DS542 Project3 : A Query Execution Engine

BoyaZhou,Yuting Liang

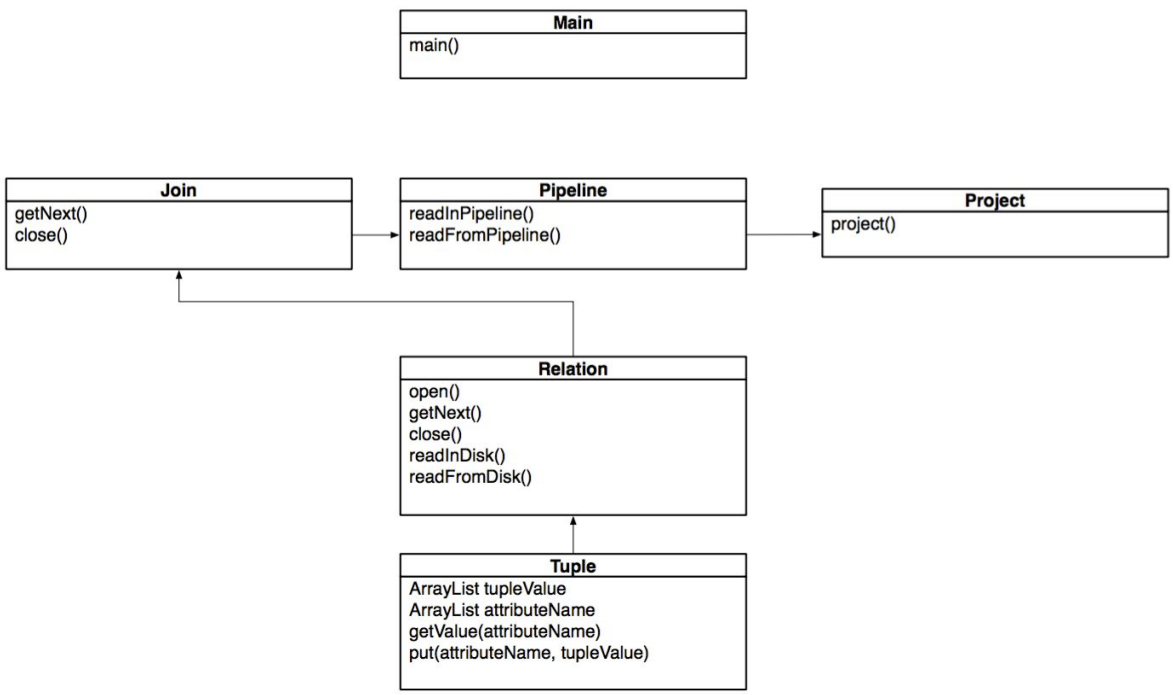
Introduction

A query execution engine. You may have realized that the functions of exercise 1 are actually methods of a `Relation` class. In a real database, we will have multiple instances of Relation, each representing one table. Modify your answer from exercise 1 to create the class and also add open(), getNext() and close() methods to it. Use the class to create and populate `city` and `country` tables. The data may be obtained from the MySQL World Database sample. Use the class to find all cities whose population is more than 40% of the population of their entire country.

Assumption

1.we use the most simple join with considering block but not index
2.we put the two csv in a variable to simulate disk, rather than save in the real disk
3.we assume that each block can conntain 12 tuples
4.we assume that each pipeline can contain 30 tuples
5.join use data read from each block in memory, project use data from pipeline from join
6.in this small project, we only define 2 operators, join and project, we join first, and project certain attributes

UML



Design decisions

Tuple

we use tuple to simulate each record in the block or in the block in memory, we change something in exercise1 we made before to fit this question.

Relation

It is a list of tuple when it's in disk, however, when we read it to memory, each time we can read only 1 block(open()), so we make a variable named record to help us find next time where we should begin to read on the disk.

getNext() is defined as what introduction said. When we define close(), we should make record++.

Join Mechanism:

```
getNext(){
        left.getNext()
        if(left==null){
                return results
        }
        right.open
        results=null
        while(true){
                right=right.getNext()
                if(right==null){
                        right.close()
                        break
                }

                if(left fit right){
                results.add
                }
        }
        return getNext()
}
```

Pipeline & Project

When after join, we put the result in pipeline, so project can use the result in pipeline, however, we didn't have finished pipeline(this version before deadline is just a naive version) part. Need some improvement in the future.