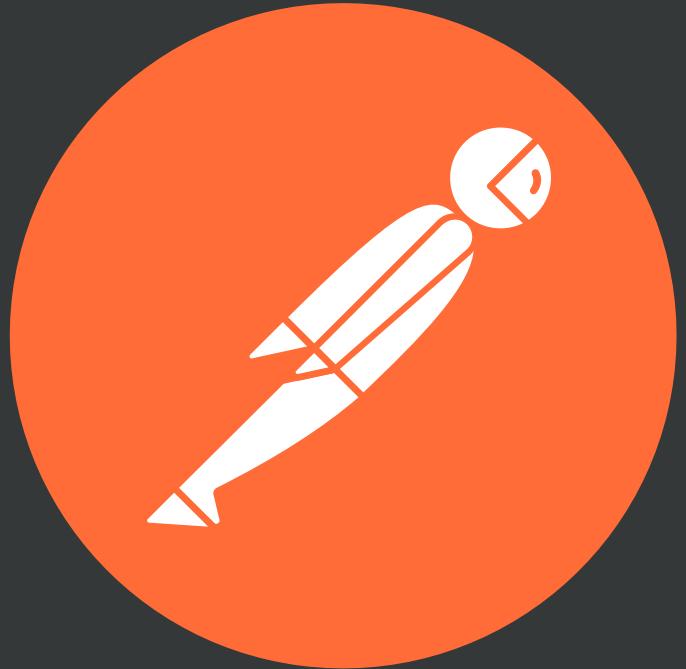


POSTMAN

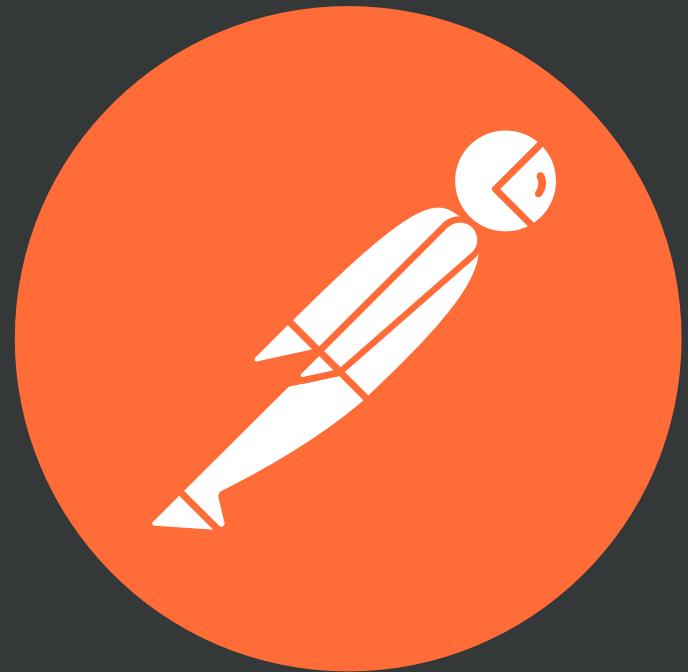


THE LEGEND OF
P O S T M A N
BREATH OF THE API



Not just the
POSTMAN





POSTMAN

And

newman

API Testing Made Easy

Caleb Meyer



the Caleb Meyer

Senior Software Engineer
Dad of the Millennium (Falcon)
Arbiter of Documentation
Script Ninja
Veteran Mercenary

History
Basics
Requests
Collections
Variables
Testing
Advanced

Covered Today

Postman started as
a ~~Chrome Extension~~
Side Project

Postman

Home Workspaces Reports Explore Search Postman Sign In Create Account

Prod Zone 2

Untitled Request

Save Send

GET Enter request URL

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

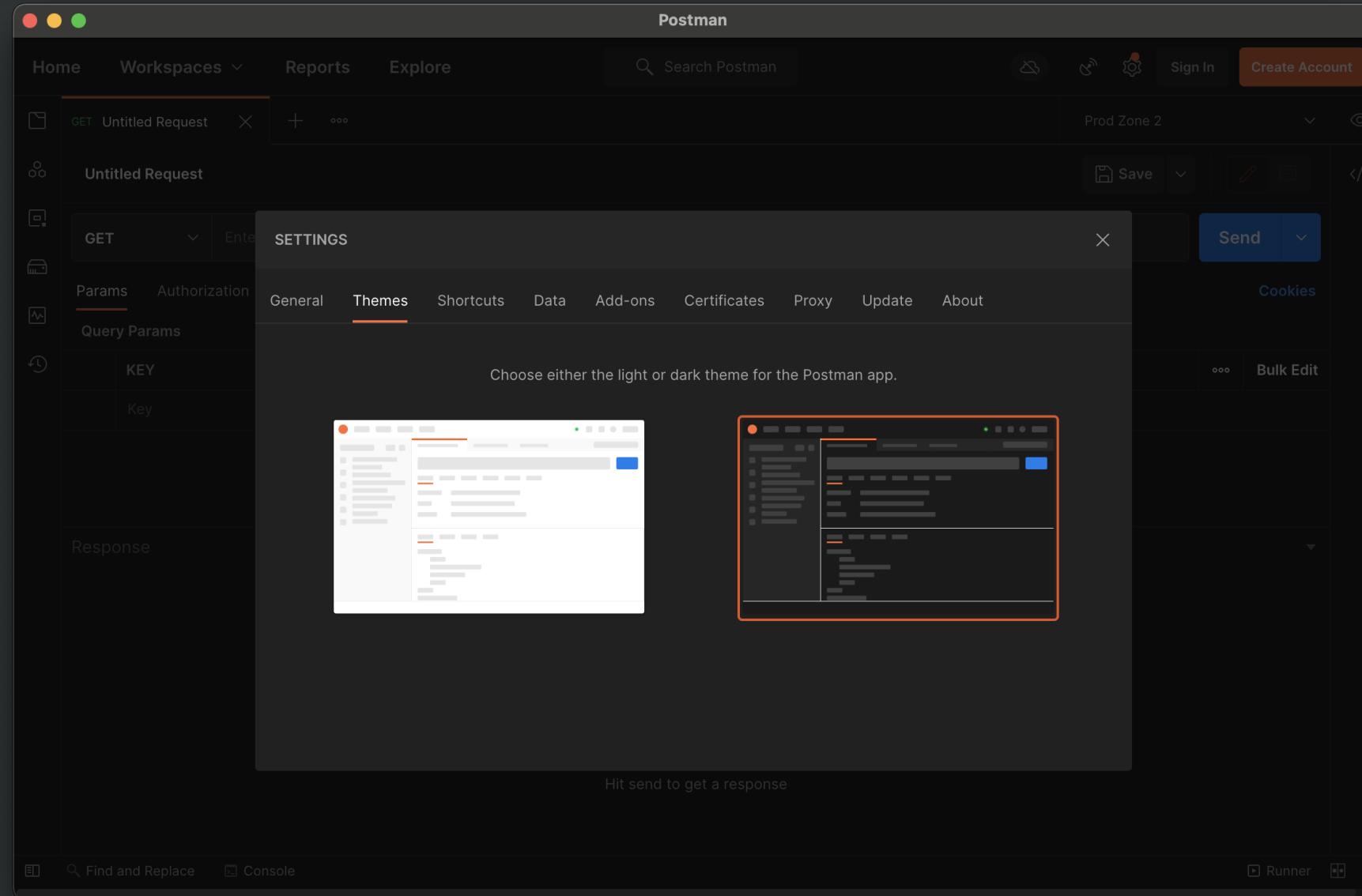
	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Response

