

# How to invoke an API

(Step-by-step tutorial)

## Content

Create app.....	2
Step 1: Create new app.....	2
Step 2: Get API key and secret.....	3
Step 3: Subscribe for API.....	4
Step 4: Confirm Subscription .....	4
Try-it functionality.....	5
Step 1: Install your QWAC.....	5
Step 2: Get Token.....	6
Step 3: Invoke API .....	7
Postman .....	9
Step 1: Configure your QWAC.....	9
Step 2: Get Token.....	10
Step 3: Method and API .....	11
Step 4: Headers .....	11
Step 5: Body .....	12
Step 6: Invoke API .....	12

# Create app

## Step 1: Create new app

Developer portal -> Apps -> Create new app

### Create a new application

**Title \***

**Certificate**

```
---BEGIN CERTIFICATE---  
MIIOTCCBiGgAwIBAgIIYWhVuHHU6sowDQYJKoZIhvcNAQELBQAwwgYoxCzA  
JBgNV  
BAYTAKJHMRgwFgYDVQRhDA9OVFJCRy0yMDEyMzAOMjYxIDAeBgNVBAoM  
FOJPUKID
```

**Description**

**Application OAuth Redirect URL(s)**

✕

Add another item

✕ Cancel ✓ Submit

## Step 2: Get API key and secret

✔ Application created successfully. ✕

# API Key and Secret

The API Key and Secret have been generated for your application.

### Key

3527f1ef0fc6b1e7319667db8085a42f



Show

### Secret

dcfa8a6609138027cfd04d902af632dc



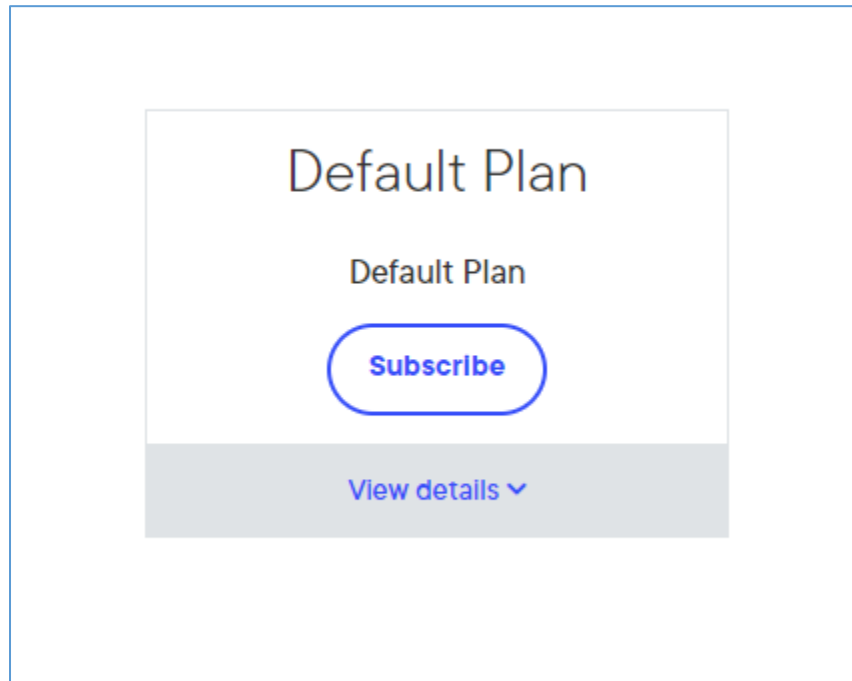
Show

The Secret will only be displayed here one time. Please copy your API Secret and keep it for your records.

Continue

### Step 3: Subscribe for API

Developer portal -> API Products -> API -> Subscribe



### Step 4: Confirm Subscription

## Confirm Subscription

A subscription will be created for the product, plan and application you selected.

