Redis Data Structure Description

1.Use a Redis string “courseID” to store the number of existing data from MongoDB. Besides, the negative numebr can be used as the score of the sorted set “courseList”, e.x. if the “courseID” is 1, the score of this course is -1. In this case, if we use “ZRANGEWITHSCORE”, the courses will be ordered by time (since the data in MongoDB is ordered by tim) and the new course will have a smallest score so that it can be shown at the front of the website page.
2.Use a Redis string “studentID” to store the number of students.
3.Use a sorted set “courseList” to store all existing course. If a course is deleted, it will also be deleted from the sorted set so that the website will show data correctly. And the score of the sorted set is -courseID we mentioned before.
4.Use a set “studentList” to store all existing students to check whether the student is signed up for the system.
5.Use a hash like “student:$studentID” to store each student who signed up in the system. The information of each student like first name, last name, address, etc are stored in each hash.
6.Use a hash like “course:$courseID” to store each course’s information like course name, car type, start time, duration, etc.
7.Use a hash like “course:$courseID:coach” to store the coach of each course. Since it is a one to many realationship, each course has a certain coach. The information of coach like first name, last name, ratings, etc are stored in the data structure.
8.Use a set like “course:$courseID:students” to store all students of each course. The data in the set is studentID, like [1,4,5].