Лабораторная работа № 6

студента группы ИТз-221

Дмитриева Дмитрия Анатольевича

*Выполнение:\_\_\_\_\_\_\_\_\_ Защита:\_\_\_\_\_\_\_\_\_\_\_*

PL/pgSQL функции

*Цель работы***:** изучить PL/pgSQL функции, триггеры, обработку ошибок, условия, вывод ошибок и сообщений.

**Ход работы:**

***Вариант 8***

1. Ознакомился с функционалом PL/pgSQL функций, триггеров, обработок ошибок, условий и выводом ошибок и сообщений.
2. Создал набор процедур с входными и выходными параметрами и процедуры, возвращающие набор записей. Первая процедура добавляет бронь для клиента в определенный номер, заполняя дату заселения, а также выставляя администратора (рис. 1).



Рисунок 1 – Процедура добавления брони

1. Протестировал работу процедуры (рис. 2,3)

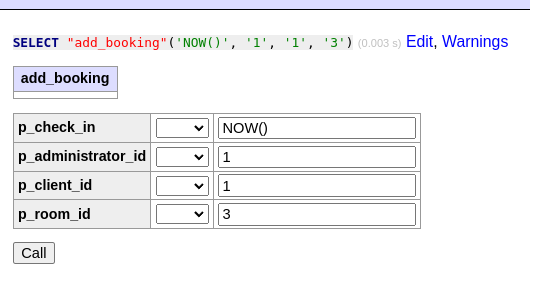


Рисунок 2 – Вызов процедуры

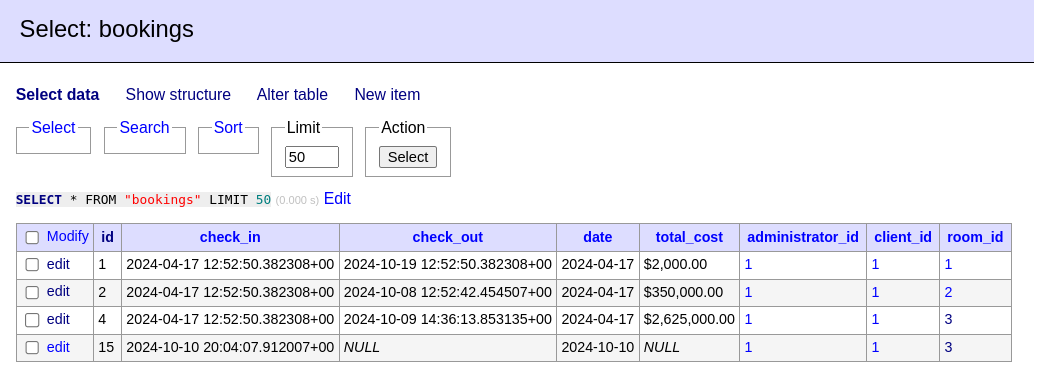


Рисунок 3 – Добавленная бронь

1. Вторая процедура получает выборку броней по переданному идентификатору клиента (рис. 4)

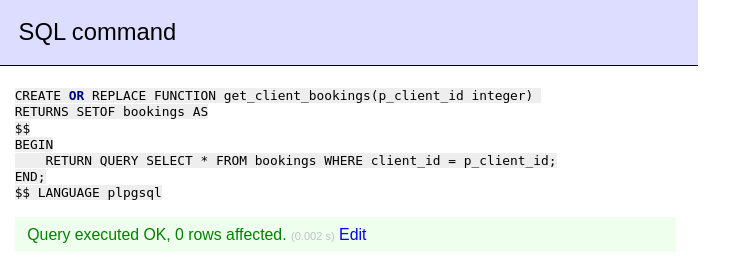


Рисунок 4 – Процедура получения брони

1. Результат работы процедуры (рис. 5)

