

Министерство образования Республики Беларусь

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЕТ  
по лабораторной работе № 3  
на тему  
РЕАЛИЗАЦИЯ SQL-ЗАПРОСОВ ДЛЯ СОЗДАНИЯ СХЕМЫ БАЗЫ  
ДАННЫХ И ЗАПРОСОВ НА МОДИФИКАЦИЮ ДАННЫХ  
«ШКОЛА»

Студент:

А.Н. Климович

Преподаватель:

Д.В. Куприянова

МИНСК 2024

## 1 ЦЕЛЬ РАБОТЫ

Сформировать SQL-запросы для создания таблиц базы данных и выполнить их в СУБД, заполнить таблицы данными с помощью оператора INSERT.

## 2 ИСХОДНЫЕ ДАННЫЕ

В лабораторной работе необходимо реализовать схемы базы данных по ранее построенной в лабораторной работе 2 реляционной схеме данных.

Требуется сформировать SQL-запросы для создания таблиц базы данных и выполнить их в СУБД. Требуется заполнить таблицы данными с помощью оператора INSERT. Реляционная схема данных приведена на рисунке 1.1.

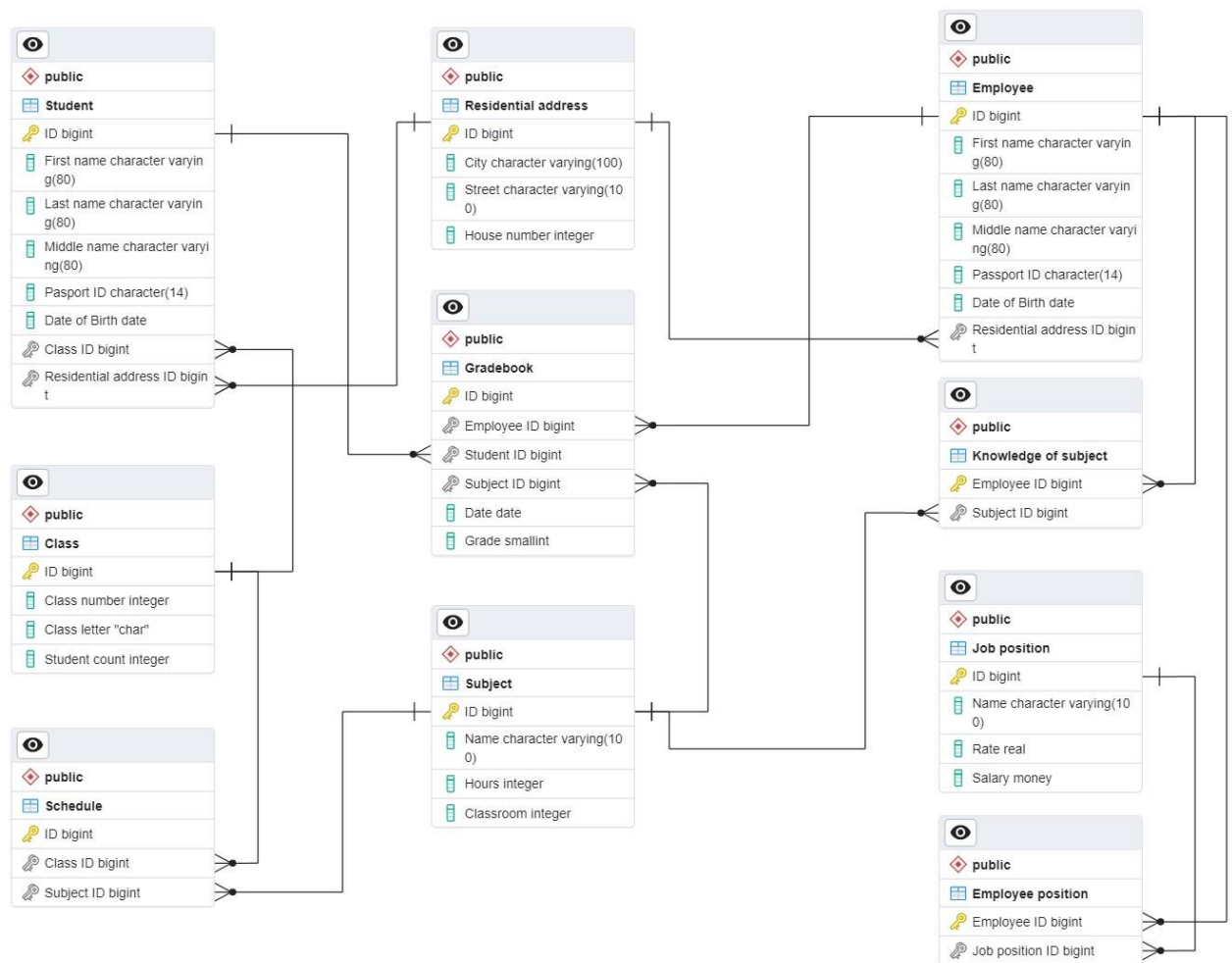


Рисунок 1.1 – Реляционная схема «Школа»

## 3 ВЫПОЛНЕНИЕ РАБОТЫ

### 3.1 Порядок выполнения работы

Создать в СУБД новую схему данных для хранения пользовательских объектов.

1. В этой новой схеме данных с помощью скрипта с запросами на языке DDL SQL реализовать таблицы, соответствующие реляционным отношениям схемы данных полученной в лабораторной работе №2 (см. рисунок 3.1), с помощью одного (желательно) оператора `CREATE TABLE` для каждой таблицы в следующем порядке:

- реализовать простую структуру таблиц, включающую только набор столбцов с добавлением описаний первичного ключа;
- дополнить описание таблицы реализацией ограничений для описания внешних ключей; для внешних ключей установить свойства контроля целостности данных (каскадное удаление и обновление), если это возможно в целевой СУБД;
- дополнить описание таблицы реализацией ограничений для описания бизнес-правил;
- дополнить описание таблицы реализацией комментариев для значимых элементов таблицы.

2. Заполнить с помощью SQL-скрипта с использованием оператора `INSERT` таблицы строками данных для проверки правильного выбора первичных ключей и работоспособности ссылок между таблицами:

- строками данных сначала заполнять мастер-таблицы (или таблицы, которые НЕ ссылаются на другие таблицы);
- в каждую таблицу добавить 30 строк осмысленных данных;
- если не удастся добавить данные в таблицу по причине нарушения уникальности первичного ключа, то следует перепроверить описание этого первичного ключа и его смысл для реального мира;
- если не удастся добавить данные в таблицу по причине нарушения ссылочной целостности, то следует убедиться, что целевые данные существуют, иначе перепроверить описание внешнего ключа.

3. Рассмотреть простые действия по изменению структуры таблицы (переименование столбца таблицы, добавление и удаление ограничений на столбец таблицы или всю таблицу) и реализовать их с помощью оператора `ALTER TABLE`.

4. Создать временную таблицу с помощью оператора `CREATE TABLE` и удалить ее с помощью оператора `DROP TABLE`.

5. Экспортировать результаты работы в SQL-скрипт, сравнить полученный скрипт со скриптами, созданными на этапах 2 и 3.

## 3.2 Работа с таблицами в SQL

Для начала нужно создать новую схему данных для хранения пользовательских объектов. Для этого воспользуемся средствами пользовательского интерфейса:

Сначала кликнем правой кнопкой мыши (ПКМ) на Schemas (см. рисунок 3.1).

