

WEB PROGRAMMING PROJECT

CRUD API - Ruby On Rails

Group - 10

Tushar Kumar Patni : B180122CS

Ritik Gautam : B180630CS

Aditya Jha : B180648CS

[Source Code](#)

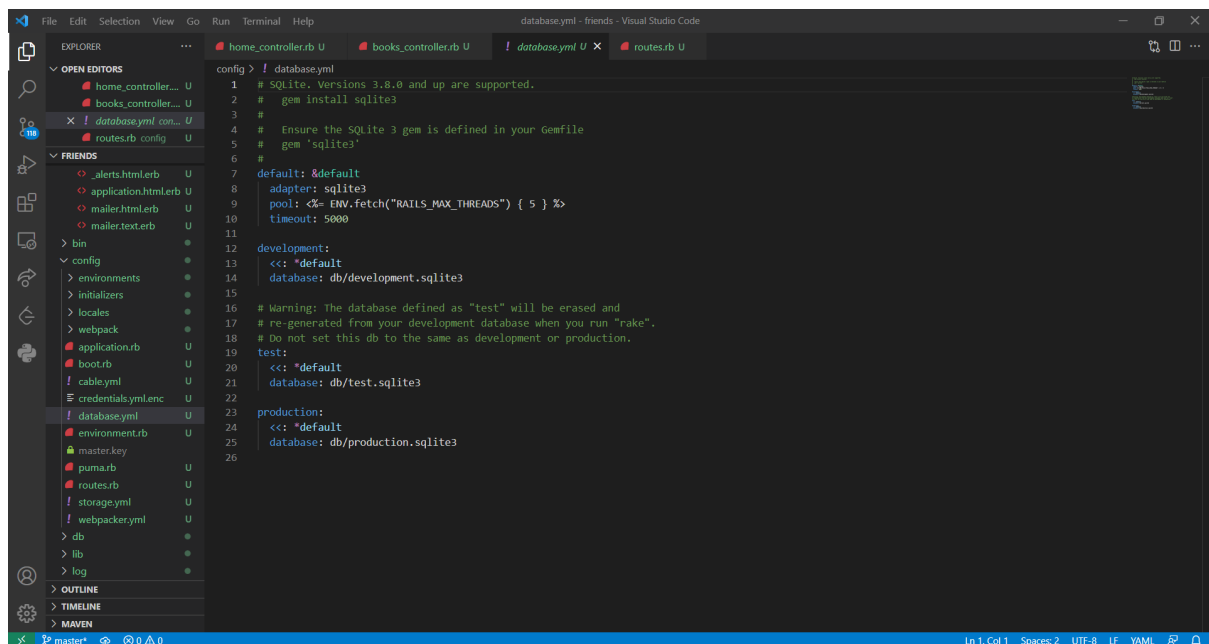
STEPS:

1. Create the application:

```
rails new books app --api  
cd FRIENDS
```

2. Setup the database: We are using sqlite3

```
gem 'sqlite3'
```



```
config > ! database.yml  
1 # SQLite. Versions 3.8.0 and up are supported.  
2 # gem install sqlite3  
3 #  
4 # Ensure the SQLite 3 gem is defined in your Gemfile  
5 # gem 'sqlite3'  
6 #  
7 default: &default  
8 adapter: sqlite3  
9 pool: <%= ENV.fetch("RAILS_MAX_THREADS") { 5 } %>  
10 timeout: 5000  
11  
12 development:  
13 <<: *default  
14 database: db/development.sqlite3  
15  
16 # Warning: The database defined as "test" will be erased and  
17 # re-generated from your development database when you run "rake".  
18 # Do not set this db to the same as development or production.  
19 test:  
20 <<: *default  
21 database: db/test.sqlite3  
22  
23 production:  
24 <<: *default  
25 database: db/production.sqlite3  
26
```

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

```
Book[Book_ID, Book_Title, Author, Publishers, Year]
```

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

Run the following command to create a model:

```
rails g model Book Book_ID Book_Title Author Year Publishers
```

