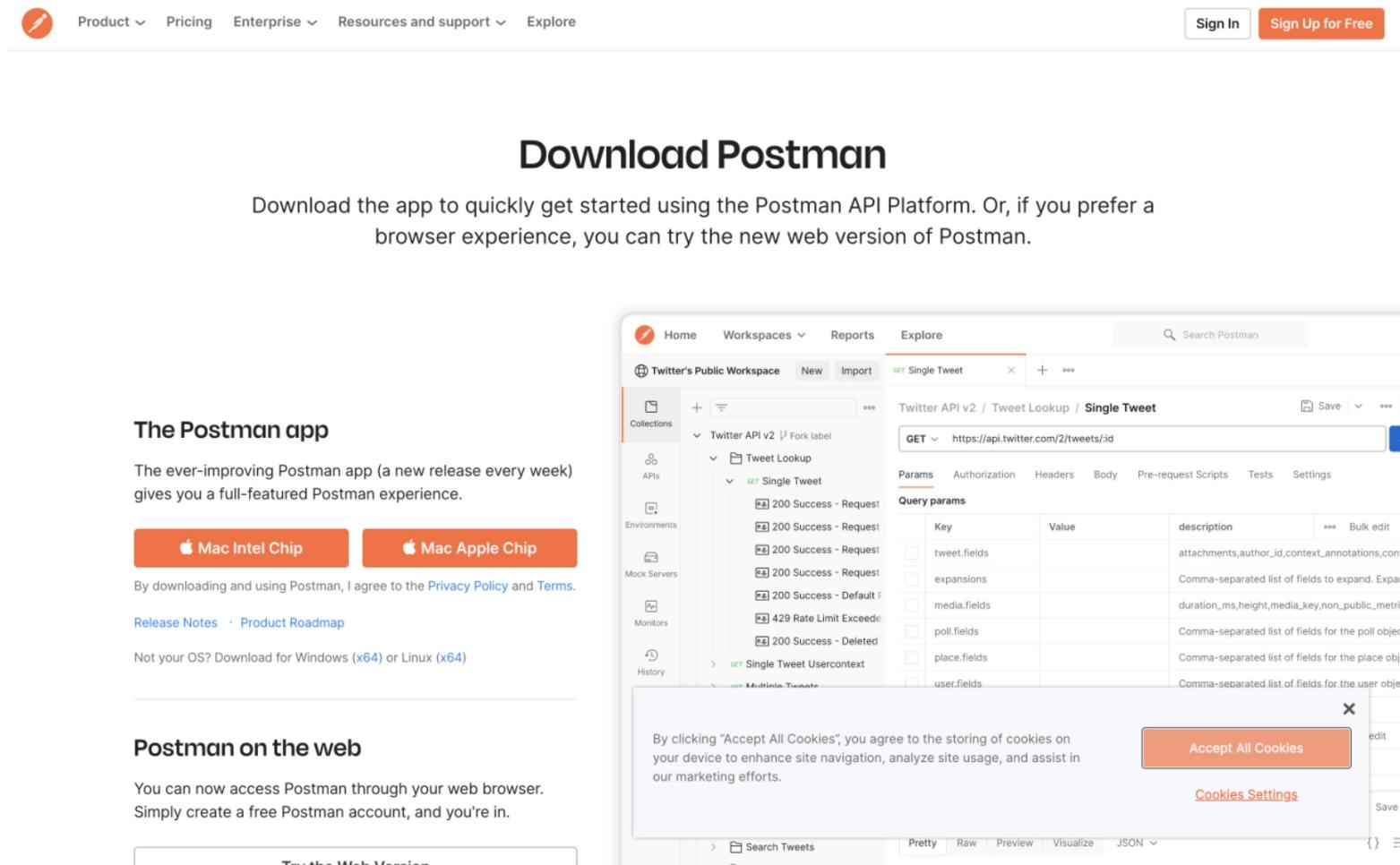


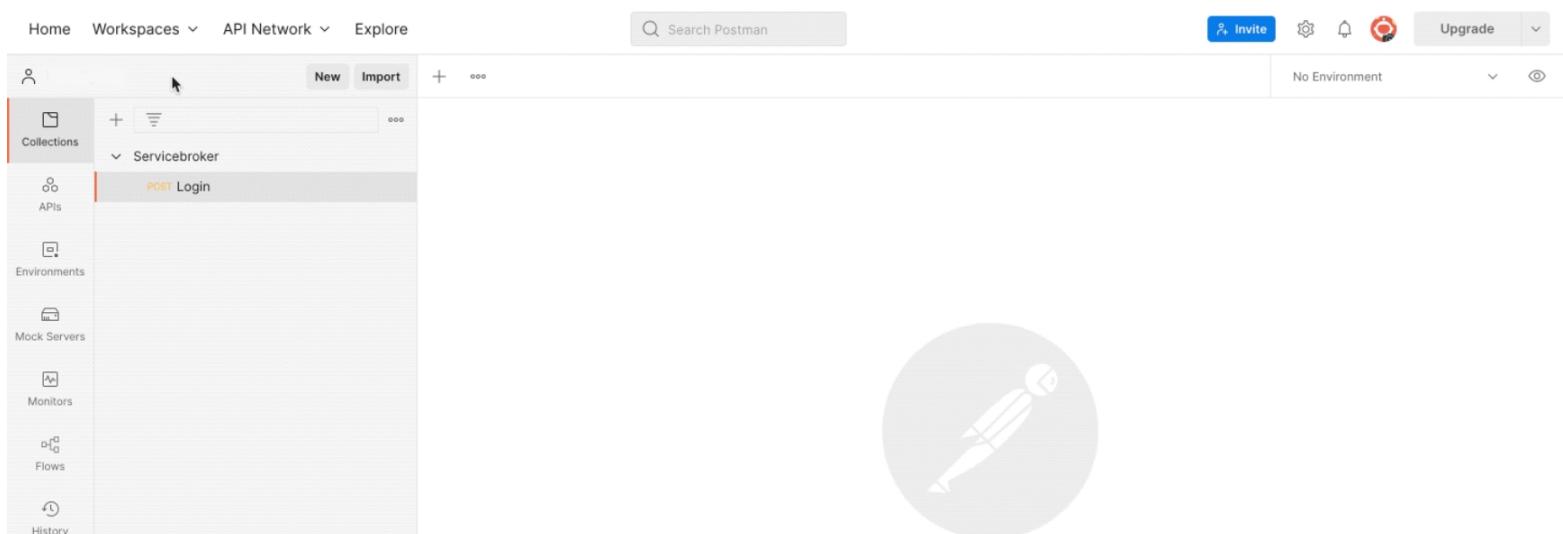
Using Postman with Apollo Server

Download the **Postman** app from <https://www.postman.com/downloads/>:

