jc CLI output to JSON

CONCEPTS

jc only converts text to JSON; it does not filter. Pair it with jq. Most parsers return an array of objects, but some (like uptime) return one object, so check the schema with jc -h --<parser>.

Command output ps, dig, ls, ...
↓ text
jc --<parser> or magic
↓ JSON
jq / script filter, select, format

jc sits between text tools and JSON tools

RUNNING JC

jc *<command>* **[args]**
Magic syntax: run and parse in one step

<command> **| jc --***<parser>*
Standard syntax: parse piped text

cat *<file>* **| jc --***<parser>*
Parse a saved file or capture

jc *<start>*:*<end>* **--***<parser>*
Parse only a slice of input lines

jc dig *<host>* **| jq -r '.[].answer[].data'**
Magic syntax feeding jq

ps aux | jc --ps
Same, with the standard syntax

OUTPUT OPTIONS

jc -p *<command>*
Pretty-print JSON

jc -y *<command>*
YAML output

jc -r *<command>*
Raw output: strings, no int or date conversion

jc -M *<command>*
Add _jc_meta: parser, timestamp, exit code

OUTPUT OPTIONS (cont.)

jc -d *<command>*
Debug a parse failure (-dd verbose)

COMMON PARSERS

jc ps aux
Processes (--ps)

jc ls -l *<dir>*
Directory listing (--ls)

jc df -h
Disk usage per filesystem (--df)

jc dig *<host>*
DNS answers and timings (--dig)

jc ss -tulpn
Sockets (--ss)

jc ip route
Routing table (--ip-route)

jc lsblk
Block devices as a tree (--lsblk)

jc systemctl list-units
Units and their state (--systemctl)

jc stat *<file>*
File metadata (--stat)

jc free
Memory use (--free)

FILE AND STRING PARSERS

jc --csv < *<file>*.csv
CSV rows to objects

jc --asciitable < *<table>*.txt
Any aligned text table to JSON

STREAMING AND HELP

<command> **| jc --***<parser>***-s**
Streaming parser: one JSON line per input line

jc -h
List parsers

jc -h --*<parser>*
Parser docs, schema and examples