



2023 Backend Intern Homework

題目敘述

- 讀完下述這一篇後，可以來規劃一下這樣的“共用的 Key-Value 列表系統”應該要如何實作？
- <https://medium.com/dcardlab/de07f45295f6>
 - 文章補充說明
 - 每個列表都會有個固定的 key，然而 page 內容會不斷低更新，所以需要透過 GetHead 拿到第一個 page key，然後透過 GetPage 拿到內容，每個 page 的 response 也都會帶下一個 page 的 page key，一直拿下去到沒有下一個 page 為止
 - GetHead("some-list-key") → {nextPageKey: "abce"}
 - GetPage("abce") → {articles, nextPageKey: "jyds"}
 - GetPage("jyds") → {articles, nextPageKey: "xyz"}
 - GetPage("xyz") → {articles}
 - 情境
 - 假設 Dcard 的個人化推薦文章列表(也就代表每個 user 會有自己的列表)是用“共用的 Key-Value 列表系統”來實作的，內部的算法 team 會 call API 一小時更新一次列表，所以要需要考慮使用量可能會很大
 - 需求：
 - 請以 go lang 進行開發
 - 設計與實作 RESTful API
 - Get Head / Get Page API
 - Set 的 API 可以思考一下怎麼設計比較方便使用
 - 文章提到的壓縮部份可以先不處理
 - 因為是共用 module，實際不會直接對 end-user，因此可以先不用考慮 auth
 - storage 的選擇，文章內是提到 PostgreSQL，但不一定要用 PostgreSQL，請在 README.md 裡面說明一下選擇是理由。如果一樣用 PostgreSQL，也還是需說明一下原因

- 每個列表的內容只須要保留一天，但是為了避免 storage 儲存太多請清掉不用內容，可以想想除了一筆筆刪之外怎麼清除更有效率？
- 請寫相對應的 **unit test**
- Bonus
 - 對內 Set 相關的 API 不一定要用 RESTful API 來進行溝通，也可以像文章提到的 gRPC Bidirectional Streaming 的方式，或者有其他想法都可以實作看看

提交方式

- 請將原始碼壓縮成 ZIP 檔提交，亦或是提供 Github Repo。