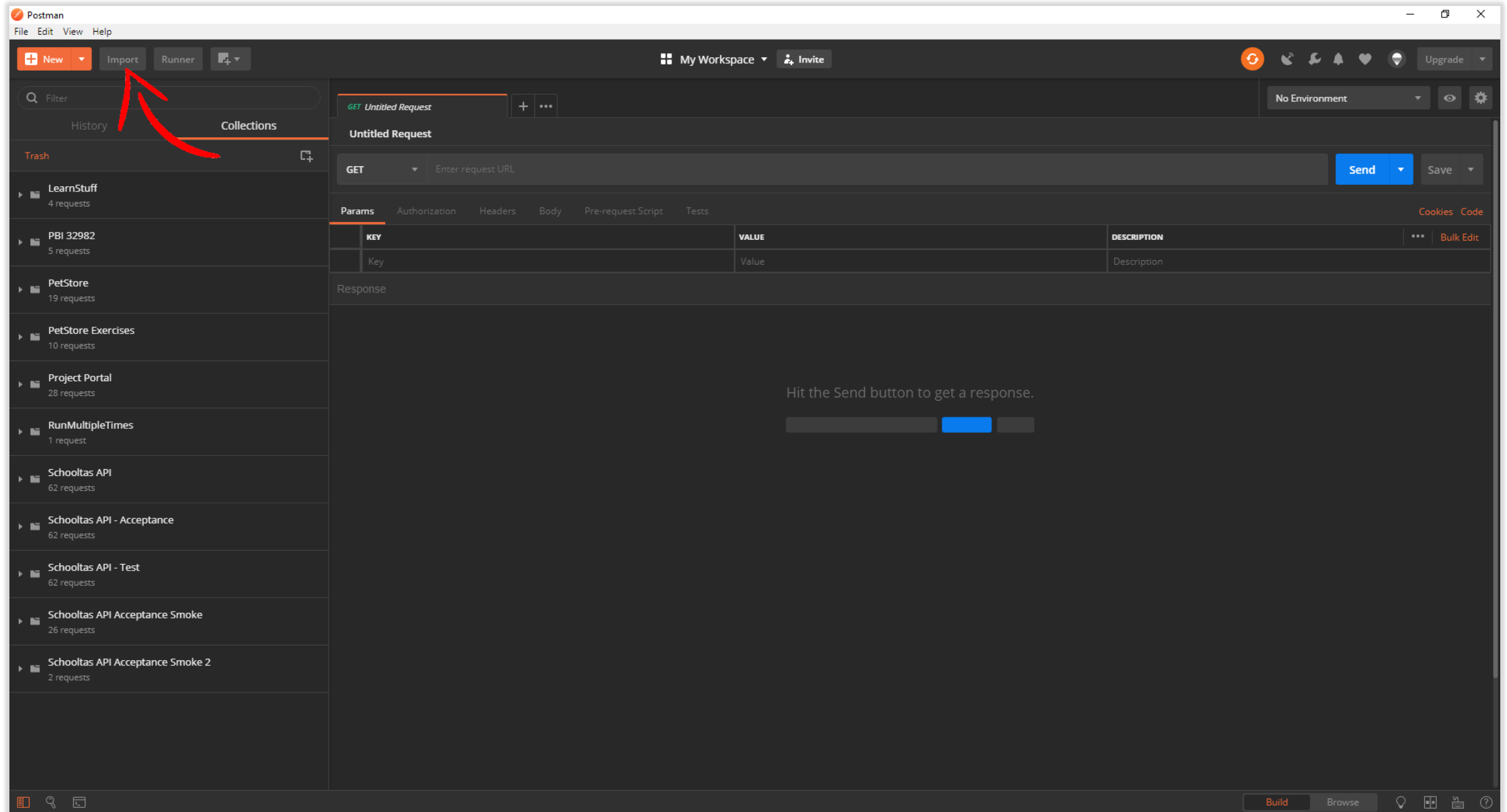
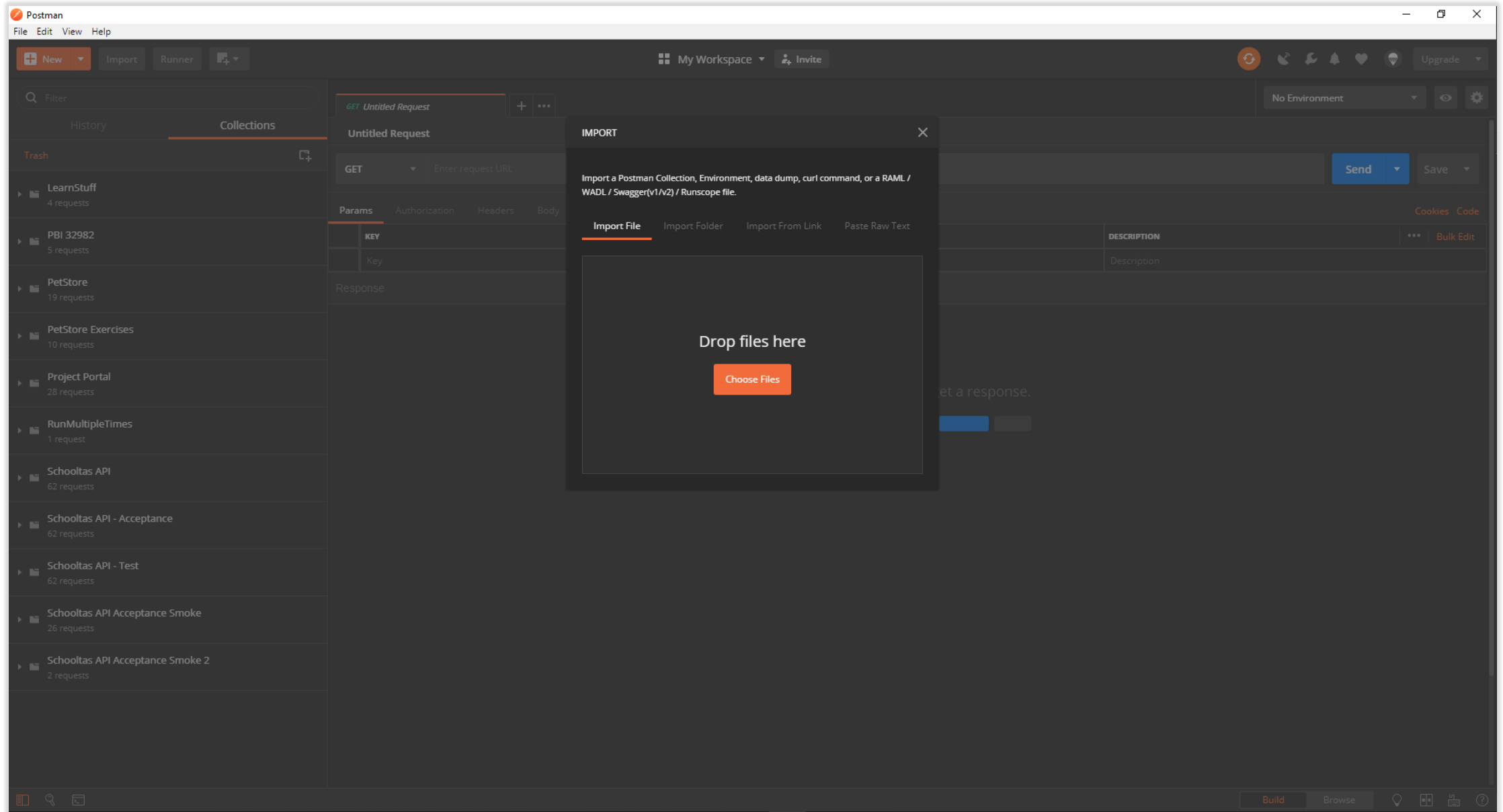


