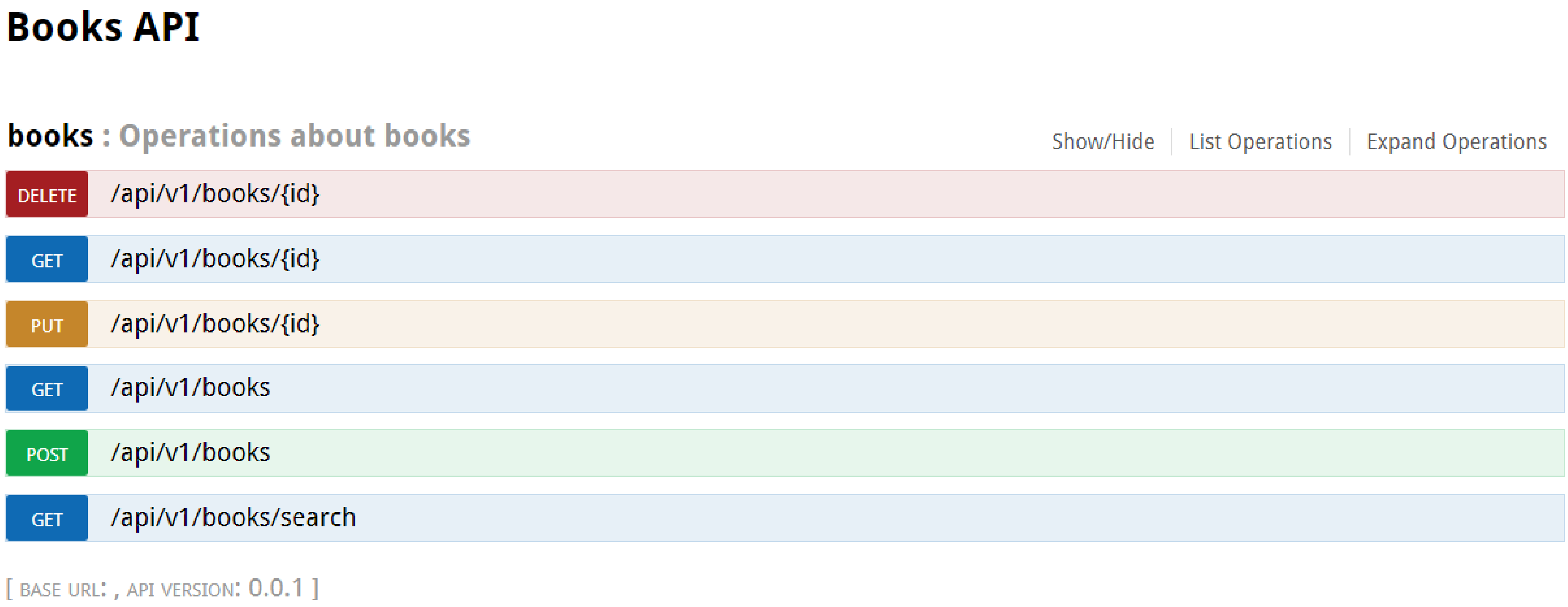
Ruby on Rails Grape API Example

API Endpoints



Books API

books : Operations about books

DELETE /api/v1/books/{id}

GET /api/v1/books/{id}

PUT /api/v1/books/{id}

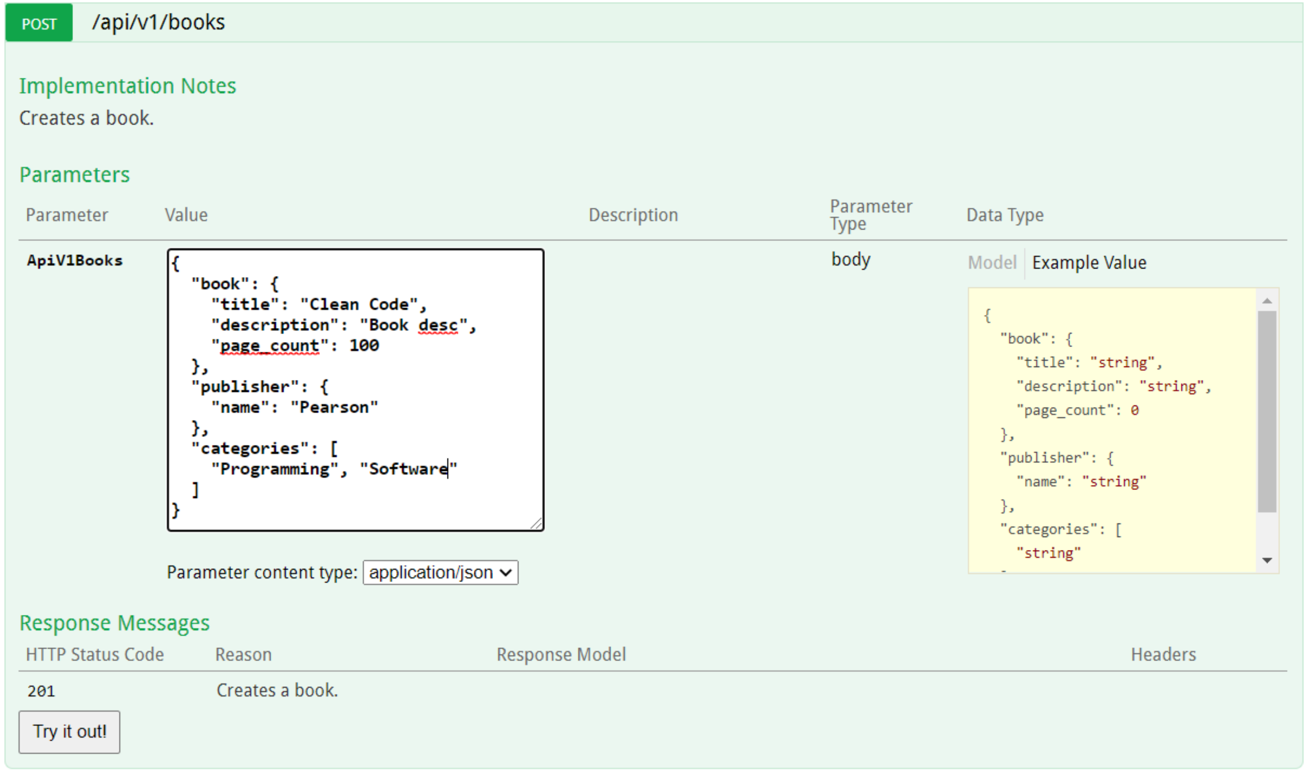
GET /api/v1/books

POST /api/v1/books

GET /api/v1/books/search

[BASE URL: , API VERSION: 0.0.1]

Create endpoint request



/api/v1/books

Implementation Notes

Creates a book.

Parameters

Parameter	Value	Description	Parameter Type	Data Type
ApiV1Books	(see below)		body	Model | Example Value

```
{
  "book": {
    "title": "Clean Code",
    "description": "Book desc",
    "page_count": 100
  },
  "publisher": {
    "name": "Pearson"
  },
  "categories": [
    "Programming", "Software"
  ]
}
```

Example Value:

