

**Федеральное государственное автономное образовательное
учреждение высшего образования
"Национальный исследовательский университет
"Высшая школа экономики"
Московский институт электроники и математики им. А.Н.Тихонова
Департамент компьютерной инженерии**

Лабораторная работа № 7
по курсу «Базы данных»
Тема: Создание триггеров в СУБД

Студент: Молчанов Данил
Андреевич

Группа: М_БД_8

Преподаватель: Чернышов Л.Н.

Дата:

Оценка:

Москва, 2023

1. Постановка задачи

Вариант 2

1. Проверка значений всех полей отношения "Сотрудники", для которых могут быть определены домены (в т.ч., ИНН может содержать только цифры, (возраст сотрудника)-(стаж на прежних работах)-(стаж работы на данном предприятии) не может быть меньше 16 лет, а дата поступления на работу должна быть не больше текущей даты). Если при вводе данных дата поступления не указана, устанавливать текущую дату.
2. Установка значения поля "пол", если оно не установлено. Правила: если отчество заканчивается на '-НА', то пол женский, если на '-ИЧ', то мужской. В противном случае триггер должен генерировать ошибку.

2. Решение

1. Проверка значений всех полей отношения "Сотрудники", для которых могут быть определены домены (в т.ч., ИНН может содержать только цифры, (возраст сотрудника)-(стаж на прежних работах)-(стаж работы на данном предприятии) не может быть меньше 16 лет, а дата поступления на работу должна быть не больше текущей даты). Если при вводе данных дата поступления не указана, устанавливать текущую дату.

```
CREATE
OR REPLACE FUNCTION verify_employee_data() RETURNS trigger AS
$$ BEGIN CASE WHEN NOT (
    new.taxpayer_number LIKE '%[^0-9]%'
    AND extract(
        year
        from
            age(new.date_of_birth)
        ) - get_experience(
            new.work_experience, new.date_of_taking_office
        ) > 16
    AND new.date_of_taking_office < CURRENT_TIMESTAMP
) THEN RAISE EXCEPTION '%',
'Invalid data';
ELSE RETURN NEW;
CASE WHEN new.date_of_taking_office END CASE;
END;
$$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER verify_employee_data_trigger BEFORE INSERT
OR
UPDATE
    ON employee FOR EACH ROW EXECUTE PROCEDURE
verify_employee_data();
```

Тестирование:

```
72 INSERT INTO employee VALUES
73 (1043, 'Molchanov', 'Danil Andreevich', '120020239102', 'm', '2005-05-30', 2, 'first post', '2019-05-20', 0)
```

Data Output Messages Notifications

ERROR: ОШИБКА: Invalid data
CONTEXT: функция PL/pgSQL verify_employee_data(), строка 6, оператор RAISE

SQL state: P0001

Рисунок 1 – результат работы триггера на проверку возраста

```
72 INSERT INTO employee VALUES
73 (1043, 'Molchanov', 'Danil Andreevich', '120ww02023ks', 'm', '1995-05-30', 2, 'first post', '2019-05-20', 0)
```

Data Output Messages Notifications

ERROR: ОШИБКА: Invalid data
CONTEXT: функция PL/pgSQL verify_employee_data(), строка 6, оператор RAISE

Рисунок 2 – результат работы триггера на проверку ИНН

2. Установка значения поля "пол", если оно не установлено. Правила: если отчество заканчивается на '-НА', то пол женский, если на '-ИЧ', то мужской. В противном случае триггер должен генерировать ошибку:

```
CREATE
OR REPLACE FUNCTION update_gender() RETURNS trigger AS $BODY$
BEGIN NEW.gender = CASE WHEN NEW.name_and_patronymic ~ 'ich'
THEN 'm' WHEN NEW.name_and_patronymic ~ 'vna' THEN 'f' ELSE
'u' END CASE;
RETURN NEW;
END;
$BODY$ LANGUAGE plpgsql;
CREATE TRIGGER "update_employees_gender_on_insert_trigger"
BEFORE INSERT ON employee FOR EACH ROW EXECUTE PROCEDURE
"update_gender"();
```

Тестирование:

```
69 INSERT INTO employee (security_number, surname, name_and_patronymic, taxpayer_number, date_of_birth, curr_department
70 VALUES (1043, 'Molchanov', 'Danil Andreevich', '120220202355', '1985-05-30', 2, 'first post', '2019-05-20', 5)
```

Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 38 msec.

	security_number [PK] numeric (6)	surname character (20)	name_and_patronymic character (30)	taxpayer_number character (12)	gender character (1)
1	101	Shelest	Maxim Antonovich	12345678901	m
2	1043	Molchanov	Danil Andreevich	120220202355	m

Рисунок 3 – результат работы триггера на автозаполнение пола

```
69 INSERT INTO employee (security_number, surname, name_and_patronymic, taxpayer_number, date_of_birth, curr_department
70 VALUES (1043, 'Molchanov', 'Danil Andree', '120220202355', '1985-05-30', 2, 'first post', '2019-05-20', 5)
```

Data Output Messages Notifications

ERROR: ОШИБКА: новая строка в отношении "employee" нарушает ограничение-проверку "employee_gender_check"

Рисунок 4 – результат работы триггера на автозаполнение пола

Список литературы:

1. PostgreSQL Documentation - The PostgreSQL Global Development Group (1996 - 2023) [Электронный ресурс]. URL: <https://www.postgresql.org/docs/>