

# DML

## О г л а в л е н и е

### DML

Г р у п п ы к о м а н д SQL

О п е р а т о р SELECT

С т р у к т у р а о п е р а т о р а SELECT

У с л о в и я в ы п о л н е н и я о п е р а т о р а SELECT

JOINS

П о д з а п р о с ы (subqueries)

Set Operators

Г р у п п и р о в к а

А г р е г а т н ы е ф у н к ц и и

У с л о в и я ANY, ALL, SOME

Л о г и ч е с к и й о п е р а т о р у с л о в и я IN

Л о г и ч е с к и й о п е р а т о р у с л о в и я LIKE

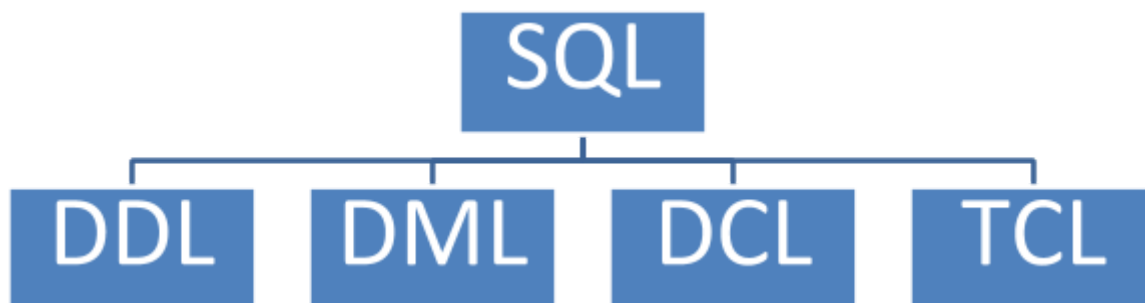
Ф у н к ц и я REGEXP\_LIKE

Л о г и ч е с к и й о п е р а т о р у с л о в и я EXISTS

Л о г и ч е с к и й о п е р а т о р у с л о в и я BETWEEN

В ы р а ж е н и е CASE

## Г р у п п ы к о м а н д SQL



DDL – Data Definition Language – команды ddl позволяют изменять структуру баз данных (create, alter, drop)

DML – Data Manipulation Language – команды dml позволяют извлекать или изменять содержимое баз данных (то есть сами данные). К DML командам относят, в частности, следующие команды: **Select, Insert, Update, Delete, Merge**, Explain Plan, Lock Table

DCL – Data Control Language – команды dcl позволяют разграничивать права пользователя (grant, deny, revoke)

TCL – Transaction Control Language – команды tcl позволяют управлять транзакцией (set transaction, commit, rollback)

## Оператор SELECT

Select – это единственный sql-оператор, который используется при запросе данных.

Минимально возможная структура запроса выглядит следующим образом:

**Select** <some column>

**From** <some table>

Ключевые слова **select** и **from** присутствуют в любом запросе

## Структура оператора SELECT

**Select** [All | Distinct ] <[Table. | TableAlias.]<\* | Column> [ColumnAlias], ....

**From** Table1 [Table1Alias], Table2 [Table2Alias]

[**Where** <conditions>]

[**Order by** <Column1 | Column1Alias>, ... [Asc | Desc]]

## Условия выполнения оператора SELECT

1. Для выполнения запроса select к таблице или представлению, пользователю необходимо иметь привилегию select на эту таблицу или представление
2. Пользователю-владельцу схемы, в которой лежит представление, необходимо иметь привилегии select на все таблицы из этого представления

Существует, также, системная привилегия SELECT ANY TABLE, которая дает пользователю право на извлечение данных из любой таблицы или представления (view).

## JOINS

Соединения таблиц появляются при запросе к более чем одной таблице. При этом при соединении трех и более таблиц, Oracle сначала соединяет две из них, потом результат соединения соединяет со следующей таблицей и так далее.