Hit send to get a response

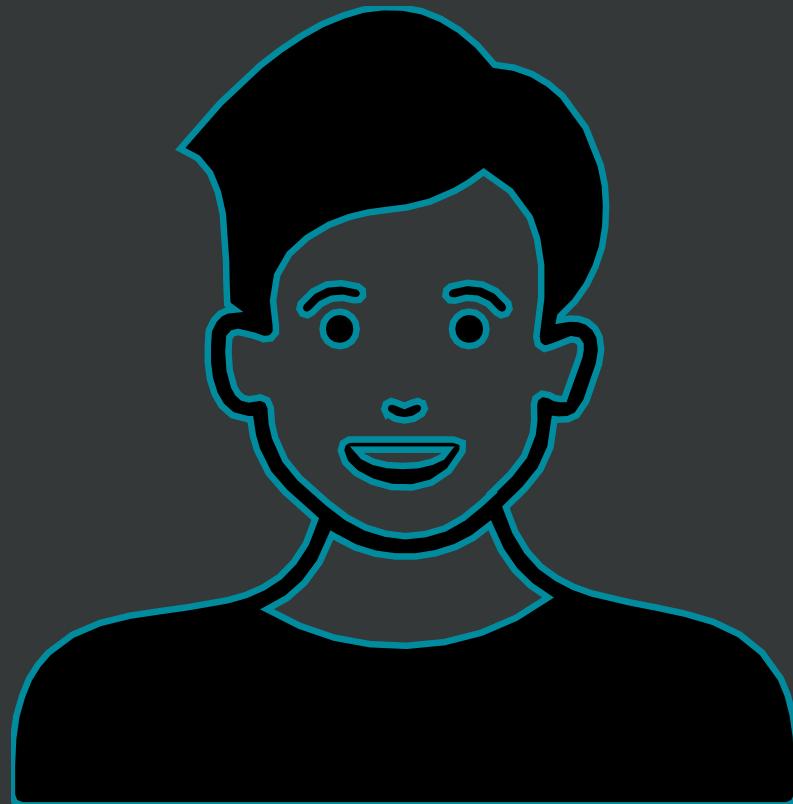
Find and Replace Console Runner

Too
bright!

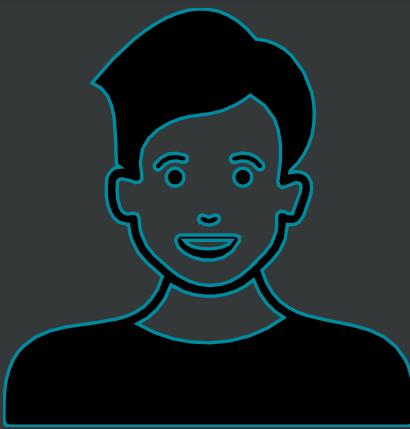


Much
Better!

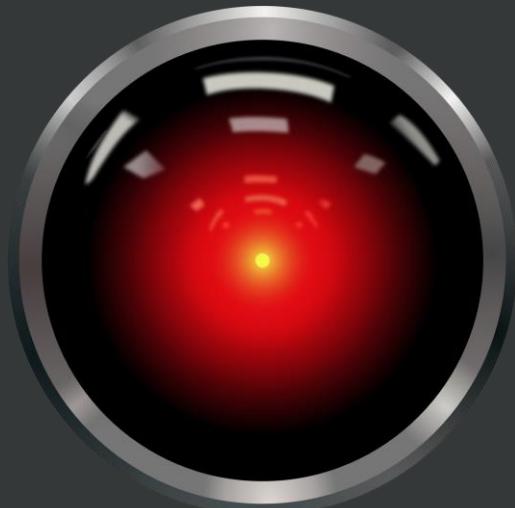
http://
y
p
e
x
r
t
a
n
s
f
e
r
p
a
t
o
l



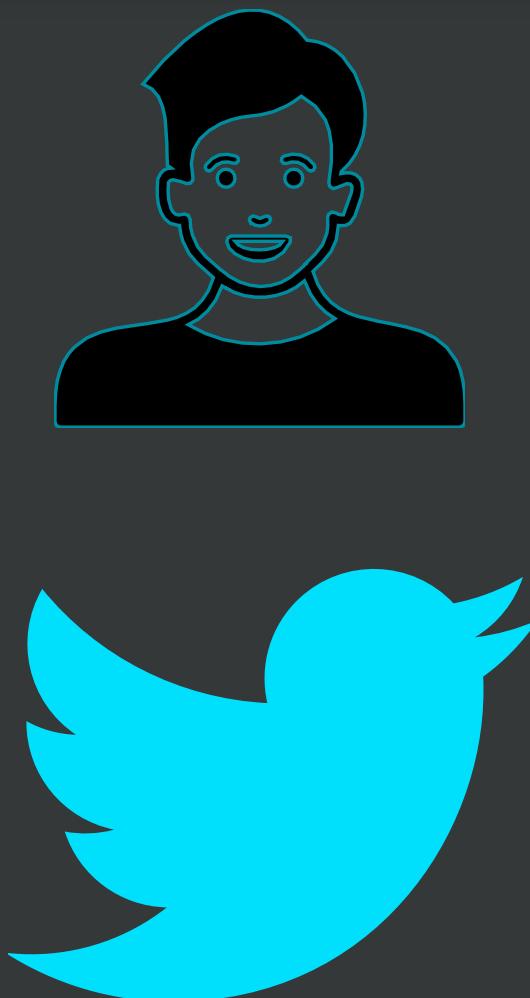
Hello, I want to
GET /tweets



GET /tweets



401 Unauthorized



GET /tweets

200
OK

A screenshot of a tweet from Elon Musk (@elonmusk) dated 7 Aug 2018. The tweet reads: "Am considering taking Tesla private at \$420. Funding secured." The tweet has 15,897 retweets and 87,589 likes. The interface shows standard Twitter interaction icons like reply, retweet, like, and message.

Elon Musk

@elonmusk

Follow

Am considering taking Tesla private at \$420. Funding secured.

12:48 PM - 7 Aug 2018

15,897 Retweets 87,589 Likes

f i T

6.3K 16K 88K

GET



GET

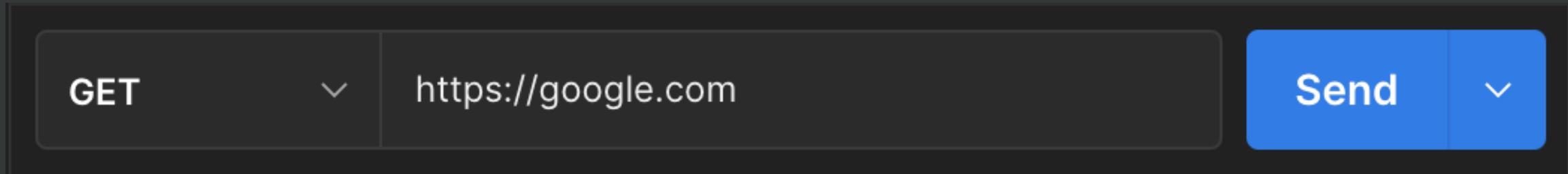
POST

PUT

PATCH

DELETE

HTTP Methods



The URL

Protocol Domain Path Query Parameters

The screenshot shows a Postman request configuration. At the top, there are four labels with blue arrows pointing to specific parts of the URL in the 'URL' field:

- Protocol**: Points to the 'https:' part of the URL.
- Domain**: Points to the 'httpbin.org' part of the URL.
- Path**: Points to the '/get' part of the URL.
- Query Parameters**: Points to the '?query=search for data&format=utf-8' part of the URL.

