Занятие 10

База данных. MySQL

Основы MySQL

SQL - structured query language -
«язык структурированных запросов»

SQL - это стандартный язык доступа
и управления базами данных (БД)

Структурированный язык запросов - это стандартный язык доступа к БД, таким как SQL Server, Oracle, MySQL, Sybase и Access.

Программная инструкция для получения данных из БД, называется запросом к базе данных.

Основы MySQL

Сервер приложений предоставляет возможность использовать такие ресурсы сервера, как базы данных. Например, динамическая страница содержит запрос к БД. Выполнив этот запрос, сервер получит данные из базы, и поместит их в HTML-код страницы

Если хранить данные в БД, то оформление сайта и конкретные данные будут разделены. Вместо того, чтобы создавать все страницы в виде отдельных HTML-файлов, пишутся только шаблоны. Таким образом можно обновить информацию в одном источнике и продублировать это изменение на всем веб-сайте без редактирования каждой страницы вручную.

Основы MySQL

Сервер приложений не может непосредственно получить данные из БД, поскольку они используют специфические форматы хранения данных. Попытка получить их похожа на попытку открыть документ Microsoft Word с помощью текстового редактора Notepad. Поэтому для подключения к БД сервер приложений использует посредника - драйвер базы данных.

Драйвер базы данных - программный модуль, с помощью которого устанавливается взаимодействие между сервером приложений и базой данных

Основы MySQL

После того, как драйвер установит соединение, выполняется запрос к базе, в результате чего формируется набор записей.

Набор записей – данные из одной или нескольких таблиц БД.

Набор записей возвращается серверу приложений, и он использует их для формирования страницы.

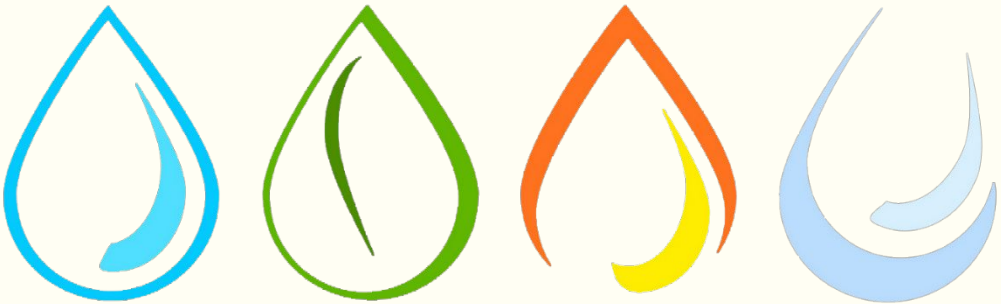
Основные MySQL запросы

SELECT - извлекает данные из таблицы БД

UPDATE - обновляет данные в таблице БД

DELETE - уничтожает данные в таблице БД

INSERT INTO- вставляет новые данные в таблицу БД



STORMNET

Пример запроса

Persons

LastName	Name	Adress	City
Polyakov	Denis	Lyibyanka, 25	Moscow
Ivanov	Mihail	Sadovaya, 17	Kazan'
Popandopulo	Ermak	Hutorskay, 4	Kiev

SELECT LastName FROM Persons;

LastName
Polyakov
Ivanov
Popandopulo

STORMNET

Основные команды. Select

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

SELECT * FROM Customers;

SELECT CustomerName, City
FROM Customers;

STORMNET

Выбор всех столбцов [SELECT *]

```
SELECT * FROM Customers;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK

Основные команды [conditions]

Operator	Description
=	Equal
<> / !=	Not equal
> / >=	Grater / grater or equal
< / <=	Less / less or equal
Like	Search for a pattern (‘%’ and ‘_’)
in ()	To specify multiply values
Not	Inverts the value for Like, In
And	For mandatory creation union (1 and 2)
Or	For optional creation union (1 or 2)
Where	Specify certain value
Between	Selects values within a range (numbers, text, or dates)
Is	For NULL values searching
Limit	Reduce the number of results

STORMNET

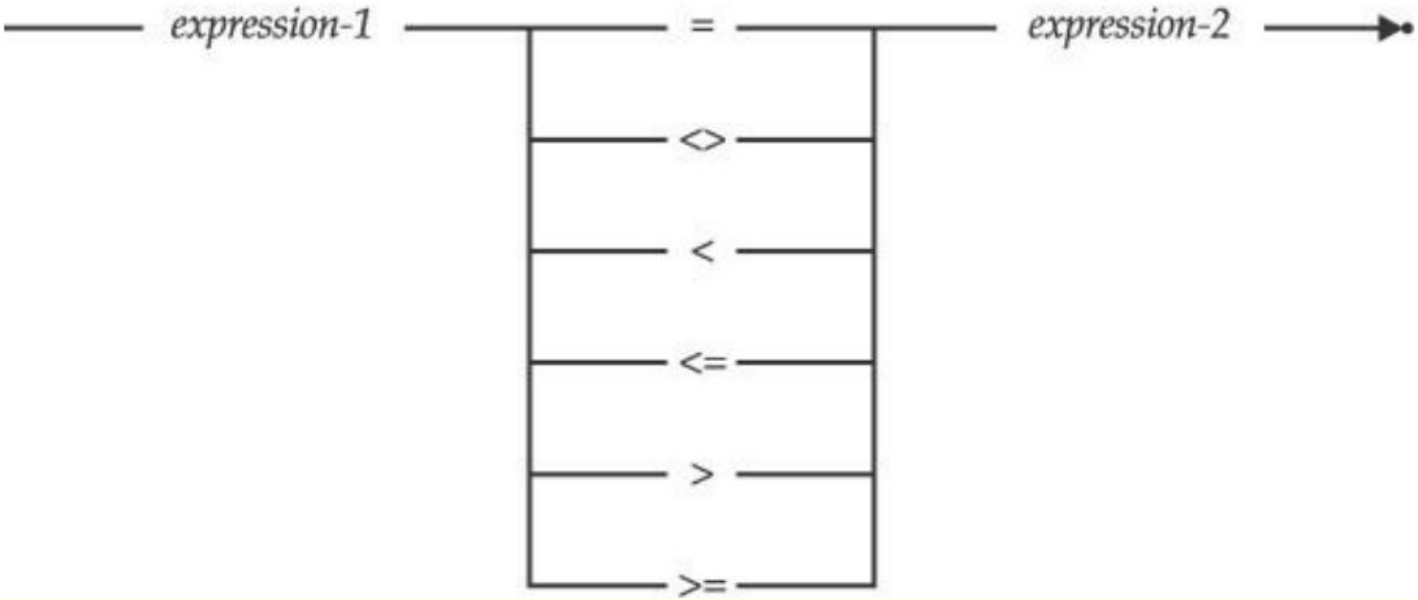
Подстановочные. Wildcards

Wildcard	Description
*	All
%	Replace any number of symbols (including 0)
_	Replace 1 symbol
[abc]	With any of symbol from the amount -> A or B or C

Условия поиска

- Сравнение.
- Диапазон.
- Роли уровня базы данных.
- Соответствие шаблону.
- Пустое значение.

Сравнительный Тест [=, <>, <, <=, >, >=]



Сравнительный Тест [=, <>, <, <=, >, >=]

