

# Youtube

## Kids Whitelist

[youtube kids whitelist](#)

## Audio-Download

Mittlerweile gibt es sehr interessante Kanäle auf Youtube. Für Viel-Autofahrer sind aber oft die Audiodaten in offline-form besonders interessant.

Helfen kann hier das tool youtube-dl, einem kleinen Konsolenprogramm.

## Installation

```
sudo apt install youtube-dl libav-tools
```

Hier eine Beispielhafte Befehlszeile, um den gesamten Kanal „KenFM“ als Vorbis-Ogg herunter zu laden:

```
youtube-dl -f 'bestaudio' -x --audio-format vorbis --audio-quality 4 --download-archive archive.txt --output '%(upload_date)s %(title)s.%(ext)s' https://www.youtube.com/user/wwwKenFMde/videos?flow=list&view=0&sort=dd&live_view=500
```

```
[download] Downloading video 549 of 594
[youtube] 8CObiPzyb0s: Downloading webpage
[youtube] 8CObiPzyb0s: Downloading video info webpage
[youtube] 8CObiPzyb0s: Extracting video information
[youtube] 8CObiPzyb0s: Downloading MPD manifest
[download] Destination: 20130220 KenFM am Telefon - Christoph Hörstel über Bundeswehr-Soldaten in Mali.webm
[download] 100% of 22.08MiB in 00:12
[ffmpeg] Destination: 20130220 KenFM am Telefon - Christoph Hörstel über Bundeswehr-Soldaten in Mali.ogg
```

## Auszug aus der man-page

```
youtube-dl

--download-archive FILE
    Download only videos not listed in the archive file. Record
    the IDs of all downloaded videos in it.

--dateafter DATE
    Download only videos uploaded on or after this date (i.e.
    inclusive)
-r, --rate-limit LIMIT
```

```
Maximum download rate in bytes per second (e.g. 50K or 4.2M)
-x, --extract-audio
    Convert video files to audio-only files (requires ffmpeg or avconv and
    ffprobe or avprobe)
--audio-format FORMAT
    Specify audio format: "best", "aac", "vorbis", "mp3", "m4a", "opus",
    or "wav"; "best" by default

--audio-quality QUALITY
    Specify ffmpeg/avconv audio quality, insert a value between 0 (better)
    and 9 (worse) for VBR or a specific bitrate
    Embed thumbnail in the audio as cover art

-o, --output TEMPLATE
    Output filename template. Use %(title)s to get the title,
    %(uploader)s for the uploader name, %(uploader_id)s for
    the uploader nickname if different, %(autonumber)s to get an
    automatically incremented number, %(ext)s for the file-
    name extension, %(format)s for the format description (like "22 -
    1280x720" or "HD"), %(format_id)s for the unique
    id of the format (like YouTube's itags: "137"), %(upload_date)s for
    the upload date (YYYYMMDD), %(extractor)s for
    the provider (youtube, metacafe, etc), %(id)s for the video id,
    %(playlist_title)s, %(playlist_id)s, or %(playlist)s
    (=title if present, ID otherwise) for the playlist the video is
    in, %(playlist_index)s for the position in the
    playlist. %(height)s and %(width)s for the width and height of the
    video format. %(resolution)s for a textual de-
    scription of the resolution of the video format. %% for a literal
    percent. Use - to output to stdout. Can also be
    used to download to a different directory, for example
    with -o '/my/downloads/%(uploader)s /%(ti-
    tle)s-%(id)s.%(ext)s' .
    --output '%(upload_date)s %(title)s.%(ext)s'
```

requires:

ffmpeg or avconv and ffprobe or avprobe

```
// test: könig der löwen - der ewige kreis:
youtube-dl -f 'bestaudio' -x --audio-format vorbis --audio-quality 4
https://www.youtube.com/watch?v=7QgL6CyoQaI
```

```
youtube-dl -f 'bestaudio' -x --audio-format vorbis --audio-quality 4 --
download-archive archive.txt --output '%(upload_date)s %(title)s.%(ext)s'
https://www.youtube.com/user/wwwKenFMde/videos?flow=list&view=0&sort=dd&live
_view=500
```

[linux](#), [tools](#), [software](#), [deutsch](#), [script](#), [shell](#)

From:

<https://www.zeilhofer.co.at/wiki/> - **Verschiedenste Artikel von Karl Zeilhofer**

Permanent link:

<https://www.zeilhofer.co.at/wiki/doku.php?id=youtube>

Last update: **2018/03/26 00:55**