The URL field contains the full URL: `https://httpbin.org/get?query=search for data&format=utf-8`.

Below the URL field, there are several tabs:

- Params (highlighted with an orange underline)
- Authorization
- Headers (7)
- Body
- Pre-request Script
- Tests
- Settings

The 'Params' tab is active. Below it, under 'Query Params', is a table:

	KEY	VALUE
<input checked="" type="checkbox"/>	query	search for data
<input checked="" type="checkbox"/>	format	utf-8
	Key	Value

Params ● Authorization ● Headers (8) Body Pre-request Script Tests Settings

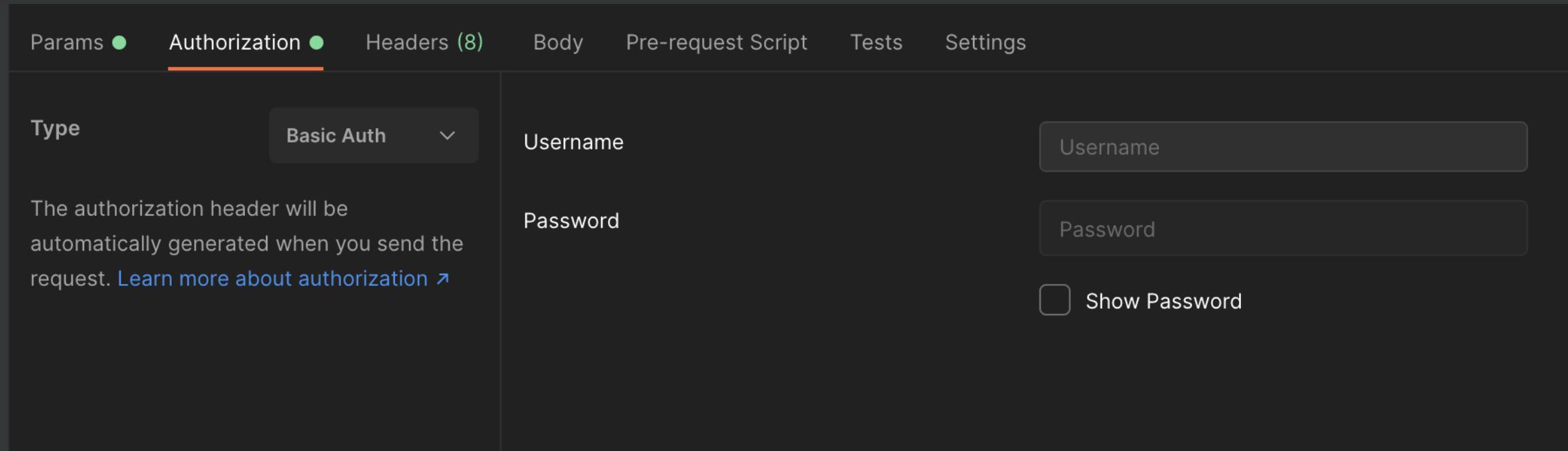
Type Basic Auth ▾

The authorization header will be automatically generated when you send the request. [Learn more about authorization ↗](#)

Username Username

Password Password

Show Password



Authorization



Params ● Authorization Headers (7) Body Pre-request Script Tests Settings

Headers ⚡ Hide auto-generated headers

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Cache-Control ⓘ	no-cache	
<input checked="" type="checkbox"/> Postman-Token ⓘ	<calculated when request is sent>	
<input checked="" type="checkbox"/> Host ⓘ	<calculated when request is sent>	
<input checked="" type="checkbox"/> User-Agent ⓘ	PostmanRuntime/7.28.0	
<input checked="" type="checkbox"/> Accept ⓘ	*/*	
<input checked="" type="checkbox"/> Accept-Encoding ⓘ	gzip, deflate, br	
<input checked="" type="checkbox"/> Connection ⓘ	keep-alive	

Key Value Description

Params ● Authorization Headers (9) Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON ▾

```
1  {
2    "id": 1,
3    "name": "Caleb",
4    "addresses": [
5      "ICG2",
6      "ICL7",
7      "INN.1024.1.8",
8      "WFH"
9    ]
10 }
11 |
```



Send



Making Requests

The screenshot shows the Postman application interface. At the top, there's a header bar with 'GET' selected, a URL field containing 'https://httpbin.org/get?query=search for data&format=utf-8', and a 'Send' button. Below the header are tabs for 'Params', 'Authorization', 'Headers (9)', 'Body', 'Pre-request Script', 'Tests', and 'Settings'. The 'Params' tab is active, showing two entries: 'query' with value 'search for data' and 'format' with value 'utf-8'. The 'Body' tab is also active, displaying the raw JSON response from the server. The response body is:

```
1  [
2   "args": {
3     "format": "utf-8",
4     "query": "search for data"
5   },
6   "headers": {
7     "Accept": "*/*",
8     "Accept-Encoding": "gzip, deflate, br",
9     "Cache-Control": "no-cache",
10    "Host": "httpbin.org",
11    "Postman-Token": "c2ce559d-3163-4ebf-a354-dcc3c2250bf5",
12    "User-Agent": "PostmanRuntime/7.28.0",
13    "X-Amzn-Trace-Id": "Root=1-60db57d0-7453028a5e5747d40e09c9d9"
14  },
15  "origin": "159.140.252.97",
16  "url": "https://httpbin.org/get?query=search for data&format=utf-8"
17 ]
```

The status bar at the bottom indicates a 200 OK status, 68 ms time, and 730 B size. There are also 'Save Response' and search/filter icons.

Responses

Response Status Codes

100 Continue
400 Bad Request
302 Found
204 No Content
504 Gateway Timeout
404 Not Found
201 Created
503 Service Unavailable
401 Unauthorized
500 Internal Server Error
200 OK

Tabs

The screenshot shows the Postman application interface. At the top, there is a navigation bar with links: History, Basics, Requests, Collections (highlighted in orange), Variables, Testing, Advanced, and a Help link. Below the navigation bar, the word "Tabs" is displayed in large white letters.

The main area of the interface shows a tab bar with two tabs: "GET GET with Params" and "POST POST with body". The "POST POST with body" tab is currently active, indicated by an orange underline. To the right of the tabs are buttons for "Save", "More", and "Edit".

The active tab displays a "Postman Presentation / POST with body" section. It includes a method dropdown set to "POST", a URL input field containing "https://httpbin.org/post", and a "Send" button. Below these are sections for "Params", "Authorization", "Headers (9)", "Body" (which is currently selected, indicated by an orange underline), "Pre-request Script", "Tests", and "Settings".

The "Body" section shows various serialization options: "none", "form-data", "x-www-form-urlencoded", "raw" (selected, indicated by an orange circle), "binary", "GraphQL", and "JSON" (selected, indicated by an orange underline). To the right of the "Body" section are buttons for "Cookies" and "Beautify".