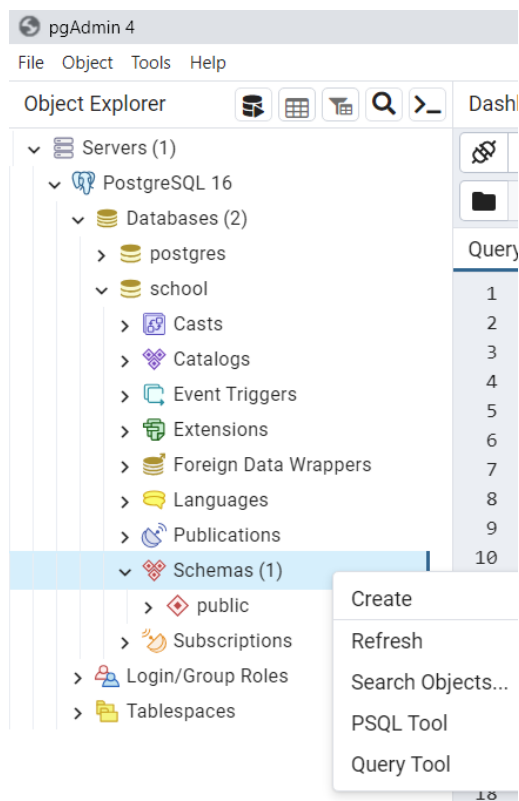


Рисунок 3.1 – Клик по Schemas для создания схемы данных

Далее в выпадающем списке выберем «Create->Schema...». Откроется окно создания схемы данных (см. рисунок 3.2), где нужно ввести название схемы данных, после чего создастся новая схема данных.

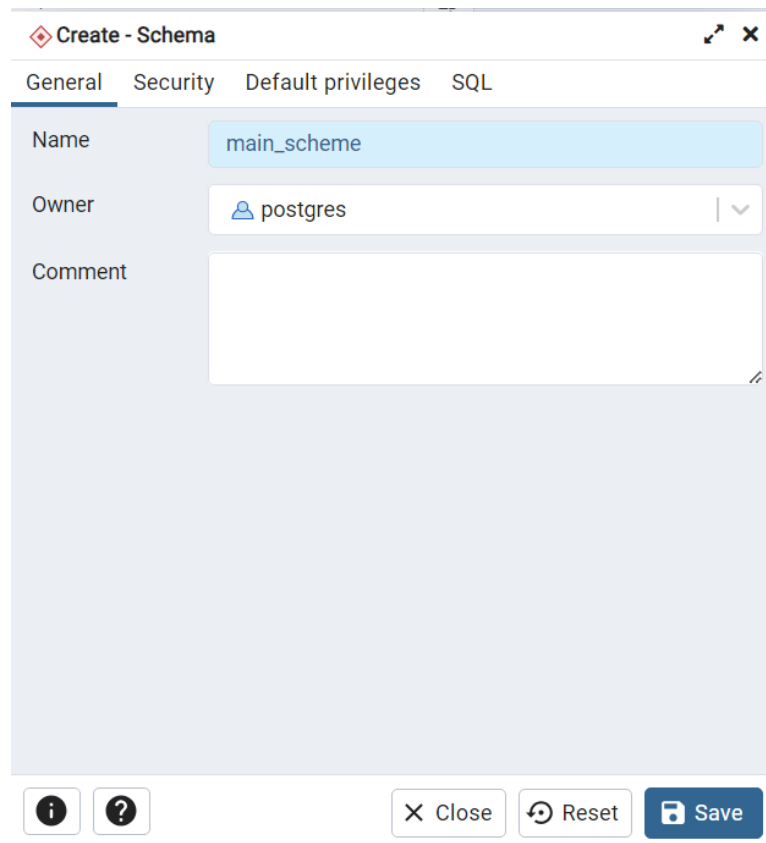


Рисунок 3.2 – Окно создания схемы данных

Для создания таблиц был написан следующий SQL-скрипт:

```
DROP TABLE IF EXISTS main_scheme.student CASCADE;
DROP TABLE IF EXISTS main_scheme.employee CASCADE;
DROP TABLE IF EXISTS main_scheme.residential_address CASCADE;
DROP TABLE IF EXISTS main_scheme.job_position CASCADE;
DROP TABLE IF EXISTS main_scheme.gradebook CASCADE;
DROP TABLE IF EXISTS main_scheme.class CASCADE;
DROP TABLE IF EXISTS main_scheme.schedule CASCADE;
DROP TABLE IF EXISTS main_scheme.subject CASCADE;
DROP TABLE IF EXISTS main_scheme.employee_position CASCADE;
DROP TABLE IF EXISTS main_scheme.knowledge_of_subject CASCADE;

CREATE TABLE IF NOT EXISTS main_scheme.student
(
    id SERIAL PRIMARY KEY,
    first_name CHARACTER VARYING(100) NOT NULL,
    last_name CHARACTER VARYING(100) NOT NULL,
    middle_name CHARACTER VARYING(100) NOT NULL,
    passport_id CHARACTER(14) UNIQUE NOT NULL,
    date_of_birth DATE NOT NULL,
    class_id BIGINT NOT NULL,
    residential_address_id BIGINT NOT NULL
);
CREATE TABLE IF NOT EXISTS main_scheme.class
(
    id SERIAL PRIMARY KEY,
    class_number INTEGER NOT NULL,
```

```

        class_letter      CHAR NOT NULL,
        student_count     INTEGER NOT NULL
    );

CREATE TABLE IF NOT EXISTS main_scheme.subject
(
    id          SERIAL PRIMARY KEY,
    name        CHARACTER VARYING(100) NOT NULL,
    hours       INTEGER DEFAULT(0) CHECK(hours >= 0),
    classroom   INTEGER NOT NULL
);

CREATE TABLE IF NOT EXISTS main_scheme.residential_address
(
    id          SERIAL PRIMARY KEY,
    city        CHARACTER VARYING(100) NOT NULL,
    street      CHARACTER VARYING(100) NOT NULL,
    house_number INTEGER CHECK(house_number >= 1)
);

CREATE TABLE IF NOT EXISTS main_scheme.job_position
(
    id          SERIAL PRIMARY KEY,
    name        CHARACTER VARYING(100) NOT NULL,
    rate        REAL CHECK(rate >= 0) NOT NULL,
    salary      MONEY NOT NULL
);

CREATE TABLE IF NOT EXISTS main_scheme.schedule
(
    id          SERIAL PRIMARY KEY,
    class_id    BIGINT NOT NULL,
    subject_id  BIGINT NOT NULL
);

CREATE TABLE IF NOT EXISTS main_scheme.employee
(
    id          SERIAL PRIMARY KEY,
    first_name  CHARACTER VARYING(100) NOT NULL,
    last_name   CHARACTER VARYING(100) NOT NULL,
    middle_name CHARACTER VARYING(100) NOT NULL,
    passport_id CHARACTER(14) UNIQUE NOT NULL,
    date_of_birth DATE NOT NULL,
    residential_address_id BIGINT NOT NULL
);

CREATE TABLE IF NOT EXISTS main_scheme.knowledge_of_subject
(
    employee_id BIGINT PRIMARY KEY,
    subject_id  BIGINT NOT NULL
);

CREATE TABLE IF NOT EXISTS main_scheme.employee_position
(
    employee_id    BIGINT PRIMARY KEY,
    job_position_id BIGINT NOT NULL
);

