



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 № 6

Студенты группы *ИКБО-50-23 Вражсенко Д.О.*

.

(подпись)

Преподаватель *Мажсей Я.В.*

(подпись)

Отчет представлен «___»____2025 г.

Москва 2025 г.

Цель работы:

Целью данной практической работы является формирование у студентов углубленных практических навыков по управлению данными и реализации сложной бизнес-логики в СУБД PostgreSQL с использованием триггеров и курсоров.

ХОД ВЫПОЛНЕНИЯ РАБОТЫ

Все используемые таблицы индивидуальной схемы данных:

Таблица 1. Таблица *employee* (Сотрудник)

④	123 ↪ id_en	123 ↗ id_pc	A-Z last_name	A-Z first_name	A-Z phone_num	A-Z registration_address	⌚ employment_date	⌚ contract_end_date
1	1	1	Петров	Иван	9123456789	ул. Ленина, 10	2020-01-15	2025-01-15
2	2	1	Сидорова	Мария	9234567890	пр. Мира, 5	2019-05-20	2024-05-20
3	3	1	Иванов	Алексей	9345678901	ул. Советская, 3	2021-03-10	2026-03-10
4	4	2	Васильев	Дмитрий	9456789012	ул. Гагарина, 7	2022-02-01	2023-02-01
5	5	3	Кузнецова	Ольга	9567890123	пр. Победы, 12	2021-07-15	2024-07-15
6	6	4	Смирнов	Андрей	9678901234	ул. Лесная, 9	2023-01-10	2023-12-31
7	7	2	Орлов	Михаил	9789012345	ул. Центральная, 15	2022-03-01	2025-03-01
8	8	3	Волкова	Екатерина	9890123456	пр. Солнечный, 8	2021-11-10	2024-11-10
9	9	4	Никитин	Артем	9901234567	ул. Весенняя, 22	2023-02-15	2024-02-15
10	10	5	Семенова	Анна	9012345678	ул. Осенняя, 11	2020-08-20	2025-08-20
11	11	6	Козлов	Денис	9123456780	пр. Гагарина, 33	2023-04-10	2024-04-10

Таблица 2. Таблица *position* (Должность)

④	123 ↪ id_position	A-Z position_name	A-Z access_category	123 salary
1	1	Менеджер	Администрация	50 000
2	2	Повар	Кухня	35 000
3	3	Кассир	Обслуживание	30 000
4	4	Уборщик	Обслуживание	25 000
5	5	Старший менеджер	Администрация	70 000
6	6	Шеф-повар	Кухня	60 000
7	7	Старший кассир	Обслуживание	40 000

Таблица 3. Таблица *employee_inventory* (Связующая таблица Сотрудник-Инвентарь)

④	123 ↪ id_employee_inventory	123 ↗ id_employee	123 ↗ id_inventory
1	1	4	1
2	2	5	2
3	3	4	3
4	4	6	4
5	5	1	5
6	6	2	6

Таблица 4. Таблица *inventory* (Инвентарь)

④	123 ↪ id_inventory	A-Z item_name	A-Z category	A-Z status
1	1	Нож поварской	Кухонное оборудование	В использовании
2	2	Кассовый аппарат	Кассовое оборудование	На складе
3	3	Фартук	Спецодежда	В использовании
4	4	Стол кухонный	Мебель	На складе
5	5	Компьютер	Офисная техника	В использовании
6	6	Принтер	Офисная техника	На складе

Таблица 5. Таблица termination_request (Запрос на увольнение)

	123 → id_termination_request	123 ↗ id_department	123 ↗ id_employee	⌚ request_date	A-Z reason	A-Z status	⌚ termination_date
1	1	3	6	2023-11-01	Окончание контракта	Завершено	2023-12-31
2	2	2	5	2023-10-15	Переход на другую работу	В процессе	2024-01-01
3	3	1	4	2023-09-15	Несоответствие должности	Завершено	2023-09-30
4	4	2	8	2023-08-10	Нарушение трудовой дисциплины	Завершено	2023-08-25
5	5	3	3	2023-07-05	Смена места жительства	Завершено	2023-07-20
6	6	1	7	2023-06-20	Окончание контракта	Завершено	2023-06-30
7	7	2	9	2023-05-15	По соглашению сторон	Завершено	2023-05-31
8	8	3	10	2023-04-10	Сокращение штата	Завершено	2023-04-25
9	9	1	11	2023-03-05	Неудовлетворенность зарплатой	Завершено	2023-03-20