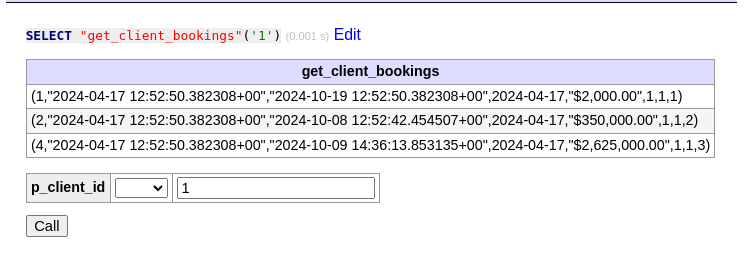


Рисунок 5 – Процедура получения брони

1. Создал INSERT, DELETE и UPDATE триггеры для таблиц разработанной БД. Первый триггер направлен на проверку присутствия передаваемого клиента и номера в базе данных (рис. 6). Данный триггер выполняется перед вставкой и выводит сообщение в случае отсутствия данных (рис. 7).

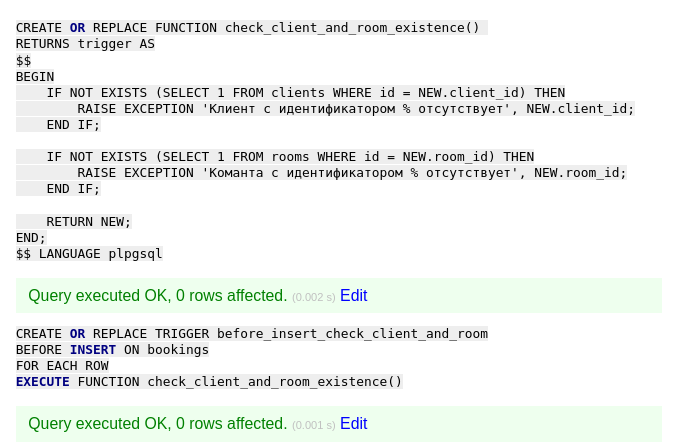


Рисунок 6 – Триггер проверки клиента и номера

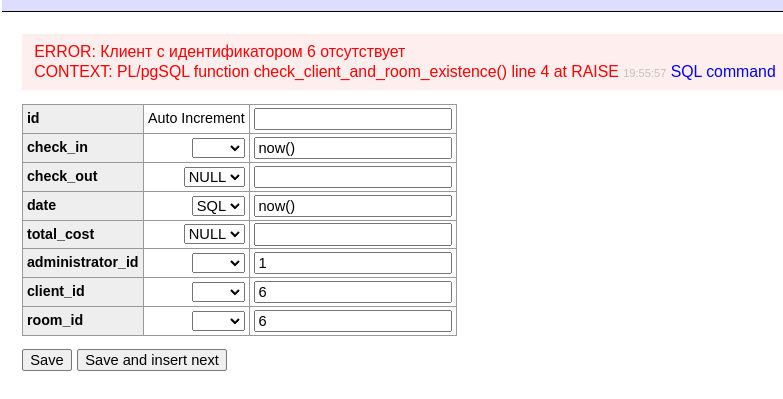


Рисунок 7 – Ошибка при вставке брони

1. Второй триггер направлен на проверку гражданства, а также паспорта при обновлении данных пользователя (рис. 8). В случае если гражданство не соответствует допустимым или паспорт не валиден, выводится ошибка. Триггер выполняется перед обновлением, чтобы не допустить обновления данных (рис. 9)