The "Body" input area contains the following JSON payload:

```
1 {  
2   "id": 1,  
3   "name": "Caleb"  
4 }
```

Tabs

The screenshot shows the Postman application interface. At the top, there is a navigation bar with several tabs: 'GET G.', 'POST F X' (which is selected), 'GET U.', '→', '+', '...', and 'Prod Zone 2'. Below the navigation bar, the title 'Postman Presentation / POST with body' is displayed. The main workspace shows a 'POST' request to 'https://httpbin.org/post'. The 'Body' tab is selected, showing a JSON payload:

```
1 {
2   "id": 1,
3   "name": "Caleb"
4 }
```

Below the body, there are tabs for 'Params', 'Authorization', 'Headers (9)', 'Pre-request Script', 'Tests', 'Settings', 'Cookies', and 'Beautify'. The 'Body' tab has a green dot icon next to it. The 'JSON' dropdown menu is open, showing options: 'none', 'form-data', 'x-www-form-urlencoded', 'raw', 'binary', 'GraphQL', and 'JSON' (which is selected).

History

Scratch Pad New Import ← GET G. POST X GET U. GET U. GET → + ... Prod Zone 2

Collections APIs Environments Mock Servers Monitors History

Postman Presentation / **POST with body** Save ...

POST https://httpbin.org/post Send

Params Auth Headers (9) Body ● Pre-req. Tests Settings Cookies Beautify

raw JSON

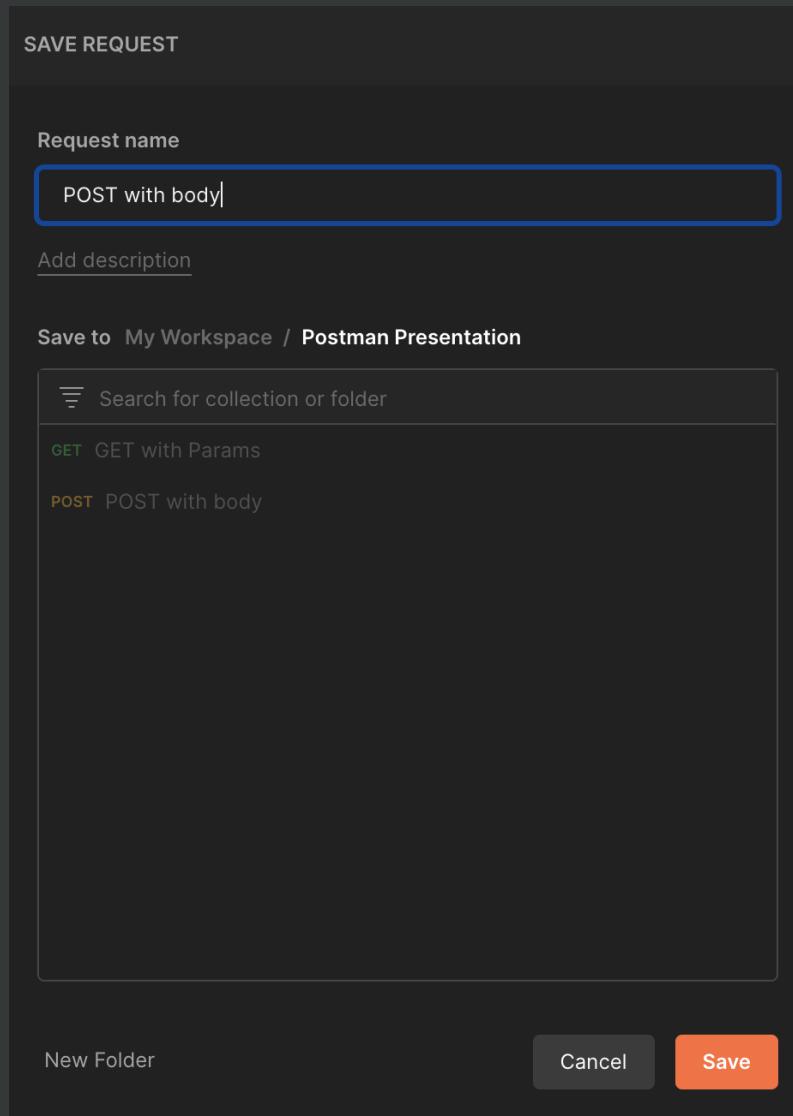
```
1 {  
2   "id": 1,  
3   "name": "Caleb"  
4 }
```

Body 200 OK 324 ms 852 B Save Response

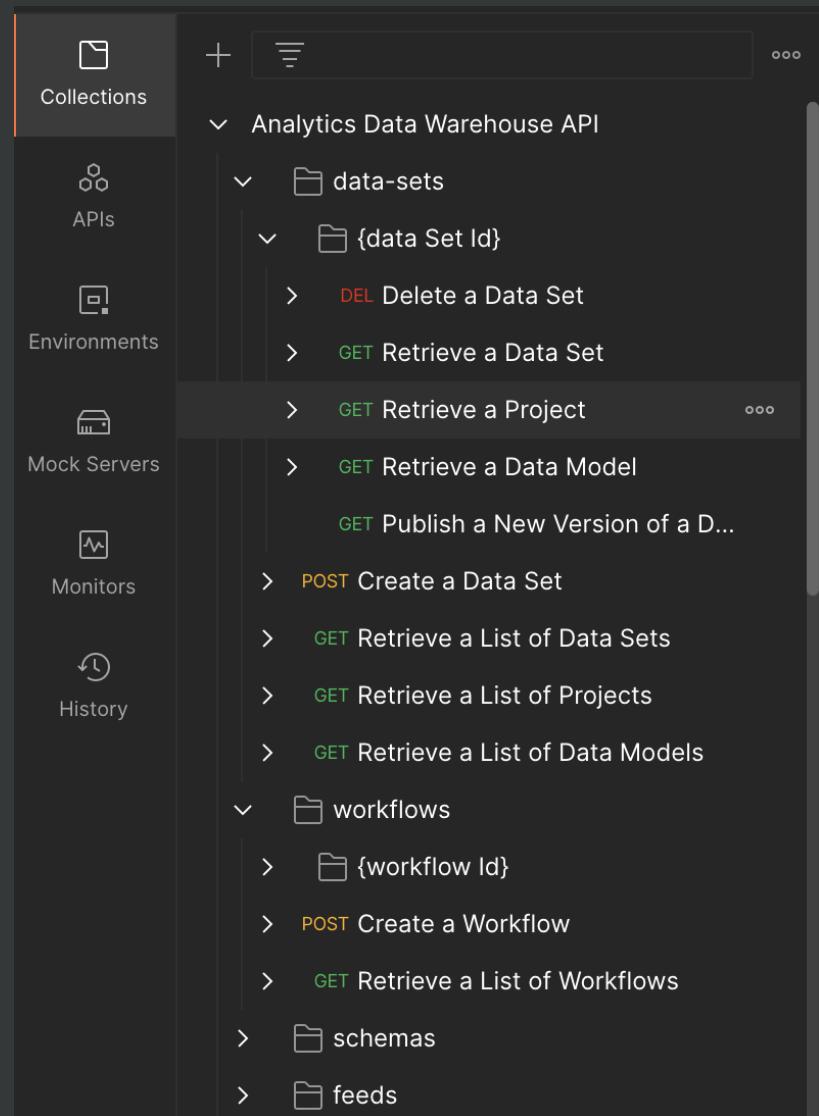
Pretty Raw Preview Visualize JSON

Q

The screenshot shows the Postman application's interface. The left sidebar has tabs for Scratch Pad, Collections, APIs, Environments, Mock Servers, Monitors, and History, with History currently selected. The main workspace shows a POST request to https://httpbin.org/post with a JSON body containing an object with id: 1 and name: Caleb. The response status is 200 OK with a time of 324 ms and a size of 852 B.



Save
a
Request
To
a
Collection



Postman

Home Workspaces Reports Explore

Scratch Pad New Import

Collections + ⌂ ...