**Product**

Account Information

**Plan**

Default Plan

**Application**

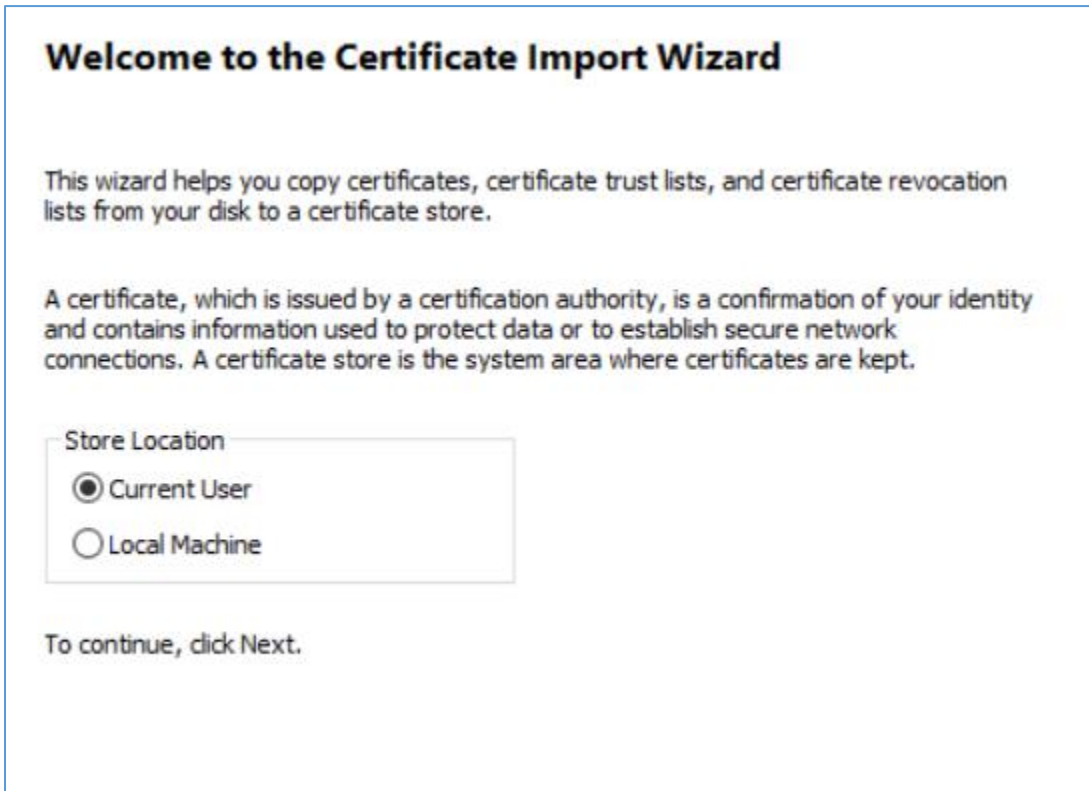
DemoApp

[Previous](#) [Next](#)

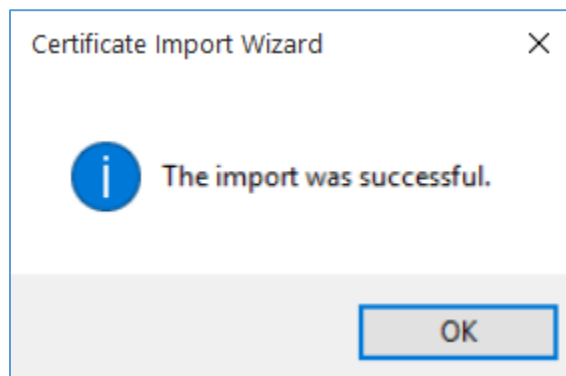
# Try-it functionality

## Step 1: Install your QWAC

You can use own QWAC or the test one provided in documentation section of the Developer portal (The password is "1234"). Follow the wizard for installation of the certificate.