The following files will be created on running the command:

- a. app/models/book.rb
- b. db/migrate/[date_time]_create_books.rb

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

```
12
13 ActiveRecord::Schema.define(version: 2021_11_15_151241) do
14
15   create_table "books", force: :cascade do |t|
16     t.integer "Book_ID"
17     t.string "Book_Title"
18     t.string "Author"
19     t.string "Publishers"
20     t.integer "Year"
21     t.datetime "created_at", precision: 6, null: false
22     t.datetime "updated_at", precision: 6, null: false
23   end
24
25 end
26
```

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

```
rails db:migrate
```

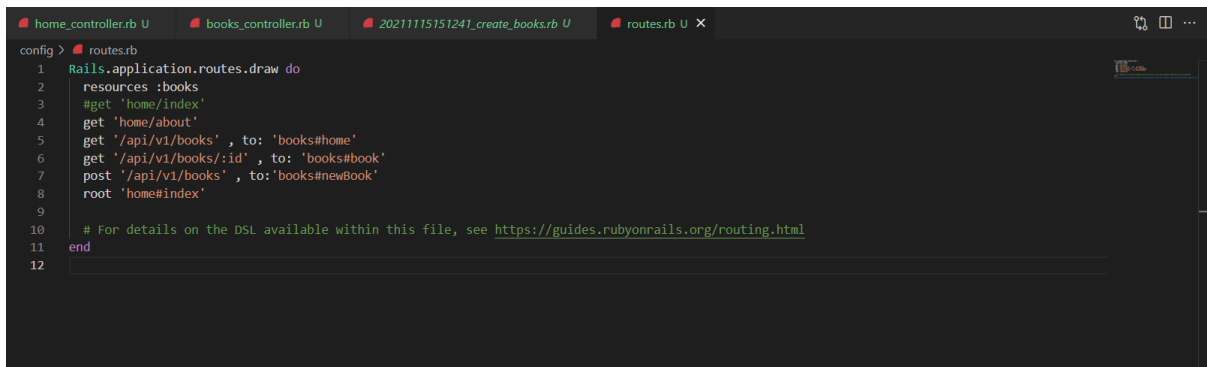
5. Make an api folder in the app/controllers folder with the following contents:

v1/books_controller.rb

```
home_controller.rb U books_controller.rb X 20211115151241_create_books.rb U routes.rb U
app > controllers > books_controller.rb
1 class BooksController < ApplicationController
2   skip_before_action :verify_authenticity_token
3   before_action :set_book, only: %i[ show edit update destroy ]
4
5   # GET /books or /books.json
6   def index
7     @books = Book.all
8   end
9   def home
10    @books = Book.all
11    render json: @books
12  end
13
14  def book
15    @book = Book.find(params[:id])
16    render json: @book
17  end
18  # GET /books/1 or /books/1.json
19  def show
20  end
21
22  def newBook
23    @book = Book.new(book_params)
24    if @book.save
25      render json: { book: @book, status: 'success' }
26    else
27      render json: { book: @book.errors, status: 'fail' }
28    end
29  end
30
31  # GET /books/new
32  def new
33    @book = Book.new
34  end
35
36  # GET /books/1/edit
37  def edit
```

```
home_controller.rb U books_controller.rb X 20211115151241_create_books.rb U routes.rb U
app > controllers > books_controller.rb
41 def create
42   @book = Book.new(book_params)
43
44   respond_to do |format|
45     if @book.save
46       format.html { redirect_to @book, notice: "Book was successfully created." }
47       format.json { render :show, status: :created, location: @book }
48     else
49       format.html { render :new, status: :unprocessable_entity }
50       format.json { render json: @book.errors, status: :unprocessable_entity }
51     end
52   end
53 end
54
55 # PATCH/PUT /books/1 or /books/1.json
56 def update
57   book = Book.find(params[:id])
58   if book.update(book_params)
59     render json: {status: 'SUCCESS', message:'Updated book', data:book},status: :ok
60   else
61     render json: {status: 'ERROR', message:'book not updated', data:book.errors},status: :unprocessable_entity
62   end
63 end
64
65 # DELETE /books/1 or /books/1.json
66 def destroy
67   book = Book.find(params[:id])
68   book.destroy
69   render json: {status: 'SUCCESS', message:'Deleted book', data:book},status: :ok
70 end
71
72 private
73 # Use callbacks to share common setup or constraints between actions.
74 def set_book
75   @book = Book.find(params[:id])
76 end
77
```

