
Jupyter console Documentation

Release 4.1.0

The Jupyter Development Team

January 13, 2016

1	Configuration options	3
2	Changes in Jupyter console	11
2.1	4.1	11
2.2	4.0	11

The Jupyter console is a terminal frontend for kernels using the Jupyter protocol. The console can be installed with:

```
pip install jupyter-console
```

And started with:

```
jupyter console
```

To see configuration options:

```
jupyter console -h
```

To start the console with a particular kernel, ask for it by name:

```
jupyter console --kernel=julia-0.4
```

A list of available kernels can be seen with:

```
jupyter kernelspec list
```

You can connect to a live kernel (e.g. one running in a notebook) with its ID:

```
jupyter console --existing KERNEL_ID
```

or even connect to the most recently started kernel by default:

```
jupyter console --existing
```

Contents:

Configuration options

These options can be set in `~/ .jupyter/jupyter_console_config.py`, or at the command line when you start it.

ConnectionFileMixin.connection_file [Unicode] Default: ''

JSON file in which to store connection info [default: kernel-<pid>.json]

This file will contain the IP, ports, and authentication key needed to connect clients to this kernel. By default, this file will be created in the security dir of the current profile, but can be specified by absolute path.

ConnectionFileMixin.control_port [Int] Default: 0

set the control (ROUTER) port [default: random]

ConnectionFileMixin.hb_port [Int] Default: 0

set the heartbeat port [default: random]

ConnectionFileMixin.iopub_port [Int] Default: 0

set the iopub (PUB) port [default: random]

ConnectionFileMixin.ip [Unicode] Default: ''

Set the kernel's IP address [default localhost]. If the IP address is something other than localhost, then Consoles on other machines will be able to connect to the Kernel, so be careful!

ConnectionFileMixin.shell_port [Int] Default: 0

set the shell (ROUTER) port [default: random]

ConnectionFileMixin.stdin_port [Int] Default: 0

set the stdin (ROUTER) port [default: random]

ConnectionFileMixin.transport ['tcp'|'ipc'] Default: 'tcp'

No description

JupyterConsoleApp.confirm_exit [CBool] Default: True

Set to display confirmation dialog on exit. You can always use 'exit' or 'quit', to force a direct exit without any confirmation.

JupyterConsoleApp.existing [CUnicode] Default: ''

Connect to an already running kernel

JupyterConsoleApp.kernel_name [Unicode] Default: 'python'

The name of the default kernel to start.

JupyterConsoleApp.sshkey [Unicode] Default: ''

Path to the ssh key to use for logging in to the ssh server.

JupyterConsoleApp.sshserver [Unicode] Default: ''

The SSH server to use to connect to the kernel.

Application.log_datefmt [Unicode] Default: '%Y-%m-%d %H:%M:%S'

The date format used by logging formatters for %(asctime)s

Application.log_format [Unicode] Default: '%(name)s %(levelname)s %(message)s'

The Logging format template

Application.log_level [0|10|20|30|40|50|'DEBUG'|'INFO'|'WARN'|'ERROR'|'CRITICAL'] Default: 30

Set the log level by value or name.

JupyterApp.answer_yes [Bool] Default: False

Answer yes to any prompts.

JupyterApp.config_file [Unicode] Default: ''

Full path of a config file.

JupyterApp.config_file_name [Unicode] Default: ''

Specify a config file to load.

JupyterApp.generate_config [Bool] Default: False

Generate default config file.

InteractiveShell.ast_node_interactivity ['all'|'last'|'last_expr'|'none'] Default: 'last_expr'

'all', 'last', 'last_expr' or 'none', specifying which nodes should be run interactively (displaying output from expressions).

InteractiveShell.ast_transformers [List] Default: `traitlets.Undefined`

A list of `ast.NodeTransformer` subclass instances, which will be applied to user input before code is run.

InteractiveShell.autocall [0|1|2] Default: 0

Make IPython automatically call any callable object even if you didn't type explicit parentheses. For example, 'str 43' becomes 'str(43)' automatically. The value can be '0' to disable the feature, '1' for 'smart' autocall, where it is not applied if there are no more arguments on the line, and '2' for 'full' autocall, where all callable objects are automatically called (even if no arguments are present).

InteractiveShell.autoindent [CBool] Default: True

Autoindent IPython code entered interactively.

InteractiveShell.automagic [CBool] Default: True

Enable magic commands to be called without the leading %.

InteractiveShell.banner1 [Unicode] Default: 'Python 3.4.3 (default, Oct 14 2015, 20:28:29)\nType "copyri...'

The part of the banner to be printed before the profile

InteractiveShell.banner2 [Unicode] Default: ''

The part of the banner to be printed after the profile

InteractiveShell.cache_size [Int] Default: 1000

Set the size of the output cache. The default is 1000, you can change it permanently in your config file. Setting it to 0 completely disables the caching system, and the minimum value accepted is 20 (if you provide a value less than 20, it is reset to 0 and a warning is issued). This limit is defined because otherwise you'll spend more time re-flushing a too small cache than working

InteractiveShell.color_info [CBool] Default: True

Use colors for displaying information about objects. Because this information is passed through a pager (like 'less'), and some pagers get confused with color codes, this capability can be turned off.

