IPython -- An enhanced Interactive Python - Quick Reference Card

ipython	Open IPython terminal console
juypter console	Open terminal console (4.0 and later)
ipython qtconsole	Open IPython qtconsole
juptyer qtconsole	Open qtconsole (4.0 and later)
ipython notebook	Open IPython Notebook (deprecated)
jupyter notebook	Open Notebook (browser interface, 4.0 and later)
ipython --help-all	Show all IPython start flags

Within IPython:

obj?, obj??	Get help, or more help for object (also works as ?obj, ??obj).
?foo.*abc*	List names in 'foo' containing 'abc' in them.
%magic	Information about IPython's 'magic' % functions.

Magic functions are prefixed by % or %%, and typically take their arguments without parentheses, quotes or even commas for convenience. Line magics take a single % and cell magics are prefixed with two %%.

Example magic function calls:

%alias d ls -F	'd' is now an alias for 'ls -F'
alias d ls -F	Works if 'alias' not a python name
alist = %alias	Get list of aliases to 'alist'
cd /usr/share	Obvious. cd -<tab> to choose from visited dirs.
%cd??	See help AND source for magic %cd
%timeit x=10	time the 'x=10' statement with high precision.
%%timeit x=2**100 x**100	time 'x**100' with a setup of 'x=2**100'; setup code is not counted. This is an example of a cell magic.

System commands:

!cp a.txt b/	System command escape, calls os.system()
cp a.txt b/	after %rehashx, most system commands work without !
cp ${f}.txt $bar	Variable expansion in magics and system commands
files = !ls /usr	Capture sytem command output
files.s, files.l, files.n	"a b c", ['a','b','c'], 'a\nb\nc'

History:

_i, _ii, _iii	Previous, next previous, next next previous input
_i4, _ih[2:5]	Input history line 4, lines 2-4
exec _i81	Execute input history line #81 again
%rep 81	Edit input history line #81
_, __, ___	previous, next previous, next next previous output
_dh	Directory history
_oh	Output history
%hist	Command history of current session.
%hist -g foo	Search command history of (almost) all sessions for 'foo'.
%hist -g	Command history of (almost) all sessions.
%hist 1/2-8	Command history containing lines 2-8 of session 1.
%hist 1/ ~2/	Command history of session 1 and 2 sessions before current.
%hist ~8/1-~6/5	Command history from line 1 of 8 sessions ago to line 5 of 6 sessions ago.
%edit 0/	Open editor to execute code with history of current session.

Autocall:

f 1,2	f(1,2) # Off by default, enable with %autocall magic.
/f 1,2	f(1,2) (forced autoparen)
,f 1 2	f("1","2")
;f 1 2	f("1 2")

Remember: TAB completion works in many contexts, not just file names or python names.