Таблица 6. Таблица department (Отдел)

	123 → id_department	A-Z department_name	A-Z department_head
1	1	Кухня	Иван Петров
2	2	Обслуживание	Мария Сидорова
3	3	Администрация	Алексей Иванов

Таблица 7. Таблица payment (Выплата)

	123 → id_payment	123 ↗ id_termination_request	123 amount	⌚ payment_date	A-Z comment
1	1	1	25 000	2023-12-31	Окончательный расчет по увольнению
2	2	3	35 000	2023-09-30	Расчет при увольнении
3	3	4	28 000	2023-08-25	Окончательный расчет
4	4	5	45 000	2023-07-20	Компенсационные выплаты
5	5	6	32 000	2023-06-30	Расчет по увольнению
6	6	7	38 000	2023-05-31	Выплаты при увольнении
7	7	8	42 000	2023-04-25	Компенсация за сокращение
8	8	9	29 000	2023-03-20	Окончательный расчет

Таблица 8. Таблица document (Архив документов)

	123 → id_document	A-Z document_type	⌚ creation_date	123 ↗ id_termination_request	A-Z content
1	1	Заявление на увольнение	2023-10-30	1	Заявление на увольнение по собственному
2	2	Приказ об увольнении	2023-12-31	1	Приказ №123 об увольнении Смирнова А.
3	3	Заявление на увольнение	2023-09-10	3	Заявление от Васильева Д.
4	4	Приказ об увольнении	2023-09-30	3	Приказ №124 об увольнении Васильева ,

Таблица 9. Таблица request_termination_type (Связующая таблица Запрос на увольнение- Вид увольнения)

	123 → id_request_termination_type	123 ↗ id_termination_request	123 ↗ id_termination_type
1	1	1	1
2	2	2	3
3	3	3	2
4	4	4	3
5	5	5	1
6	6	6	1
7	7	7	2
8	8	8	2
9	9	9	1

Таблица 10. Таблица termination_type (Вид увольнения)

	123 → id_termination_type	A-Z type_name	A-Z description
1	1	По собственному желанию	Увольнение по инициативе сотрудника
2	2	Сокращение штата	Увольнение в связи с сокращением численности персонала
3	3	Нарушение дисциплины	Увольнение за нарушение трудовой дисциплины

Задание №1: триггеры

DBeaver 25.2.3 - <dbstud> Задания

```
CREATE OR REPLACE FUNCTION validate_employee_inventory_return()
RETURNS TRIGGER AS $$
DECLARE
    inventory_count INTEGER;
BEGIN
    SELECT COUNT(*)
    INTO inventory_count
    FROM employee.inventory
    WHERE id.employee = NEW.id.employee;
    IF inventory_count > 0 THEN
        RAISE EXCEPTION 'Сотрудник имеет несданный инвентарь. Увольнение невозможно.';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;
```

CREATE OR REPLACE TRIGGER prevent_termination_with_inventory
BEFORE INSERT ON termination_request
FOR EACH ROW
EXECUTE FUNCTION validate_employee_inventory_return();

Статистика 1 ×

Name	Value
Queries	2
Updated Rows	0
Execute time	0.06
Fetch time	0.06
Total time	0.06
Start time	2025-11-14 14:50:15.572
Finish time	2025-11-14 14:50:15.594

MSK | ru | Запись | Инт. вставка | 21:1:597 | Sel: 0 | 0 | Обновить | Сохранить

Рисунок 1 – Сложная валидация (проверка)

DBeaver 25.2.3 - <dbstud> Выход

```
INSERT INTO termination_request (id_department, id_employee, request_date, reason, status, termination_date)
VALUES (3, 3, '2023-12-01', 'Тестовая причина', 'В процессе', '2023-12-31');

--СОТРУДНИК С ИНВЕНТАРЕМ
INSERT INTO termination_request (id_department, id_employee, request_date, reason, status, termination_date)
VALUES (1, 4, '2023-12-01', 'Тестовая причина', 'В процессе', '2023-12-31');
```