InteractiveShell.colors ['NoColor' | 'LightBG' | 'Linux'] Default: 'Linux'

Set the color scheme (NoColor, Linux, or LightBG).

InteractiveShell.debug [CBool] Default: False

No description

InteractiveShell.deep_reload [CBool] Default: False

Deprecated

Will be removed in IPython 6.0

Enable deep (recursive) reloading by default. IPython can use the `deep_reload` module which reloads changes in modules recursively (it replaces the `reload()` function, so you don't need to change anything to use it). *deep_reload* forces a full reload of modules whose code may have changed, which the default `reload()` function does not. When `deep_reload` is off, IPython will use the normal `reload()`, but `deep_reload` will still be available as `dreload()`.

InteractiveShell.disable_failing_post_execute [CBool] Default: False

Don't call post-execute functions that have failed in the past.

InteractiveShell.display_page [Bool] Default: False

If True, anything that would be passed to the pager will be displayed as regular output instead.

InteractiveShell.history_length [Int] Default: 10000

No description

InteractiveShell.history_load_length [Int] Default: 1000

The number of saved history entries to be loaded into the readline buffer at startup.

InteractiveShell.ipython_dir [Unicode] Default: ''

No description

InteractiveShell.logappend [Unicode] Default: ''

Start logging to the given file in append mode. Use *logfile* to specify a log file to **overwrite** logs to.

InteractiveShell.logfile [Unicode] Default: ''

The name of the logfile to use.

InteractiveShell.logstart [CBool] Default: False

Start logging to the default log file in overwrite mode. Use *logappend* to specify a log file to **append** logs to.

InteractiveShell.multiline_history [CBool] Default: True

Save multi-line entries as one entry in readline history

InteractiveShell.object_info_string_level [0|1|2] Default: 0

No description

InteractiveShell.pdb [CBool] Default: False

Automatically call the pdb debugger after every exception.

InteractiveShell.prompt_in1 [Unicode] Default: 'In [\\#]: '

Deprecated, will be removed in IPython 5.0, use PromptManager.in_template

InteractiveShell.prompt_in2 [Unicode] Default: ' .\\D.: '

Deprecated, will be removed in IPython 5.0, use PromptManager.in2_template

InteractiveShell.prompt_out [Unicode] Default: 'Out [\\#]: '

Deprecated, will be removed in IPython 5.0, use PromptManager.out_template

InteractiveShell.prompts_pad_left [CBool] Default: True

Deprecated, will be removed in IPython 5.0, use PromptManager.justify

InteractiveShell.quiet [CBool] Default: False

No description

InteractiveShell.readline_parse_and_bind [List] Default: traitlets.Undefined

No description

InteractiveShell.readline_remove_delims [Unicode] Default: '-/~'

No description

InteractiveShell.readline_use [CBool] Default: True

No description

InteractiveShell.separate_in [SeparateUnicode] Default: '\\n'

No description

InteractiveShell.separate_out [SeparateUnicode] Default: ''

No description

InteractiveShell.separate_out2 [SeparateUnicode] Default: ''

No description

InteractiveShell.show_rewritten_input [CBool] Default: True

Show rewritten input, e.g. for autocall.

InteractiveShell.wildcards_case_sensitive [CBool] Default: True

No description

InteractiveShell.xmode ['Context'|'Plain'|'Verbose'] Default: 'Context'

No description

TerminalInteractiveShell.autoedit_syntax [CBool] Default: False

auto editing of files with syntax errors.

TerminalInteractiveShell.confirm_exit [CBool] Default: `True`

Set to confirm when you try to exit IPython with an EOF (Control-D in Unix, Control-Z/Enter in Windows). By typing 'exit' or 'quit', you can force a direct exit without any confirmation.

TerminalInteractiveShell.editor [Unicode] Default: `'vi'`

Set the editor used by IPython (default to \$EDITOR/vi/notepad).

TerminalInteractiveShell.pager [Unicode] Default: `'less'`

The shell program to be used for paging.

TerminalInteractiveShell.screen_length [Int] Default: `0`

Number of lines of your screen, used to control printing of very long strings. Strings longer than this number of lines will be sent through a pager instead of directly printed. The default value for this is 0, which means IPython will auto-detect your screen size every time it needs to print certain potentially long strings (this doesn't change the behavior of the 'print' keyword, it's only triggered internally). If for some reason this isn't working well (it needs curses support), specify it yourself. Otherwise don't change the default.

TerminalInteractiveShell.term_title [CBool] Default: `False`

Enable auto setting the terminal title.

ZMQTerminalInteractiveShell.callable_image_handler : Any

Callable object called via 'callable' image handler with one argument, *data*, which is `msg["content"]["data"]` where *msg* is the message from iopub channel. For example, you can find base64 encoded PNG data as `data['image/png']`.