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



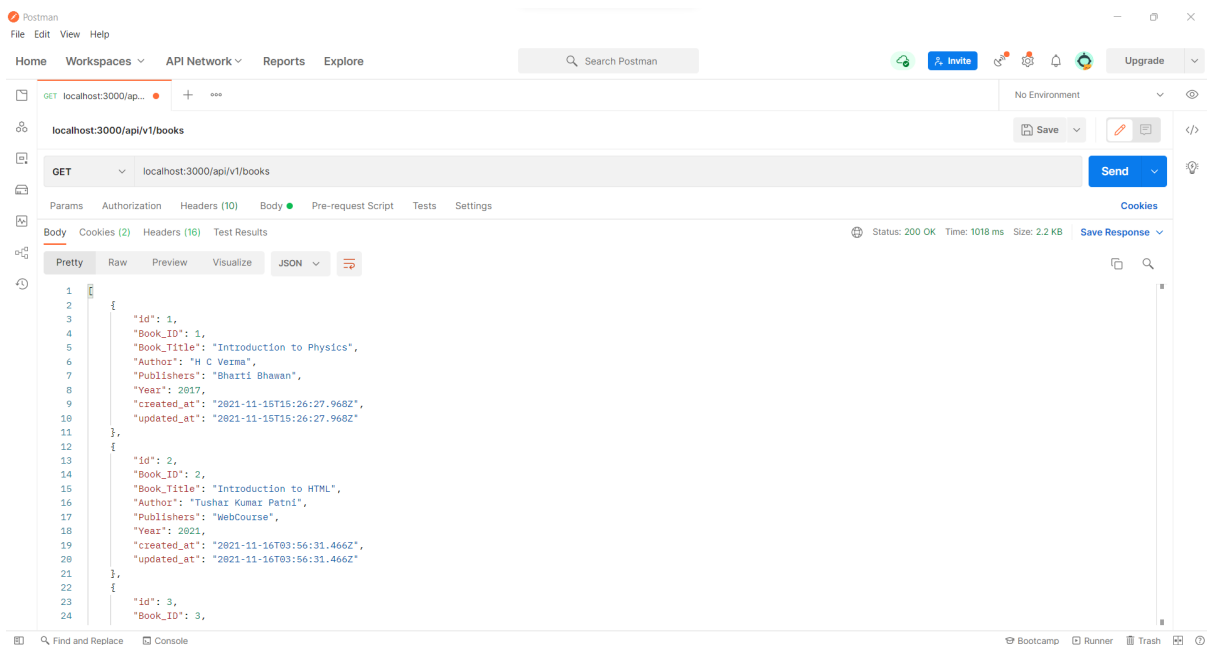
```
config/routes.rb
1 Rails.application.routes.draw do
2   resources :books
3   #get 'home/index'
4   get 'home/about'
5   get '/api/v1/books', to: 'books#home'
6   get '/api/v1/books/:id', to: 'books#book'
7   post '/api/v1/books', to: 'books#newBook'
8   root 'home#index'
9
10  # For details on the DSL available within this file, see https://guides.rubyonrails.org/routing.html
11 end
12
```

