WEB PROGRAMMING
CRUD API - Ruby On Rails

Submitted by: Group 3

- Soujanya Namburi
- Sreelakshmi K
- Ajay P Sakthikumar

Click here for the code

STEPS:

1. Create the application

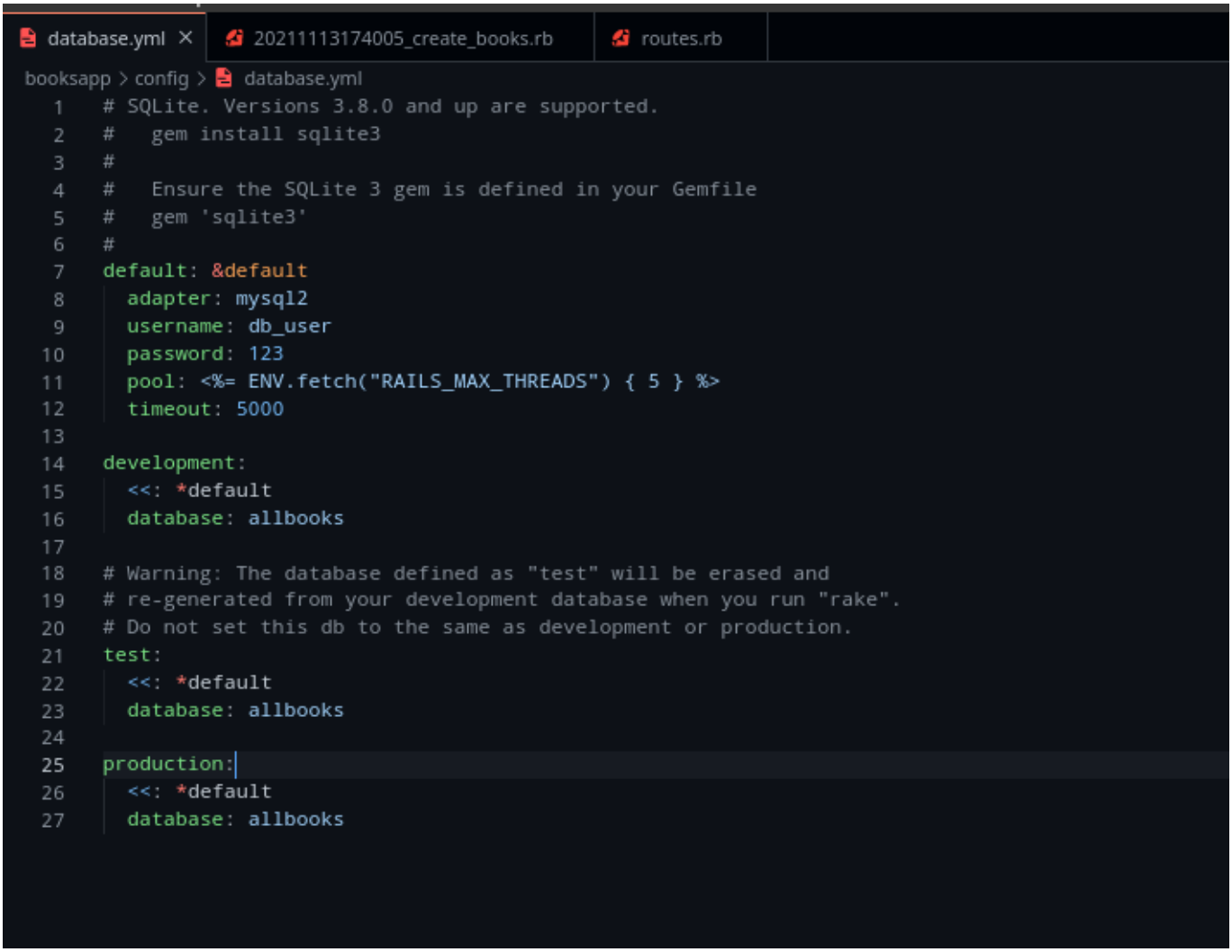
```
rails new booksapp --api
cd booksapp
```