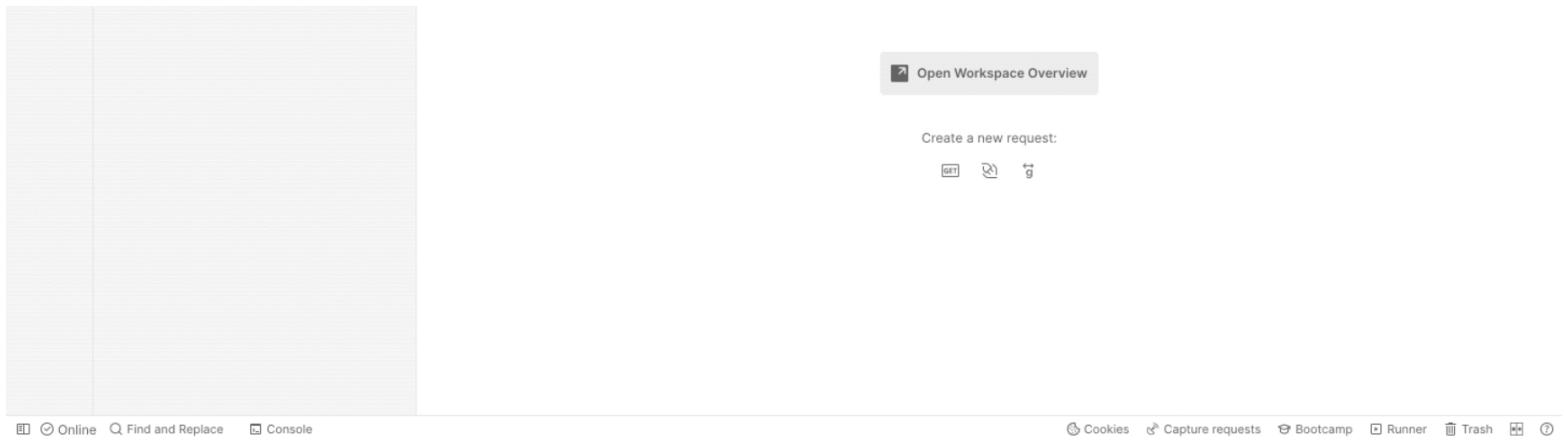


The screenshot shows the Postman website's download page. At the top, there's a navigation bar with links for Product, Pricing, Enterprise, Resources and support, and Explore. On the right are Sign In and Sign Up for Free buttons. Below the navigation is a large orange button labeled "Download Postman". Underneath it, text says "Download the app to quickly get started using the Postman API Platform. Or, if you prefer a browser experience, you can try the new web version of Postman." To the left, there's a section titled "The Postman app" with two download buttons for Mac Intel Chip and Mac Apple Chip. It also includes links for Privacy Policy and Terms, Release Notes, and a Product Roadmap. A note for non-Mac users provides links for Windows (x64) and Linux (x64). To the right, a larger screenshot of the Postman interface shows a workspace named "Twitter's Public Workspace" with a "Single Tweet" collection. The interface includes tabs for Home, Workspaces, Reports, and Explore. On the left sidebar, there are sections for Collections, APIs, Environments, Mock Servers, Monitors, and History. The main workspace area shows a request for "GET / Single Tweet" with the URL <https://api.twitter.com/2/tweets/:id>. The "Query params" section lists several parameters like tweet.fields, expansions, media.fields, poll.fields, place.fields, and user.fields. At the bottom, there's a note about accepting cookies and buttons for "Accept All Cookies" and "Cookies Settings".

Open Postman and in the main menu, click on **Workspaces** followed by the **Create Workspace** button:



The screenshot shows the Postman application interface. At the top, there's a navigation bar with Home, Workspaces, API Network, and Explore. A search bar is also at the top. On the right, there are buttons for Invite, Upgrade, and a profile icon. The main workspace area shows a "Servicebroker" collection with a "Login" POST request highlighted. On the left, there's a sidebar with icons for Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. A large circular icon with a pen nib is centered in the bottom right of the workspace area.



Type [REDACTED] in the **Name** field and click on the **Personal** radio button under **Visibility**. Then, click on the **Create Workspace** button:

A screenshot of the "Create workspace" dialog in Postman. The title bar says "Create workspace". There are three main sections: 1) "Name" with a text input field containing "[REDACTED]". 2) "Summary" with a text input field containing "Add a brief summary about this workspace." 3) "Visibility" with four radio button options: "Personal" (selected, "Only you can access"), "Private" (unselected, "Only invited team members can access"), "Team" (unselected, "All team members can access"), and "Public" (unselected, "Everyone can view"). At the bottom are "Create Workspace" and "Cancel" buttons.

