



GreenData

Введение в базы данных и SQL

Прикладной курс

Часть 3. Простые SQL запросы

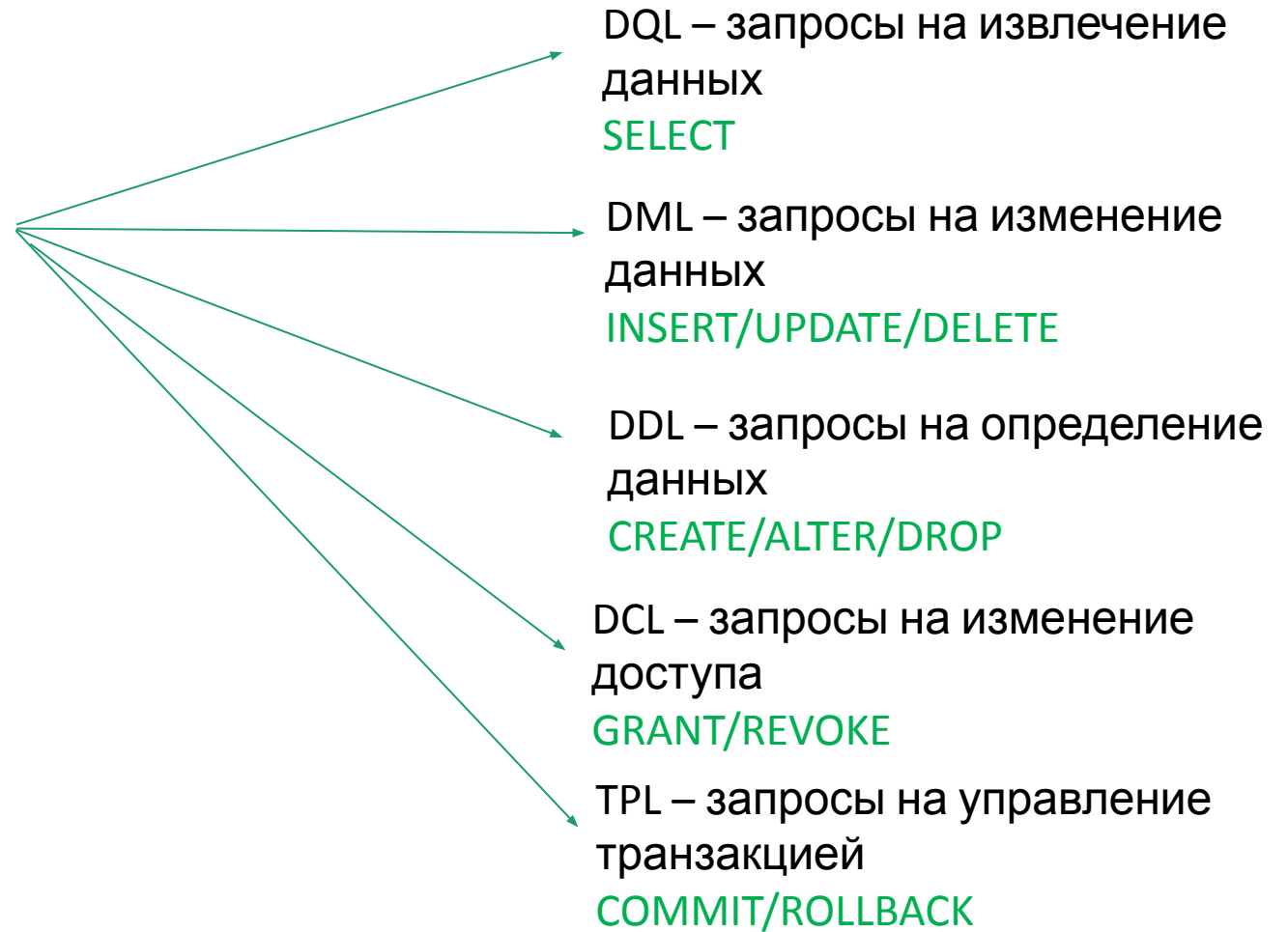
(С) 2021 Архипов А.В.

