

Лабораторная работа 4. Функции и язык PL/pgSQL

Постановка задачи

Лабораторная работа посвящена упрощению работы аналитика с помощью создания и использования функций. При выполнении задания необходимо:

- Составить SQL-скрипты для создания нескольких функций, упрощающих манипуляции с данными.
- Продемонстрировать полученные знания о возможностях языка PL/pgSQL. В скриптах должны использоваться:
 - Циклы.
 - Ветвления.
 - Переменные.
 - Курсоры.
 - Исключения.
- Обосновать преимущества механизма функций перед механизмом представлений.

Темы для проработки

- Функции.
<https://postgrespro.ru/docs/postgrespro/11/xfunc-sql>
- PL/PgSQL
<https://postgrespro.ru/docs/postgrespro/11/plpgsql>
- Основные операторы
<https://postgrespro.ru/docs/postgrespro/11/plpgsql-statements>
- Управляющие структуры
<https://postgrespro.ru/docs/postgrespro/11/plpgsql-control-structures>
- Курсоры
<https://postgrespro.ru/docs/postgrespro/11/plpgsql-cursors>
- Полное описание синтаксиса встретившихся команд
<https://postgrespro.ru/docs/postgrespro/11/sql-commands>

Примеры вопросов

- Объяснить, как работают написанные запросы.
- Исправить неверно работающий запрос (запросы).
- Упростить один или несколько запросов.
- Написать или модифицировать запрос по сформулированному заданию.
- Описать в каких случаях целесообразно создавать функции.
- Рассказать о курсорах, как и зачем используются.
- Рассказать о работе с циклами

Индексы

Постановка задачи

практическое задание посвящено ускорению выполнения запросов. Для этого могут быть использованы механизмы секционирования, наследования и индексов. Для выполнения задания необходим достаточно большой объем данных, чтобы оптимизация была целесообразной (рядка 1 млн. строк в каждой таблице). Необходимо подготовить два запроса:

- Запрос к одной таблице, содержащий фильтрацию по нескольким полям.
- Запрос к нескольким связанным таблицам, содержащий фильтрацию по нескольким полям.

Для каждого из этих запросов необходимо провести следующие шаги:

- Получить план выполнения запроса без использования индексов.
- Получить статистику (IO и Time) выполнения запроса без использования индексов.
- Создать нужные индексы, позволяющие ускорить запрос.
- Получить план выполнения запроса с использованием индексов и сравнить с первоначальным
- Получить статистику выполнения запроса с использованием индексов и сравнить с первоначальной
- Оценить эффективность выполнения оптимизированного запроса.

Также необходимо оценить полезность индексов для организации полнотекстового поиска, фильтрации с использованием массива и json формата.

Для таблицы объемом больше 100 млн. записей произвести оптимизацию, позволяющую быстро удалять старые данные, ускорить вставку и чтение данных.

Темы для проработки

- SEO @

° V ° O --

† ° #yyU 7 k ou° uou#o

лнотекстовый поиск

Наследование таблиц

Секционирование таблиц

Полное описание синтаксиса встретившихся команд

Примеры вопросов

- В ем а и

- В каких случаях имеет смысл создавать индексы? Какие колонки следует включать в индекс и почему?
- Какие существуют способы внутренней организации индексов?
- Рассказать о проблеме фрагментации индексов. Как бороться с фрагментацией?
- Имеет ли значение порядок указания колонок при создании индекса?
- В чем разница между Index Scan и Index Seek?
- В чем разница между секционированием и наследованием?
- Зачем нужен ANALYZE?
- Исправить ошибки в подготовленных выборках.
- Могут ли индексы ухудшить производительность? Если да, то продемонстрировать это.
- На что влияет порядок сортировки (ASC\DESC) при создании индекса? Продemonстрировать это.
- Продemonстрировать полезность индекса по выражению.
- Продemonстрировать полезность частичного индекса.