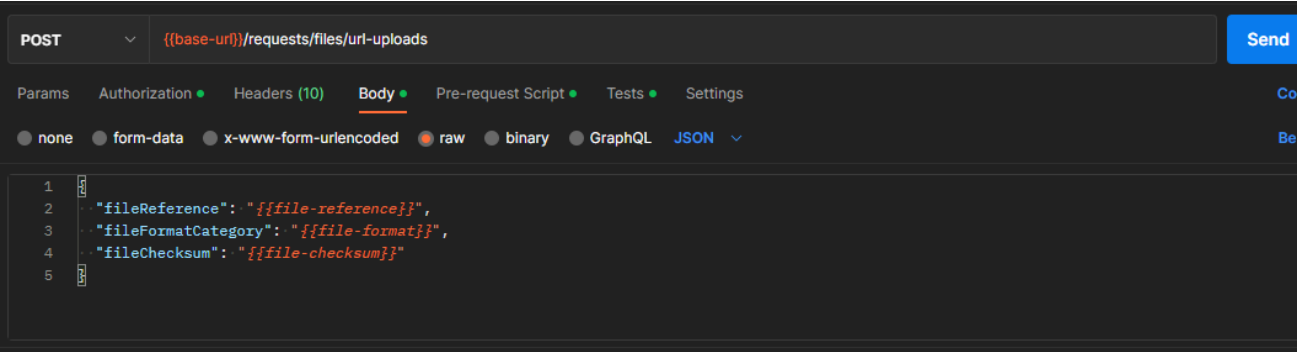


b.1.3.2.- obtain fileId and a presigned URL to upload a big size document file (POST)

[[Request configuration](#)] [[a.- Headers](#)] [[b.- Body](#)] [[c.- Pre-request script](#)] [[d.- Tests script](#)] [[Expected response](#)] [[e.- Body](#)] [[f.- Headers](#)] [[g.- Test results](#)]

The Postman request:



Request configuration

a.- Headers

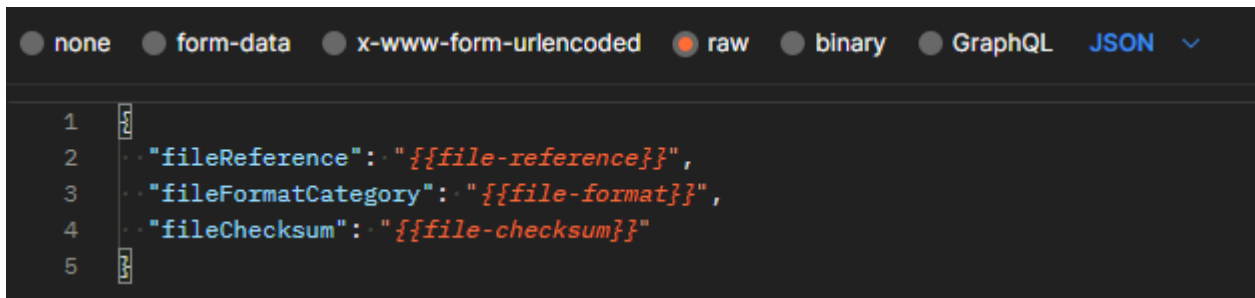
Standard headers, nothing to update here.

Headers Hide auto-generated headers			
	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	Authorization	Bearer eyJ0eXAiOiJKV1QiLCJraWQiOiJmVWRmbEJSa3c5...	
<input checked="" type="checkbox"/>	Content-Type	application/json	
<input checked="" type="checkbox"/>	Content-Length	<calculated when request is sent>	
<input checked="" type="checkbox"/>	Host	<calculated when request is sent>	
<input checked="" type="checkbox"/>	User-Agent	PostmanRuntime/7.31.0	
<input checked="" type="checkbox"/>	Accept	/*/*	
<input checked="" type="checkbox"/>	Accept-Encoding	gzip, deflate, br	
<input checked="" type="checkbox"/>	Connection	keep-alive	
<input checked="" type="checkbox"/>	Content-Type	application/json	
<input checked="" type="checkbox"/>	Accept	application/json	

b.- Body

Values that are used for this demo (selected as Postman type RAW):

Key	Value	Notes
fileReference	"{{file-reference}}"	optional reference used by the service consumer to identify the file
fileFormatCategory	"{{file-format}}"	Can be pdf,zip,xml (for the current test b.1.3.- UPdate registration it will by zip)
fileChecksum	"{{file-checksum}}"	mandatory checksum of the file to upload for validation purpose. Please check here to see how to generate it, then update the Postman environment



c.- Pre-request script

This script automatically generates a general reference and updates the Postman environment with it.