# Postman Semi-Automated Testing for Rest API

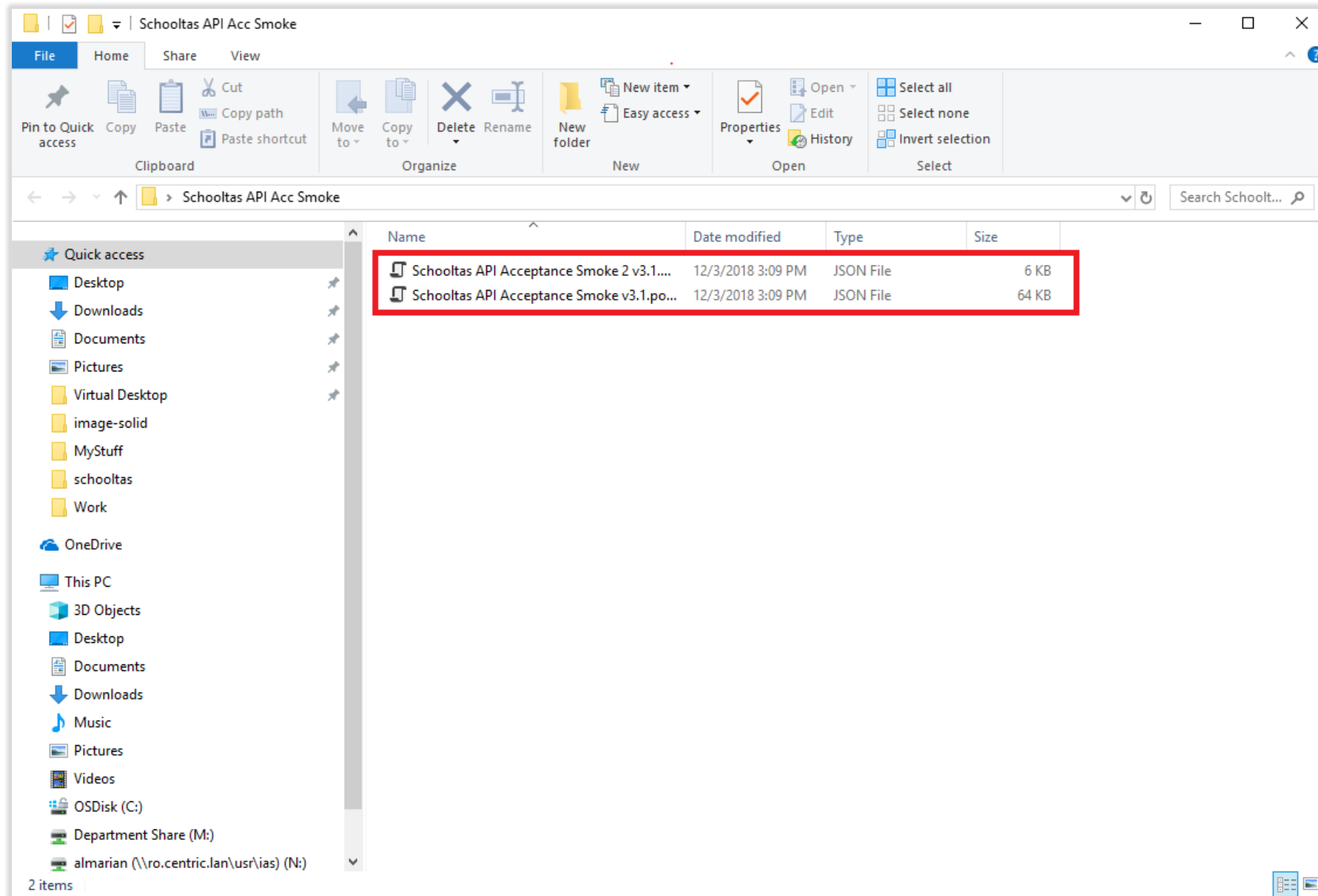
# Import the Postman Collection



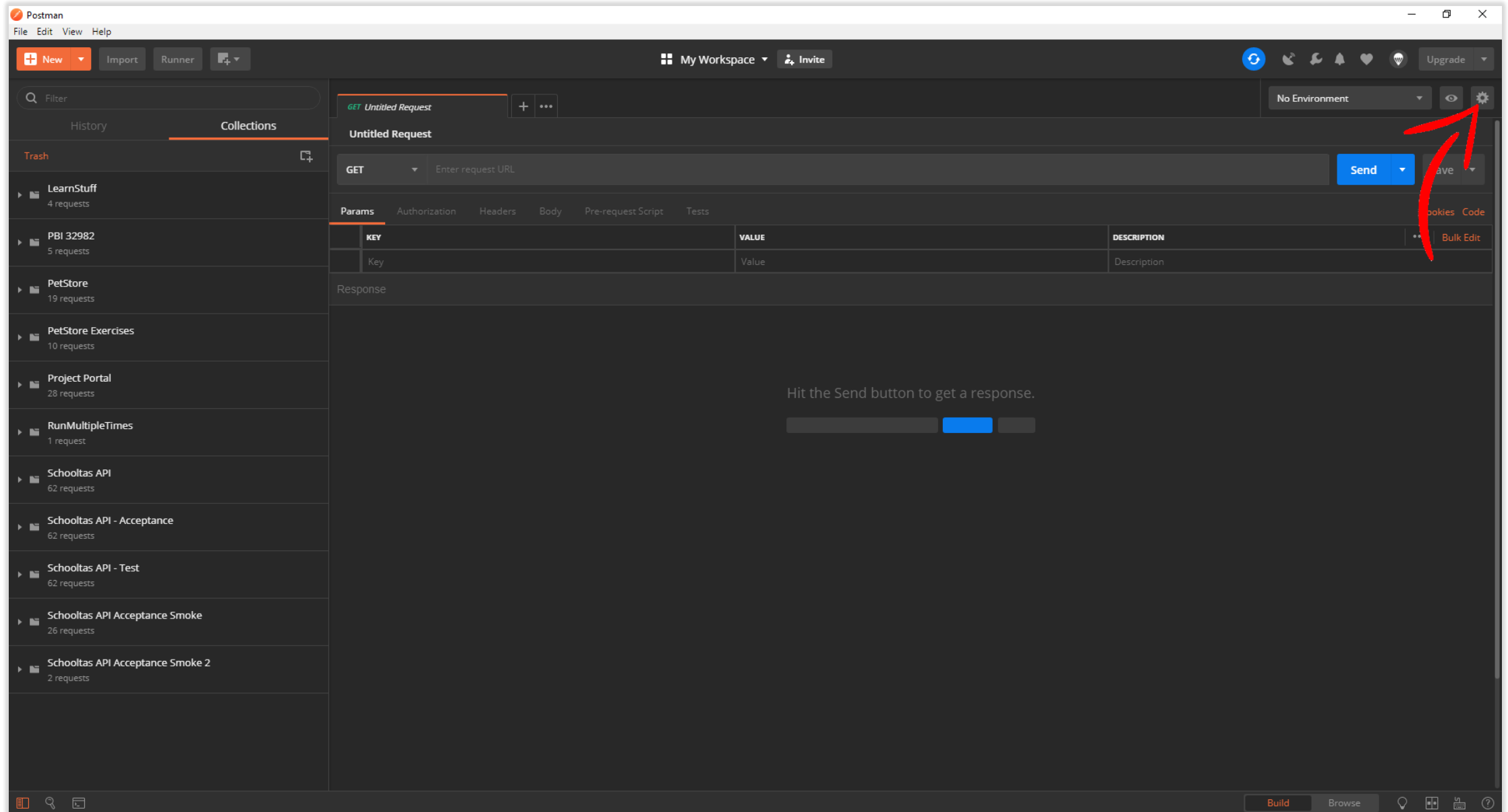
# Drag and Drop or Choose from files the collection



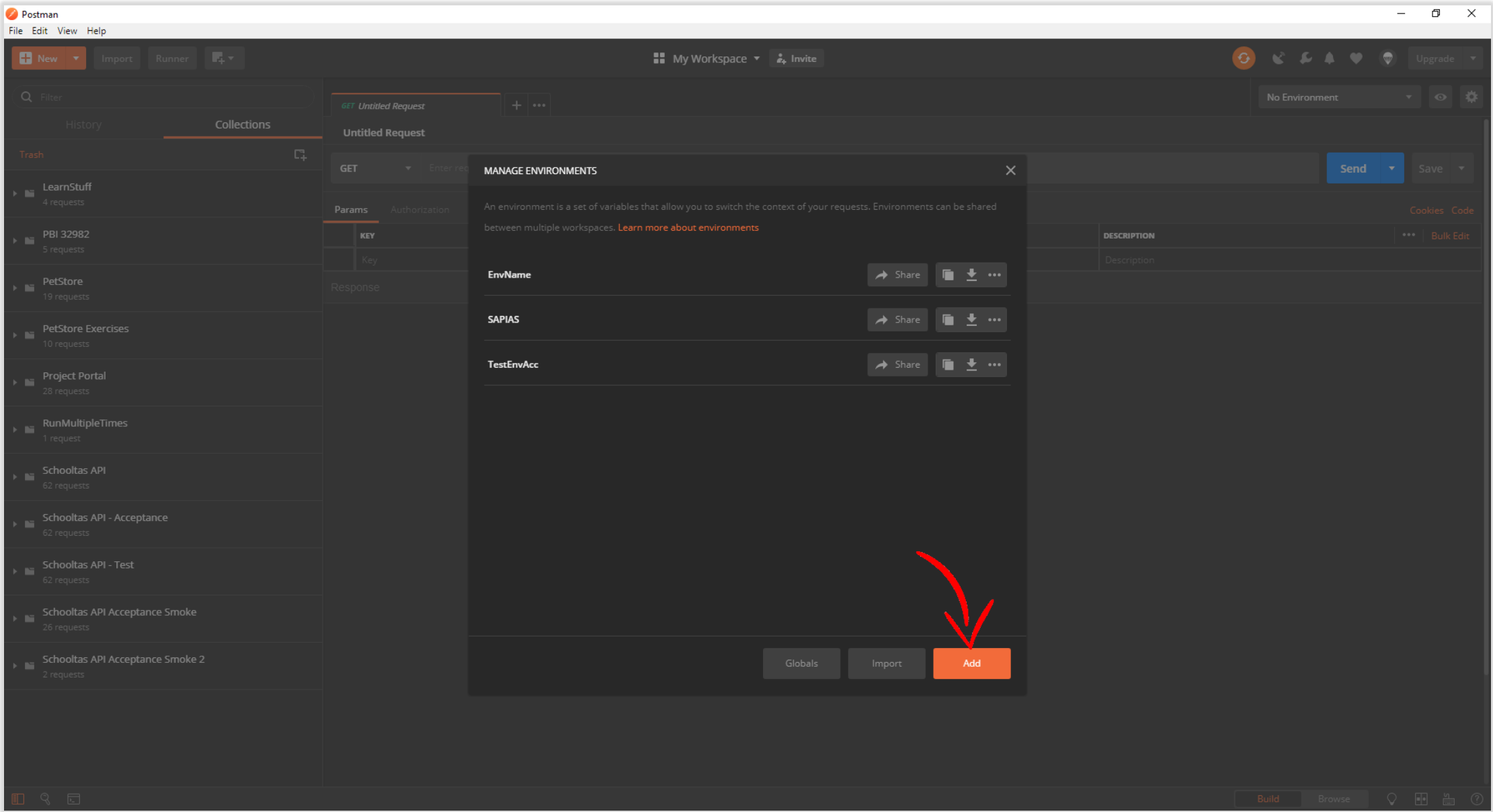
# From the archive import both collections