SQL (Structured Query Language)

Структурированный язык запросов

Как было до SQL:

```
assign(f, 'c:\data\somefile.dat');  
reset(f);  
readline(f, s1);  
readline(f, s2);  
close(f);
```



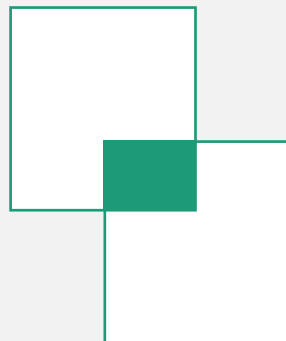
Объединени

е



Пересечени

е



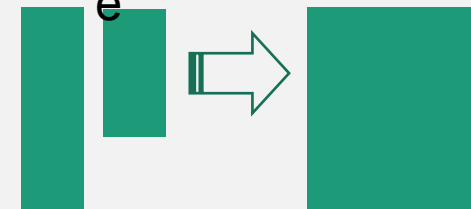
Вычитани

е



Произведени

е



Традиционны
е

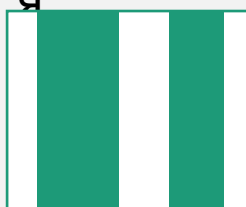
Выборк

а



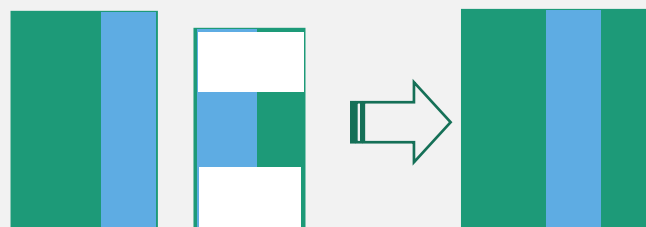
Проекци

я



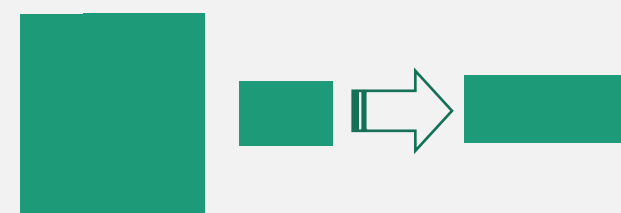
Соединени

е



Делени

е



Специализированн
ые



Объединени
е



```
SELECT first_name,
       second_name,
       last_name
FROM math_students
UNION
SELECT first_name,
       second_name,
       last_name
FROM sport_students
```

UNION – медленный оператор!

UNION ALL – быстрый, но не убирает повторяющиеся записи



	first_name character varying (500)	second_name character varying (500)	last_name character varying (500)
1	Лев	Николаевич	Толстой
2	Артем	Петрович	Чайников
3	Сергей	Васильевич	Левитан
4	Андрей	Иванович	Седов
5	Петр	Савельевич	Симонов
6	Антон	Павлович	Чехов
7	Михаил	Егорович	Архипов



Использован
ие
звездочек в
приложении
Запрещен

o

```
SELECT *  
FROM math_students  
UNION  
SELECT *  
FROM sport_students
```

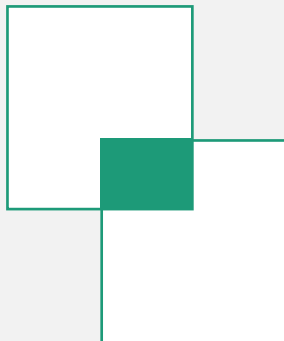
Зачем мне
перечислять поля,
когда я могу поставить
звездочку!

Я тут добавила
спортсменам
столбец с
телефончиком

Мой запрос теперь не
работает!



Пересечени
е



```
SELECT first_name,  
       second_name,  
       last_name  
FROM math_students  
INTERSECT  
SELECT first_name,  
       second_name,  
       last_name  
FROM sport_students
```



	first_name character varying (500)	second_name character varying (500)	last_name character varying (500)
1	Андрей	Иванович	Седов



Вычитани

е



```
SELECT first_name,  
       second_name,  
       last_name  
FROM math_students  
EXCEPT  
SELECT first_name,  
       second_name,  
       last_name  
FROM sport_students
```

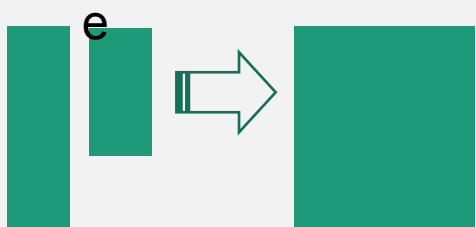


	first_name character varying (500)	second_name character varying (500)	last_name character varying (500)
1	Лев	Николаевич	Толстой
2	Артем	Петрович	Чайников
3	Сергей	Васильевич	Левитан
4	Антон	Павлович	Чехов



