

Лабораторная работа №1

Тема: проектирование реляционной базы данных PostgreSQL

Вариант 15

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

Задание: спроектировать согласно варианту реляционную базу данных, используя методологию IDEF1X, для этого необходимо:

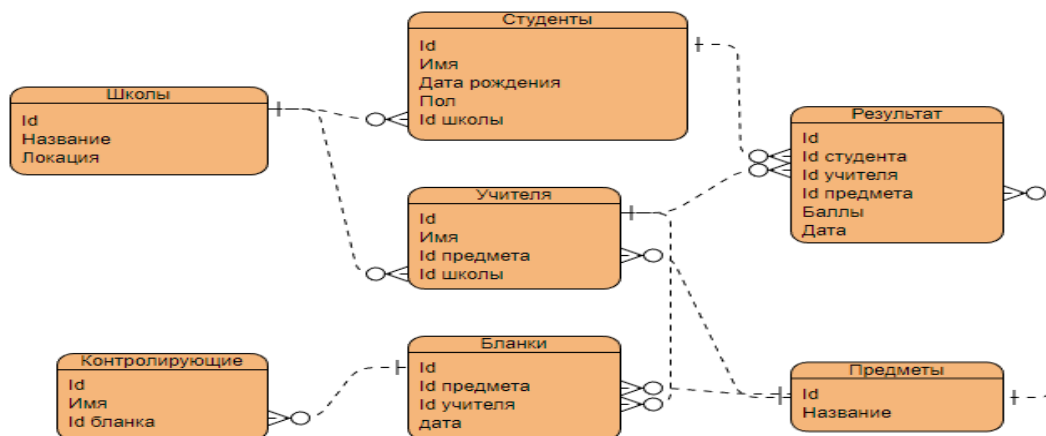
- построить ER-модель (логическую и физическую) с помощью CASEсредства, например dbdiagram.io или Lucidchart (не менее 7 сущностей),
- описать ограничения целостности (CONSTRAINT, UNIQUE)
- вывести программный код создания БД на языке целевой СУБД (прямое проектирование),
- перенести код в СУБД (выполнить),
- * внести изменения в схему БД и построить новую физическую модель измененной базы данных (обратное проектирование), если доступно создание ODBC источника данных

Описываемая база, обработка результатов ЕГЭ, выделенные сущности:

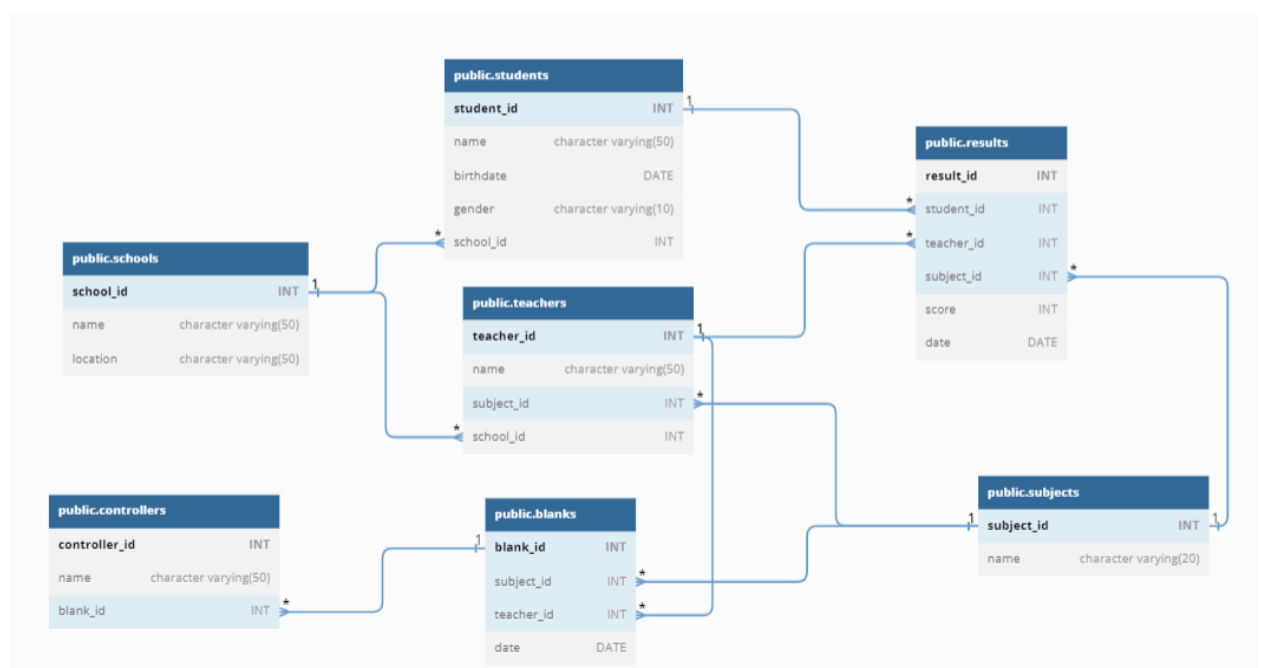
- "Бланк" содержит информацию о бланках для экзаменов, включая уникальный идентификатор, идентификатор преподавателя и идентификатор предмета.
- "Контролирующие" отображает преподавателей, которые контролируют проведение экзамена, включая уникальный идентификатор, идентификатор преподавателя и идентификатор предмета.
- "Предмет" представляет собой дисциплину, по которой проходит экзамен. Каждый предмет имеет уникальный идентификатор и название.
- "Преподаватели" отображает учителей, которые проводят экзамен и оценивают работу студентов. Каждый преподаватель имеет уникальный идентификатор, имя и работает в определенной школе.

