|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | http://manuel.manuelles.nl/images/posts/redis.png | **Redis** Cheat Sheet | | | |
| |  |  | | --- | --- | | Strings |  | | APPEND key value | Append | | BITCOUNT key [ start stop ] | Count # of set bits | | BITOP AND dest [ src ]+ | Bitwise AND | | BITOP OR dest [ src ]+ | Bitwise OR | | BITOP XOR dest [ src ]+ | Bitwise XOR | | BITOP NOT dest src | Bitwise NOT | | BITPOS key bit [ start stop ] | Find first set bit | | DECR key | Decrement integer | | DECRBY key by | Subtract from integer | | GET key | Get by key | | GETBIT key offset | Get bit by index | | GETRANGE key start end | Get substring | | GETSET key value | Set, returning old value | | INCR key | Increment integer | | INCRBY key by | Add to integer | | INCRBYFLOAT key by | Add to float | | MGET [ key ]+ | Get multiple | | MSET [ key value ]+ | Set multiple | | MSETNX [ key value ]+ | Set multiple if doesn’t exist | | PSETEX key ms value | Set with expiry (ms) | | SET key value | Set | | SETBIT key offset value | Set bit by index | | SETEX key secs value | Set with expiry (s) | | SETNX key value | Set if doesn’t exist | | SETRANGE key offset value | Set substring | | STRLEN key | Get length |   *Strings can be used as numbers, arrays, bit sets and binary data*   |  |  | | --- | --- | | Databases |  | | DEL [ key ]+ | Delete item(s) | | DUMP key | Serialise item | | EXISTS [ key ]+ | Check for key | | EXPIRE key s | Set timeout on item | | EXPIREAT key ts | Set timeout by timestamp | | KEYS pattern | Get keys matching pattern | | MIGRATE | Transfer item between instances | | MOVE key db | Transfer item between databases | | OBJECT | Inspect item | | PERSIST key | Remove timeout | | PEXPIRE key ms | Set timeout (ms) | | PEXPIREAT key ts | Set timeout (timestamp) | | PTTL key | Get item TTL (ms) | | RANDOMKEY | Get random key | | RENAME key new | Change item’s key | | RENAMENX key new | Change key if new key doesn’t exist | | RESTORE key | Deserialise | | SCAN key cursor [ MATCH pattern ] [ COUNT count ] | Iterate keys | | SORT | Get or store sorted copy | | TTL key | Get item TTL (s) | | TYPE key | Get type of item |   *Times are specified in seconds (s) or milliseconds (ms) Timestamps (s) are specified as seconds since January 1, 1970* | |  |  | | --- | --- | | Hashes |  | | HDEL key [ field ]+ | Delete field(s) | | HEXISTS key field | Check for field | | HGET key field | Get item | | HGETALL key | Return all fields / values | | HINCRBY key field by | Add to integer value | | HINCRBYFLOAT key field by | Add to float value | | HKEYS key | Return all fields | | HLEN key | Get number of fields | | HMGET key [ field ]+ | Get multiple items | | HMSET key [ field value ]+ | Set multiple items | | HSCAN key cursor [ MATCH pattern ] [ COUNT count ] | Iterate fields | | HSET key field value | Set field | | HSETNX key field value | Set field if doesn’t exist | | HSTRLEN key field | Get string length of field | | HVALS key | Return all values |  |  |  | | --- | --- | | Sets |  | | SADD key [ member ]+ | Add item | | SCARD key | Get size of set | | SDIFF [ key ]+ | Get difference | | SDIFFSTORE dest [ key ]+ | Store difference | | SINTER [ key ]+ | Intersection | | SINTERSTORE dest [ key ]+ | Store intersection | | SISMEMBER key member | Check for item | | SMEMBERS key | Get all | | SMOVE src dest member | Move item to another set | | SPOP key [ count ]? | Pop random item | | SRANDMEMBER key [ count ] | Get random item | | SREM key [ member ]+ | Remove matching | | SSCAN key cursor [ MATCH pattern ] [ COUNT count ] | Iterate items | | SUNION[ key ]+ | Union | | SUNIONSTORE dest [ key ]+ | Store union |  |  |  | | --- | --- | | Sorted Sets |  | | ZADD key [ options ] [ score item ]+ | Add item | | ZCARD key | Get number of items | | ZCOUNT key min max | Number of items with score range | | ZINCRBY key incr member | Add to score | | ZINTERSTORE | Store intersection | | ZLEXCOUNT key min max | Lexicographical range count | | ZRANGE key start stop  [ WITHSCORES ] | Get items within rank range | | ZRANGEBYLEX key min max [ LIMIT offset count ] | Get items within lexicographical range | | ZRANGEBYSCORE key min max [ WITHSCORES ] [ LIMIT offset count ] | Get items within score range | | ZRANK key member | Get item rank | | ZREM key [ member ]+ | Remote item(s) | | ZREMRANGEBYLEX key min max | Remove items within lexicographical range | | ZREMRANGEBYRANK key start stop | Remove items within rank range | | ZREMRANGEBYSCORE key min max | Remove items within score range | | ZREVRANGE | ZRANGE in reverse order | | ZREVRANGEBYLEX | ZRANGEBYLEX in reverse order | | ZREVRANGEBYSCORE | ZRANGEBYSCORE in reverse order | | ZREVRANK | ZRANK in reverse order | | ZSCAN key cursor [ MATCH pattern ] [ COUNT count ] | Iterate items | | ZSCORE key member | Get item score | | ZUNIONSTORE dest numkeys [ key ]+ [ WEIGHTS [ weight ]+ ] [ AGGREGATE SUM|MIN|MAX ] | Store union |   *Lexicographical commands require items to have the same score* | |  |  | | --- | --- | | Lists |  | | BLPOP [ key ]+ timeout | Blocking left pop | | BRPOP [ key ]+ timeout | Blocking right pop | | BRPOPLPUSH src dest timeout | Blocking rotate | | LINDEX key index | Access by index | | LINSERT key BEFORE|AFTER pivot value | Insert next to | | LLEN key | Get length | | LPOP key | Pop from start | | LPUSH key [ value ]+ | Push onto start | | LPUSHX key value | Push if list exists | | LRANGE key start stop | Access range | | LREM key count value | Remove occurrences | | LSET key index value | Set item by index | | LTRIM list start stop | Remove start/end items | | RPOP key | Pop from end | | RPOPLPUSH src dest | Rotate | | RPUSH key [ value ]+ | Push onto end | | RPUSHX key value | Push onto end if list exists |  |  |  | | --- | --- | | Client/Server |  | | AUTH password | Request authentication | | ECHO message | Return message | | PING | Test connection | | QUIT | Close connection | | SELECT index | Set current database by index |  |  |  | | --- | --- | | Scripts |  | | EVAL | Run | | EVALSHA | Run cached | | SCRIPT EXISTS | Check by hash | | SCRIPT FLUSH | Clear cache | | SCRIPT KILL | Kill running script | | SCRIPT LOAD | Add to cache | |
|  | | |