redis-cli Redis command line

CONCEPTS

Every redis-cli call sends one command and prints the typed reply. On a terminal you see (integer) or quotes; piped output switches to raw automatically.

Input args, stdin, prompt
↓ RESP
redis-cli modes and flags
↓ TCP / TLS
Server db 0-15, cluster
↓ typed reply
Output tty, raw, csv, json

Never run KEYS * or FLUSHALL on a production instance: KEYS blocks the server, FLUSHALL deletes every database. Use --scan and think twice.

CONNECT

redis-cli -h *<host>* **-p** *<port>*
Connect to host and port

redis-cli -u redis:// *<user>*:*<pass>*@*<host>*:*<port>*/*<db>*
Connect with a URI

redis-cli --user *<name>* **--askpass**
Log in, prompt for password

REDISCLI_AUTH=*<pass>* **redis-cli**
Pass password via environment

RUN AND SCRIPT

redis-cli *<command>* **[args]**
Run one command and exit

redis-cli --raw *<command>*
Plain output without type labels

redis-cli < *<commands.txt>*
Run commands from a file

redis-cli -r *<n>* **-i** *<sec>* *<command>*
Repeat a command

PROMPT

HELP *<command>*
Show syntax of a command

SELECT *<db>*
Switch database

KEYS

EXISTS *<key>*
Does the key exist

DEL *<key>*
Delete a key

TYPE *<key>*
Show the key's type

EXPIRE *<key>* *<seconds>*
Set a time to live

TTL *<key>*
Seconds left (-1 none, -2 gone)

SCAN *<cursor>* **MATCH** *<pat>* **COUNT** *<n>*
Iterate keys in steps

STRINGS AND COUNTERS

SET *<key>* *<value>*
Store a string

GET *<key>*
Read a string

SET *<key>* *<value>* **EX** *<seconds>* **NX**
Set with TTL if missing

INCR *<key>*
Add 1 to a counter

HASHES AND LISTS

HSET *<key>* *<field>* *<value>*
Set hash fields

HGET *<key>* *<field>*
Read one field

HGETALL *<key>*
Read all fields

LPUSH *<key>* *<value>*
Push on the left

SETS AND SORTED SETS

SADD *<key>* *<member>*
Add set members

ZADD *<key>* *<score>* *<member>*
Add with a score

INSPECT AND MONITOR

INFO
Full server report

redis-cli --scan --pattern '*<pat>***'**
List matching keys