```
SELECT * FROM Customers WHERE PostalCode = 44000 Limit 5;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
18	Du monde entier	Janine Labrune	67, rue des Cinquante Otages	Nantes	44000	France
26	France restauration	Carine Schmitt	54, rue Royale	Nantes	44000	France

STORMNET

Основные команды. Where

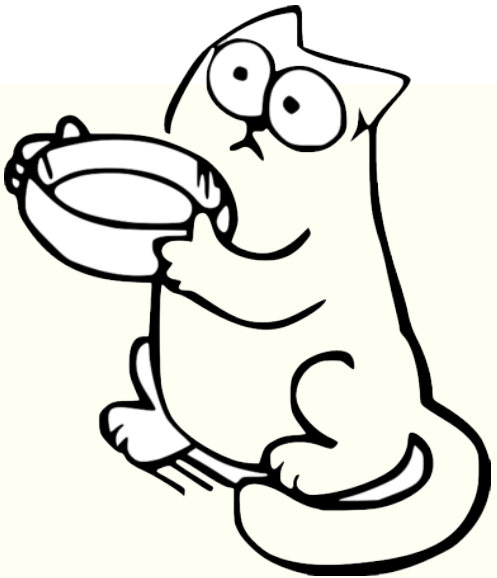
```
SELECT *
FROM Customers
WHERE Country = 'Mexico';
```

2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

STORMNET

Примеры запроса

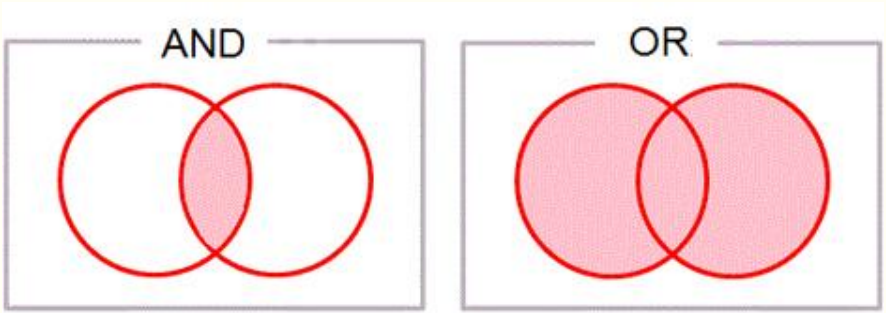
```
10
11    DELETE FROM user_db.users WHERE user_id = "1513253207972823";
12    DELETE FROM vs_db.users WHERE user_id = "1513253207972823"
13
14    SELECT * FROM casino_slots.xml_filter WHERE game_type_id = 340
```



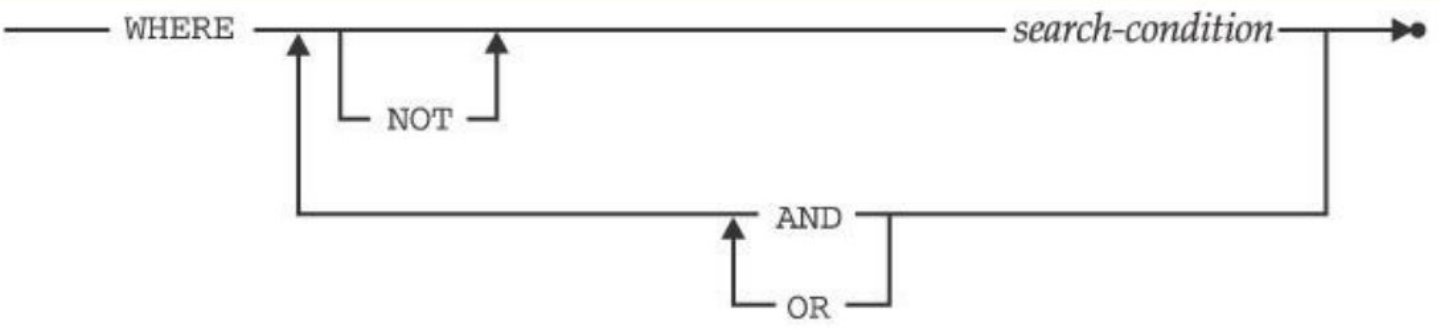
Основные команды. And & Or

SELECT * FROM Customers
WHERE Country = 'Germany‘ **AND** City = 'Berlin';

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany



Составные условия поиска [AND, OR, and NOT]



Составные условия поиска [AND, OR, and NOT]

```
SELECT * FROM Customers where Country = 'Sweden' or Country = 'Germany' limit 6;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
17	Drachenblut Delikatessend	Sven Ottlieb	Walserweg 21	Aachen	52066	Germany
24	Folk och fä HB	Maria Larsson	Åkergatan 24	Bräcke	S-844 67	Sweden
25	Frankenversand	Peter Franken	Berliner Platz 43	München	80805	Germany

```
SELECT * FROM Customers where Country = 'Sweden' and Country = 'Germany' limit 6;
```

No result.

STORMNET

Основные команды. And & Or

SELECT * FROM Customers
WHERE City = 'Berlin‘ **OR** City = ‘London';

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK

Основные команды. And & Or

```
SELECT * FROM Customers
WHERE Country = ‘Mexico'
AND (City = ‘Mexico' OR City = ‘Hanalulu');
```

2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México	05021	Mexico

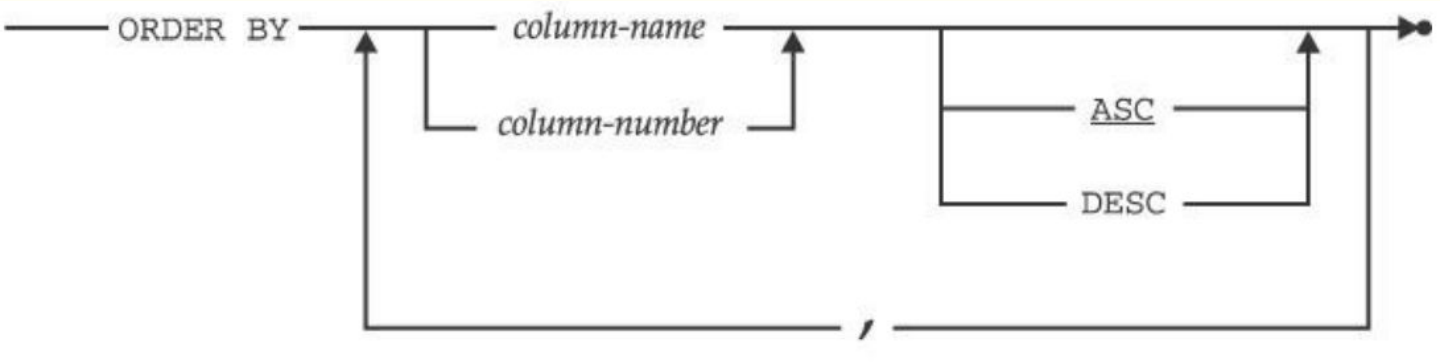
Основные команды. Order by

The ORDER BY keyword is used to sort the result-set by one or more columns, it's ascending by default.
If you want descending order use DESC keyword.

SELECT * FROM Customers
ORDER BY Country **DESC**;

4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
.....						
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany

Сортировка результатов запроса



Сортировка результатов запроса

