Lab 1 **Hanjing Wang(516030910015)**
Database System ACM Class, Zhiyuan College, SJTU
Prof. **Feifei Li** Due Date: April 23, 2019

Design Decisions

simpledb.TupleDesc

private TupleDesc constructor

A private constructor is added along with the arraycopy method with consideration of merge performance. It is not neccesary to use deep copy and constructor in the merge method since there is no public API for users to modify a tuple desciptor once constructed.

simpledb.Catalog

Attribute: idMappedTable and nameMappedTable

Give the consideration that database systems are usually query-intense in the real world applications, the efficiency of query operation is one of the first-order problems. Therefore, two ConcurrentHashMap **idMappedTable** and **nameMappedTable** maintain two mappings, ¡tableid - Table¿ and ¡tablename - tableid¿ correspongdingly, to efficiently respond to different queries.

simpledb.BufferPool

pidMapedTrans

In addition to the PageId-to-Page table, a PageId-to-Tid table is maintained as a simple lock. When the getPage method try to access a page, its page id is first checked to make sure the page will not be opened by two transcations, therefore being threadsafe. And when a page is not busy, it is removed from the table by releasePage method.

simpledb.HeapFile

public DbFileIterator iterator

Since the interpage iterator has been implemented, the HeapFile iterator is designed as a wrapper of it with additional design enabling it switching from page to page. When a new page is accessed, the iterator locks it and releases it when the iterator leaves.

Difficulties

I spent approximately 15 hours in the project. The major difficulty lay in the comprehension of the whole architect of the design since there are tens of definitions and methods. I usually found my confused after spotting a new definition. Thankfully, the instructions of the docs helped me a lot. And another problem is that we are not familiar with Java and the related tool chain.