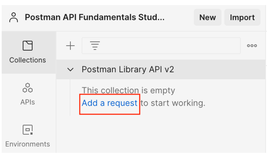
Task: Get books from the Library API

First things first: a librarian must know how to view all the books in the library catalog.

According to the API documentation, you can get all the books in the library by making a request to `GET https://library-api.postmanlabs.com/books`. Here, `GET` is the **request method,** and the **request URL** indicates where the request is sent. We'll cover what that means soon - but first, let's get our hands dirty with our first request!

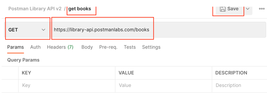
Make your first request.

1. Create a new request by either clicking **Add a request** inside your new Collection or hovering on your Collection, then click the three dots icon and "Add request"



2. Name your request "**get books**". Set the request method to **GET**, and the **request URL** to `GET https://library-api.postmanlabs.com/books`

Previous Next

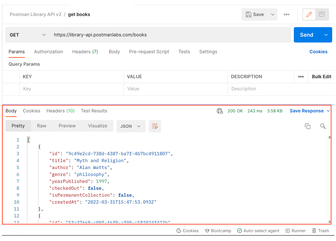


3. **Send** your request by clicking the Send button

View the response

If everything goes well, you will see a response from the server in the lower half of Postman.

It should look like this: a JSON (JavaScript Object Notation) response body with an **array** of book **objects**. You can scroll down to see more books.



You might see different books because this public library is being modified in real-time by other Postman librarians worldwide!

Request methods

When we make an HTTP call to a server, we specify a **request method** that indicates the type of operation we are about to perform. These are also called **HTTP verbs**.

Some common HTTP request methods correspond to the CRUD operations mentioned earlier. You can see a list of more methods here.

Method name	Operation
`GET`	Retrieve data (**R**ead)
`POST`	Send data (**C**reate)
`PUT/PATCH`	Update data (**U**pdate) * `PUT` usually replaces an entire resource, whereas `PATCH` usually is for partial updates
`DELETE`	Delete data (**D**elete)

Since we are "getting" books and not modifying any data, it makes sense that we are making a `GET` request.

These are just conventions - it all depends on how the API is coded. To know which method to use, always read the documentation for the API you're working with!

Refer to the API Documentation here: Postman Library API v2 docs

Request URL

In addition to a request method, a request must include a **request URL** that indicates *where* to make the API call. A request URL has three parts: a **protocol** (such as `http://` or `https://`), **host** (location of the server), and **path** (route on the server). In REST APIs, the path often points to a reference entity, like "books".

Protocol	Host	Path
`https://`	`library-api.postmanlabs.com`	`/books`

Paths and complete URLs are also sometimes called **API endpoints**.

Response status codes

The Postman Library API v2 has returned a **response status code** of "**200 OK**". Status codes are indicators of whether a request failed or succeeded.

Status codes have conventions. For example, any status code starting with a "2xx" (a "200-level response") represents a successful call. Get familiar with other status code categories:

Code range	Meaning	Example
`2xx`	Success	`200` - OK `201` - Created `204` - No content (silent OK)
`3xx`	Redirection	`301` - Moved (path changed)
`4xx`	Client error	`400` - Bad request `401` - Unauthorized `403` - Not Permitted `404` - Not Found
`5xx`	Server error	`500` - Internal server error `502` - Bad gateway `504` - Gateway timeout

That's a lot to remember 😱! Have no fear - in Postman, you can hover over any response code to see what it means.