```
SELECT Distinct Country FROM Customers order by Country limit 6;
```

Country
Argentina
Austria
Belgium
Brazil
Canada
Denmark

```
SELECT Distinct Country FROM Customers order by Country desc limit 5;
```

Country
Venezuela
USA
UK
Switzerland
Sweden

STORMNET

Основные команды. Insert into

INSERT INTO Customers
(CustomerName, ContactName, Address, City, PostalCode, Country)
VALUES
('Cardinal‘ ,'Tom Erichsen‘ ,'Skagen 21','Stavanger','4006','Norway');

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Cardinal	Tom Erichsen	Skagen 21	Stavanger	4006	Norway

STORMNET

Основные команды. Update

UPDATE Customers
SET ContactName = 'Alfred Schmidt', City = 'Hamburg'
WHERE CustomerName = 'Alfreds Futterkiste';

If you skip the WHERE clause all rows would be updated!

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfred Schmidt	Maria Anders	Obere Str. 57	Hamburg	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

STORMNET

Основные команды. Delete

DELETE FROM Customers
WHERE CustomerName = 'Alfreds Fkiste';

DELETE FROM Customers;
All rows will be deleted!

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Основные команды. Wildcards

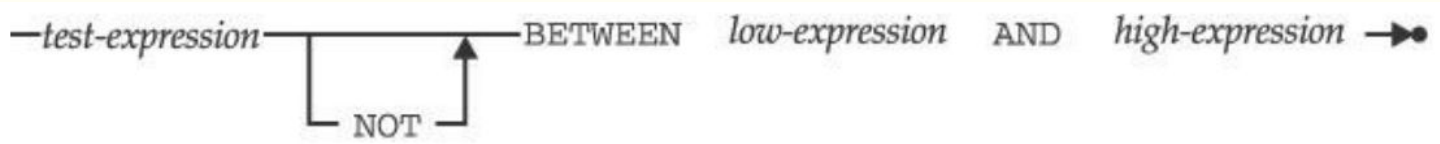
SELECT * FROM Customers
WHERE City LIKE 'ber%';

SELECT * FROM Customers
WHERE City LIKE '[bsp]%';

SELECT * FROM Customers
WHERE City LIKE '_erlin';

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany

Тестирование Диапазона [BETWEEN]



Основные команды. Between

SELECT * FROM Products
WHERE Price **BETWEEN** 10 AND 20;

SELECT * FROM Products
WHERE ProductName **BETWEEN** 'C' AND 'M';

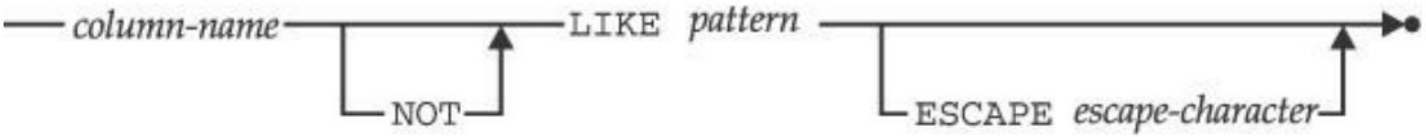
Тестирование Диапазона [BETWEEN]

```
SELECT * FROM Customers WHERE PostalCode BETWEEN 44000 and 78000 limit 5;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
17	Drachenblut Delikatessend	Sven Ottlieb	Walserweg 21	Aachen	52066	Germany
18	Du monde entier	Janine Labrune	67, rue des Cinquante Otages	Nantes	44000	France
23	Folies gourmandes	Martine Rancé	184, chaussée de Tournai	Lille	59000	France

STORMNET

Тестирование соответствия шаблону [LIKE]



Шаблоны знаков:

% - заменяет ноль или более символов;

_ - заменяет один символ;

[charlist] - устанавливает диапазоны символов для соответствия;

[^ charlist] или [! charlist] - устанавливает соответствие только символу, не указанному в скобках.

