|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра цифровой трансформации (ЦТ)**

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ**

по дисциплине «Разработка баз данных»

**Практическое занятие № 5**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студенты группы | *ИКБО-50-23 Враженко Д.О.*  *.* | (подпись) | |  |
| Преподаватель | *Мажей Я.В.* | (подпись) | |  |
| Отчет представлен | «\_\_\_»\_\_\_\_\_\_\_\_2025 г. | |  | |

Москва 2025 г.

**Цель работы**:

Работа направлена на формирование у студентов углубленных навыков работы с объектами баз данных в СУБД PostgreSQL, смещая акцент от прямого манипулирования данными к созданию переиспользуемых логических конструкций.

**Постановка задачи**:

**Предварительное задание**: в начале работы необходимо добавить скриншоты **всех используемых таблиц**.

**Задание №1: создание модифицируемого представления**

Для вашей базы данных создать **простое модифицируемое представление**, которое отбирает строки из одной таблицы по определенному критерию.

*Например, для БД «Аптека» можно создать представление, отображающее лекарства только от одного производителя «Bayer AG»).*

**Задание №2: модификация данных через представление**

Продемонстрировать возможность изменения данных в базовой таблице через представление, созданное в Задании №1. Для этого необходимо выполнить **два запроса**:

1. **Добавить** новую запись с помощью оператора **INSERT**.
2. **Удалить** существующую запись с помощью оператора **DELETE**.

**Задание №3: создание немодифицируемого аналитического представления**

Для вашей базы данных создать единое **немодифицируемое представление** для аналитических целей.

Представление должно объединять данные как минимум **из двух таблиц** и содержать **агрегирующие функции** (**COUNT**, **SUM**, **AVG** и т.д.) и группировку (**GROUP BY**).

*Например, для БД «Аптека» такое представление могло бы для каждого производителя выводить общее количество наименований лекарств и их среднюю цену.*

**Задание №4: использование аналитического представления в запросах**

Написать **SELECT**-запрос, который использует созданное в **Задании №3** аналитическое представление в качестве источника данных для дальнейшей фильтрации или анализа.

*Например, можно отобрать производителей, у которых средняя цена на продукцию превышает определенное значение.*

**Задание №5: создание и обновление материализованного представления**

1. Создать **материализованное представление** для ускорения выполнения ресурсоемкого аналитического запроса.  
   *Например, для БД «Аптека» можно создать представление, которое заранее рассчитывает общую сумму продаж для каждого покупателя.*
2. **Продемонстрировать** процесс обновления данных в представлении с помощью команды REFRESH MATERIALIZED VIEW viewName;.

**Задание №6: разработка пользовательской функции для аналитических вычислений**

1. **Разработать** пользовательскую функцию, которая инкапсулирует комплексный аналитический расчет. Функция должна принимать на вход идентификатор *(например, manufacturer\_id)* и возвращать одно скалярное значение *(например, общую сумму продаж продукции данного производителя)*, вычисленное на основе соединения нескольких таблиц и применения агрегатных функций.
2. **Продемонстрировать** вызов функции в составе SELECT-запроса.

**Задание №7: разработка хранимой процедуры для выполнения сложной операции**

Разработайте хранимую процедуру, которая выполняет безопасную операцию по изменению данных. Процедура должна принимать на вход ID какой-либо записи и числовое значение *(например, количество)*.

Внутри процедуры необходимо проверить, достаточно ли текущего значения в числовом поле одной таблицы для выполнения операции.

* Если **да** – уменьшите это значение и добавьте новую запись в другую, связанную таблицу.
* Если **нет** – операция должна полностью прерваться, не внося никаких изменений в данные.

Для сообщения о результате используйте выходной параметр, который вернёт статус успеха или неудачи.

**Задание №8: демонстрация вызова хранимой процедуры**

Привести **два примера вызова процедуры**, созданной в Задании №7:

* **Успешный вызов**, который добавляет в вашу базу данных уникальную запись.
* **Неудачный вызов**, который демонстрирует срабатывание реализованной проверки целостности и возврат пользовательской ошибки.

**Каждый** SQL-запрос **сопроводить комментарием**, объясняющим его назначение и логику работы с учетом специфики вашей базы данных.

**ХОД ВЫПОЛНЕНИЯ РАБОТЫ**

Все используемые таблицы индивидуальной схемы данных:

Таблица 1. Таблица employee (Сотрудник)

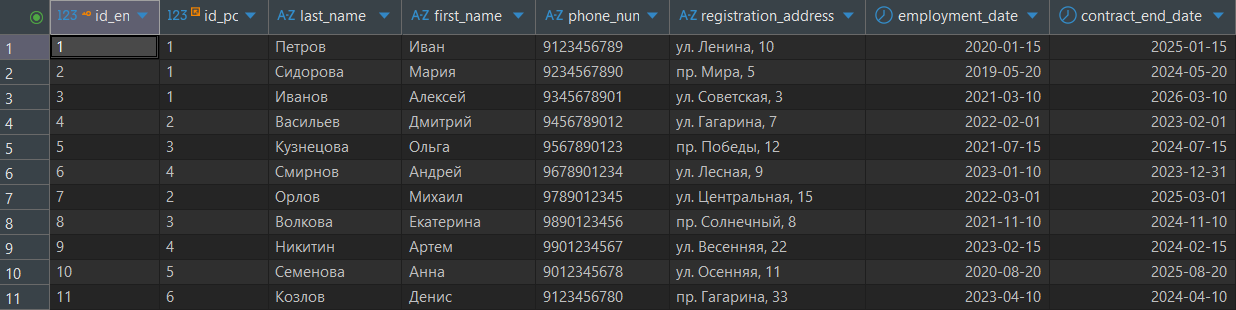


Таблица 2. Таблица position (Должность)

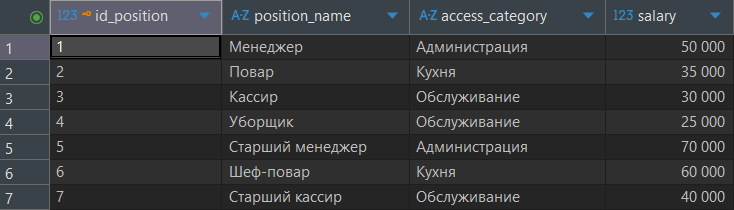


Таблица 3. Таблица employee\_inventory (Связующая таблица Сотрудник-Инвентарь)

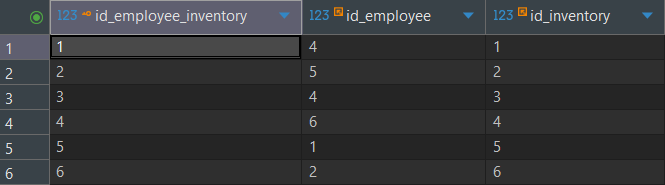


Таблица 4. Таблица inventory (Инвентарь)

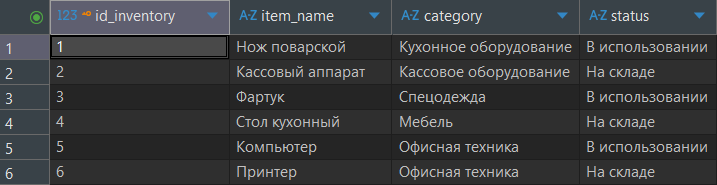


Таблица 5. Таблица termination\_request (Запрос на увольнение)

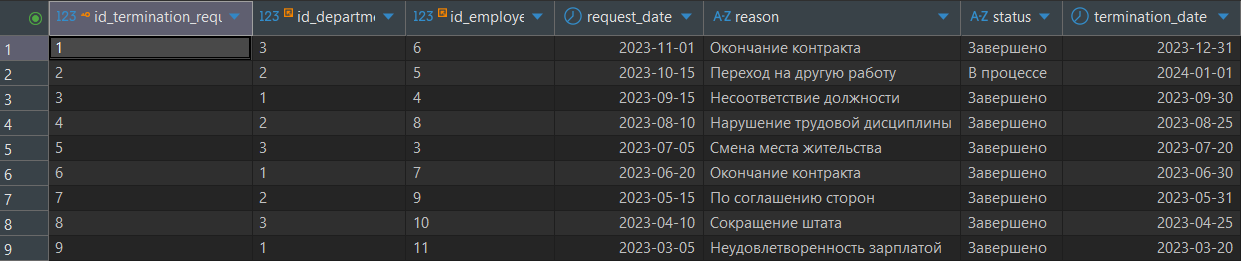


Таблица 6. Таблица department (Отдел)

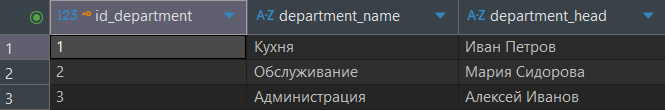


Таблица 7. Таблица payment (Выплата)

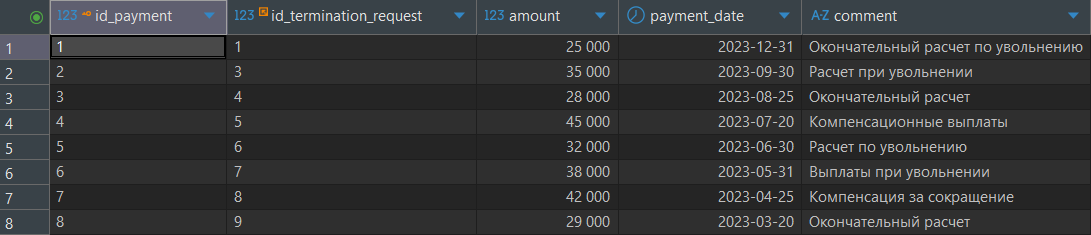


Таблица 8. Таблица document (Архив документов)