Условия связи таблиц задаются или в секции FROM, или в секции WHERE

- Inner join (table1.id = table2.id\_table1) – внутренние соединения – возвращаются только те записи, которые удовлетворяют условию соединения
- Outer join – внешнее соединение
  - Left outer join (table1.id = table2.id\_table1(+)) – левое внешнее соединение – соединяет те записи из обеих таблиц, которые удовлетворяют условию соединения, а также возвращает все записи из левой таблицы, что не удовлетворяют условию соединения.. Пустые поля заполняет значениями null.
  - Right outer join (table1.id(+) = table2.id\_table1) – правое внешнее соединение – соединяет те записи из обеих таблиц, которые удовлетворяют условию соединения, а также возвращает все записи из правой таблицы, что не удовлетворяют условию соединения. Пустые поля заполняет значениями null.
  - Full outer join – полное внешнее соединение – соединяет те записи из обеих таблиц, которые удовлетворяют условию соединения, а также возвращает все записи, что не удовлетворяют условию соединения. Пустые поля заполняет значениями null.
- Cartesian product – декартово произведение – эквивалентно перемножению множеств, а именно, возвращает все возможные соединения (комбинации) записей из обеих таблиц
- Self join – иерархический запрос – соединение таблицы с самой собой
- Partitioned outer join – выделяет часть таблицы (поля, определенные в partitioned clause) в отдельную партицию, которую соединяет с другой таблицей как outer join (или left outer join, или right outer join)
- Antijoin – соединение таблиц с помощью оператора условия IN
- Semijoin – соединение таблиц с помощью оператора условия EXISTS

## Подзапросы (subqueries)

Подзапрос (subquery) – это select-запрос, вложенный в другой sql-запрос.

- Подзапрос в секции select
  - Видит таблицы из внешнего from
  - Может выбирать не более одной записи и одного значения
- Подзапрос в секции from
  - Воспринимается в запросе, как view.

- Не видит таблицы из внешнего from
- Подзапрос в секции where
  - Видит таблицы из внешнего from
  - Для операции “=” должен возвращать только одну строку и одно значение
  - Для операции in должен возвращать только одно значение (одно поле), но произвольное количество записей (или не возвращать ничего)

При выполнении select оператора с подзапросами, Oracle самостоятельно может изменить структуру запроса (на идентичную по бизнес-смыслу) и избавиться от подзапроса. При необходимости это можно предотвратить, указав хинт /\*+ NO\_QUERY\_TRANSFORMATION \*/ или добавив в вывод подзапроса псевдостолбец rownum.

## Set Operators

- ✓ Union All – возвращает объединение запросов
- ✓ Union – после объединения запросов делает distinct
- ✓ Intersect – возвращает пересечение запросов
- ✓ Minus – вычитает результаты одного запроса из результатов другого

### Restrictions:

- Запросы должны возвращать одинаковое количество полей, с одинаковыми типами данных в одинаковом порядке
- Order by один на весь запрос и следует в конце

## Группировка

- ✓ Group by **clause** – самый часто встречающийся вид группировки – группирует по перечисленным полям/выражениям
- ✓ Group by **cube** – используется для построения отчетов – группирует по всем комбинациям указанных полей/выражений
- ✓ Group by **rollup** – можно считать «урезанной» версией group by cube – группирует по указанным выражениям по следующему алгоритму: сначала группирует по всем выражениям, потом по всем выражениям минус одно (самое правое), после по всем выражениям минус два (самых правых) и так далее
- ✓ Group by **grouping sets** – самый гибкий аналог group by cube – группирует по указанным наборам выражений

## Restrictions

- Кроме полей, указанных в условии группировки, в select-листе можно выводить только значения агрегатных функций.
- Если в группировке указана колонка с объектным типом, запрос не может быть распараллелен
- В условии группировки нельзя использовать поля/выражения с типами данных lob, varray, nested tables

## Агрегатные функции

Возвращают агрегированное значение по набору данных.

- ✓ Sum – возвращает сумму по группе
- ✓ Min – возвращает минимальное значение по группе
- ✓ Max – возвращает максимальное значение по группе
- ✓ Count – возвращает количество записей в группе
- ✓ Avg - возвращает среднее арифметическое значение по группе
- ✓ All | Distinct – при вычислении значения агрегатной функции, с distinct не учитываются дубликаты.