Execution Error

Error occurred during SQL query execution

Причина:
SQL Error [P0001]: ERROR: Сотрудник имеет несданный инвентарь.
Увольнение невозможно.
Где: PL/pgSQL function validate_employee_inventory_return() line 10 at RAISE

Остановить | Повторить | Сведения >

vrazhenko... ×

Свойства Диаграмма

Название: vrag2

Комментарий:

Namespace ID: 2717c

Таблицы Внешние таблицы Представления Мат. представления Индексы Функции Последовательности Типы данных Агрегатные функции Права доступа Исходный код

Обновить | Сохранить

Рисунок 2 – Демонстрация работы сложной валидации

```

CREATE OR REPLACE FUNCTION auto_create_termination_documents()
RETURNS TRIGGER AS $$$
BEGIN
    INSERT INTO document (document_type, creation_date, id_termination_request, content)
    VALUES (
        'Заявление на увольнение',
        CURRENT_DATE,
        NEW.id_termination_request,
        'Автоматически созданное заявление на увольнение для сотрудника ID: ' || NEW.id_employee
    );
    INSERT INTO document (document_type, creation_date, id_termination_request, content)
    VALUES (
        'Приказ об увольнении',
        NEW.termination_date,
        NEW.id_termination_request,
        'Приказ об увольнении сотрудника ID: ' || NEW.id_employee || '. Причина: ' || NEW.reason
    );
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER trg_auto_create_documents
AFTER INSERT ON termination_request
FOR EACH ROW
EXECUTE FUNCTION auto_create_termination_documents();

```

Статистика 1

Name	Value
Queries	2
Updated Rows	0
Execute time	0.06
Fetch time	0.06
Total time	0.06
Start time	2025-11-14 15:02:43.833
Finish time	2025-11-14 15:02:43.864

Рисунок 3 – Поддержание согласованности (дениормализация)

```

INSERT INTO termination_request (id_department, id_employee, request_date, reason, status, termination_date)
VALUES (2, 5, '2023-12-01', 'Окончание контракта', 'Завершено', '2023-12-31');

SELECT * FROM document WHERE id_termination_request = (SELECT MAX(id_termination_request) FROM termination_request);

```

document 1

	id_document	document_type	creation_date	id_termination_request	content
1	5	Заявление на увольнение	2025-11-14 20:00:00	2	Автоматически созданное заявление на увольнение для сотрудника ID: 5
2	6	Приказ об увольнении	2023-12-31 20:00:00	2	Приказ об увольнении сотрудника ID: 5. Причина: Окончание контракта

Рисунок 4 – Демонстрация работы поддержания согласованности

DBeaver 25.2.3 - <dbstud> Задания

```

CREATE TABLE IF NOT EXISTS termination_audit (
    id_audit serial PRIMARY KEY,
    id_termination_request integer NOT NULL,
    old_status varchar(50),
    new_status varchar(50),
    change_date timestamp DEFAULT CURRENT_TIMESTAMP,
    changed_by varchar(100) DEFAULT CURRENT_USER
);

CREATE OR REPLACE FUNCTION log_termination_status_changes()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.status IS DISTINCT FROM OLD.status THEN
        INSERT INTO termination_audit (id_termination_request, old_status, new_status)
        VALUES (NEW.id_termination_request, OLD.status, NEW.status);
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER trg_log_status_changes
AFTER UPDATE ON termination_request
FOR EACH ROW
EXECUTE FUNCTION log_termination_status_changes();

```

Статистика 1

Name	Value
Queries	3
Updated Rows	0
Execute time	0.06
Fetch time	0.06
Total time	0.06
Start time	2025-11-14 15:16:29.102
Finish time	2025-11-14 15:16:29.147

Рисунок 5 – Аудит и логирование

DBeaver 25.2.3 - <dbstud> Вывод

```

UPDATE termination_request
SET status = 'Завершено'
WHERE id_termination_request = 2;

SELECT * FROM termination_audit;

