**Db\_normalization\_2  
Hometask**

**Part 1: Выбор сценария**

Для данной работы был выбран сценарий: **Система высшего образования.** Эта система будет наглядно представлять список учащихся, преподавателей, предметов, а также позволит составлять расписание вместе с отслеживанием успеваемости по результатам экзаменов/зачётов.

**Part 2: Проектирование Базы Данных и Документация**

**1. Студенты (students)  
 2. Преподаватели (teachers)  
 3. Предметы (subjects)  
 4. Расписание (schedule)  
 5. Успеваемость (academical\_records)  
 6.Факультеты (faculties)  
 7. Кафедры (departments)  
 8. Учёные степени (academic\_degrees)  
 9. Группы студентов (students\_groups)**

**Проектирование Таблиц:**

1. **Table Name: students**• **Description:** **Cодержит информацию о студентах ВУЗа**• **Attributes:**  
    ■ student\_id: SERIAL, PK, NOT NULL, UNIQUE

■ first\_name: VARCHAR(20), NOT NULL

■ last\_name: VARCHAR(20), NOT NULL

■ birth\_date: DATE, NOT NULL

■ email: VARCHAR(100), NOT NULL, UNIQUE

■ admission\_year: INTEGER, NOT NULL

■ group\_id: INTEGER, FK (REFERENCES students\_groups), NOT NULL

■ faculty\_id: INTEGER, FK (REFERENCES faculties), NOT NULL

• **Constraints:**

■ PK\_students: PRIMARY KEY (student\_id)

■ FK\_students\_faculties: FOREIGN KEY (faculty\_id) REFERENCES faculties(faculty\_id)

■ FK\_students\_students\_group: FOREIGN KEY (group\_id) REFERENCES students\_groups(group\_id)

■ CHK\_admission\_year: CHECK (admission\_year BETWEEN 1920 AND EXTRACT(YEAR FROM CEURRENT\_YEAR)))

2. **Table Name: teachers**

• **Description:** содержит информацию о преподавательском составе.

• **Attributes:**

**■** teacher\_id: SERIAL, PK, NOT NULL, UNIQUE

■ first\_name: VARCHAR(50), NOT NULL

■ last\_name: VARCHAR(50), NOT NULL

■ email: VARCHAR(100), NOT NULL, UNIQUE

■ phone: VARCHAR(20), NOT NULL, UNIQUE

■ degree\_id: INTEGER, FK (REFERENCES academic\_degree), NOT NULL

■ department\_id: INTEGER, FK (REFERENCES departments), NOT NULL

■ hire\_date: DATE, NOT NULL

• **Constraints:**

■ PK\_teachers: PRIMARY KEY (teacher\_id)

■ FK\_teachers\_academic\_degrees: FOREIGN KEY (degree\_id) REFERENCES academic\_degrees(degree\_id)

■ FK\_teachers\_departments: FOREIGN KEY (department\_id) REFERENCES departments(department\_id)

■ CHK\_hire\_date: CHECK (hire\_date <= CURRENT\_DATE))

3. **Table Name: subjects**

• **Description: C**одержит информацию об учебных предметах.

• **Attributes:**

**■** subject\_id: INTEGER, PK, NOT NULL, UNIQUE

■ subject\_name: VARCHAR(100), NOT NULL

■ description: TEXT

■ hours: INTEGER, NOT NULL

• **Constraints:**

**■** PK\_subjects: PRIMARY KEY (subject\_id)

■ CHK\_hours: CHECK (hours > 0)

4. **Table Name: schedule**

• **Description:** содержит информацию о расписании занятий.

• **Attributes:**

■ schedule\_id: SERIAL, PK, NOT NULL, UNIQUE

■ subject\_id: INTEGER, FK (REFERENCES subjects), NOT NULL

■ group\_id: INTEGER, FK (REFERENCES students\_groups), NOT NULL

■ teacher\_id: INTEGER, FK (REFERENCES teachers) NOT NULL

■ audience\_number: VARCHAR(20), NOT NULL

■ day\_of\_week: INTEGER, NOT NULL

■ start\_time: TIME, NOT NULL

■ end\_time: TIME, NOT NULL

■ semester: INTEGER, NOT NULL

• **Constraints:**

**■** PK\_schedule: PRIMARY KEY (schedule\_id)

■ FK\_schedule\_subjects: FOREIGN KEY (subject\_id) REFERENCES subjects(subject\_id)