In the main menu, click on **Environments** followed by the **+ (Create new Environment)** button or the **Create Environment** link:

A screenshot of the Postman interface showing the environments section. The top navigation bar includes "Home", "Workspaces", "API Network", "Explore", "Search Postman", "Invite" (blue button), "Settings" (gear icon), "Capture requests" (camera icon), "Bootcamp" (globe icon), "Runner" (play icon), "Trash" (trash bin icon), and "Upgrade". The main area shows a sidebar with "Collections" and "New Import" buttons. A red bar highlights the "Overview" tab, which displays a "+ Add" button and a user icon. To the right, it says "No Environment" and has a "Workspace Settings" link.

Add **SB-DEV** in the **Name** field and complete the **Variable** and **Initial Value** fields with the following information:

Variable Initial	Value
requestUrl	[REDACTED]
learner7	[REDACTED]
learner7p	[REDACTED]
learner7Id	[REDACTED]

Click on the **Reset All** button to copy the **Initial Value** variables from the field to the **Current Value** field:

VARIABLE	TYPE ⓘ	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ	Persist All	Reset All
requestUrl	default	▼	▼		
learnerId	default	▼	▼		
username	default	▼	▼		
password	secret	▼	▼		
learner7	default	▼	▼		
learner7p	default	▼	▼		
learner7Id	default	▼	▼		
subjectIdEN	default	▼	▼		

The screenshot shows the Postman interface with the 'Flows' tab selected. On the left, there's a sidebar with icons for History, Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area displays a table of variables:

<input checked="" type="checkbox"/>	subjectIdFR	default	▼
<input checked="" type="checkbox"/>	subjectIdDE	default	▼
<input checked="" type="checkbox"/>	karlHorningTest	default	▼
<input checked="" type="checkbox"/>	karlHorningTestP	default	▼
<input checked="" type="checkbox"/>	karlHorningTestId	default	▼
Add a new variable			

At the bottom, there are navigation links: Online, Find and Replace, Console, Cookies, Capture requests, Bootcamp, Runner, Trash, and Help.

Then, click on the **Save** button:

The screenshot shows the Postman interface with the 'Environments' tab selected. On the left, there's a sidebar with icons for Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area displays a table of environment variables for the 'SB-DEV' environment:

	VARIABLE	TYPE ⓘ	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ	...	Persist All	Reset All
<input checked="" type="checkbox"/>	requestUrl	default	▼				
<input checked="" type="checkbox"/>	learnerId	default	▼				
<input checked="" type="checkbox"/>	username	default	▼				
<input checked="" type="checkbox"/>	password	secret	▼				
<input checked="" type="checkbox"/>	learner7	default	▼				
<input checked="" type="checkbox"/>	learner7p	default	▼				
<input checked="" type="checkbox"/>	learner7Id	default	▼				
<input checked="" type="checkbox"/>	subjectIdEN	default	▼				
<input checked="" type="checkbox"/>	subjectIdFR	default	▼				
<input checked="" type="checkbox"/>	subjectIdDE	default	▼				
<input checked="" type="checkbox"/>	karlHorningTest	default	▼				
<input checked="" type="checkbox"/>	karlHorningTestP	default	▼				
<input checked="" type="checkbox"/>	karlHorningTestId	default	▼				
Add a new variable							

At the bottom, there are navigation links: Online, Find and Replace, Console, Cookies, Capture requests, Bootcamp, Runner, Trash, and Help. The 'Save' button is highlighted with a red box.

Click on the **Collections** tab, then click on the **+ (Create new Collection)** button in the top left of the **Collections** panel. Alternatively, click on the **Create collection** button in the middle of the **Collections** panel:

The screenshot shows the Postman interface with the 'Collections' tab selected. On the left, there's a sidebar with icons for History, Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area shows the 'Collections' panel with a 'New' button and an 'Import' button. At the bottom, there are navigation links: Online, Find and Replace, Console, Cookies, Capture requests, Bootcamp, Runner, Trash, and Help.