You have to see a message like this:



## Step 2: Get Token

Developer portal -> API Product -> API -> Request -> Try-it

1. Select your application as Client ID
2. Paste your client secret
3. Select scope
4. Press Get Token
5. Choose your QWAC from the list of certificates
6. The field Access Token will be filled in

**Identification**

Client ID  
DemoApp

Client secret  
.....

---

**Authorization**

Scopes  
 consents

Token URL <https://psd2-apiuat.borica.bg/tbibank/psd2/oaauth/oauth2/token>

Access Token

.....

### Step 3: Invoke API

Developer portal -> API Product -> API -> Request -> Try-it

Fill in the required headers like X-Request-ID, Date and others. You can use the Generate options for each header if you like.

The screenshot shows the 'Parameters' section of a developer portal interface. A dropdown menu is set to 'Header'. Below it, there are several header configuration options:

- Accept**: A dropdown menu currently showing 'application/json'.
- Content-Type**: A dropdown menu currently showing 'application/json'.
- X-Request-ID\***: A text input field with a 'Generate' button to its right. Below the field is a tooltip: "ID of the request, unique to the call, as determined by the initiating party."
- PSU-ID\***: A text input field with a 'Generate' button to its right. Below the field is a tooltip: "Might be mandated in the ASPSP's documentation, if OAuth is not chosen as Pre-Step."
- PSU-ID-Type**: A text input field with a 'Generate' button to its right. Below the field is a tooltip: "Type of the PSU-ID, needed in scenarios where PSUs have several PSU-IDs as access possibility."

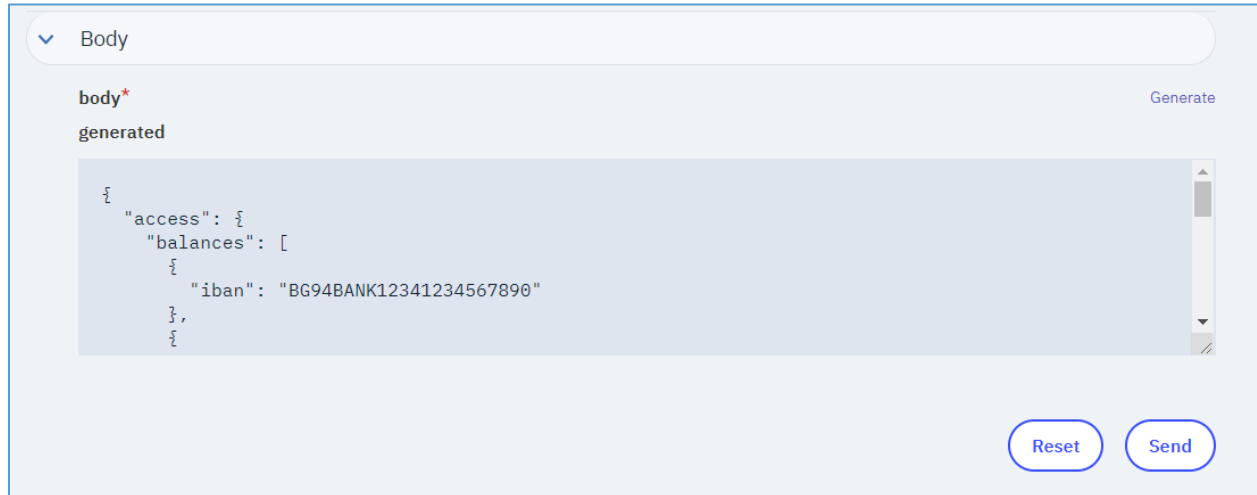
For X-Client-Certificate use must fill the PEM (BASE64) value of your QWAC without following:

- -----BEGIN certificate
- -----END certificate
- CR/LF

The screenshot shows the 'X-Client-Certificate\*' field in a developer portal interface. The field contains a long, multi-line base64-encoded string. The string starts with 'MIIInjCCBoagAwIBA...' and ends with '...'. The text is displayed in a light blue background with a vertical scrollbar on the right side.

Fill in the body if is required. You can use the Generate option if you like.