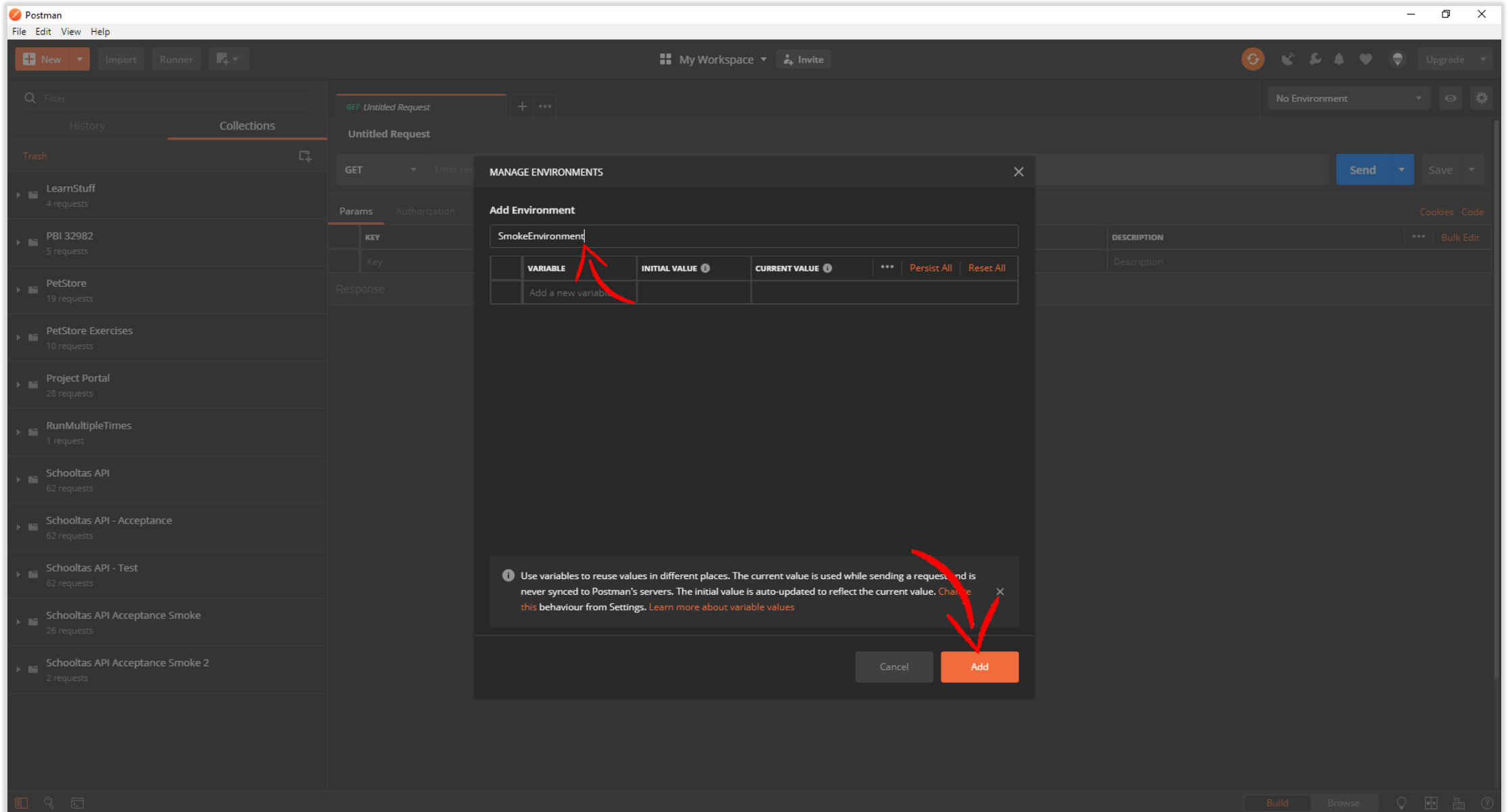
# Open the Manage Environments window



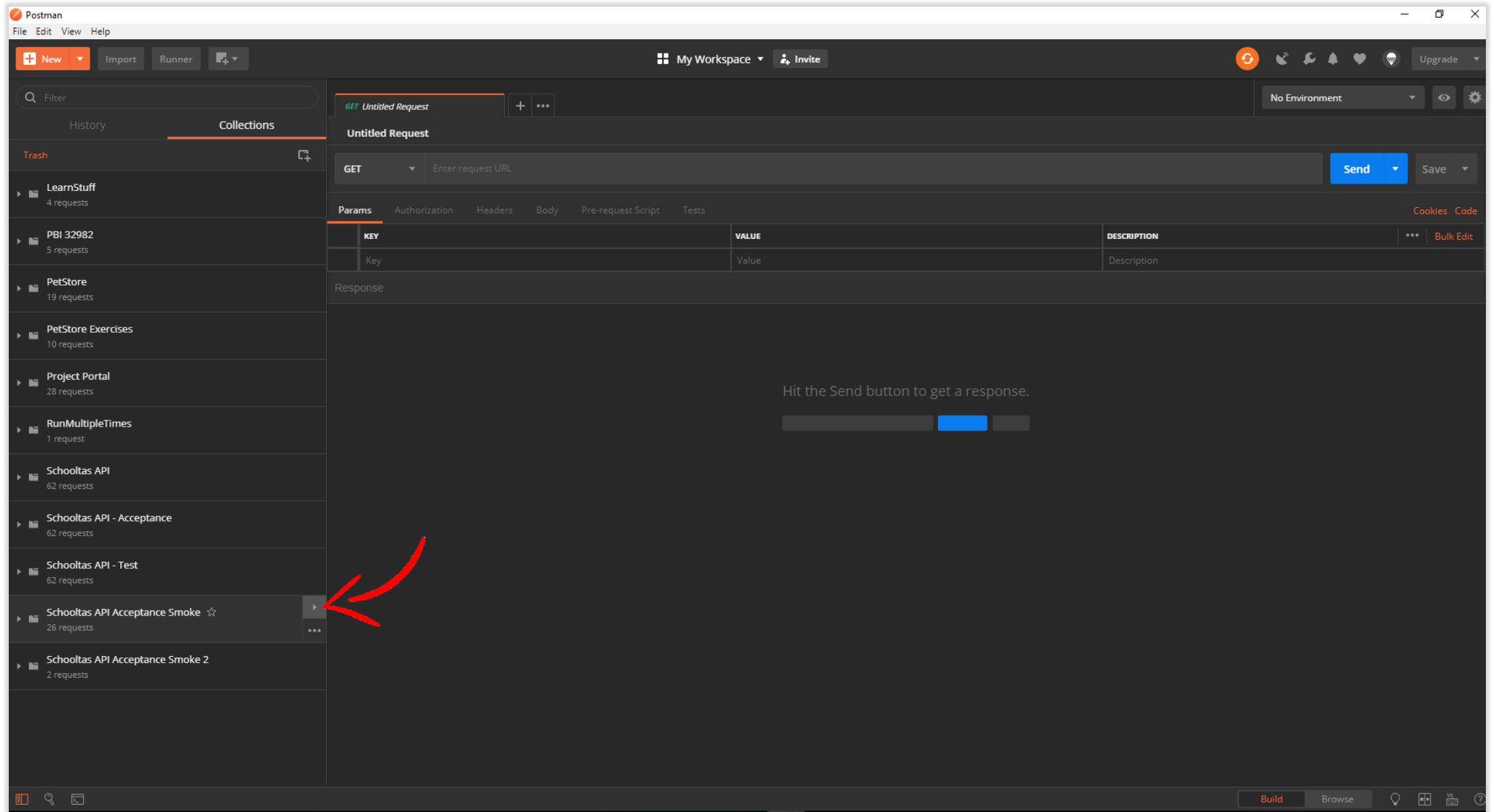
Create new environment when you run it for the first time



