Cheatography

redis Cheat Sheet

by wilsonsong via cheatography.com/72444/cs/18328/

Keys

DEL *key ...*	Delete item(s)	redis.Int
EXISTS *key*	Check for key	redis.Int
EXPIRE *key seconds*	Set timeout on item	redis.Int
EXPIREAT *key timestamp*	Set timeout by timestamp	redis.Int
PEXPIRE *key milliseconds*	Set timeout (ms)	redis.Int
PEXPIREAT *key milliseconds-timestamp*	Set timeout (ms timestamp)	redis.Int
PTTL *key*	Get item time to live (ms)	redis.Int
TTL *key*	Get item time to live	redis.Int

Strings

GET *key*	Get value of key	redis.String
INCR *key*	Increment integer	redis.Int
INCRBY *key increment*	Increment integer	redis.Int
SET *key value* [EX *second*] [NX|XX]	Set key	redis.String
SETEX *key* ***seconds*** *value*	Set with expiry (seconds)	redis.String
SETNX *key value*	Set if doesn't exist	redis.Int

Hashes

HDEL *key field ...*	Delete item	redis.Int
HEXISTS *key field*	Check for item	redis.Int
HGET *key field*	Get item	redis.String
HGETALL *key*	Return all items	redis.Strings/ redis.StringMap
HINCRBY *key field increment*	Add to integer value	redis.Int
HINCRBYFLOAT *key field increment*	Add to float value	redis.Float64
HKEYS *key*	Return all keys	redis.Strings
HLEN *key*	Get number of items	redis.Int
HMGET *key field ...*	Get multiple items	redis.Strings
HMSET *key field value ...*	Set multiple items	redis.Strings/redis.StringMap
HSET *key value*	Set item	redis.Int
HSETNX *key value*	Set item if doesn't exist	redis.Int
HSCAN *key cursor* [MATCH *pattern*] [COUNT *count*]	Iterate items	redis.MultiBulk

Lists

LINDEX *key index*	Access by index	redis.String
LINSERT *key* BEFORE|AFTER *pivot value*	Insert next to	redis.Int
LLEN *key*	Get length	redis.Int
LPOP *key*	Pop from start	redis.String
LPUSH *key value ...*	Push onto start	redis.Int
LPUSHX *key value*	Push if list exists	redis.Int

Lists (cont)		
LRANGE *key start stop*	Access range	redis.Strings
LREM *key count value*	Remove	redis.Int
LSET *key index value*	Set item by index	redis.String
LTRIM *key start stop*	Trim to specified range	redis.String
RPOP *key*	Pop from end	redis.String
RPUSH *key value ...*	Push onto end	redis.Int
RPUSHX *key value*	Push onto end if list exists	redis.Int

Sets		
SADD *key member ...*	Add item	redis.Int
SCARD *key*	Get size	redis.Int
SISMEMBER *key member*	Check for item	redis.Int
SPOP *key*	Pop random item	redis.String
SRANDMEMBER *key count*	Get random item	redis.Strings
SREM *key member ...*	Remove matching	redis.Int
SSCAN *key cursor* [MATCH *pattern*] [COUNT *count*]	Iterate items	redis.MultiBulk

Sorted Sets		
ZADD *key score member...*	Add item	redis.Int
ZCARD *key*	Get number of items	redis.Int
ZCOUNT *key min max*	Number of items within score range	redis.Int
ZINCRBY *key increment member*	Add to score	redis.String
ZRANGE *key start stop* [WITHSCORES]	Get items within rank range	redis.Strings/ redis.StringMap
ZRANGEBYSCORE *key min max* [WITHSCORES] [LIMIT *offset count*]	Get items within score range	redis.Strings/ redis.StringMap
ZRANK *key member*	Get item rank	redis.Int
ZREM *key member ...*	Remove item(s)	redis.Int
ZREMRANGEBYRANK *key start stop*	Remove items within rank range	redis.Int
ZREMRANGEBYSCORE *key min max*	Remove items within score range	redis.Int
ZREVRANK *key member*	ZRANGE in reverse order	redis.Int
ZSCORE *key member*	Get item score	redis.String
ZSCAN *key cursor* [MATCH *pattern*] [COUNT *count*]	Iterate items	redis.MultiBulk