APIs > Analytics Data Warehouse API

Environments > Business Intelligence Public API

Dev

Postman Presentation

Postman Presentation

Authorization Watch 0 Fork Run Save Share ...

This authorization method will be used for every request in this collection. You can override this by specifying one in the request.

Type OAuth 1.0

The authorization data will be automatically generated when you send the request.
[Learn more about authorization](#)

Add auth data to Request Body / Request URL

Postman will automatically choose between body and URL based on the request method.

Signature Method HMAC-SHA1

Consumer Key

Consumer Secret

Access Token

Token Secret

> ADVANCED

Find and Replace Console

Runner



The screenshot shows the Postman interface with two modal dialogs open:

- EXPORT COLLECTION** dialog (left):
 - Header: "Postman Presentation will be exported as a JSON file. Export as:"
 - Options:
 - Collection v2
 - Collection v2.1 (recommended)
 - Text area (partially visible): "Postman Presentation will be exported as a JSON file. Export as: Collection v2.1 (recommended) ..."
 - Buttons: "Cancel" and "Export" (orange button)

A portion of the JSON collection file is visible in the background.

Import dialog (right):
 - Header: "Import" (with close button)
 - Tab bar: File (selected), Folder, Link, Raw text, Code repository
 - Text area: "Drag and drop Postman data or any of the formats below"
 - Format buttons: OpenAPI, RAML, GraphQL, cURL, WADL
 - Text: "OR"
 - Button: "Upload Files"

GET ▼ https://cernerdemo.api.us.staginghealtheintent.com/analytics-data-warehouse/v1/data-sets/e3f2c366-d9dd-11eb-b8bc-0242ac13000:...

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	workflow_id	09cf3452-d9de-11eb-b8bc-0242ac130003	
	Key	Value	Description

This is a lot to
copy and paste

The screenshot shows the Postman interface with a modal dialog for saving a variable. The main window displays a GET request to a URL containing a placeholder for a variable. Below the URL, there are tabs for Params, Authorization, Headers (7), Body, Pre-request Script, Tests, and Settings. The Params tab is selected. The modal dialog has fields for Name (public_api), Value (https://cernerdemo.api.us.stag...), and Scope (Select A Scope). The Scope dropdown is open, showing three options: Global (selected), Environment: Staging, and Collection.

Saving a Variable

Environments

The screenshot shows a variable management interface with the following details:

Environment: Dev

Variables List:

VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ
consumer_key	4c95a0b0-1053-4a59-8e16-f71254e40667	4c95a0b0-1053-4a59-8e16-f71254e40667 Redacted
consumer_secret		ConsumerKey=4c95a0b0-1053-4a59-8e16-f71254e40667&ExpiresOn...
access_token		Redacted
access_token_secret		Redacted
tableau_uri	http://tableaugateway.us-zone0.devhealtheintent.net/a...	http://tableaugateway.us-zone0.devhealtheintent.net/api/3.2
tableau_username		admin
tableau_password		Redacted
tableau_auth_token		Redacted
tableau_site_id	ab629b87-273a-4555-b3b4-aa3b522cfbfa	ab629b87-273a-4555-b3b4-aa3b522cfbfa
tableau_user_id	7b4414ee-15be-432d-bbce-2dcd8e18e926	7b4414ee-15be-432d-bbce-2dcd8e18e926
data_warehouse_public_api_uri	https://app.api.us.devhealtheintent.com/analytics-data...	https://app.api.us.devhealtheintent.com/analytics-data-warehouse/v1
oauth_access_token_uri	https://api.devcernercare.com/oauth/access	https://api.devcernercare.com/oauth/access
bearer_token	eyJhbGciOiJub25lIn0.eyJpc3MiOiJodHRwczpcL1wvYXV0aG9yaXphdG...	eyJhbGciOiJub25lIn0.eyJpc3MiOiJodHRwczpcL1wvYXV0aG9yaXphdG...
public_api_base_uri	https://app.api.us.devhealtheintent.com	https://app.api.us.devhealtheintent.com
project_public_api_uri	https://app.api.us.devhealtheintent.com/healtheanalyti...	https://app.api.us.devhealtheintent.com/healtheanalytics-application/v1

Info Bar: Use variables to reuse values in different places. Work with the current value of a variable to prevent sharing sensitive values with your team. [Learn more about variable values](#)

The sidebar displays the following environments:

- Dev
- No Environment
- Dev
- Prod US (zone 0)
- Prod Zone 1
- Prod Zone 2
- Staging

GET



https://cernerdemo.api.us.healtheintent.com/core/v1

GET

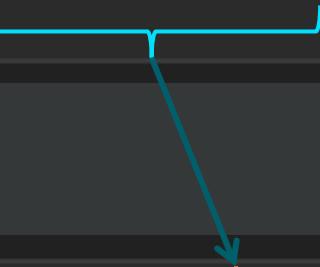


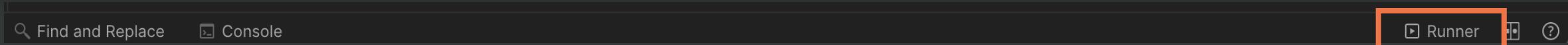
https://{{tenant_mnemonic}}.api.us.healtheintent.com/core/v1

GET



{{public_api}}/core/v1





Testing with the Runner

The screenshot shows the "Runner" configuration screen. At the top left, it says "RUN ORDER" with buttons for "Deselect All", "Select All", and "Reset". To the right, there's a dropdown menu set to "Prod Zone 2". The main area contains several configuration options:

- "Iterations": A text input field containing the value "1".
- "Delay": A text input field containing the value "0" followed by "ms".
- "Data": A button labeled "Select File".
- A group of checkboxes:
 - Save responses (i)
 - Keep variable values (i)
 - Run collection without using stored cookies
 - Save cookies after collection run (i)
- A large blue "Start Run" button at the bottom right.

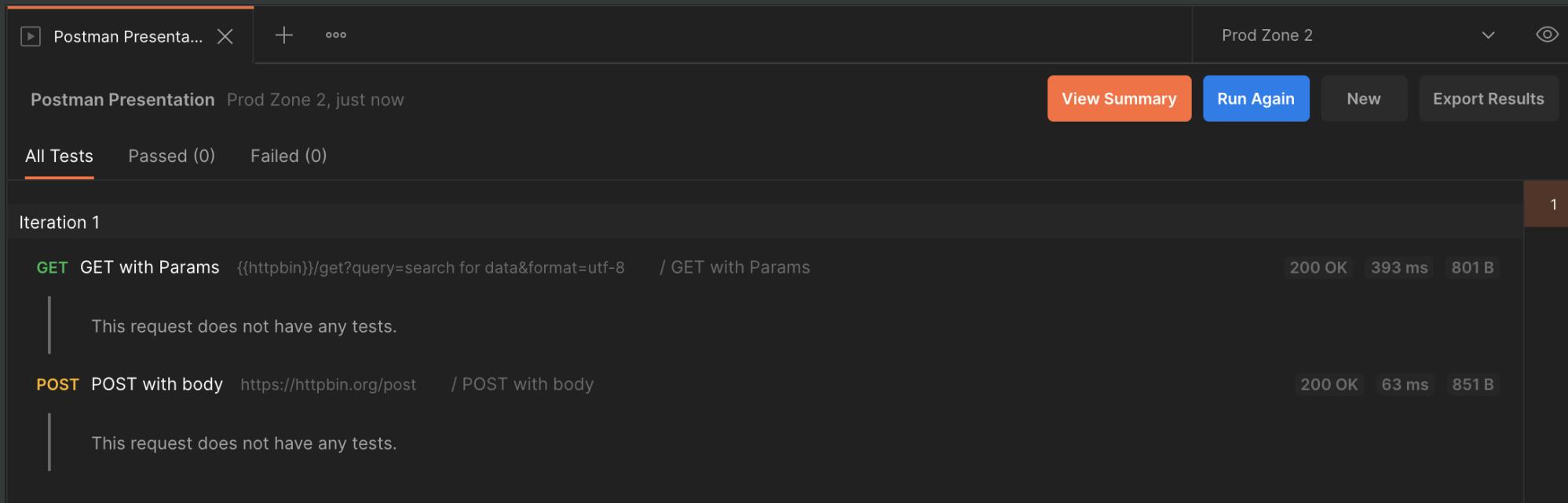
On the left side of the main panel, there's a placeholder message: "Drag a collection or folder from your sidebar to get started". At the bottom left, another message says: "You can find your previous collection runs in [history](#) or [import](#) past runs."