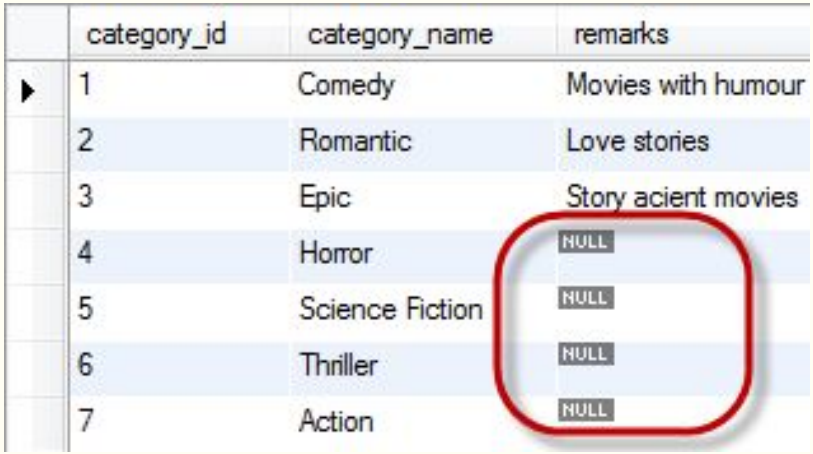
Тестирование соответствия шаблону [LIKE]

```
SELECT * FROM Customers WHERE PostalCode LIKE '440%' limit 5;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
18	Du monde entier	Janine Labrune	67, rue des Cinquante Otages	Nantes	44000	France
26	France restauration	Carine Schmitt	54, rue Royale	Nantes	44000	France
79	Toms Spezialitäten	Karin Josephs	Luisenstr. 48	Münster	44087	Germany

STORMNET

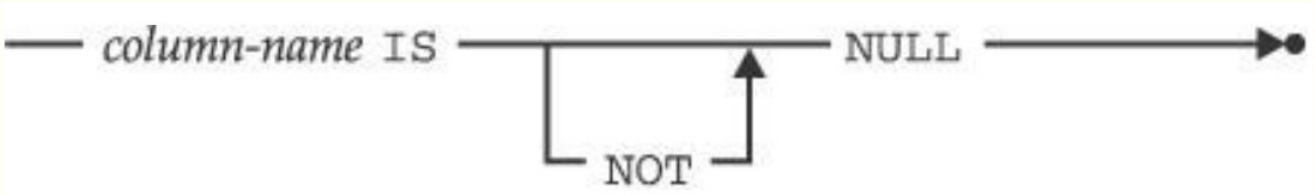
Недостающие данные [нулевые значения]



category_id	category_name	remarks
1	Comedy	Movies with humour
2	Romantic	Love stories
3	Epic	Story acient movies
4	Horror	NULL
5	Science Fiction	NULL
6	Thriller	NULL
7	Action	NULL

STORMNET

Тестирование нулевого значения



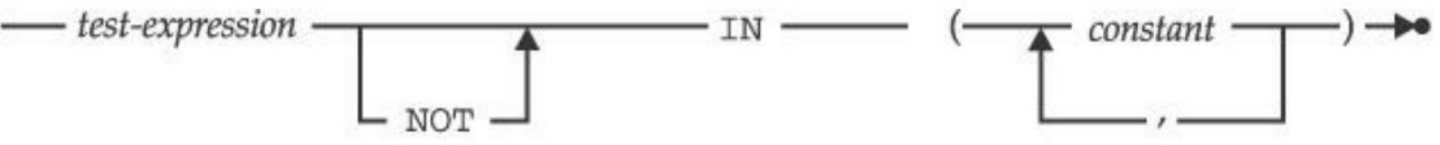
Тестирование нулевого значения

```
SELECT * FROM employees where building is null;
```

Role	Name	Building	Years_employed
Engineer	Yancy I.		0
Artist	Oliver P.		0

STORMNET

Тестирование роли уровня базы данных [IN]



Тестирование роли уровня базы данных [IN]

```
SELECT * FROM Customers WHERE PostalCode IN (12209, 44000, 78000) limit 5;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
18	Du monde entier	Janine Labrune	67, rue des Cinquante Otages	Nantes	44000	France
26	France restauration	Carine Schmitt	54, rue Royale	Nantes	44000	France
40	La corne d'abondance	Daniel Tonini	67, avenue de l'Europe	Versailles	78000	France

Основные команды. Distinct

Distinct – команда для вывода уникальных имен (без повторов)

SELECT distinct Country
FROM Customers

Country
Germany
Mexico
UK
Sweden

Повторяющиеся строки [DISTINCT]

ID_CLASS
347
347
348
349
350

STORMNET

Повторяющиеся строки [DISTINCT]

```
SELECT City FROM Customers limit 4;
```

City
Berlin
México D.F.
México D.F.
London

```
SELECT distinct City FROM Customers limit 4;
```

City
Berlin
México D.F.
London
Luleå

STORMNET

Основные команды. Drop

Таблицы и базы данных можно удалить с помощью этой команды.

DROP statement.

DROP TABLE table_name;

DROP DATABASE database_name;

STORMNET

Комплексные Select

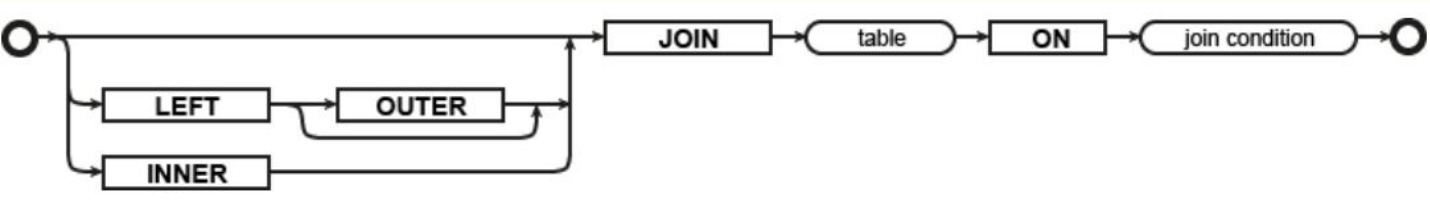
SQL JOIN

Выбирает значения из нескольких таблиц одновременно.

Наиболее часто используемых 4:

- Inner Join
- Left join
- Right Join
- Full Outer Join

Объединения [JOINS]

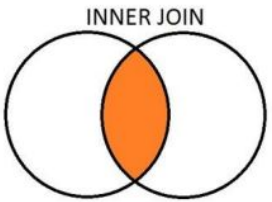


JOIN это горизонтальное объединение

```
mysql> SELECT customerID, customerName FROM customers limit 5;
+------------+--------------------------------------+
| customerID | customerName                         |
+------------+--------------------------------------+
|          1 | Alfreds Futterkiste                  |
|          2 | Ana Trujillo Emparedados y helados   |
|          3 | Antonio Moreno Taquerna              |
|          4 | Around the Horn                      |
|          5 | Berglunds snabbkup                   |
+------------+--------------------------------------+
```

```
mysql> SELECT orderID, customerID, orderDate FROM orders limit 5
+---------+------------+------------+
| orderID | customerID | orderDate  |
+---------+------------+------------+
|   10248 |         90 | 1996-07-04 |
|   10249 |         81 | 1996-07-05 |
|   10250 |         34 | 1996-07-08 |
|   10251 |         84 | 1996-07-08 |
|   10252 |         76 | 1996-07-09 |
+---------+------------+------------+
```

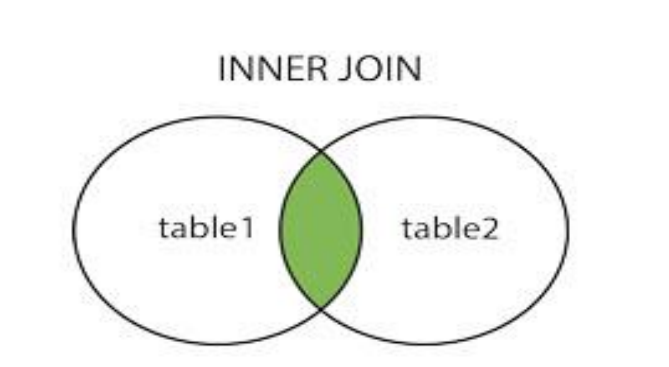
```
mysql> SELECT customers.customerID, customers.customerName, orders.orderID, orders.customerID, orders.orderDate FROM customers
    -> JOIN orders ON customers.customerID = orders.customerID
    -> ORDER BY customers.customerID
    -> LIMIT 5;
+------------+--------------------------------------+---------+------------+------------+
| customerID | customerName                         | orderID | customerID | orderDate  |
+------------+--------------------------------------+---------+------------+------------+
|          2 | Ana Trujillo Emparedados y helados   |   10308 |          2 | 1996-09-18 |
|          3 | Antonio Moreno Taquerna              |   10365 |          3 | 1996-11-27 |
|          4 | Around the Horn                      |   10383 |          4 | 1996-12-16 |
|          4 | Around the Horn                      |   10355 |          4 | 1996-11-15 |
|          5 | Berglunds snabbkup                   |   10278 |          5 | 1996-08-12 |
+------------+--------------------------------------+---------+------------+------------+
```



