<https://www.w3schools.com/>

sql-ex.ru

Виды баз данных:

1. Реалиционные – табличка (строка запись, таблица атрибут) как жесткий диск

Реалиация внутри баз данных – это взаимосвязи между таблицами, которые строятся на связи между столбиками.

Свойства реляции:

- фильтрация данных

- каскадирование данных, то есть каскадное удаление, удаляя строку из материнской таблицы, данные по объекту удалятся и из дочерних таблиц.

Виды:

MySQL

Postgrace

Oracle

1. Нереалиционные – списки, как оперативная база

Reddis

MangoDB

CУБД – системы управления базами данных. Содержат в себя базы данных

Базы данных содержат в себе таблицу(ы)

Таблицы содержат в себе строки и столбцы.

DBever PDadmin – инструменты позволяющие подключится к базам данных. Называются IDE – системы разработки (типо VS code)

**create** **table** salary(

id serial **primary** **key**,

month\_salary **int** **not** **null**

);

**create** **table** roles(

id serial **primary** **key**,

role\_title **varchar**(30) **unique** **not** **null**

);

**create** **table** roles\_salary(

id serial **primary** **key**,

id\_role **int** **not** **null**,

id\_salary **int** **not** **null**,

**foreign** **key** (id\_role) **references** roles(id),

**foreign** **key** (id\_salary) **references** salary(id)

);

**insert** **into** salary (month\_salary)

**values** (500),

(700),

(900),

(1000),

(1200),

(1400),

(1500),

(1700),

(2000);

**insert** **into** roles(role\_title)

**values** ('QA\_manual\_junior'),

('QA\_manual\_middle'),

('QA\_manual\_senior'),

('QA\_automation\_junior'),

('QA\_automation\_middle'),

('QA\_automation\_senior'),

('Java\_developer\_junior'),

('Java\_developer\_middle'),

('Java\_developer\_senior'),

('CEO'),

('CTO'),

('HR'),

('Manager'),

('Designer');

**insert** **into** roles\_salary(id\_role, id\_salary)

**values** (1, 1),

(2,4),

(3, 7),

(3, 7),

(8, 6),

(8, 6),

(11, 15);

**SELECT** role\_title, month\_salary **FROM** roles\_salary

**join** roles **on** roles\_salary.id\_role = roles.id

**join** salary **on** roles\_salary.id\_salary = salary.id;

**alter** **table** roles

**add** **column** parking **int**;

**alter** **table** roles

**rename** **column** parking **to** taxi

**alter** **table** roles

**drop** **column** taxi

**update** salary

**set** month\_salary = 1450

**where** id = 5

**select** id, month\_salary **from** salary

**UPDATE** salary

**SET** month\_salary = **CASE**

**WHEN** id = 1 **THEN** 1000

**WHEN** id = 2 **THEN** 1100

**WHEN** id = 3 **THEN** 1200

**WHEN** id = 4 **THEN** 1300

**WHEN** id = 5 **THEN** 1400

**WHEN** id = 6 **THEN** 1500

**WHEN** id = 7 **THEN** 1600

**WHEN** id = 8 **THEN** 1700

**WHEN** id = 9 **THEN** 1800

**WHEN** id = 10 **THEN** 1900

**WHEN** id = 11 **THEN** 2000

**WHEN** id = 12 **THEN** 2100

**WHEN** id = 13 **THEN** 2200

**WHEN** id = 14 **THEN** 2300

**WHEN** id = 15 **THEN** 2400

**WHEN** id = 16 **THEN** 2500

**ELSE** month\_salary

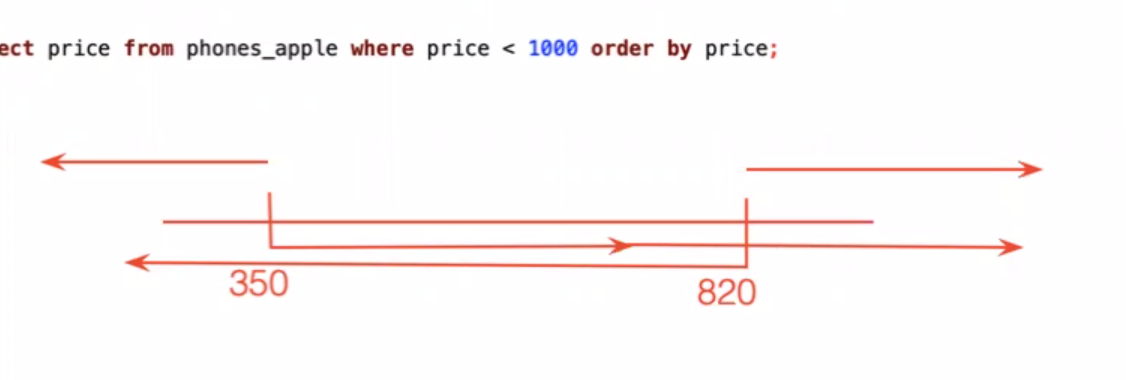
**END**;

В случае с ALL ниже в подзапросе мы смотрим больше максимального или меньше минимального

В случае с ANY мы смотрим меньше максимального, больше минимального

Операторы сравнения могут быть использованы для возврата значений списка в сравнении с одним значением подзапроса

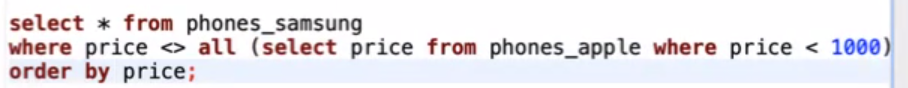
Оператор in / not in могу сравнивать списки – (есть один во втором или нет)





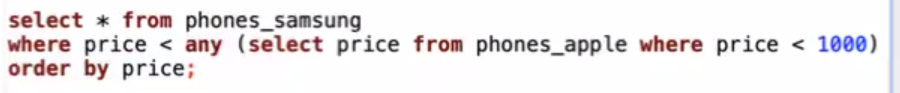
Вывести самуснги у которых цена меньше самой маленькой цены айфонов

В подзапросе where price < 1000 – это граничное значение до или от которого смотрим (в случае если “=><”)

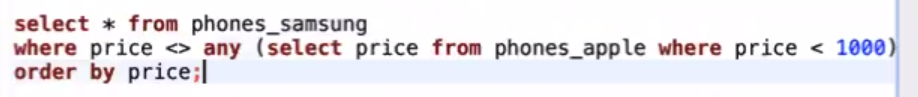


Возвращает самсунги цена которых не входит в список цен айфонов (not in)

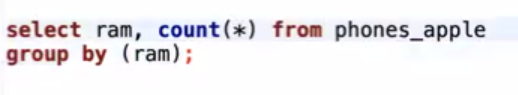
\* всё что больше максимального и всё что меньше минимального, всё что не входит в диапозон

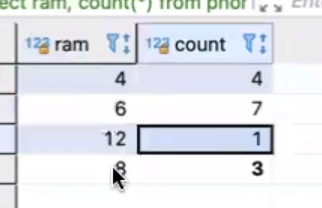


Возвращает самсунги цена которых меньше чем максимальная цена айфона



Возвращает всё что больше минимального и всё что меньше максимального



показывает количество телефонов с разным объемом памяти