```

```

);

CREATE TABLE IF NOT EXISTS main_scheme.gradebook
(
    id SERIAL PRIMARY KEY,
    employee_id BIGINT NOT NULL,
    student_id BIGINT NOT NULL,
    subject_id BIGINT NOT NULL,
    grade_date DATE NOT NULL,
    grade SMALLINT NOT NULL
);

ALTER TABLE IF EXISTS main_scheme.student
ADD CONSTRAINT fk_class
    FOREIGN KEY (class_id)
    REFERENCES main_scheme.class (id)
    ON DELETE CASCADE;

ALTER TABLE IF EXISTS main_scheme.student
ADD CONSTRAINT fk_student_residential_address
    FOREIGN KEY (residential_address_id)
    REFERENCES main_scheme.residential_address (id)
    ON DELETE CASCADE;

ALTER TABLE IF EXISTS main_scheme.schedule
ADD CONSTRAINT fk_schedule
    FOREIGN KEY (class_id)
    REFERENCES main_scheme.class (id)
    ON DELETE CASCADE;

ALTER TABLE IF EXISTS main_scheme.schedule
ADD CONSTRAINT fk_subject
    FOREIGN KEY (subject_id)
    REFERENCES main_scheme.subject (id)
    ON DELETE CASCADE;

ALTER TABLE IF EXISTS main_scheme.employee
ADD CONSTRAINT fk_employee_residential_address
    FOREIGN KEY (residential_address_id)
    REFERENCES main_scheme.residential_address (id)
    ON DELETE CASCADE;

ALTER TABLE IF EXISTS main_scheme.knowledge_of_subject
ADD CONSTRAINT fk_subject
    FOREIGN KEY (subject_id)
    REFERENCES main_scheme.subject (id)
    ON DELETE CASCADE;

```

```
ALTER TABLE IF EXISTS main_scheme.knowledge_of_subject
ADD CONSTRAINT fk_employee
    FOREIGN KEY (employee_id)
    REFERENCES main_scheme.employee (id)
    ON DELETE CASCADE;
```

```
ALTER TABLE IF EXISTS main_scheme.employee_position
ADD CONSTRAINT fk_employee
    FOREIGN KEY (employee_id)
    REFERENCES main_scheme.employee (id)
    ON DELETE CASCADE;
```

```
ALTER TABLE IF EXISTS main_scheme.employee_position
ADD CONSTRAINT fk_job_position
    FOREIGN KEY (job_position_id)
    REFERENCES main_scheme.job_position (id)
    ON DELETE CASCADE;
```

```
ALTER TABLE IF EXISTS main_scheme.gradebook
ADD CONSTRAINT fk_subject
    FOREIGN KEY (subject_id)
    REFERENCES main_scheme.subject (id)
    ON DELETE CASCADE;
```

```
ALTER TABLE IF EXISTS main_scheme.gradebook
ADD CONSTRAINT fk_employee
    FOREIGN KEY (employee_id)
    REFERENCES main_scheme.employee (id)
    ON DELETE CASCADE;
```

```
ALTER TABLE IF EXISTS main_scheme.gradebook
ADD CONSTRAINT fk_student
    FOREIGN KEY (student_id)
    REFERENCES main_scheme.student (id)
    ON DELETE CASCADE;
```

```
COMMENT ON COLUMN main_scheme.gradebook.grade IS 'Возможные значения:
от 1 до 10';
```

**Для заполнения созданных таблиц был создан следующий SQL-скрипт:**

```
INSERT INTO main_scheme.residential_address (city, street,
house_number)
VALUES ('Нью-Йорк', 'Бродвей', 123),
       ('Лос-Анджелес', 'Сансет-бульвар', 456),
       ('Чикаго', 'Мичиган-авеню', 789),
       ('Хьюстон', 'Мэйн-стрит', 321),
       ('Финикс', 'Центральная авеню', 654),
       ('Филадельфия', 'Маркет-стрит', 987),
       ('Сан-Антонио', 'Риверволк', 147),
```



```

('Сан-Диего', 'Оушен-бульвар', 258),
('Даллас', 'Элм-стрит', 369),
('Сан-Хосе', 'Первая улица', 753),
('Остин', 'Конгресс-авеню', 852),
('Джэксонвилл', 'Бей-стрит', 963),
('Индианаполис', 'Монумент-серкуль', 159),
('Сан-Франциско', 'Ломбард-стрит', 357),
('Колумбус', 'Хай-стрит', 468),
('Форт-Уэрт', 'Санданс-сквер', 579),
('Шарлотт', 'Трейд-стрит', 684),
('Сиэтл', 'Пайк-стрит', 795),
('Денвер', 'Колфакс-авеню', 876),
('Вашингтон', 'Пенсильвания-авеню', 987),
('Бостон', 'Ньюбери-стрит', 369),
('Нэшвилл', 'Бродвей', 753),
('Балтимор', 'Иннер-Харбор', 852),
('Оклахома-Сити', 'Бриктаун', 963),
('Портленд', 'Пёрл-дистрикт', 159),
('Лас-Вегас', 'Стрип', 357),
('Милуоки', 'Уотер-стрит', 468),
('Альбукерке', 'Шоссе 66', 579),
('Тусон', 'Четвертая авеню', 684),
('Майами', 'Оушен Драйв', 795),
('Минск', 'Независимости', 123),
('Логойск', 'Интернациональная', 456),
('Витебск', 'Космонавтов', 789),
('Гродно', 'Малинина', 321),
('Гомель', 'Могилевская', 654),
('Гомель', 'Луговая', 987),
('Несвиж', 'Первомайская', 147),
('Мир', 'Пушкинская', 258),
('Брест', 'Деревянная', 369),
('Кобрин', 'Цветочная', 753),
('Мозырь', 'Победителей', 852),
('Орша', 'Центральная', 963),
('Орша', 'Центральная', 159),
('Кобрин', 'Лесная', 357),
('Клецк', 'Домашняя', 468),
('Несвиж', 'Московская', 579),
('Солигорск', 'Солевая', 684),
('Держинск', 'Горная', 795),
('Фаниполь', 'Клубничная', 876),
('Лунинец', 'Солнечная', 987),
('Калинковичи', 'Карла Маркса', 369),
('Могилев', 'Центральная', 753),
('Климовичи', 'Лесная', 852),
('Минск', 'Каменная', 963),
('Минск', 'Космонавтов', 159),
('Минск', 'Железная', 357),
('Марьяна Горка', 'Интернациональная', 468),
('Волковыск', 'Советская', 579),
('Гродно', 'Полевая', 684),
('Минск', 'Мирная', 795);

```

```

INSERT INTO main_scheme.class (class_number, class_letter,
student_count)