- "Результат" отображает результаты экзамена, включая уникальный идентификатор, идентификатор студента, идентификатор преподавателя, оценку и идентификатор предмета.
- Сущность "Студенты" представляет собой учеников, которые участвуют в экзамене. Каждый студент имеет уникальный идентификатор, имя и принадлежит к определенной школе.
- "Школа" 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 БД:

```
CREATE TABLE schools (  
    school_id INT PRIMARY KEY,  
    name character varying(50) NOT NULL,  
    location character varying(50) NOT NULL,  
    CONSTRAINT school_name UNIQUE(name),  
    CONSTRAINT school_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)
);
```

```
INSERT INTO schools
VALUES
('1', 'Школа №1','ул. Пушкина дом 1 корпус 2'),
('2', 'Школа №2','ул. Талнахская дом 30');
```

```
INSERT INTO subjects
VALUES
('1', 'Математика'),
('2', 'Физика'),
('3', 'Химия');
```

```
INSERT INTO teachers
VALUES
('1', 'Иванов Иван Иванович','1', '1'),
('2', 'Петров Петр Петрович','2', '1'),
('3', 'Сидоров Сидор Сидорович','3', '2');
```

INSERT INTO blanks

VALUES

('1', '1', '1','2021-06-02'),

('2', '2', '2','2021-06-03'),

('3', '3', '3','2021-06-08');

INSERT INTO students

VALUES

('1', 'Иванов Иван','2004-03-01','М', '1'),

('2', 'Петрова Мария','2003-05-12','Ж', '1'),

('3', 'Сидоров Сидор','2002-09-29','М', '2');

INSERT INTO controllers

VALUES

('1', 'Иванов Петр Сидорович', '1'),

('2', 'Петров Сидор Иванович', '2'),

('3', 'Сидоров Иван Петрович', '3');