Press the Send button



The screenshot shows a REST client interface with a 'Body' tab selected. The body is labeled 'body\*' and 'generated'. A 'Generate' button is visible in the top right corner. The body content is a JSON object:

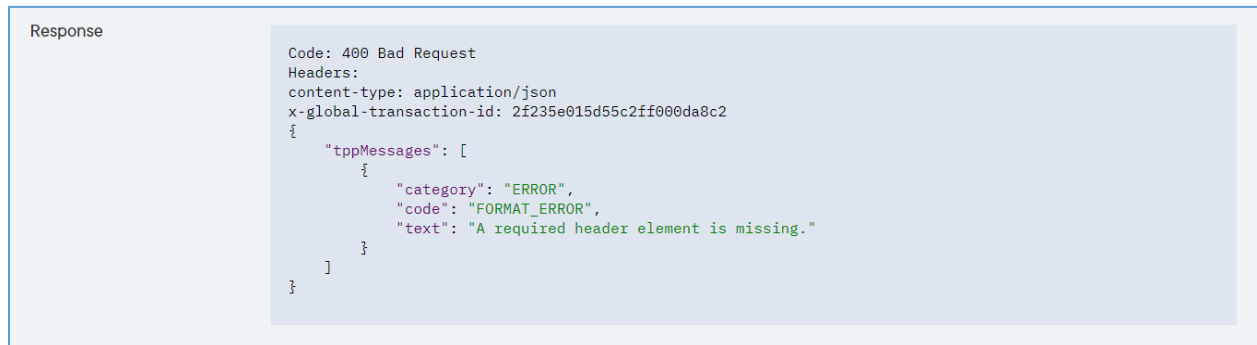
```
{
  "access": {
    "balances": [
      {
        "iban": "BG94BANK12341234567890"
      },
      {

```

At the bottom right, there are two buttons: 'Reset' and 'Send'.

Read the response.

**Important!!!** - Most of the browsers are not passing the Date header which is required because it is considered as unsafe. That is the reason for response bellow. Therefore we recommend to use a tool like Postman, SoapUI or other.



The screenshot shows a REST client interface with a 'Response' tab selected. The response content is:

```
Code: 400 Bad Request
Headers:
content-type: application/json
x-global-transaction-id: 2f235e015d55c2ff00da8c2
{
  "tppMessages": [
    {
      "category": "ERROR",
      "code": "FORMAT_ERROR",
      "text": "A required header element is missing."
    }
  ]
}
```

