

# Step by Step Guide to get the number of likes of YouTube video using Google API

This tutorial demonstrates how to use YouTube API v3 to get the number of likes of videos from YouTube. YouTube offers various powerful ways to play around with videos. You can create awesome applications using the API and also can embed videos from YouTube. It has client libraries in various programming languages to do that. In this application, we will use Google API's .Net Client Library.

Assuming the OAuth Key json file of user is already downloaded from Google developer console. The Youtube API V3 has Videos: List method which returns a list of videos that match the API request parameters. The HTTP request made to API is GET <https://www.googleapis.com/youtube/v3/videos> . The required parameters - part and filter id is used to get the statistics of the video - like , dislike , views etc.

The part parameter specifies a comma-separated list of one or more video resource properties that the API response will include.

The id parameter specifies a comma-separated list of the YouTube video ID(s) for the resource(s) that are being retrieved. In a video resource, the id property specifies the video's ID.

There is no request body when calling the method. If successful , this method returns a response body with following structure, having statistics are video Resource.

```
{
  "kind": "youtube#videoListResponse",
  "etag": etag,
  "nextPageToken": string,
  "prevPageToken": string,
  "pageInfo": {
    "totalResults": integer,
    "resultsPerPage": integer
  },
  "items": [
    video Resource
  ]
}
```

```
"statistics": {
  "viewCount": unsigned long,
  "likeCount": unsigned long,
  "dislikeCount": unsigned long,
  "favoriteCount": unsigned long,
  "commentCount": unsigned long
},
```

To validate the request we can verify the below youtube video using the Google's [APIs Explorer](#) for the [YouTube Data API](#) . The page shows the list of methods provided to get details of youtube videos, the one we are interested in is Video.list

<https://developers.google.com/apis-explorer/#p/youtube/v3/youtube.videos.list>

Now enter the parameters part= id, statistics and id=cBFq\_6Zj2aA , video id for below Youtube video. The video ID is captured from the youtube url . We will validate the no. of likes of the video is 26K or not.





## APIs Explorer



### Services

Learn more about using the YouTube Data API by reading the [documentation](#).

### All Versions

[Services](#) > [YouTube Data API v3](#) > [youtube.videos.list](#)

### Request History

**part**

The part parameter specifies a comma-separated list of properties of the video resource that you want to include in the API response. For example, if you set `part=snippet`, the API response will contain only the snippet property of the video resource.

chart

The chart parameter identifies the chart that you want to use to display the video's statistics.

hl

The hl parameter instructs the API to retrieve video statistics in the specified language. The parameter value must be a language code supported by the API. If the video resource details are available in that language, the API response will contain localized details. If the localized details are not available, the API response will contain the default details.

id

The id parameter specifies a comma-separated list of the IDs of the video resources that you want to retrieve. If the list is empty, the API will return an error.

locale

DEPRECATED (string)

maxHeight

The maxHeight parameter specifies a maximum height in pixels for the video player. The parameter value must be a positive integer. The parameter is used in conjunction with the maxWidth parameter to ensure that the video player does not exceed the specified dimensions.

maxResults

The maxResults parameter specifies the maximum number of video resources to return in the response. The parameter value must be a positive integer. The parameter is used in conjunction with the myRating parameter to filter the results.

maxWidth

The maxWidth parameter specifies a maximum width in pixels for the video player. The parameter value must be a positive integer. The parameter is used in conjunction with the maxHeight parameter to ensure that the video player does not exceed the specified dimensions.

myRating

Set this parameter's value to like or dislike the video.

onBehalfOfContentOwner

Note: This parameter is intended exclusively for use with the YouTube Data API v3. It is not supported by the YouTube Data API v2. The parameter value must be the OAuth 2.0 access token of the content owner's account. This parameter is intended to allow content owners to authenticate on behalf of their channels. The parameter value must be the OAuth 2.0 access token of the content owner. (string)

pageToken

The pageToken parameter identifies a specific page of the results. The parameter value must be the pageToken property of the previous page's response. The parameter is used in conjunction with the myRating and chart parameters, but only when the myRating parameter is set.

regionCode

The regionCode parameter instructs the API to return video resources that are available in the specified region. The parameter value must be a two-letter ISO 3166-1 alpha-2 country code. The parameter is used in conjunction with the chart parameter. The parameter is not supported by the YouTube Data API v2.

videoCategoryId

The videoCategoryId parameter identifies the video category for the video resources. The parameter value must be the videoCategoryId property of the video resource. The parameter is used in conjunction with the chart parameter. By default, the API returns video resources from the 'For You' category.

fields

Selector specifying which fields to include in the response. For more information, see [Use fields editor](#).

