

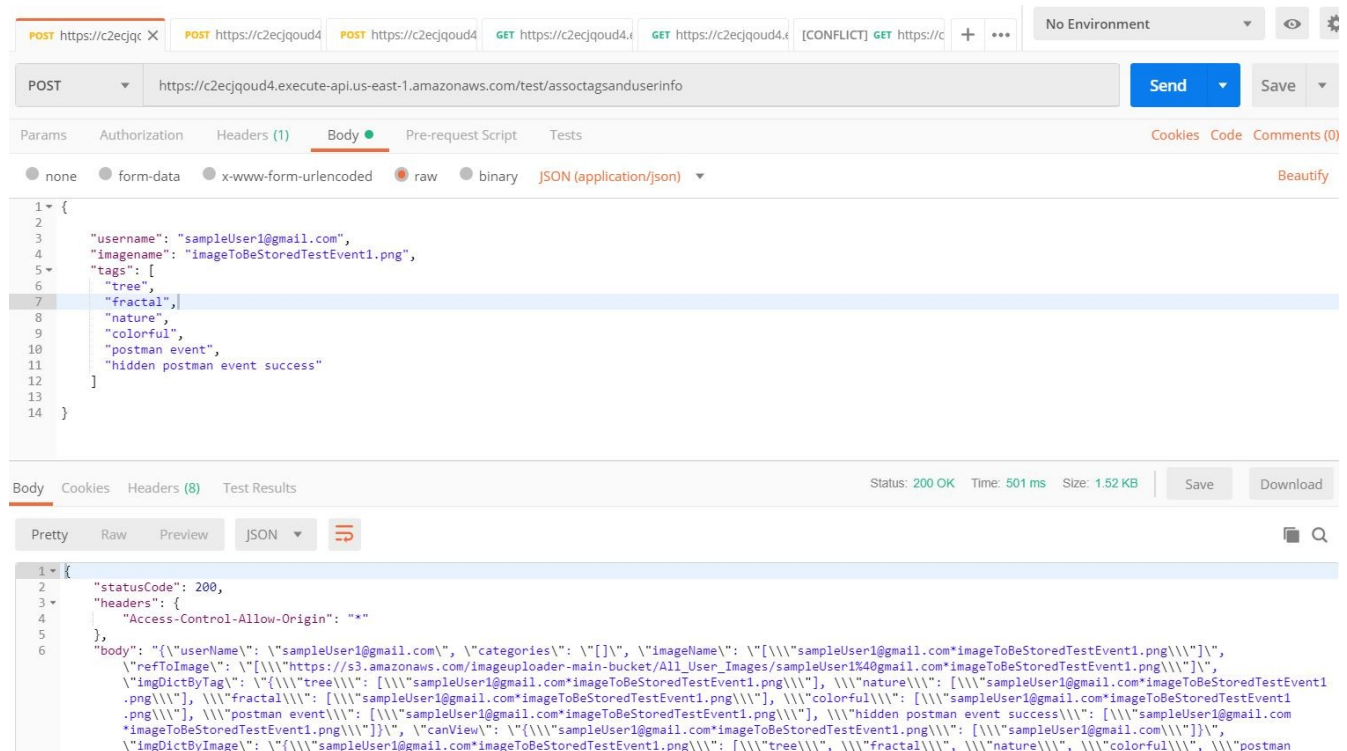
Updated 3/20/2019

POST:

<https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/assoctagsanduserinfo>

Description: Use this to associate a list of tags for a particular image of a user. This is basically adding an image, its tags for a user, and adds the user itself as the true owner of the image. It also updates the tags for an image, so whatever list of tags you pass for an image, all those tags will become the new tags for that image.

