**CREATE DATABASE student\_support\_app;**

USE student\_support\_app;  
  
  
CREATE TABLE **users** (

user\_id INT AUTO\_INCREMENT PRIMARY KEY, -- Уникальный ID пользователя

name VARCHAR(255) NOT NULL, -- Имя пользователя

email VARCHAR(255) UNIQUE NOT NULL, -- Email пользователя

password VARCHAR(255) NOT NULL, -- Хэш пароля

is\_employed BOOLEAN DEFAULT FALSE, -- Работает ли студент

is\_international BOOLEAN DEFAULT FALSE, -- Иностранный студент

is\_nonresident BOOLEAN DEFAULT FALSE, -- Иногородний студент

is\_freshman BOOLEAN DEFAULT FALSE -- Первокурсник

);

CREATE TABLE **tests** (

test\_id INT AUTO\_INCREMENT PRIMARY KEY, -- Уникальный ID теста

title VARCHAR(255) NOT NULL, -- Название теста

description TEXT -- Описание теста

);

CREATE TABLE **questions** (

question\_id INT AUTO\_INCREMENT PRIMARY KEY, -- Уникальный ID вопроса

test\_id INT NOT NULL, -- Связь с тестом

question\_text TEXT NOT NULL, -- Текст вопроса

question\_type ENUM('single\_choice', 'multi\_choice', 'open') NOT NULL, -- Тип вопроса

correct\_answer TEXT, -- Правильный ответ (если применимо)

FOREIGN KEY (test\_id) REFERENCES tests(test\_id) ON DELETE CASCADE

);

CREATE TABLE **results** (

result\_id INT AUTO\_INCREMENT PRIMARY KEY, -- Уникальный ID результата

user\_id INT NOT NULL, -- Связь с пользователем

test\_id INT NOT NULL, -- Связь с тестом

test\_date DATETIME NOT NULL, -- Дата прохождения теста

score DECIMAL(5, 2) NOT NULL, -- Результат в баллах

hours\_worked INT DEFAULT 0, -- Количество отработанных часов

work\_intensity ENUM('none', 'physical', 'moderate', 'mental') DEFAULT 'none', -- Интенсивность работы

follow\_up\_checked BOOLEAN DEFAULT FALSE, -- Проверялось ли выполнение рекомендаций

FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,

FOREIGN KEY (test\_id) REFERENCES tests(test\_id) ON DELETE CASCADE

);

CREATE TABLE **recommendations** (

recommendation\_id INT AUTO\_INCREMENT PRIMARY KEY, -- Уникальный ID рекомендации

test\_id INT NOT NULL, -- Связь с тестом

score\_range VARCHAR(50) NOT NULL, -- Диапазон баллов (например, "0-50")

text TEXT NOT NULL, -- Текст рекомендации

resources TEXT, -- Ссылки на материалы

FOREIGN KEY (test\_id) REFERENCES tests(test\_id) ON DELETE CASCADE

);

CREATE TABLE **recommendation\_tracking** (

tracking\_id INT AUTO\_INCREMENT PRIMARY KEY, -- Уникальный ID записи

user\_id INT NOT NULL, -- Связь с пользователем

result\_id INT NOT NULL, -- Связь с результатом теста

recommendation\_id INT NOT NULL, -- Связь с рекомендацией

completion\_status ENUM('completed', 'partial', 'not\_completed') NOT NULL, -- Статус выполнения

feedback TEXT, -- Комментарии пользователя (опционально)

completion\_date DATETIME, -- Дата выполнения

FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,

FOREIGN KEY (result\_id) REFERENCES results(result\_id) ON DELETE CASCADE,

FOREIGN KEY (recommendation\_id) REFERENCES recommendations(recommendation\_id) ON DELETE CASCADE

);

**Важная информация для описания в курсовой:**

**1. Таблица users (Пользователи)**

* **Внешние ключи:** В этой таблице нет внешних ключей, так как это таблица, содержащая основную информацию о пользователях.
* **Описание:** Таблица хранит данные пользователей системы, включая их имя, email, пароль и метки, такие как "иностранный студент", "иногородний студент" и "первокурсник".

**2. Таблица tests (Тесты)**

* **Внешние ключи:** Нет внешних ключей в этой таблице, так как она служит для хранения данных о тестах.
* **Описание:** Таблица содержит информацию о тестах, таких как название теста и описание. Эта таблица используется для связывания тестов с вопросами и результатами.

**3. Таблица questions (Вопросы)**

* **Внешние ключи:**
  + test\_id — это внешний ключ, ссылающийся на таблицу tests, что означает, что каждый вопрос связан с конкретным тестом.
* **Описание:** Таблица хранит вопросы для тестов, их тип (например, выбор из нескольких вариантов) и правильные ответы.

**4. Таблица results (Результаты тестов)**

* **Внешние ключи:**
  + user\_id — внешний ключ, ссылающийся на таблицу users. Каждый результат теста связан с конкретным пользователем.
  + test\_id — внешний ключ, ссылающийся на таблицу tests. Каждый результат связан с определённым тестом.
* **Описание:** Таблица хранит результаты тестов, включая количество баллов, дату прохождения теста и данные о работе студента за неделю (если они запрашиваются). Эта таблица также отслеживает, проверялись ли рекомендации для каждого теста.