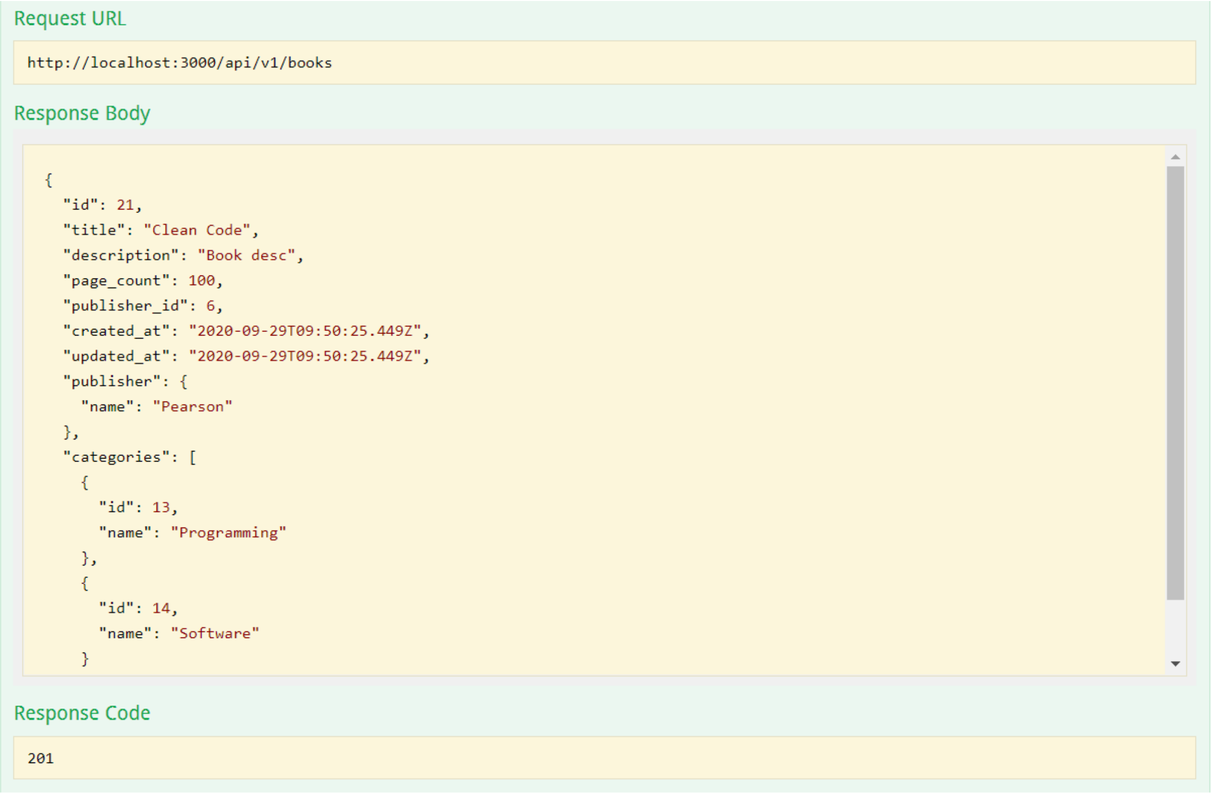
```
{
  "book": {
    "title": "string",
    "description": "string",
    "page_count": 0
  },
  "publisher": {
    "name": "string"
  },
  "categories": [
    "string"
```

Parameter content type: application/json

Response Messages

HTTP Status Code	Reason	Response Model	Headers
201	Creates a book.		