```

termination_audit 1

	id_audit	id_termination_request	old_status	new_status	change_date	changed_by
1	1	2	В процессе	Завершено	2025-11-14 15:19:01.795	vrazhenko_do

Рисунок 6 – Демонстрация работы аудита и логирования (часть 1)

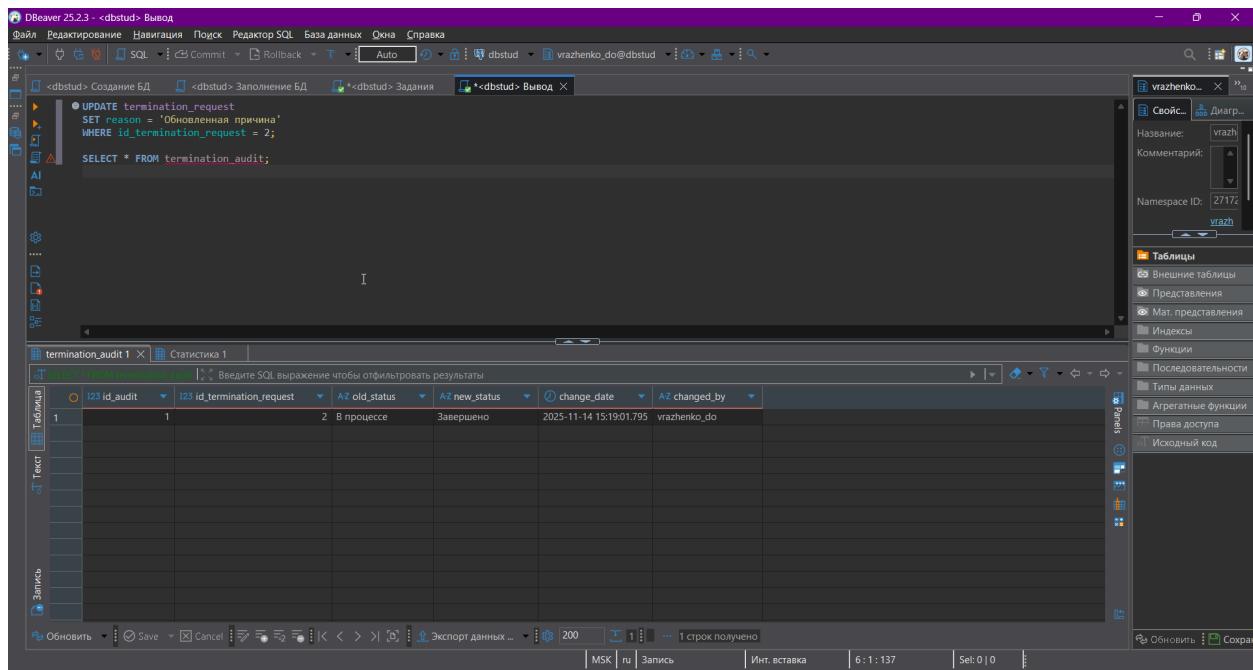


Рисунок 7 – Демонстрация работы аудита и логирования (часть 2)