	VARIABLE	TYPE ⓘ	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ	...	Persist All	Reset All
<input checked="" type="checkbox"/>	requestUrl	default	▼				
<input checked="" type="checkbox"/>	learnerId	default	▼				
<input checked="" type="checkbox"/>	username	default	▼				
<input checked="" type="checkbox"/>	password	secret	▼				
<input checked="" type="checkbox"/>	learner7	default	▼				
<input checked="" type="checkbox"/>	learner7p	default	▼				
<input checked="" type="checkbox"/>	learner7ld	default	▼				
<input checked="" type="checkbox"/>	subjectIdEN	default	▼				
<input checked="" type="checkbox"/>	subjectIdFR	default	▼				
<input checked="" type="checkbox"/>	subjectIdDE	default	▼				
<input checked="" type="checkbox"/>	karlHorningTest	default	▼				
<input checked="" type="checkbox"/>	karlHorningTestP	default	▼				
<input checked="" type="checkbox"/>	karlHorningTestId	default	▼				
<input checked="" type="checkbox"/>	token	default	▼				
Add a new variable							

ⓘ Use variables to reuse values and protect sensitive data
Store sensitive data in variable type secret to keep its values masked on the screen. Learn more about variable type
Work with the current value of a variable to prevent sharing sensitive values with your team. Learn more about variable values

Online Find and Replace Console

Cookies Capture requests Bootcamp Runner Trash

Name the new collection **Servicebroker**. Then either click on the ellipse (...) next to **Servicebroker** and choose **Add request** from the menu or click on **Add a request** under the **Servicebroker** name:

Home Workspaces API Network Explore Search Postman

New Import Overview SB-DEV Servicebroker + ... No Environment

collections + ...

Servicebroker

Authorization Pre-request Script Tests Variables

This collection is empty Add a request to start working.

Type No Auth

This collection does not use any authorization. Learn more about authorization

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash ?

Name the new request **Login** and change the request type from **GET** to **POST**:

Home Workspaces API Network Explore Search Postman Invite Upgrade

Collections APIs Environments Mock Servers Monitors Flows History

Overview SB-DEV Servicebroker GET Login + ... No Environment

Servicebroker / **Login**

GET Enter request URL

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Response

Enter the URL and click Send to get a response



Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash ?

In the **Enter request URL** field, enter `{{requestUrl}}`:

Home Workspaces API Network Explore Search Postman Invite Upgrade

Collections APIs Environments Mock Servers Monitors Flows History

Overview SB-DEV Servicebroker POST Login + ... No Environment

Servicebroker / **Login**

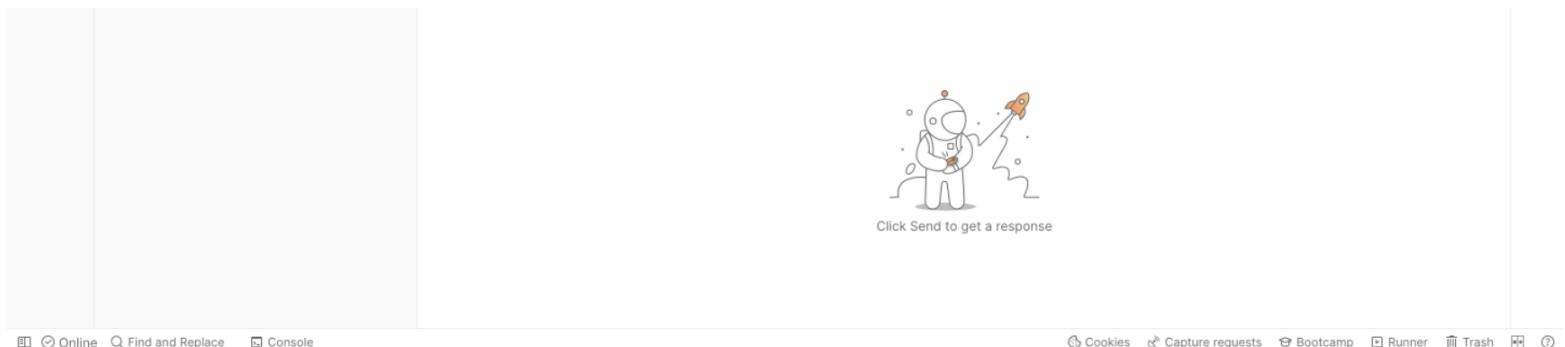
POST {{requestUrl}}

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

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Response



Note: In Postman, we can access variables stored in the **Environments** tab in **Collections** by surrounding the variable name with two sets of curly braces.

