МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное   
учреждение высшего образования

**«Южно-Уральский государственный университет**

**(национальный исследовательский университет)»**

**Высшая школа электроники и компьютерных наук**

**Кафедра системного программирования**

**ОТЧЁТ**

о выполнении практического задания №7-8

по дисциплине

«Технологии Баз данных»

|  |  |
| --- | --- |
|  | Выполнил:  студент группы КЭ-301  А.С. Гордеев  Проверил:  ст. преподаватель кафедры СП, Л.Н. Петрова |

Челябинск-2022

# **Задание 7. Разработка ограничений безопасности данных**

Экономист имеет право осуществлять выборку, добавление, обновление и удаление данных о поставщиках и поставках, а также выборку данных о деталях и проектах.

Директор имеет право осуществлять выборку, добавление, обновление и удаление данных о поставщиках, деталях и проектах, а также выборку и удаление данных о поставках.

Данные ограничения представлены в листинге 1.

Листинг 1 — задание ролей экономиста и директора

DROP ROLE IF EXISTS economist;

DROP ROLE IF EXISTS director;

-- Роли

CREATE ROLE economist WITH LOGIN;

GRANT ALL ON suppliers TO economist;

GRANT ALL ON supplies TO economist;

GRANT SELECT ON details TO economist;

GRANT SELECT ON projects TO economist;

GRANT SELECT ON e\_suppliers\_view TO economist;

GRANT SELECT ON e\_details\_view TO economist;

GRANT SELECT ON e\_projects\_view TO economist;

CREATE ROLE director WITH LOGIN;

GRANT ALL ON suppliers TO director;

GRANT ALL ON details TO director;

GRANT ALL ON projects TO director;

GRANT SELECT ON supplies TO director;

GRANT DELETE ON supplies TO director;

# **Задание 8. Разработка триггеров**

1. Ненадежный поставщик не может поставлять дорогую деталь (то есть не может существовать поставка детали с ценой 1000 руб. и более от поставщика с рейтингом менее 6).
2. Вес поставки не должен превышать 1.5 т (вес поставки вычисляется как произведение веса одной детали на количество деталей в поставке).
3. Суммарная стоимость всех поставок по проекту не должна превышать бюджет проекта (стоимость одной поставки вычисляется как произведение цены детали на количество деталей в данной поставке).

Листинг 2 — создание триггеров

-- Триггеры

-- 1. Ненадежный поставщик не может поставлять дорогую деталь

CREATE OR REPLACE FUNCTION unreliable\_suppliers()

RETURNS TRIGGER

LANGUAGE plpgsql

AS

$$

BEGIN

RETURN CASE

WHEN (SELECT reliability = 'unreliable' FROM suppliers WHERE id = NEW.supplier\_id) AND

(SELECT cheapness = 'expensive' FROM details WHERE id = NEW.detail\_id) THEN NULL

ELSE NEW END;

END;

$$;

CREATE OR REPLACE TRIGGER unreliable\_supplier\_insert

BEFORE INSERT

ON supplies

FOR EACH ROW

EXECUTE FUNCTION unreliable\_suppliers();

CREATE OR REPLACE TRIGGER unreliable\_supplier\_update

BEFORE UPDATE

ON supplies

FOR EACH ROW

EXECUTE FUNCTION unreliable\_suppliers();

-- 2. Вес поставки не должен превышать 1.5 т

CREATE OR REPLACE FUNCTION block\_overweight\_supplies()

RETURNS TRIGGER

LANGUAGE plpgsql

AS

$$

BEGIN

RETURN CASE

WHEN (SELECT MAX(weight) FROM details WHERE id = NEW.detail\_id) \* NEW.amount > 1500000 THEN NULL

ELSE NEW END;

END;

$$;

CREATE OR REPLACE TRIGGER overweight\_supplies\_insert

BEFORE INSERT

ON supplies

FOR EACH ROW

EXECUTE FUNCTION block\_overweight\_supplies();

CREATE OR REPLACE TRIGGER overweight\_supplies\_update

BEFORE UPDATE

ON supplies

FOR EACH ROW

EXECUTE FUNCTION block\_overweight\_supplies();

-- 3. Суммарная стоимость всех поставок по проекту не должна превышать бюджет проекта

CREATE OR REPLACE FUNCTION project\_overdraft()

RETURNS TRIGGER

LANGUAGE plpgsql

AS

$$

BEGIN

RETURN CASE

WHEN (SELECT SUM(detail.price \* supply.amount)

FROM supplies supply

INNER JOIN details detail ON detail.id = supply.detail\_id

WHERE project\_id = NEW.project\_id) +

NEW.amount \* (SELECT SUM(price) FROM details WHERE id = NEW.detail\_id) >

(SELECT SUM(budget) FROM projects WHERE id = NEW.project\_id)

THEN NULL

ELSE NEW END;

END;

$$;

CREATE OR REPLACE TRIGGER overdraft\_projects\_insert

BEFORE INSERT

ON supplies

FOR EACH ROW

EXECUTE FUNCTION project\_overdraft();

CREATE OR REPLACE TRIGGER overdraft\_projects\_update

BEFORE UPDATE

ON supplies

FOR EACH ROW

EXECUTE FUNCTION project\_overdraft();