```

```
VALUES (10, 'А', 24),
       (3, 'Б', 27),
       (8, 'В', 32),
       (4, 'Г', 28),
       (2, 'Д', 26),
       (6, 'Е', 30),
       (7, 'Ж', 31),
       (5, 'З', 29),
       (1, 'И', 25),
       (11, 'К', 25),
       (9, 'Л', 13),
       (10, 'М', 19),
       (1, 'Н', 18),
       (11, 'О', 26),
       (11, 'В', 25),
       (6, 'П', 14),
       (3, 'Р', 7),
       (2, 'С', 15),
       (8, 'Т', 13),
       (7, 'У', 14),
       (9, 'Ф', 14),
       (2, 'Х', 7),
       (2, 'Ш', 8),
       (2, 'Ц', 16),
       (4, 'Ч', 9),
       (6, 'Я', 11),
       (7, 'Э', 12),
       (5, 'Щ', 10),
       (3, 'Ы', 15),
       (9, 'А', 14),
       (10, 'Б', 14),
       (8, 'Б', 13);
```

```
INSERT INTO main_scheme.subject (name, hours, classroom)
VALUES ('Математика', 60, 101),
       ('Физика', 45, 102),
       ('История', 30, 103),
       ('Литература', 45, 104),
       ('Биология', 60, 105),
       ('Химия', 45, 106),
       ('География', 30, 107),
       ('Английский язык', 60, 108),
       ('Французский язык', 45, 109),
       ('Немецкий язык', 30, 110),
       ('Испанский язык', 45, 111),
       ('Информатика', 60, 112),
       ('Физкультура', 30, 113),
       ('Музыка', 45, 114),
       ('ИЗО', 60, 115),
       ('Трудовое обучение', 30, 116),
       ('Обществознание', 45, 117),
       ('Право', 60, 118),
       ('Экономика', 30, 119),
       ('Психология', 45, 120),
       ('Философия', 60, 121),
       ('Религиоведение', 30, 122),
```

```

('Хореография', 45, 123),
('Театральное искусство', 60, 124),
('Основы медицины', 30, 125),
('Дизайн', 45, 126),
('Астрономия', 60, 127),
('Геология', 30, 128),
('Архитектура', 45, 129),
('Лингвистика', 60, 130);

INSERT INTO main_scheme.job_position (name, rate, salary)
VALUES ('Директор', 1.0, 100000),
       ('Заместитель директора', 0.9, 90000),
       ('Учитель математики', 0.8, 80000),
       ('Учитель русского языка', 0.8, 80000),
       ('Учитель физики', 0.8, 80000),
       ('Учитель химии', 0.8, 80000),
       ('Учитель биологии', 0.8, 80000),
       ('Учитель истории', 0.8, 80000),
       ('Учитель географии', 0.8, 80000),
       ('Учитель английского языка', 0.8, 80000),
       ('Учитель физической культуры', 0.8, 80000),
       ('Учитель искусства', 0.8, 80000),
       ('Учитель музыки', 0.8, 80000),
       ('Учитель информатики', 0.8, 80000),
       ('Учитель технологии', 0.8, 80000),
       ('Учитель начальных классов', 0.8, 80000),
       ('Педагог-психолог', 0.7, 70000),
       ('Социальный педагог', 0.7, 70000),
       ('Логопед', 0.7, 70000),
       ('Библиотекарь', 0.7, 70000),
       ('Администратор', 0.6, 60000),
       ('Секретарь', 0.6, 60000),
       ('Бухгалтер', 0.6, 60000),
       ('Сторож', 0.5, 50000),
       ('Уборщик', 0.5, 50000),
       ('Воспитатель', 0.7, 70000),
       ('Медсестра', 0.6, 60000),
       ('Дежурный по физической культуре', 0.7, 70000),
       ('Дежурный по охране', 0.6, 60000),
       ('Дежурный по уборке', 0.5, 50000);

INSERT INTO main_scheme.student (first_name, last_name, middle_name,
passport_id, date_of_birth, class_id, residential_address_id) VALUES
('Иван', 'Иванов', 'Иванович', 'AB1234567CD89', '2007-05-10', 1, 1),
('Петр', 'Петров', 'Петрович', 'EF9876543GH21', '2008-02-15', 2, 2),
('Анна', 'Сидорова', 'Александровна', 'KL4567890MN32',
'2009-08-20', 3, 3),
('Мария', 'Козлова', 'Игоревна', 'OP6543210QR54', '2010-07-
25', 4, 4),
('Сергей', 'Васильев', 'Дмитриевич', 'ST0987654UV76', '2011-
03-30', 5, 5),
('Елена', 'Никитина', 'Павловна', 'WX2345678YZ98', '2012-11-
05', 6, 6),
('Алексей', 'Григорьев', 'Андреевич', 'CD8765432AB09',
'2013-09-12', 7, 7),
('Ольга', 'Павлова', 'Сергеевна', 'EF2109876GH43', '2014-04-
18', 8, 8),

```

```

        ('Дмитрий', 'Федоров', 'Владимирович', 'IJ0987654KL76',
'2015-12-23', 9, 9),
        ('Наталья', 'Иванова', 'Михайловна', 'MN5432109OP65', '2016-
10-28', 10, 10),
        ('Ирина', 'Петрова', 'Алексеевна', 'QR7654321ST87', '2017-
06-02', 11, 11),
        ('Андрей', 'Сидоров', 'Петрович', 'UV1098765WX32', '2018-01-
07', 12, 12),
        ('Виктория', 'Смирнова', 'Анатољевна', 'YZ8765432WX10',
'2007-09-12', 13, 13),
        ('Артем', 'Кузнецов', 'Иванович', 'BC5432109DE87', '2008-07-
17', 14, 14),
        ('Татьяна', 'Морозова', 'Алексеевна', 'FG3210987HI54',
'2009-04-22', 15, 15),
        ('Григорий', 'Новиков', 'Петрович', 'JK1098765LM32', '2010-
12-27', 16, 16),
        ('Василиса', 'Зайцева', 'Дмитриевна', 'NO9876543PQ21',
'2011-10-02', 17, 17),
        ('Егор', 'Павлов', 'Сергеевич', 'RS7654321TU98', '2012-05-
08', 18, 18),
        ('Людмила', 'Соколова', 'Игоревна', 'VW5432109XY76', '2013-
01-13', 19, 19),
        ('Александра', 'Иванова', 'Александровна', 'AB2109876CD43',
'2014-08-18', 20, 20),
        ('Денис', 'Петров', 'Владимирович', 'EF8765432GH65', '2015-
06-23', 21, 21),
        ('Екатерина', 'Сидорова', 'Михайловна', 'IJ6543210KL87',
'2016-02-28', 22, 22),
        ('Антон', 'Козлов', 'Павлович', 'MN4321098OP54', '2017-11-
03', 23, 23),
        ('Олег', 'Васильев', 'Андреевич', 'QR0987654ST32', '2018-07-
08', 24, 24),
        ('Марина', 'Григорьева', 'Петровна', 'UV7654321WX09', '2007-
03-13', 25, 25),
        ('Николай', 'Никитин', 'Сергеевич', 'CD3210987AB76', '2008-
12-18', 26, 26),
        ('Елена', 'Федорова', 'Ивановна', 'GH9876543IJ43', '2009-10-
23', 27, 27),
        ('Игорь', 'Иванов', 'Александрович', 'KL6543210MN65', '2010-
06-28', 28, 28),
        ('Лариса', 'Петрова', 'Владимировна', 'OP2109876QR87',
'2011-02-03', 29, 29),
        ('Владимир', 'Сидоров', 'Михайлович', 'ST4321098UV54',
'2012-09-08', 30, 30),
        ('Анастасия', 'Козлова', 'Павловна', 'WX6543210YZ32', '2013-
07-13', 21, 30);

