Задание 1

Выполним запрос, используя функцию to\_tsvector.



Результатом выполнения запроса будет являться следующая строка лексем



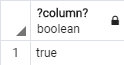
Изучив документацию, выясняется, что «да» не отображается в векторе, так как является «стоп-словом» - встречается слишком, чтобы поиск по ним был осмысленным.

Задание 2

Выполним седующие запросы.



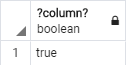
Результат:



Следующим запросом является:



Результат:



Последний запрос:



Результат:



Отвечая на вопросы:

1. Символ @@ является оператором соответствия для полнотекстового поиска.
2. Во втором запросе, в отличие от третьего используется множественное число слова fox, поэтому, при выполнении функции to\_tsvector, первый словарь,  который распознал фрагмент, выдаёт одну или несколько представляющих его лексем. Поэтому foxes превращается в fox, так как один из словарей понимает, что слово foxes — это слово fox во множественном числе. Слово foxhound - не является множественным числом, либо однокоренным co словом fox. Поэтому результатом третьего запроса было false.
3. Выполним запрос



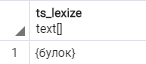
Результатом выполнения является следующее:



Для ответа на вопрос выполним следующий запрос:



Результатом которого будет:



Функция ts\_lexize возвращает массив лексем, если входной *фрагмент* известен словарю, либо пустой массив, если этот фрагмент считается в словаре стоп-словом.

Так же выполним следующий запрос:

.

Результат: 

Соответственно, при выполнении исходного запроса to\_tsvector распознает слово булок как лексему ‘булок’, а слово булка имеет лексему ‘булк’, именно из-за разницы лексем слово булка и не было найдено в запросе.

1. Выполним следующие запросы:



Результатом является следующее:



Выполним следующий запрос:



Результатом будет следующее:



При выполнении аналогичного запроса представленного в пункте 3, выясняется, что лексемой слова пирожки является “пирожк”, лексемой пирожок является “пирожок”, лексемой же слова пирожков является так же “пирожк”. Именно поэтому результатом первого запроса является true, а второго false.

3 Задание

1. Выполним следующие запросы:



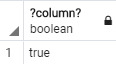
Результат:



Следующий запрос:



Результат:



Следующий запрос:



Результат:



Последний запрос:



Результат:



Выполним аналогичные запросы для предложения на русском языке



Результат:



Следующий запрос:



Результат:



Следующий результат:



Результат:



Последний запрос:



Результат:



1. Английский язык является языком по умолчанию, поэтому его не нужно указывать в первом аргументе. Если не указывать анализатор, то возьмётся язык по умолчанию.

4 Задание

Выполним следующий запрос:



Результатом выполнения запросов является:



1. Оператор <-> является оператором поиска фраз <-> (ПРЕДШЕСТВУЕТ). В нашем случае представлена вариация оператора ПРЕДШЕСТВУЕТ вида <*N*>, где *N* — целочисленная константа, задающая расстояние между двумя искомыми лексемами. Соответственно <2> обозначает расстояние между лексемами ‘мягких’ и ‘булок’.
2. Выполним запрос для поиска фразы:



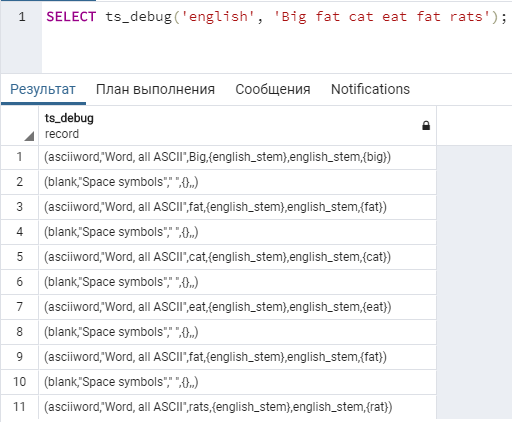
Результат:



1. Функция phraseto\_tsquery преобразует неформатированный текст запроса в значение tsquery. Она вставляет между оставшимися словами оператор <->. Стоп-слова не отбрасываются, а подсчитываются, и вместо операторов <-> используются операторы <*N*> с подсчитанным числом.

Задание 5

1. Функция ts\_debug - выводит информацию обо всех фрагментах данного документа, которые были выданы анализатором и обработаны настроенными словарями.



1. Функция ts\_headline принимает документ вместе с запросом и возвращает выдержку из документа, в которой выделяются слова из запроса.