■ FK\_schedule\_students\_groups: FOREIGN KEY (group\_id) REFERENCES students\_groups(group\_id)

■ FK\_schedule\_teachers: FOREIGN KEY (teacher\_id) REFERENCES teachers(teacher\_id)

■ CHK\_day\_of\_week: CHECK (day\_of\_week BETWEEN 1 AND 7)

■ CHK\_semester: CHECK (semester IN (1, 2))

5. **Table Name: academical\_records**

• **Description:** **Таблица для реализации связи многие-ко-многим.** **Связывает** **студентов с предметами, хранит информацию о семестре, оценке и дате получения оценки.**

• **Attributes:**

**■** record\_id: SERIAL, PK, NOT NULL, UNIQUE

■ student\_id: INTEGER, FK, NOT NULL

■subject\_id: INTEGER, FK, NOT NULL

■ semester: INTEGER, NOT NULL

■ grade: INTEGER, NOT NULL

■ exam\_date: DATE

■ teacher\_id: INTEGER, FK

• **Constraints:**

■ PK\_academical\_records: PRIMARY KEY (record\_id)

■ FK\_academical\_records\_students: FOREIGN KEY (student\_id) REFERENCES students(student\_id)

■ FK\_academical\_records\_subjects: FOREIGN KEY (subject\_id) REFERENCES subjects(subject\_id)

■ FK-acadenical\_record\_teachers: FOREIGN KEY (teacher\_id) REFERENCES teachers(teacher\_id)

■ CHK\_semester: CHECK (semester IN (1, 2))

■ CHK\_grade: CHECK (grade BETWEEN 1 AND 10 OR grade IS NULL)

■ CHK\_exan\_date: CHECK (exam\_date CURRENT\_DATE OR exam\_date IS NULL)

6. **Table Name: faculties**

• **Description: содержит названия факультетов ВУЗа.**

• **Attributes:**

■ faculty\_id: SERIAL, PK, NOT NULL, UNIQUE

■ faculty\_name: VARCHAR(200), NOT NULL

• **Constraints:**

■ PK\_faculties: PRIMARY KEY (faculty\_id)

7. **Table Name: departments**

• **Description: содержит информацию о кафедрах ВУЗа.**

• **Attributes:**

■ department\_id: SERIAL, PK, NOT NULL, UNIQUE

■ department\_name: VARCHAR(200), NOT NULL

• **Constraints:**

■ PK\_departments: PRIMARY KEY (department\_id)

8. **Table Name: academic\_degrees**

• **Description: содрежит информацию о степенях образования.**

• **Attributes:**

■ degree\_id: SERIAL, NOT NULL, UNIQUE

■ degree\_name: VARCHAR(200), NOT NULL

• **Constraints:**

■ PK\_academic\_degrees: PRIMARY KEY (degree\_id)

9. **Table Name: students\_groups**

• **Description: содержит информацию о группах студентов ВУЗа.**

• **Attributes:**

■ group\_id: SERIAL, PK, NOT NULL, UNIQUE

■ group\_name: VARCHAR(100), NOT NULL

■ faculty\_id: INTEGER, FK (REFERENCES faculties), NOT NULL

■ enrollment\_date: DATE, NOT NULL

■ teacher\_id: INTEGER, FK (REFERENCES teachers), NOT NULL

• **Constraints:**

■ PK\_studenst\_groups: PRIMARY KEY (group\_id)

■ FK\_students\_groups\_faculties: FOREIGN KEY (faculty\_id) REFERENCES faculties(faculty\_id)

■ FK\_students\_groups\_teachers: FOREIGN KEY (teacher\_id) REFERENCES teachers(teacher\_id)

■ CHK\_enrollment\_date: CHECK (enrollment\_date <= CURRENT\_DATE)

**Взаимосвязи:**

**1. teachers и schedule(Один-ко-многим)**: Один преподаватель может вести несколько занятий.

* schedule.teacher\_id является внешним ключом, ссылающимся на teachers.teacher\_id

**2. subjects и schedule(Один-ко-многим)**: Один предмет может быть в расписании много раз.

* schedule.subject\_id является внешним ключом, ссылающимся на subjects.subject\_id

**3. students и students\_groups (Многие-к-одному):** Один студент принадлежит только одной группе.

* students.group\_id является внешним ключом, ссылающимся на students\_groups.group\_id.