STORMNET

Комплексные Select

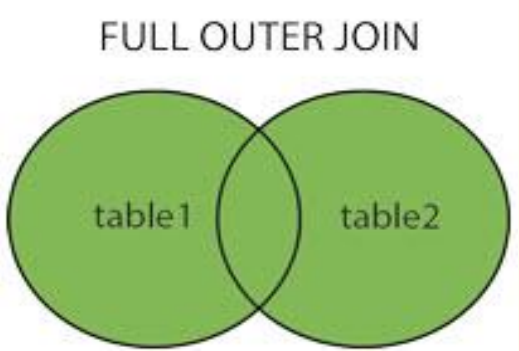
SQL INNER JOIN – Вернет все записи из всех таблиц, где есть пересечение




```
SELECT Product.ProductID, Orders.Quantity, Orders.OrderDate
FROM Orders
INNER JOIN Product
ON Product.ProductID=Orders.OrderID;
```

Комплексные Select

SQL FULL OUTER JOIN – Вернет все записи из обеих таблицы совместив записи, где есть пересечение.




```
SELECT Product.ProductID, Orders.Quantity, Orders.OrderDate
FROM Orders
FULL OUTER JOIN Product
ON Product.ProductID=Orders.OrderID;
```

Внутреннее vs Левое Объединение

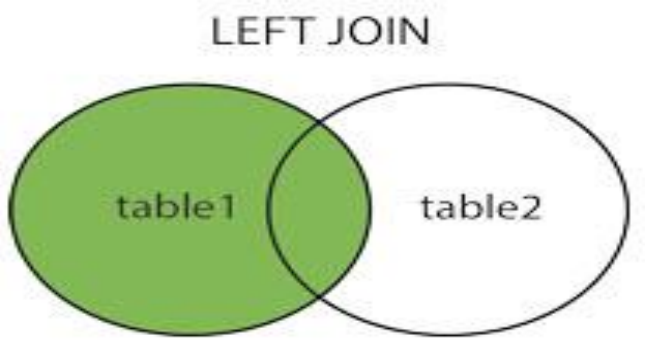
```
mysql> SELECT customers.customerID, customers.customerName, orders.orderDate FROM customers
    -> JOIN orders ON customers.customerID = orders.customerID
    -> ORDER BY customers.customerID
    -> LIMIT 5;
+------------+------------------------------------+------------+
| customerID | customerName                       | orderDate  |
+------------+------------------------------------+------------+
|          2 | Ana Trujillo Emparedados y helados | 1996-09-18 |
|          3 | Antonio Moreno Taquerna            | 1996-11-27 |
|          4 | Around the Horn                    | 1996-12-16 |
|          4 | Around the Horn                    | 1996-11-15 |
|          5 | Berglunds snabbkup                 | 1996-08-12 |
+------------+------------------------------------+------------+
```

```
mysql> SELECT customers.customerID, customers.customerName, orders.orderDate FROM customers
    -> LEFT JOIN orders ON customers.customerID = orders.customerID
    -> ORDER BY customers.customerID
    -> LIMIT 5;
+------------+------------------------------------+------------+
| customerID | customerName                       | orderDate  |
+------------+------------------------------------+------------+
|          1 | Alfreds Futterkiste                | NULL       |
|          2 | Ana Trujillo Emparedados y helados | 1996-09-18 |
|          3 | Antonio Moreno Taquerna            | 1996-11-27 |
|          4 | Around the Horn                    | 1996-12-16 |
|          4 | Around the Horn                    | 1996-11-15 |
+------------+------------------------------------+------------+
```


STORMNET

Комплексные Select

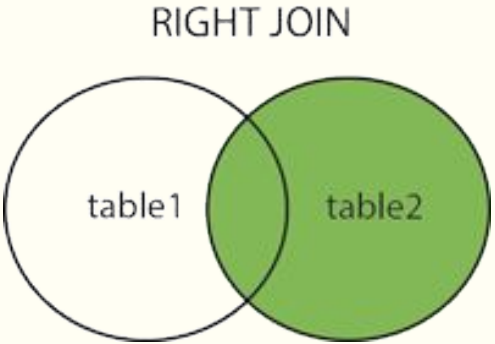
SQL LEFT JOIN – Вернет все записи из левой таблицы и записи из правой, где есть пересечение.




```
SELECT Product.ProductID, Orders.Quantity, Orders.OrderDate
FROM Orders
LEFT JOIN Product
ON Product.ProductID=Orders.OrderID;
```