Задание №2: курсоры

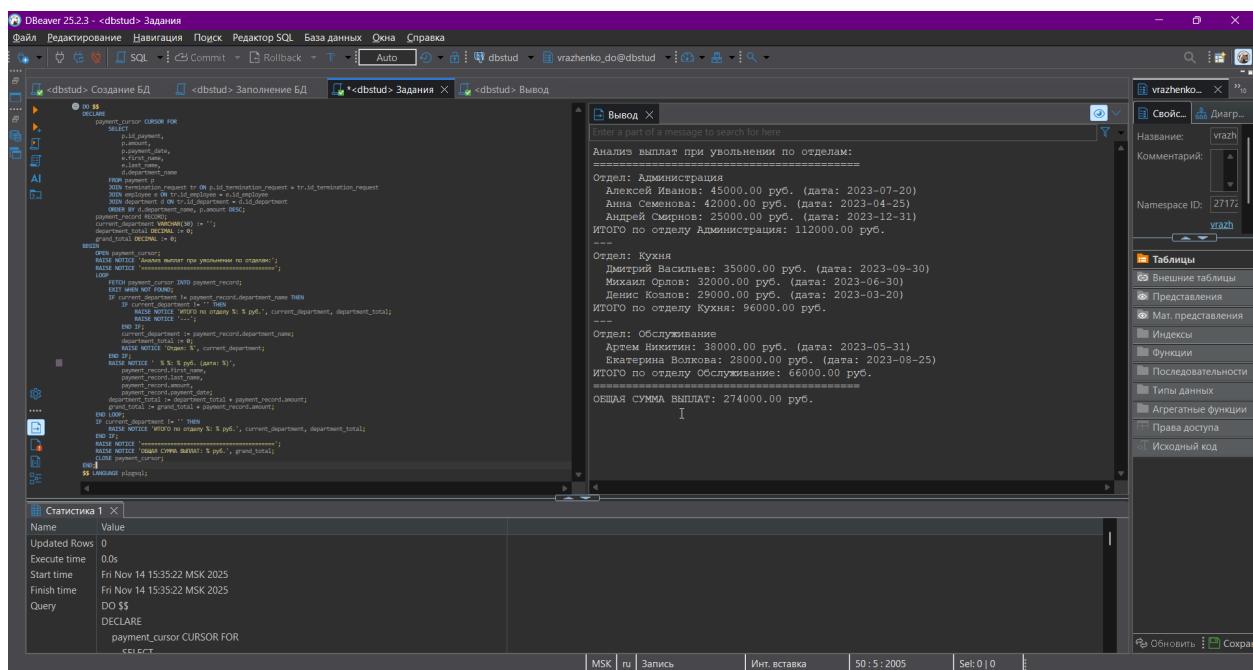


Рисунок 8 – Явный курсор

DBeaver 25.2.3 - <dbstud> Задания

Файл Редактирование Навигация Поиск Редактор SQL База данных Окна Справка

<dbstud> Создание БД <dbstud> Заполнение БД <dbstud> Задания <dbstud> Выход

```

CREATE OR REPLACE FUNCTION get_termination_statistics()
RETURNS TABLE(
    termination_type VARCHAR(50),
    termination_count BIGINT,
    avg_payment DECIMAL,
    total_payments DECIMAL
) AS $$
DECLARE
    stat_record RECORD;
BEGIN
    FOR stat_record IN
        SELECT
            tt.type_name,
            COUNT(DISTINCT tr.id_termination_request) as count_terminations,
            AVG(g.amount) as average_payment,
            SUM(g.amount) as total_payment
        FROM termination_type tt
        LEFT JOIN request_termination_type rtt ON tt.id_termination_type = rtt.id_termination_type
        LEFT JOIN termination_requests tr ON rtt.id_termination_request = tr.id_termination_request
        LEFT JOIN payment g ON tr.id_termination_request = g.id_termination_request
        GROUP BY tt.id_termination_type, tt.type_name
        ORDER BY count_terminations DESC
    LOOP
        termination_type := stat_record.type_name;
        termination_count := stat_record.count_terminations;
        total_payments := stat_record.total_payment;
        avg_payment := stat_record.average_payment;
        total_payments := stat_record.total_payment;
        RETURN NEXT;
    END LOOP;
END;
$$ LANGUAGE plpgsql;
SELECT * FROM get_termination_statistics();

```

Результат 1 | Статистика 1

По собственному желанию

	termination_type	termination_count	avg_payment	total_payments
1	По собственному желанию	4	32 750	131 000
2	Сокращение штата	3	38 333,333333333	115 000
3	Нарушение дисциплины	2	28 000	28 000

Обновить Save Cancel Экспорт данных ... 200 3 ... 3 строк получено - 0.0s, 2025-11-14 в 15:36:37 MSK | и | Запись | Инт. вставка | Sel: 0 | 0 |

Свойства... Диаграмма...

Название: уразх
Комментарий:
Namespace ID: 2717c
уразх

Таблицы
Внешние таблицы
Представления
Мат. представления
Индексы
Функции
Последовательности
Типы данных
Агрегатные функции
Права доступа
Исходный код

Рисунок 9 – Неявный курсор