

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
(Университет ИТМО)

Факультет программной инженерии и компьютерной техники

Отчет по лабораторной работе №5

по дисциплине « База данных »

Тема: Data Mining на основе модели CRISP-DM

Выполнила:

студентка гр.№Р33212 Ян Цзяфэн

Санкт-Петербург

2020

Задание

Для выполнения лабораторной работы №5 необходимо:

- Добавить в ранее созданную базу данных (лр №4) триггеры для обеспечения комплексных ограничений целостности.
- Реализовать функции и процедуры на основе описания бизнес-процессов, определенных при описании предметной области (лр №1). Должна быть обеспечена проверка корректности вводимых данных для созданных функций и процедур.
- Необходимо произвести анализ использования созданной базы данных, выявить наиболее часто используемые объекты базы данных, виды запросов к ним. Результаты должны быть представлены в виде текстового описания.
- На основании полученного описания требуется создать подходящие индексы и доказать, что они будут полезны для представленных в описании случаев использования базы данных.

Отчёт по лабораторной работе должен содержать:

- титульный лист;
- текст задания;
- код триггеров, функций, процедур;
- описание наиболее часто используемых сценариев при работе с базой данных;
- описание индексов и обоснование их использования;
- выводы по работе;

Темы для подготовки к защите лабораторной работы:

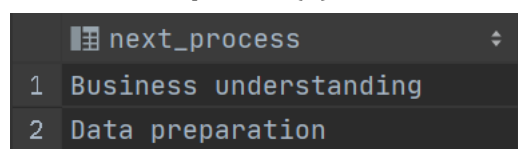
- PL/pgSQL
- процедуры, функции
- триггеры
- индексы

Код создания функции и триггеров

--Вывод названия следующего процесса по процессам

```
CREATE OR REPLACE FUNCTION next_process (id integer)
RETURNS SETOF VARCHAR(32) AS $$
    SELECT process_name FROM process WHERE process_id IN (
        SELECT next_id FROM next WHERE process_id = id);
$$ LANGUAGE sql;
```

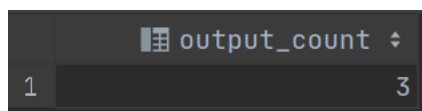
```
SELECT next_process(2);
```



--Сколько результатов мы должны получить в данной задаче?

```
CREATE OR REPLACE FUNCTION output_count(id integer)
RETURNS INTEGER AS $$
DECLARE
    i INTEGER;
BEGIN
    SELECT DISTINCT count(output_name) into i FROM output o WHERE o.output_id IN (
        SELECT output_id FROM activity WHERE task_id = id);
    RETURN i;
end;
$$ LANGUAGE plpgsql;
```

```
SELECT output_count(1);
```



output_count
3

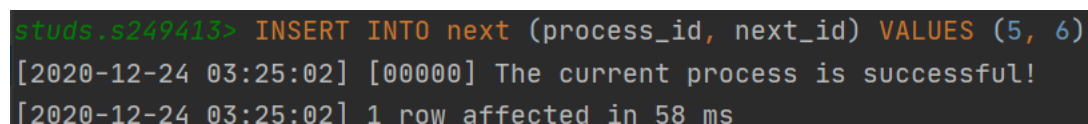
--Функция триггера для объяснения состояния процесса.

```
CREATE OR REPLACE FUNCTION process_state()
RETURNS trigger AS $$
BEGIN
    if ((SELECT next_id FROM next WHERE process_id=NEW.process_id)<NEW.process_id)
        then
            RAISE NOTICE 'The current process has failed.';
        else
            RAISE NOTICE 'The current process is successful!';
        end if;
    RETURN NEW;
end;
$$ LANGUAGE plpgsql;
```

--Триггера для объяснения состояния процесса.

```
CREATE TRIGGER process_state_trigger AFTER INSERT ON next
FOR EACH ROW EXECUTE PROCEDURE process_state();
```

```
INSERT INTO next (process_id, next_id) VALUES (5, 6);
```



```
studs.s249413> INSERT INTO next (process_id, next_id) VALUES (5, 6)
[2020-12-24 03:25:02] [00000] The current process is successful!
[2020-12-24 03:25:02] 1 row affected in 58 ms
```

Описание часто используемых сценариев

Основной действующей сущностью в данной базе данных являются процесс. В работе с процессами наиболее частым запросом является вывод списка следующих процессов, задач и результатов текущего процесса. Для этого были созданы два индекса.

Описание индексов

```
CREATE INDEX output_index ON output(output_id);  
CREATE INDEX task_index ON task(process_id);
```

Выводы по работе

В данной работе были реализованы функции, процедуры и триггеры для работы с базой данных. Также были выбраны и описаны необходимые индексы.