

# Introduction and Installation

## Installation

Install [the mod](#) and its two dependencies [Architectury](#) and [Rhino](#).

Make you use the most resent version of each mods for your version.

If you are using 1.16 fabric then use [this](#) instead.

When you first install KubeJS, you will need to launch Minecraft with the mods (and the game not crashing) to generate the some folders and files.

## The kubejs folder

### Finding it

Everything you do in KubeJS in located in the kubejs folder in your instance.

- In PolyMC the file structure will look like `polymc > instances > instance name > minecraft > kubejs`
- In CurseForge launcher the file structure will look like `curseforge > minecraft > instances > instance name > kubejs`
- In all of the above cases the `instance name` is the name of the instance
- In the normal Minecraft launcher it will be `.minecraft > kubejs`, unless you changed the instance folder.

From now on this will be referenced as the kubejs folder.

### The contents of it

- `startup_scripts`
  - Scripts that get loaded once during game startup
  - Used for adding items and other things that can only happen while the game is loading
  - Can reload code **not in an event** with `/kubejs reload_startup_scripts`
  - To reload all the code you must restart the game
- `client_scripts`
  - Scripts that get loaded every time client resources reload
  - Used for:
    - JEI events

- tooltips
  - other client side things
- Can reload code **not in an event** with `/kubejs reload client_scripts`
- Can reload all the code in client\_scripts with F3+T
- `server_scripts`
  - Scripts that get loaded every time server resources reload (world load, `/reload`)
  - Used for modifying:
    - recipes
    - tags
    - loot tables
    - handling server events
  - Can reload code **not in an event** with `/kubejs reload server_scripts`
  - Can be all the code in server\_scripts with `/reload`
- `exported`
  - Data dumps like texture atlases end up here
- `config`
  - KubeJS config storage.
  - This is also the only directory that scripts can access other than world directory
- `assets`
  - Acts as a resource pack
  - you can put any client resources in here, like:
    - textures
      - Example: `assets/kubejs/textures/item/test_item.png`
    - models
    - lang
    - etc.
  - Can be reloaded by pressing F3 + T
  - Can reload **only** the lang files (so faster) `/kubejs reload lang`
  - Read more about it [here](#).
- `data`
  - Acts as a datapack
  - you can put any server resources in here, like:
    - loot tables
      - Example: `data/kubejs/loot_tables/blocks/test_block.json`
    - functions
    - etc
  - Can be reloaded with `/reload`
  - Read more about it [here](#).
- `README.txt`
  - Contains the information here

You can find type-specific logs in `logs/kubejs/` directory

# Other Useful Tools

Code is just a language that computers can understand. However, the grammar of the language, called syntax for code, is very precise. When you code has a syntactical error, the computer does not know what to do and will probably do something that you do not desire.

With KubeJS we will be writing a lot of code, so it important to avoid these errors. Luckily, there are tools called code editors, that can help us write code correctly.

We recommend installing [Visual Studio Code](#) as it is light-ish and has great built in JS support. Now when you edit you java script files, it will not only warn you when you make most syntactical errors, but also help you prevent them in the first place.

---

Revision #6

Created 4 October 2022 20:11:58 by Q6

Updated 14 October 2023 05:16:28 by Lat