Create endpoint response



Request URL

```
http://localhost:3000/api/v1/books
```

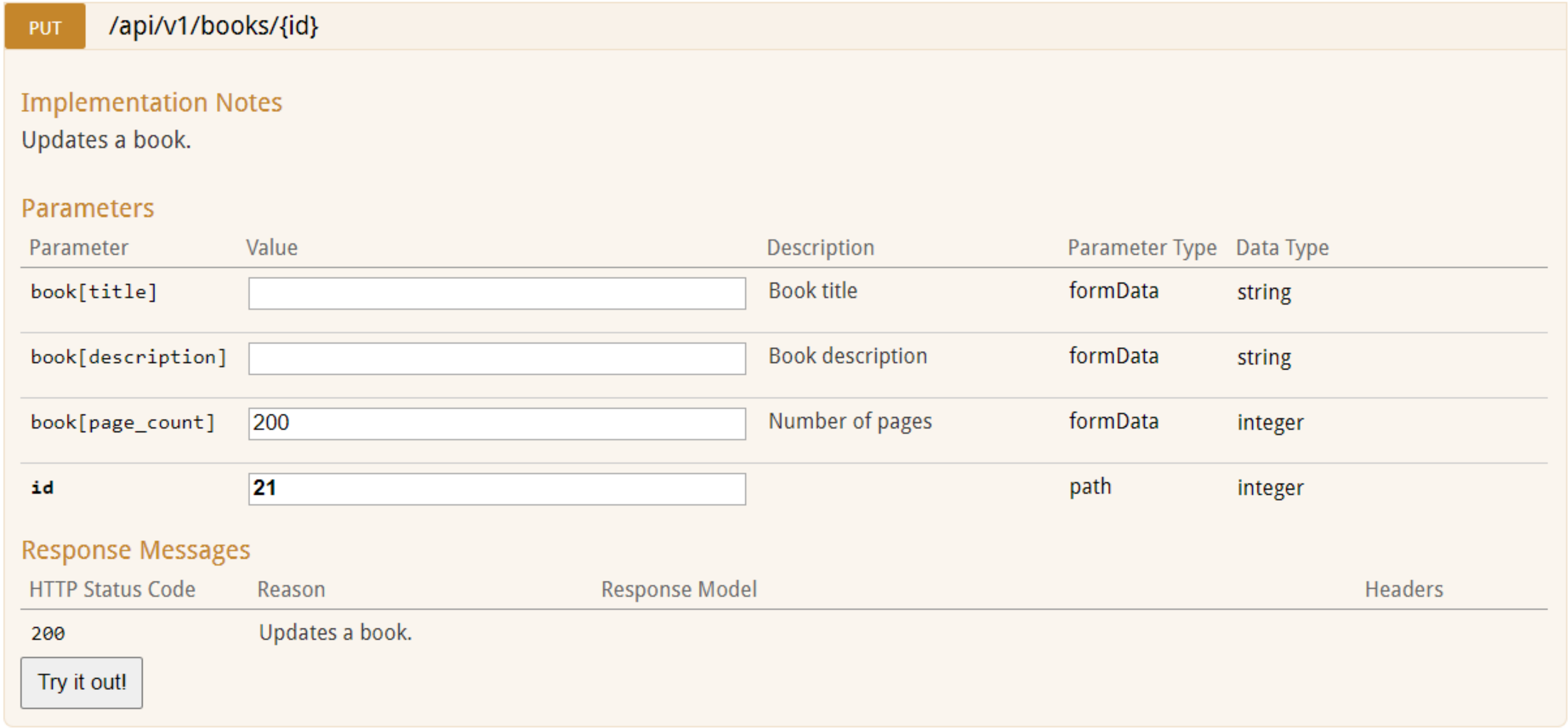
Response Body

```
{
  "id": 21,
  "title": "Clean Code",
  "description": "Book desc",
  "page_count": 100,
  "publisher_id": 6,
  "created_at": "2020-09-29T09:50:25.449Z",
  "updated_at": "2020-09-29T09:50:25.449Z",
  "publisher": {
    "name": "Pearson"
  },
  "categories": [
    {
      "id": 13,
      "name": "Programming"
    },
    {
      "id": 14,
      "name": "Software"
    }
```

Response Code

```
201
```

Update endpoint request



/api/v1/books/{id}

Implementation Notes

Updates a book.