INSERT INTO results

VALUES

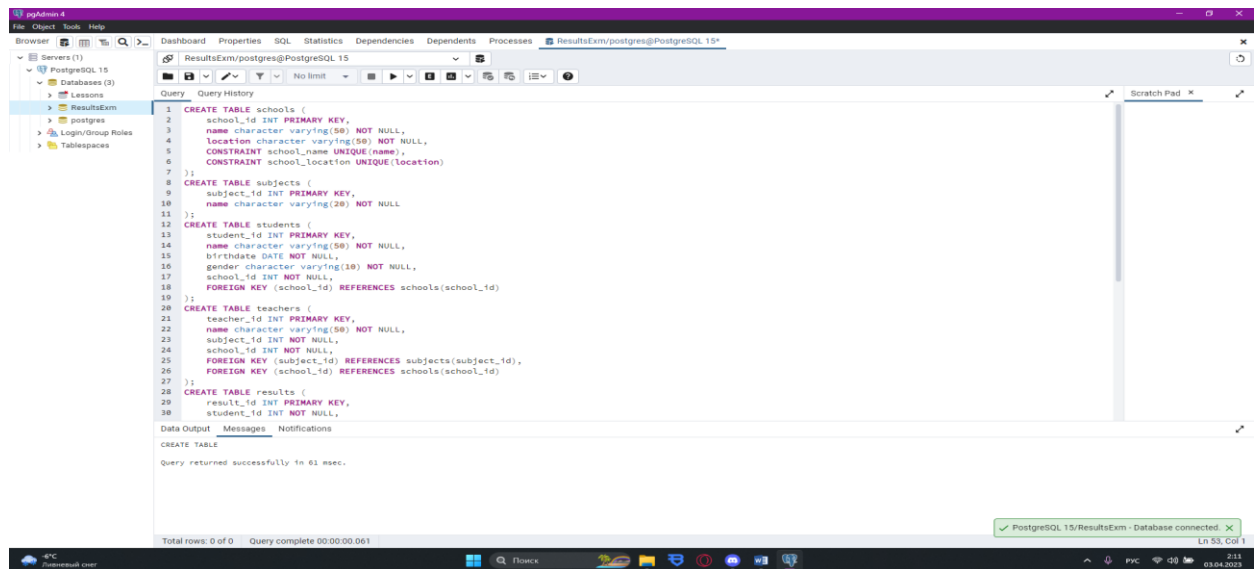
('1', '1', '1', '1', '1','2021-06-02'),

('2', '2', '1', '2', '1','2021-06-03'),

('3', '3', '2', '3', '2','2021-06-08');

Скриншоты выполнения кода:

Create table



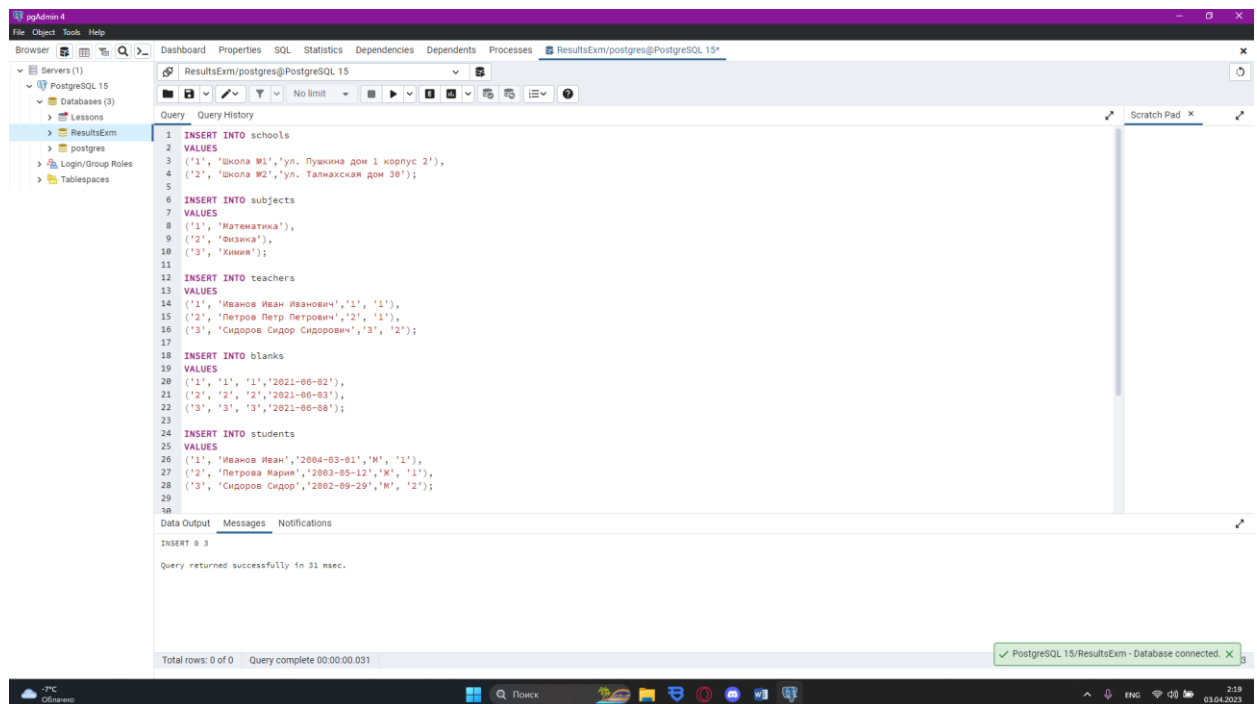
The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure: Servers (1) > PostgreSQL 15 > Databases (3) > ResultsExm. The main window shows the SQL editor with a script to create several tables: schools, subjects, students, teachers, and results. The Results window at the bottom shows the message "Query returned successfully in 61 msec." and "Total rows: 0 of 0".

```
1 CREATE TABLE schools (  
2   school_id INT PRIMARY KEY,  
3   name character varying(50) NOT NULL,  
4   location character varying(50) NOT NULL,  
5   CONSTRAINT school_name UNIQUE(name),  
6   CONSTRAINT school_location UNIQUE(location)  
7 );  
8  
9 CREATE TABLE subjects (  
10  subject_id INT PRIMARY KEY,  
11  name character varying(20) NOT NULL  
12 );  
13  
14 CREATE TABLE students (  
15  student_id INT PRIMARY KEY,  
16  name character varying(50) NOT NULL,  
17  birthdate DATE NOT NULL,  
18  gender character varying(10) NOT NULL,  
19  school_id INT NOT NULL,  
20  FOREIGN KEY (school_id) REFERENCES schools(school_id)  
21 );  
22  
23 CREATE TABLE teachers (  
24  teacher_id INT PRIMARY KEY,  
25  name character varying(50) NOT NULL,  
26  subject_id INT NOT NULL,  
27  school_id INT NOT NULL,  
28  FOREIGN KEY (subject_id) REFERENCES subjects(subject_id),  
29  FOREIGN KEY (school_id) REFERENCES schools(school_id)  
30 );  
31  
32 CREATE TABLE results (  
33  result_id INT PRIMARY KEY,  
34  student_id INT NOT NULL,  
35  teacher_id INT NOT NULL,  
36  subject_id INT NOT NULL,  
37  score INT NOT NULL  
38 );
```

