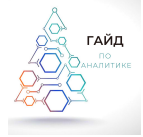



Шпаргалка по базовому SQL. Часть 2



АГРЕГАЦИЯ И ГРУППИРОВКА

GROUP BY группирует вместе строки, которые имеют одинаковые значения в определенных полях. Group by подводит итог (агрегирует) для каждого уникального значения столбца, в котором применена группировка.

CITY				
id	name	country_id		
1	Paris	1		
101	Marseille	1		
102	Lyon	1		
2	Berlin	2		
103	Hamburg	2		
104	Munich	2		
3	Warsaw	4		
105	Cracow	4		



CITY	
country_id	count
1	3
2	3
4	2

ФУНКЦИИ АГРЕГАЦИИ

- **avg(expr)** - среднее значение строк в рамках группы
- **count(expr)** - количество значений в строках в рамках группы
- **max(expr)** - максимальное значение в группе
- **min(expr)** - минимальное значение в группе
- **sum(expr)** - сумма значений в рамках группы

ПРИМЕРЫ ЗАПРОСОВ

Найти количество городов:

```
SELECT COUNT(*)  
FROM city;
```

Найти количество городов с ненулевым рейтингом:

```
SELECT COUNT(rating)  
FROM city;
```

найти количество уникальных стран:

```
SELECT COUNT(DISTINCT country_id)  
FROM city;
```

Найти наибольшее и наименьшее население среди стран:

```
SELECT MIN(population), MAX(population)  
FROM country;
```

Найти общее население городов по каждой стране:

```
SELECT country_id, SUM(population)  
FROM city  
GROUP BY country_id;
```

Найти средний рейтинг городов по каждой стране, с фильтром по среднему рейтингу больше 3.0:

```
SELECT country_id, AVG(rating)  
FROM city  
GROUP BY country_id  
HAVING AVG(rating) > 3.0;
```

ПОДЗАПРОСЫ

Подзапрос это запрос, который вложен в другой запрос. Ниже представлены разные типы подзапросов.

ОДИНОЧНОЕ ЗНАЧЕНИЕ

Простой подзапрос возвращает возвращает единственное значение в заданном столбце. Может использоваться с операторами сравнения =, <, <=, >, or >=.

Этот запрос находит города с таким же рейтингом, как Париж:

```
SELECT name FROM city  
WHERE rating = (  
    SELECT rating  
    FROM city  
    WHERE name = 'Paris'  
);
```

МНОЖЕСТВЕННЫЕ ЗНАЧЕНИЯ

Подзапрос также может возвращать несколько полей или несколько строк. Такие подзапросы работают с операторами IN, EXISTS, ALL, or ANY.

Этот запрос находит города в странах с населением более 20 миллионов человек:

```
SELECT name  
FROM city  
WHERE country_id IN (  
    SELECT country_id  
    FROM country  
    WHERE population > 20000000  
);
```

СВЯЗАННЫЙ ПОДЗАПРОС

Связанный подзапрос относится к таблицам, представленным во внешнем запросе. Связанный подзапрос зависит от внешнего запроса. Его нельзя запустить независимо от внешнего запроса.

Этот запрос находит города с населением больше, чем в среднем по стране:

```
SELECT *  
FROM city main_city  
WHERE population > (  
    SELECT AVG(population)  
    FROM city average_city  
    WHERE average_city.country_id = main_city.country_id  
);
```

Этот запрос находит страны, в которых есть хотя бы один город:

```
SELECT name  
FROM country  
WHERE EXISTS (  
    SELECT *  
    FROM city  
    WHERE country_id = country.id  
);
```

ОПЕРАЦИИ НАД МНОЖЕСТВАМИ

Операции над множествами используются для объединения результатов двух или более запросов в единый результат. Комбинированные запросы должны возвращать одинаковое количество полей и совместимых типов данных. Имена соответствующих полей могут быть разными.

CYCLING			SKATING		
id	name	country	id	name	country
1	YK	DE	1	YK	DE
2	ZG	DE	2	DF	DE
3	WT	PL	3	AK	PL
...

UNION

UNION объединяет результаты двух запросов и удаляет дубликаты. UNION ALL не удаляет дубликаты.

Этот запрос получает немецких велосипедистов вместе с немецкими скейтерами:

```
SELECT name  
FROM cycling  
WHERE country = 'DE'  
UNION / UNION ALL  
SELECT name  
FROM skating  
WHERE country = 'DE';
```



INTERSECT

INTERSECT возвращает только те строки, которые присутствуют в обоих результатах запросов.

Этот запрос возвращает немецких велосипедистов, которые, к тому же, являются немецкими скейтерами:

```
SELECT name  
FROM cycling  
WHERE country = 'DE'  
INTERSECT  
SELECT name  
FROM skating  
WHERE country = 'DE';
```



EXCEPT

EXCEPT возвращает только те строки, которые присутствуют в результате первого запроса, но отсутствуют в результате второго запроса.

Этот запрос отображает немецких велосипедистов, которые не являются немецкими скейтерами:

```
SELECT name  
FROM cycling  
WHERE country = 'DE'  
EXCEPT / MINUS  
SELECT name  
FROM skating  
WHERE country = 'DE';
```