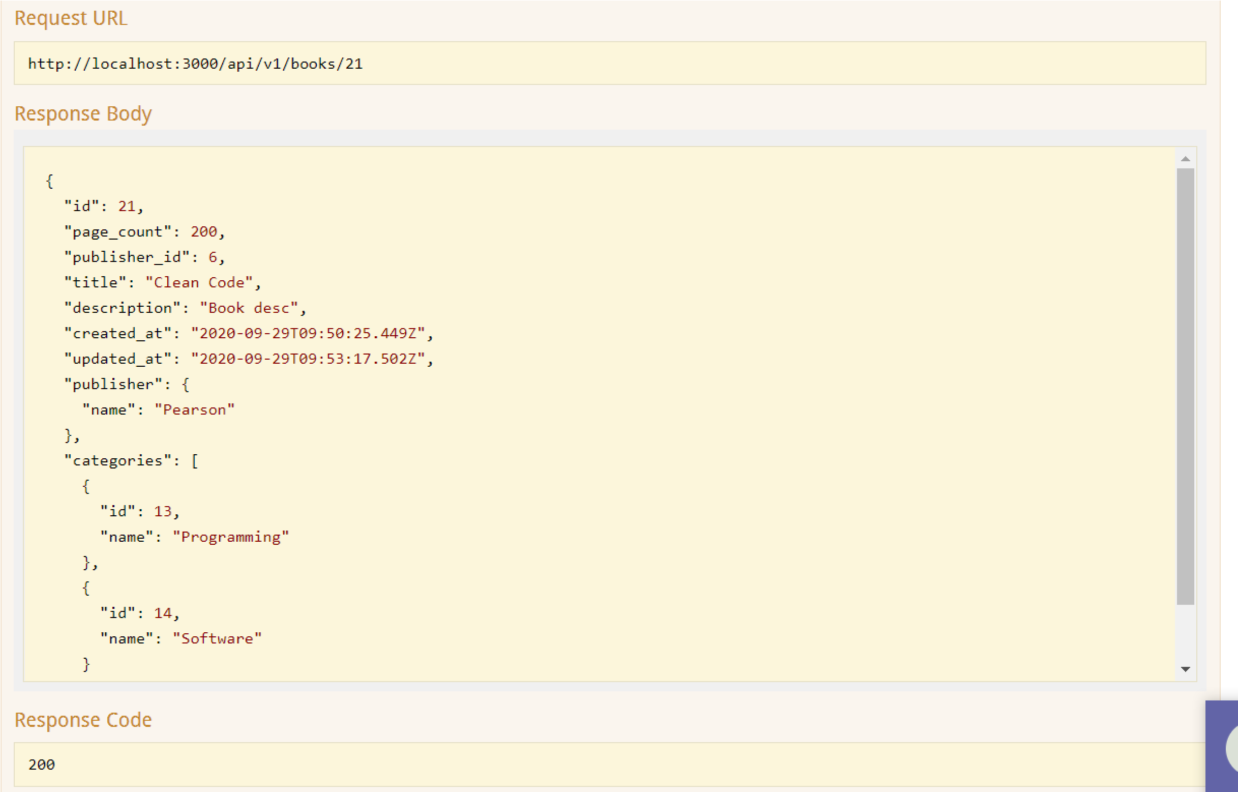
Parameters

Parameter	Value	Description	Parameter Type	Data Type
book[title]		Book title	formData	string
book[description]		Book description	formData	string
book[page_count]	200	Number of pages	formData	integer
id	**21**		path	integer

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	Updates a book.		