2. Set up the database:

We are using mysql2 instead of sqlite3, so make the following change in the Gemfile.

```
# gem 'sqlite3'
gem 'mysql2'
```

Make the following changes to database.yml file as well



3. We use the MVC (Model View Controller) Architecture here. We create a model Book with the given attributes.

```
Book[B_id, B_title, Author, Publisher, Year]
```

PS. we will have another id element in the database. That is used to perform the CRUD operations. B_id is a different attribute

Run the following command to create a model:

```
rails g model Book B_title Author Year Publisher
```

The following files will be created on running the command:

```
    a. app/models/books.rb
    b. db/migrate/[date_time]_create_books.rb
```

This is the final schema: (In db/migrate/schema.rb)

```
ActiveRecord::Schema.define(version: 2021_11_13_174005) do

  create_table "books", charset: "utf8mb4", collation: "utf8mb4_unicode_ci", force: :cascade do |t|
    t.float "B_id"
    t.string "B_title"
    t.string "Author"
    t.string "Publisher"
    t.date "Year"
    t.datetime "created_at", precision: 6, null: false
    t.datetime "updated_at", precision: 6, null: false
  end

end
```

4. Now that we created the database with appropriate columns, we migrate it to the actual database using the following command:

```
rails db:migrate
```

5. We could also initiate values by editing the db/migrate/seeds.rb file. We have used an external library to make dummy values on which we could work on.

```
5.times do
    Book.create({
        B_id: Faker::Number.number(digits: 4),
        B_title: Faker::Book.title,
        Author: Faker::Book.author,
        Publisher: Faker::Book.publisher,
        Year: Faker::Date.between(from: '1900-09-23', to: '2014-09-25'),
    })
end
```

This can be initialised using the following command:

```
rails db:seed
```

6. Make an api folder in the app/controllers folder with the following contents: v1/books_controller.rb

```
module Api
    module V1
        class BooksController < ApplicationController
            def index
                books = Book.order('created_at DESC');
                render json: {status: 'SUCCESS', message:'Loaded books',
data:books},status: :ok
            end
```

```
            def show
                book = Book.find(params[:id])
                render json: {status: 'SUCCESS', message:'Loaded book',
data:book},status: :ok
            end

            def create
                book = Book.new(book_params)

                if book.save
                    render json: {status: 'SUCCESS', message:'Saved book',
data:book},status: :ok
                else
                    render json: {status: 'ERROR', message:'book not saved',
data:book.errors},status: :unprocessable_entity
                end
            end

            def destroy
                book = Book.find(params[:id])
                book.destroy
                render json: {status: 'SUCCESS', message:'Deleted book',
data:book},status: :ok
            end

            def update
                book = Book.find(params[:id])
                if book.update(book_params)
                    render json: {status: 'SUCCESS', message:'Updated book',
data:book},status: :ok
                else
                    render json: {status: 'ERROR', message:'book not updated',
data:book.errors},status: :unprocessable_entity
                end
            end

            private

            def book_params
                params.permit(:B_id, :B_title, :Author, :Publisher, :Year)
            end
        end
    end
  end
```