**4. students\_groups и faculties (Многие-к-одному):** Одна группа принадлежит одному факультету.

* students\_groups.faculty\_id является внешним ключом, ссылающимся на faculties.faculty\_id.

**5. teachers и departments (Многие-к-одному):** Один преподаватель привязан к одной кафедре.

* teachers.department\_id является внешним ключом, ссылающимся на departments.department\_id.

**6. teachers и academic\_degrees (Многие-к-одному):** Один преподаватель имеет одну ученую степень.

* teachers.degree\_id является внешним ключом, ссылающимся на academic\_degrees.degree\_id.

**7. students\_groups и teachers (Многие-к-одному):** У одной группы есть один куратор-преподаватель.

* students\_groups.teacher\_id является внешним ключом, ссылающимся на teachers.teacher\_id.

**8. students, subjects и teachers в academical\_records(Многие-ко-многим)**: Один студент может изучать несколько предметов так же,как и один предмет может изучаться многими студентами.

* academical\_records.student\_id является внешним ключом, ссылающимся на students.student\_id
* academical\_records.subject\_id является внешним ключом, ссылающимся на subjects.subject\_id
* academical\_records.teacher\_id является внешним ключом, ссылающимся на teachers.teacher\_id

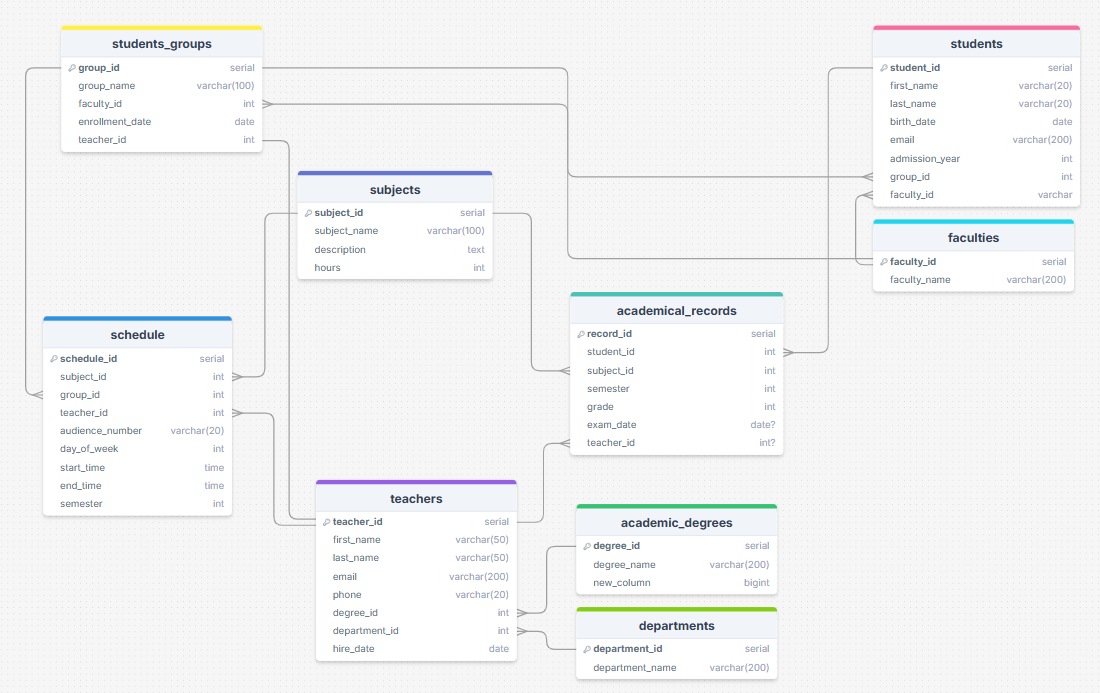
**9. students и faculties (Многие-к-одному):** Один студент принадлежит одному факультету.

* students.faculty\_id является внешним ключом, ссылающимся на faculties.faculty\_id.

**10. schedule и students\_groups (Многие-к-одному):** Одно занятие в расписании предназначено для одной группы.

* schedule.group\_id является внешним ключом, ссылающимся на students\_groups.group\_id.

**Part 3: ER-Диаграмма**

****