

Redis → Eviction Strategies

Eviction Strategies:

↳ Redis stores data in memory (RAM), which is limited

↳ When Redis reaches 100% RAM usage, it must "evict (remove)" some keys to make space.

Strategies:

① TTL (Time-to-Live)

↳ a key is deleted automatically when its expiration time is reached

Ex: In the web sessions

② LRU (least recently used)

↳ evicts the key that was not accessed for the longest time

Ex:

E-commerce app with millions of product pages

Hot products = accessed frequently

Old products = few visits

* Redis keeps frequently accessed products in memory and removes the old ones.

Imp: LRU prevents removing a frequently used item - high cache hit rate

③ LFU (least frequently used)

↳ evicts the key accessed the least number of times, regardless of recency.

Ex: Trending / popular content

viral feeds

leaderboards

Ads ranking

④ FIFO (First in First out)

↳ evicts the key that was inserted first, regardless of usage

Ex: Stream / Batched data storage

* you only need recent logs; older logs can disappear

-

⑤ Random Eviction

↳ choose any key to delete
- No logic

Use-case: High throughput + non-critical data

Ex:

- caching analytics counters
- IoT sensor data ingestion
- Temporary ML model features

Advantages:

- extremely fast - no update check
- good for heavy-write systems