Dear Artru.

API url that I referred from:

<https://developers.google.com/youtube/v3/docs/channels/list?apix_params=%7B%22part%22%3A%5B%22snippet%2CcontentDetails%2Cstatistics%22%5D%2C%22id%22%3A%5B%22UC_x5XG1OV2P6uZZ5FSM9Ttw%22%5D%7D>

list (by channel ID)

*This example retrieves channel data for the Google Developers YouTube channel. It uses the id request parameter to identify the channel by its YouTube channel ID.*

- Javascript

<script src="https://apis.google.com/js/api.js"></script>

<script>

/\*\*

\* Sample JavaScript code for youtube.channels.list

\* See instructions for running APIs Explorer code samples locally:

\* https://developers.google.com/explorer-help/guides/code\_samples#javascript

\*/

function authenticate() {

return gapi.auth2.getAuthInstance()

.signIn({scope: "https://www.googleapis.com/auth/youtube.readonly"})

.then(function() { console.log("Sign-in successful"); },

function(err) { console.error("Error signing in", err); });

}

function loadClient() {

gapi.client.setApiKey("YOUR\_API\_KEY");

return gapi.client.load("https://www.googleapis.com/discovery/v1/apis/youtube/v3/rest")

.then(function() { console.log("GAPI client loaded for API"); },

function(err) { console.error("Error loading GAPI client for API", err); });

}

// Make sure the client is loaded and sign-in is complete before calling this method.

function execute() {

return gapi.client.youtube.channels.list({

"part": [

"snippet,contentDetails,statistics"

],

"id": [

"UC\_x5XG1OV2P6uZZ5FSM9Ttw"

]

})

.then(function(response) {

// Handle the results here (response.result has the parsed body).

console.log("Response", response);

},

function(err) { console.error("Execute error", err); });

}

gapi.load("client:auth2", function() {

gapi.auth2.init({client\_id: "YOUR\_CLIENT\_ID"});

});

</script>

<button onclick="authenticate().then(loadClient)">authorize and load</button>

<button onclick="execute()">execute</button>

- PHP

<?php

/\*\*

\* Sample PHP code for youtube.channels.list

\* See instructions for running these code samples locally:

\* https://developers.google.com/explorer-help/guides/code\_samples#php

\*/

if (!file\_exists(\_\_DIR\_\_ . '/vendor/autoload.php')) {

throw new Exception(sprintf('Please run "composer require google/apiclient:~2.0" in "%s"', \_\_DIR\_\_));

}

require\_once \_\_DIR\_\_ . '/vendor/autoload.php';

$client = new Google\_Client();

$client->setApplicationName('API code samples');

$client->setScopes([

'https://www.googleapis.com/auth/youtube.readonly',

]);

// TODO: For this request to work, you must replace

// "YOUR\_CLIENT\_SECRET\_FILE.json" with a pointer to your

// client\_secret.json file. For more information, see

// https://cloud.google.com/iam/docs/creating-managing-service-account-keys

$client->setAuthConfig('YOUR\_CLIENT\_SECRET\_FILE.json');

$client->setAccessType('offline');

// Request authorization from the user.

$authUrl = $client->createAuthUrl();

printf("Open this link in your browser:\n%s\n", $authUrl);

print('Enter verification code: ');

$authCode = trim(fgets(STDIN));

// Exchange authorization code for an access token.

$accessToken = $client->fetchAccessTokenWithAuthCode($authCode);

$client->setAccessToken($accessToken);

// Define service object for making API requests.

$service = new Google\_Service\_YouTube($client);

$queryParams = [

'id' => 'UC\_x5XG1OV2P6uZZ5FSM9Ttw'

];

$response = $service->channels->listChannels('snippet,contentDetails,statistics', $queryParams);

print\_r($response);

list (by YouTube username)

*This example retrieves channel data for the GoogleDevelopers YouTube channel. It uses the forUsername request parameter to identify the channel by its YouTube username.*

-Javascript

<script src="https://apis.google.com/js/api.js"></script>

<script>

/\*\*

\* Sample JavaScript code for youtube.channels.list

\* See instructions for running APIs Explorer code samples locally:

\* https://developers.google.com/explorer-help/guides/code\_samples#javascript

\*/

function authenticate() {

return gapi.auth2.getAuthInstance()

.signIn({scope: "https://www.googleapis.com/auth/youtube.readonly"})

.then(function() { console.log("Sign-in successful"); },

function(err) { console.error("Error signing in", err); });

}

function loadClient() {

gapi.client.setApiKey("YOUR\_API\_KEY");

return gapi.client.load("https://www.googleapis.com/discovery/v1/apis/youtube/v3/rest")

.then(function() { console.log("GAPI client loaded for API"); },

function(err) { console.error("Error loading GAPI client for API", err); });

}

// Make sure the client is loaded and sign-in is complete before calling this method.

function execute() {

return gapi.client.youtube.channels.list({

"part": [

"snippet,contentDetails,statistics"

],

"forUsername": "GoogleDevelopers"

})

.then(function(response) {

// Handle the results here (response.result has the parsed body).

console.log("Response", response);

},

function(err) { console.error("Execute error", err); });

