

# Video Keeper Version 2.0.1

Made and maintained by Dan Martineau

Video Keeper is an elegant solution for keeping track of online videos that one wishes to watch and/or keep organized. The idea was born back in 2018 when I found myself spending a long time looking through my YouTube subscriptions page and opening an uncomfortably large amount of video tabs. I would end up with a massive text file of URLs, and it was quite clunky. Video Keeper aims to be a receptacle to paste video URLs into and have the video metadata—title, author, and date—be automatically obtained and stored.

The program is written in Java and has been tested on both Windows and Linux. While it has not been tested on Mac OS, the program is engineered to be fully platform independent and should run anywhere that Java 8 or above will run.

**Java must be installed in order to run this program.**

See Oracle's website to download Java: <https://www.java.com/>

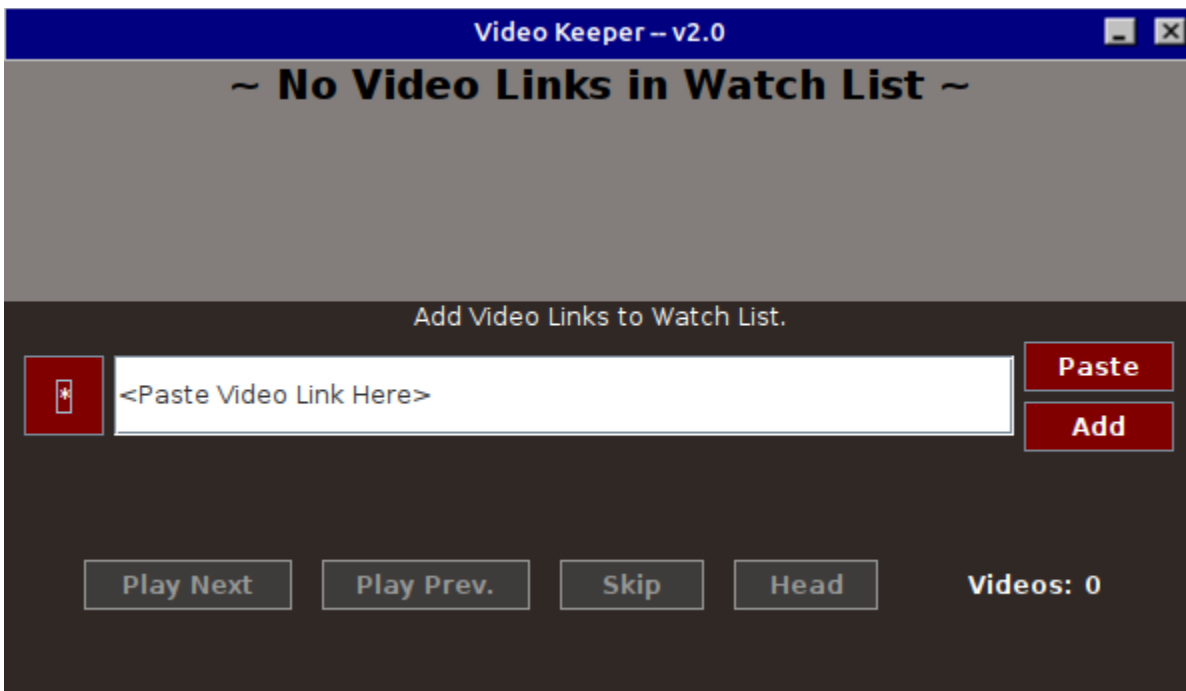
*This program is free software licensed under the GPL 3.0. A copy of the license is provided with the software. In short, this program's source code may be viewed, modified, and redistributed in any way that a user sees fit so long as the original creator, Dan Martineau, is given credit. Video Keeper comes with no warranty; it is to be used at one's own risk.*

**The source code for this project is available in the "src" directory found at:**

[https://github.com/justdan325/video\\_keeper/tree/master/src](https://github.com/justdan325/video_keeper/tree/master/src)

---

Begin by downloading the JAR file (VideoKeeper2pt0.jar) found alongside this manual. It is an executable file, so move it to a location where you would like it to be permanent on your computer. It is recommended to move it to its own dedicated directory. Double click the JAR, and you will be greeted with the following window:



Operation is very simple.

To add a video to the watch list, copy the video URL to your clipboard and paste it in the main text field that reads “paste video link here.” Once pasted, verify that the contents are the video URL and then hit “add.” Alternatively, you can also hit *Enter*. Once the video is “added,” its metadata will be obtained. For example:



