

무중단 서비스 실현: Active-Active 이중화

Commands

Introduction

Strings

Lists

LISTS Introduction

LPUSH

RPOP

LPOP

RPUSH

LRANGE

LLS **Ent**

LLEN

LINDEX

LPOS **6.0.6**

LSET

LINSERT

LRM

LRM **Ent**

LTRIM

LMOVE **6.2**

RPOPLPUSH

LPUSHX

RPUSHX

BLPOP

BRPOP

BLMOVE **6.2**

BRPOPLPUSH

Sets

Sorted Sets (ZSets)

Hashes

Streams

Common Keys

Recyclebin **New**

Bits

Pub/Sub

Lua Script

HyperLogLog(PF)

Geo

Connection

Redis on Windows

LISTS Introduction



레디스 개발자 교육 신청



레디스 정기점검/기술지원
Redis Technical Support



레디스 엔터프라이즈 서버
Redis Enterprise Server

소개

Lists는 key와 value가 **일 대 다** 관계입니다.

value는 입력된 순서대로 저장됩니다.

Lists는 주로 큐(Queue)와 스택(Stack)으로 사용됩니다.

큐(Queue)는 들어오는 데이터를 순서대로 처리할때 사용합니다.

스택(Stack)은 웹브라우저의 백버튼 처럼 주로 되돌아 갈때 사용합니다.

키(리스트)의 생성과 삭제

value가 저장되면 키(리스트)는 생성됩니다. 키에 value가 하나도 없으면 키는 삭제됩니다.

즉, 키(리스트)의 생성과 삭제를 위한 별도의 작업은 필요없습니다.

명령어 요약

- **SET (PUSH):** LPUSH, RPUSH, LPUSHX, RPUSHX, LSET, LINSERT, RPOPLPUSH
- **GET:** LRANGE, LINDEX, LLEN
- **POP:** LPOP, RPOP, BLPOP, BRPOP
- **REM:** LREM, LTRIM
- **BLOCK:** BLPOP, BRPOP, BRPOPLPUSH

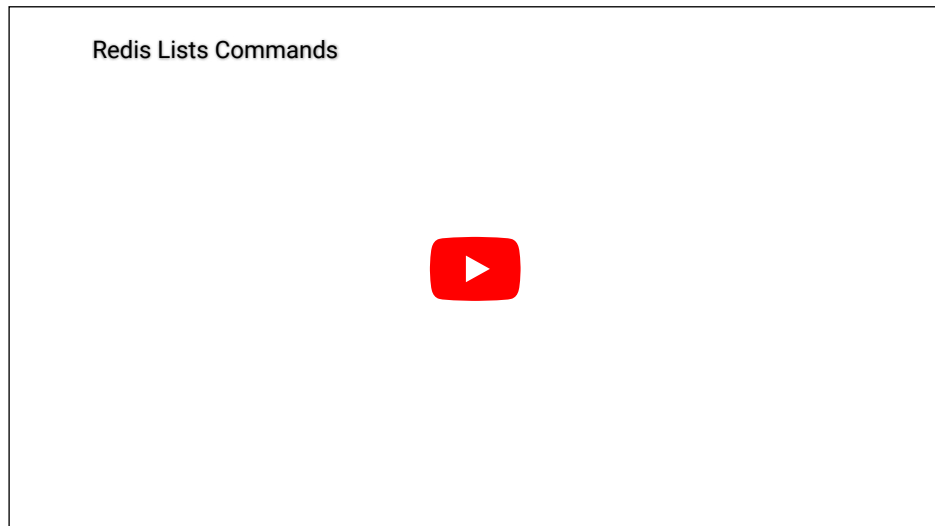
Lists 명령어 리스트

Commands	Version	Syntax	Description
LPUSH	1.0.0	key value [value ...]	왼쪽에서 리스트의 오른쪽에 데이터를 저장
RPOP	1.0.0	key	리스트 오른쪽에서 데이터를 꺼내고, 리스트에서는 삭제
LPOP	1.0.0	key	리스트 왼쪽에서 데이터를 꺼내고, 리스트에서는 삭제
RPUSH	1.0.0	key value [value ...]	오른쪽에서 리스트의 왼쪽에 데이터를 저장
LRANGE	1.0.0	key start stop	인덱스로 범위를 지정해서 리스트 조회
LLEN	1.0.0	key	리스트에서 데이터의 총 갯수를 조회
LINDEX	1.0.0	key index	인덱스로 특정 위치의 데이터를 조회
LSET	1.0.0	key index value	인덱스로 특정 위치의 값을 바꿈

LREM	1.0.0	key count value	값을 지정해서 삭제
LTRIM	1.0.0	key start stop	인덱스로 지정한 범위 밖의 값들을 삭제
RPOPLPUSH	1.2.0	key src_key dest_key	RPOP + LPUSH
BLPOP	2.0.0	key [key ...] timeout	리스트에 값이 없을 경우, 지정한 시간만큼 기다려서 값이 들어오면 LPOP 실행
BRPOP	2.0.0	key [key ...] timeout	리스트에 값이 없을 경우, 지정한 시간만큼 기다려서 값이 들어오면 RPOP 실행
BRPOPLPUSH	2.2.0	src_key dest_key timeout	리스트에 값이 없을 경우, 지정한 시간만큼 기다려서 값이 들어오면 RPOPLPUSH 실행
LINSERT	2.2.0	key BEFORE AFTER pivot value	지정한 값 앞/뒤에 새 값 저장
LPUSHX	2.2.0	key value	기존에 리스트가 있을 경우에만 LPUSH 실행
RPUSHX	2.2.0	key value	기존에 리스트가 있을 경우에만 RPUSH 실행
LPOS	6.0.6	key element	값으로 인덱스를 조회
LMOVE	6.2.0	source destination	리스트간 데이터 이동
BLMOVE	6.2.0	source destination	리스트간 데이터 이동 - 대기
LLS	Ent 6.0.0	key pattern	패턴(pattern)으로 값(value) 조회
LRM	Ent 6.0.0	key pattern	패턴(pattern)으로 값(value) 삭제

Total : 22

Lists 명령 설명



Redis Internal : LIST 내부 데이터 구조

- **짚 리스트(ZIP LIST): 메모리 절약형 데이터 구조**
 - 짚 리스트는 LIST, SORTED SET, HASH에서 공통으로 사용되는 데이터 구조로,
 - 데이터 구조와 주요 오퍼레이션에 대해서 알아보니다.
 - 그리고 LIST에서 어떻게 사용되는지 살펴봅니다.
- **리스트의 메인 데이터 구조 : 링크드 리스트(Linked List)**
링크드 리스트의 데이터 구조와 주요 오퍼레이션에 대해서 알아보니다.
- **퀵 리스트(Quick List)**
버전 3.2 전에는 엔트리 개수가 512개 이하면 짚 리스트에 저장되고, 513개부터는 링크드 리스트에 저장

되었다.

레디스 버전 3.2부터 리스트(LIST)의 내부 데이터 타입으로 퀵 리스트 하나로 확정되었다.
무엇이 달라졌는지 알아보시다.

<< GETSET

LISTS Introduction

LPUSH >>



✉ redisgate@gmail.com

☎ 02.503.2235

🏠 서울시 강남구 강남대로 342 역삼빌딩 5층 (역삼동) 우 06242

Copyright © 2014-2024 redisGate
All right reserved