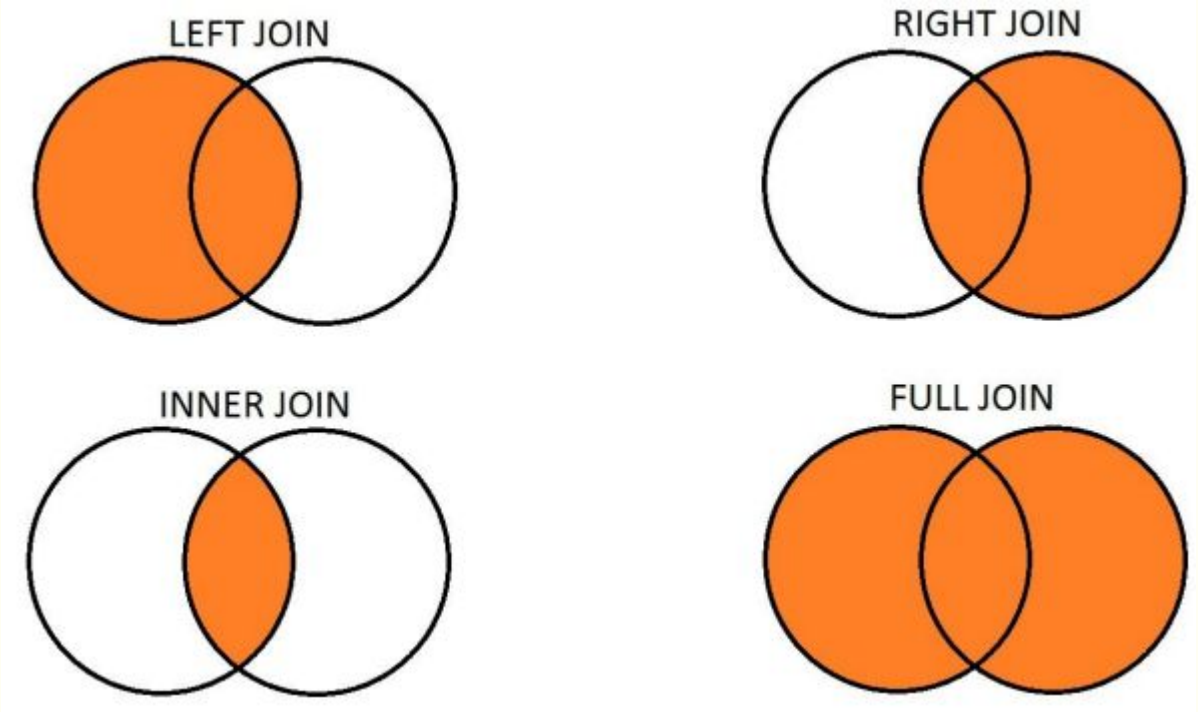
Комплексные Select

SQL RIGHT JOIN – Вернет все записи из правой таблицы и записи из левой, где есть пересечение.




```
SELECT Product.ProductID, Orders.Quantity, Orders.OrderDate
FROM Orders
RIGHT JOIN Product
ON Product.ProductID=Orders.OrderID;
```

Объединение



Нормализация баз данных

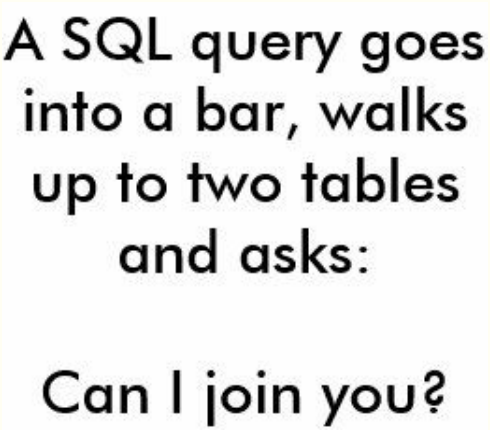
- это процесс организации данных в базе данных, включающий создание таблиц и установление отношений между ними в соответствии с правилами, которые обеспечивают защиту данных и делают базу данных более гибкой, устраняя избыточность и несогласованные зависимости.

Денормализация

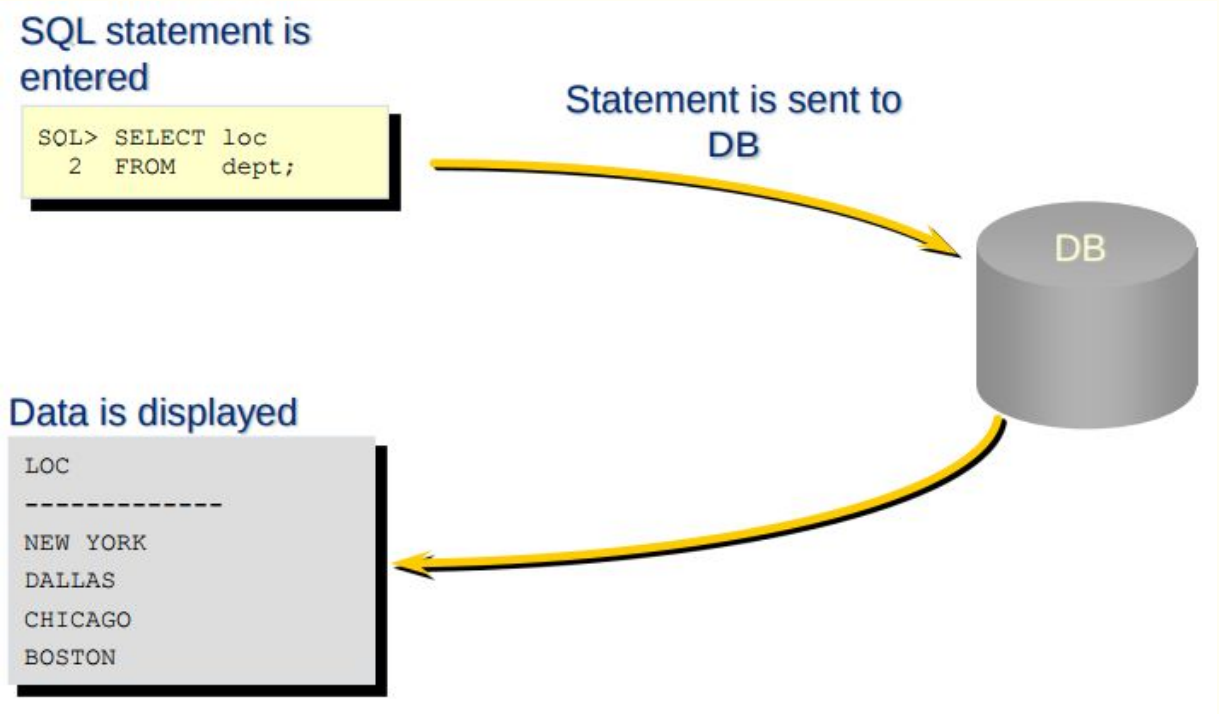
- Улучшение производительности сложных запросов, которые выполняются долго за счет добавления необходимой избыточности данных;
- Упрощение выполнения запросов на клиенте (вплоть до 1-й таблицы) ;
- Улучшение эффективности работы клиента с базой;
- Упрощает структуру для конечного пользователя.

Реляционная база данных

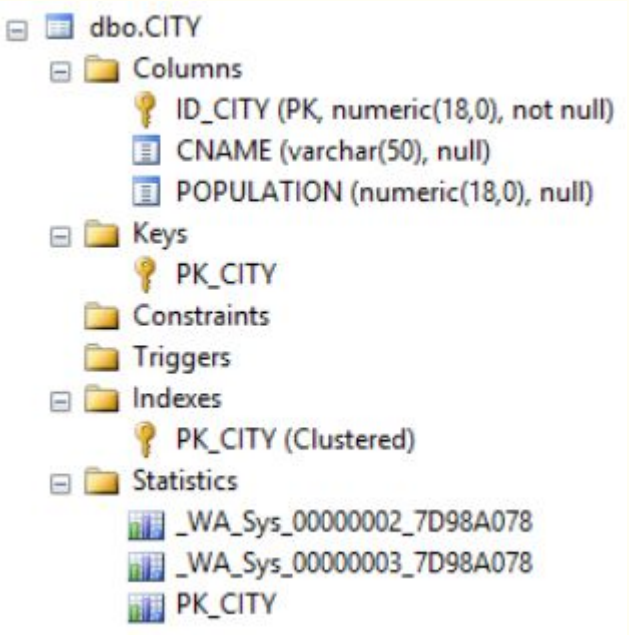
- Доступ и изменение данных в реляционной базе данных осуществляется при помощи выполнения команд на структурированном языке запросов (SQL).



