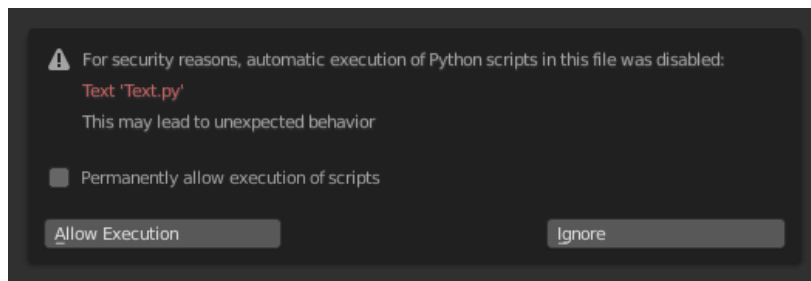


[Skip to content](#)

# Scripting & Security

The ability to include Python scripts within blend-files is valuable for advanced tasks such as rigging and automation. However, it poses a security risk since Python does not restrict what a script can do. Therefore, you should only run scripts from sources you know and trust. Automatic execution is disabled by default; however, some blend-files need this to function properly.

When a blend-file tries to execute a script and is not allowed, a dialog will appear. In it, you can choose to *Allow Execution* or to *Ignore* the scripts.



An Auto-run warning in the Info editor's header.

## Scripts in Blend-Files

### Auto Execution

Here are the different ways blend-files may automatically run scripts.

#### Registered Text-Blocks

A text data-block can have its *Register* option enabled which means it will load on start.

#### Animation Drivers

Python expressions can be used to *Drive* values and are often used in more advanced rigs and animations.

### Manual Execution

There are other ways scripts in a blend-file may execute that require user interaction (therefore will run even when auto execution is off), but you should be aware that this is the case since it is not necessarily obvious.

- Running a script in the Text editor.
- Rendering with Freestyle, because Freestyle uses scripts to control line styles.

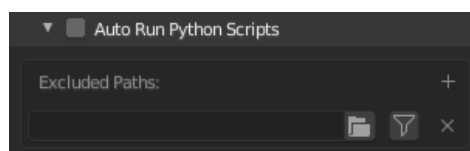
## Controlling Script Execution

Blender provides a number of ways to control whether scripts from a blend-file are allowed to automatically execute.

First, the File Browser has the option **Trusted Source** which you can use on a case-by-case basis to control auto execution. Since you may forget to set this, or may open a file without going through the File Browser, you can change the default (described next).

### Setting Defaults

In the Preferences, there is the toggle to [Auto Run Python Scripts](#). This means the **Trusted Source** option in the File Browser will be enabled by default and scripts can run when blend-files are loaded without using the File Browser. Once enabled you have the option to exclude certain directories; a typical configuration would be to trust all paths except for the download directory.



The Auto Run Python Scripts checkbox.

## Command Line

You may want to perform batch rendering or some other task from the command line, running Blender without an interface. In this case, the Preferences are still used but you may want to override them:

- Enable with `-y` or `--enable-autoexec`
- Disable with `-Y` or `--disable-autoexec`

### Example

To render an animation in background mode, allowing drivers and other scripts to run:

```
blender --background --enable-autoexec my_movie.blend --render-anim
```

#### Note

These command-line arguments can be used to start a regular Blender instance and will still override the Preferences.

[Previous](#)  
[Introduction](#)

Copyright © : This page is licensed under a CC-BY-SA 4.0 Int. License

[No](#)  
[Add-on Tutor](#)

Made with [Furo](#)

Last updated on 2025-05-10

[View Source](#)  
[View Translation](#)  
[Report issue on this page](#)