Update endpoint response



Request URL

```
http://localhost:3000/api/v1/books/21
```

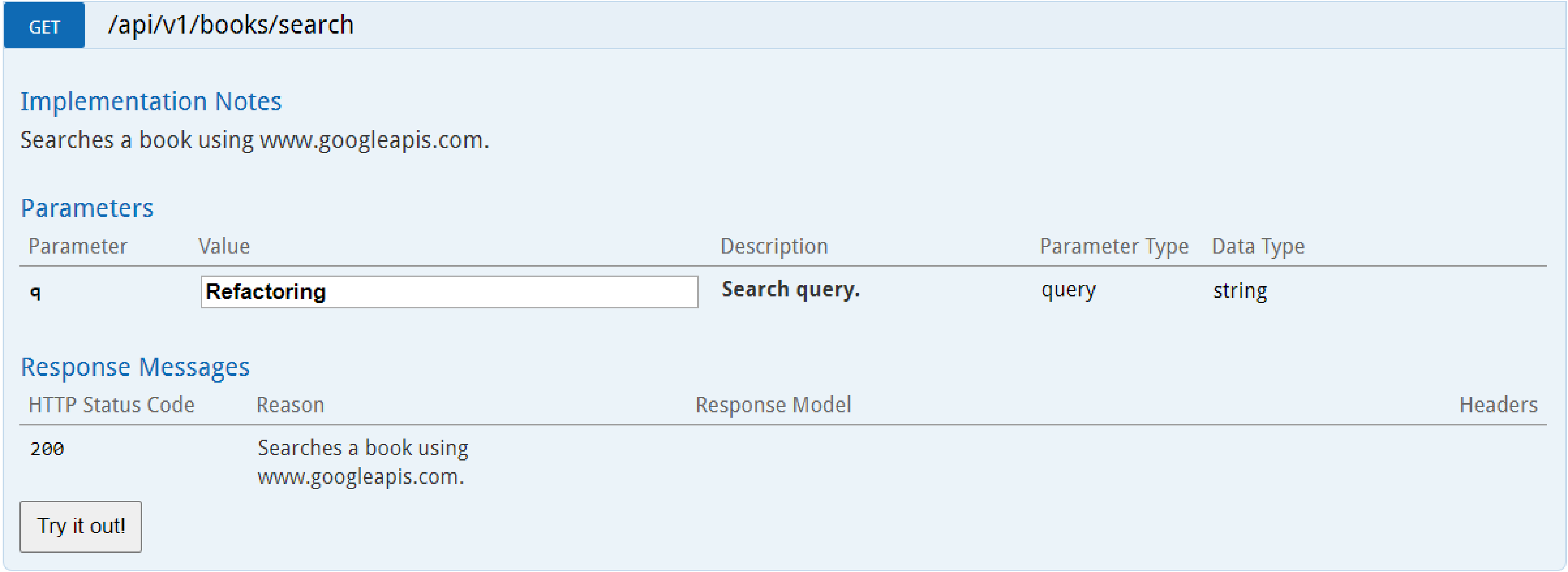
Response Body

```
{
  "id": 21,
  "page_count": 200,
  "publisher_id": 6,
  "title": "Clean Code",
  "description": "Book desc",
  "created_at": "2020-09-29T09:50:25.449Z",
  "updated_at": "2020-09-29T09:53:17.502Z",
  "publisher": {
    "name": "Pearson"
  },
  "categories": [
    {
      "id": 13,
      "name": "Programming"
    },
    {
      "id": 14,
      "name": "Software"
    }
```

Response Code

```
200
```

Search endpoint request



GET /api/v1/books/search

Implementation Notes

Searches a book using www.googleapis.com.