Add a Collection...

The screenshot shows the Postman Runner interface with the following configuration:

- RUN ORDER:** Contains two selected items:
 - GET GET with Params
 - POST POST with body
- Deselect All | Select All | Reset** buttons.
- Iterations:** Set to 1.
- Delay:** Set to 0 ms.
- Data:** A "Select File" button.
- Checkboxes (unchecked):**
 - Save responses
 - Run collection without using stored cookies
- Checkboxes (checked):**
 - Keep variable values
 - Save cookies after collection run
- Run Postman Presentation** button.

...and press RUN



The screenshot shows the Postman application interface. At the top, there's a navigation bar with links for History, Basics, Requests, Collections, Variables, Testing, Advanced, and a search bar. Below the navigation is a large title "...and press RUN". The main area is a dark-themed window titled "Postman Presentation" under "Prod Zone 2". It shows a summary of the run: "All Tests Passed (0) Failed (0)". There are buttons for "View Summary", "Run Again" (which is highlighted in blue), "New", and "Export Results". Below this, it says "Iteration 1". Under "Iteration 1", there are two requests listed: 1. A "GET" request to "{{httpbin}}/get?query=search for data&format=utf-8" with a status of "200 OK", response time "393 ms", and size "801 B". A note next to it says "This request does not have any tests." 2. A "POST" request to "https://httpbin.org/post" with a status of "200 OK", response time "63 ms", and size "851 B". A note next to it says "This request does not have any tests." The overall theme is minimalist with a focus on the results of the API testing.

There is no third step

```
~ $ npm i -g newman                                     Thu Jul 1 15:08:47 2021
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

changed 127 packages, and audited 128 packages in 6s

5 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
~ |
```

The logo for Aperture Laboratories, featuring a stylized camera lens on the left and the company name "APERTURE LABORATORIES" in large, bold, sans-serif capital letters on the right.

6.7s < Thu Jul 1 15:08:58 2021



```
● History ● Basics ● Requests ● Collections ● Variables ● Testing ● Advanced ●

newman run ./Postman\ Presentation.postman_collection.json
newman
Postman Presentation
→ GET with Params
GET https://httpbin.org/get?query=search for data&format=utf-8 [200 OK, 801B, 433ms]
→ POST with body
POST https://httpbin.org/post [200 OK, 851B, 73ms]



|                    | executed | failed |
|--------------------|----------|--------|
| iterations         | 1        | 0      |
| requests           | 2        | 0      |
| test-scripts       | 2        | 0      |
| prerequest-scripts | 2        | 0      |
| assertions         | 0        | 0      |


total run duration: 574ms
total data received: 1.16KB (approx)
average response time: 253ms [min: 73ms, max: 433ms, s.d.: 180ms]

Thu Jul 1 15:15:03 2021
1897ms < Thu Jul 1 15:15:08 2021
```

Params ● Authorization Headers (9) Body ● Pre-request Script Tests Settings Cookies

1

Pre-request scripts are written in JavaScript, and are run before the request is sent. >
[Learn more about pre-request scripts](#)

SNIPPETS

[Get an environment variable](#)
[Get a global variable](#)
[Get a variable](#)
[Get a collection variable](#)
[Set an environment variable](#)
[Set a global variable](#)
[Set a collection variable](#)
[Clear an environment variable](#)
[Clear a global variable](#)
[Clear a collection variable](#)

• History • Basics • Requests • Collections • Variables • Testing • Advanced •

Analytics Project Service Integration Tests / Projects / Create a Project

POST http://localhost:3000/projects

Params Authorization Headers (10) Body ● Pre-request Script **Tests** ● Settings

```
1 pm.test("Status code is 201 (created)", function () {  
2     pm.response.to.have.status(201);  
3 };  
4  
5 pm.test("It uses the given name", function () {  
6     var jsonData = pm.response.json();  
7     pm.expect(jsonData.name).to.eql('Test Project');  
8 };  
9  
10 pm.collectionVariables.set("project_id", pm.response.json().id);
```

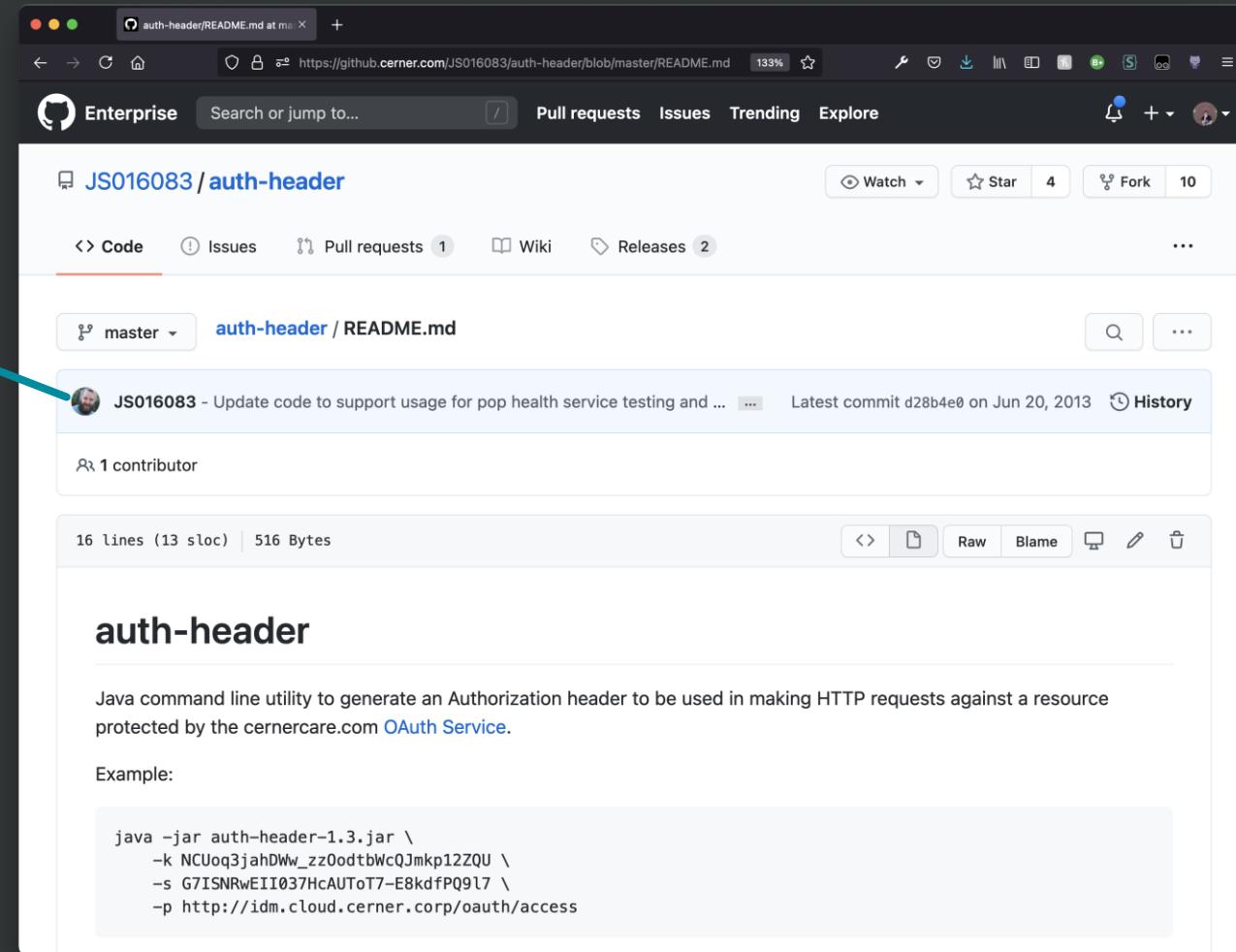
All Tests Passed (11) Failed (1)