```

```

INSERT INTO main_scheme.schedule (class_id, subject_id)
VALUES (1, 1),
        (2, 2),
        (3, 3),
        (4, 4),
        (5, 5),
        (6, 6),
        (7, 7),
        (8, 8),

```

```

(9, 9),
(10, 10),
(11, 11),
(12, 12),
(13, 13),
(14, 14),
(15, 15),
(16, 16),
(17, 17),
(18, 18),
(19, 19),
(20, 20),
(21, 21),
(22, 22),
(23, 23),
(24, 24),
(25, 25),
(26, 26),
(27, 27),
(28, 28),
(29, 29),
(30, 30);

```

```

INSERT INTO main_scheme.employee (first_name, last_name, middle_name,
passport_id, date_of_birth, residential_address_id)
VALUES ('Иван', 'Глецевич', 'Иванович', 'AB1234547CD89', '1978-05-
15', 31),
      ('Петр', 'Петров', 'Петрович', 'EF9876543UH21', '1998-02-
15', 42),
      ('Анна', 'Сидорова', 'Александровна', 'KL6567890MN32',
'1989-08-20', 43),
      ('Мария', 'Козлова', 'Игоревна', 'OP6543230QR54', '1971-07-
25', 34),
      ('Сергей', 'Васильев', 'Дмитриевич', 'ST0187654UV76', '1990-
03-30', 35),
      ('Елена', 'Никитина', 'Павловна', 'WX2345478YZ98', '1980-11-
05', 36),
      ('Алексей', 'Тригорьев', 'Андреевич', 'CD4765432AB09',
'1981-09-12', 37),
      ('Ольга', 'Павлова', 'Сергеевна', 'EF2109376GH43', '1982-04-
18', 38),
      ('Дмитрий', 'Федоров', 'Владимирович', 'IK0987654KL76',
'1982-12-23', 39),
      ('Наталья', 'Иванова', 'Михайловна', 'MN5332109OP65', '1985-
10-28', 31),
      ('Ирина', 'Петрова', 'Алексеевна', 'QR7653321ST87', '1989-
06-02', 41),
      ('Андрей', 'Сидоров', 'Петрович', 'UV1098365WX32', '1975-01-
07', 42),
      ('Виктория', 'Смирнова', 'Анатольевна', 'ZZ8765432WX10',
'2000-09-12', 33),
      ('Артем', 'Кузнецов', 'Иванович', 'BC5432409DE87', '1999-07-
17', 14),
      ('Татьяна', 'Морозова', 'Алексеевна', 'FG4210987HI54',
'1973-04-22', 25),

```

```

('Григорий', 'Новиков', 'Петрович', 'JK1058765LM32', '1974-
12-27', 16),
('Василиса', 'Зайцева', 'Дмитриевна', 'NO5876543PQ21',
'1987-10-02', 57),
('Егор', 'Павлов', 'Сергеевич', 'RS7654320TU98', '1984-05-
08', 58),
('Людмила', 'Соколова', 'Игоревна', 'VW5442109XY76', '1966-
01-13', 49),
('Александра', 'Иванова', 'Александровна', 'AB2409876CD43',
'1983-08-18', 30),
('Денис', 'Петров', 'Владимирович', 'EF8765462GH65', '1979-
06-23', 44),
('Екатерина', 'Сидорова', 'Михайловна', 'IJ6643210KL87',
'1970-02-28', 42),
('Антон', 'Козлов', 'Павлович', 'MN4321098OP64', '1998-11-
03', 43),
('Олег', 'Васильев', 'Андреевич', 'QR0987664ST32', '1996-07-
08', 54),
('Марина', 'Григорьева', 'Петровна', 'UV7634321WX09', '1991-
03-13', 54),
('Николай', 'Никитин', 'Сергеевич', 'CD3213987AB76', '2001-
12-18', 56),
('Елена', 'Федорова', 'Ивановна', 'GH9876533IJ43', '2000-10-
23', 57),
('Игорь', 'Иванов', 'Александрович', 'KL6533210MN65', '2000-
06-28', 48),
('Лариса', 'Петрова', 'Владимировна', 'OP2309876QR87',
'1972-02-03', 49),
('Владимир', 'Сидоров', 'Михайлович', 'ST4921098UV54',
'1994-09-08', 43),
('Анастасия', 'Козлова', 'Павловна', 'WX6549210YZ32', '1995-
07-13', 39);

```

```

INSERT INTO main_scheme.gradebook (employee_id, student_id,
subject_id, grade_date, grade)
VALUES (1, 1, 1, '2023-01-15', 8),
(2, 2, 2, '2023-02-20', 7),
(3, 3, 3, '2023-03-25', 9),
(4, 4, 1, '2023-04-30', 6),
(5, 5, 2, '2023-05-05', 8),
(6, 6, 3, '2023-06-10', 7),
(7, 7, 1, '2023-07-15', 9),
(8, 8, 2, '2023-08-20', 5),
(9, 9, 3, '2023-09-25', 8),
(1, 1, 1, '2023-10-30', 7),
(1, 1, 2, '2023-11-05', 6),
(1, 2, 3, '2023-12-10', 9),
(3, 3, 1, '2024-01-15', 8),
(4, 4, 2, '2024-02-20', 7),
(5, 5, 3, '2024-03-25', 9),
(6, 6, 1, '2024-04-30', 6),
(7, 7, 2, '2024-05-05', 8),
(8, 8, 3, '2024-06-10', 7),
(1, 9, 1, '2024-07-15', 9),
(2, 2, 2, '2024-08-20', 5),
(2, 2, 3, '2024-09-25', 8),

```

```

(2, 2, 1, '2024-10-30', 7),
(2, 3, 2, '2024-11-05', 6),
(4, 4, 3, '2024-12-10', 9),
(5, 12, 1, '2023-01-15', 8),
(6, 12, 2, '2023-02-20', 7),
(7, 17, 3, '2023-03-25', 9),
(8, 18, 1, '2023-04-30', 6),
(9, 19, 2, '2023-05-05', 8),
(3, 13, 3, '2023-06-10', 7);

```

```

INSERT INTO main_scheme.knowledge_of_subject (employee_id, subject_id)
VALUES (1, 1),

```

```

(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10),
(11, 11),
(12, 12),
(13, 13),
(14, 14),
(15, 15),
(16, 16),
(17, 17),
(18, 18),
(19, 19),
(20, 20),
(21, 21),
(22, 22),
(23, 23),
(24, 24),
(25, 25),
(26, 26),
(27, 27),
(28, 28),
(29, 29),
(30, 30);

```

```

INSERT INTO main_scheme.employee_position (employee_id,
job_position_id)

```

```

VALUES (1, 1),
(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10),
(11, 11),

```

```
(12, 12),
(13, 13),
(14, 14),
(15, 15),
(16, 16),
(17, 17),
(18, 18),
(19, 19),
(20, 20),
(21, 21),
(22, 22),
(23, 23),
(24, 24),
(25, 25),
(26, 26),
(27, 27),
(28, 28),
(29, 29),
(30, 30);
```

Результаты скрипта представлены на рисунках 3.3 – 3.12.

	id [PK] integer	class_number integer	class_letter character	student_count integer
1	1	10	A	24
2	2	3	Б	27
3	3	8	В	32
4	4	4	Г	28
5	5	2	Д	26
6	6	6	Е	30
7	7	7	Ж	31
8	8	5	З	29
9	9	1	И	25
10	10	11	К	25
11	11	9	Л	13
12	12	10	М	19
13	13	1	Н	18
14	14	11	О	26
15	15	11	В	25
16	16	6	П	14
17	17	3	Р	7
18	18	2	С	15
19	19	8	Т	13
20	20	7	У	14
21	21	9	Ф	14
22	22	2	Х	7