Screenshot of when user isn't yet in database (brand new user):

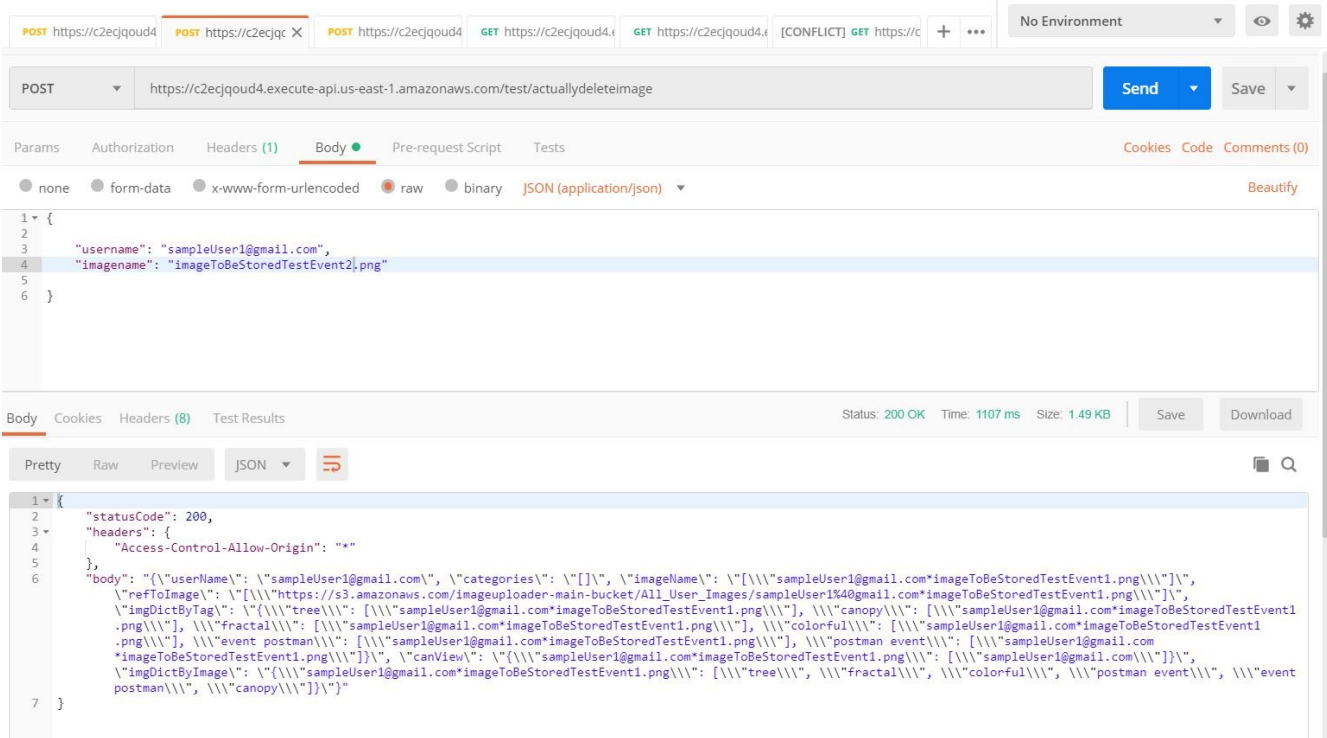


Screenshot of when you want to add a new image and tags:



Description: Deletes all images related to given user/owner.

Screenshot of info needed:

