ФГБОУ ВО

«Заполярный государственный университет им. Н.М. Федоровского»

Кафедра \_\_ИСиТ\_\_

Специальность \_\_ИЭ-21\_\_

ОТЧЕТ

о выполнении лабораторной работы №1

Выполнил:

Чирин Ю. А.

Дата:

« 3 » апреля 2023 г.

*Чирин Ю. А. ИЭ-21*

**Лабораторная работа №1**

**Тема:** проектирование реляционной базы данных PostgreSQL

**Вариант 15.** Обработка результатов ЕГЭ

**Цель работы:** Получение практических навыков проектирования реляционных баз данных и работы в CASE-средстве.

**Задание:** Спроектировать согласно варианту реляционную базу данных, используя методологию IDEF1Х, для этого необходимо:

− построить ER-модель (логическую и физическую) с помощью CASEсредства, например dbdiagram.io или Lucidchart (не менее 7 сущностей), − описать ограничения целостности (CONSTRAINT, UNIQUE)

− вывести программный код создания БД на языке целевой СУБД (прямое проектирование),

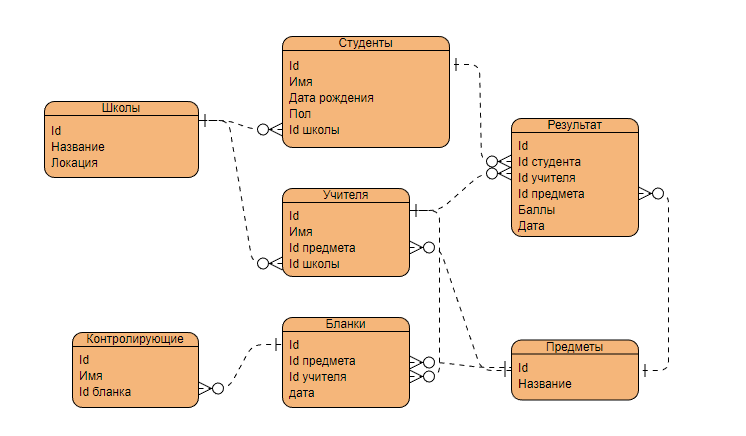
− перенести код в СУБД (выполнить),

− \* внести изменения в схему БД и построить новую физическую модель измененной базы данных (обратное проектирование), если доступно создание ODBC источника данных

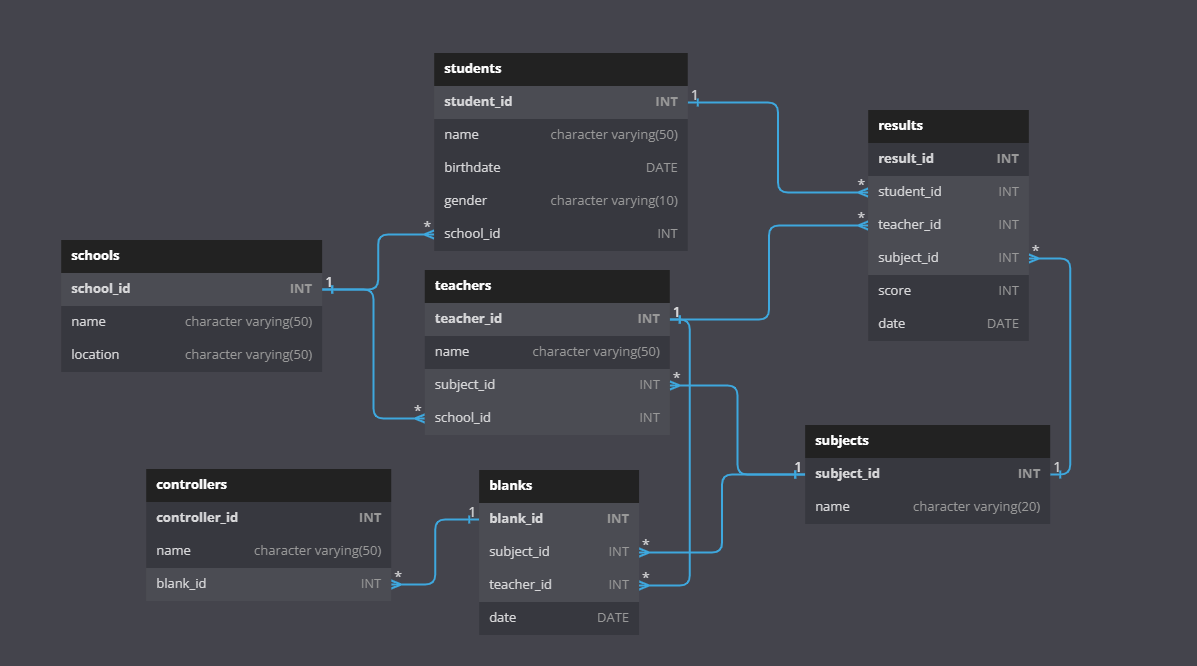
**Описываемая база данных обработки результатов ЕГЭ, выделенные сущности:**