Parameters

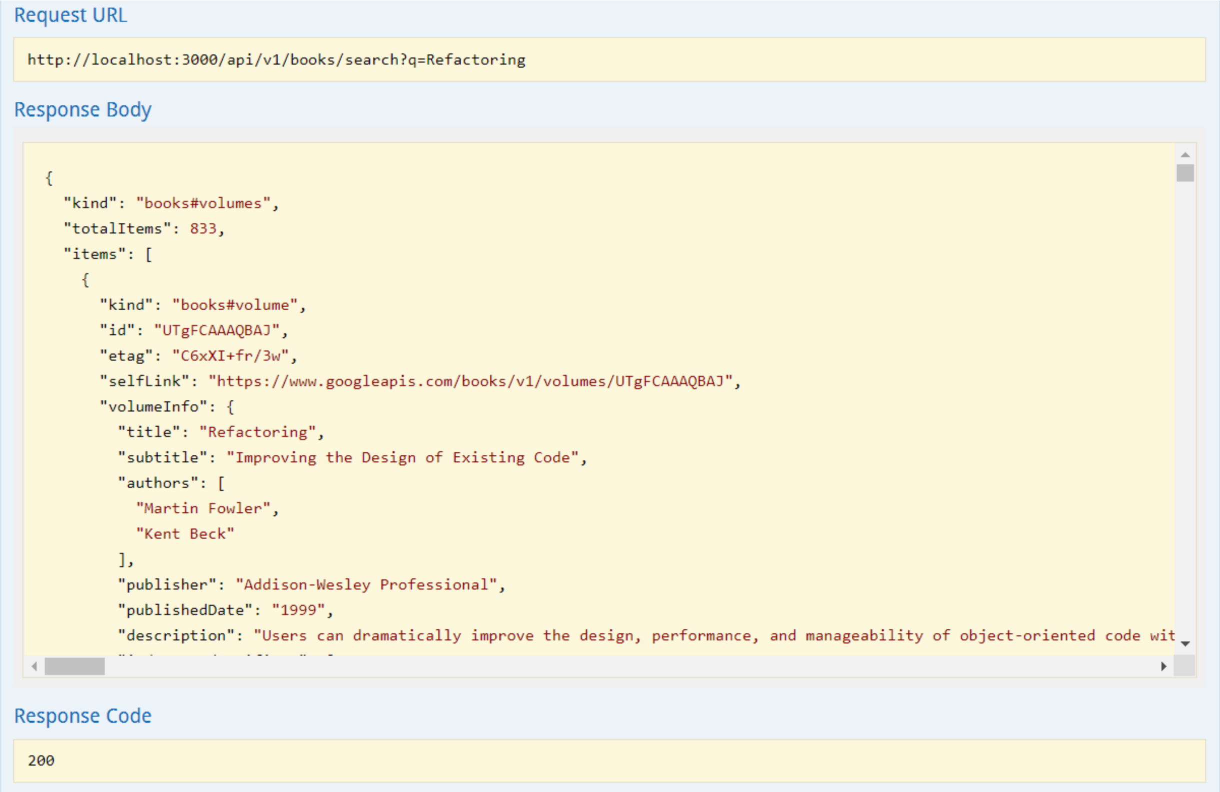
Parameter	Value	Description	Parameter Type	Data Type
q	Refactoring	**Search query.**	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	Searches a book using www.googleapis.com.		