**bold red** = required

Authorize and execute

[Execute without OAuth](#)

Select the OAuth scopes and select the google account to authorise.

### Select OAuth 2.0 scopes:

Scopes are used to grant an application different levels of access to data on behalf of the end user. Each API may declare one or more scopes. [Learn more about OAuth 2.0](#)

YouTube Data API declares the following scopes. Select which ones you want to grant to APIs Explorer.

- ☒ <https://www.googleapis.com/auth/youtube>  
Manage your YouTube account
- ☒ <https://www.googleapis.com/auth/youtube.force-ssl>  
See, edit, and permanently delete your YouTube videos, ratings, comments and captions
- ☒ <https://www.googleapis.com/auth/youtube.readonly>  
View your YouTube account
- ☐ <https://www.googleapis.com/auth/youtube.upload>  
Manage your YouTube videos
- ☒ <https://www.googleapis.com/auth/youtubepartner>  
View and manage your assets and associated content on YouTube
- ☐ <https://www.googleapis.com/auth/youtubepartner-channel-audit>  
View private information of your YouTube channel relevant during the audit process with a YouTube partner


Add additional scopes (optional):


[Authorize and execute](#) [Cancel](#)


### Sign in with Google

## Choose an account

to continue to [Google APIs Explorer](#)

**Mala Karmakar**  
karmakarmala@gmail.com

**Chandan Choudhury**  
chandanindira@gmail.com

 [Use another account](#)

To continue, Google will share your name, email address, language preference, and profile picture with Google APIs Explorer.

We can check the no. of likes is 26640 which is same as in the youtube video page.

### Request

```
GET https://www.googleapis.com/youtube/v3/videos?part=id%2C++statistics&id=cBFq_6Zj2aA&key={YOUR_API_KEY}
```