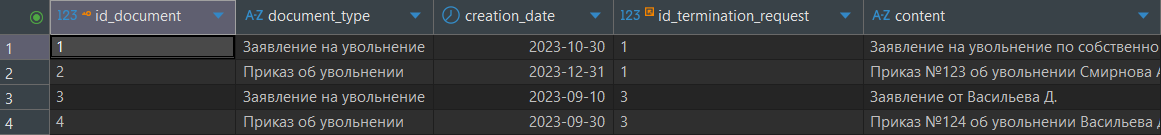


Таблица 9. Таблица request\_termination\_type (Связующая таблица Запрос на увольнение-Вид увольнения)

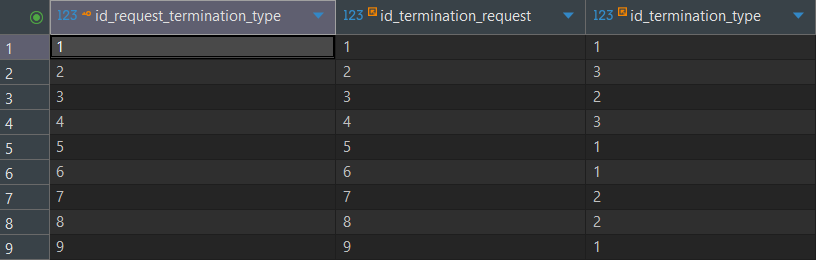
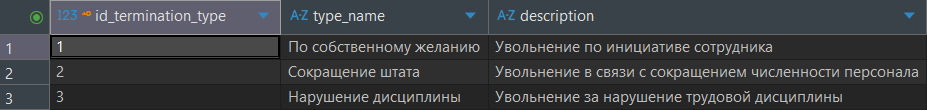


Таблица 10. Таблица termination\_type (Вид увольнения)



