Для данной работы выбран сценарий: Система высшего образования. Эта система будет управлять студентами, преподавателями, курсами, записями на курсы

Идентификация сущностей и атрибутов:

1. Студенты (Students)
2. Курсы (Courses)
3. Запись на курсы (Enrollments)
4. Преподаватели (Professors)
5. **Table Name: Students**

**Description:** Данные о студентах.

**Attributes:**

StudentID – INTEGER, PK, NOT NULL, UNIQUE

IDCardNumber – INTEGER, NOT NULL, UNIQUE

StudentName – VARCHAR(100), NOT NULL

GroupName – INTEGER, NOT NULL

**Constraints:**

PK\_Students – PRIMARY KEY (StudentID)

UQ\_ IDCardNumber – UNIQUE (IDCardNumber)

1. **Table Name: Courses**

**Description:** Информация о студентах, записавшихся на курсы.

**Attributes:**

CourseID – INTEGER, PK, NOT NULL, UNIQUE

CourseName – VARCHAR(50), NOT NULL

CourseHours – INTEGER, NOT NULL, CHECK >0

**Constraints:**

PK\_Courses – PRIMARY KEY (CourseID)

CHK\_CourseHours – CHECK (CourseHours > 0 AND CourseHours < 300)

1. **Table Name: Enrollments**

**Description:** Связующая таблица для записи студентов на курсы (многие-ко-многим).

**Attributes:**

EnrollmentID – INTEGER, PK, NOT NULL, UNIQUE

StudentID – INTEGER, FK (Students table), NOT NULL

CourseID – INTEGER, FK (Courses table), NOT NULL

* **Constraints:**

PK\_Enrollments – PRIMARY KEY (EnrollmentID)

FK\_Enrollments\_Students – FOREIGN KEY (StudentID)

FK\_Enrollments\_Courses – FOREIGN KEY (CourseID)

UQ\_StudentCourse – UNIQUE (StudentID, CourseID)

1. **Table Name: pROFESSORS**

**Description:** Преподаватели курсов

**Attributes:**

ProfessorID – INTEGER, NOT NULL

ProfessorName – VARCHAR (100), NOT NULL

CourseID – INTEGER, NOT NULL, FK, UNIQUE

Professors.CourseID → Courses.CourseID (FK, UNIQUE)

**Constraints:**

ProfessorID – PRIMARY KEY

UQ\_Professor\_Course – UNIQUE (CourseID)

**3. Взаимосвязи**

**- Students и Courses (многие-ко-многим):**

Несколько студентов могут записаться на несколько курсов. Связь реализована через промежуточную таблицу Enrollment. StudentID ссылается на Students.StudentID и CourseID ссылается на Corses.CourseID как внешние ключи

**- Courses и Professors (один-к-одному):**

На один курс один преподаватель. Один преподаватель на один курс. CourseID ссылается на Courses.CourseID как FK. Связь один-к-одному реализована через свойство UNIQUE на CourseID