# Type the Environment name and press the Add button

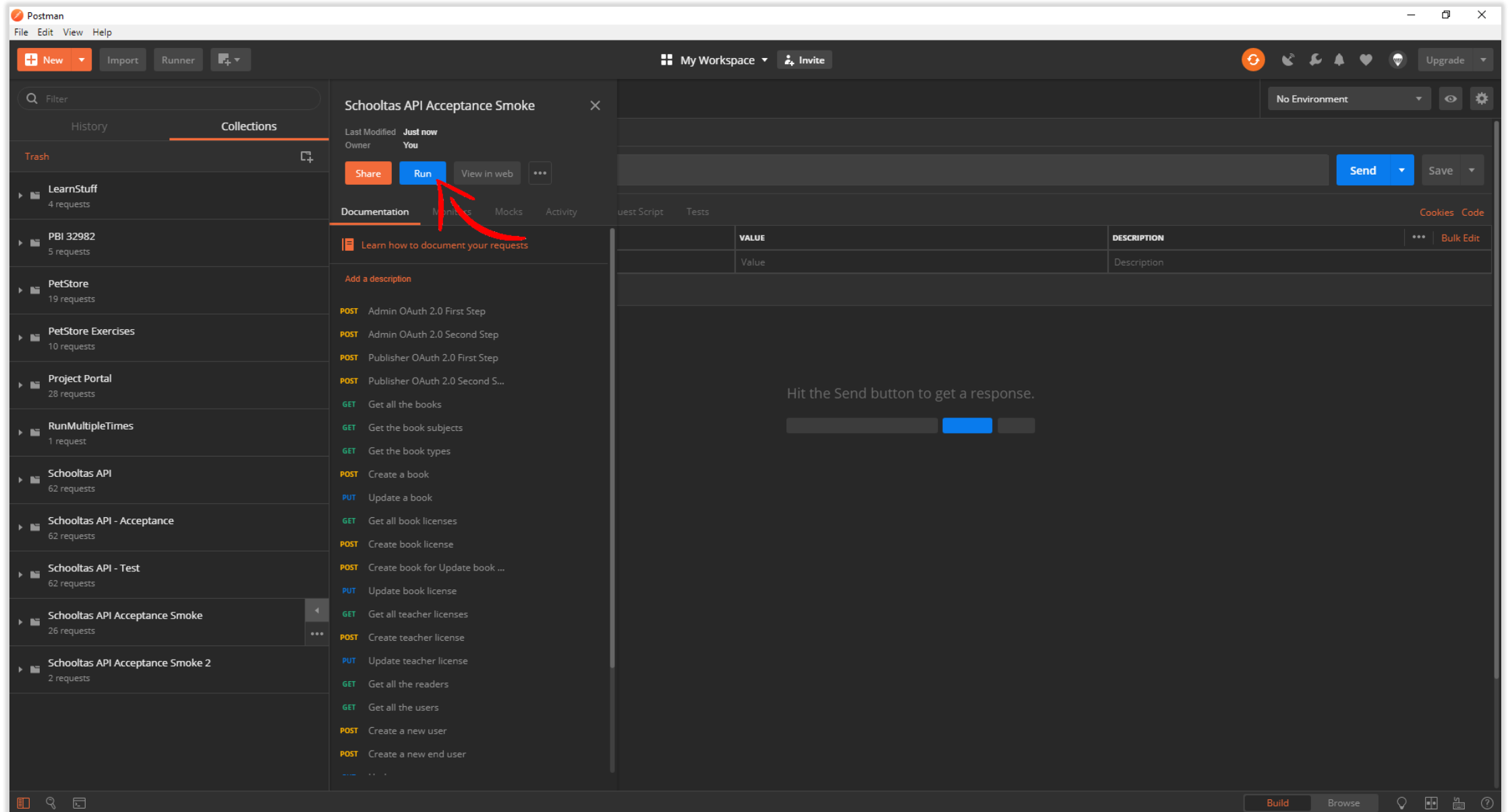


Hover over the collection and Press the right arrow to extend the collection view