Click on the **Authorization** tab, then in the **Type** dropdown, select **No Auth**:

The screenshot shows the Postman interface with the 'Collections' sidebar open, displaying a collection named 'Servicebroker'. The 'Authorization' tab is selected in the request configuration area. The 'Type' dropdown menu is open, showing options like 'OAuth', 'Basic Auth', 'No Auth', and 'Custom'. The 'No Auth' option is highlighted with a red border. The main workspace shows a placeholder message 'Click Send to get a response'.

Click on the **Body** tab (in the same area as the **Authorization** tab), then select the **raw** radio button and choose **JSON** from the dropdown on the right:

The screenshot shows the Postman interface with the 'Collections' sidebar open, displaying a collection named 'Servicebroker'. The 'Body' tab is selected in the request configuration area. A dropdown menu is open next to the 'raw' radio button, showing options like 'Text', 'JSON', 'CSV', etc., with 'JSON' being the selected choice. The main workspace shows a placeholder message 'Click Send to get a response'.

The screenshot shows the Postman application interface. On the left, there's a sidebar with icons for APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area has tabs for Params, Authorization, Headers (7), Body (selected), Pre-request Script, Tests, and Settings. Below these tabs, there are radio buttons for none, form-data, x-www-form-urlencoded, raw, binary, and GraphQL. A message says "This request does not have a body". In the center, there's a "Response" section with a small illustration of an astronaut launching a rocket. Below the illustration, a button says "Click Send to get a response". At the bottom of the screen, there are various status indicators and links for Cookies, Capture requests, Bootcamp, Runner, Trash, and Help.

Copy the following code and paste it in the text area:

```
{
  "operationName": "Login",
  "variables": {
    "authcontextid": "",
      "role": "learner",
      "username": "{{learner7}}",
      "password": "{{learner7p}}"
  },
  "query": "mutation Login($authcontextid: UUID!, $role: String!, $username: String!, $password: String!) {\n  login(input: {authcontextid: $authcontextid, role: $role, username: $username, password: $password}) {\n    jwt\n  }\n}\n"
}
```

Navigate to uuidgenerator.net, generate a UUID, and paste it next to **authcontextid** in the JSON:

```
{
  "operationName": "Login",
  "variables": {
    "authcontextid": "3b11ed0a-e903-4fe3-a33a-1a419e08baa9",
    "role": "learner",
    "username": "{{learner7}}",
    "password": "{{learner7p}}"
  },
  "query": "mutation Login($authcontextid: UUID!, $role: String!, $username: String!, $password: String!) {\n  login(input: {authcontextid: $authcontextid, role: $role, username: $username, password: $password}) {\n    jwt\n  }\n}\n"
}
```

Click on the **Tests** tab (in the same area as the **Authorization** and **Body** tabs), then copy the following code and paste it in the text area:

```
const jsonResponse = pm.response.json();
const jwtToken = jsonResponse.data?.login?.jwt;

if(jwtToken){
  let [jwtHeader, jwtPayload, jwtSignature] = jwtToken.split('.')
  const words = CryptoJS.enc.Base64.parse(jwtPayload);
  const text = CryptoJS.enc.Utf8.stringify(words);
  const jsonData = JSON.parse(text);
  console.log(jsonData);
  pm.environment.set("learnerId", jsonData.id);
}

pm.environment.set("token", jwtToken);
```

Note: As the JWT is prefixed by other information in the JSON return object, the above code is used to separate the JWT.