### Response

```
200
- Show headers -

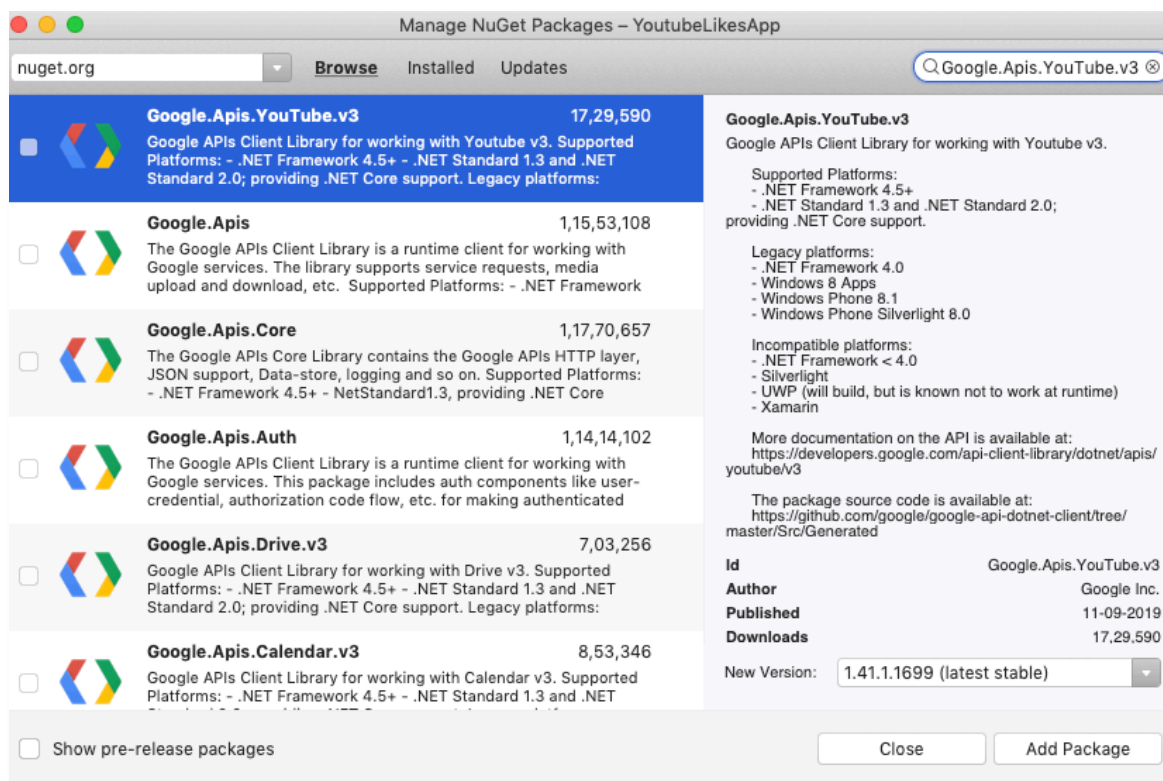
- {
  "kind": "youtube#videoListResponse",
  "etag": "\"p4VTdlkQv3HQeTEaXgvLePAydmU/qBntMb7EryttCfH85NrcTG8gyCo\"",
  "pageInfo": {
    "totalResults": 1,
    "resultsPerPage": 1
  },
  "items": [
    - {
      "kind": "youtube#video",
      "etag": "\"p4VTdlkQv3HQeTEaXgvLePAydmU/tsyL2gneIbUT_7cDqbj6P9YsrEo\"",
      "id": "cBFq_6Zj2aA",
      "statistics": {
        "viewCount": "2415038",
        "likeCount": "26640",
        "dislikeCount": "916",
        "favoriteCount": "0",
        "commentCount": "2657"
      }
    }
  ]
}
```

## C# Code to consume the service

There are two ways we can call the Youtube API either make a HTTP request to get the response using jQuery , javascript , PHP etc. or using Client libraries.

We are going to use .Net Client library to consume the service and get the Youtube videos number of likes. You can check all the details in the link <https://developers.google.com/api-client-library/dotnet/apis/youtube/v3>

1. Download Google.Apis.YouTube.v3 from nuGet manager.



2. Create a .Net Console application
3. Create a method to call the Authorise the user and get the YoutubeService
4. Create method to call the YoutubeService method- Video.List method to get the count of likes.

```

private async Task<YouTubeService> GetYouTubeService(string userEmail)
{
    UserCredential credential;
    using (var stream = new FileStream("client_secrets.json", FileMode.Open, FileAccess.Read))
    {
        credential = await GoogleWebAuthorizationBroker.AuthorizeAsync(
            GoogleClientSecrets.Load(stream).Secrets,
            // This OAuth 2.0 access scope allows for read-only access to the authenticated
            // user's account, but not other types of account access.
            new[] { YouTubeService.Scope.YoutubeReadonly },
            userEmail,
            CancellationToken.None,
            new FileDataStore(GetType().ToString())
        );
    }

    var youtubeService = new YouTubeService(new BaseClientService.Initializer()
    {
        HttpClientInitializer = credential,
        ApplicationName = GetType().ToString()
    });

    return youtubeService;
}

```

```

private async Task<ulong?> GetVideoLikes(YouTubeService service)
{
    ulong? likecount = 0;

    var videolistRequest = service.Videos.List("statistic");
    videolistRequest.Id = "VIDEO_ID"; // Replace with video id

    // Call the video.list method to retrieve results matching the specified query term.
    var videolistResponse = await videolistRequest.ExecuteAsync();

    foreach (var Videos in videolistResponse.Items)
    {
        likecount = Videos.Statistics.LikeCount;
    }

    Console.WriteLine("The number of likes for Video is {0}", likecount);
    return likecount;
}

```