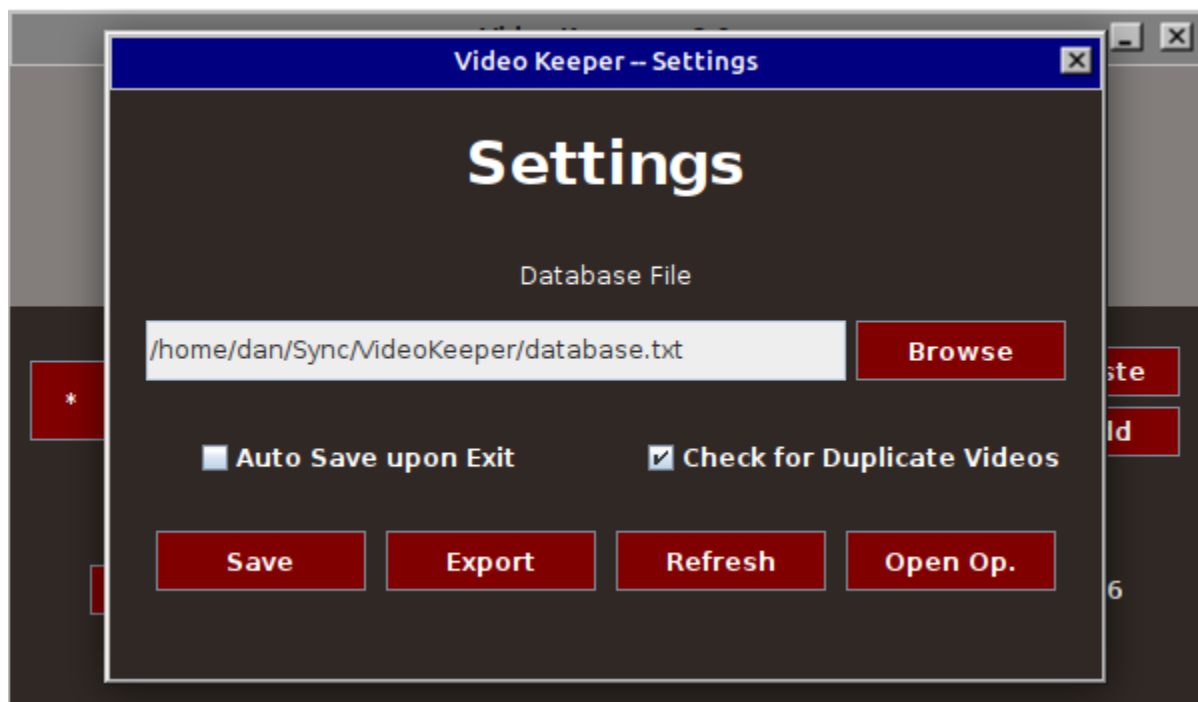
N.B. Supported websites are currently:

- YouTube
- Twitch (limited support)
- Vimeo
- Odysee
- Daily Motion
- BITCHUTE
- Rumble

To play the video that has been added, simply click “play next.” The video will open in your default web browser if the URL is valid and will subsequently be removed from the watch list. Note that how the video is opened can be changed. More on that to come.

If the browser window gets closed accidentally or something similar happens, clicking “play prev.” will open the previous video that had just been opened. Clicking “skip” will skip over the current video in the list and move onto the next (this does not remove the skipped video from the list). Clicking “head” will return to the top of the watch list.

The left-most button marked with a star will open the settings menu when clicked.



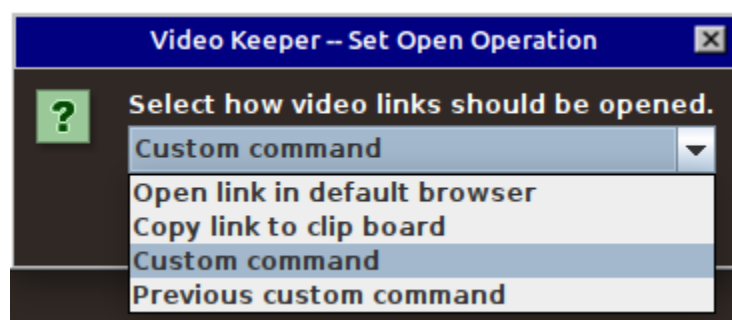
Here we are greeted with the default database file. The default will be “database.txt,” which is stored in the same place that the JAR file is stored. Clicking the “browse” button will allow you to change to another plain text file. This way, you can make use of multiple watch lists.

Below the file field, there are two check boxes. When selected, the left-most box will automatically save changes to the current watch list when the program is exited or when watch list is changed. Otherwise, you will be prompted to either save or discard changes in those situations.

The right-most box will check for duplicate videos in the current watch list. This is convenient when you are unsure if you’ve added a video or not. When in doubt, add the video again and if it is already in the list, then you will receive a warning and will be prompted to add it a second time or discard it.

The bottom four buttons serve the following functions:

- **Save:** Save changes to the current watch list.
- **Export:** export the current watch list to a plain text file of URLs without metadata.
- **Refresh:** Reload **all** video metadata within the list.
- **Open Op.:** This will allow you to change how opening videos is handled. See next section.



The default is “open link in default browser.” One can also choose to copy the link to the system clip board. Finally, one can input a custom command. This custom command will be the same as typing a command into a terminal/DOS prompt.

e.g. `firefox %VIDEOLINK%`

N.B. As the prompt will state, %VIDEOLINK% is the variable for the video URL. In other words, this means that the previous example would execute as:

`firefox https://www.youtube.com/watch?v=dQw4w9WgXcQ`

Thus opening the video in FireFox even if your default browser is Google Chrome.

Lastly, there is the “previous custom command” option. This will replace whatever the current open operation is with whichever **custom** command was input beforehand. The previous custom command in question will pop up and ask you to verify that it is indeed the desired command.