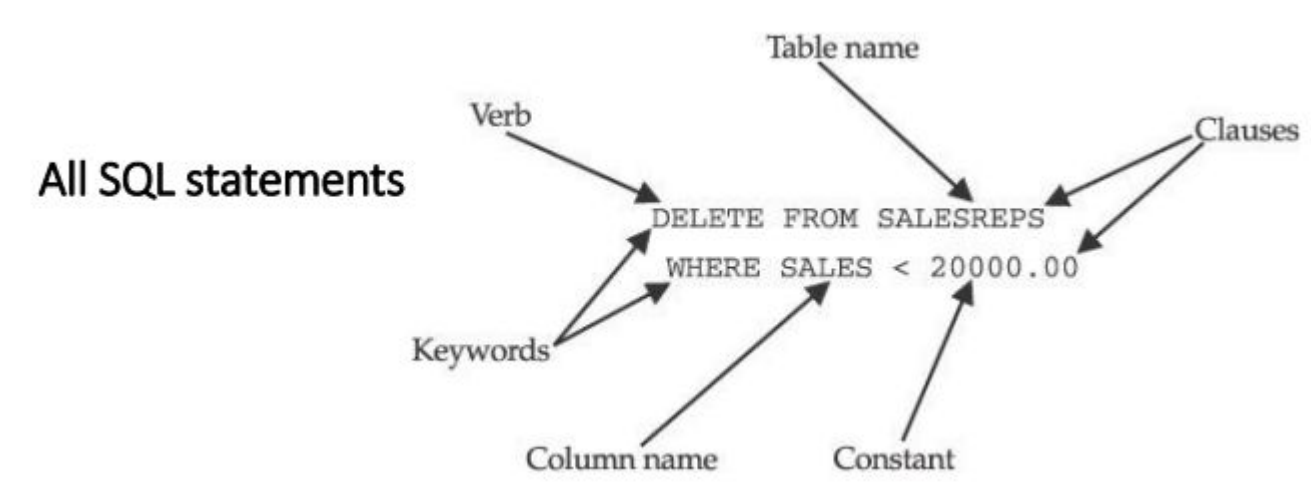
SQL как средство общения с СУБД



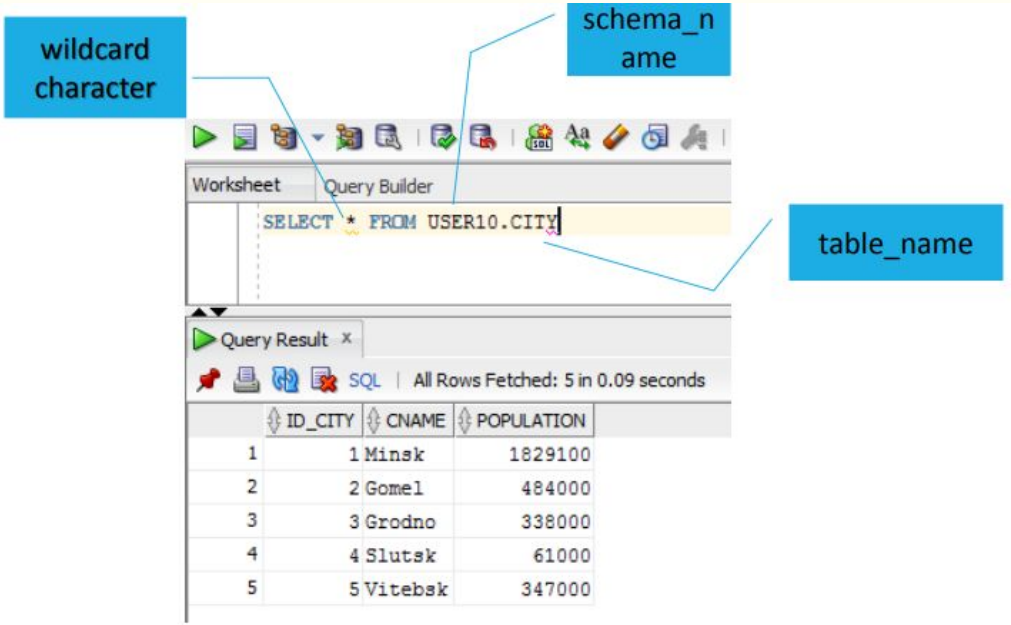
Структура


dbo.CITY
Columns
ID_CITY (PK, numeric(18,0), not null)
CNAME (varchar(50), null)
POPULATION (numeric(18,0), null)
Keys
PK_CITY
Constraints
Triggers
Indexes
PK_CITY (Clustered)
Statistics
_WA_Sys_00000002_7D98A078
_WA_Sys_00000003_7D98A078
PK_CITY

