно содержать имена столбцов и определяет столбец, элементы которого вычисляются с помощью соответствующих элементов этих столбцов.

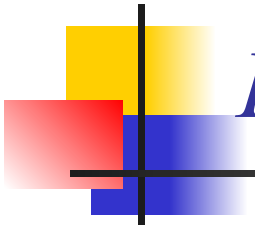
Так как агрегатная функция возвращает одно значение, то она не может быть аргументом другой агрегатной функции. Агрегатные функции используются в предложениях SELECT и HAVING.

Предложение GROUP BY

Предложение GROUP BY позволяет запрашивать данные, которые являются итоговыми для отдельных групп строк таблицы получаемой в предложении FROM. Если в запросе имеется предложение GROUP BY , то строки запрашиваемой таблицы разбиваются на группы.

На запросы с применением предложения GROUP BY налагаются следующие ограничения – возвращаемыми столбцами, указанными в предложении SELECT, могут быть:

- *Константа ;*
- *Агрегатная функция;*
- *Столбце группировки;*
- *Выражение, включающее в себе перечисленные выше элементы.*



Предложение HAVING

Для того, чтобы из групп строк, получаемых после предложения GROUP BY, выбрать требуемые группы, используется предложение HAVING. В это предложении за ключевым словом HAVING следует условие поиска групп, аналогичное условию поиска в предложении WHERE.

Условию поиска применяется к группам строк, получаемых после предложения GROUP BY и поэтому на элементы, входящие в условие поиска, налагаются те же ограничения, какие были перечислены для возвращаемых столбцов при рассмотрении предложения GROUP BY. Таким образом, в условие поиска могут входить константы , агрегатные функции, столбцы группировки и выражение из них.

Предложение HAVING в основном используется вместе с предложением GROUP BY. Но язык SQL допускает и отдельное применение предложения HAVING. В этом случае результат запроса рассматривается как одна группа и предложение HAVING выполняет те же функции что и предложение WHERE.