Define the above methods

7. Edit the routes.rb file to update the versions. The versions are made to made the API more scalable, config/routes.rb

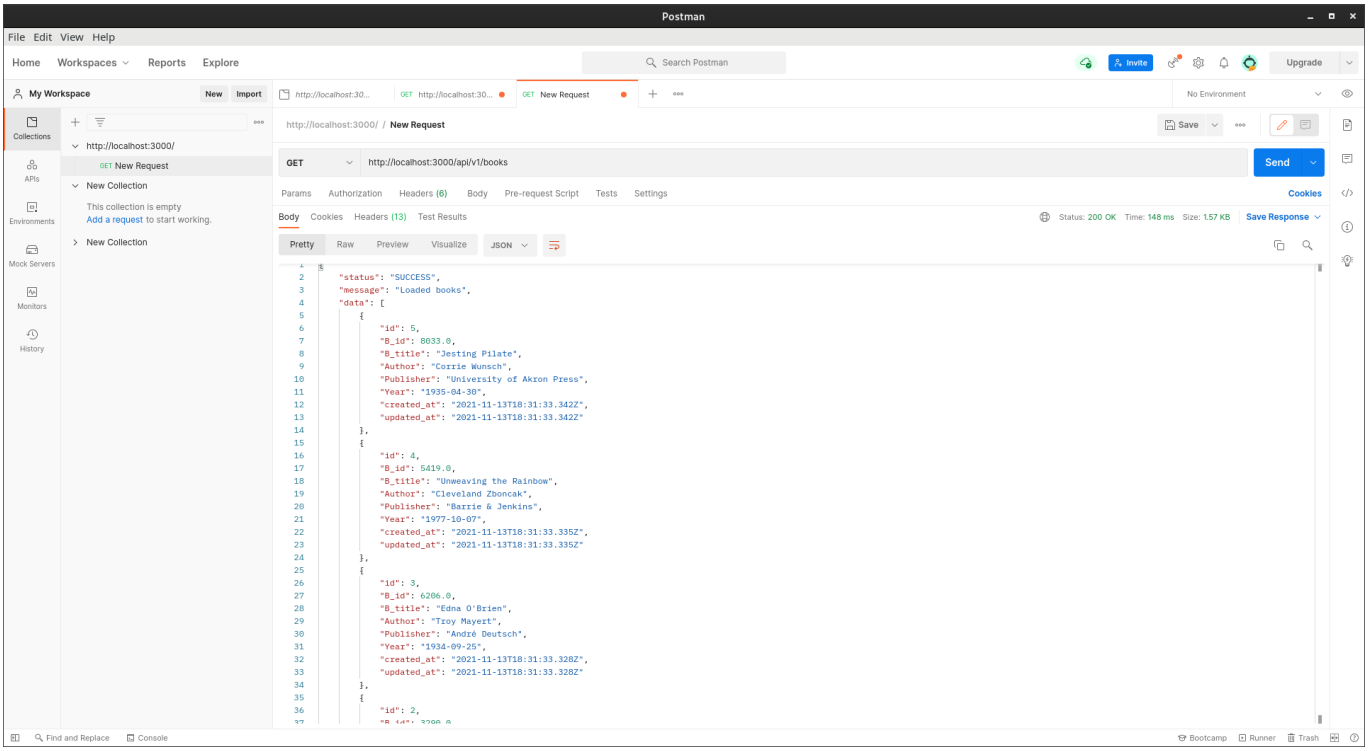
```
Rails.application.routes.draw do
  namespace 'api' do
    namespace 'v1' do
      resources :books
    end
  end
end
```