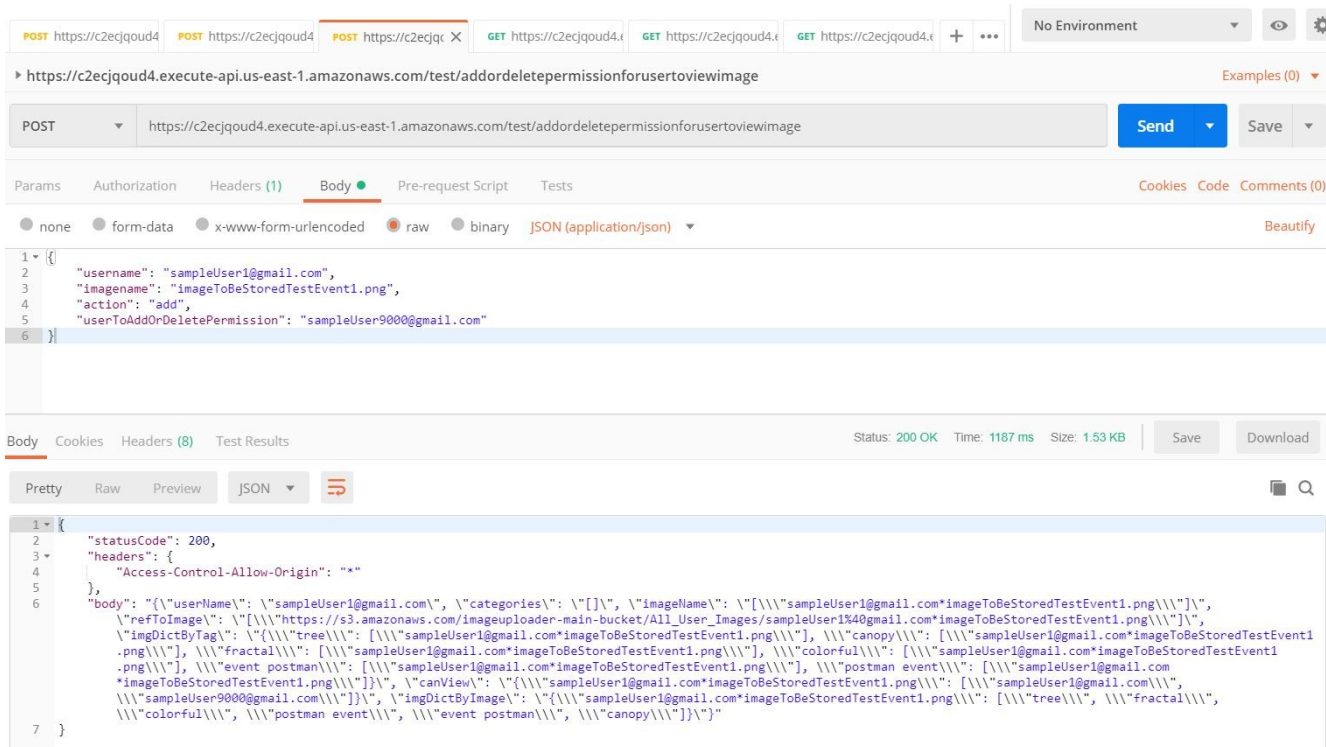


POST:

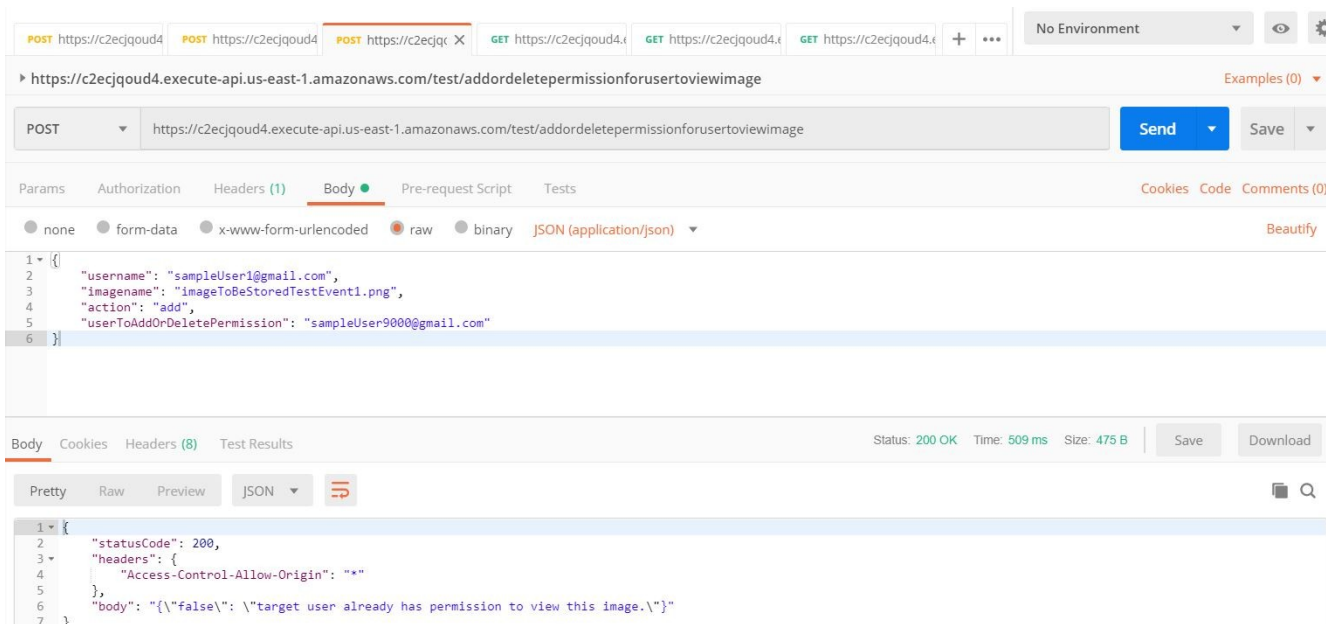
<https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/addordeletepermissionforusertoviewimage>