}

gapi.load("client:auth2", function() {

gapi.auth2.init({client\_id: "YOUR\_CLIENT\_ID"});

});

</script>

<button onclick="authenticate().then(loadClient)">authorize and load</button>

<button onclick="execute()">execute</button>

- PHP

<?php

/\*\*

\* Sample PHP code for youtube.channels.list

\* See instructions for running these code samples locally:

\* https://developers.google.com/explorer-help/guides/code\_samples#php

\*/

if (!file\_exists(\_\_DIR\_\_ . '/vendor/autoload.php')) {

throw new Exception(sprintf('Please run "composer require google/apiclient:~2.0" in "%s"', \_\_DIR\_\_));

}

require\_once \_\_DIR\_\_ . '/vendor/autoload.php';

$client = new Google\_Client();

$client->setApplicationName('API code samples');

$client->setScopes([

'https://www.googleapis.com/auth/youtube.readonly',

]);

// TODO: For this request to work, you must replace

// "YOUR\_CLIENT\_SECRET\_FILE.json" with a pointer to your

// client\_secret.json file. For more information, see

// https://cloud.google.com/iam/docs/creating-managing-service-account-keys

$client->setAuthConfig('YOUR\_CLIENT\_SECRET\_FILE.json');

$client->setAccessType('offline');

// Request authorization from the user.

$authUrl = $client->createAuthUrl();

printf("Open this link in your browser:\n%s\n", $authUrl);

print('Enter verification code: ');

$authCode = trim(fgets(STDIN));

// Exchange authorization code for an access token.

$accessToken = $client->fetchAccessTokenWithAuthCode($authCode);

$client->setAccessToken($accessToken);

// Define service object for making API requests.

$service = new Google\_Service\_YouTube($client);

$queryParams = [

'forUsername' => 'GoogleDevelopers'

];

$response = $service->channels->listChannels('snippet,contentDetails,statistics', $queryParams);

print\_r($response);

list (my channel)

*This example retrieves the channel data for the authorized user's YouTube channel. It uses the mine request parameter to indicate that the API should only return channels owned by the user authorizing the request.*

- Javascript

<script src="https://apis.google.com/js/api.js"></script>

<script>

/\*\*

\* Sample JavaScript code for youtube.channels.list

\* See instructions for running APIs Explorer code samples locally:

\* https://developers.google.com/explorer-help/guides/code\_samples#javascript

\*/

function authenticate() {

return gapi.auth2.getAuthInstance()

.signIn({scope: "https://www.googleapis.com/auth/youtube.readonly"})

.then(function() { console.log("Sign-in successful"); },

function(err) { console.error("Error signing in", err); });

}

function loadClient() {

gapi.client.setApiKey("YOUR\_API\_KEY");

return gapi.client.load("https://www.googleapis.com/discovery/v1/apis/youtube/v3/rest")

.then(function() { console.log("GAPI client loaded for API"); },

function(err) { console.error("Error loading GAPI client for API", err); });

}

// Make sure the client is loaded and sign-in is complete before calling this method.

function execute() {

return gapi.client.youtube.channels.list({

"part": [

"snippet,contentDetails,statistics"

],

"mine": true

})

.then(function(response) {

// Handle the results here (response.result has the parsed body).

console.log("Response", response);

},

function(err) { console.error("Execute error", err); });

}

gapi.load("client:auth2", function() {

gapi.auth2.init({client\_id: "YOUR\_CLIENT\_ID"});

});

</script>

<button onclick="authenticate().then(loadClient)">authorize and load</button>

<button onclick="execute()">execute</button>

- PHP

<?php

/\*\*

\* Sample PHP code for youtube.channels.list

\* See instructions for running these code samples locally:

\* https://developers.google.com/explorer-help/guides/code\_samples#php

\*/

if (!file\_exists(\_\_DIR\_\_ . '/vendor/autoload.php')) {

throw new Exception(sprintf('Please run "composer require google/apiclient:~2.0" in "%s"', \_\_DIR\_\_));

}

require\_once \_\_DIR\_\_ . '/vendor/autoload.php';

$client = new Google\_Client();

$client->setApplicationName('API code samples');

$client->setScopes([

'https://www.googleapis.com/auth/youtube.readonly',

]);

// TODO: For this request to work, you must replace

// "YOUR\_CLIENT\_SECRET\_FILE.json" with a pointer to your

// client\_secret.json file. For more information, see

// https://cloud.google.com/iam/docs/creating-managing-service-account-keys

$client->setAuthConfig('YOUR\_CLIENT\_SECRET\_FILE.json');

$client->setAccessType('offline');

// Request authorization from the user.

$authUrl = $client->createAuthUrl();

printf("Open this link in your browser:\n%s\n", $authUrl);

print('Enter verification code: ');

$authCode = trim(fgets(STDIN));

// Exchange authorization code for an access token.

$accessToken = $client->fetchAccessTokenWithAuthCode($authCode);

$client->setAccessToken($accessToken);

// Define service object for making API requests.

$service = new Google\_Service\_YouTube($client);

$queryParams = [

'mine' => true

];

$response = $service->channels->listChannels('snippet,contentDetails,statistics', $queryParams);

print\_r($response);

I hope my reference would be a help for you.

Have a nice evening.

Bozo.