23	23	2	Ш	8
24	24	2	Ц	16
25	25	4	Ч	9
26	26	6	Я	11
27	27	7	Э	12
28	28	5	Щ	10
29	29	3	Ы	15
30	30	9	А	14
31	31	10	Б	14
32	32	8	Б	13
Total rows: 32 of 32		Query complete 00:00:00.167		

Рисунок 3.3 – Таблица class

id [PK] integer	first_name character varying (100)	last_name character varying (100)	middle_name character varying (100)	passport_id character	date_of_birth date	residential_address_id bigint
1	Иван	Глецевич	Иванович	AB1234547CD89	1978-05-15	31
2	Петр	Петров	Петрович	EF9876543UH21	1998-02-15	42
3	Анна	Сидорова	Александровна	KL6567890MN32	1989-08-20	43
4	Мария	Козлова	Игоревна	OP6543230QR54	1971-07-25	34
5	Сергей	Васильев	Дмитриевич	ST0187654UV76	1990-03-30	35
6	Елена	Никитина	Павловна	WX2345478YZ98	1980-11-05	36
7	Алексей	Григорьев	Андреевич	CD4765432AB09	1981-09-12	37
8	Ольга	Павлова	Сергеевна	EF2109376GH43	1982-04-18	38
9	Дмитрий	Федоров	Владимирович	IK0987654KL76	1982-12-23	39
10	Наталья	Иванова	Михайловна	MN5332109OP65	1985-10-28	31
11	Ирина	Петрова	Алексеевна	QR7653321ST87	1989-06-02	41
12	Андрей	Сидоров	Петрович	UV1098365WX32	1975-01-07	42
13	Виктория	Смирнова	Анатольевна	ZZ8765432WX10	2000-09-12	33
14	Артем	Кузнецов	Иванович	BC5432409DE87	1999-07-17	14
15	Татьяна	Морозова	Алексеевна	FG4210987HI54	1973-04-22	25
16	Григорий	Новиков	Петрович	JK1058765LM32	1974-12-27	16
17	Василиса	Зайцева	Дмитриевна	NO5876543PQ21	1987-10-02	57
18	Егор	Павлов	Сергеевич	RS7654320TU98	1984-05-08	58
19	Людмила	Соколова	Игоревна	VW5442109XY76	1966-01-13	49
20	Александра	Иванова	Александровна	AB2409876CD43	1983-08-18	30
21	Денис	Петров	Владимирович	EF8765462GH65	1979-06-23	44
22	Екатерина	Сидорова	Михайловна	IJ6643210KL87	1970-02-28	42
23	Антон	Козлов	Павлович	MN4321098OP64	1998-11-03	43
24	Олег	Васильев	Андреевич	QR0987664ST32	1996-07-08	54
25	Марина	Григорьева	Петровна	UV7634321WX09	1991-03-13	54
26	Николай	Никитин	Сергеевич	CD3213987AB76	2001-12-18	56
27	Елена	Федорова	Ивановна	GH9876533IJ43	2000-10-23	57
28	Игорь	Иванов	Александрович	KL6533210MN65	2000-06-28	48
29	Лариса	Петрова	Владимировна	OP2309876QR87	1972-02-03	49
30	Владимир	Сидоров	Михайлович	ST4921098UV54	1994-09-08	43
31	Анастасия	Козлова	Павловна	WX6549210YZ32	1995-07-13	39
Total rows: 31 of 31	Query complete 00:00:00.252					

Рисунок 3.4 – Таблица employee

	employee_id [PK] bigint	job_position_id bigint
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23
24	24	24
25	25	25
26	26	26
27	27	27
28	28	28
29	29	29
30	30	30

Рисунок 3.5 – Таблица employee\_position

	id [PK] integer	employee_id bigint	student_id bigint	subject_id bigint	grade_date date	grade smallint
1	1	1	1	1	2023-01-15	8
2	2	2	2	2	2023-02-20	7
3	3	3	3	3	2023-03-25	9
4	4	4	4	1	2023-04-30	6
5	5	5	5	2	2023-05-05	8
6	6	6	6	3	2023-06-10	7
7	7	7	7	1	2023-07-15	9
8	8	8	8	2	2023-08-20	5
9	9	9	9	3	2023-09-25	8
10	10	1	1	1	2023-10-30	7
11	11	1	1	2	2023-11-05	6
12	12	1	2	3	2023-12-10	9
13	13	3	3	1	2024-01-15	8
14	14	4	4	2	2024-02-20	7
15	15	5	5	3	2024-03-25	9
16	16	6	6	1	2024-04-30	6
17	17	7	7	2	2024-05-05	8
18	18	8	8	3	2024-06-10	7
19	19	1	9	1	2024-07-15	9
20	20	2	2	2	2024-08-20	5
21	21	2	2	3	2024-09-25	8
22	22	2	2	1	2024-10-30	7

23	23	2	3	2	2024-11-05	6
24	24	4	4	3	2024-12-10	9
25	25	5	12	1	2023-01-15	8
26	26	6	12	2	2023-02-20	7
27	27	7	17	3	2023-03-25	9
28	28	8	18	1	2023-04-30	6
29	29	9	19	2	2023-05-05	8
30	30	3	13	3	2023-06-10	7

Рисунок 3.6 – Таблица gradebook

	id [PK] integer	name character varying (100)	rate real	salary money
1	1	Директор	1	100 000,00 ?
2	2	Заместитель директора	0.9	90 000,00 ?
3	3	Учитель математики	0.8	80 000,00 ?
4	4	Учитель русского языка	0.8	80 000,00 ?
5	5	Учитель физики	0.8	80 000,00 ?
6	6	Учитель химии	0.8	80 000,00 ?
7	7	Учитель биологии	0.8	80 000,00 ?
8	8	Учитель истории	0.8	80 000,00 ?
9	9	Учитель географии	0.8	80 000,00 ?
10	10	Учитель английского языка	0.8	80 000,00 ?
11	11	Учитель физической культуры	0.8	80 000,00 ?
12	12	Учитель искусства	0.8	80 000,00 ?
13	13	Учитель музыки	0.8	80 000,00 ?
14	14	Учитель информатики	0.8	80 000,00 ?
15	15	Учитель технологии	0.8	80 000,00 ?
16	16	Учитель начальных классов	0.8	80 000,00 ?
17	17	Педагог-психолог	0.7	70 000,00 ?
18	18	Социальный педагог	0.7	70 000,00 ?
19	19	Логопед	0.7	70 000,00 ?
20	20	Библиотекарь	0.7	70 000,00 ?
21	21	Администратор	0.6	60 000,00 ?
22	22	Секретарь	0.6	60 000,00 ?
23	23	Бухгалтер	0.6	60 000,00 ?
24	24	Сторож	0.5	50 000,00 ?
25	25	Уборщик	0.5	50 000,00 ?
26	26	Воспитатель	0.7	70 000,00 ?
27	27	Медсестра	0.6	60 000,00 ?
28	28	Дежурный по физической культуре	0.7	70 000,00 ?
29	29	Дежурный по охране	0.6	60 000,00 ?
30	30	Дежурный по уборке	0.5	50 000,00 ?

