**Part 1: Выбор Сценария**

Для данной работы выбран сценарий: Система высшего образования. Эта система будет управлять студентами, предметами, преподавателями и расписанием занятий.

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

**Идентификация Сущностей и Атрибутов:**

1. Студенты (Students): ID, имя, фамилия, группа.
2. Предметы (Courses): ID, название, кафедра.
3. Преподаватели (Teachers): ID, имя, фамилия, кафедра, email.
4. Расписание (Schedule): ID, группа, день, время, предмет, преподаватель.

**Проектирование Таблиц:**

1. Table Name: Students

Description: Хранит информацию о студентах.

Attributes:

* StudentID: INTEGER, PK, NOT NULL, UNIQUE
* FirstName: VARCHAR(100), NOT NULL
* LastName: VARCHAR(100), NOT NULL
* GroupID: INTEGER, FK (REFERENCES Groups), NOT NULL

Constraints:

* PK\_Students: PRIMARY KEY (StudentID)
* FK\_Students\_Groups: FOREIGN KEY (GroupID) REFERENCES Groups (GroupID)

2. Table Name: Courses

Description: Содержит информацию об учебных предметах.

Attributes:

* CourseID: INTEGER, PK, NOT NULL, UNIQUE
* CourseName: VARCHAR(100), NOT NULL, UNIQUE
* Department: VARCHAR(100), NOT NULL

Constraints:

* PK\_Courses: PRIMARY KEY (CourseID)

3. Table Name: Teachers

Description: Хранит данные о преподавателях.

Attributes:

* TeacherID: INTEGER, PK, NOT NULL, UNIQUE
* FirstName: VARCHAR(100), NOT NULL
* LastName: VARCHAR(100), NOT NULL
* Email: VARCHAR(255), UNIQUE

Constraints:

* PK\_Teachers: PRIMARY KEY (TeacherID)
* UQ\_Email: UNIQUE (Email)

4. Table Name: Schedule

Description: Хранит данные о расписании.

Attributes:

* ID: INTEGER, PK, NOT NULL, UNIQUE
* GroupID: INTEGER, FK (REFERENCES Groups), NOT NULL
* Date: DATE, NOT NULL
* CourseID: INTEGER, FK (REFERENCES Courses), NOT NULL
* TeacherID: INTEGER, FK (REFERENCES Teachers), NOT NULL

Constraints:

* PK\_Schedule: PRIMARY KEY (ID)
* FK\_Schedule\_Groups: FOREIGN KEY (GroupID) REFERENCES Groups (GroupID)
* FK\_Schedule\_Courses: FOREIGN KEY (CourseID) REFERENCES Courses (CourseID)
* FK\_Schedule\_Teachers: FOREIGN KEY (TeacherID) REFERENCES Teachers (TeacherID)

1. Table Name: Courses\_teachers

Attributes:

* ID: INTEGER, PK, NOT NULL, UNIQUE
* CourseID: INTEGER, FK (REFERENCES Courses), NOT NULL
* TeacherID: INTEGER, FK (REFERENCES Teachers), NOT NULL

Constraints:

* PK\_ Courses\_teachers: PRIMARY KEY (ID)
* FK\_Courses: FOREIGN KEY (CourseID) REFERENCES Courses(CourseID)
* FK\_Teachers: FOREIGN KEY (TeacherID) REFERENCES Teachers(TeacherID)

1. Table Name: Groups

Attributes:

* + - GroupID: INTEGER, PK, NOT NULL, UNIQUE
    - GroupName: VARCHAR(10), NOT NULL, UNIQUE

Constraints:

* PK\_ Groups: PRIMARY KEY (GroupID)

**Взаимосвязи**

**Schedule и Courses (один-ко-многим):** один предмет может встречаться в расписании многократно, одна строка расписания вмещает только один предмет.

* Schedule.CourseID является внешним ключом, ссылающимся на Courses.CourseID.

**Schedule и Teachers (один-ко-многим):** один преподаватель может быть записан в нескольких строках расписания, одна строка расписания вмещает только одного преподавателя.

* Schedule.TeacherID является внешним ключом, ссылающимся на Teachers.TeacherID.

**Courses и Teachers (многие-ко-многим):** один предмет может преподаваться несколькими преподавателями, один преподаватель может преподавать несколько предметов.

* Courses\_teachers.CourseID является внешним ключом, ссылающимся на Courses.CourseID.
* Courses\_teachers. TeacherID является внешним ключом, ссылающимся на Teachers.TeacherID.

**Students и Groups (один-ко-многим):** один студент входит только в одну группу, в одну группу может входить множество студентов.

* Students.GroupID является внешним ключом, ссылающимся на Groups.GroupID.

**Part 3: ER-диаграмма**

