

Лаб№6 Функции, Хранимые Процедуры и Триггеры 2023

Оформление отчета:

1. Сформулировать на русском языке (в виде текста) описание объектов вашей БД в соответствии с вашим вариантом задания.
2. Написать код создающий данные объекты БД на языке pgsql или SQL.
3. Приложить результаты выполнения функций, процедур, триггеров в виде картинки.

Задания:

Функции и Хранимые процедуры

1. Создать функцию **Period(..., ...)** с двумя входными параметрами типа date, которая выберет строки из дочерней таблицы в диапазоне дат, указанных первым и вторым аргументами при вызове функции **Period(..., ...)**.
2. Создать функцию **Sum_object(...)**(с одним параметром), которая возвращает список имен **объектов** из родительской таблицы на основании данных дочерней таблицы. Список **объектов** определяется значением параметра, исходя из условия, что суммарное количество **объектов** должно быть больше, чем заданное значение в параметре.
3. Создать функцию **row_count(...)**, которая подсчитывает количество строк дочерней таблицы, даты которых находятся между параметрами date_from и date_to.
4. Создать хранимую процедуру **object_stat(...)**, которая подсчитывает минимальное, максимальное и среднее значение **объектов** в дочерней таблице, входным параметром является **имя объекта**.
5. Создать хранимую процедуру **objects_stat(...)**, которая подсчитывает минимальное, максимальное и среднее значение каждого объекта в дочерней таблице и выводит имя объекта, входным параметром является имя объекта из родительской таблицы.
6. Создать хранимую процедуру **Itog(...)** с одним входным параметром, которая выводит наименование **объекта** по суммарному количеству **объектов**:
 - а) Оценка «Незначительный объект», если число объектов меньше 2
 - б) Оценка «Обычный объект», если число объектов больше 2 и меньше или равно 3
 - с) Оценка «Значительный объект», если число объектов больше 3

Триггеры

7. Создать триггер **After_Delete**, который при удалении записи из родительской таблицы удалял бы все связанные записи из дочерней таблицы. Показать результат работы триггера.
8. Создать триггер **Before_Delete**, который при удалении записи из дочерней таблицы выводил бы имя **объекта** родительской таблицы. Показать результат работы триггера.
9. Создать триггер **ins_sum**, который связывает триггер с таблицей для инструкций INSERT. Это действует как сумматор, чтобы суммировать значения, вставленные в один из столбцов дочерней таблицы. Триггер должен активироваться перед каждой строкой, вставленной в таблицу. Показать результат работы триггера.
10. Создать триггер **Before_Update_Value** на событие UPDATE, который увеличивает значение числового поля дочерней таблицы на 10%. Показать результат работы триггера.