Рисунок 3.7 – Таблица job\_position

	employee_id [PK] bigint	subject_id bigint
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23
24	24	24
25	25	25
26	26	26
27	27	27
28	28	28
29	29	29
30	30	30

Рисунок 3.8 – Таблица knowledge\_of\_subject

	id [PK] integer	city character varying (100)	street character varying (100)	house_number integer
1	1	Нью-Йорк	Бродвей	123
2	2	Лос-Анджелес	Сансет-бульвар	456
3	3	Чикаго	Мичиган-авеню	789
4	4	Хьюстон	Мэйн-стрит	321
5	5	Финикс	Центральная авеню	654
6	6	Филадельфия	Маркет-стрит	987
7	7	Сан-Антонио	Риверволк	147
8	8	Сан-Диего	Оушен-бульвар	258
9	9	Даллас	Элм-стрит	369
10	10	Сан-Хосе	Первая улица	753
11	11	Остин	Конгресс-авеню	852
12	12	Джэксонвилл	Бей-стрит	963
13	13	Индианаполис	Монумент-серкуль	159
14	14	Сан-Франциско	Ломбард-стрит	357
15	15	Колумбус	Хай-стрит	468
16	16	Форт-Уэрт	Санданс-сквер	579
17	17	Шарлотт	Трейд-стрит	684
18	18	Сиэтл	Пайк-стрит	795
19	19	Денвер	Колфакс-авеню	876
20	20	Вашингтон	Пенсильвания-авеню	987
21	21	Бостон	Ньюбери-стрит	369
22	22	Нэшвилл	Бродвей	753
23	23	Балтимор	Иннер-Харбор	852
24	24	Оклахома-Сити	Бриктаун	963
25	25	Портленд	Пёрл-дистрикт	159
26	26	Лас-Вегас	Стрип	357
27	27	Милуоки	Уотер-стрит	468
28	28	Альбукерке	Шоссе 66	579
29	29	Тусон	Четвертая авеню	684
30	30	Майами	Оушен Драйв	795
31	31	Минск	Независимости	123
32	32	Логойск	Интернациональная	456
33	33	Витебск	Космонавтов	789
34	34	Гродно	Малинина	321
35	35	Гомель	Могилевская	654
36	36	Гомель	Луговая	987
37	37	Несвиж	Первомайская	147
38	38	Мир	Пушкинская	258
39	39	Брест	Деревянная	369
40	40	Кобрин	Цветочная	753
41	41	Мозырь	Победителей	852
42	42	Орша	Центральная	963
43	43	Орша	Центральная	159
44	44	Кобрин	Лесная	357
45	45	Клецк	Домашняя	468
46	46	Несвиж	Московская	579
47	47	Солигорск	Солевая	684
48	48	Держинск	Горная	795
49	49	Фаниполь	Клубничная	876
50	50	Лунинец	Солнечная	987
51	51	Калинковичи	Карла Маркса	369
52	52	Могилев	Центральная	753
53	53	Климовичи	Лесная	852
54	54	Минск	Каменная	963
55	55	Минск	Космонавтов	159
56	56	Минск	Железная	357
57	57	Марьина Горка	Интернациональная	468
58	58	Волковыск	Советская	579
59	59	Гродно	Полевая	684
60	60	Минск	Мирная	795

Рисунок 3.9 – Таблица residential\_address

	id [PK] integer	class_id bigint	subject_id bigint
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10
11	11	11	11
12	12	12	12
13	13	13	13
14	14	14	14
15	15	15	15
16	16	16	16
17	17	17	17
18	18	18	18
19	19	19	19
20	20	20	20
21	21	21	21
22	22	22	22
23	23	23	23
24	24	24	24
25	25	25	25
26	26	26	26
27	27	27	27
28	28	28	28
29	29	29	29
30	30	30	30

Рисунок 3.10 – Таблица schedule

	id [PK] integer	first_name character varying (100)	last_name character varying (100)	patronymic character varying (100)	passport_id character	date_of_birth date	class_id bigint	residential_address_id bigint
1	1	Иван	Иванов	Иванович	AB1234567CD89	2007-05-10	1	1
2	2	Петр	Петров	Петрович	EF9876543GH21	2008-02-15	2	2
3	3	Анна	Сидорова	Александровна	KL4567890MN32	2009-08-20	3	3
4	4	Мария	Козлова	Игоревна	OP6543210QR54	2010-07-25	4	4
5	5	Сергей	Васильев	Дмитриевич	ST0987654UV76	2011-03-30	5	5
6	6	Елена	Никитина	Павловна	WX2345678YZ98	2012-11-05	6	6
7	7	Алексей	Григорьев	Андреевич	CD8765432AB09	2013-09-12	7	7
8	8	Ольга	Павлова	Сергеевна	EF2109876GH43	2014-04-18	8	8
9	9	Дмитрий	Федоров	Владимирович	IJ0987654KL76	2015-12-23	9	9
10	10	Наталья	Иванова	Михайловна	MN5432109OP65	2016-10-28	10	10
11	11	Ирина	Петрова	Алексеевна	QR7654321ST87	2017-06-02	11	11
12	12	Андрей	Сидоров	Петрович	UV1098765WX32	2018-01-07	12	12
13	13	Виктория	Смирнова	Анатольевна	YZ8765432WX10	2007-09-12	13	13
14	14	Артем	Кузнецов	Иванович	BC5432109DE87	2008-07-17	14	14
15	15	Татьяна	Морозова	Алексеевна	FG3210987HI54	2009-04-22	15	15
16	16	Григорий	Новиков	Петрович	JK1098765LM32	2010-12-27	16	16
17	17	Василиса	Зайцева	Дмитриевна	NO9876543PQ21	2011-10-02	17	17
18	18	Егор	Павлов	Сергеевич	RS7654321TU98	2012-05-08	18	18
19	19	Людмила	Соколова	Игоревна	VW5432109XY76	2013-01-13	19	19
20	20	Александра	Иванова	Александровна	AB2109876CD43	2014-08-18	20	20
21	21	Денис	Петров	Владимирович	EF8765432GH65	2015-06-23	21	21
22	22	Екатерина	Сидорова	Михайловна	IJ6543210KL87	2016-02-28	22	22

23	23	Антон	Козлов	Павлович	MN43210980P54	2017-11-03	23	23
24	24	Олег	Васильев	Андреевич	QR0987654ST32	2018-07-08	24	24
25	25	Марина	Григорьева	Петровна	UV7654321WX09	2007-03-13	25	25
26	26	Николай	Никитин	Сергеевич	CD3210987AB76	2008-12-18	26	26
27	27	Елена	Федорова	Ивановна	GH9876543IJ43	2009-10-23	27	27
28	28	Игорь	Иванов	Александрович	KL6543210MN65	2010-06-28	28	28
29	29	Лариса	Петрова	Владимировна	OP2109876QR87	2011-02-03	29	29
30	30	Владимир	Сидоров	Михайлович	ST4321098UV54	2012-09-08	30	30
31	31	Анастасия	Козлова	Павловна	WX6543210YZ32	2013-07-13	21	30

