Тест-кейсы к API-приложению «Bookstore REST API»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер теста | Название теста | Шаги | Ожидаемый результат | Статус |
| 1 | Создание автора | 1. Отправить POST /authors с {"name": "Толстой"} | Автор успешно создан | Пройдено |
| 2 | Получение автора | 1. Отправить GET /authors/{id} | Возвращается автор с указанным ID | Пройдено |
| 3 | Создание книги без автора | 1. Отправить POST /books с несуществующим author\_id | Ошибка 400 "Author does not exist" | Пройдено |
| 4 | Создание книги | 1. Создать автора 2. POST /books с author\_id | Книга успешно создается | Пройдено |
| 5 | Дубликат книги у автора | 1. Добавить вторую книгу тому же author\_id | Ошибка 400 "Author can have only one book" | Пройдено |
| 6 | Удаление автора | 1. DELETE /authors/{id} | Автор и книга удаляются (каскадное удаление) | Пройдено |
| 7 | Получение всех авторов | 1. GET /authors | Список всех авторов | Пройдено |
| 8 | Получение всех книг | 1. GET /books | Список всех книг | Пройдено |
| 9 | Фильтрация книг по author\_id | 1. GET /books?author\_id=X | Только книги автора X | Пройдено |
| 10 | Обновление книги | 1. PATCH /books/{id} с новым name | Имя книги обновляется | Пройдено |
| 11 | Обновление автора | 1. PATCH /authors/{id} с новым name | Имя автора обновляется | Пройдено |
| 12 | Удаление книги | 1. DELETE /books/{id} | Книга удаляется | Пройдено |
| 13 | Получение книги по ID | 1. GET /books/{id} | Возвращается книга с заданным ID | Пройдено |
| 14 | Ошибка при получении несуществующего автора | 1. GET /authors/9999 | Ошибка 404 "Author not found" | Пройдено |
| 15 | Ошибка при удалении несуществующей книги | 1. DELETE /books/9999 | Ошибка 400 "Book does not exist" | Пройдено |