ZMQTerminalInteractiveShell.image_handler : 'PIL'|'stream'|"tempfile"|"callable"

Handler for image type output. This is useful, for example, when connecting to the kernel in which pylab inline backend is activated. There are four handlers defined. 'PIL': Use Python Imaging Library to popup image; 'stream': Use an external program to show the image. Image will be fed into the STDIN of the program. You will need to configure *stream_image_handler*; 'tempfile': Use an external program to show the image. Image will be saved in a temporary file and the program is called with the temporary file. You will need to configure *tempfile_image_handler*; 'callable': You can set any Python callable which is called with the image data. You will need to configure *callable_image_handler*.

ZMQTerminalInteractiveShell.include_other_output [Bool] Default: `False`

Whether to include output from clients other than this one sharing the same kernel.

Outputs are not displayed until enter is pressed.

ZMQTerminalInteractiveShell.kernel_is_complete_timeout [Float] Default: `1`

Timeout (in seconds) for giving up on a kernel's `is_complete` response.

If the kernel does not respond at any point within this time, the kernel will no longer be asked if code is complete, and the console will default to the built-in `is_complete` test.

ZMQTerminalInteractiveShell.kernel_timeout [Float] Default: `60`

Timeout for giving up on a kernel (in seconds).

On first connect and restart, the console tests whether the kernel is running and responsive by sending `kernel_info_requests`. This sets the timeout in seconds for how long the kernel can take before being presumed dead.

ZMQTerminalInteractiveShell.mime_preference [List] Default: `traitlets.Undefined`

Preferred object representation MIME type in order. First matched MIME type will be used.

ZMQTerminalInteractiveShell.other_output_prefix [Unicode] Default: `'[remote] '`

Prefix to add to outputs coming from clients other than this one.

Only relevant if `include_other_output` is `True`.

ZMQTerminalInteractiveShell.stream_image_handler [List] Default: `traitlets.Undefined`

Command to invoke an image viewer program when you are using 'stream' image handler. This option is a list of string where the first element is the command itself and reminders are the options for the command. Raw image data is given as STDIN to the program.

ZMQTerminalInteractiveShell.tempfile_image_handler [List] Default: `traitlets.Undefined`

Command to invoke an image viewer program when you are using 'tempfile' image handler. This option is a list of string where the first element is the command itself and reminders are the options for the command. You can use `{file}` and `{format}` in the string to represent the location of the generated image file and image format.

ZMQTerminalInteractiveShell.use_kernel_is_complete [Bool] Default: `True`

Whether to use the kernel's `is_complete` message handling. If `False`, then the frontend will use its own `is_complete` handler.

KernelManager.autorestart [Bool] Default: `False`

Should we autorestart the kernel if it dies.

KernelManager.kernel_cmd [List] Default: `traitlets.Undefined`

DEPRECATED: Use `kernel_name` instead.

The Popen Command to launch the kernel. Override this if you have a custom kernel. If `kernel_cmd` is specified in a configuration file, Jupyter does not pass any arguments to the kernel, because it cannot make any assumptions about the arguments that the kernel understands. In particular, this means that the kernel does not receive the option `-debug` if it given on the Jupyter command line.

Session.buffer_threshold [Int] Default: `1024`

Threshold (in bytes) beyond which an object's buffer should be extracted to avoid pickling.

Session.copy_threshold [Int] Default: `65536`

Threshold (in bytes) beyond which a buffer should be sent without copying.

Session.debug [Bool] Default: `False`

Debug output in the Session

Session.digest_history_size [Int] Default: `65536`

The maximum number of digests to remember.

The digest history will be culled when it exceeds this value.

Session.item_threshold [Int] Default: `64`

The maximum number of items for a container to be introspected for custom serialization. Containers larger than this are pickled outright.

Session.key [CBytes] Default: `b''`

execution key, for signing messages.

Session.keyfile [Unicode] Default: `''`

path to file containing execution key.

Session.metadata [Dict] Default: `traitlets.Undefined`

Metadata dictionary, which serves as the default top-level metadata dict for each message.

Session.packer [DottedObjectName] Default: `'json'`

The name of the packer for serializing messages. Should be one of `'json'`, `'pickle'`, or an import name for a custom callable serializer.

Session.session [CUnicode] Default: `''`

The UUID identifying this session.

Session.signature_scheme [Unicode] Default: `'hmac-sha256'`

The digest scheme used to construct the message signatures. Must have the form `'hmac-HASH'`.

Session.unpacker [DottedObjectName] Default: `'json'`

The name of the unpacker for unserializing messages. Only used with custom functions for *packer*.

Session.username [Unicode] Default: `'username'`

Username for the Session. Default is your system username.

Changes in Jupyter console

A summary of changes in Jupyter console releases.

2.1 4.1

2.1.1 4.1.0

- readline/completion fixes
- use `is_complete` messages to determine if input is complete (important for non-Python kernels)
- fix: 4.0 was looking for `jupyter_console_config` in IPython config directories, not Jupyter

2.2 4.0

2.2.1 4.0.3

- fix `jupyter console --generate-config`

2.2.2 4.0.2

- `setuptools` fixes for Windows

2.2.3 4.0.0

First release as a standalone package.