PolyU COMP2021 Assignment 5

Deadline: 20 Nov 2015, 11:00am

Introduction

Goal: By doing this assignment, you will have hand-on experiences of applying design patterns in an OO project.

Executing an SQL query in database systems is based on at least two design patterns: *Iterator* and *Chain of Responsibility*.

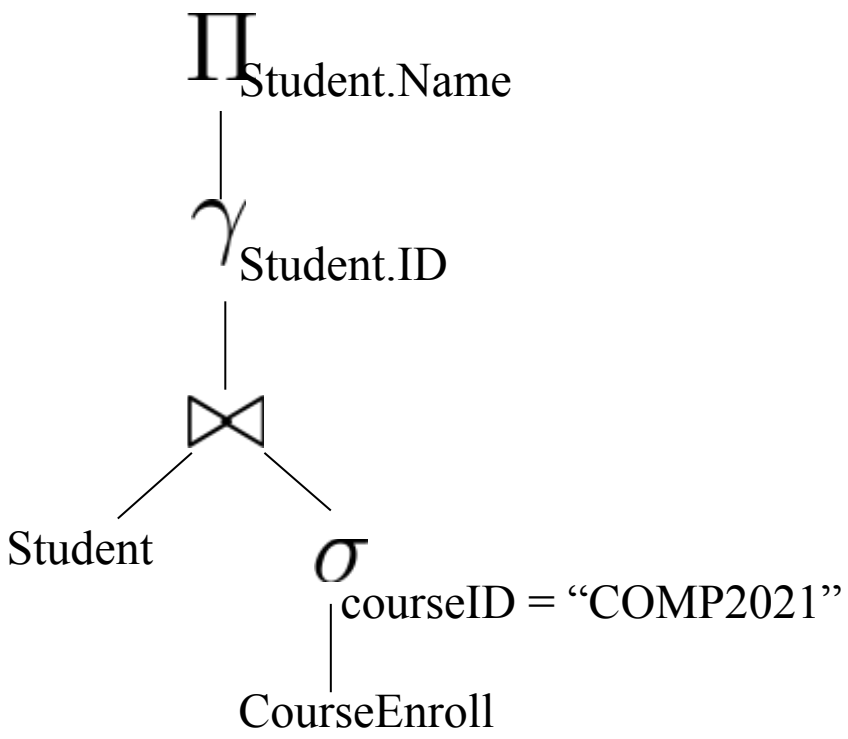
Recall that in SQL, we have keywords and concepts of “SELECT”, “JOIN”, “WHERE”, “ORDER-BY”. The following example SQL is trying to find the names of students that enroll in “COMP2021” (the SQL syntax may not be the standard one):

```
SELECT Student.Name
FROM Student JOIN CourseEnroll
WHERE CourseEnroll.courseID = “COMP2021”
ORDER BY Student.ID
```

Inside a database, the above SQL is transformed into an internal expression call a *tree of relational algebra operators*. The following shows a rough mapping of SQL keywords/concept to relational algebra operators:

SQL keyword/concept	Relational operator
SELECT	PROJECTION Π
WHERE	SELECTION σ
JOIN	JOIN (ON common attribute) $\bowtie$
ORDER BY	SORT γ

So, for the example SQL above, the database would transform the SQL to:



In the implementation:

1) each relational operator is **connected through chain-of-responsibility.**
2) each relational operator implements the given iterator interface such that the output calls the root operator’s ‘next()’ and the root operator calls its children’s ‘next()’.
3) There is a “Table” class that also implements the iterator interface with implementation of reading a file and each next() call returns one record being read. If all records are read, return null (to indicate that it has finished reading)
4) The “Selection” class has a next() method, when being called, it will call its child’s next() to get a tuple, examine whether that tuple’s attribute has satisfied the filter (e.g., couresID = “COMP2021”) and return it if yes or calls its child operator’s next() again if not.
5) The “Join” class has a next() method, when being called, it will call its child**ren**’s next() to get tuples to find which pair of tuples could be joined and join them.
6) The “Sort” class has a next() method, when being called, it will call its child’s next() to fetch all tuples and sort them.
7) The “Projection” class has a next() method, when being called, it will call its child’s next() to get a tuple, keeping only the attributes who are of interested (e.g., Student.Name) and discarding the rest away. Usually Projection is the root operator in the tree.