Click on the **Save** button to save the request:

The screenshot shows the Postman interface with the following details:

- Header Bar:** Home, Workspaces, API Network, Explore, Search Postman, Invite, Settings, Upgrade.
- Sidebar:** Collections (Servicebroker), APIs (Servicebroker).
- Request Details:**
 - Method: POST
 - URL: {{requestUrl}}
 - Headers: (8)
 - Body: (green dot)
 - Pre-request Script: (green dot)
 - Tests: (green dot)
 - Settings
- Bottom Navigation:** Save, Send, Cookies.

Screenshots of the Postman interface showing the environment setup and a test script for extracting a JWT token from a response.

Left Sidebar:

- Environments
- Mock Servers
- Monitors
- Flows
- History

Test Script (Response tab):

```

1 const jsonResponse = pm.response.json();
2
3 const jwtToken = jsonResponse.data?.login?.jwt;
4
5 if(jwtToken){
6     let [jwtHeader, jwtPayload, jwtSignature] = jwtToken.split('.')
7     const words = CryptoJS.enc.Base64.parse(jwtPayload);
8     const text = CryptoJS.enc.Utf8.stringify(words);
9     const jsonData = JSON.parse(text);
10    console.log(jsonData);
11    pm.environment.set("learnerId", jsonData.id);
12 }
13
14 pm.environment.set("token", jwtToken);

```

Right Sidebar:

- 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
 - Set a collection variable
 - Clear an environment variable

Bottom:

- Online
- Find and Replace
- Console
- Cookies
- Capture requests
- Bootcamp
- Runner
- Trash
- ?

In the top right corner of **Postman**, click on the down arrow and select **SB_DEV**:

Screenshot of the Postman interface showing the 'Login' request configuration and an error message.

Left Sidebar:

- Collections
 - + New
 - Import
 - Overview
 - POST Login
- APIs
- Environments
- Mock Servers
- Monitors
- Flows
- History

Request Configuration:

Method: POST **URL:** {{requestUrl}}

Tests Tab:

```

1 const jsonResponse = pm.response.json();
2 const jwtToken = jsonResponse.data?.login?.jwt;
3
4 if(jwtToken){
5     let [jwtHeader, jwtPayload, jwtSignature] = jwtToken.split('.')
6     const words = CryptoJS.enc.Base64.parse(jwtPayload);
7     const text = CryptoJS.enc.Utf8.stringify(words);
8     const jsonData = JSON.parse(text);
9     console.log(jsonData);
10    pm.environment.set("learnerId", jsonData.id);
11 }
12
13 pm.environment.set("token", jwtToken);

```

Right Sidebar:

- 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
 - Set a collection variable
 - Clear an environment variable

Bottom:

- Online
- Find and Replace
- Console
- Cookies
- Capture requests
- Bootcamp
- Runner
- Trash
- ?

Error Message:

Could not send request
Error: getaddrinfo ENOTFOUND {{requesturl}} | [View in Console](#)

Click on the **Send** button to send a POST request:

New Import

POST Login

Servicebroker / Login

POST {{(requestUrl)}}

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

```
1 const jsonResponse = pm.response.json();
2
3 const jwtToken = jsonResponse.data?.login?.jwt;
4
5 if(jwtToken){
6     let [jwtHeader, jwtPayload, jwtSignature] = jwtToken.split('.');
7     const words = CryptoJS.enc.Base64.parse(jwtPayload);
8     const text = CryptoJS.enc.Utf8.stringify(words);
9     const jsonData = JSON.parse(text);
10    console.log(jsonData);
11    pm.environment.set("learnerId", jsonData.id);
12 }
13
14 pm.environment.set("token", jwtToken);
```

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
- Set a collection variable
- Clear an environment variable

Response

Click Send to get a response

Offline Find and Replace Console

Cookies Capture requests Bootcamp Runner Trash

Note: The JWT will automatically be saved in **Environments > SB-DEV > token** and is accessible when running other queries and mutations