Description: Given an existing username and image name, can either "add" or "delete" the access authorization for userToAddOrDeletePermission (the non-owner).

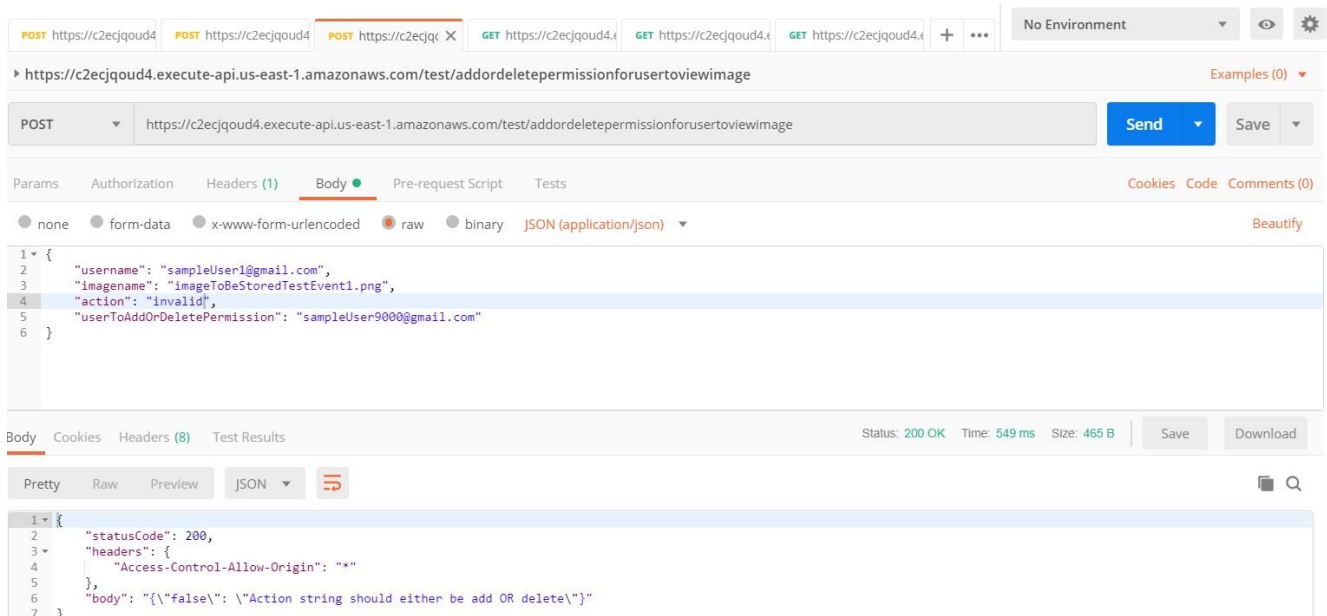
Screenshot of giving access to a user that isn't the owner ("username") of the image:



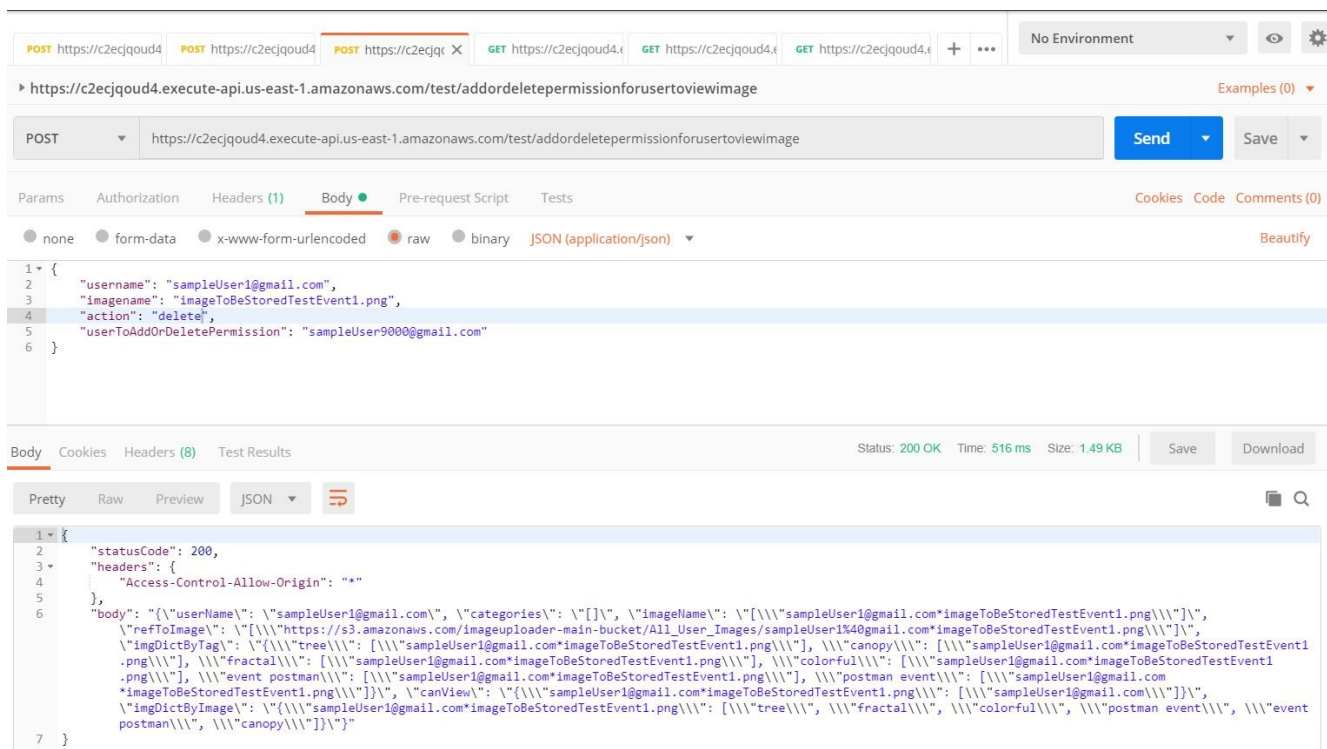
Screenshot of when attempting to give a non-owner (already has access) access to an image:



Screenshot of when attempting to pass an “action” that is neither "add" nor "delete":



Screenshot of revoking access to a user that isn't the owner ("username") of the image:



Screenshot of when attempting to revoke non-owner (had no access) access to an image:



POST https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/addordeletepermissionforusertoviewimage

Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```

1 {
2   "username": "sampleUser1@gmail.com",
3   "imagename": "imageToBeStoredTestEvent1.png",
4   "action": "delete",
5   "userToAddOrDeletePermission": "sampleUser9000@gmail.com"
6 }

```

Body Cookies Headers (8) Test Results Status: 200 OK Time: 449 ms Size: 470 B Save Download

Pretty Raw Preview JSON

```

1 {
2   "statusCode": 200,
3   "headers": {
4     "Access-Control-Allow-Origin": "*"
5   },
6   "body": "{\\\"false\\\": \\\"target user already doesn't exist for this image.\\\"}"
7 }

```

GET:

<https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/getuserimages?username=sampleUser1@gmail.com>

Description: Use this endpoint to get a dictionary of all images, each with a list of tags, for a particular user. Replace username value with another username.

GET https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/getuserimages?username=sampleUser1@gmail.com

Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE
<input checked="" type="checkbox"/> username	sampleUser1@gmail.com
Key	Value

Body Cookies Headers (8) Test Results Status: 200 OK Time: 1081 ms Size: 700 B Save Download

Pretty Raw Preview JSON

