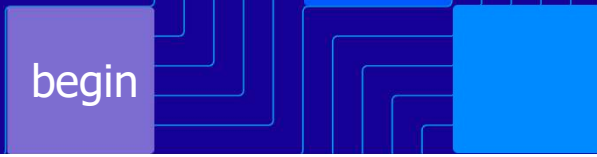
БЛОК 8. ИНДЕКСЫ

ПОЛНОТЕКСТОВЫЙ ПОИСК



ЦЕЛЬ

01

Познакомиться с понятием полнотекстового поиска

02

Понять задачи полнотекстового поиска

03

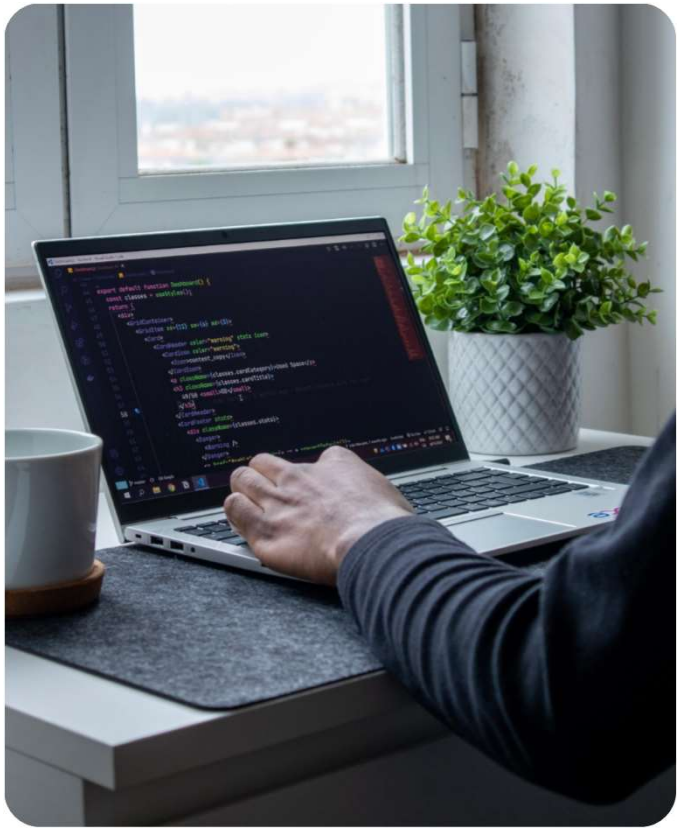
Узнать его реализацию и особенности

04

Научиться создавать индекс для полнотекстового поиска

СОДЕРЖАНИЕ УРОКА

1. Полнотекстовый поиск
2. Механизм работы
3. Практика



ПОЛНОТЕКСТОВЫЙ ПОИСК

Полнотекстовый поиск (или просто **поиск текста**) — это возможность находить **документы** на естественном языке, соответствующие **запросу**, и, возможно, дополнительно сортировать их по релевантности для этого запроса.

Наиболее распространённая задача — найти все документы, содержащие **слова запроса**, и выдать их отсортированными по степени **соответствия** запросу.

Предназначение - ранжирование слов и ускорение поиска по тексту

Например:

```
SELECT 'a fat cats sat on a mat and ate a fat rat'::tsvector @@
  'rat & cat'::tsquery;
```

https://postgrespro.ru/docs/postgresql/14/textsearch

МЕХАНИЗМ РАБОТЫ

Полнотекстовая индексация заключается в предварительной обработке документов и сохранении индекса для последующего быстрого поиска.

Предварительная обработка включает следующие операции:

1. **Разбор документов на фрагменты.** При этом полезно выделить различные классы фрагментов, например, числа, слова, словосочетания, почтовые адреса и т. д.
2. **Преобразование фрагментов в лексемы.** Лексема — это нормализованный фрагмент, в котором разные словоформы приведены к одной
3. **Хранение документов в форме, подготовленной для поиска.** Например, каждый документ может быть представлен в виде сортированного массива нормализованных лексем

ПРАКТИКА

Посмотрим различные варианты осуществления полнотекстового поиска

ИНДЕКСИРОВАНИЕ ПОЛНОТЕКСТОВОГО ПОИСКА

Существует заблуждение, что нужно создать отдельно поле для хранения лексем для полнотекстового поиска.

На самом деле у нас есть функциональный индекс и мы используем его:

CREATE INDEX idx ON test USING GIN (to_tsvector('english',col2));

И при поиске при использовании той же функции будет использован индекс!

select * from test where to_tsvector('english',col2) @@ to_tsquery('abs');

Важно! функции to_tsvector('english',col2)) и to_tsvector(col2) - разные!!!
Несмотря на значение языка по умолчанию english - количество аргументов разное!!!

ОСОБЕННОСТИ ПОЛНОТЕКСТОВОГО ПОИСКА

- Реализация для русского языка не самая лучшая, как мы уже увидели на практике - возможно есть смысл использовать для этого выгрузку в ElasticSearch (https://www.elastic.co/elasticsearch/)
- Для английских языков есть несколько словарей, можно с ними поэкспериментировать для более лучшего распознавания https://postgrespro.ru/docs/postgresql/14/textsearch-dictionaries
- Важно для индексов - разное количество аргументов - вызываться будут разные функции

Дополнительный материал для изучения:

http://www.sai.msu.su/~megera/postgres/talks/fts_pgsql_intro.html
https://habr.com/ru/post/442170/

ПОДВЕДЕНИЕ ИТОГОВ

ИТОГИ ЗАНЯТИЯ

01
Поняли задачи полнотекстового поиска

02
Узнали его реализацию и особенности

03
Создали индекс для полнотекстового поиска

ЗАДАНИЕ ДЛЯ САМОПРОВЕРКИ

Цель задания:
Реализовать полнотекстовый поиск

Пошаговый план выполнения:

1. Создать таблицу 100 000 строк текста используя фидл https://www.db-fiddle.com/f/gW1N26Cht89J5ZezCff4dL/4
2. Создать полнотекстовый индекс по этому текстовому полю
3. Проверить, что при поиске используется индекс
4. Если не получилось, эталонное решение закомментировано в фидле

Задание закончено

СПАСИБО!

На следующем занятии мы рассмотрим тему:

- Статистика

end