Query returned successfully in 61 msec.

Total rows: 0 of 0 Query complete 00:00:00.061

Insert into [имя таблицы] values



The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure: Servers (1) > PostgreSQL 15 > Databases (3) > ResultsExm. The main window shows the SQL editor with a script to insert data into the tables created in the previous step. The Results window at the bottom shows the message "Query returned successfully in 31 msec." and "Total rows: 0 of 0".

```
1 INSERT INTO schools  
2 VALUES  
3 ('1', 'Школа №1', 'ул. Пушкина дом 1 корпус 2'),  
4 ('2', 'Школа №2', 'ул. Талнахская дом 38');  
5  
6 INSERT INTO subjects  
7 VALUES  
8 ('1', 'Математика'),  
9 ('2', 'Физика'),  
10 ('3', 'Химия');  
11  
12 INSERT INTO teachers  
13 VALUES  
14 ('1', 'Иванов Иван Иванович', '1', '1'),  
15 ('2', 'Петров Петр Петрович', '2', '1'),  
16 ('3', 'Сидоров Сидор Сидорович', '3', '2');  
17  
18 INSERT INTO blanks  
19 VALUES  
20 ('1', '1', '1', '2021-06-02'),  
21 ('2', '2', '2', '2021-06-03'),  
22 ('3', '3', '3', '2021-06-08');  
23  
24 INSERT INTO students  
25 VALUES  
26 ('1', 'Иванов Иван', '2004-03-01', 'М', '1'),  
27 ('2', 'Петрова Мария', '2003-05-12', 'Ж', '1'),  
28 ('3', 'Сидоров Сидор', '2002-09-29', 'М', '2');  
29  
30
```

Query returned successfully in 31 msec.

Total rows: 0 of 0 Query complete 00:00:00.031

ФГБОУ ВО

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

Кафедра __ИСиТ__

Специальность __ИЭ-21__

ОТЧЕТ

о выполнении лабораторной работы

Выполнил:

Чирин Ю.А.

Дата:

« 3 » апреля 2023 г.