```

1 {
2   "statusCode": 200,
3   "headers": {
4     "Access-Control-Allow-Origin": "*"
5   },
6   "body": "{\\\"sampleUser1@gmail.com*imageToBeStoredTestEvent1.png\\\": [\\\"tree\\\", \\\"fractal\\\", \\\"colorful\\\", \\\"postman event\\\", \\\"event postman\\\", \\\"canopy\\\"], \\\"sampleUser1@gmail.com*imageToBeStoredTestEvent2.png\\\": [\\\"tree\\\", \\\"fractal\\\", \\\"nature\\\", \\\"colorful\\\", \\\"postman event\\\", \\\"2nd postman event\\\"]}"
7 }

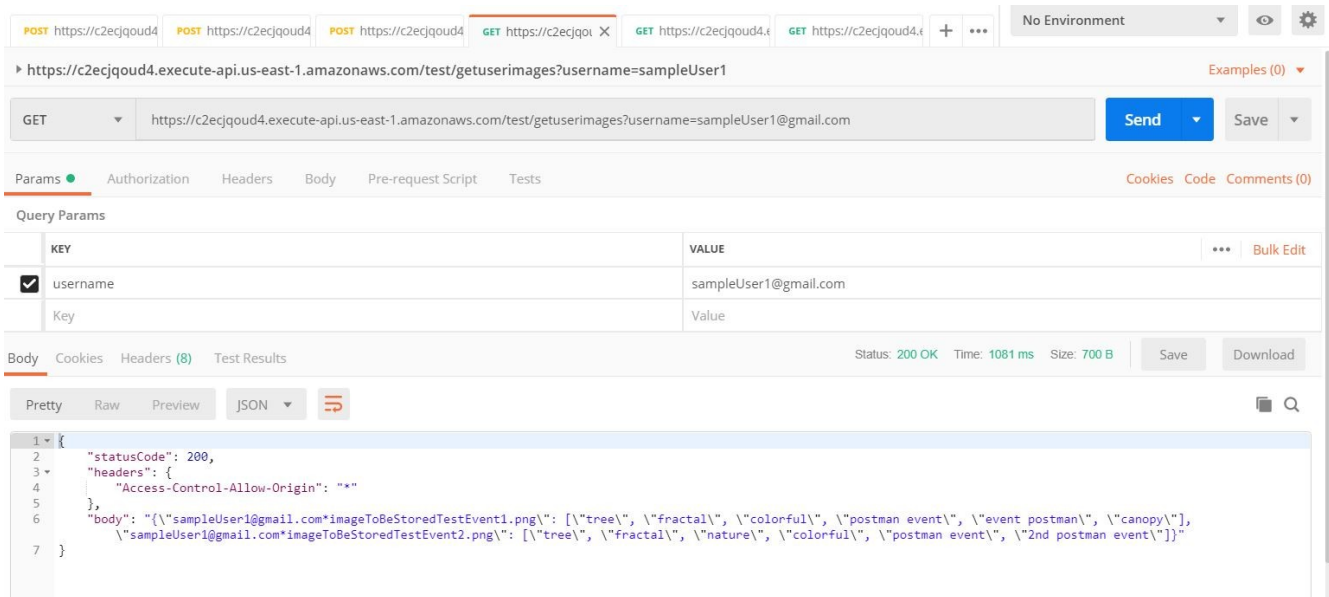
```

GET:

<https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/getuserimages?username=sampleUser1@gmail.com>

Description: Given an existing username, returns a dictionary of all images associated with the passed in username and the list of tags associated with each respective image.

Screenshot:

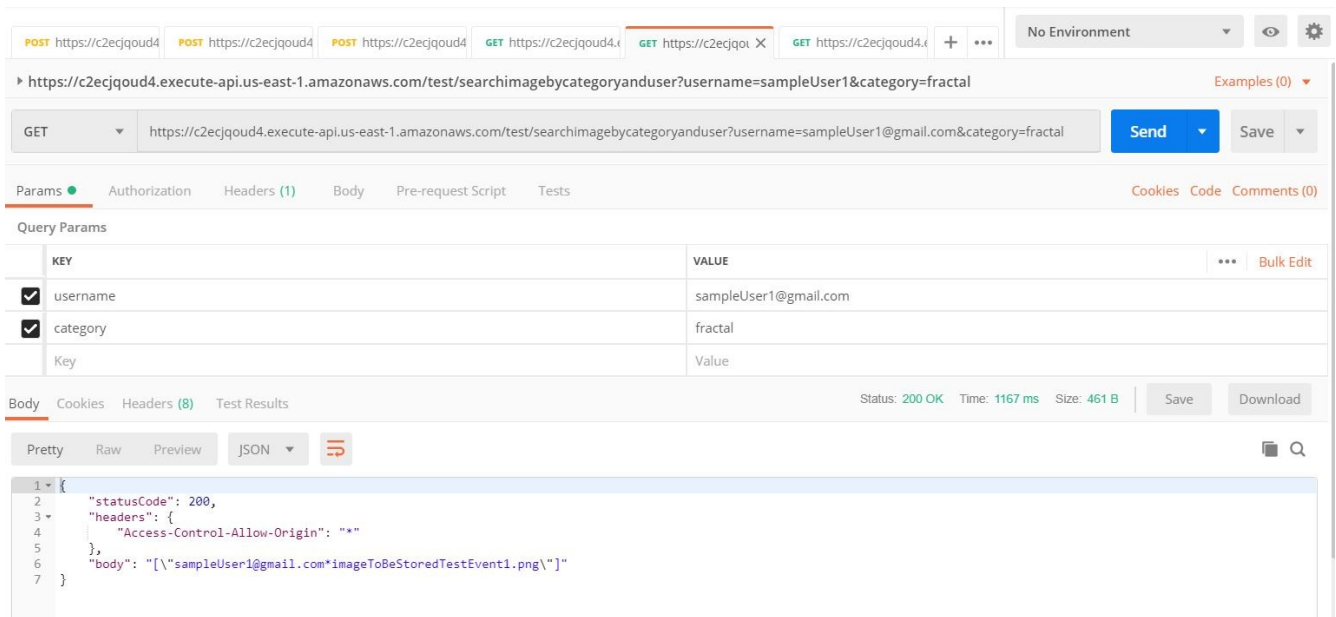


GET:

<https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/searchimagebycategoryanduser?username=sampleUser1@gmail.com&category=fractal>

Description: Given an existing username and category (or tag), returns the list of images (must extract image name from each corresponding element in the list) associated with the passed in tag.

Screenshot:



GET:

<https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/getuserswithpermissiontoviewimages?username=sampleUser1@gmail.com>