# Задание №1: создание модифицируемого представления

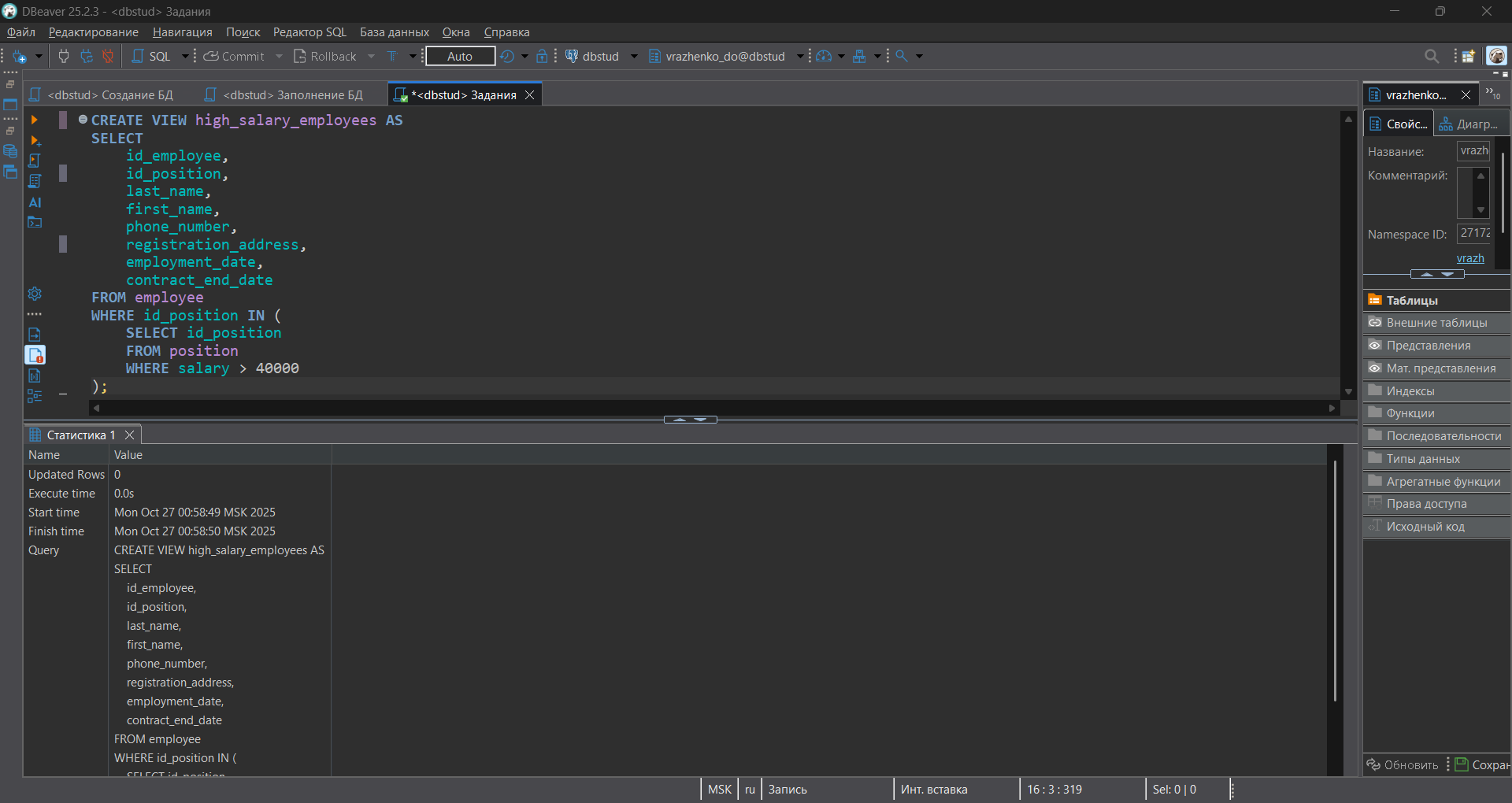


Рисунок 1 − Создание модифицируемого представления

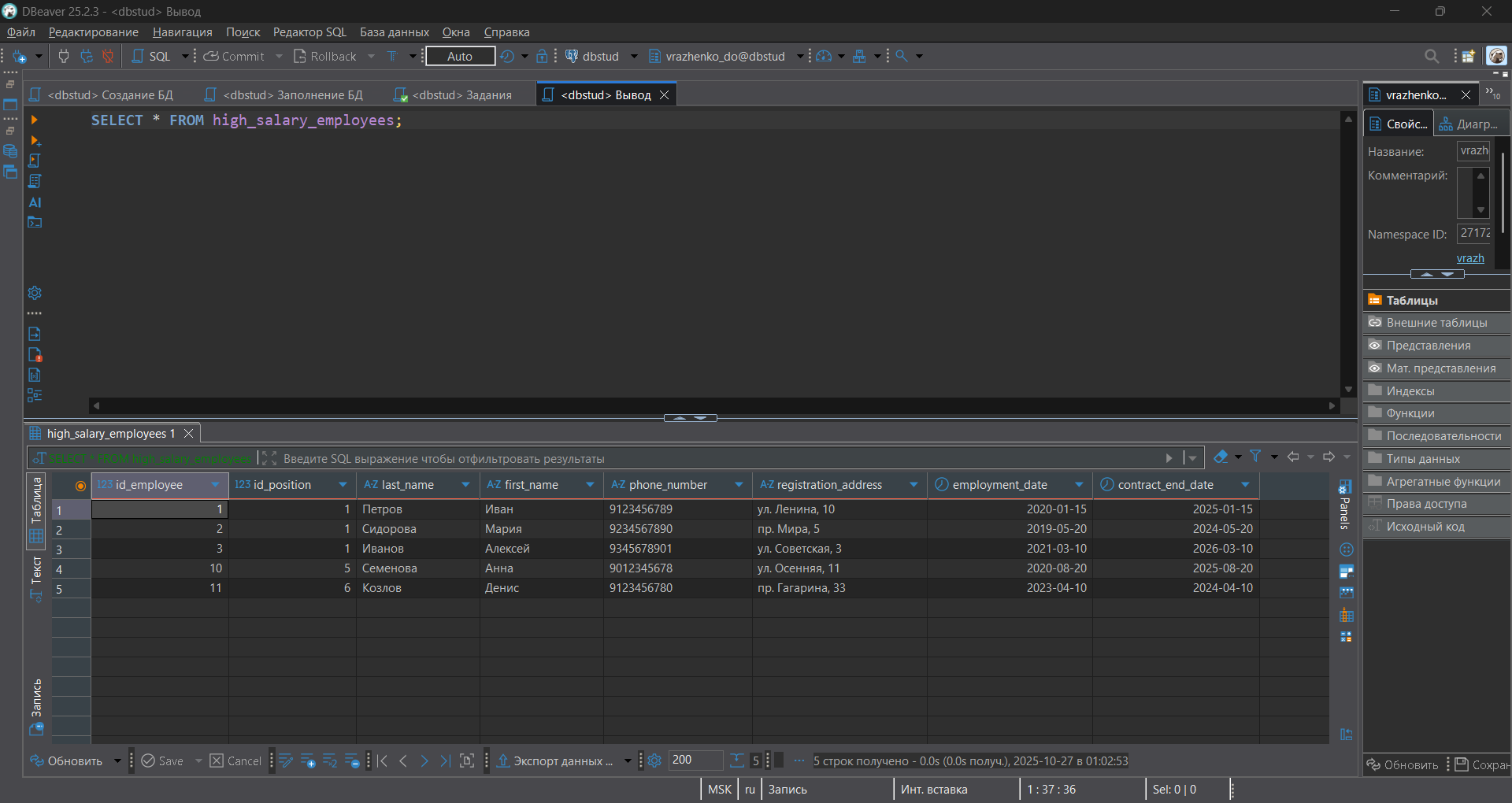


Рисунок 2 − Результат выполнения

# Задание №2: модификация данных через представление

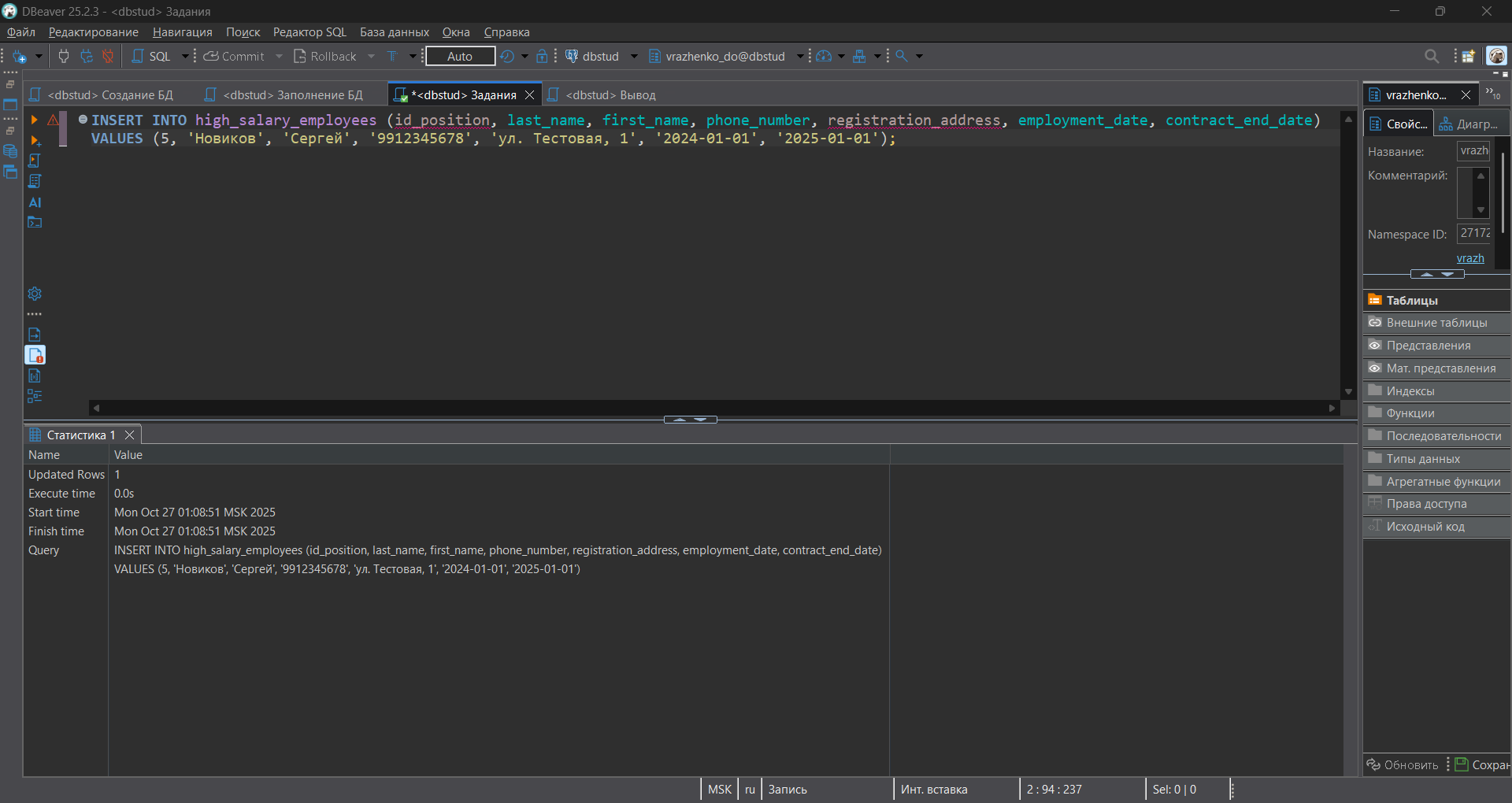