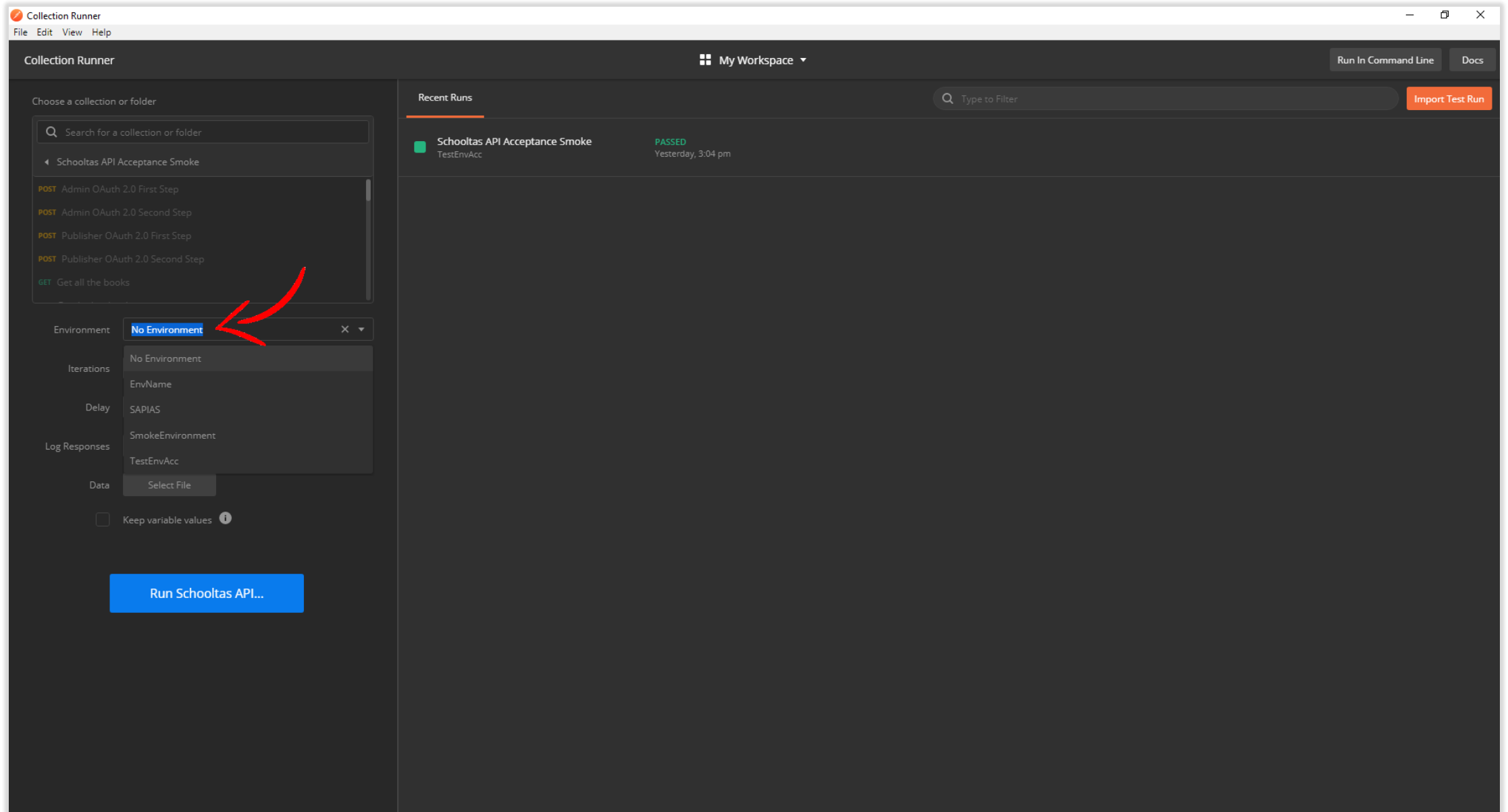




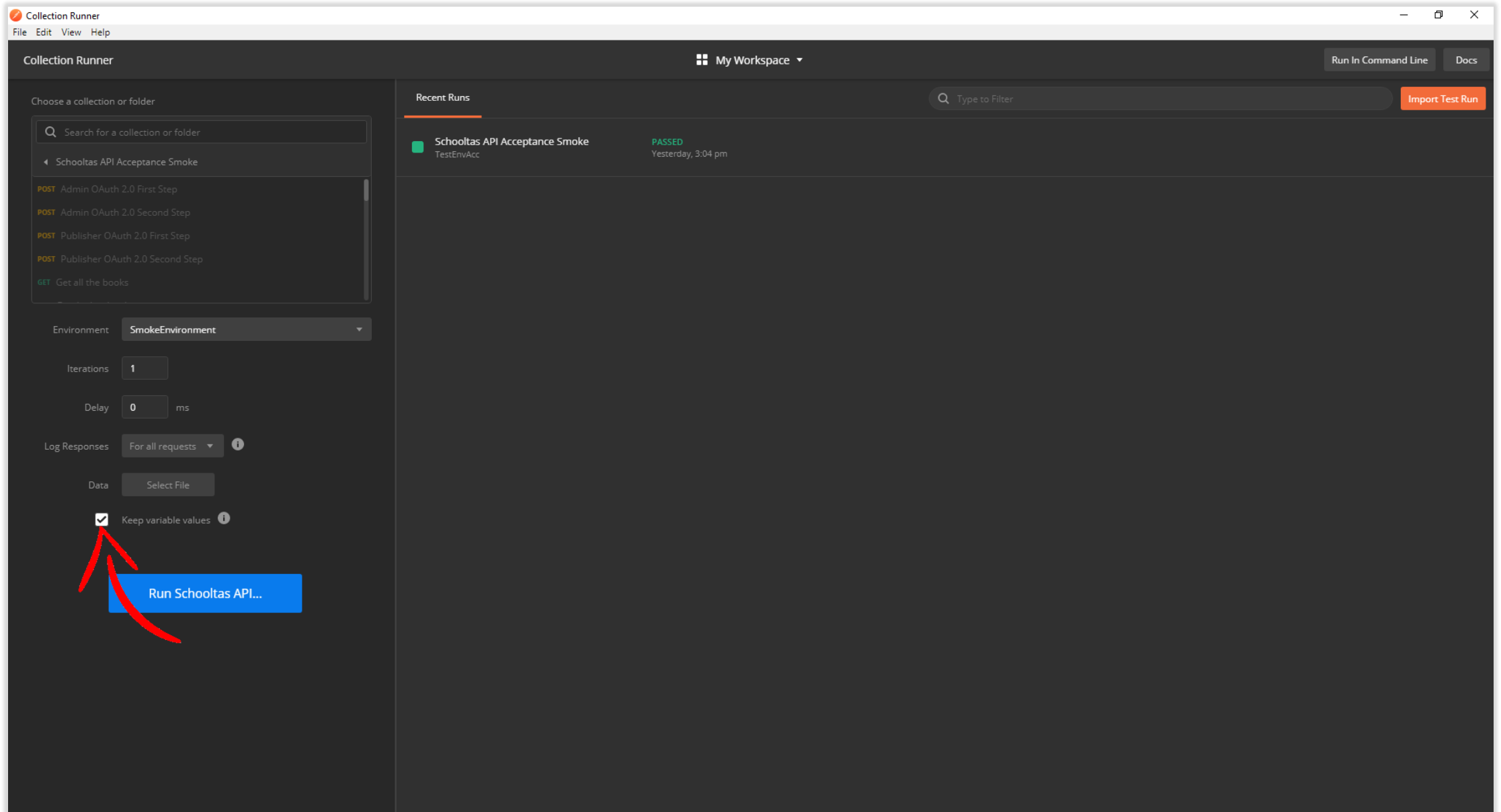
# Press the Run button to open the Collection Runner



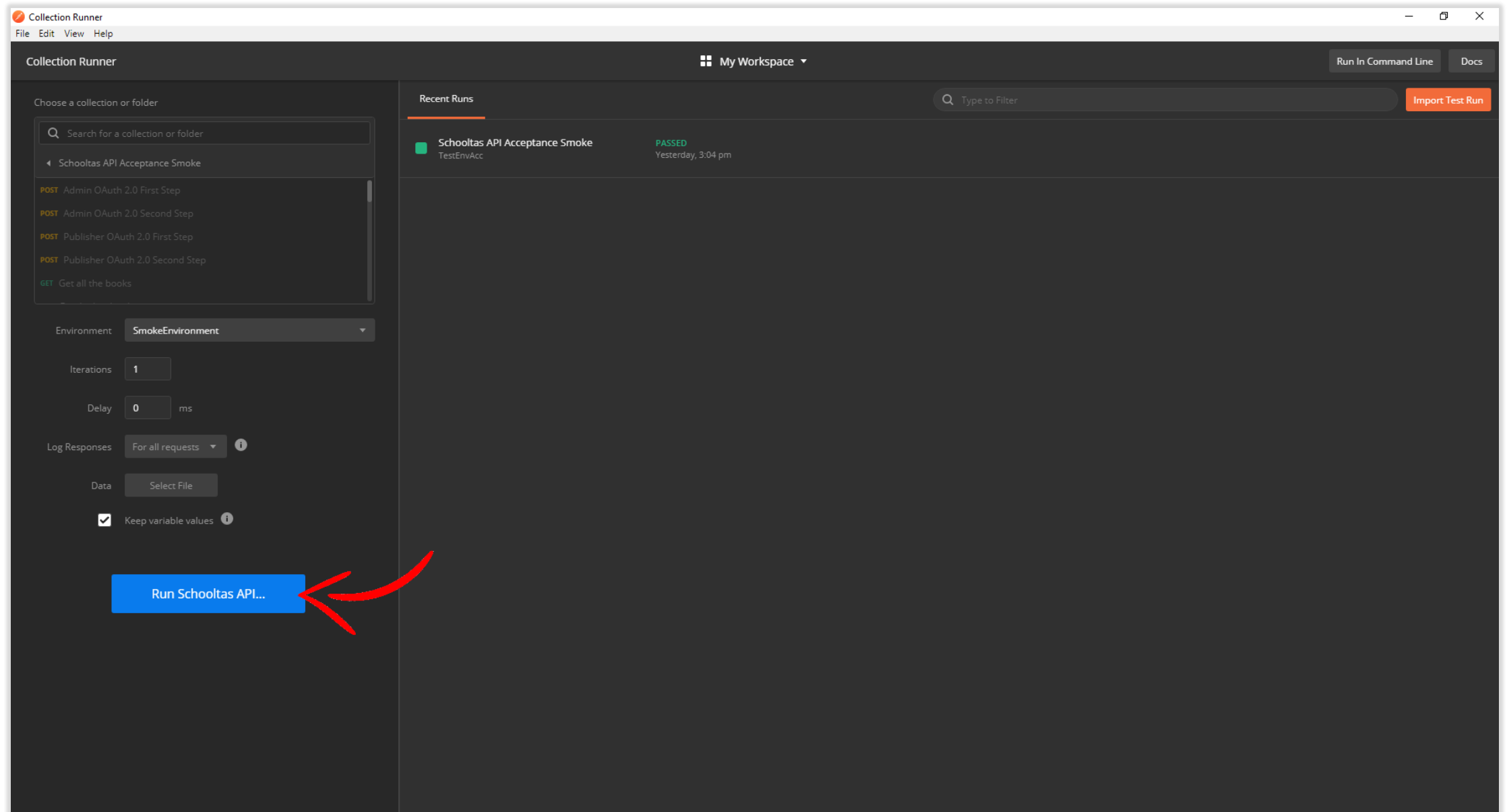
# Pick the Environment you just made



# Make sure that the **Keep variable values** is checked



# Press the **Run** button and watch the results



# All the passed or failed tests can be seen here

The screenshot displays the 'Collection Runner' application interface. At the top, a navigation bar includes 'Collection Runner', 'Run Results', and 'My Workspace'. A summary bar highlights the test 'Schooltas API Acceptance Smoke' with 77 passed and 0 failed tests. Below this, a table lists individual test steps for 'Iteration 1', showing their status (e.g., PASS, FAIL), HTTP methods, URLs, and response details.