* **Бланки(blanks) –** содержит информацию о бланках для экзаменов, включая уникальный идентификатор, идентификатор преподавателя и идентификатор предмета.
* **Контролирующие(controllers)** – отображает преподавателей, которые контролируют проведение экзамена, включая уникальный идентификатор, идентификатор преподавателя и идентификатор предмета.
* **Предметы(subjects)** – представляет собой дисциплину, по которой проходит экзамен. Каждый предмет имеет уникальный идентификатор и название.
* **Преподаватели(teachers) –** отображает учителей, которые проводят экзамен и оценивают работу студентов. Каждый преподаватель имеет уникальный идентификатор, имя и работает в определенной школе.
* **Результаты(results) –** отображает результаты экзамена, включая уникальный идентификатор, идентификатор студента, идентификатор преподавателя, оценку и идентификатор предмета.
* **Студенты(students) –** представляет собой учеников, которые участвуют в экзамене. Каждый студент имеет уникальный идентификатор, имя и принадлежит к определенной школе.
* **Школы(schools) –** представляет собой образовательную организацию, в которой проходит экзамен. Каждая школа имеет уникальный идентификатор и название.

Логическая модель базы данных:



Физическая модель базы данных:



**Код создания базы данных:**

CREATE TABLE schools (

school\_id INT PRIMARY KEY,

name character varying(50) NOT NULL,

location character varying(50) NOT NULL,

CONSTRAINT schools\_name UNIQUE(name),

CONSTRAINT schools\_location UNIQUE(location)

);

CREATE TABLE subjects (

subject\_id INT PRIMARY KEY,

name character varying(20) NOT NULL

);

CREATE TABLE students (

student\_id INT PRIMARY KEY,

name character varying(50) NOT NULL,

birthdate DATE NOT NULL,

gender character varying(10) NOT NULL,

school\_id INT NOT NULL,

FOREIGN KEY(school\_id) REFERENCES schools(school\_id)

);

CREATE TABLE teachers (

teacher\_id INT PRIMARY KEY,

name character varying(50) NOT NULL,

subject\_id INT NOT NULL,

school\_id INT NOT NULL,

FOREIGN KEY(subject\_id) REFERENCES subjects(subject\_id),

FOREIGN KEY(school\_id) REFERENCES schools(school\_id)

);

CREATE TABLE results (

result\_id INT PRIMARY KEY,

student\_id INT NOT NULL,

teacher\_id INT NOT NULL,

subject\_id INT NOT NULL,

score INT NOT NULL,

date DATE NOT NULL,

FOREIGN KEY(student\_id) REFERENCES students(student\_id),

FOREIGN KEY(teacher\_id) REFERENCES teachers(teacher\_id),

FOREIGN KEY(subject\_id) REFERENCES subjects(subject\_id)

);

CREATE TABLE blanks (

blank\_id INT PRIMARY KEY,

subject\_id INT NOT NULL,

teacher\_id INT NOT NULL,

date DATE NOT NULL,

FOREIGN KEY(subject\_id) REFERENCES subjects(subject\_id),

FOREIGN KEY(teacher\_id) REFERENCES teachers(teacher\_id)

);

CREATE TABLE controllers (

controller\_id INT PRIMARY KEY,

name character varying(50) NOT NULL,

blank\_id INT NOT NULL,

FOREIGN KEY(blank\_id) REFERENCES blanks(blank\_id)

);

**Скриншот создания базы данных:**