Рисунок 3 − Выполнение INSERT

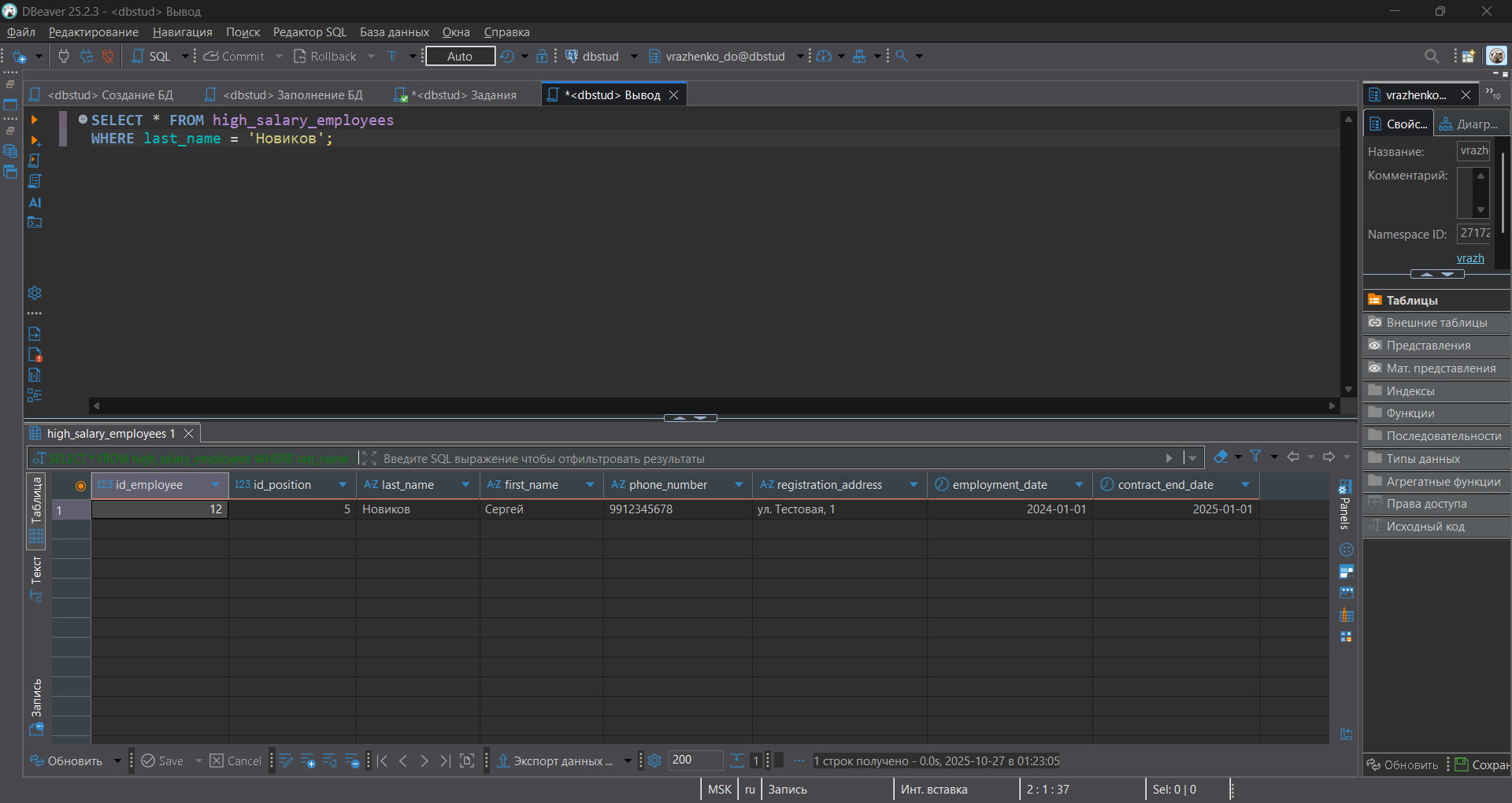


Рисунок 4 − Подтверждение добавления

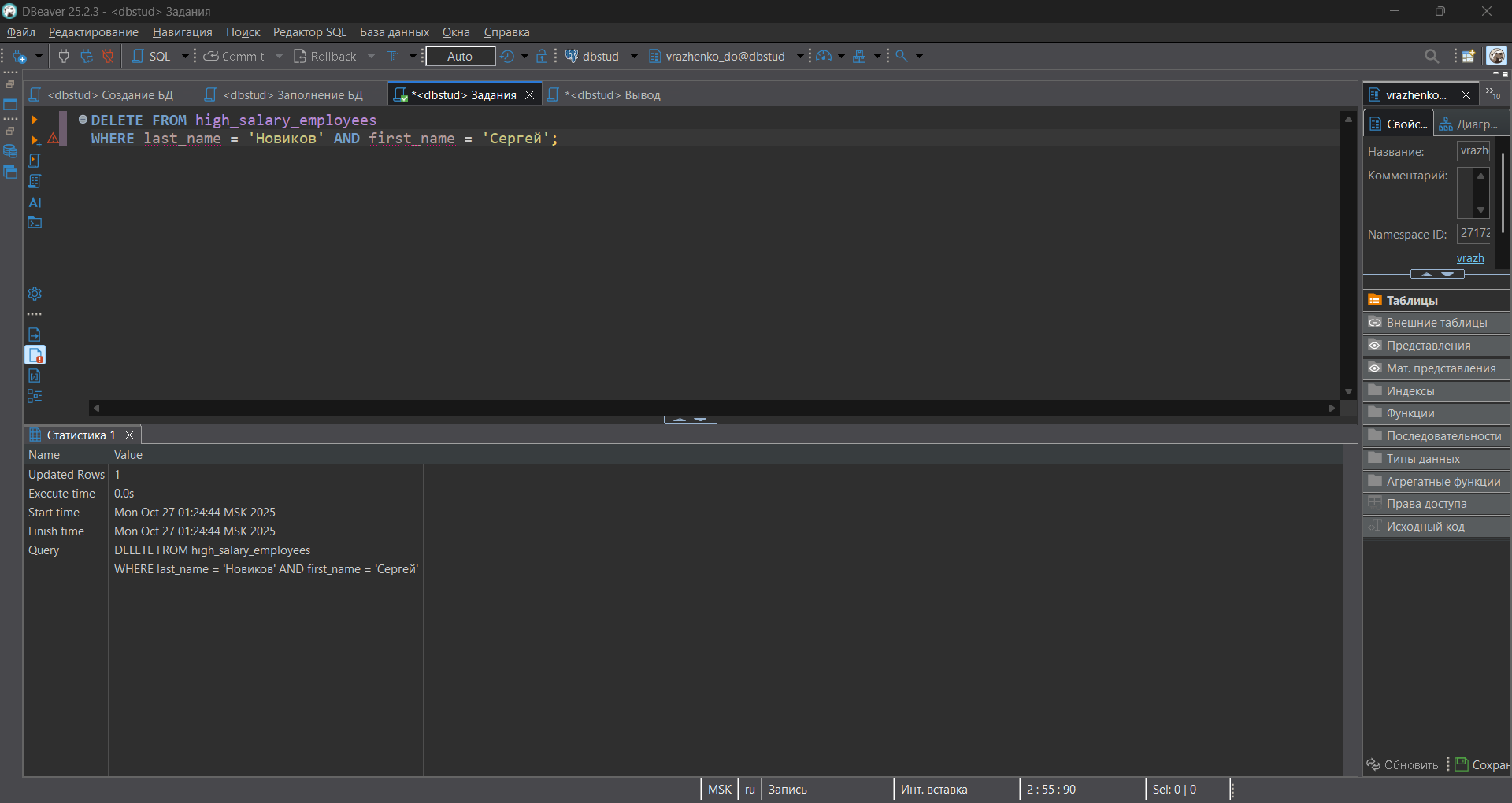


Рисунок 5 − Выполнение DELETE

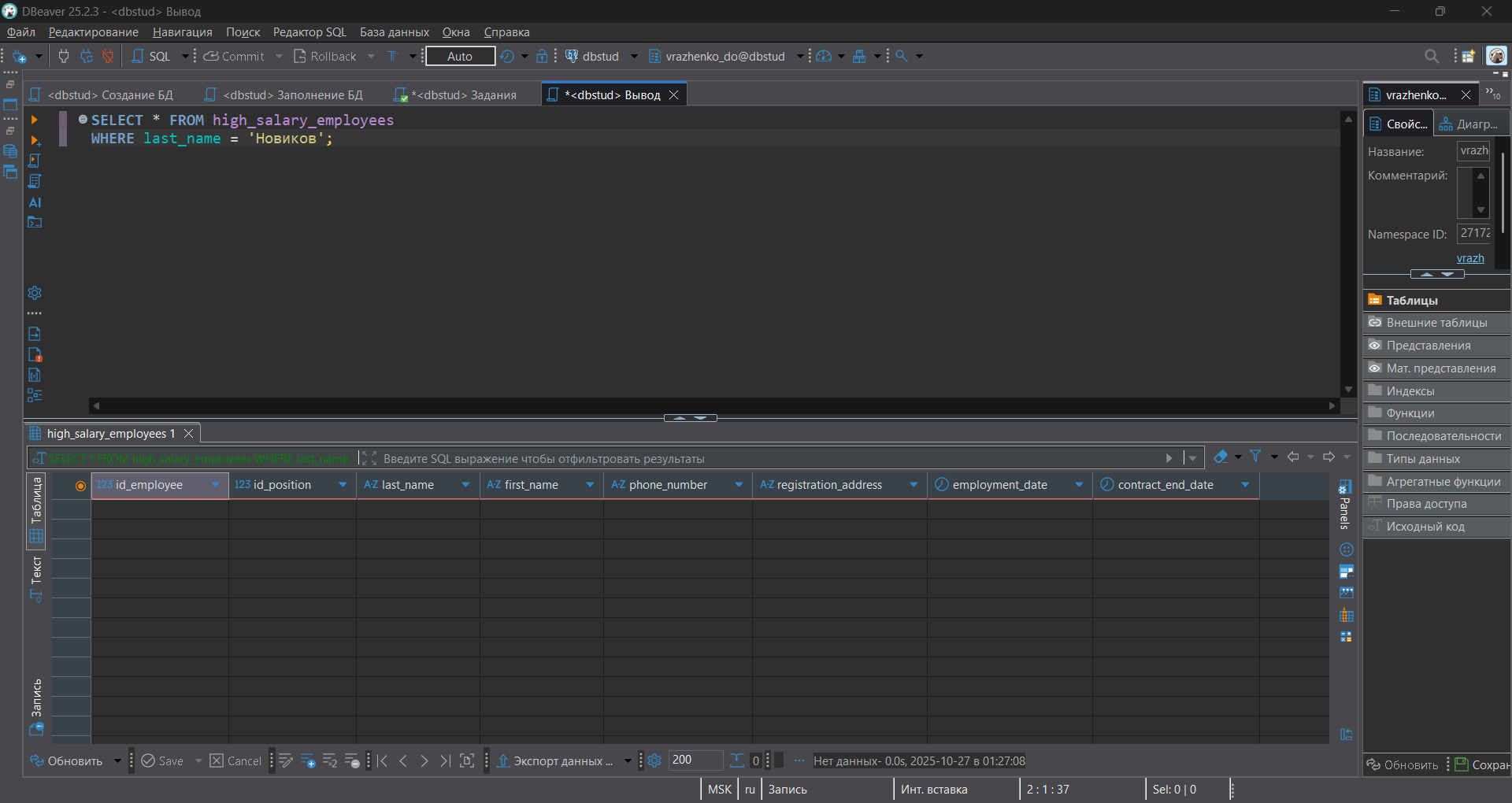


Рисунок 6 − Подтверждение удаления

# Задание №3: создание немодифицируемого аналитического представления

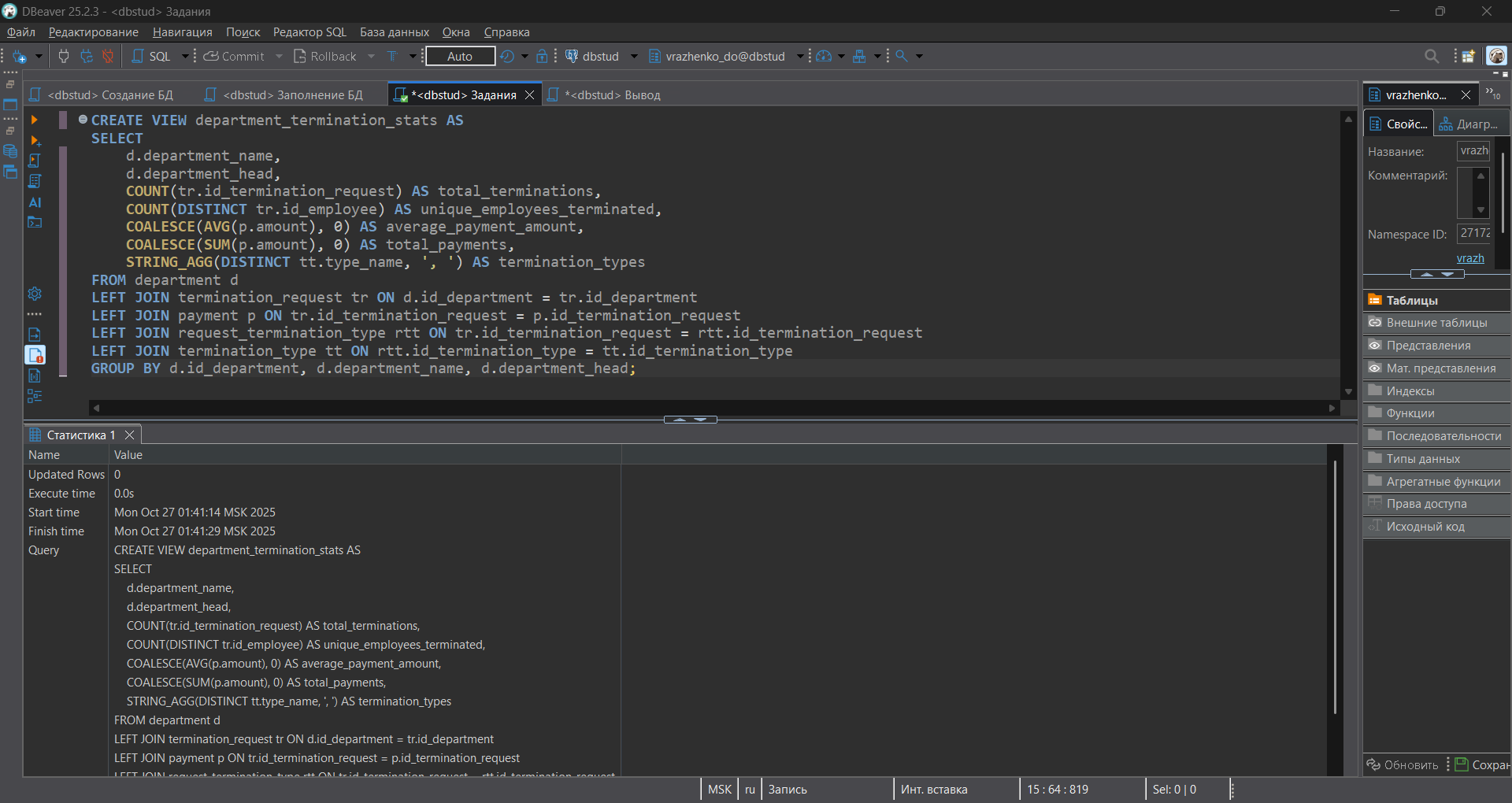


