

# Решение задания по теме "Система высшего образования"

## Часть 1: Выбор сценария

Для данной работы выбран сценарий: **Система высшего образования**. Система будет отслеживать кафедры, студентов, преподавателей, предметы, зачисления на курсы.

## Часть 2: Проектирование базы данных

### Идентификация Сущностей и Атрибутов:

1. Кафедра (Department);
2. Преподаватель (Teacher);
3. Предмет (Subject);
4. Студент (Student)
5. Зачисление на курс (Enrollment)

### Проектирование таблиц:

#### 1. Table Name: Department

- **Description:** Информация о кафедрах университета
- **Attributes:**
  - *id*: INTEGER, PK, NOT NULL, UNIQUE
  - *department\_name*: VARCHAR(100), NOT NULL
  - *description*: TEXT
- **Constraints:**
  - *PK\_Department*: PRIMARY KEY (id)
  - *UQ\_DepartmentName*: UNIQUE (department\_name)

#### 2. Table Name: Teacher

- **Description:** Данные преподавателей
- **Attributes:**
  - *id*: INTEGER, PK, NOT NULL, UNIQUE
  - *department\_id*: INTEGER, FK (REFERENCES Department), NOT NULL
  - *first\_name*: VARCHAR(100), NOT NULL
  - *last\_name*: VARCHAR(100), NOT NULL
  - *birthday\_date*: DATE
  - *email*: VARCHAR(255), UNIQUE
  - *phone*: VARCHAR(20), UNIQUE
- **Constraints:**
  - *PK\_Teacher*: PRIMARY KEY (id)
  - *FK\_Teacher\_Department*: FOREIGN KEY (department\_id) REFERENCES Department(id)
  - *CHK\_Birthday*: CHECK (birthday\_date <= CURRENT\_DATE - INTERVAL '22 YEARS')

#### 3. Table Name: Subject

- **Description:** Учебные предметы
- **Attributes:**
  - *id*: INTEGER, PK, NOT NULL, UNIQUE
  - *teacher\_id*: INTEGER, FK (REFERENCES Teacher), NOT NULL
  - *subject\_name*: VARCHAR(100), NOT NULL
  - *number\_hours*: INTEGER

- **Constraints:**
  - *PK\_Subject*: PRIMARY KEY (id)
  - *FK\_Subject\_Teacher*: FOREIGN KEY (teacher\_id) REFERENCES Teacher(id)
  - *CHK\_Hours*: CHECK (number\_hours > 0)

#### 4. Table Name: Student

- **Description:** Данные студентов
- **Attributes:**
  - *id*: INTEGER, PK, NOT NULL, UNIQUE
  - *first\_name*: VARCHAR(100), NOT NULL
  - *last\_name*: VARCHAR(100), NOT NULL
  - *birthday\_date*: DATE
  - *email*: VARCHAR(255), UNIQUE
  - *phone*: VARCHAR(20), UNIQUE
- **Constraints:**
  - *PK\_Student*: PRIMARY KEY (id)
  - *CHK\_StudentAge*: CHECK (birthday\_date <= CURRENT\_DATE - INTERVAL '16 YEARS')

#### 5. Table Name: Enrollment

- **Description:** Записи о зачислении студентов на предметы
- **Attributes:**
  - *student\_id*: INTEGER, PK/FK (REFERENCES Student), NOT NULL
  - *subject\_id*: INTEGER, PK/FK (REFERENCES Subject), NOT NULL
  - *enrollment\_date*: DATE, DEFAULT CURRENT\_DATE
- **Constraints:**
  - *PK\_Enrollment*: PRIMARY KEY (student\_id, subject\_id)
  - *FK\_Enrollment\_Student*: FOREIGN KEY (student\_id) REFERENCES Student(id)
  - *FK\_Enrollment\_Subject*: FOREIGN KEY (subject\_id) REFERENCES Subject(id)

#### Взаимосвязи:

1. **Department и Teacher (Один-ко-многим):** Одна кафедра (Department) может иметь много преподавателей (Teacher). Каждый преподаватель принадлежит одной кафедре.
  - *Department.department\_id* является внешним ключом, который ссылается на *Teacher.department\_id*
2. **Teacher и Subject (Один-ко-Многим):** Один преподаватель может вести несколько предметов (Subject), но каждый предмет преподается только одним преподавателем (в данной модели).
  - *Teacher.teacher\_id* является внешним ключом, который ссылается на *Subject.teacher\_id*
3. **Student N и M Subject (Многие-ко-многим через Enrollment):** Один студент может быть записан на много предметов, но один предмет может изучаться многими студентами. Поэтому появляется проблема: прямая связь невозможна в реляционных БД – требуется промежуточная таблица Enrollment.
  - Таблица *Enrollment* содержит:
    - student\_id* (FK -> Student.id).
    - subject\_id* (FK -> Subject.id).
 Эти поля вместе образуют составной первичный ключ (PK).

#### Part 3: ER-диаграмма