8. The API is ready. Run the app using the following command:

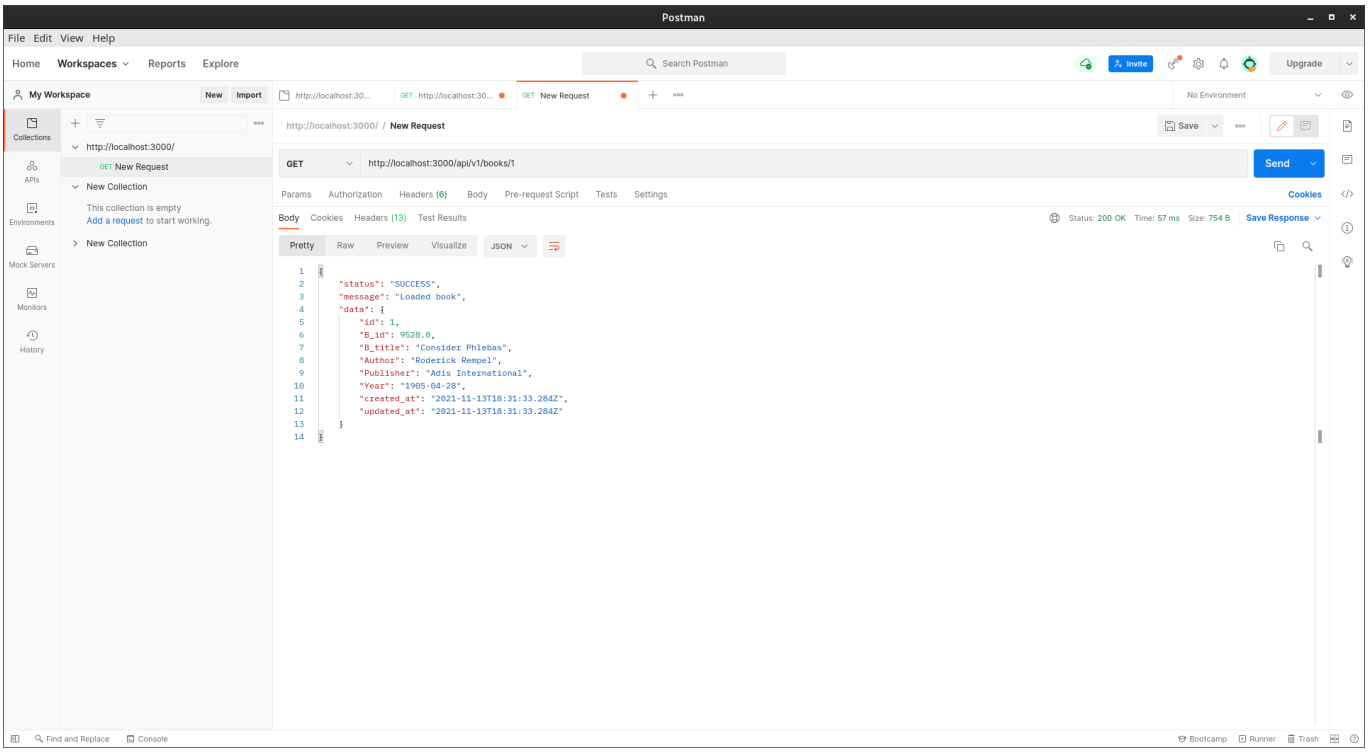
```
rails s
```

9. Give the below requests using POSTMAN or any other services

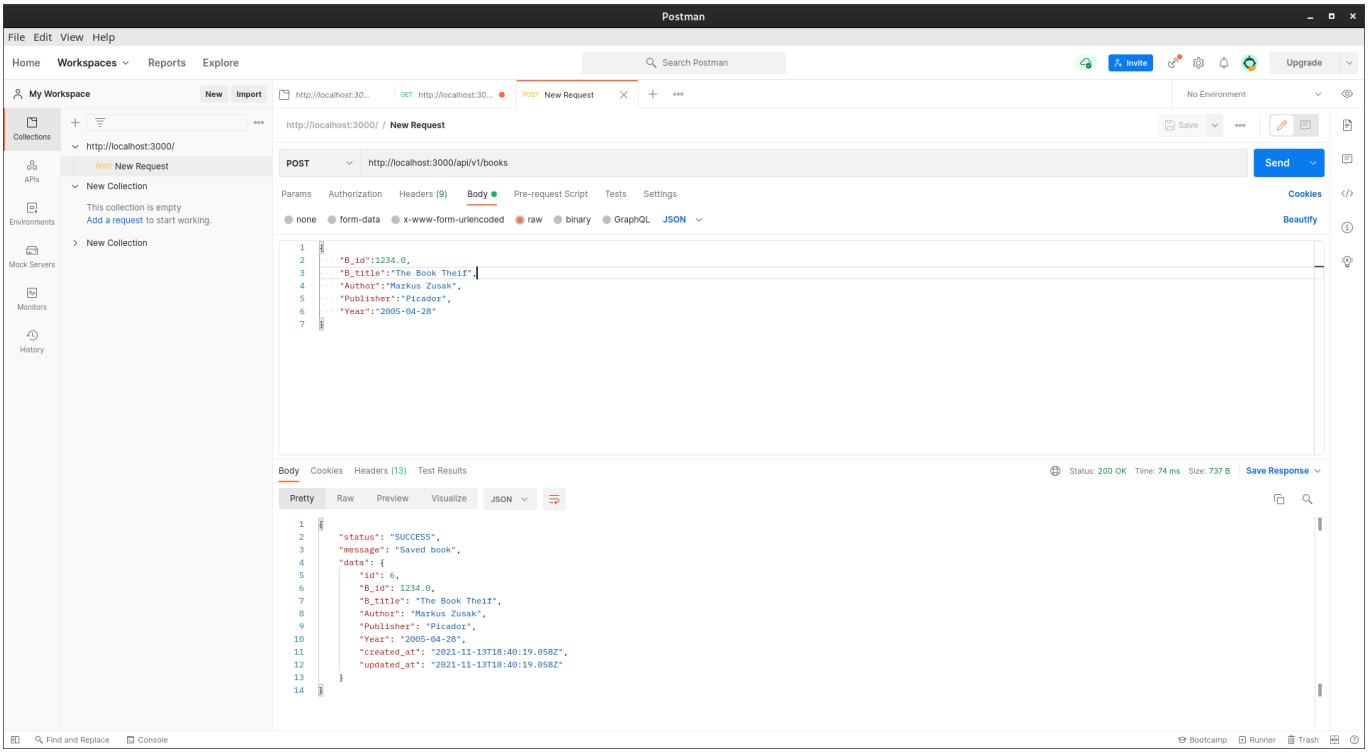
GET ALL



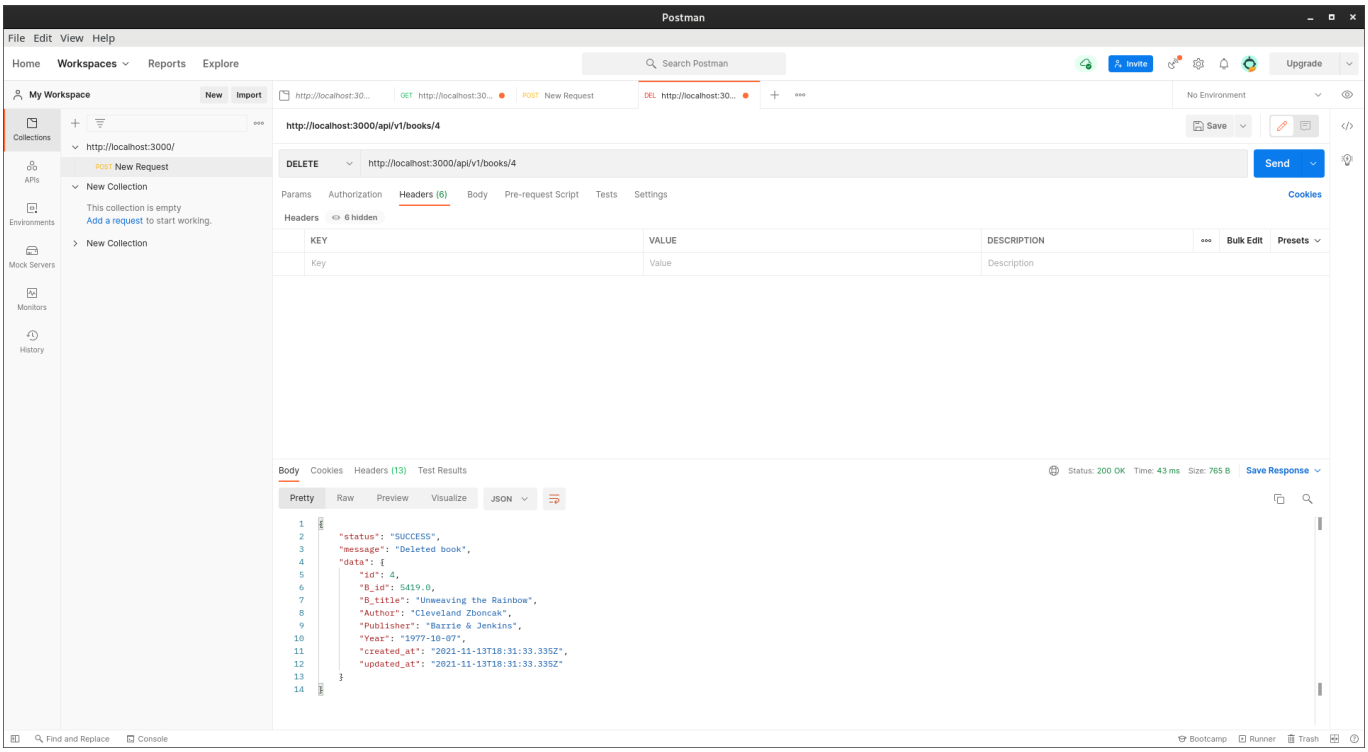
GET with certain ID


Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
My Workspace
New Import
Collections
APIs
Environments
Mock Servers
Monitors
History
http://localhost:3000/
GET New Request
New Collection
This collection is empty
Add a request to start working.
New Collection
http://localhost:3000/ / New Request
Save
GET
http://localhost:3000/api/v1/books/1
Send
Params Authorization Headers (6) Body Pre-request Script Tests Settings
Cookies
Body Cookies Headers (13) Test Results
Status: 200 OK Time: 57 ms Size: 754 B
Save Response
Pretty Raw Preview Visualize JSON
{
"status": "SUCCESS",
"message": "Loaded book",
"data": {
"id": 1,
"B_id": 9528.0,
"B_title": "Consider Phlebas",
"Author": "Roderick Rempel",
"Publisher": "Adis International",
"Year": "1905-04-28",
"created_at": "2021-11-13T18:31:33.284Z",
"updated_at": "2021-11-13T18:31:33.284Z"
}
Find and Replace
Console
Bootcamp
Runner
Trash