Рисунок 7 − Создание немодифицируемого представления

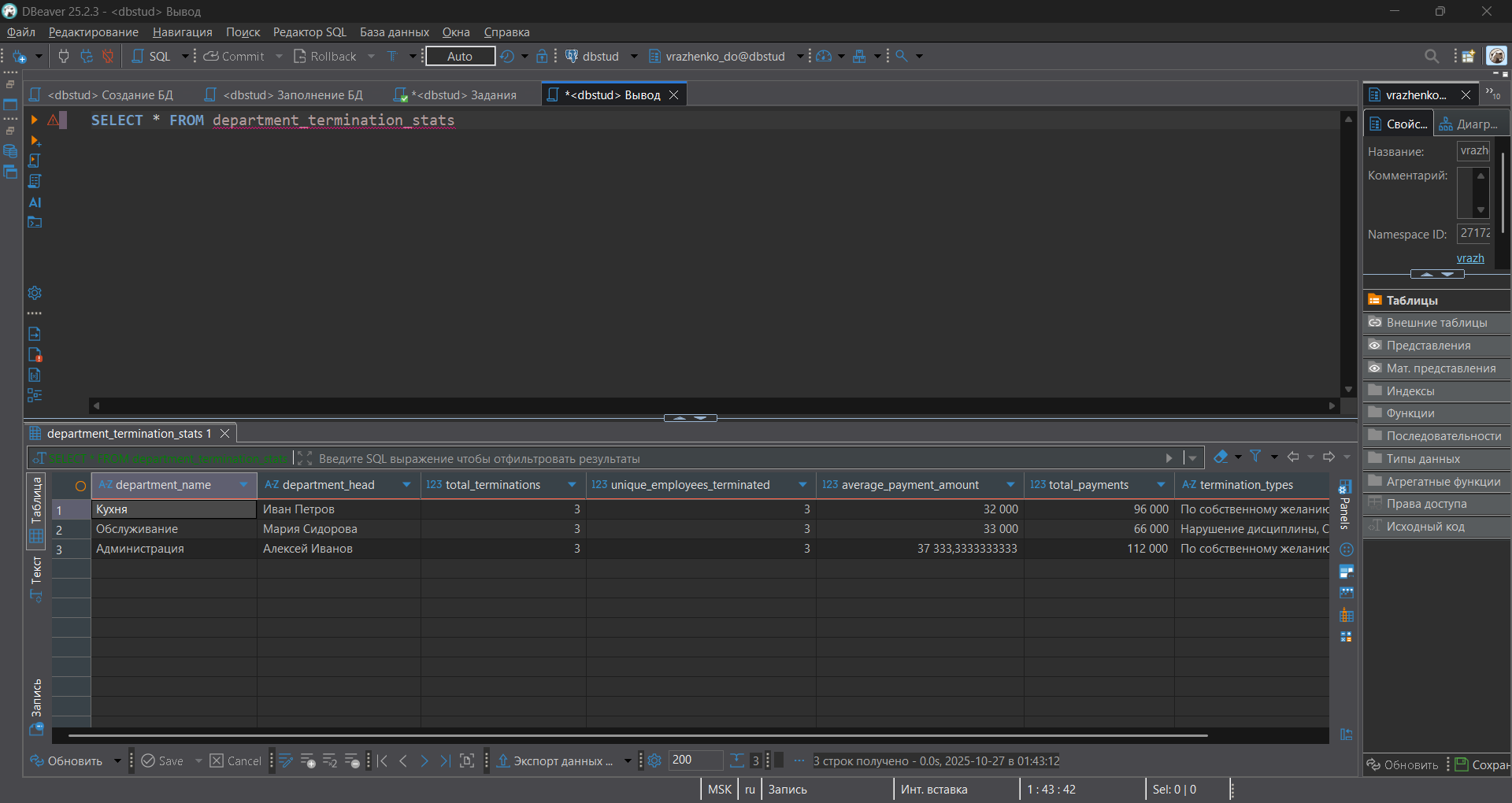


Рисунок 8 − Результат выполнения

# Задание №4: использование аналитического представления в запросах

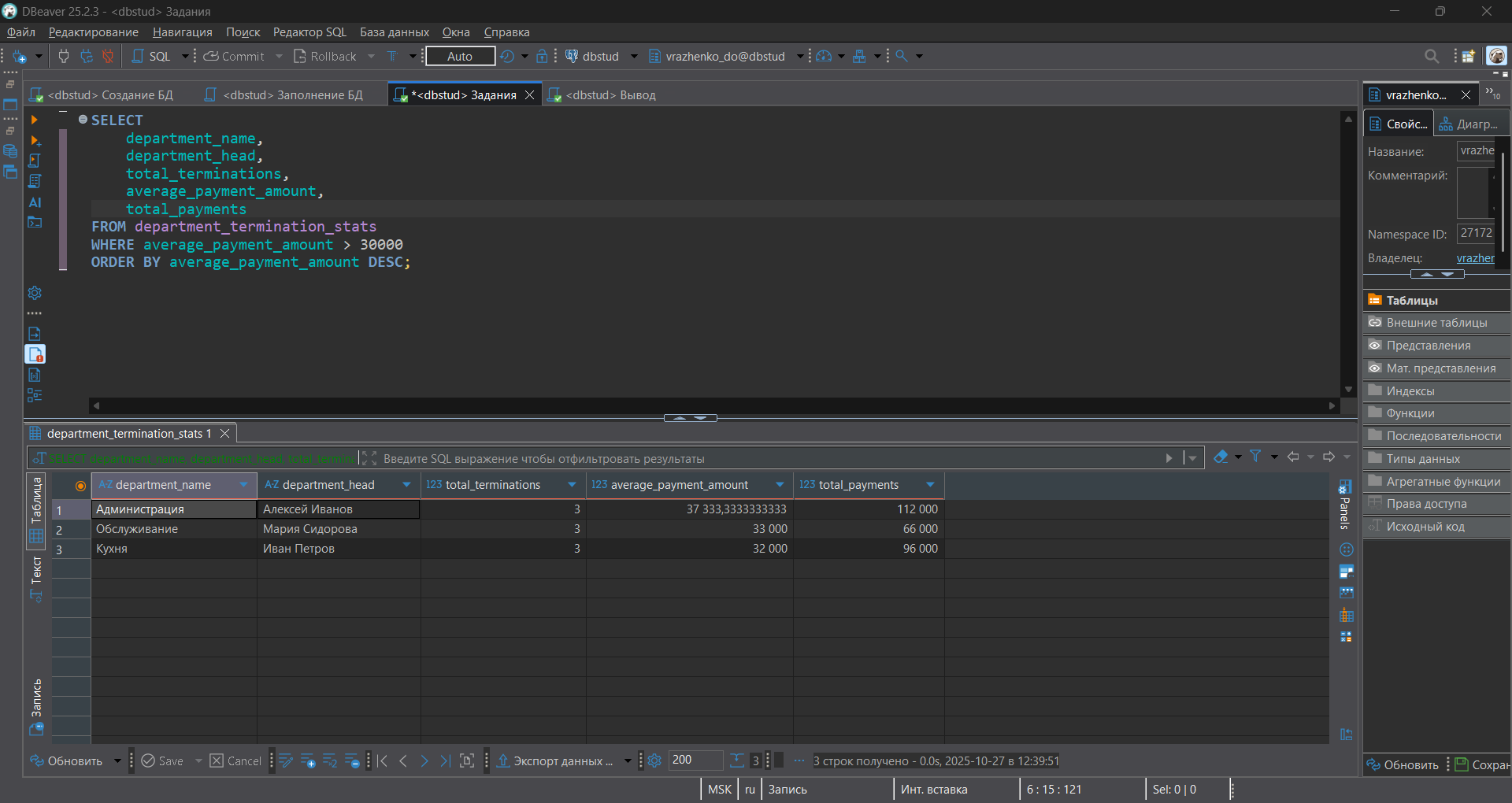


Рисунок 9 − Запрос с использованием аналитического представления