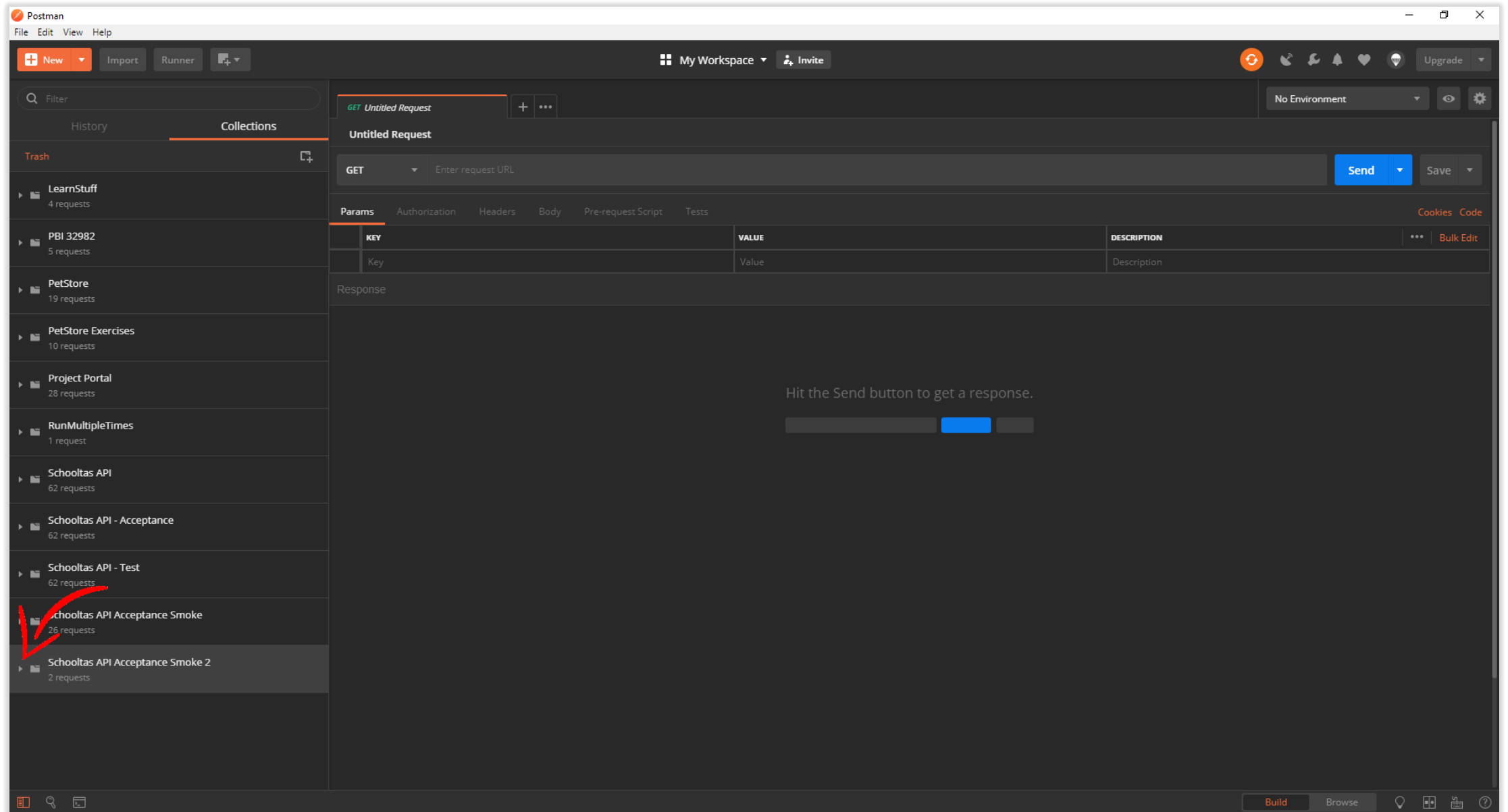
Iteration	Test Step	Status	Response
Iteration 1	POST Admin OAuth 2.0 First Step	PASS	302 Found, 351 ms, 0 B
	PASS Status code is 302	PASS	
	POST Admin OAuth 2.0 Second Step	PASS	200 OK, 95 ms, 138 B
	PASS Status code is 200	PASS	
	POST Publisher OAuth 2.0 First Step	PASS	302 Found, 82 ms, 0 B
	PASS Status code is 302	PASS	
	POST Publisher OAuth 2.0 Second Step	PASS	200 OK, 65 ms, 138 B
	PASS Status code is 200	PASS	
	GET Get all the books	PASS	200 OK, 1154 ms, 50.158 KB
	PASS Status code is 200	PASS	
	GET Get the book subjects	PASS	200 OK, 929 ms, 8.358 KB
	PASS Status code is 200	PASS	
	GET Get the book types	PASS	200 OK, 73 ms, 750 B
	PASS Status code is 200	PASS	
	POST Create a book	PASS	200 OK, 390 ms, 442 B
	PASS Status code is 200	PASS	
	PASS Ean is 6982196298141	PASS	

# Now comes the manually testing part

Due to the lack of knowledge and postman missing functionality, the create a new reader request cannot be automated, so it needs to be tested manually.

The update a reader request cannot be automated, due to its dependencies on the create a new reader request.

# Extend the second collection



# Open the Create a new reader request

The screenshot shows the Postman application interface. On the left sidebar, under the 'Collections' tab, the 'Schooltas API' collection is expanded, and the 'Create a new reader' request is selected. A red arrow points to this request. The main panel displays the details of this POST request, including the URL `https://api-acc.schooltas.net/v1/readers?language={{language}}` and a parameter table.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> language	<code>{{language}}</code>	
Key	Value	Description

Below the parameter table, the 'Response' section is visible, displaying the message: 'Hit the Send button to get a response.' with a blue 'Send' button.



# Set your created environment

The image shows the Postman application interface. On the left is a sidebar with a 'Collections' tab and a list of API collections including 'LearnStuff', 'PBI 32982', 'PetStore', 'PetStore Exercises', 'Project Portal', 'RunMultipleTimes', 'Schooltas API', 'Schooltas API - Acceptance', 'Schooltas API - Test', 'Schooltas API Acceptance Smoke', and 'Schooltas API Acceptance Smoke 2'. The main workspace displays a REST client request for a POST method to the URL 'https://api-acc.schooltas.net/v1/readers?language={{language}}'. The 'Params' tab is active, showing a parameter table with a key 'language' and value '{{language}}'. The 'Response' section shows a message: 'Hit the Send button to get a response.' with a blue 'Send' button. In the top right corner, a dropdown menu is set to 'No Environment', indicated by a red arrow.

Postman

File Edit View Help

New Import Runner +

My Workspace Invite

Filter

History Collections

Trash

LearnStuff  
4 requests

PBI 32982  
5 requests

PetStore  
19 requests

PetStore Exercises  
10 requests

Project Portal  
28 requests

RunMultipleTimes  
1 request

Schooltas API  
62 requests

Schooltas API - Acceptance  
62 requests

Schooltas API - Test  
62 requests

Schooltas API Acceptance Smoke  
26 requests

Schooltas API Acceptance Smoke 2  
2 requests

POST Create a new reader

Create a new reader

POST https://api-acc.schooltas.net/v1/readers?language={{language}}

Send Save

Params Authorization Headers Body Pre-request Script Tests

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> language	{{language}}	
Key	Value	Description

Response

Hit the Send button to get a response.

Build Browse

# Go to the Body tab of the request

The screenshot shows the Postman application interface. On the left is a sidebar with a 'Collections' tab selected, displaying a list of request collections. The main panel shows a request named 'POST Create a new reader' with the URL 'https://api-acc.schooltas.net/v1/readers?language={{language}}'. The 'Body' tab is selected and highlighted with a red arrow. Below the tab, the 'form-data' radio button is chosen. A table lists the form data entries: 'title' with value '{{reader\_title}}', 'subject' with value '{{book\_subject}}', 'pdfFile' with a 'Choose Files' button, and 'language' with value '{{language}}'. The bottom of the panel shows a 'Response' section with a message 'Hit the Send button to get a response.' and a 'Send' button.

Postman

File Edit View Help

New Import Runner

My Workspace Invite

SmokeEnvironment

Filter

History Collections

Trash

LearnStuff  
4 requests

PBI 32982  
5 requests

PetStore  
19 requests

PetStore Exercises  
10 requests

Project Portal  
28 requests

RunMultipleTimes  
1 request

Schooltas API  
62 requests

Schooltas API - Acceptance  
62 requests

Schooltas API - Test  
62 requests

Schooltas API Acceptance Smoke  
26 requests

Schooltas API Acceptance Smoke 2  
2 requests

POST Create a new reader

PUT Update a reader

POST Create a new reader

Params Authorization Headers **Body** Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary

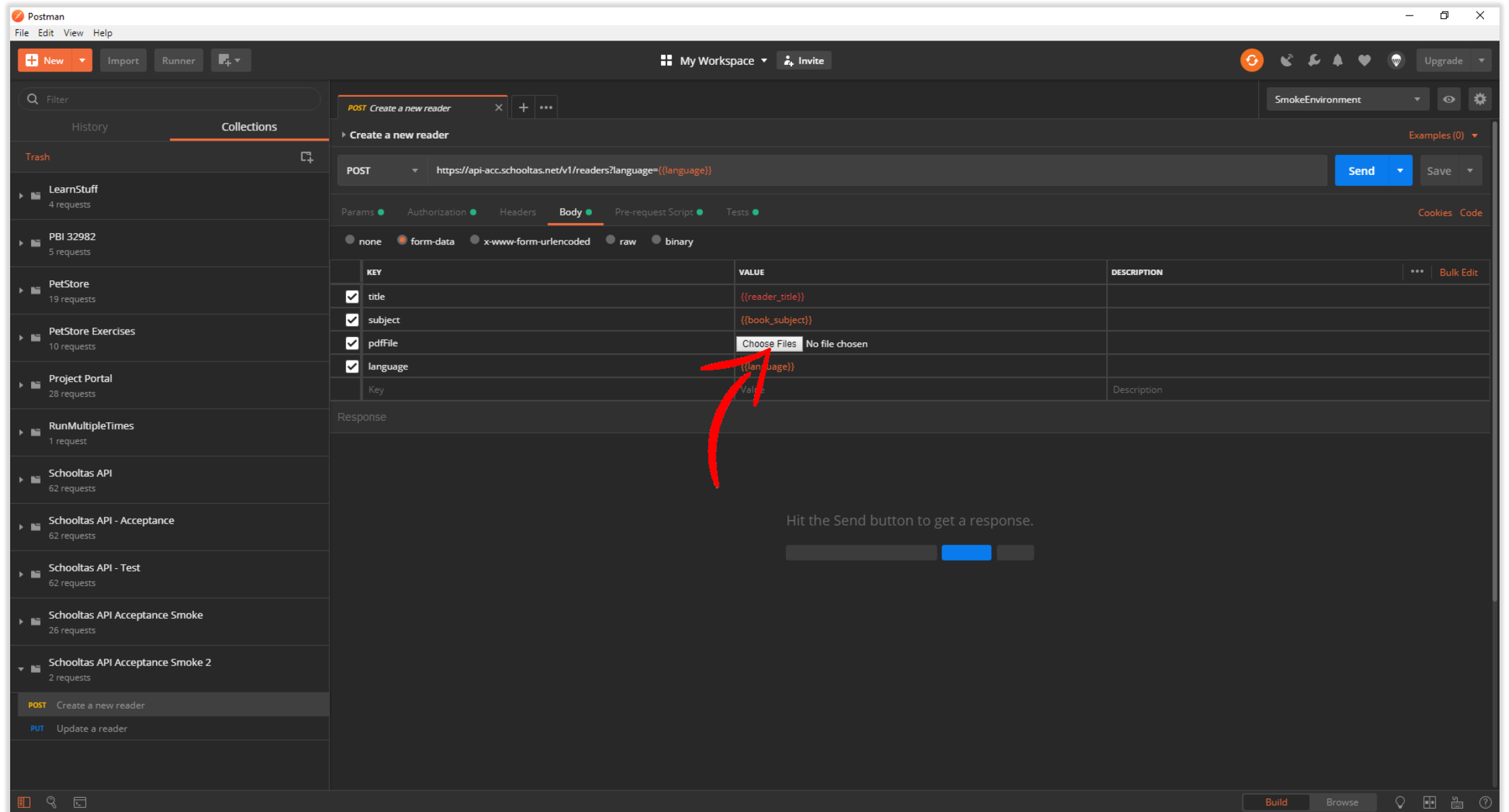
KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> title	{{reader_title}}	
<input checked="" type="checkbox"/> subject	{{book_subject}}	
<input checked="" type="checkbox"/> pdfFile	Choose Files No file chosen	
<input checked="" type="checkbox"/> language	{{language}}	
Key	Value	Description

Response

Hit the Send button to get a response.

Build Browse

# Choose a PDF for the reader



# Send the request

The image shows the Postman application interface. On the left is a sidebar with a 'Collections' tab. It lists several collections: 'LearnStuff' (4 requests), 'PBI 32982' (5 requests), 'PetStore' (19 requests), 'PetStore Exercises' (10 requests), 'Project Portal' (28 requests), 'RunMultipleTimes' (1 request), 'Schooltas API' (62 requests), 'Schooltas API - Acceptance' (62 requests), 'Schooltas API - Test' (62 requests), 'Schooltas API Acceptance Smoke' (26 requests), and 'Schooltas API Acceptance Smoke 2' (2 requests). The 'POST Create a new reader' request is selected at the bottom of the sidebar.

The main panel displays the configuration for the 'POST Create a new reader' request. The URL is `https://api-acc.schooltas.net/v1/readers?language={{language}}`. The 'Body' tab is active, showing a 'form-data' type. The body is a table with the following data:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> title	<code>{{reader_title}}</code>	
<input checked="" type="checkbox"/> subject	<code>{{book_subject}}</code>	
<input checked="" type="checkbox"/> pdfFile	Choose Files PostmanAutomationSteps.pdf	
<input checked="" type="checkbox"/> language	<code>{{language}}</code>	
Key	Value	Description

At the top right of the request configuration area, there is a blue 'Send' button. A red arrow points to this button. Below the 'Send' button is a 'Save' button. The bottom of the main panel shows a 'Response' section with the text 'Hit the Send button to get a response.' and a progress bar.

# Go to the Test Results tab

The screenshot displays the Postman application interface. On the left, a sidebar shows a list of collections, with 'Schooltas API Acceptance Smoke 2' selected at the bottom. The main workspace shows a POST request named 'Create a new reader' to the URL 'https://api-acc.schooltas.net/v1/readers?language={{language}}'. The 'Body' tab is active, showing a form with fields for 'title', 'subject', 'pdfFile', and 'language'. A red arrow points from the 'subject' field to the 'Test Results' tab. The 'Test Results' tab is selected, showing a JSON response with details about a reader and a PDF document. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 1405 ms', and 'Size: 1.39 KB'.

POST Create a new reader

Create a new reader

POST https://api-acc.schooltas.net/v1/readers?language={{language}}

Params Authorization Headers (1) Body Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> title	{{reader_title}}	
<input checked="" type="checkbox"/> subject	{{book_subject}}	
<input checked="" type="checkbox"/> pdfFile	Choose Files PostmanAutomationSteps.pdf	
<input checked="" type="checkbox"/> language	{{language}}	
Key	Value	Description

Body Cookies (1) Headers (11) Test Results (3/3)

Status: 200 OK Time: 1405 ms Size: 1.39 KB Save Download

Pretty Raw Preview JSON

```
1 {
2   "id": "9cc6b1be-4efc-4439-99a7-845fe5e606e2",
3   "ean": "Reader54c3b13",
4   "title": "Test Postman Reader",
5   "subject": {
6     "id": "b7fbc328-342f-4812-b187-053120861b20",
7     "translatedName": "Dutch"
8   },
9   "organization": {
10    "id": "3d67daf7-728e-4347-93b8-e8f39d615700",
11    "type": "School"
12  },
13  "coverUrl": "https://acceptance-applicationstack-1xj2-pdfcontentbucket-wai00nng9de5.s3.eu-central-1.amazonaws.com/book-content/covers/book_new_56e860fd-d9ba-4d51-949c-ea46c5b1be99.jpg",
14  "pdfUrl": "https://acceptance-applicationstack-1xj2-pdfcontentbucket-wai00nng9de5.s3.eu-central-1.amazonaws.com/organization-data/school/3d67daf7-728e-4347-93b8-e8f39d615700/document/PostmanAutomationSteps_4e523de2-7468-4716-9c4d-f6312562bfce.pdf",
15  "type": {
16    "name": "TeacherMaterial",
17    "translatedName": "Dutch"
18  },
19  "active": false,
20  "settings": {
21    "indexPageNumber": 0,
22    "pageNumberOffset": 1
23  },
24  "tableOfContents": null,
25  "pdfDownloadable": false
26 }
```

# Watch the results

The image shows the Postman application interface. On the left is a sidebar with a 'Collections' tab and a list of collections including 'LearnStuff', 'PBI 32982', 'PetStore', 'PetStore Exercises', 'Project Portal', 'RunMultipleTimes', 'Schooltas API', 'Schooltas API - Acceptance', 'Schooltas API - Test', 'Schooltas API Acceptance Smoke', and 'Schooltas API Acceptance Smoke 2'. The main panel displays a POST request to 'https://api-acc.schooltas.net/v1/readers?language={{language}}'. The 'Body' tab is selected, showing a form with fields: title (value: {{reader\_title}}), subject (value: {{book\_subject}}), pdfFile (value: PostmanAutomationSteps.pdf), and language (value: {{language}}). Below the body, the 'Test Results' tab is active, showing three passed tests: 'Status code is 200', 'Title is correct', and 'The reader subject is correct'. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 1405 ms', and 'Size: 1.39 KB'.