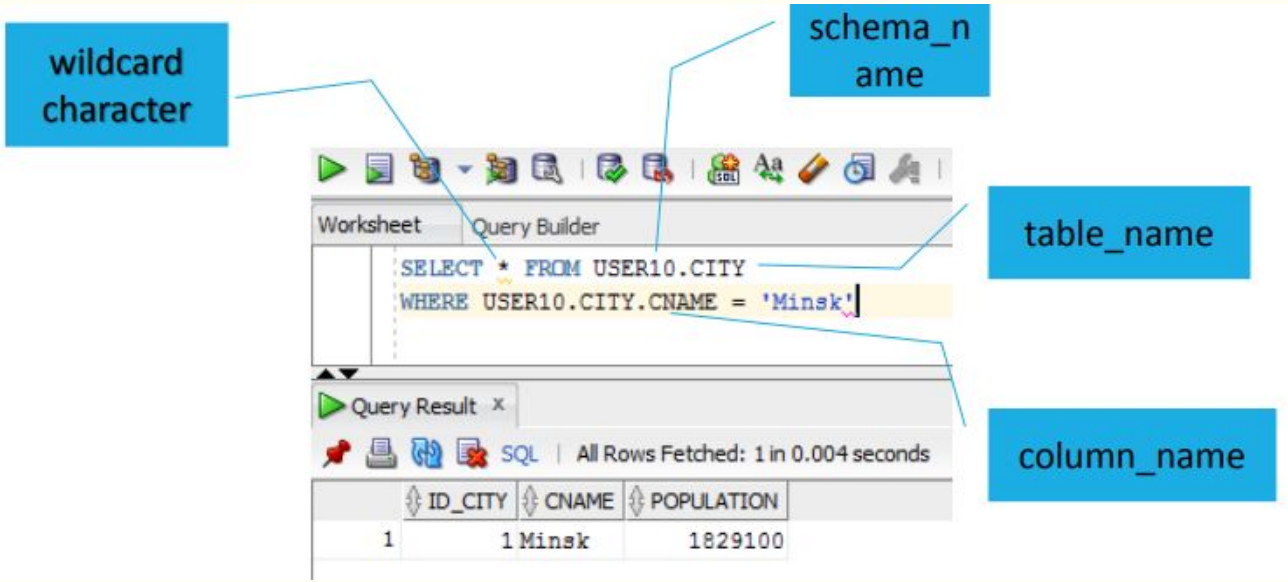
Запросы


All SQL statements
Table name
Verb
Clauses
DELETE FROM SALESREPS
WHERE SALES < 20000.00
Keywords
Column name
Constant

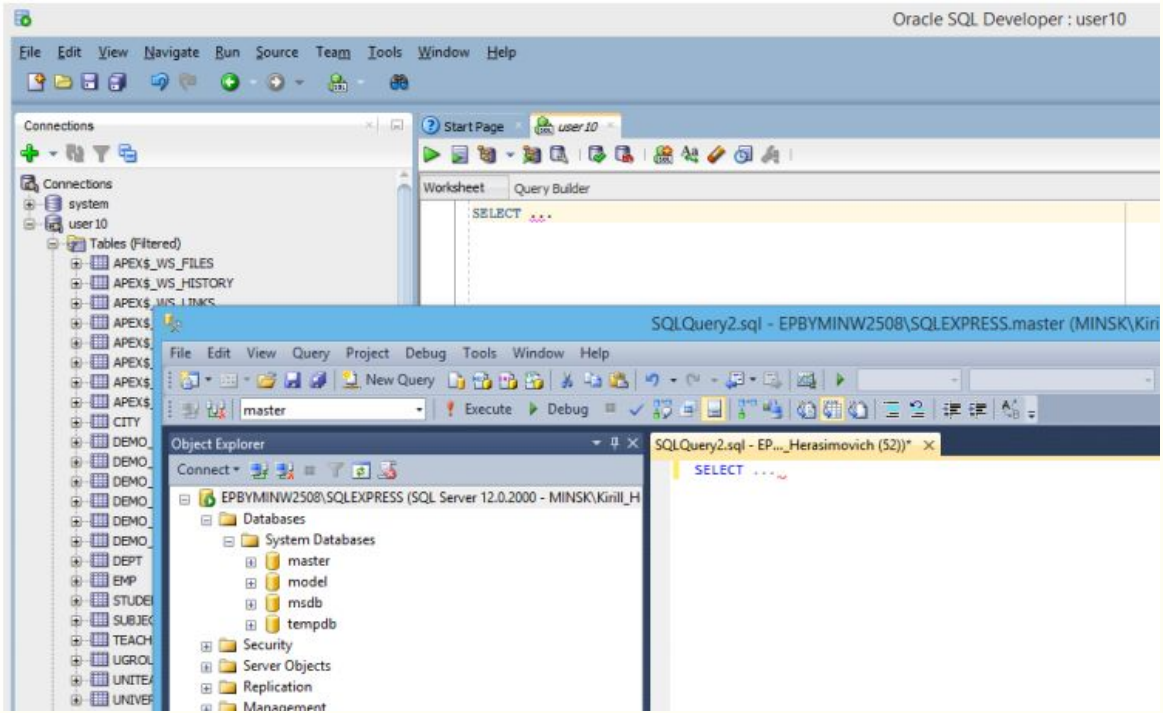
Названия таблиц



Названия колонок



Используйте любую интегрированную среду разработки [IDE] для работы с SQL



Встроенные функции

- ***Агрегатные функции*** выполняют вычисление на наборе значений в столбце и возвращают одиночное значение.
- ***Скалярные функции*** SQL возвращают одно значение, основанное на входном значении.

Встроенные функции

- ***Полезные агрегатные функции:***

  AVG() - Возвращает среднее значение
  COUNT() - Возвращает количество строк
  MAX() - Возвращает наибольшее значение
  MIN() - Возвращает наименьшее значение
  SUM() - Возвращает сумму

- ***Скалярные функции:***

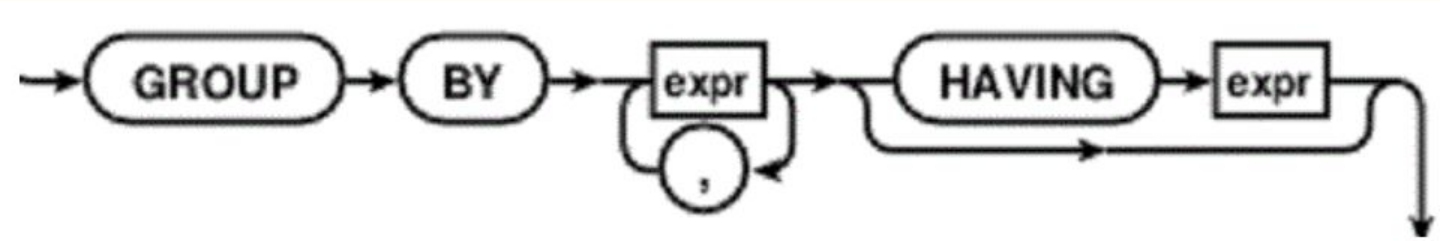
  CHAR_LENGTH() - Возвращает длину текстового поля в символах.
  ROUND() - Возвращает числовое значение, округленное до указанной длины или точности.
  NOW() - Возвращает текущее время и дату.
  FORMAT() - Форматирование для определенного отображения поля

Агрегатные функции

```
SELECT count(*) FROM Employees;
```

count(*)
10

Агрегатные функции [группировки и фильтрация]



Агрегатные функции [GROUP BY Clause]

SELECT Country, count(Country) FROM Customers Group by Country;

Country	count(Country)
Argentina	3
Austria	2
Belgium	2
Brazil	9
Canada	3

Оператор **GROUP BY** используется вместе с агрегатными функциями для группировки результата по признакам одного или более столбцов.

STORMNET

Агрегатные функции [HAVING Clause]

```
SELECT Country, count(Country) FROM Customers Group by Country HAVING count(Country) > 8;
```

Country	count(Country)
Brazil	9
France	11
Germany	11
USA	13

Оператор **HAVING** был добавлен в SQL потому, что оператор WHERE не может быть использован вместе с агрегатной функцией.

STORMNET

STORMNET