МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский экономический университет им. Г.В. Плеханова»

Московский приборостроительный техникум

**Курсовой проект**

ПМ 02 Осуществление интеграции программных модулей

МДК 11.01 «Технология разработки и защиты баз данных»

Специальность 09.02.07 «Информационные системы и программирование»

Квалификация: Программист

Тема: «Разработка программного комплекса детского нейроцентра «Природа разума»»

**Скрипт БД**

Листов: 5

|  |  |
| --- | --- |
|  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / А.А. Шимбирёв  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 год  Исполнитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / С.А. Мельников  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 год |

2025

# АННОТАЦИЯ

В данном программном документе приведен скрипт базы данных для программного комплекса детского нейроцентра «Природа разума».

СОДЕРЖАНИЕ

[АННОТАЦИЯ 2](#_Toc168464824)

[1. СКРИПТ БАЗЫ ДАННЫХ 4](#_Toc168464825)

[1.1. Наименование скрипта 4](#_Toc168464826)

[1.2. Область применения скрипта 4](#_Toc168464827)

[1.3. Скрипт 4](#_Toc168464828)

# СКРИПТ БАЗЫ ДАННЫХ

## Наименование скрипта

Наименование – «models.sql».

## Область применения скрипта

Скрипт предназначен для программного комплекса детского нейроцентра «Природа разума»

## Скрипт

CREATE TYPE subdirectories AS ENUM (

'Диагностика',

'Анамнез',

'План работы',

'Комментарии специалистов',

'Фотографии и Видео'

);

CREATE TABLE roles (

id SERIAL PRIMARY KEY,

created\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

name VARCHAR(255) UNIQUE NOT NULL,

description VARCHAR(1000)

);

CREATE OR REPLACE FUNCTION update\_roles\_updated\_at()

RETURNS TRIGGER AS $$

BEGIN

NEW.updated\_at = CURRENT\_TIMESTAMP;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER roles\_updated\_at\_trigger

BEFORE UPDATE ON roles

FOR EACH ROW

EXECUTE FUNCTION update\_roles\_updated\_at();

CREATE TABLE users (

id SERIAL PRIMARY KEY,

created\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

fio VARCHAR(255) NOT NULL,

login VARCHAR(50) NOT NULL,

password BYTEA NOT NULL,

active BOOLEAN DEFAULT TRUE,

email VARCHAR(255) UNIQUE,

photo\_url VARCHAR(255),

role\_id INTEGER NOT NULL REFERENCES roles(id) ON DELETE CASCADE

);

CREATE OR REPLACE FUNCTION update\_users\_updated\_at()

RETURNS TRIGGER AS $$

BEGIN

NEW.updated\_at = CURRENT\_TIMESTAMP;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER users\_updated\_at\_trigger

BEFORE UPDATE ON users

FOR EACH ROW

EXECUTE FUNCTION update\_users\_updated\_at();

CREATE TABLE patients (

id SERIAL PRIMARY KEY,

created\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

fio VARCHAR(255) NOT NULL,

date\_of\_birth DATE NOT NULL

);

CREATE OR REPLACE FUNCTION update\_patients\_updated\_at()

RETURNS TRIGGER AS $$

BEGIN

NEW.updated\_at = CURRENT\_TIMESTAMP;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER patients\_updated\_at\_trigger

BEFORE UPDATE ON patients

FOR EACH ROW

EXECUTE FUNCTION update\_patients\_updated\_at();

CREATE TABLE documents (

id SERIAL PRIMARY KEY,

created\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

name VARCHAR(255) NOT NULL,

data BYTEA NOT NULL,

patient\_id INTEGER NOT NULL REFERENCES patients(id) ON DELETE CASCADE,

subdirectory\_type subdirectories NOT NULL,

author\_id INTEGER REFERENCES users(id) ON DELETE SET NULL

);

CREATE OR REPLACE FUNCTION update\_documents\_updated\_at()

RETURNS TRIGGER AS $$

BEGIN

NEW.updated\_at = CURRENT\_TIMESTAMP;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER documents\_updated\_at\_trigger

BEFORE UPDATE ON documents

FOR EACH ROW

EXECUTE FUNCTION update\_documents\_updated\_at();