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

```
rails s
```

8. POSTMAN SCREENSHOTS (CRUD OPERATIONS)

1. GET ALL



```
GET localhost:3000/api/v1/books
Status: 200 OK Time: 1018 ms Size: 2.2 KB
{
  "id": 1,
  "Book_ID": 1,
  "Book_Title": "Introduction to Physics",
  "Author": "H C Verma",
  "Publishers": "Bharti Bhawan",
  "Year": 2017,
  "created_at": "2021-11-15T15:26:27.968Z",
  "updated_at": "2021-11-15T15:26:27.968Z"
},
{
  "id": 2,
  "Book_ID": 2,
  "Book_Title": "Introduction to HTML",
  "Author": "Tushar Kumar Patni",
  "Publishers": "webCourse",
  "Year": 2021,
  "created_at": "2021-11-16T03:56:31.466Z",
  "updated_at": "2021-11-16T03:56:31.466Z"
},
{
  "id": 3,
  "Book_ID": 3,
  "Book_Title": "Introduction to JavaScript",
  "Author": "Tushar Kumar Patni",
  "Publishers": "webCourse",
  "Year": 2021,
  "created_at": "2021-11-16T03:56:31.466Z",
  "updated_at": "2021-11-16T03:56:31.466Z"
}
```

2. GET BY ID

Postman interface showing a GET request to `localhost:3000/api/v1/books/3`. The response is a JSON object with the following details:

```
1 {
2   "id": 3,
3   "Book_ID": 3,
4   "Book_Title": "Javascript And Ajax",
5   "Author": "Aditya Jha",
6   "Publishers": "WebCourse",
7   "Year": 2019,
8   "created_at": "2021-11-16T03:56:41.734Z",
9   "updated_at": "2021-11-16T03:57:41.756Z"
10 }
```

Status: 200 OK Time: 851 ms Size: 1.26 KB Save Response

3. POST

Postman interface showing a POST request to `localhost:3000/api/v1/books/`. The request body is a JSON object with the following details:

```
1 {
2   "Book_ID": 1,
3   "Book_Title": "Introduction to DBMS",
4   "Author": "Aditya Jha",
5   "Publishers": "NIT CALICUT",
6   "Year": 2018
7 }
```

The response is a JSON object with the following details:

```
1 {
2   "book": {
3     "id": 8,
4     "Book_ID": 1,
5     "Book_Title": "Introduction to DBMS",
6     "Author": "Aditya Jha",
7     "Publishers": "NIT CALICUT",
8     "Year": 2018,
9     "created_at": "2021-11-16T10:44:53.100Z",
10    "updated_at": "2021-11-16T10:44:53.100Z"
11  },
12  "status": "success"
13 }
```

Status: 200 OK Time: 730 ms Size: 1.26 KB Save Response

4. PUT (UPDATE)

Postman interface showing a PUT request to `localhost:3000/books/3.json`. The request body is a JSON object:

```
{
  "Book_ID": 3,
  "Book_Title": "Introduction to Data Structures",
  "Author": "Tushar Kumar Patni",
  "Publishers": "NIT CALICUT",
  "Year": 2019
}
```

The response is a JSON object:

```
{
  "status": "SUCCESS",
  "message": "Updated book",
  "data": {
    "Book_ID": 3,
    "Book_Title": "Introduction to Data Structures",
    "Author": "Tushar Kumar Patni",
    "Publishers": "NIT CALICUT",
    "Year": 2019,
    "id": 3,
    "created_at": "2021-11-16T03:56:41.734Z",
    "updated_at": "2021-11-16T18:31:27.016Z"
  }
}
```

5. DELETE

Postman interface showing a DELETE request to `localhost:3000/books/2.json`. The response is a JSON object:

```
{
  "status": "SUCCESS",
  "message": "Deleted book",
  "data": {
    "id": 2,
    "Book_ID": 2,
    "Book_Title": "Introduction to HTML",
    "Author": "Tushar Kumar Patni",
    "Publishers": "WebCourse",
    "Year": 2021,
    "created_at": "2021-11-16T03:56:31.466Z",
    "updated_at": "2021-11-16T03:56:31.466Z"
  }
}
```

INDIVIDUAL CONTRIBUTIONS OF GROUP MEMBERS

TOKEN OF THANKS

We would like to thank Mrs Suvidha ks for her support for doing this project.