* Для Oracle **MINUS**

Произведени



```
SELECT p.first_name,  
       p.second_name,  
       p.last_name,  
       l.name,  
       l.eng_name  
FROM programmers p,  
     programmer_level l  
ORDER BY 1, 2, 3, 5
```

```
SELECT c.credit_num,  
       c.amount As credit_amount,  
       m.n As mon_num,  
       m.n*month_payment As paid  
FROM month_num m,  
     my_credit c  
ORDER BY c.credit_num, m.n
```



	first_name character varying (500)	second_name character varying (500)	last_name character varying (500)	name character varying (500)	eng_name character varying (500)
1	Арте́м	Петрови́ч	Симонов	Младший	June
2	Арте́м	Петрови́ч	Симонов	Средний	Middle
3	Арте́м	Петрови́ч	Симонов	Высший	Senior
4	Сере́гей	Василье́вич	Чудинов	Младший	June
5	Сере́гей	Василье́вич	Чудинов	Средний	Middle
6	Сере́гей	Василье́вич	Чудинов	Высший	Senior

^
Задание! Добавить
колонку с остатком
долга по кредиту!



Выборк

a

1

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE year = 2019
```

2

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE mark = 'KIA'
```



	id [PK] integer	mark character varying	model character varying	year integer
1	3	KIA	RIO	2019
2	4	KIA	OPTIMA	2019



Выборк

a

3

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE year = 2019  
       AND mark = 'KIA'
```

4

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE mark = 'KIA'  
       OR mark = 'MAZDA'
```

5

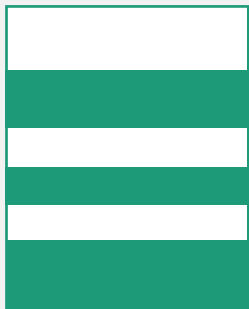
```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE year in (2019, 2020)
```

6

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE year not in (2019, 2020)
```

Выборка

а



7

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE mark LIKE '%DA%'
```

8

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE mark LIKE '_IA'
```



	id [PK] integer	mark character varying	model character varying	year integer
1	3	KIA	RIO	2019
2	4	KIA	OPTIMA	2019

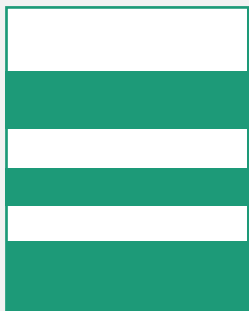
% - любое количество любых символов

_ - один любой символ, не пустой

\%, \% - экранирование

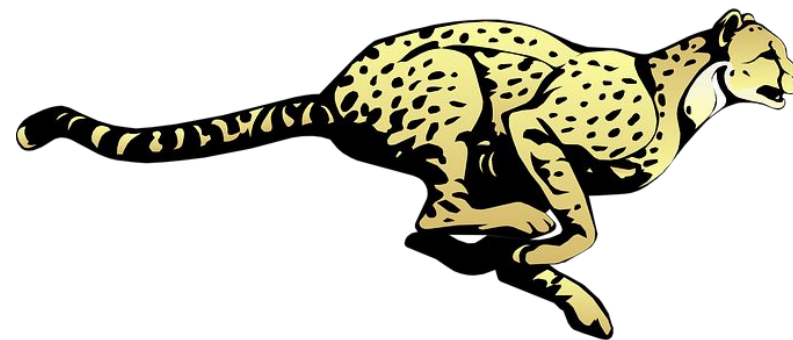
Выборк

а



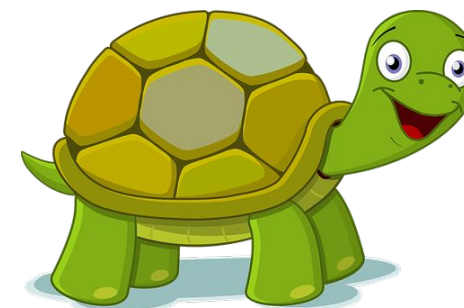
9

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE mark LIKE 'MA%'
```