**5. Таблица recommendations (Рекомендации)**

* **Внешние ключи:**
  + test\_id — внешний ключ, ссылающийся на таблицу tests. Каждая рекомендация связана с определённым тестом.
* **Описание:** Таблица содержит рекомендации для студентов на основе результатов тестов. Для каждого теста могут быть определены разные рекомендации, основанные на набранных баллах.

**6. Таблица recommendation\_tracking (Отслеживание выполнения рекомендаций)**

* **Внешние ключи:**
  + user\_id — внешний ключ, ссылающийся на таблицу users. Каждый трекинг выполнения рекомендаций связан с пользователем.
  + result\_id — внешний ключ, ссылающийся на таблицу results. Каждый трекинг связан с конкретным результатом теста.
  + recommendation\_id — внешний ключ, ссылающийся на таблицу recommendations. Каждый трекинг связан с конкретной рекомендацией.
* **Описание:** Таблица отслеживает выполнение рекомендаций студентами, сохраняя информацию о статусе выполнения (выполнено, частично выполнено, не выполнено) и комментарии, если они есть. Также сохраняется дата выполнения.

**1. Таблица users (Пользователи)**

* **Связь:** Эта таблица **не имеет внешних ключей**, но она участвует в **связях один ко многим** с другими таблицами, например, с таблицей results (пользователь может иметь много результатов тестов).
* **Тип связи:** **Один ко многим** с таблицей results (один пользователь может иметь несколько результатов).

**2. Таблица tests (Тесты)**

* **Связь:** Эта таблица имеет связь с таблицами questions, results, recommendations (каждый тест может иметь много вопросов, результатов и рекомендаций).
* **Тип связи:**
  + **Один ко многим** с таблицей questions (один тест может иметь много вопросов).
  + **Один ко многим** с таблицей results (один тест может иметь много результатов).
  + **Один ко многим** с таблицей recommendations (один тест может иметь много рекомендаций).

**3. Таблица questions (Вопросы)**

* **Связь:** В этой таблице есть внешний ключ, который ссылается на таблицу tests. Каждый вопрос связан с одним тестом.
* **Тип связи:** **Многие к одному** с таблицей tests (каждый вопрос относится только к одному тесту, но тест может содержать много вопросов).

**4. Таблица results (Результаты тестов)**

* **Связь:**
  + Эта таблица связана с таблицей users и таблицей tests.
  + **Тип связи:**
    - **Многие к одному** с таблицей users (много результатов могут быть связаны с одним пользователем).
    - **Многие к одному** с таблицей tests (много результатов могут быть связаны с одним тестом).
    - **Один ко многим** с таблицей recommendation\_tracking (один результат может иметь много записей в recommendation\_tracking).

**5. Таблица recommendations (Рекомендации)**

* **Связь:** В этой таблице есть внешний ключ, который ссылается на таблицу tests. Каждая рекомендация связана с одним тестом.
* **Тип связи:** **Многие к одному** с таблицей tests (много рекомендаций могут быть связаны с одним тестом).

**6. Таблица recommendation\_tracking (Отслеживание выполнения рекомендаций)**

* **Связь:** Эта таблица используется для отслеживания выполнения рекомендаций пользователями и содержит внешние ключи, которые ссылаются на таблицы users, results, и recommendations. Каждая запись в recommendation\_tracking отслеживает выполнение конкретной рекомендации для конкретного пользователя и его результата.
* **Тип связи:**
  + **Многие к одному** с таблицей users (много записей о выполнении рекомендаций могут быть связаны с одним пользователем).
  + **Многие к одному** с таблицей results (много записей о выполнении рекомендаций могут быть связаны с одним результатом).
  + **Многие к одному** с таблицей recommendations (много записей о выполнении рекомендаций могут быть связаны с одной рекомендацией).

**Сложная связь (многие ко многим):** Таблица recommendation\_tracking обеспечивает связь **многие ко многим** между таблицами users, results, и recommendations. Один пользователь может выполнять несколько рекомендаций по результатам разных тестов, и каждая рекомендация может быть выполнена множеством пользователей.

**Итак, типы связей между таблицами:**

1. **Один ко многим (One to Many):**
   * **tests → questions:** Один тест может содержать много вопросов.
   * **tests → results:** Один тест может иметь много результатов.
   * **tests → recommendations:** Один тест может иметь много рекомендаций.
   * **users → results:** Один пользователь может иметь много результатов тестов.
   * **results → recommendation\_tracking:** Один результат может иметь много записей о выполнении рекомендаций.
2. **Многие ко одному (Many to One):**
   * **questions → tests:** Каждый вопрос связан с одним тестом, но один тест может иметь много вопросов.
   * **results → users:** Каждый результат связан с одним пользователем, но один пользователь может иметь много результатов.
   * **results → tests:** Каждый результат связан с одним тестом, но один тест может иметь много результатов.
   * **recommendations → tests:** Каждая рекомендация связана с одним тестом, но один тест может иметь много рекомендаций.
   * **recommendation\_tracking → users:** Каждый трекинг связан с одним пользователем, но один пользователь может иметь много записей.
   * **recommendation\_tracking → results:** Каждый трекинг связан с одним результатом, но один результат может иметь много записей.
   * **recommendation\_tracking → recommendations:** Каждый трекинг связан с одной рекомендацией, но одна рекомендация может быть выполнена множеством пользователей.
3. **Многие ко многим (Many to Many):**
   * **users ↔ recommendations через recommendation\_tracking:** Один пользователь может выполнить много рекомендаций, и каждая рекомендация может быть выполнена многими пользователями.