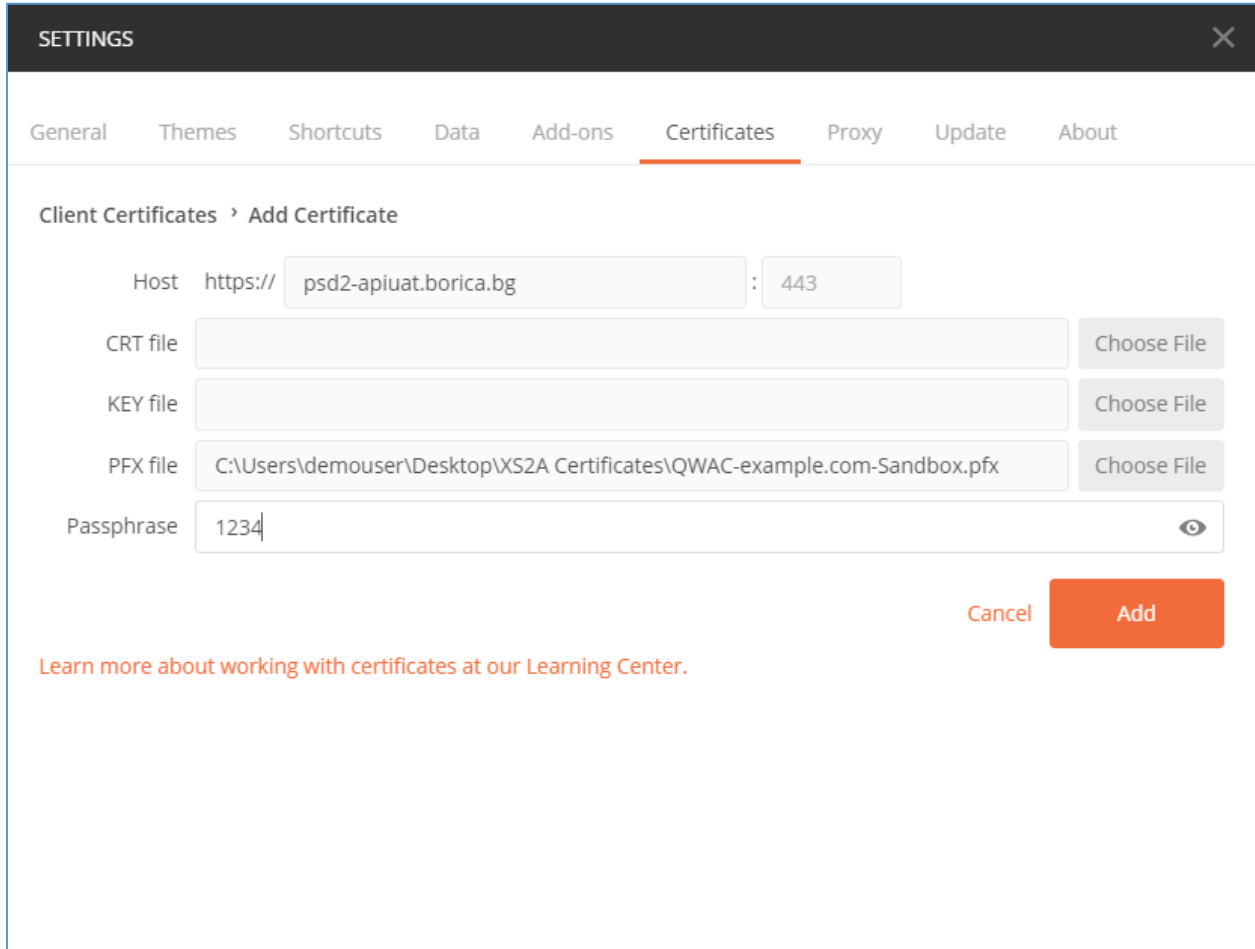


# Postman

## Step 1: Configure your QWAC

Postman -> Settings -> Certificates -> Add certificate

1. Fill in the host of APIs
2. Choose your QWAC certificate
3. Write the passphrase for your QWAC



The screenshot shows the 'SETTINGS' dialog box in Postman, with the 'Certificates' tab selected. The 'Client Certificates' section is active, showing the 'Add Certificate' form. The 'Host' field is filled with 'https://psd2-apiuat.borica.bg' and the port is '443'. The 'CRT file', 'KEY file', and 'PFX file' fields are empty, each with a 'Choose File' button. The 'Passphrase' field contains '1234'. There are 'Cancel' and 'Add' buttons at the bottom right. A link to the Learning Center is also visible.

SETTINGS

General Themes Shortcuts Data Add-ons Certificates Proxy Update About

Client Certificates › Add Certificate

Host https:// psd2-apiuat.borica.bg : 443

CRT file Choose File

KEY file Choose File

PFX file C:\Users\demouser\Desktop\XS2A Certificates\QWAC-example.com-Sandbox.pfx Choose File

Passphrase 1234

Cancel Add

[Learn more about working with certificates at our Learning Center.](#)

4. Press button Add

## Step 2: Get Token

Postman -> Authorization

1. Choose OAuth 2.0 as type
2. Press button "Get New Access Token"
3. Select Client Credentials for Grant Type
4. Fill the Access Token URL (described in Developer portal)
5. Fill Client ID & Client Secret of your application
6. Fill Scope (described in Developer portal)
7. Press button "Request Token"
8. After that, press button "Use Token"