10

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE mark LIKE '%ZD%'
```

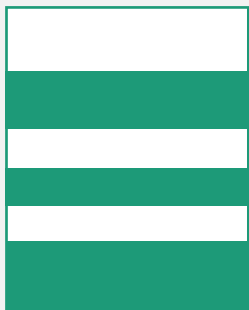


Использование
Like



Выборк

а



Практический кейс

Справочник клиентов, как могут приходить клиенты:

ООО 'Работа не волк'

ООО «Ракета»

ОАО Ветер

ООО, «Одежда для программистов»

Я помню, только слово «Ракета».

Быстрый поиск не подойдет.

Исходное	Для поиска
ООО 'Работа не волк'	Работа не волк, ООО
ООО «Ракета»	Ракета, ООО
ОАО Ветер	Ветер, ОАО
ООО, «Одежда для программистов»	Одежда для программистов, ООО

Выборк

a

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE UPPER(model) like UPPER('granta')
```

```
SELECT id,  
       mark,  
       model,  
       year  
FROM best_car  
WHERE LOWER(model) like LOWER('granta')
```



	Data Output	Explain	Messages	Notifications
	id [PK] integer	mark character varying	model character varying	year integer
1	1	LADA	GRANTA	2020

Проекци
я



1) Делаем проекцию по столбцам MARK,
MODEL:

```
SELECT DISTINCT  
    mark,  
    model  
FROM best_car
```

2) Делаем проекцию по столбцам MARK,
YEAR

```
SELECT DISTINCT  
    mark,  
    year  
FROM best_car
```

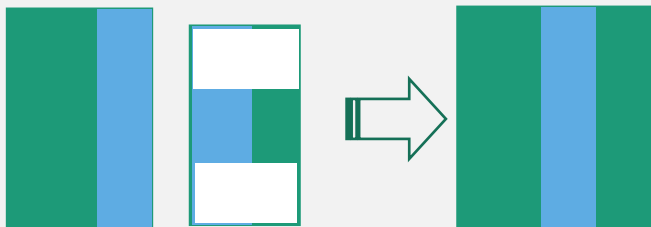


	mark	year
1	MAZDA	2020
2	LADA	2020
3	MAZDA	2019
4	KIA	2019



Используй DISTINCT
в алгоритмах?
Проверь модель!

Соединение



Внутреннее соединение **INNER JOIN**

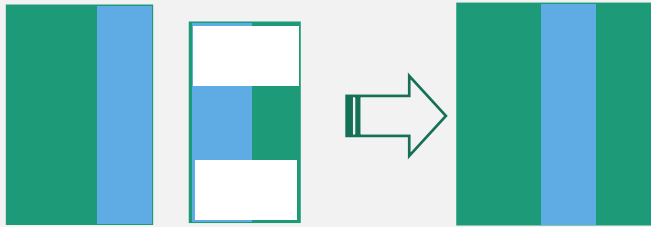
	last_name character varying (100)	first_name character varying (100)	position_name character varying (500)
1	Иванов	Иван	Ассистент
2	Петров	Петр	Преподаватель
3	Сидоров	Сергей	Преподаватель
4	Зайцев	Зигмунд	Ассистент

```
SELECT
    t.last_name As last_name,
    t.first_name As first_name,
    p.name As position_name
FROM my_teacher t
INNER JOIN teacher_position p ON (t.position_id = p.id)
```

Альтернативный вариант:

```
SELECT
    t.last_name As last_name,
    t.first_name As first_name,
    p.name As position_name
FROM my_teacher t,
    teacher_position p
WHERE t.position_id = p.id
```


Соединение



LEFT JOIN

```
SELECT
    t.last_name As last_name,
    t.first_name As first_name,
    p.name As position_name
FROM my_teacher t
LEFT OUTER JOIN teacher_position p ON (t.position_id = p.id)
```

RIGHT JOIN

```
SELECT
    t.last_name As last_name,
    t.first_name As first_name,
    p.name As position_name
FROM my_teacher t
RIGHT OUTER JOIN teacher_position p ON (t.position_id = p.id)
```

Внешнее соединение

OUTER JOIN

	last_name character varying (100)	first_name character varying (100)	position_name character varying (500)
1	Иванов	Иван	Ассистент
2	Петров	Петр	Преподаватель
3	Сидоров	Сергей	Преподаватель
4	Зайцев	Зигмунд	Ассистент
5	Быстров	Борис	[null]
6	Седов	Сергей	[null]
7	[null]	[null]	Доцент
8	[null]	[null]	Старший преподаватель
9	[null]	[null]	Профессор