Iteration 1

Method	URL	Assertion	Status	Time	Size
GET	Basic Health http://localhost:3000/meta/health	/ Set Up / Basic Health	200 OK	2107 ms	3.669 KB
		Pass Status code is 200			
		Pass Body contains 'is available'			
GET	Generate OAuth https://api.devcernercare.com/oauth/access	/ Set Up / Generate OAuth	200 OK	233 ms	1.129 KB
		This request does not have any tests.			
GET	Projects Index http://localhost:3000/projects?clientId=test	/ Projects / Projects Index	200 OK	601 ms	1.091 KB
		Pass Status code is 200			
		Fail There are no projects Assertion: expected [Array(1)] to deeply equal []			
POST	Create a Project http://localhost:3000/projects	/ Projects / Create a Project	201 Created	129 ms	970 B
		Pass Status code is 201 (created)			
		Pass It uses the given name			
GET	View that Project http://localhost:3000/projects/{{project_id}}?clientId=test	/ Projects / View that Project	200 OK	91 ms	886 B
		Pass Status code is 200			
		Pass It has the same name			
PUT	Update that Project http://localhost:3000/projects/{{project_id}}?clientId=test&name>New Name	/ Projects / Update that Project	204 No Content	66 ms	447 B
		Pass Status code is 204 (no content)			

Automate OAuth

Thanks
John!



Automate OAuth

❖ oauthgen.rb

```
1 #!/usr/bin/env ruby
2 # Before you run this script the first time, you'll need to run-
3 # gem install oauth clipboard docopt ffi-
4 -
5 begin-
6   require 'oauth'-
7   require 'oauth/consumer'-
8   require 'oauth/signature/plaintext'-
9   require 'clipboard'-
10  require 'docopt'-
11 rescue LoadError-
12   puts 'Please run `gem install oauth clipboard docopt ffi` first'-
13   exit-
14 end-
15 -
16 doc = <<DOCOPT-
17 Oauth Header Generator-
18 Usage:-
19 #__FILE__ (dev|staging|prod) [-n] [-o]-
20 #__FILE__ -h | --help-
21 #__FILE__ -v | --version-
22 -
23 Options:-
24 -n --no-copy      Don't copy the header to the clipboard-
25 -o --output       Output the header to the screen-
26 -h --help         Show this message-
27 -v --version      Show the version-
28 DOCOPT-
```

Thanks
Me!

Automate OAuth

Postman

Home Workspaces Reports Explore Search Postman Sign In Create Account

Prod Services / Populate Oauth (Run this first!)

GET {{oauth_access_token_uri}}

Params Authorization Headers (8) Body Pre-request Script Tests Settings

```
1 // Add this in a Postman test for the oauth GET call
2 var regex = /oauth_token=(.*)&oauth_token_secret=(^&*)/g;
3 var decoded = decodeURIComponent(responseBody);
4 var match = regex.exec(decoded);
5
6 var accessToken = match[1];
7 var tokenSecret = match[2];
8
9 pm.environment.set("access_token", accessToken);
10 pm.environment.set("access_token_secret", tokenSecret);
11
```

Cookies

Test scripts are written in JavaScript, and are run after the response is received. Learn more about tests scripts

SNIPPETS

Get an environment variable
Get a global variable
Get a variable
Get a collection variable
Set an environment variable
Set a global variable

Response

Hit send to get a response

<https://regexr.com/>

RegExr: Learn, Build, & Test Reg

Untitled Pattern Save New

Expression

/([A-Z])\w+/g

Text Tests NEW 29 matches (1.0ms)

RegExr was created by gskinner.com, and is proudly hosted by Media Temple.

Edit the Expression & Text to see matches. Roll over matches or the expression for details. PCRE & JavaScript flavors of RegEx are supported. Validate your expression with Tests mode.

Tools Replace List Details Explain

Roll-over elements below to highlight in the Expression above. Click to open in Reference.

(Capturing group #1. Groups multiple tokens together and creates a capture group for extracting a substring or using a backreference.

[Character set. Match any character in the set.

Automate OAuth

Params **Authorization** Headers (11) Body ● Pre-request Script ● Tests ● Settings

Type Inherit auth from parent

The authorization header will be automatically generated when you send the request. [Learn more about authorization ↗](#)

Prod Services [Watch](#) 0 [Fork](#)

Authorization Pre-request Script Tests Variables

This authorization method will be used for every request in this collection. You can override this by specifying one in the request.

Type OAuth 1.0

The authorization data will be automatically generated when you send the request. [Learn more about authorization ↗](#)

Add auth data to Request Headers

Signature Method ⓘ PLAINTEXT

Consumer Key ⓘ {{consumer_key}}

Consumer Secret ⓘ {{consumer_secret}}

Access Token {{access_token}}

Token Secret ⓘ {{access_token_secret}}

> ADVANCED

Each Request On the Collection

Params Authorization Headers (11) Body ● **Pre-request Script** ● Tests ● Settings

```
1 data_sets = pm.environment.get("data_sets");
2
3 if (!data_sets) {
4   data_sets = [
5     17659, 272199, 407395, 421638, 425324, 425326, 430838, 430841, 430847, 430853, 432536, 432539,
6     432542, 432548, 450655, 453488, 453501, 453508, 458809, 459214, 459265, 462118, 462541,
7     462576, 462630, 462771, 462798, 462801, 462802, 462808, 462996, 463470, 463656, 463787,
8     463904, 463991, 463994, 464024, 464077, 464105, 464111, 464114, 464246, 469394, 469491,
9     469660, 470488, 470651, 478788, 478983, 478992, 479017, 479203, 480079, 487384, 491226,
10    491793, 542484, 542487, 542491, 542499, 542502, 542505, 542511, 542513, 542527, 550521,
11    487384, 560104, 560104, 560104, 560104
12  ];
13}
14
15 pm.environment.set("id", data_sets.shift());
16 pm.environment.set("data_sets", data_sets);
```

