How to obtain YouTube download link

# Abstract

This document describes a method of how to obtain YouTube video URI from YouTube servers.

Table of Contents

[Abstract 1](#_Toc380747488)

[Introduction 1](#_Toc380747489)

[YouTube Media Encoding 2](#_Toc380747490)

[Getting Video Information 2](#_Toc380747491)

[Parsing Video Information 3](#_Toc380747492)

[Decoding URL encoded data 3](#_Toc380747493)

[Understanding video information 3](#_Toc380747494)

[Contained information: 3](#_Toc380747495)

[Understanding URL encoded FMT stream map 3](#_Toc380747496)

[Format 3](#_Toc380747497)

[Downloading video 3](#_Toc380747498)

# Introduction

This is a document containing information and instructions on how to obtain YouTube download links through the services YouTube provides, no 3rd party software is required to obtain the download links.

# YouTube Media Encoding

The following table lists the video encoding currently supported by YouTube. For more information, visit: <https://en.wikipedia.org/wiki/YouTube#Quality_and_codecs>

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **itag value**[**[1]**](https://en.wikipedia.org/wiki/YouTube#endnote_media_type_table_note_1) | **Default container** | **Video resolution** | **Video encoding** | **Video profile** | **Video bitrate (Mbit/s)** [**[2]**](https://en.wikipedia.org/wiki/YouTube#endnote_media_type_table_note_2) | **Audio encoding** | **Audio bitrate (kbit/s)** [**[2]**](https://en.wikipedia.org/wiki/YouTube#endnote_media_type_table_note_2) |
| **5** | FLV | 240p | Sorenson H.263 | N/A | 0.25 | MP3 | 64 |
| **6** | FLV | 270p | Sorenson H.263 | N/A | 0.8 | MP3 | 64 |
| **13** | 3GP | N/A | MPEG-4 Visual | N/A | 0.5 | AAC | N/A |
| **17** | 3GP | 144p | MPEG-4 Visual | Simple | 0.05 | AAC | 24 |
| **18** | MP4 | 270p/360p | H.264 | Baseline | 0.5 | AAC | 96 |
| **22** | MP4 | 720p | H.264 | High | 2-3 | AAC | 192 |
| **34** | FLV | 360p | H.264 | Main | 0.5 | AAC | 128 |
| **35** | FLV | 480p | H.264 | Main | 0.8-1 | AAC | 128 |
| **36** | 3GP | 240p | MPEG-4 Visual | Simple | 0.175 | AAC | 36 |
| **37** | MP4 | 1080p | H.264 | High | 3–5.9 | AAC | 192 |
| **38** | MP4 | 3072p | H.264 | High | 3.5-5 | AAC | 192 |
| **43** | WebM | 360p | VP8 | N/A | 0.5 | Vorbis | 128 |
| **44** | WebM | 480p | VP8 | N/A | 1 | Vorbis | 128 |
| **45** | WebM | 720p | VP8 | N/A | 2 | Vorbis | 192 |
| **46** | WebM | 1080p | VP8 | N/A | N/A | Vorbis | 192 |
| **82** | MP4 | 360p | H.264 | 3D | 0.5 | AAC | 96 |
| **83** | MP4 | 240p | H.264 | 3D | 0.5 | AAC | 96 |
| **84** | MP4 | 720p | H.264 | 3D | 2-3 | AAC | 192 |
| **85** | MP4 | 1080p | H.264 | 3D | 3-4 | AAC | 192 |
| **100** | WebM | 360p | VP8 | 3D | N/A | Vorbis | 128 |
| **101** | WebM | 360p | VP8 | 3D | N/A | Vorbis | 192 |
| **102** | WebM | 720p | VP8 | 3D | N/A | Vorbis | 192 |

Table 1: <https://en.wikipedia.org/wiki/YouTube#Quality_and_codecs>

# Obtaining Video Information

Video information can be obtained from a response of a HTTP GET request directed at YouTube’s servers targeting <http://www.youtube.com/get_video_info?video_id=VideoID> where “**videoID**” is the id of the target video.

Example: <https://www.youtube.com/watch?v=HgNLrVk1BXU> video id is “**HgNLrVk1BXU**”, in order to obtain that specific videos information, transmit a standard HTTP GET request to: <http://www.youtube.com/get_video_info?video_id=HgNLrVk1BXU>

# Parsing Video Information

## Decoding URL encoded data

Information obtained from get\_video\_info is URL encoded. To decode, use an ASCII table to map the URL encoded %XX to ASCII characters. The following 2 websites will help.

* <http://www.w3schools.com/tags/ref_urlencode.asp>
* <http://www.asciitable.com/>

## Understanding video information

### Contained information:

* Thumbnails
* Video length
* View count
* Average rating
* URL encoded FMT stream map
* Watermarks
* Keywords
* Author
* Status
* Video ID
* Title

Data regarding download link to videos is found under URL encoded FMT stream map

|  |  |
| --- | --- |
| Search for: | &url\_encoded\_fmt\_stream\_map= |

## Understanding URL encoded FMT stream map

### Format

Data is encoded using URL format and will require to be decoded to be made usable. Objects are sorted into the following structure

* Quality (human readable quality)
* itag
* Sig (signature)
* Type
* Fall back host
* URL

And is delimited by a comma after the URL parameter.

Example: quailty=\*\*\*\*&itag=\*\*&sig=\*\*\*\*&type=\*\*\*\*&fallback\_host=\*\*\*\*&url=\*\*\*\*,

# Downloading video

Download link is generated by:

1. parsing the URL encoded FMT stream map
2. selecting video quality via itag id
3. combining the following elements of a stream map in the following order
   1. URL parameter
   2. &signature=
   3. Sig parameter

Example: [http://\*\*\*\*\*\*\*\*\*&signature=\*\*\*\*\*\*\*\*\*](http://*********&signature=*********)