**Restrictions:**

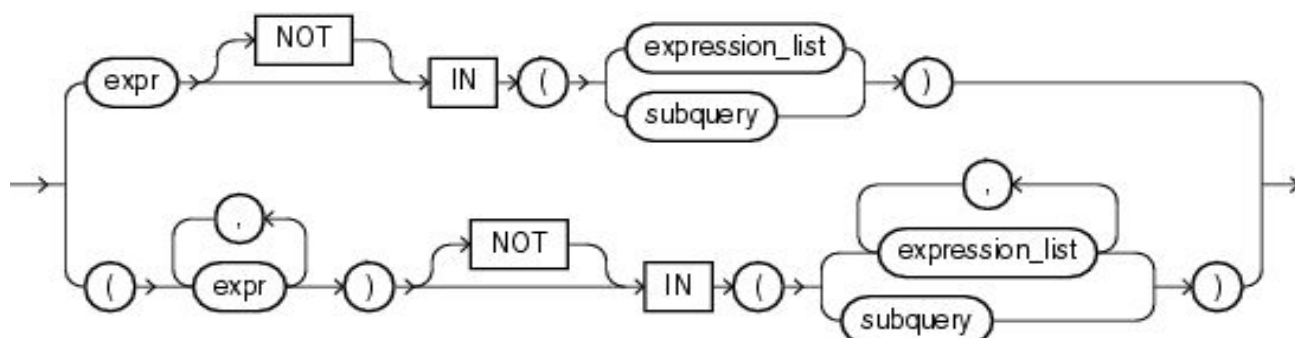
- Игнорируют значение null
- Могут использоваться только в секциях select, having и order by

## Условия ANY, ALL, SOME

Используются с операторами сравнения: =, !=, ^, <, >, <=, >=

- ✓ ANY SOME – возвращает true, если операция истинна для одного или больше значений из `expression_list`, иначе false
- ✓ ALL – возвращает true, если операция истинна для всех значений из `expression_list`, иначе false

## Логический оператор условия IN



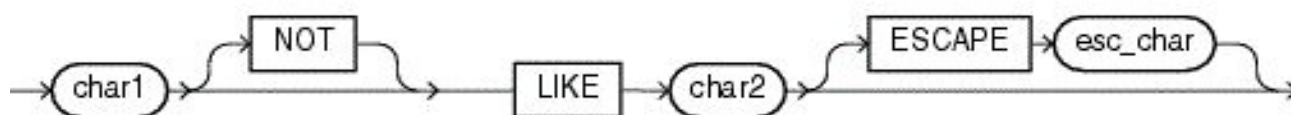
IN – эквивалентен =ANY

NOT IN – эквивалентен !=ALL

#### Restrictions:

- Можно указать до 1000 значений/выражений в expression\_list
- Эти выражения не всегда выполняются в том порядке, в каком они указаны в expression\_list
- Если в expression\_list (или результате subquery) оператора NOT IN есть значение null, результат оператора всегда будет false

## Логический оператор условия LIKE



Позволяет проверить строку на соответствие шаблону

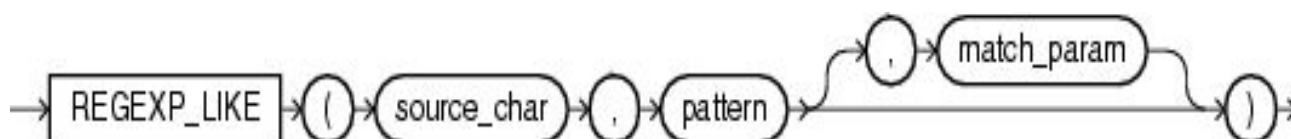
Знак подчеркивания «\_» соответствует любому одному символу

Знак процента «%» соответствует любой строке, в том числе пустой

#### Restrictions:

- Выражения char1, char2 и esc\_char могут иметь один из следующих типов данных: CHAR, VARCHAR2, NCHAR ИЛИ NVARCHAR2. Если типы данных у выражений отличаются, Oracle неявно приводит все выражения к типу данных char1
- Зависимость от регистра прописывается в настройках БД
- При поиске по индексируемому полю, Oracle может использовать индексы и улучшить производительность запроса только в том случае, если первый символ шаблона не % и \_

## Функция REGEXP\_LIKE



**source\_char** – строка (как правило, строковое поле таблицы), которую сравниваем с шаблоном

**pattern** – регулярное выражение (шаблон), максимальный размер 512 байт

**match\_param** – параметры сравнения (i – регистронезависимое сравнение, s – регистрозависимое и некоторые другие)

Синтаксис регулярного выражения (pattern) [TUT](#)

## Логический оператор условия EXISTS



Возвращает true, если результат подзапроса содержит хотя бы одну запись. Иначе false

