學習歷程

哈希表

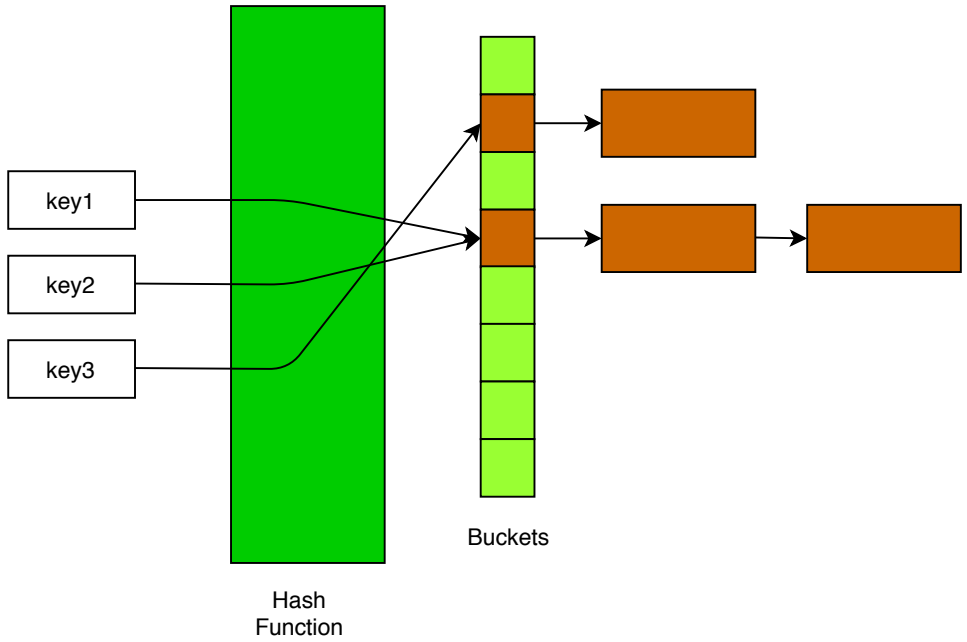
Hash Table是根據關鍵碼值 (Key-Value) 而直接進行訪問的數據結構
通過把關鍵碼值映射到表中壹個位置來訪問記錄，以加快查找的速度
哈希表的實現主要需要解決兩個問題，哈希函數和沖突解決

Hash Table的概念

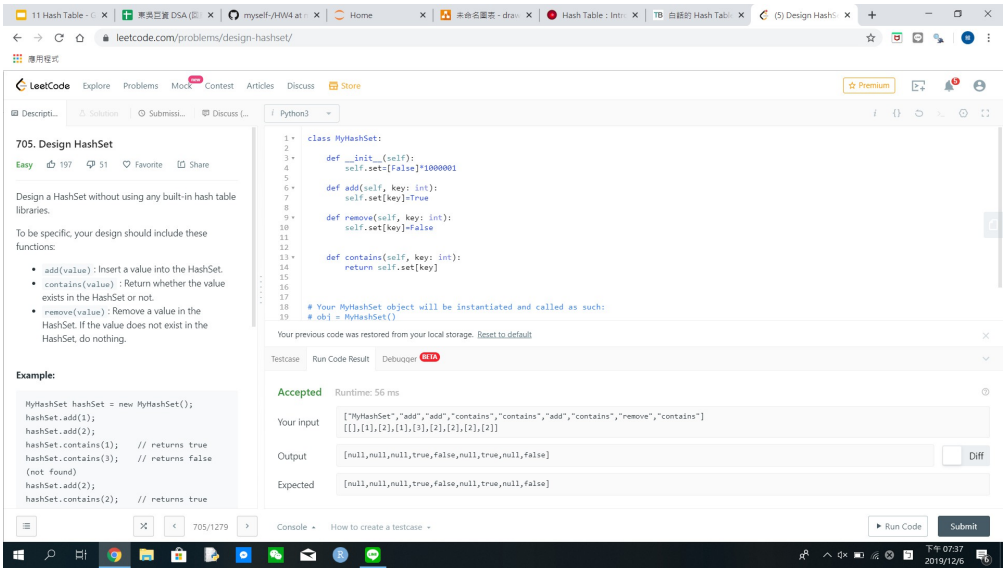
Hash Table希望能夠將存放資料的「Table」的大小(size)降到「真正會存放進Table的資料的數量」，也就是「有用到的Key的數量」。

Hash Function介紹

若把Table想像成「書桌」，slot想像成書桌的「抽屜」，那麼為了要能更快速找到物品，
當然是希望「每一個抽屜只放一個物品」，如此一來，只要拿著Key，
透過Hash Function找到對應的抽屜，就能保證是該Key所要找的物品。
反之，如果同一個抽屜裡有兩個以上的物品時，便有可能找錯物品。



因為去了星期二的課，了解到LeetCode中705題對於hash table有關連
在完成705題之後進行修改後，最終完成的我的程式碼



參考資料

http://alrightchiu.github.io/SecondRound/hash-tableintrojian-jie.html#ht
https://blog.techbridge.cc/2017/01/21/simple-hash-table-intro/
https://leetcode.com/problems/design-hashset/
https://www.youtube.com/watch?v=aZVNWYSR_sY
https://www.youtube.com/watch?v=2BldESGZKB8