

접속 및 기본 관리 (redis-cli)

Redis는 초고속 캐싱, 세션 관리, 메시지 브로커 등으로 활용되는 인메모리 키-값 데이터 구조 저장소입니다.

- CLI 접속: `redis-cli -h 127.0.0.1 -p 6379 -a "password"`
- 연결 테스트: `PING -> PONG`
- 데이터베이스 선택: `SELECT 0` (기본 0 15번 DB)
- 키 개수 확인: `DBSIZE`
- 데이터 초기화: `FLUSHDB` (현재 DB 삭제), `FLUSHALL` (전체 DB 삭제)
- 서버 상태 및 통계: `INFO` (메모리, CPU, 연결 수 등)

범용 키 명령어 (Key Operations)

- `KEYS pattern` : 패턴 일치 키 검색 (예: `KEYS user:*`) (운영 환경 주의)
- `SCAN cursor [MATCH pattern] [COUNT count]` : 안전한 비차단 키 순회
- `EXISTS key` : 키 존재 여부 확인 (1 또는 0 반환)
- `DEL key [key ...]` : 하나 이상의 키 삭제
- `TYPE key` : 키의 데이터 구조 타입 확인 (`string`, `list`, `hash` 등)
- `EXPIRE key seconds` : 키 만료 시간(초) 설정
- `TTL key` : 남은 수명 확인 (-1: 무제한, -2: 키 없음)
- `PERSIST key` : 설정된 만료 시간 제거
- `RENAME key newkey` : 키 이름 변경

5대 핵심 데이터 구조 및 명령어

1. 문자열 (Strings)

- `SET key value` : 키-값 저장
- `GET key` : 값 조회
- `SETEX key seconds value` : 만료 시간과 함께 값 저장
- `SETNX key value` : 키가 없을 때만 저장 (분산 락 구현에 활용)
- `MSET k1 v1 k2 v2 / MGET k1 k2` : 다중 키 저장 및 조회
- `INCR key / DECR key` : 정수 값 1 증가/감소
- `INCRBY key increment` : 지정한 크기만큼 증가

2. 해시 (Hashes - 객체 매핑)

- `HSET user:1 name "홍길동" age "28"` : 필드-값 저장
- `HGET user:1 name` : 특정 필드 값 조회
- `HMGET user:1 name age` : 여러 필드 값 조회
- `HGETALL user:1` : 모든 필드와 값 반환

- `HDEL user:1 age` : 필드 삭제
- `HEXISTS user:1 email` : 필드 존재 여부 확인
- `HINCRBY user:1 visits 1` : 필드 숫자 증가

3. 리스트 (Lists - 큐/스택)

- `LPU SH mylist val / RPU SH mylist val` : 왼쪽/오른쪽 삽입
- `LPOP mylist / RPOP mylist` : 왼쪽/오른쪽 꺼내기
- `LRANGE mylist 0 -1` : 전체 리스트 조회 (0부터 -1까지)
- `LLEN mylist` : 리스트 길이 반환
- `BLPOP mylist 10` : 팝 대기 (블로킹 큐 구현, 타임아웃 10초)

4. 셋 (Sets - 고유 값 집합)

- `SADD myset "a" "b" "c"` : 집합에 요소 추가
- `SMEMBERS myset` : 모든 요소 반환
- `SISMEMBER myset "a"` : 요소 포함 여부 확인
- `SREM myset "a"` : 요소 제거
- `SCARD myset` : 집합 원소 개수
- `SINTER set1 set2` : 교집합 연산
- `SUNION set1 set2` : 합집합 연산
- `SDIFF set1 set2` : 차집합 연산

5. 정렬된 셋 (Sorted Sets / ZSet - 랭킹 시스템)

- `ZADD leaderboard 100 "user1" 250 "user2"` : 점수와 함께 추가
- `ZRANGE leaderboard 0 -1 [WITHSCORES]` : 오름차순 조회
- `ZREVRANGE leaderboard 0 9 WITHSCORES` : 내림차순 TOP 10 조회
- `ZRANK leaderboard "user1"` : 오름차순 순위 (0부터 시작)
- `ZREVRANK leaderboard "user1"` : 내림차순 순위
- `ZSCORE leaderboard "user1"` : 특정 요소의 점수 확인
- `ZINCRBY leaderboard 50 "user1"` : 점수 가산

발행/구독 및 메시징 (Pub/Sub & Streams)

- 메시지 발행: `PUBLISH news "긴급 공지"`
- 채널 구독: `SUBSCRIBE news sports`
- 패턴 구독: `PSUBSCRIBE news.*`

트랜잭션 (Transactions)

MULTI

SET account:1 500

INCR transaction:count

EXEC (취소 시 DISCARD)

- 낙관적 락(Optimistic Locking): `WATCH account:1`

영속성 및 성능 모니터링

- 스냅샷 즉시 저장 (RDB): `BGSAVE`
- 실시간 실행 명령 모니터링: `MONITOR`
- 느린 쿼리 로그 확인: `SLOWLOG GET 10`