МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

УНИВЕРСИТЕТ САТПАЕВ

Институт автоматики и информационных технологий

Кафедра Кибербезопасность обработка и хранение информации



Лабораторная работа 3

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Качество выполнения работ** | **Диапазон оценки** | **Получено %/балл** |
| 1 | Не выполнено | 0% |  |
| 2 | Выполнено | 0-50% |  |
| 3 | Самостоятельная систематизация материала | 0-10% |  |
| 4 | Выполнение требуемого объема и в указанный срок | 0-5% |  |
| 5 | Использование дополнительной научной литературы | 0-5% |  |
| 6 | Уникальность выполненного задания | 0-10% |  |
| 7 | Защита работы | 0-20% |  |
|  | Итого | 0-100% |  |

Преподаватель: Кайрбеков Абылай Муратович

Студент: Ораков Абылхаир

Специальность: Информационные системы

### **Лабораторная работа 3: Создание и нормализация таблиц базы данных**

**Цель:** Научиться создавать таблицы и применять нормализацию данных.

https://github.com/DoberAbo/Lab-3-BD.git

1. На основе реляционной схемы из предыдущей лабораторной работы создайте таблицы в вашей базе данных.

Чтобы устранить избыточность, делаем 5 нормализованных таблиц

Сначала подключимся к бд

CREATE DATABASE lab3;

\c lab3;

-- Подключение к БД lab3 CREATE DATABASE lab3; \c lab3;

-Студенты

CREATE TABLE Students (

student\_id SERIAL PRIMARY KEY,

full\_name VARCHAR(100) NOT NULL,

group\_name VARCHAR(20) NOT NULL

);

-Курсы

CREATE TABLE Courses (

course\_id SERIAL PRIMARY KEY,

course\_name VARCHAR(100) NOT NULL

);

- Преподаватели

CREATE TABLE Teachers (

teacher\_id SERIAL PRIMARY KEY,

full\_name VARCHAR(100) NOT NULL

);

- Аудитория

CREATE TABLE Classrooms (

classroom\_id SERIAL PRIMARY KEY,

room\_number VARCHAR(20) NOT NULL

);

-- Таблица связей (зачисления)

CREATE TABLE Enrollments (

enrollment\_id SERIAL PRIMARY KEY,

student\_id INT NOT NULL,

course\_id INT NOT NULL,

teacher\_id INT NOT NULL,

classroom\_id INT NOT NULL,

grade INT CHECK(grade BETWEEN 1 AND 5),

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

FOREIGN KEY (course\_id) REFERENCES Courses(course\_id),

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

FOREIGN KEY (classroom\_id) REFERENCES Classrooms(classroom\_id)

);



2. Примените нормализацию до 3NF, устранив избыточность и аномалии.

## **1НФ (первая нормальная форма)**

Что сделал:

* В таблице StudentsCourses поля были неатомарные (студент, курс, преподаватель, аудитория в одной таблице).
* Вынесли студентов и курсы в отдельные таблицы. Так лудше

## **2НФ (вторая нормальная форма)**

Что сделал:

* В таблице StudentCourses преподаватель и аудитория зависели от курса, а не от студента. Теперь легче управлять бд
* Вынесли их в отдельные таблицы и связали с курсами.

## **3НФ (третья нормальная форма)**

Что сделал:

* Проверили транзитивные зависимости.
* Каждая таблица теперь хранит только свои атрибуты:
  + Students → только студенты.
  + Teachers → только преподаватели.
  + Classrooms → только аудитории.
  + Courses → только название курса + ссылки на преподавателя и аудиторию.
  + Enrollments → только зачисления.

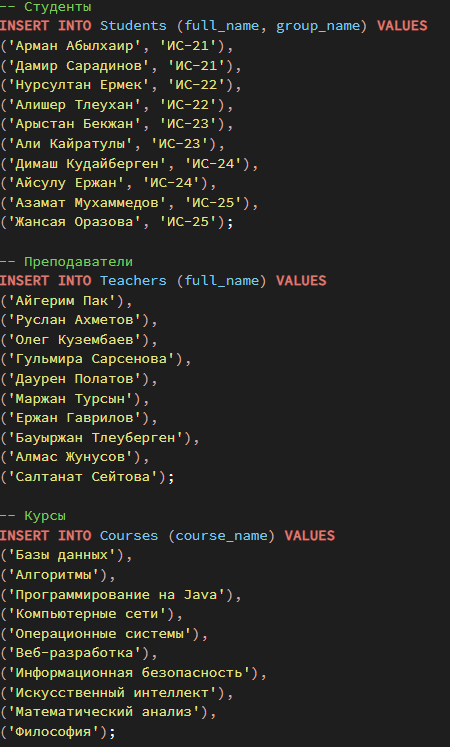
Если проще то при использований 1-2-3 НФто мы просто:

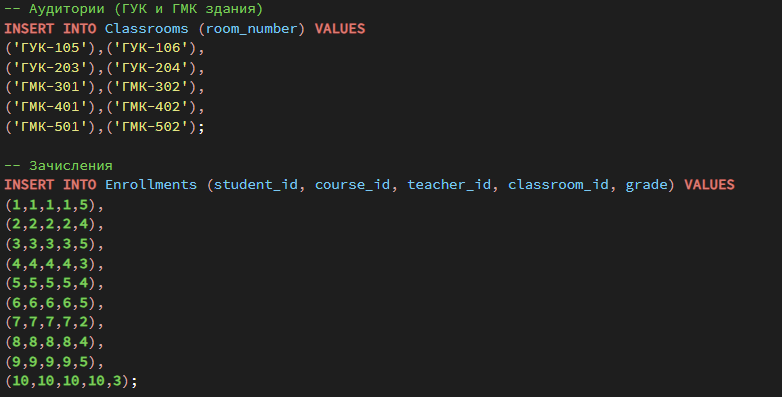
1НФ убрали повторяющиеся данные, разделили студентов и курсы.

2НФ убрали зависимости от части ключа, вынесли преподавателей и аудитории.

3НФ убрали транзитивные зависимости, каждая таблица отвечает только за свою сущность.

3. Заполните таблицы тестовыми данными (минимум 10 записей на таблицу).





Код:

-- Студенты INSERT INTO Students (full\_name, group\_name) VALUES ('Арман Абылхаир', 'ИС-21'), ('Дамир Сарадинов', 'ИС-21'), ('Нурсултан Ермек', 'ИС-22'), ('Алишер Тлеухан', 'ИС-22'), ('Арыстан Бекжан', 'ИС-23'), ('Али Кайратулы', 'ИС-23'), ('Димаш Кудайберген', 'ИС-24'), ('Айсулу Ержан', 'ИС-24'), ('Азамат Мухаммедов', 'ИС-25'), ('Жансая Оразова', 'ИС-25');

-- Курсы INSERT INTO Courses (course\_name) VALUES ('Базы данных'), ('Алгоритмы'), ('Программирование на Java'), ('Компьютерные сети'), ('Операционные системы'), ('Веб-разработка'), ('Информационная безопасность'), ('Искусственный интеллект'), ('Математический анализ'), ('Философия');

-- Преподаватели INSERT INTO Teachers (full\_name) VALUES ('Айгерим Пак'), ('Руслан Ахметов'), ('Олег Кузембаев'), ('Гульмира Сарсенова'), ('Даурен Полатов'), ('Маржан Турсын'), ('Ержан Гаврилов'), ('Бауыржан Тлеуберген'), ('Алмас Жунусов'), ('Салтанат Сейтова');

-- Аудитории (ГУК и ГМК здания) INSERT INTO Classrooms (room\_number) VALUES ('ГУК-105'),('ГУК-106'), ('ГУК-203'),('ГУК-204'), ('ГМК-301'),('ГМК-302'), ('ГМК-401'),('ГМК-402'), ('ГМК-501'),('ГМК-502');

-- Зачисления INSERT INTO Enrollments (student\_id, course\_id, teacher\_id, classroom\_id, grade) VALUES (1,1,1,1,5), (2,2,2,2,4), (3,3,3,3,5), (4,4,4,4,3), (5,5,5,5,4), (6,6,6,6,5), (7,7,7,7,2), (8,8,8,8,4), (9,9,9,9,5), (10,10,10,10,3);

4. Проверьте целостность данных, убедившись, что все ключи и связи работают корректно.

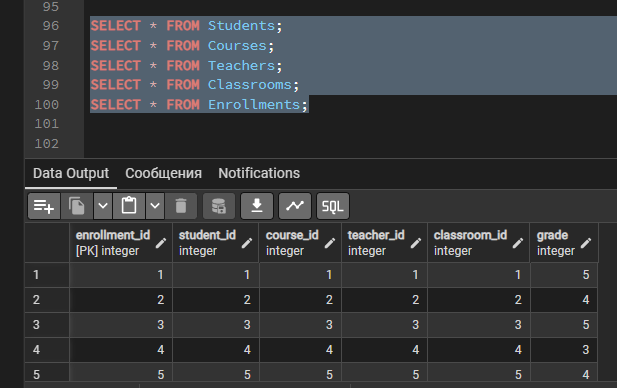
Для проверки использовать можно выводы таблицы

SELECT \* FROM Students;

SELECT \* FROM Courses;

SELECT \* FROM Teachers;

SELECT \* FROM Classrooms;



SELECT \* FROM Enrollments;

Или

INSERT INTO Enrollments (student\_id, course\_id, teacher\_id, classroom\_id, grade) пробуем вставить неправильный id

VALUES (999, 1, 1, 1, 5);