setting the file reference (only for testing)

```
// setting the fileReference

var currentTimestamp = "file_reference_created_on_" + (Math.floor(Date.now() / 1000) );

pm.environment.set('file-reference',currentTimestamp );

console.log("bearer=",pm.environment.get('access-token'))
```

d.- Tests script

The following script is executed after the request is sent, to evaluate the response and also to set up environment variables (if they are needed for following requests)

```
// following test will verify the expected response code for this request

pm.test("Status code is 201", function () {
  pm.response.to.have.status(201);
});

// this tests if the file-id is received and then updates the postman environment with it
pm.test("file-id received", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.fileId).not.eql(undefined);
  if (jsonData.fileId !== undefined) {
    console.log("file-id:"+jsonData.fileId)
    pm.environment.set("file-id", jsonData.fileId);
  }
});

// this tests if the upload URL is received and then updates the postman environment with it
pm.test("upload URL received", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.fileUploadUrl).not.eql(undefined);
  if (jsonData.fileUploadUrl !== undefined) {
    console.log("upload-url:"+jsonData.fileUploadUrl)
    pm.environment.set("upload-url", jsonData.fileUploadUrl);
  }
});
```

Expected response

e.- Body

We can see there the normal JSON response with the fileId and fileUploadUrl that is being provided by the system for the upload of the file we want to register in DAS

```
Body Cookies Headers (15) Test Results (3/3) Status: 201 Created Time: 20.48 s Size: 1.84 KB Save Response

Pretty Raw Preview Visualize JSON

1
2 "fileId": "5cb1838f-b0f5-401e-b27c-095e4aea0730",
3 "fileUploadUrl": "https://das-upload-eu-central-1-452010797180.s3.eu-central-1.amazonaws.com/das-ib/5cb1838f-b0f5-401e-b27c-095e4aea0730?Content-Type=application%2Fzip&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ASIAWSPPRNB6NY4C7H82%2F20230222%2Ffe-central-1%2Fs3%2Faws4_request&X-Amz-Date=20230222T112633Z&X-Amz-Expires=900&X-Amz-Security-Token=FwoGZXIvYXZzEE0aDAQw%2F0KBH%2FLUrK29wyLAARgZDVw36iwSz%2F%2BjkQMhW3qIQKYsBs9hWPWWU%2B1Vud%2Fe43VqJxhoe6YK50ocB70TwisfQrxKlzkJfQ285I%2B4lRabugKIfku%2FUAWFmVqxIJJMnIQxNc4MpxXJh5o8WjgdAG2Wpocj0t%2B7FBi0A0apD3%2FWXqjEgz6l0%2Fw1yZx1bawQzqKXde32Baqk9YmcxfZFeyQ38FRHh3uh2FI21LduHIPYmS5THitkNjN%2FAuQk98nSMwxv1R%2BJ7Kug5u0m2Sjp99efBjItD5zuy81pbe%2FUE3yVGuiATds0b9QuftyQEdLya1sIS3VkUPVF%2BxM%2FaB20&X-Amz-Signature=961f7d46e31798c974be7baf43874551d768e324bb446607d6061ce286231984&X-Amz-SignedHeaders=host",
4 "fileReference": "file_reference_created_on_1677065173",
5 "fileFormatCategory": "zip",
6 "fileChecksum": "c08bf963418786c2ee959d3ba97d7c71b34f7a35d058198c02af4d5f19f62fb4",
7 "fileContentType": "application/zip",
8 "fileUrlExpiresIn": "2023-02-22T11:41:33.153Z"
9
```

f.- Headers

Normal headers.

KEY	VALUE
Content-Type	application/json
Content-Length	1158
Connection	keep-alive
Date	Wed, 22 Feb 2023 11:26:34 GMT
x-amzn-Requestid	e37b3217-6cf9-421f-a6a0-bbd9d760eff2
Access-Control-Allow-Origin	*
x-amz-apigw-id	AvRJsEktIIAFVeA=
Access-Control-Allow-Methods	*
location	https://s2w1imptql.execute-api.eu-central-1.amazonaws.com/requests/files/url-uploads?l...
Access-Control-Expose-Headers	location
X-Amzn-Trace-Id	Root=1-63f5fbd7-66ac8f6b75aeddc7195b0056
X-Cache	Miss from cloudfront
Via	1.1 ccabfbceff64477665e33f03003a399c.cloudfront.net (CloudFront)
X-Amz-Cf-Pop	IAD55-P5
X-Amz-Cf-Id	RZ48eVFGx9E6nR_wJD52dBYFP0Ahr5vZslbOI8JUX-2HokzIV-42Pg==

g.- Test results

This depends on the test we set up in the test script

Body Cookies Headers (15) **Test Results (3/3)**

All Passed Skipped Failed

PASS

Status code is 201

PASS

file-id received

PASS

upload URL received