Description: Given an existing username, returns a dictionary of all images (image name must be extracted first with some sort of string split() function where the delimiter is the '\*' character), each with a list of usernames that are allowed to access their respective images.

Screenshot:



POST https://c2ecjqoud4

POST https://c2ecjqoud4

POST https://c2ecjqoud4

GET https://c2ecjqoud4

GET https://c2ecjqoud4

GET https://c2ecjqoud4 X

+

...

No Environment

▶ https://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/getuserswithpermissiontoviewimages?username=sampleUser1@gmail.com

Examples (0)

GEThttps://c2ecjqoud4.execute-api.us-east-1.amazonaws.com/test/getuserswithpermissiontoviewimages?username=sampleUser1@gmail.comSendSave

ParamsAuthorizationHeadersBodyPre-request ScriptTestsCookiesCodeComments (0)

Query Params

	KEY	VALUE	...	Bulk Edit
<input checked="" type="checkbox"/>	username	sampleUser1@gmail.com		
	Key	Value		

BodyCookiesHeaders (8)Test ResultsStatus: 200 OKTime: 1084 msSize: 490 BSaveDownload

PrettyRawPreviewJSON

1 {

2   "statusCode": 200,

3   "headers": {

4     "Access-Control-Allow-Origin": "\*"

5   },

6   "body": "{\\"sampleUser1@gmail.com\*imageToBeStoredTestEvent1.png\\": [\\"sampleUser1@gmail.com\\"]}"

7 }