

Task: Add a book

Task: Add an authorization header

Task: Use Postman Auth instead!

Quiz #5: Sending Data With Postman

Introduction to variables and scripting

Variables in Postman (Continued)

Setting variables programmatically

Task: Your first script

Task: Grab the new book id

Halfway Test

Quiz #6: Intro to Variables

PATCH and DELETE

Task: Checkout your book

Task: Delete your book

Generating code

Postman's codegen feature

Wrapping Up

Skillcheck (required)

Recap

Submit your Postman collection

Claim your badge!

Feedback

Postman Student Expert | Frequently Asked Questions

Postman Student Expert | Frequently Asked Questions

Get Help

Open Postman

Task: Checkout your book

Someone wants to check out the book you just added!

As a librarian, you will update the library database via the API to mark the book's **checkedOut** status from **false** to **true**.

The **API documentation** shows we can **update a book by id** by making a request (authorized with the API Key) with the updated information to:

**PATCH** **https://library-api.postmanlabs.com/books/:id**

Make a request to update the book

1. Hover on your **Postman Library API v2** collection, click the three dots, and select **Add request**. Name your new request **"checkout a book"**

2. Set the request method to **PATCH**

3. Set the request URL to **{{baseUr1}}/books/:id**

4. Set the value of the path variable **id** to **{{id}}**.

This will use the value of our collection variable named **id** that was set in the **Test** script of the"add a

< Previous

Next

Postman Library API v2 / checkout a book

Save

PATCH

{{baseUr1}}/books/id

Params

Auth

Headers (9)

Body

Pre-req.

Tests

Settings

Query Params

	KEY	VALUE	DESCRIPTION
	Key	Value	Description

Path Variables

	KEY	VALUE	DESCRIPTION
	id	{{id}}	Description

Response

C id

INITIAL

CURRENT f04ed19a-ff31-4777-805d-1d4cc5878d00

SCOPE Collection

5. Add a **raw JSON** body in the **Body** tab to update the **checkedOut** property to **true** :

```
{  "checkedOut": true}
```

6. **Save and Send** your request

Note: This request requires **Authorization**, which is automatically added as we added the collection level auth. The request is set to "Inherit from parent", this will use the API Key set at the collection level on our PATCH request.

You should get a **200 OK** response that shows the updated data about your book. Notice how **checkedOut** is now **true**

Home

Workspaces

API Network

Search Postman

Invite

Team

Postman API Fundamentals Stud...

New

Import

Overview

Postma

GET get box

GET get fict

GET get box

POST add b

PATCH chec

+

No environment

Collections

Environments

History

Postman Library API v2

GET get books

GET get fiction books

GET get book by id

POST add book

PATCH checkout book

Postman Library API v2 / checkout book

Save

Share

PATCH

{{baseUr1}}/books/id

Send

Params

Authorization

Headers (10)

Body

Scripts

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1 {

2 "checkedOut": true

3 }

Body

Cookies

Headers (10)

Test Results

Security

200 OK

193 ms

653 B

Save as example

Pretty

Raw

Preview

Visualize

JSON

1 {

2 "id": "7720b5f7-8865-4166-b6ef-9f4286498bba",

3 "title": "On the Taboo Against Knowing Who You Are",

4 "author": "Alan Watts",

5 "genze": "philosophy",

6 "yearPublished": 1968,

7 "checkedOut": true,

8 "isPeimanentCollection": false,

9 "createdAt": "2024-06-26T14:58:37.887Z"

10 }

Your book is updated!

Now, if you return to your "get book by id" request, update the id path variable value to **{{id}}** **Save** and **Send**, you will see the same updated data!