# **Задание №5: создание и обновление материализованного представления**

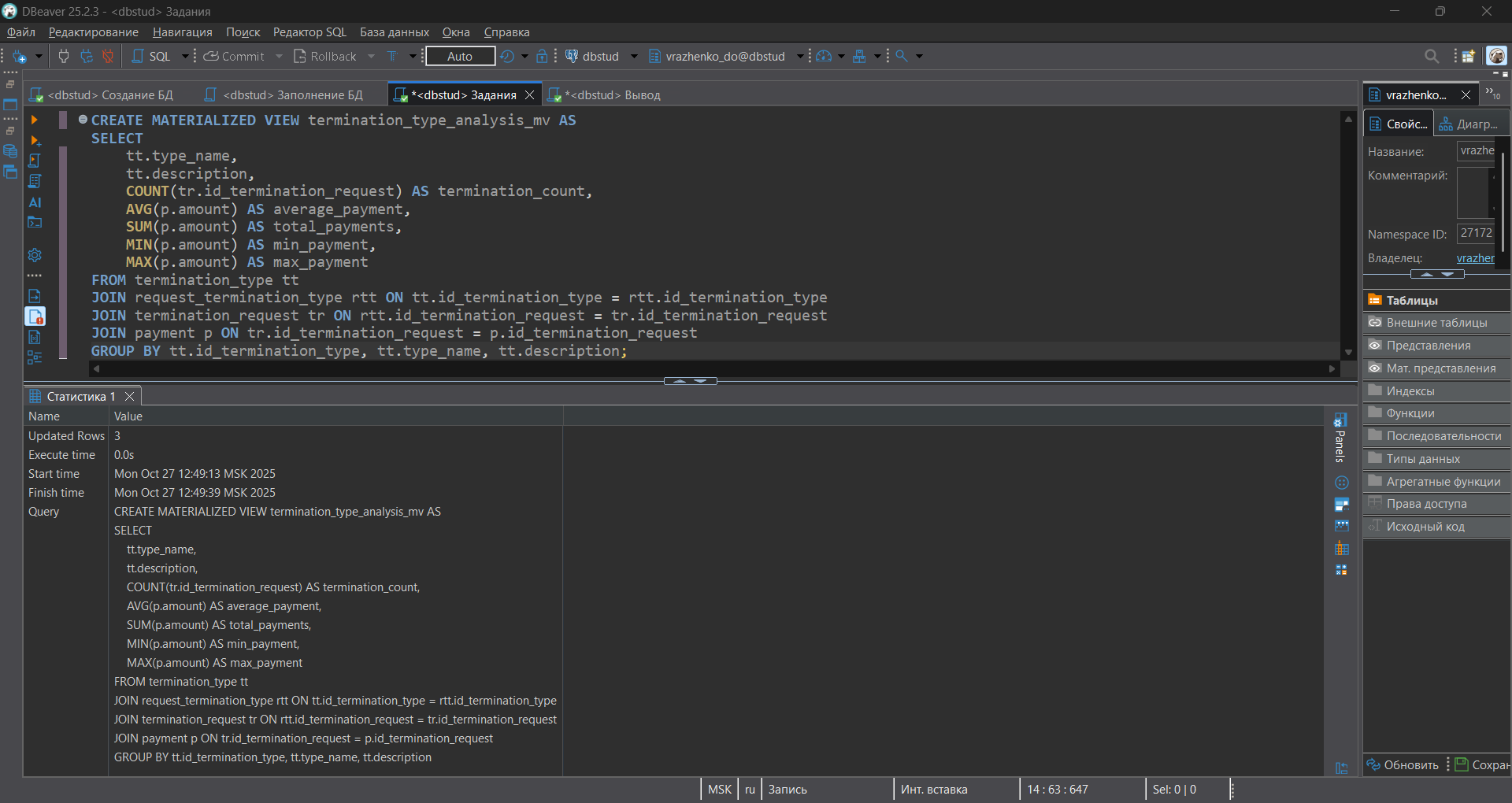


Рисунок 10 − Создание материального представления

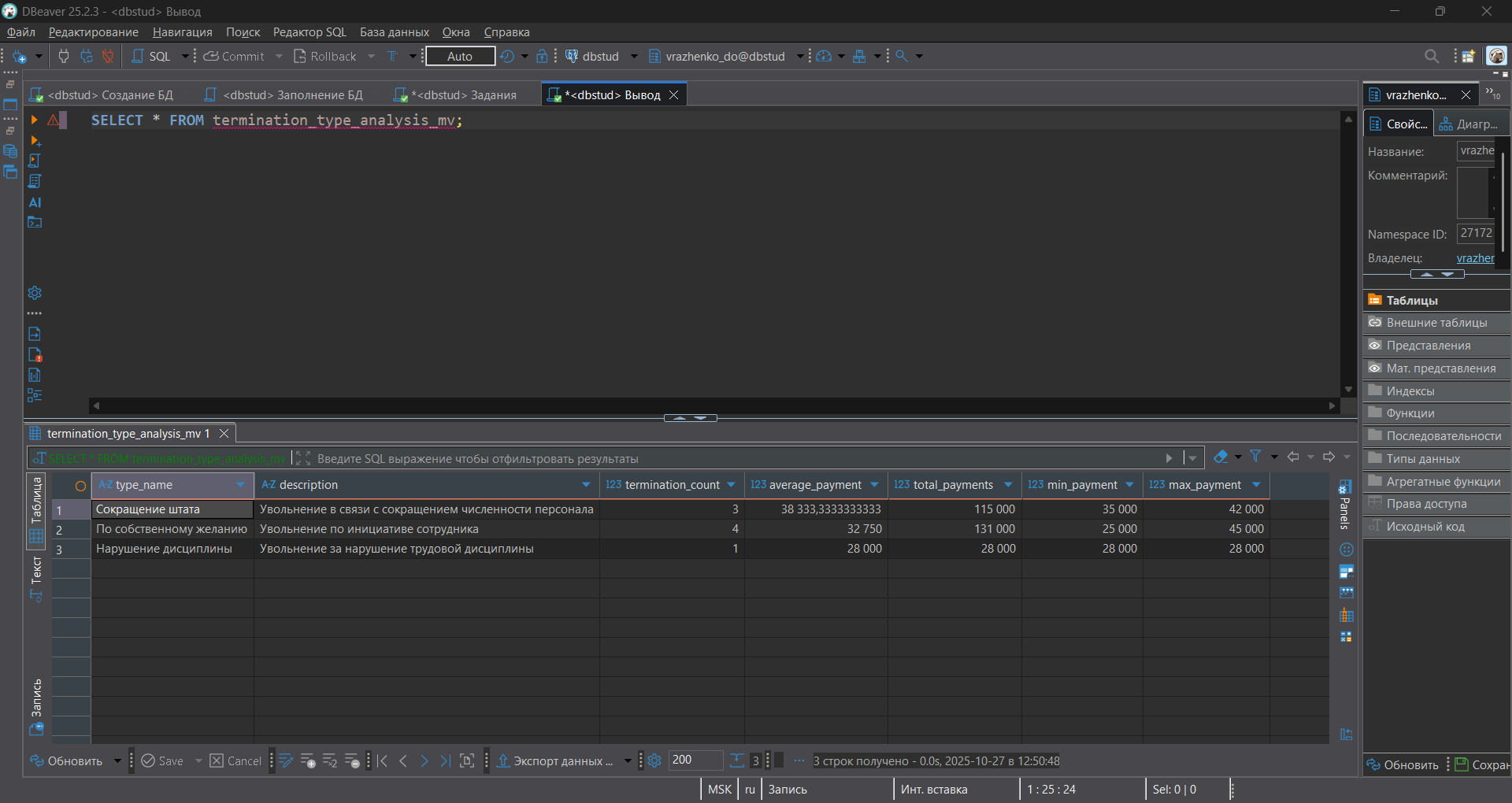


Рисунок 11 − Результат выполнения

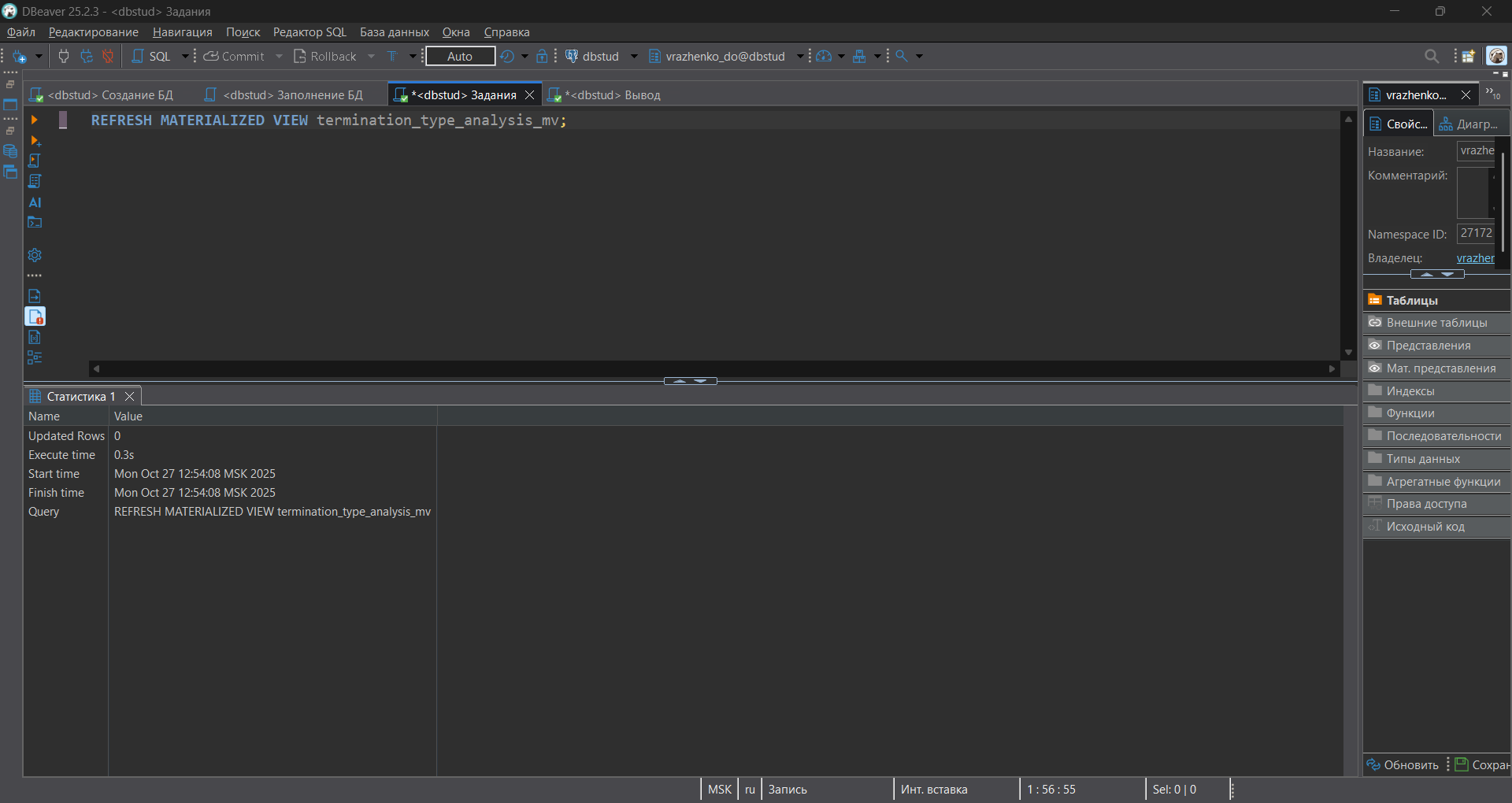