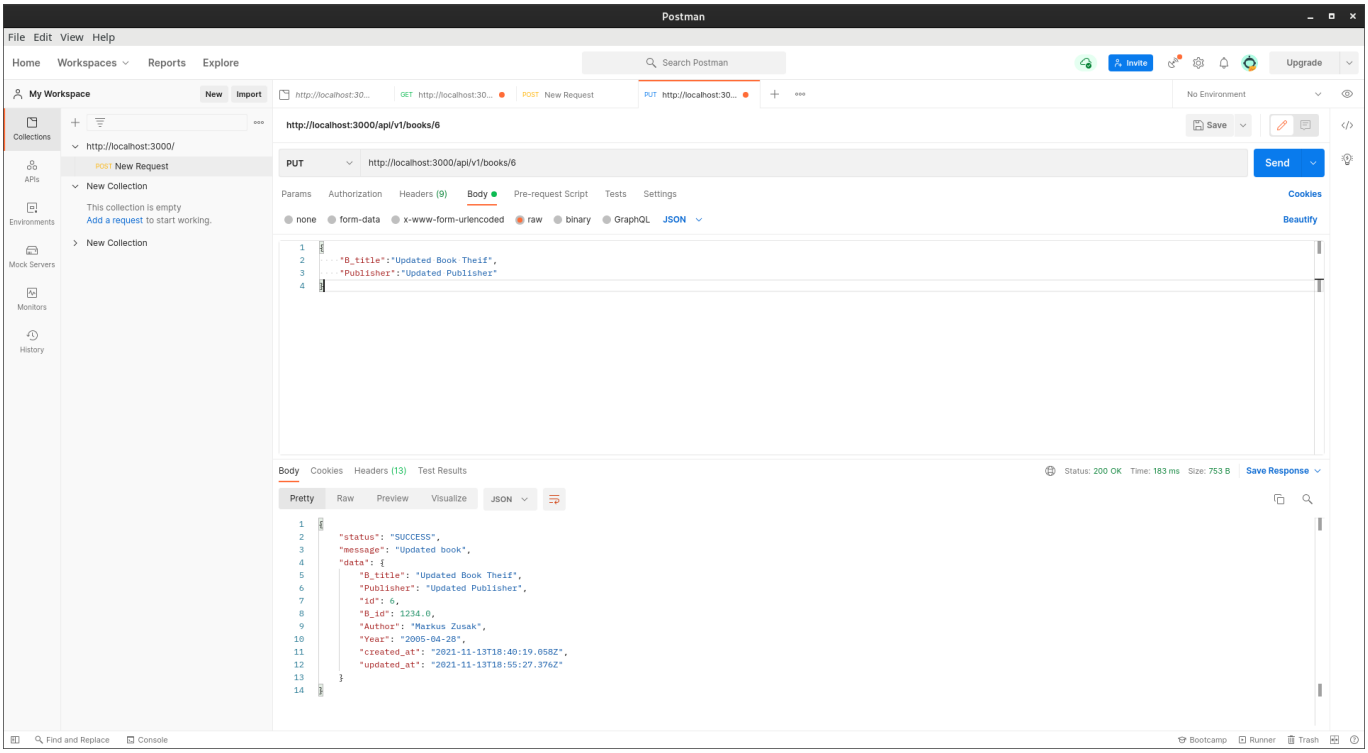
POST


Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
My Workspace
New Import
Collections
APIs
Environments
Mock Servers
Monitors
History
http://localhost:3000/
POST New Request
New Collection
This collection is empty
Add a request to start working.
New Collection
http://localhost:3000/ / New Request
Save
POST
http://localhost:3000/api/v1/books
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"B_id":1234.0,
"B_title":"The Book Theif",
"Author":"Markus Zusak",
"Publisher":"Picador",
"Year":"2005-04-28"
}
Body Cookies Headers (13) Test Results
Status: 200 OK Time: 74 ms Size: 737 B
Save Response
Pretty Raw Preview Visualize JSON
{
"status": "SUCCESS",
"message": "Saved book",
"data": {
"id": 6,
"B_id": 1234.0,
"B_title": "The Book Theif",
"Author": "Markus Zusak",
"Publisher": "Picador",
"Year": "2005-04-28",
"created_at": "2021-11-13T18:40:19.058Z",
"updated_at": "2021-11-13T18:40:19.058Z"
}
Find and Replace
Console
Bootcamp
Runner
Trash

DELETE


Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
My Workspace
New Import
Collections
APIs
Environments
Mock Servers
Monitors
History
http://localhost:3000/
POST New Request
New Collection
This collection is empty
Add a request to start working.
New Collection
http://localhost:3000/api/v1/books/4
Save
DELETE
http://localhost:3000/api/v1/books/4
Send
Params Authorization Headers (6) Body Pre-request Script Tests Settings
Cookies
Headers 6 hidden
KEY VALUE DESCRIPTION Bulk Edit Presets
Key Value Description
Body Cookies Headers (13) Test Results
Status: 200 OK Time: 43 ms Size: 765 B Save Response
Pretty Raw Preview Visualize JSON
{
"status": "SUCCESS",
"message": "Deleted book",
"data": {
"id": 4,
"B_id": 5419.0,
"B_title": "Unmeaving the Rainbow",
"Author": "Cleveland Zboncak",
"Publisher": "Barrie & Jenkins",
"Year": "1977-10-07",
"created_at": "2021-11-13T18:31:33.335Z",
"updated_at": "2021-11-13T18:31:33.335Z"
}
}
Find and Replace Console
Bootcamp Runner Trash

UPDATE


Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
My Workspace
New Import
Collections
APIs
Environments
Mock Servers
Monitors
History
http://localhost:3000/
POST New Request
New Collection
This collection is empty
Add a request to start working.
New Collection
http://localhost:3000/api/v1/books/6
Save
PUT
http://localhost:3000/api/v1/books/6
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"B_title":"Updated Book Theif",
"Publisher":"Updated Publisher"
}
Body Cookies Headers (13) Test Results
Status: 200 OK Time: 183 ms Size: 753 B Save Response
Pretty Raw Preview Visualize JSON
{
"status": "SUCCESS",
"message": "Updated book",
"data": {
"B_title": "Updated Book Theif",
"Publisher": "Updated Publisher",
"id": 6,
"B_id": 1234.0,
"Author": "Markus Zusak",
"Year": "2005-04-28",
"created_at": "2021-11-13T18:40:19.058Z",
"updated_at": "2021-11-13T18:55:27.376Z"
}
}
Find and Replace Console
Bootcamp Runner Trash