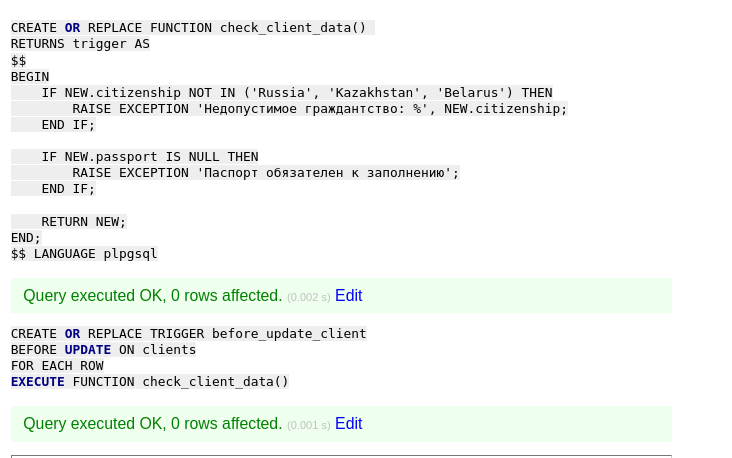


Рисунок 8 – Триггер проверки гражданства и паспорта

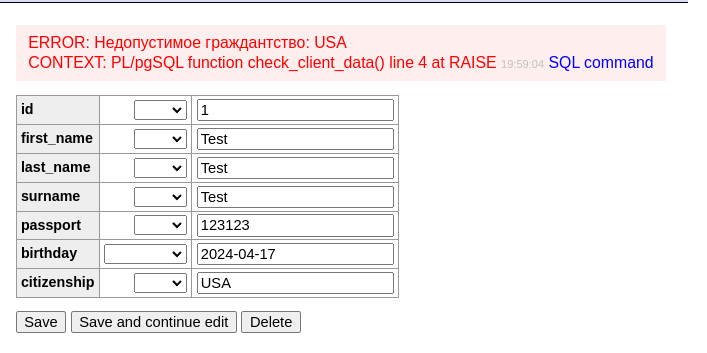


Рисунок 9 – Ошибка при обновлении клиента

1. Третий триггер направлен на проверку брони определенного номера (рис. 10). В случае если номер уже забронирован каким-либо клиентом, его нельзя удалить. Триггер выполняется перед удалением записи, чтобы не допустить нарушения согласованности базы данных (рис. 11)

 Рисунок 10 – Триггер удаления номера



Рисунок 11 – Ошибка при удалении номера, который уже забронирован

**Контрольные вопросы:**

1. Типы функций, поддерживаемые PostgreSQL:

- Функции SQL (определяются SQL-запросом)

- Функции на PL/pgSQL (процедурный язык PostgreSQL)

2. Хранимые процедуры — это программные объекты на сервере базы данных, которые содержат набор SQL-операторов и могут выполняться как единое целое. Они могут включать логику, циклы, условные операторы и работать с данными.

3. Хранимые процедуры работают быстрее, чем отдельные SQL-операторы, так как они заранее компилируются и сохраняются на сервере. Это позволяет избежать повторной компиляции и оптимизации при каждом вызове процедуры.

4. Хранимые процедуры используются для автоматизации и объединения сложных последовательностей операций с базой данных, таких как обработка транзакций, валидация данных, обеспечение целостности данных, выполнение повторяющихся задач.

5. Из хранимой процедуры можно вернуть значение с помощью оператора RETURN. Значения могут быть различных типов, таких как скалярные (integer, text, boolean), наборы строк.

6. Входные параметры в хранимую процедуру можно передавать в виде аргументов при ее вызове. Параметры могут быть различных типов данных, таких как integer, text, boolean.

7. Триггеры — это специальные объекты базы данных, которые автоматически выполняют заданную процедуру (триггерную функцию) при определенных событиях (вставка, обновление или удаление строки) в таблице или представлении.

8. Типы триггеров:

- BEFORE: срабатывает до выполнения команды (вставки, обновления, удаления).

- AFTER: срабатывает после выполнения команды.

9. Триггеры создаются с помощью команды CREATE TRIGGER и связаны с триггерной функцией. Когда происходит событие (вставка, удаление, обновление), триггер автоматически вызывает связанную функцию, которая может изменить данные, провести валидацию или выполнить другие действия.

**Вывод:** изучил PL/pgSQL функции, триггеры, обработку ошибок, условия, вывод ошибок и сообщений.