Рисунок 12 − Обновление данных

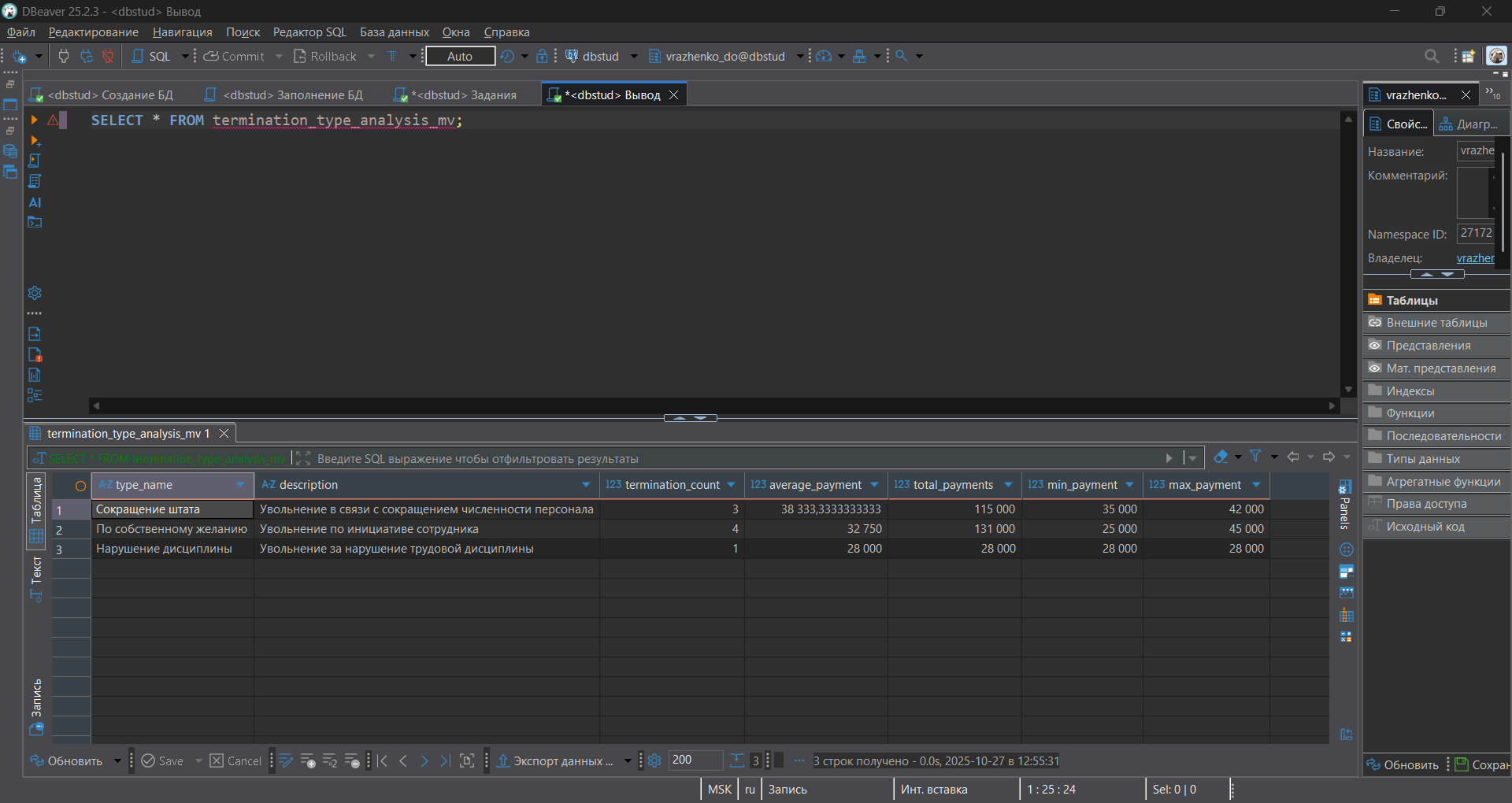


Рисунок 13 − Результат выполнения

# **Задание №6: разработка пользовательской функции для аналитических вычислений**

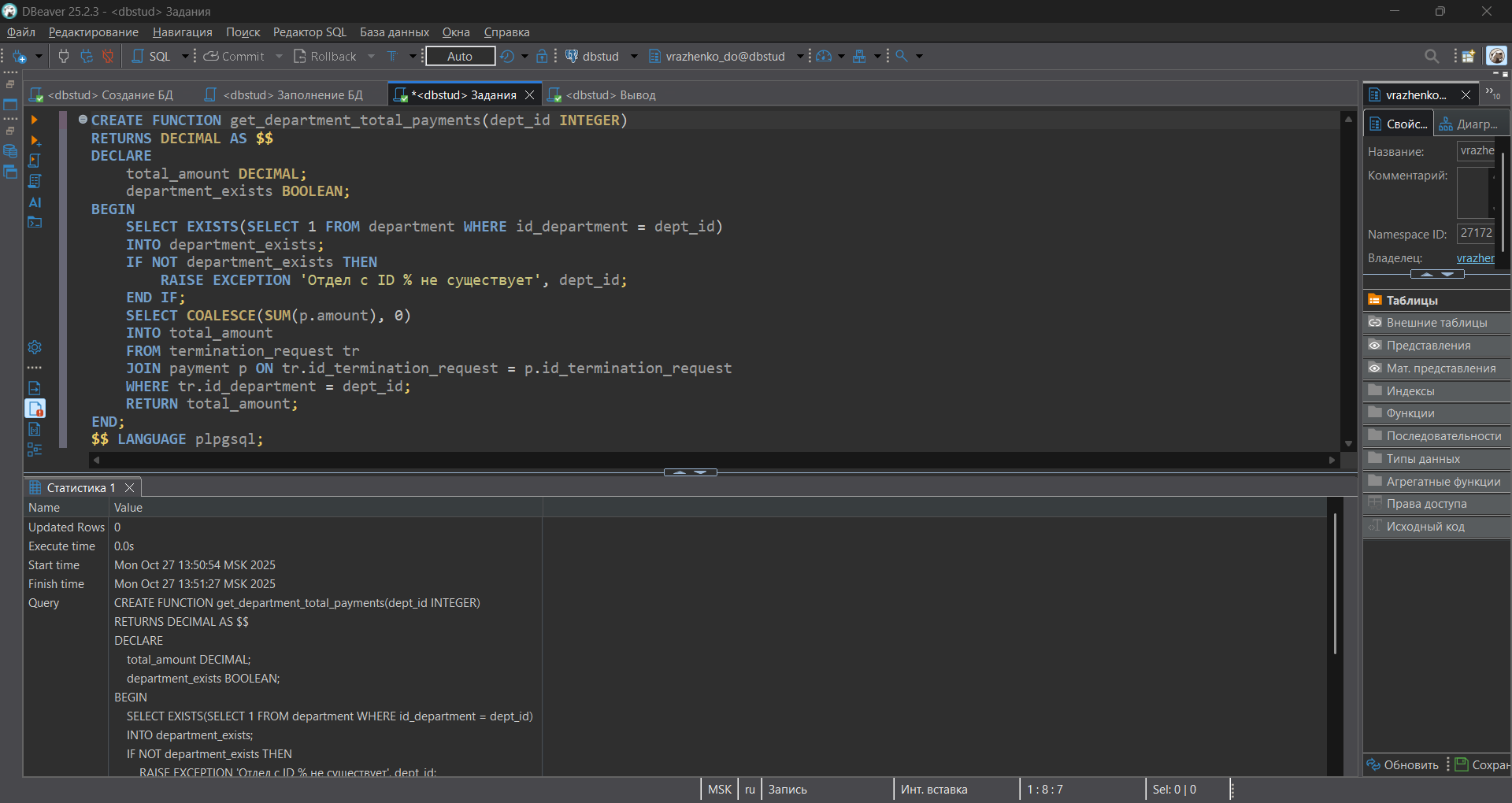


Рисунок 14 − Создание пользовательской функции

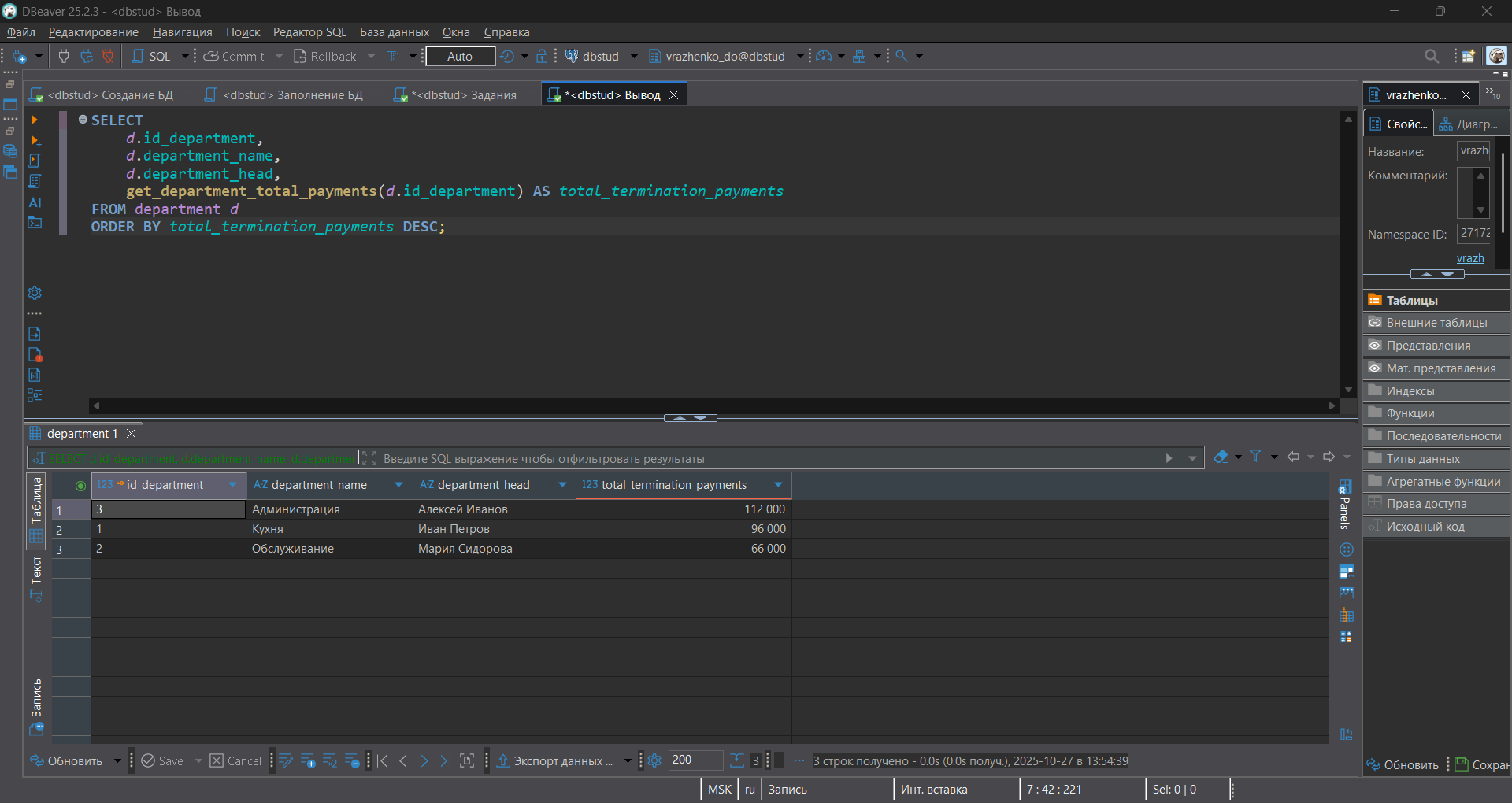


