Instructions   </> Code Examples   ▶ Screencasts   🎓 Concept   **Stage #YZ1**   (**Pending**)

---

rohitpaulk staff   1 year ago

The official Redis implementation handles expiry in two ways: "Active" and "Passive".

From the **Redis documentation**:

> A key is passively expired simply when some client tries to access it, and the key is found to be timed out.
>
> Of course this is not enough as there are expired keys that will never be accessed again. These keys should be expired anyway, so periodically Redis tests a few keys at random among keys with an expire set. All the keys that are already expired are deleted from the keyspace.

For the purposes of this challenge, the simplest implementation would be the "passive" one. That way you don't have to worry about running a separate thread to actively delete keys, you can just check if a value is expired when you receive a request.

⌃ **24**   ⌄   ↩ reply

---

jikdo 1 year ago

it will be great to add the active implementation as a challenge!

⌃ **28**   ⌄

---

thebenkogan 3 months ago

Not seeing this in the code examples, but remember you could be handling requests concurrently! So you have to think about race conditions when interacting with your in-memory DB. Mutual exclusion!

⌃ **14**   ⌄   ↩ reply

---

rohitpaulk staff   1 year ago

Wondering why "P" stands for in "PX"?

Antirez (Redis's author) answered this **here**:

> *P stands for precision, since the precision is improved by using milliseconds.*

Ready to run tests...   Show logs