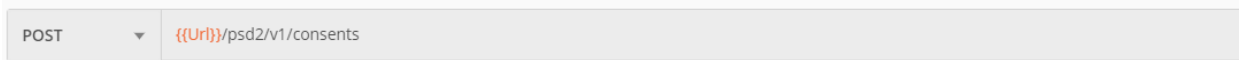
### GET NEW ACCESS TOKEN ✕

Token Name	<input type="text" value="My Token"/>
Grant Type	<input type="text" value="Client Credentials"/>
Access Token URL ⓘ	<input type="text" value="https://psd2-apiuat.borica.bg/tbibank/psd2/oaauther/oauth2/token"/>
Client ID ⓘ	<input type="text" value="76b9cb3a3f1526ad2452e4fd87a04e17"/>
Client Secret ⓘ	<input type="text" value="69011b3c6cc0e26c07bbd5c275a7843a"/>
Scope ⓘ	<input type="text" value="sandbox.consent"/>
Client Authentication	<input type="text" value="Send as Basic Auth header"/>

### Step 3: Method and API

Postman -> Headers

Fill in the method (POST, GET or DELETE) and URL of the API (described in Developer portal).

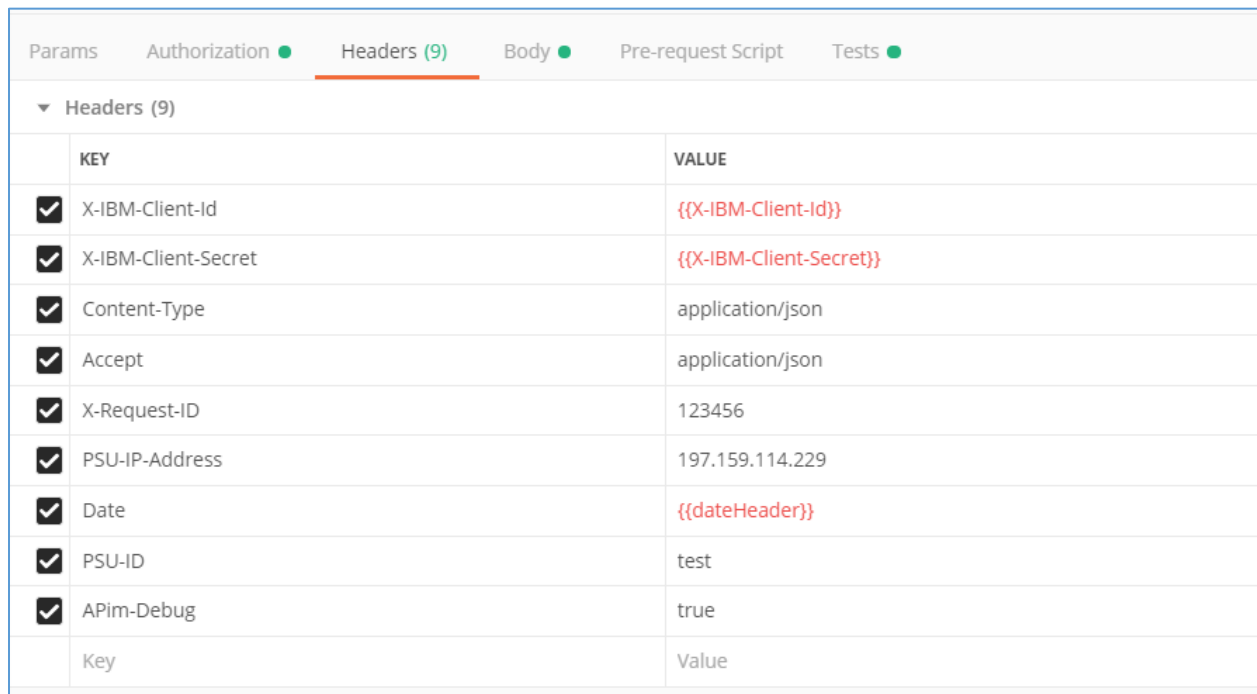


POST    {{Url}}/psd2/v1/consents

### Step 4: Headers

Postman -> Headers

Fill in the all the required headers except X-Client-Certificate. It is used only in Try-it functionality of Developer portal.