Рисунок 15 − Результат выполнения

# Задание №7: разработка хранимой процедуры для выполнения сложной операции

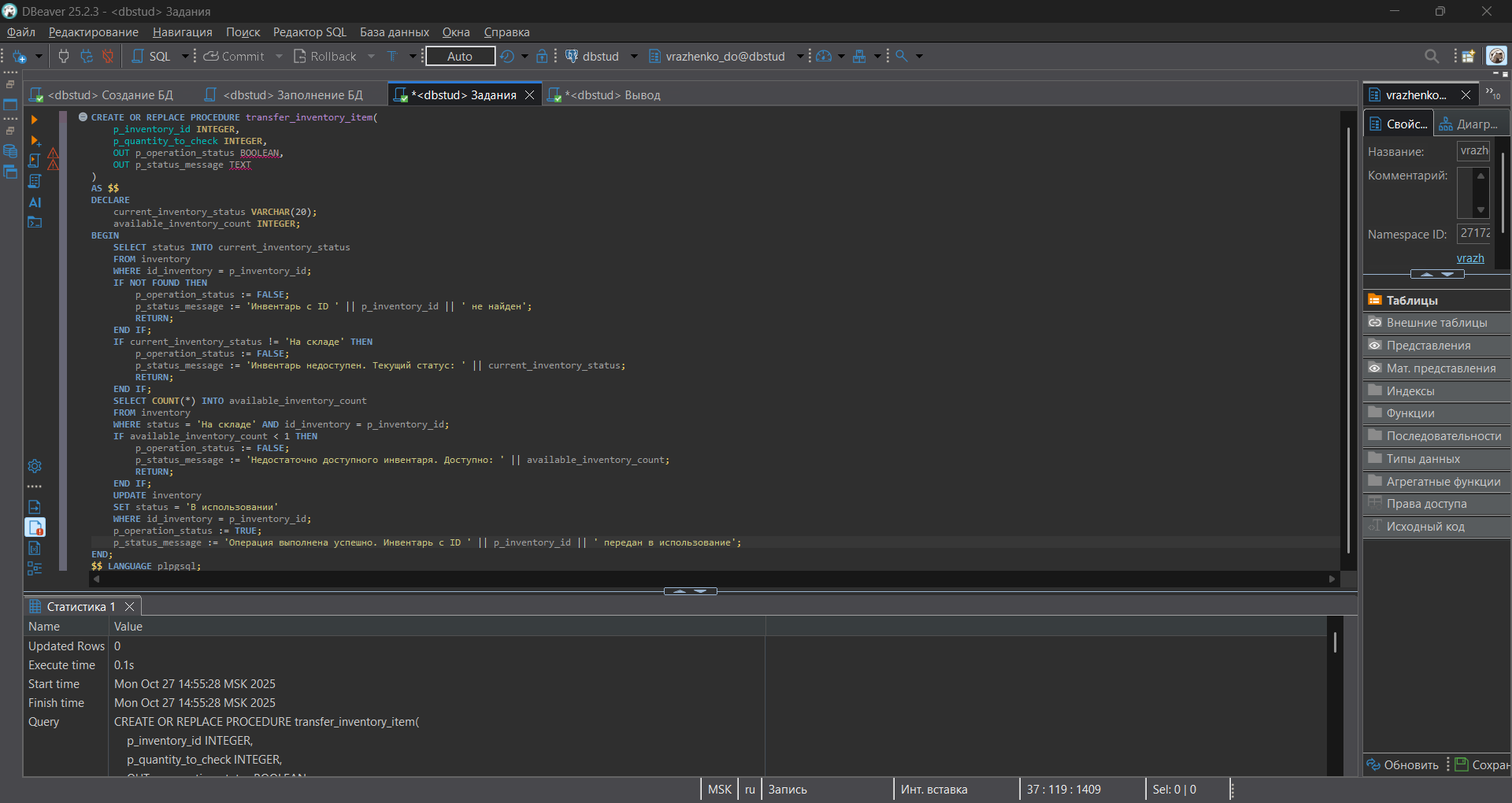


Рисунок 16 − Создание хранимой процедуры

# **Задание №8: демонстрация вызова хранимой процедуры**

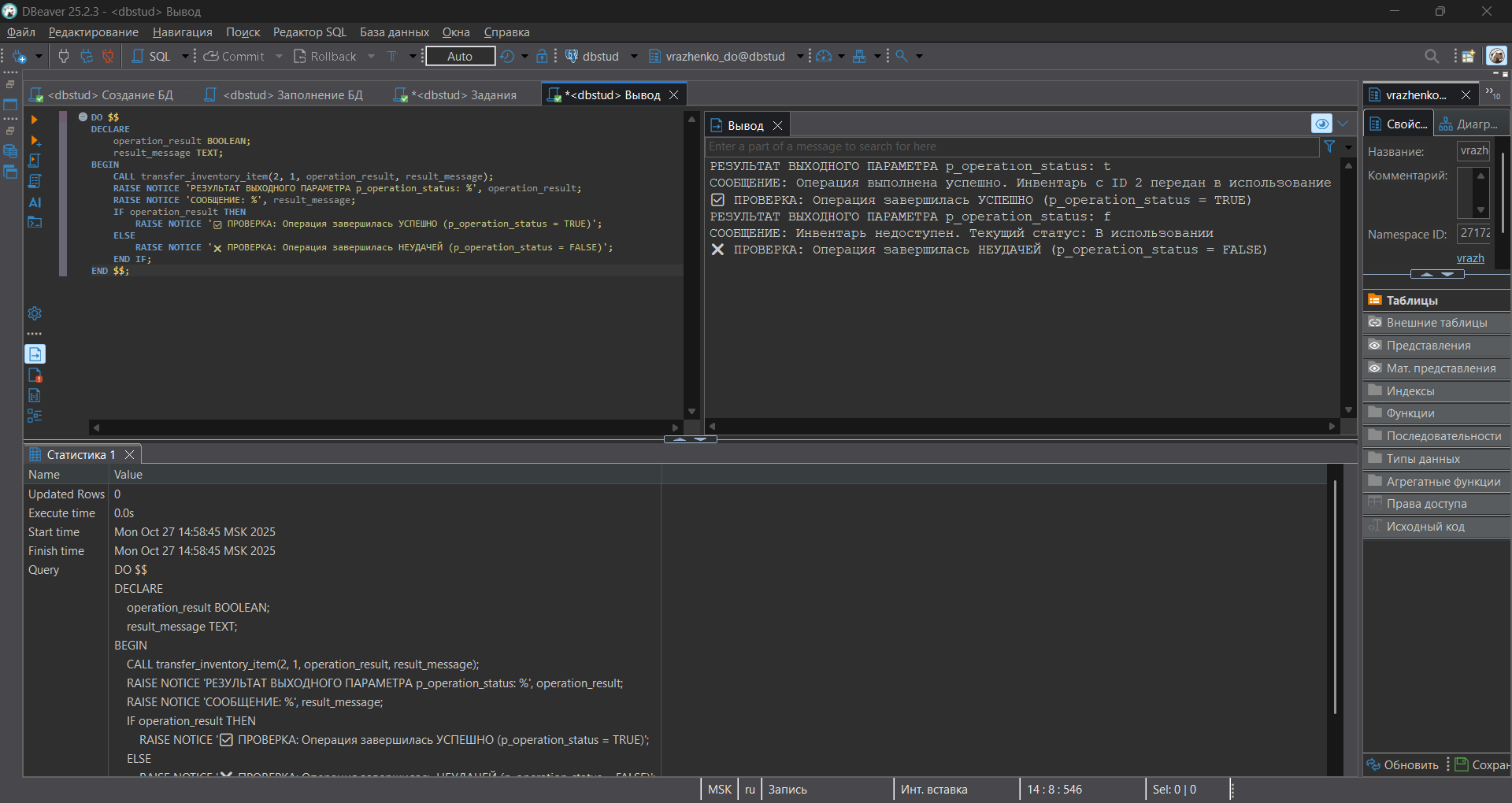


Рисунок 17 − Успешный и неудачный вызовы