Note: Use https://www.googleapis.com/books/v1/volumes?q=

Search endpoint response



Request URL

```
http://localhost:3000/api/v1/books/search?q=Refactoring
```

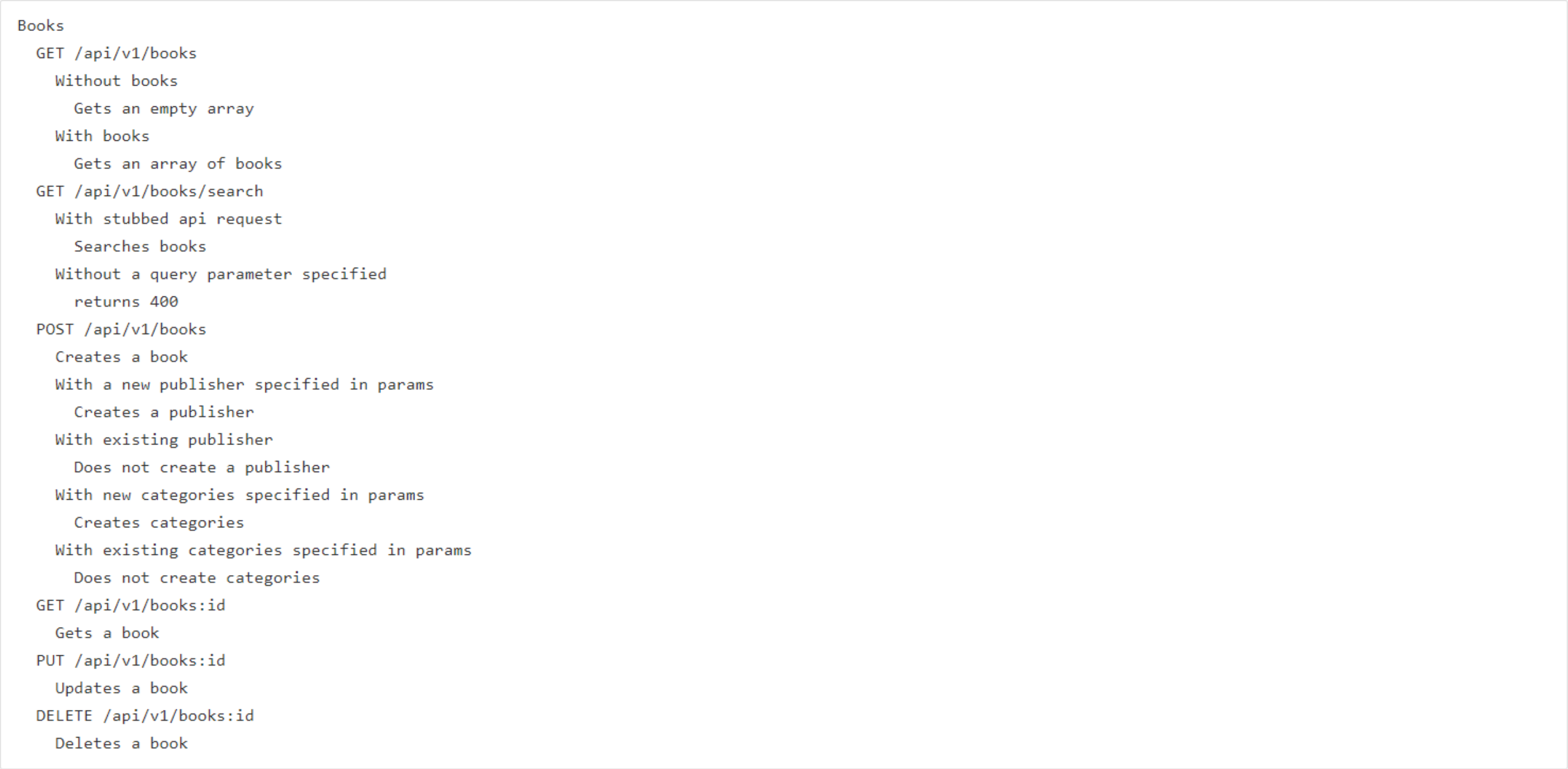
Response Body

```
{
  "kind": "books#volumes",
  "totalItems": 833,
  "items": [
    {
      "kind": "books#volume",
      "id": "UTgFCAAAQBAJ",
      "etag": "C6xXI+fr/3w",
      "selfLink": "https://www.googleapis.com/books/v1/volumes/UTgFCAAAQBAJ",
      "volumeInfo": {
        "title": "Refactoring",
        "subtitle": "Improving the Design of Existing Code",
        "authors": [
          "Martin Fowler",
          "Kent Beck"
        ],
        "publisher": "Addison-Wesley Professional",
        "publishedDate": "1999",
        "description": "Users can dramatically improve the design, performance, and manageability of object-oriented code wit
```

Response Code

```
200
```

Spec example



```
Books
  GET /api/v1/books
    Without books
      Gets an empty array
    With books
      Gets an array of books
  GET /api/v1/books/search
    With stubbed api request
      Searches books
    Without a query parameter specified
      returns 400
  POST /api/v1/books
    Creates a book
    With a new publisher specified in params
      Creates a publisher
    With existing publisher
      Does not create a publisher
    With new categories specified in params
      Creates categories
    With existing categories specified in params
      Does not create categories
  GET /api/v1/books:id
    Gets a book
  PUT /api/v1/books:id
    Updates a book
  DELETE /api/v1/books:id
    Deletes a book
```