POST Create a new reader

POST https://api-acc.schooltas.net/v1/readers?language={{language}}

Params Authorization Headers (1) Body Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> title	{{reader_title}}	
<input checked="" type="checkbox"/> subject	{{book_subject}}	
<input checked="" type="checkbox"/> pdfFile	Choose Files PostmanAutomationSteps.pdf	
<input checked="" type="checkbox"/> language	{{language}}	
Key	Value	Description

Body Cookies (1) Headers (11) Test Results (3/3)

Status: 200 OK Time: 1405 ms Size: 1.39 KB Save Download

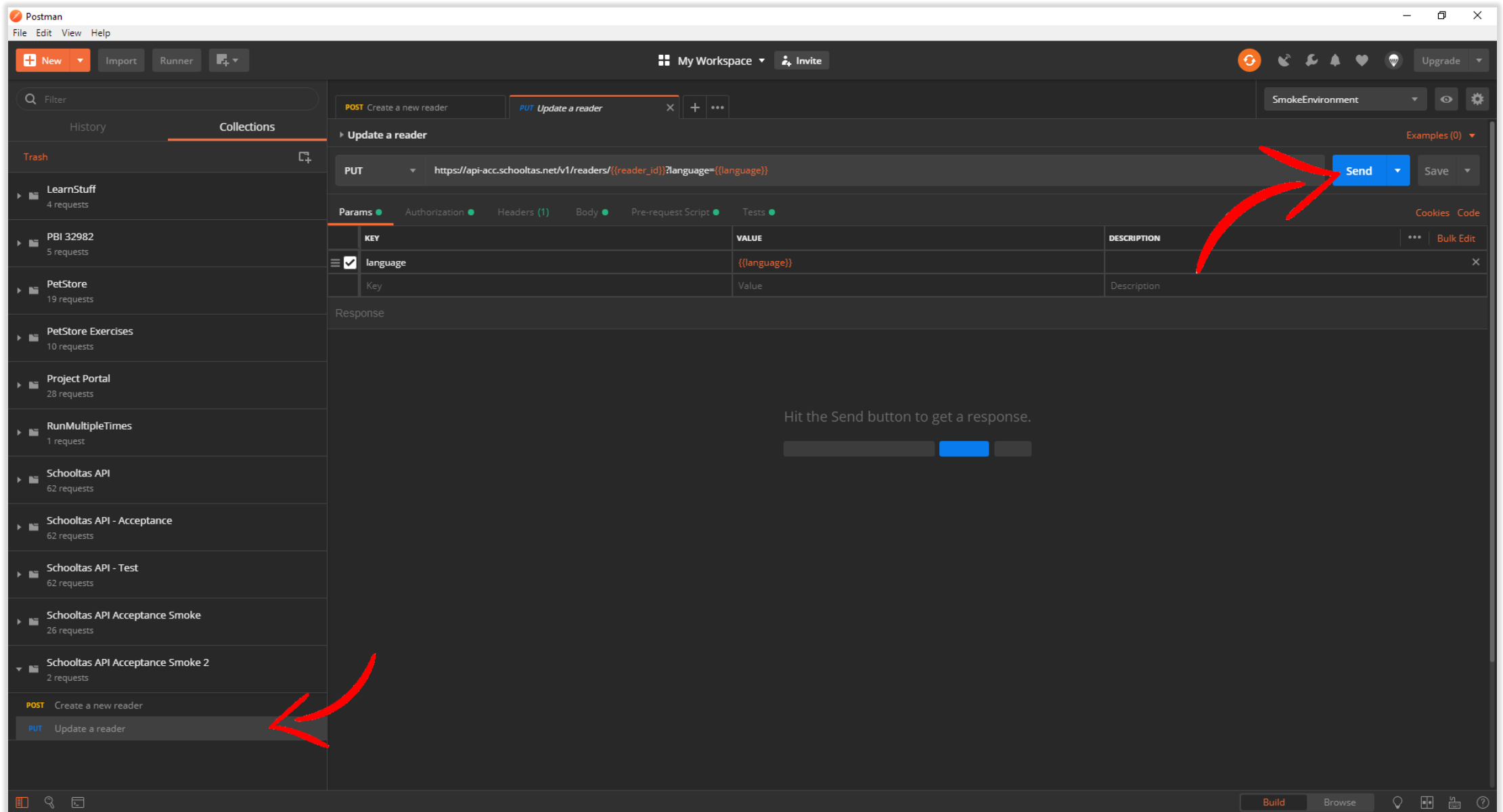
All Passed Skipped Failed

PASS Status code is 200

PASS Title is correct

PASS The reader subject is correct

# Select the next request and press the Send button



# Go to the Test Results tab

The screenshot shows the Postman application interface. On the left, there is a sidebar with a 'Collections' tab selected, showing a list of collections including 'LearnStuff', 'PBI 32982', 'PetStore', 'PetStore Exercises', 'Project Portal', 'RunMultipleTimes', 'Schooltas API', 'Schooltas API - Acceptance', 'Schooltas API - Test', 'Schooltas API Acceptance Smoke', and 'Schooltas API Acceptance Smoke 2'. The main panel displays a PUT request to 'https://api-acc.schooltas.net/v1/readers/{{reader\_id}}?language={{language}}'. The 'Test Results' tab is selected, showing a JSON response. A red arrow points to the 'Test Results' tab label.

**Update a reader**

PUT `https://api-acc.schooltas.net/v1/readers/{{reader_id}}?language={{language}}` Send Save

Params Authorization Headers (2) Body Pre-request Script Tests

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> language	{{language}}	
Key	Value	Description

Body Cookies (1) Headers (11) Test Results (5/5) Status: 200 OK Time: 418 ms Size: 1.39 KB Save Download

Pretty Raw Preview JSON ≡

```
1 {
2   "id": "9cc6b1be-4efc-4439-99a7-845fe5e606e2",
3   "ean": "Reader54c3b13",
4   "title": "Test Postman Updated Reader",
5   "subject": {
6     "id": "c94b016e-3f95-4723-838c-922alc59bbdd",
7     "translatedName": "Music"
8   },
9   "organization": {
10    "id": "3d67daf7-728e-4347-93b8-e8f39d615700",
11    "type": "School"
12  },
13  "coverUrl": "https://acceptance-applicationstack-1xj2-pdfcontentbucket-wai00nng9de5.s3.eu-central-1.amazonaws.com/book-content/covers/book_new_56e860fd-d9ba-4d51-949c-ea46c5b1be99.jpg",
14  "pdfUrl": "https://acceptance-applicationstack-1xj2-pdfcontentbucket-wai00nng9de5.s3.eu-central-1.amazonaws.com/organization-data/school/3d67daf7-728e-4347-93b8-e8f39d615700/document/PostmanAutomationSteps_4e523de2-7468-4716-9c4d-f6312562bfce.pdf",
15  "type": {
16    "name": "TeacherMaterial",
17    "translatedName": "Music"
18  },
19  "active": true,
20  "settings": {
21    "indexPageNumber": 0,
22    "pageNumberOffset": 1
23  },
24  "tableOfContents": null,
25  "pdfDownloadable": false
26 }
```

Build Browse ?



# Watch the results

The image shows the Postman application interface. On the left is a sidebar with a 'Collections' tab and a list of collections including 'LearnStuff', 'PBI 32982', 'PetStore', 'PetStore Exercises', 'Project Portal', 'RunMultipleTimes', 'Schooltas API', 'Schooltas API - Acceptance', 'Schooltas API - Test', 'Schooltas API Acceptance Smoke', and 'Schooltas API Acceptance Smoke 2'. The main workspace displays a 'PUT' request to 'https://api-acc.schooltas.net/v1/readers/{{reader\_id}}?language={{language}}'. The 'Test Results' tab is active, showing five passed tests: 'Status code is 200', 'The updated reader title is correct', 'The updated reader id is correct', 'The updated reader ean is correct', and 'The updated reader subject is correct'. These tests are enclosed in a red rectangular box. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 418 ms', and 'Size: 1.39 KB'.

Postman

File Edit View Help

New Import Runner

My Workspace Invite

SmokeEnvironment

Filter

History Collections

Trash

LearnStuff 4 requests

PBI 32982 5 requests

PetStore 19 requests

PetStore Exercises 10 requests

Project Portal 28 requests

RunMultipleTimes 1 request

Schooltas API 62 requests

Schooltas API - Acceptance 62 requests

Schooltas API - Test 62 requests

Schooltas API Acceptance Smoke 26 requests

Schooltas API Acceptance Smoke 2 2 requests

POST Create a new reader

PUT Update a reader

Update a reader

PUT https://api-acc.schooltas.net/v1/readers/{{reader\_id}}?language={{language}}

Send Save

Params Authorization Headers (2) Body Pre-request Script Tests

KEY	VALUE	DESCRIPTION
language	{{language}}	
Key	Value	Description

Body Cookies (1) Headers (11) Test Results (5/5)

Status: 200 OK Time: 418 ms Size: 1.39 KB Save Download

All Passed Skipped Failed

PASS Status code is 200

PASS The updated reader title is correct

PASS The updated reader id is correct

PASS The updated reader ean is correct

PASS The updated reader subject is correct

Build Browse