## Логический оператор условия BETWEEN

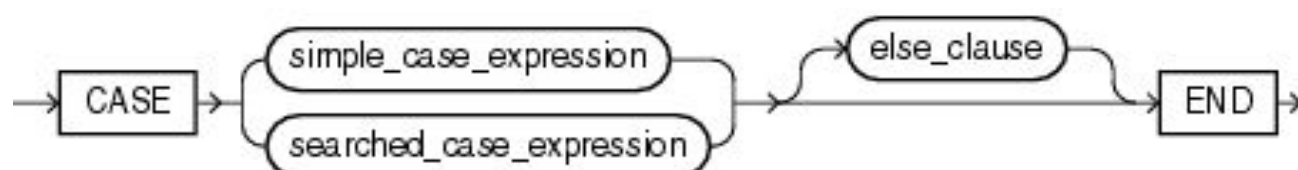


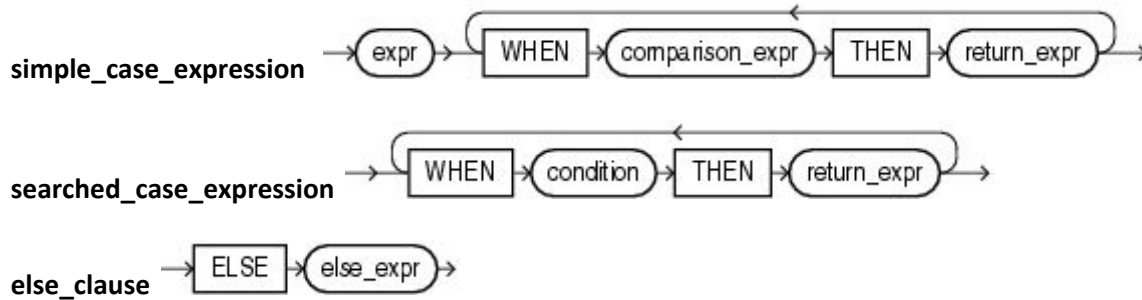
Возвращает true, если значение выражения expr1 входит в интервал expr2 и expr3 (включая границы). Т.е. эквивалентно такому условию:  $\text{expr2} \leq \text{expr1} \text{ AND } \text{expr1} \leq \text{expr3}$

### Restrictions:

- Все три выражения должны иметь числовой, строковый или datetime тип
- Если результаты выражений expr1, expr2 и expr3 имеют разные типы данных, Oracle неявно производит преобразование к единому типу данных. В случае, если преобразование невозможно, генерируется исключение
- $\text{expr1 between expr2 and expr3}$ , где  $\text{expr3} < \text{expr2}$ , возвращает null при  $\text{expr1}$  равным null, и возвращает false при  $\text{expr1}$  не равным null
- $\text{expr1 not between expr2 and expr3}$ , где  $\text{expr3} < \text{expr2}$  и  $\text{expr1}$  не null, возвращает true

## Выражение CASE





П о з в о л я е т и с п о л ь з о в а т ь if... then...else л о г и к у в sql-в ы р а ж е н и я х

#### Restrictions:

- Д л я с л у ч а я с searched\_case\_expression Oracle п р о в е р я е т в с е у с л о в и я с л е в а н а п р а в о д о п е р в о г о true, и в о з в р а щ а е т с о о т в е т с т в у ю щ и й е м у return\_expr. Е с л и т а к о г о у с л о в и я н е н а х о д и т с я , в о з в р а щ а е т с я else\_expr. Е с л и с е к ц и я ELSE о т с у т с т в у е т , в о з в р а щ а е т с я null.
- Д о п у с к а е т н е б о л e e 65535 а р г у м е н т о в
- Д л я expr, comparison\_expr и return\_expr д о п у с к а е т с л е д у ю щ и е т и п ы д а н н ы х : CHAR, VARCHAR2, NCHAR и л и NVARCHAR2, NUMBER,BINARY\_FLOAT и л и BINARY\_DOUBLE
- Т и п ы д а н н ы х expr, comparison\_expr д о л ж н ы с о в п а д а т ь . В с л у ч а е , е с л и т и п ы д а н н ы х н е с о в п а д а ю т , Oracle п ы т а е т с я н е я в н о п р и в е с т и и х к е д и н о м у т и п у