

ТЕСТОВОЕ ЗАДАНИЕ ДЛЯ ПРОГРАММИСТА JUNIOR BACKEND

ЦЕЛЬ

Реализовать упрощенную версию электронного журнала для оценок.

ТЕХНОЛОГИЧЕСКИЙ СТЕК

- FastAPI
- SQLAlchemy

ОПИСАНИЕ ЗАДАНИЯ

Создайте простое приложение для управления оценками учеников. Приложение должно включать следующие функции:

1. CRUD операции для учеников (Students):

- **Создание (Create):** Добавить нового ученика.
- **Чтение (Read):** Получить информацию об ученике.
- **Обновление (Update):** Обновить информацию об ученике.
- **Удаление (Delete):** Удалить ученика.

2. CRUD операции для оценок (Scores):

- **Создание (Create):** Добавить оценку ученику.
- **Чтение (Read):** Получить информацию об оценке.
- **Обновление (Update):** Обновить оценку ученика.
- **Удаление (Delete):** Удалить оценку.

ТРЕБОВАНИЯ К ПРОЕКТУ

1. FastAPI:

- Используйте FastAPI для создания API эндпоинтов для управления учениками и оценками.

2. SQLAlchemy:

- Используйте SQLAlchemy для взаимодействия с базой данных.
- Создайте модели для учеников и оценок.

ТЕСТОВОЕ ЗАДАНИЕ ДЛЯ ПРОГРАММИСТА JUNIOR BACKEND

ПРИМЕРНЫЕ ЭНДПОИНТЫ

1. Ученик (Student):

- POST /students/ — создать нового ученика.
- GET /students/{student_id} — получить информацию о конкретном ученике.
- PATCH /students/{student_id} — обновить информацию о конкретном ученике.
- DELETE /students/{student_id} — удалить ученика.

2. Оценка (Score):

- POST /score/ — добавить оценку ученику.
- GET /score/{score_id} — получить информацию о конкретной оценке.
- PATCH /score/{score_id} — обновить информацию о конкретной оценке.
- DELETE /score/{score_id} — удалить оценку.

ДОПОЛНИТЕЛЬНЫЕ ТРЕБОВАНИЯ

1. Документация:

- Добавьте краткую документацию о том, как запустить проект и использовать API эндпоинты.

ОЖИДАЕМЫЙ РЕЗУЛЬТАТ

- Репозиторий с исходным кодом на GitHub или другой платформе.
- Инструкции по запуску проекта и тестированию API.