The Assignment Part

You are given 2 sets of files:

- **ReadMe.pdf (this file)**
- **/Assignment5**
    - **/src**
    - **/datafile**
    - **/simpledatabase**
    - **/test**
    - **/lib**
    - **/bin**
    - **/doc**
    - **build.xml**

Your job: Implement 5 java files under /simpledatabase to make the whole thing work.

(1) **Table.java**
(2) **Selection.java**
(3) **Join.java**
(4) **Sort.java**
(5) **Projection.java**

Requirements:

● Don't modify other files except the above 5 java files, otherwise you will get 0 marks
● Make sure your 5 java files can be compiled with other files before submission,
i.e., when you run "ant compile" and "ant test" command, there will be no errors.

To begin with:

Before you start working on the assignment, you may try:

Under /Assignment 5,

- **ant compile**
  - to compile all the java files, and then run
- **ant test**

You shall see something like this:

```
Assignment5 — -bash — 80×24

clean:
   [delete] Deleting directory /Users/Julian/Desktop/Assignment5/question/Assign
ment5/bin

compile:
    [mkdir] Created dir: /Users/Julian/Desktop/Assignment5/question/Assignment5/
bin
    [javac] /Users/Julian/Desktop/Assignment5/question/Assignment5/build.xml:37:
 warning: 'includeantruntime' was not set, defaulting to build.sysclasspath=last
; set to false for repeatable builds
    [javac] Compiling 16 source files to /Users/Julian/Desktop/Assignment5/quest
ion/Assignment5/bin

test:
    [mkdir] Created dir: /Users/Julian/Desktop/Assignment5/question/Assignment5/
testreport
    [junit] Running test.Assign5TestSuite
    [junit] Tests run: 5, Failures: 5, Errors: 0, Time elapsed: 0.014 sec
    [junit] Test test.Assign5TestSuite FAILED

BUILD SUCCESSFUL
Total time: 1 second
```

Showing none of the test cases pass

After you successfully finish this assignment,

Under /Assignment 5,

- **ant compile**
  - to compile all the java files, and then run
- **ant test**

You shall see something like this:

```
Assignment5 — -bash — 80×24
JuliandeMacBook-Pro:Assignment5 Julian$ ant test
Buildfile: /Users/Julian/Desktop/Assignment5/ans/Assignment5/build.xml

clean:
   [delete] Deleting directory /Users/Julian/Desktop/Assignment5/ans/Assignment5
/bin

compile:
    [mkdir] Created dir: /Users/Julian/Desktop/Assignment5/ans/Assignment5/bin
    [javac] /Users/Julian/Desktop/Assignment5/ans/Assignment5/build.xml:37: warn
ing: 'includeantruntime' was not set, defaulting to build.sysclasspath=last; set
 to false for repeatable builds
    [javac] Compiling 16 source files to /Users/Julian/Desktop/Assignment5/ans/A
ssignment5/bin

test:
    [mkdir] Created dir: /Users/Julian/Desktop/Assignment5/ans/Assignment5/testr
eport
    [junit] Running test.Assign5TestSuite
    [junit] Tests run: 5, Failures: 0, Errors: 0, Time elapsed: 0.015 sec

BUILD SUCCESSFUL
Total time: 1 second
```

Showing all of the test cases pass

Submission:

1. ***Submit your whole project "/Assignment5" to GITHUB***
2. **Your GITHUB repository should be *public* to allow anyone can see your repository**
3. ***Put your github hyperlink into a file github.txt" and submit that to Blackboard***

How do we grade?

- First, we will find your github link from your blackboard "github.txt"
- Then we will git pull your whole project
- Then, we will run ant test to check your program
- We will add more secret test cases to ensure you have no HARD-coding.

Grading policy

One measure: Number of levels passed

In this assignment, we have 5 test cases (under /test directory) for you to attack. For each test case that passes, +20.

Late Penalty

late x day: your score = raw score * (100 – 20x)%

Plagiarism

It is easy to detect the similarity of source files, and cases will be strictly handled according to the University's regulation, so please don't risk doing that.

Questions?

Contact your assignment TA, Mr. Julian Ip (csyfip@comp.polyu.edu.hk) or ask

Eric during lecture breaks.