Loops

Prod Services / Storefront / Update data sets by id

PUT {{storefront_service_uri}}/data_sets/{{id}}

Params Authorization Headers (11) Body ● Pre-request Script ● **Tests** ● Settings

```
1 data_sets = pm.environment.get("data_sets");
2
3 if (data_sets && data_sets.length > 0) {
4   postman.setNextRequest("Delete data set by id");
5 } else {
6   postman.setNextRequest(null);
7 }
8
9
```

When to *not* use Postman

When TO use Postman

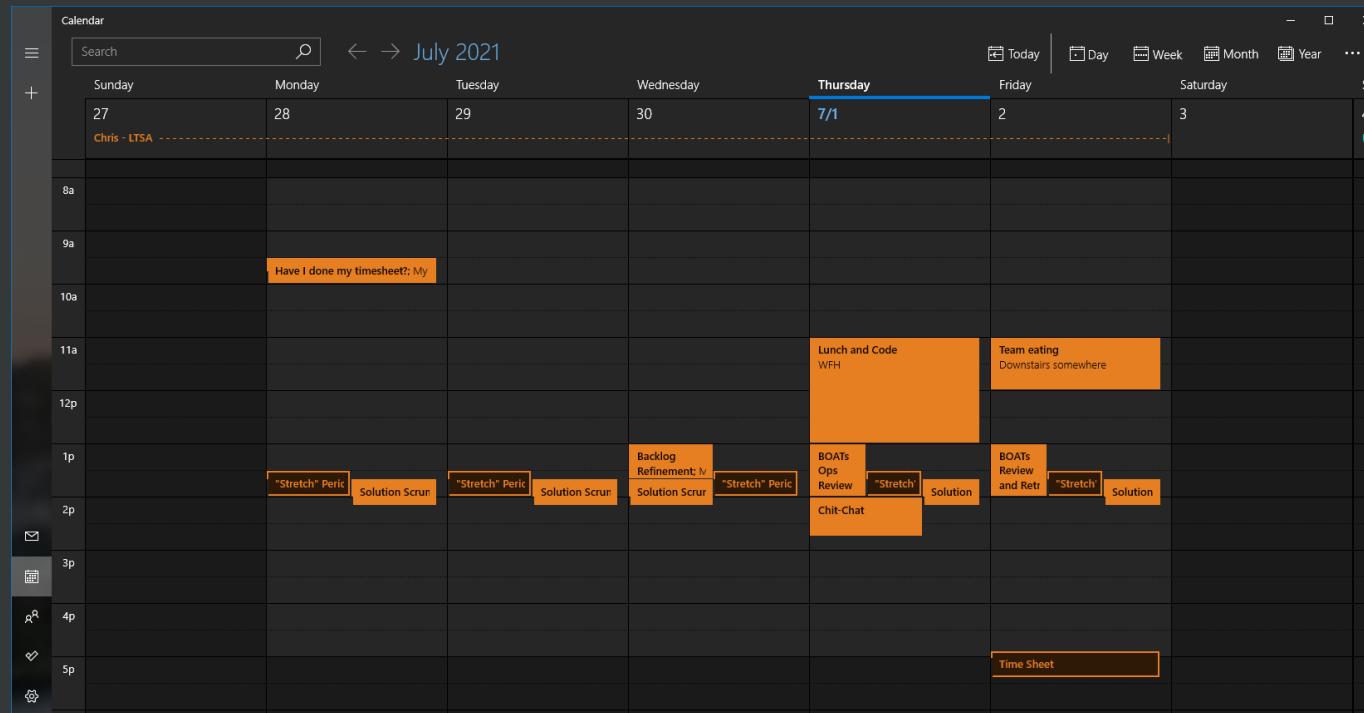
When to *not* use Postman



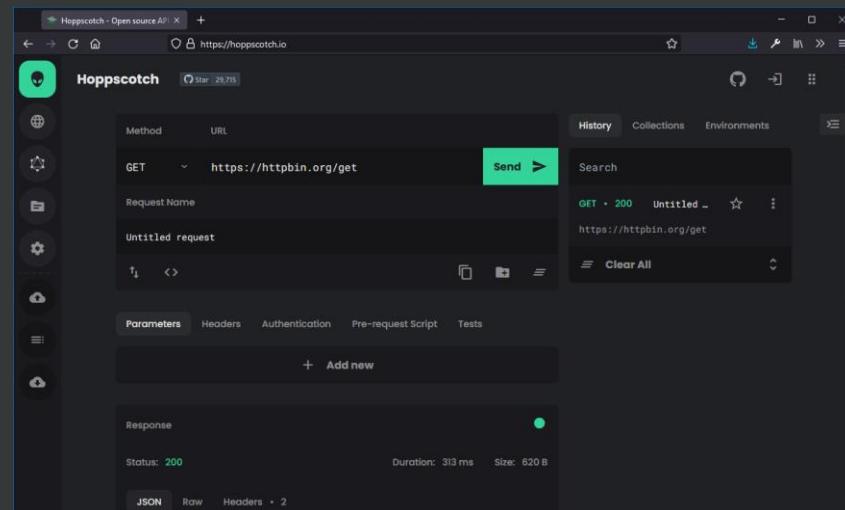
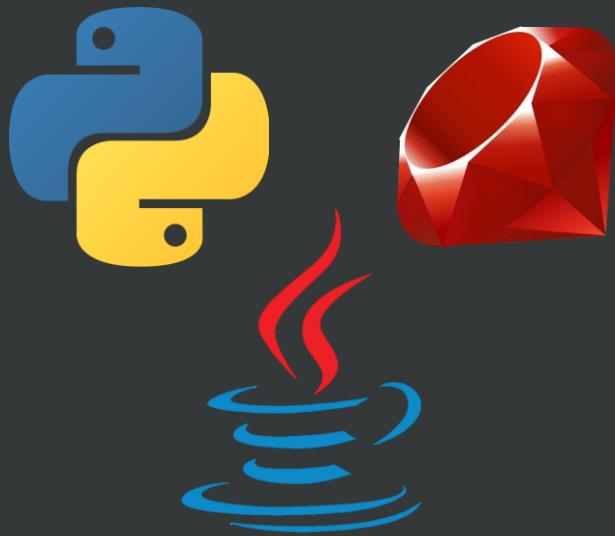
When to *not* use Postman



When to *not* use Postman



The Competition



```
[~] http PUT httpbin.org/put hello=world
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Origin: *
Connection: keep-alive
Content-Length: 452
Content-Type: application/json
Date: Thu, 06 Dec 2018 15:43:38 GMT
Server: gunicorn/19.9.0
Via: 1.1 vegur
{
    "args": {},
    "data": "{\"hello\": \"world\"}",
    "files": {},
    "form": {}
```

httpie.io

Fonts used in this presentation:

Orbitron

Courier Prime

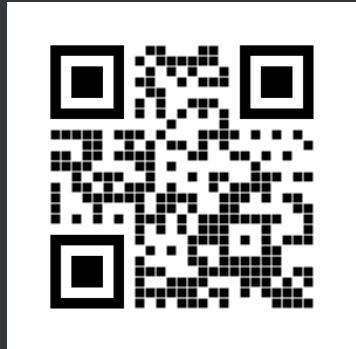
Ubuntu

Bradley Hand ITC

Merienda

Color Scheme:

- Background #343838 (cool gray)
- Foreground #FFFFFF (white)
- Accent 1 #005F6B
- Accent 2 #008C9E
- Accent 3 #00B4CC
- Accent 4 #00DFFC
- Accent 5 #E64823
- Accent 6 #EE7547 (Postman orange)



THANK YOU!

Find me on Teams

Slides can be found at

bit.ly/caleb-on-postman