Рисунок 3.11 – Таблица student

	id [PK] integer	name character varying (100)	hours integer	classroom integer
1	1	Математика	60	101
2	2	Физика	45	102
3	3	История	30	103
4	4	Литература	45	104
5	5	Биология	60	105
6	6	Химия	45	106
7	7	География	30	107
8	8	Английский язык	60	108
9	9	Французский язык	45	109
10	10	Немецкий язык	30	110
11	11	Испанский язык	45	111
12	12	Информатика	60	112
13	13	Физкультура	30	113
14	14	Музыка	45	114
15	15	ИЗО	60	115
16	16	Трудовое обучение	30	116
17	17	Обществознание	45	117
18	18	Право	60	118
19	19	Экономика	30	119
20	20	Психология	45	120
21	21	Философия	60	121
22	22	Религиоведение	30	122
23	23	Хореография	45	123
24	24	Театральное искусство	60	124
25	25	Основы медицины	30	125
26	26	Дизайн	45	126
27	27	Астрономия	60	127
28	28	Геология	30	128
29	29	Архитектура	45	129
30	30	Лингвистика	60	130

Рисунок 3.12 – Таблица subject

Для изменения структуры таблицы, например переименования столбца таблицы (см. рисунок 3.3), можно воспользоваться оператором ALTER TABLE:

```
ALTER TABLE main_scheme.student RENAME COLUMN middle_name TO patronymic;
```

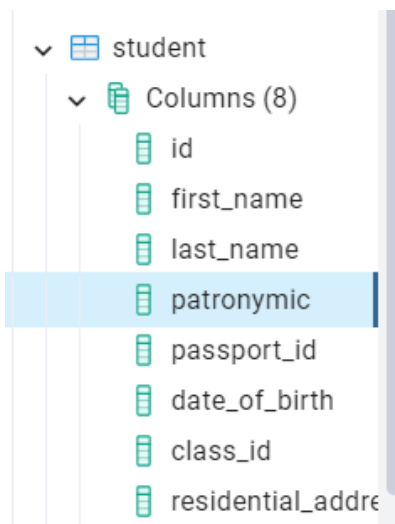


Рисунок 3.13 – Переименование названия столбца в таблице «student»

Далее создадим (см. рисунок 3.4) временную таблицу с помощью оператора CREATE TABLE и удалим ее с помощью оператора DROP TABLE:

```
CREATE TABLE main_scheme.temp (
    field1 VARCHAR(255),
    field2 VARCHAR(255),
    field3 VARCHAR(255)
);
```

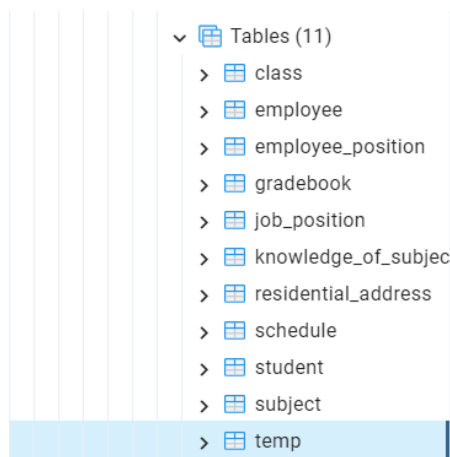


Рисунок 3.14 – Создание временной таблицы «temp»

Для удаления данной таблицы воспользуемся следующей конструкцией:

```
DROP TABLE main_scheme.temp;
```

Для экспорта данных нужно кликнуть по выбранной таблице ПКМ, после чего нажать «Import/Export data». Откроется окно для экспорта данных (см. рисунок 3.4).



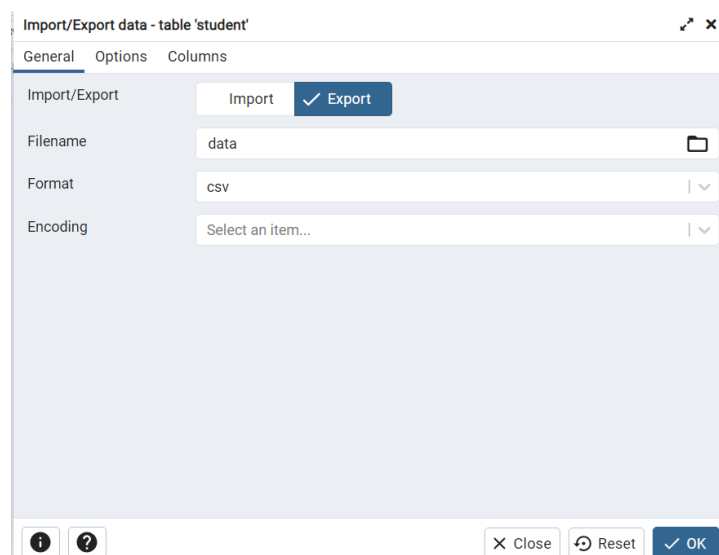


Рисунок 3.15 – Окно для экспорта данных из таблицы «student»

Выберем вкладку «Export» и дадим имя экспортируемому файл, например «data», нажимаем «ОК».

Для остальных таблиц экспорт выполняется аналогично.

Для экспорта результатов работы можно сделать следующее: нажать ПКМ по созданной базе данных «school» и выбрать «Backup». Откроется окно, изображенное на рисунке 3.5.

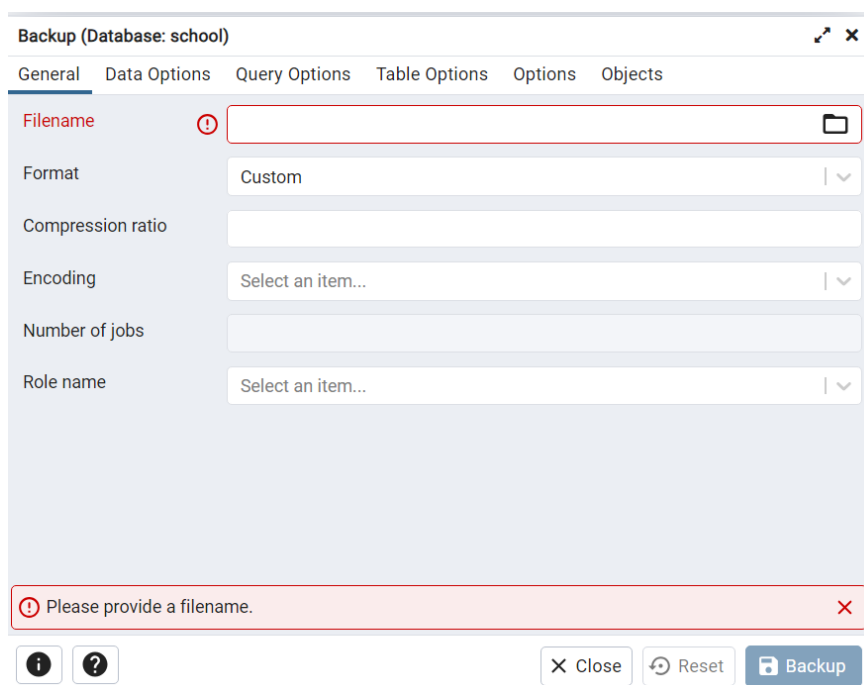


Рисунок 3.16 – Окно Backup

#### **4 ВЫВОД**

В ходе лабораторной работы была реализована реляционная схема данных с помощью SQL-операторов для создания схемы, таблиц и индексов. Также были сформированы SQL-операторы для добавления, изменения и удаления данных, реализованы транзакции, а также импорт и экспорт данных.