### Задание: Разработка серверного приложения на Node.js с REST API

#### Цель:

Научиться создавать простое серверное приложение на Node.js с использованием фреймворка Express, реализовать базовые HTTP-роуты и протестировать их через Postman.

### Описание задания:

Необходимо разработать серверное приложение на Node.js, которое предоставляет следующие возможности через REST API:

1. **Получение ФИО студента** :
   * Реализовать эндпоинт для получения полного имени студента.
2. **Получение массива данных** :
   * Реализовать эндпоинт, который возвращает массив с произвольными данными (например, список книг, фильмов, товаров и т.д.). Структура данных должна быть осмысленной и содержать минимум 5 элементов.
3. **Тестирование через Postman** :
   * Протестировать созданные роуты с помощью Postman.
   * Сделать скриншоты результатов запросов для каждого эндпоинта.
4. **Отчёт** :
   * Написать краткий отчёт о проделанной работе, где необходимо:
     + Описать структуру проекта.
     + Показать примеры запросов и ответов через Postman.
     + Приложить скриншоты результатов тестирования.

### Требования к реализации:

1. **Сервер** :
   * Использовать Node.js и фреймворк Express для создания сервера.
   * Сервер должен запускаться на порту 3000 или другом порту по вашему выбору.
   * Все роуты должны быть документированы в коде с помощью комментариев.
2. **Эндпоинты** :
   * /student: Возвращает JSON с полным именем студента (ФИО).
   * /data: Возвращает JSON с массивом данных (произвольная структура).
3. **Postman** :
   * Протестировать каждый созданный эндпоинт.
   * Сохранить коллекцию запросов в Postman и экспортировать её вместе с проектом.
4. **Отчёт** :
   * Отчёт должен быть оформлен в формате .docx или .pdf.
   * Включить описание каждого эндпоинта, примеры запросов и ответов, а также скриншоты результатов тестирования.

### Критерии оценки:

1. **Корректность работы сервера** :
   * Сервер успешно запускается и обрабатывает запросы.
   * Все эндпоинты работают правильно.
2. **Структура кода** :
   * Код читаемый, содержит комментарии.
   * Используются лучшие практики программирования.
3. **Тестирование через Postman** :
   * Все эндпоинты протестированы.
   * Коллекция запросов сохранена и приложена к решению.
4. **Отчёт** :
   * Отчёт подробно описывает работу программы.
   * Содержит все необходимые скриншоты.

### Инструкция по сдаче:

1. Создайте архив (zip или tar.gz) с файлами проекта, включая:
   * Исходный код сервера.
   * Экспорт коллекции запросов из Postman.
   * Отчёт в формате .docx или .pdf.