FULL JOIN

```
SELECT
    t.last_name As last_name,
    t.first_name As first_name,
    p.name As position_name
FROM my_teacher t
FULL OUTER JOIN teacher_position p ON (t.position_id = p.id)
```

Деление

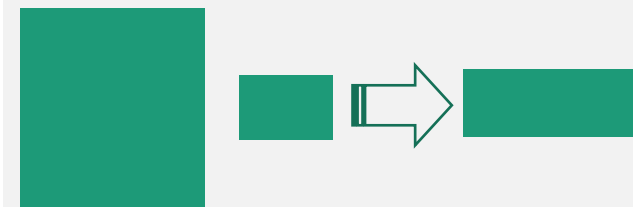


Таблица 1. Преподаватели

ФИО	Предмет
Архипов А.	Введение в БД и SQL
Зигаленко В.	Управление проектами ●
Савельев И.	Искусство продаж
Зигаленко В.	Искусство продаж ●
Кузьмина Н.	Управление риском
Архипов А.	Управление проектами

Таблица 2.
Дисциплины,
которые хочу посетить

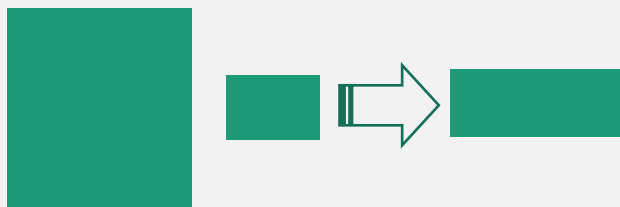
Предмет
Управление проектами
Искусство продаж

Напомню о
чем тема

3. Преподаватели,
которые сразу читают два предмета

ФИО	Предмет
Зигаленко В.	Управление проектами
Зигаленко В.	Искусство продаж

Деление



АГРЕГАТНЫЕ ФУНКЦИИ COUNT, SUM, AVG, MIN, MAX

Сравните

```
SELECT COUNT(id)  
FROM my_teacher
```

```
SELECT COUNT(position_id)  
FROM my_teacher
```

Считает записи с
непустым
атрибутом

Немного об
агрегатных
функциях

```
SELECT SUM(id)  
FROM my_teacher
```

```
SELECT AVG(id)  
FROM my_teacher
```

```
SELECT MIN(id)  
FROM my_teacher
```

Мат. операция
по столбцу

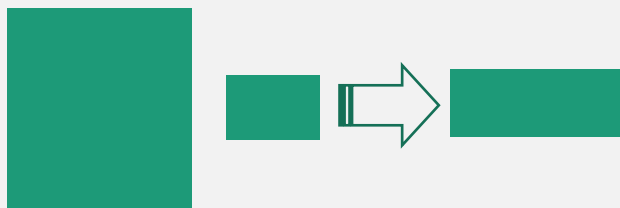
```
SELECT  
    position_id,  
    COUNT(*) As n  
FROM my_teacher  
GROUP BY position_id
```

```
SELECT  
    teacher As name,  
    Count(*) As n  
FROM gd_teacher_subject  
GROUP BY teacher  
HAVING COUNT(*) = 1
```

Группировки



Деление



Делимо
е

```
SELECT
  teacher,
  subject
FROM gd_teacher_subject
```

Делител
ь

```
SELECT
  subject
FROM my_favorite_subject
```

Частно
е

```
SELECT teacher
FROM gd_teacher_subject all_teachers,
     my_favorite_subject my_subjects
WHERE all_teachers.subject = my_subjects.subject
GROUP BY teacher
HAVING COUNT(*) = 2
```

```
SELECT teacher
FROM gd_teacher_subject all_teachers,
     my_favorite_subject my_subjects,
     (SELECT COUNT(*) n
      FROM my_favorite_subject) v_subject_count
WHERE all_teachers.subject = my_subjects.subject
GROUP BY teacher, v_subject_count.n
HAVING COUNT(*) = v_subject_count.n
```

Задачи для самостоятельного решения

1. Найдите все города на букву “К” в таблице `my_city`
2. Найдите город из 4 букв в таблице `my_city` (не использовать функцию длины)
3. Найдите город в названии, которого две буквы «о» в таблице `my_city`
4. Выведите город с наименьшим населением в 2000 году, а затем в 2010.
Используя таблицы: `my_city`, `city_population`

Проверим, что отложилось 😊





www.greendatasoft.ru