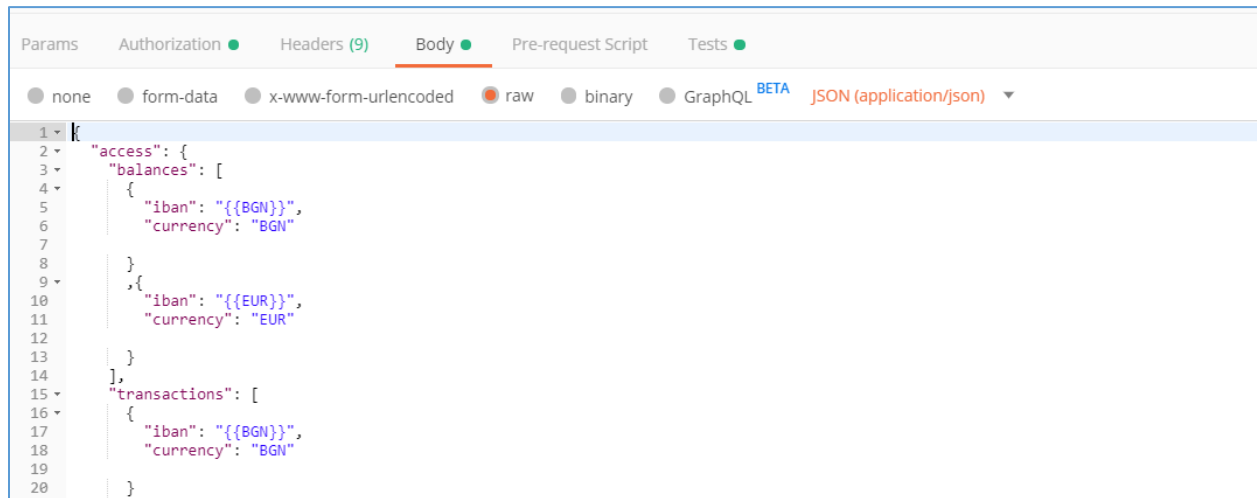
	KEY	VALUE
<input checked="" type="checkbox"/>	X-IBM-Client-Id	{{X-IBM-Client-Id}}
<input checked="" type="checkbox"/>	X-IBM-Client-Secret	{{X-IBM-Client-Secret}}
<input checked="" type="checkbox"/>	Content-Type	application/json
<input checked="" type="checkbox"/>	Accept	application/json
<input checked="" type="checkbox"/>	X-Request-ID	123456
<input checked="" type="checkbox"/>	PSU-IP-Address	197.159.114.229
<input checked="" type="checkbox"/>	Date	{{dateHeader}}
<input checked="" type="checkbox"/>	PSU-ID	test
<input checked="" type="checkbox"/>	APim-Debug	true
	Key	Value

## Step 5: Body

Postman -> Body

Fill in the body if is needed.

Select raw and JSON (application/json).



The screenshot shows the Postman interface with the 'Body' tab selected. The 'raw' radio button is chosen, and the content type is set to 'JSON (application/json)'. The body contains the following JSON:

```
1 | {
2 |   "access": {
3 |     "balances": [
4 |       {
5 |         "iban": "{{BGN}}",
6 |         "currency": "BGN"
7 |       },
8 |       {
9 |         "iban": "{{EUR}}",
10 |        "currency": "EUR"
11 |       }
12 |     ],
13 |     "transactions": [
14 |       {
15 |         "iban": "{{BGN}}",
16 |         "currency": "BGN"
17 |       }
18 |     ]
19 |   }
20 | }
```

## Step 6: Invoke API

Press button Send