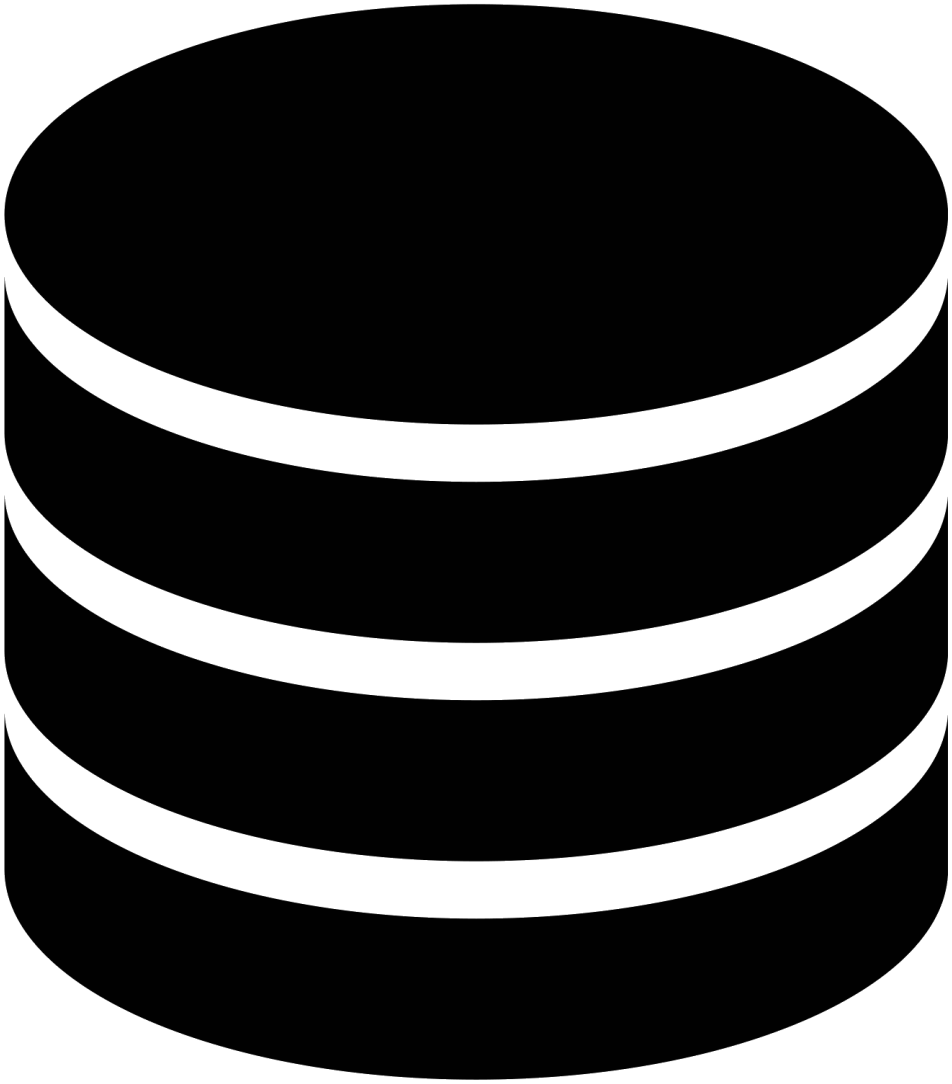
Simple DBMS

OOP Assignment 3



Karim Mohamed ali

Youssef abdallah Youssef

Michael ramy ramsis

Mazen Ahmed Khater

Application Description and features:

An object oriented xml-based Database management that handle most important SQL queries such as (create database, create table, insert into, select, update, …).

The DBMS support simple conditions: =, >, < and multiple conditions: AND, OR, NOT.

Design:

Overview:

The design of the project relied on the interpreter design pattern. Moreover, other design patterns were also used to develop the project such as the factory pattern and the singleton pattern.

Design Patterns Used:

Interpreter Pattern:

In our case the abstract expression was represented by the interface expressions which different classes implement such as select, insert, update, …

Parser:

There is a special class for parsing the sql queries using java regular expressions, it validates the query then sends the appropriate data extracted from the query to the expressions factory.

Factory pattern:

it gets the extracted data for each from the parser then returns an instance of sql expression implementing the interface expressions to serve the interpreter pattern.

Adapter pattern:

Interpret() method in the expressions interface returns a List<Row> type however, the return type for the methods: insert, delete, update is just an integer representing the number of updated rows. In order to solve such a problem we created adaptee that returns the count of the updated rows and to which we delegate the request.

Singleton:

In our case there was only one parser, and only one controller the so called "Façade".

Façade Pattern:

The class that makes everything work together and has a reference to all other classes in the application. It is like a higher level interface that makes the subsystem easier to use.

Main Classes:

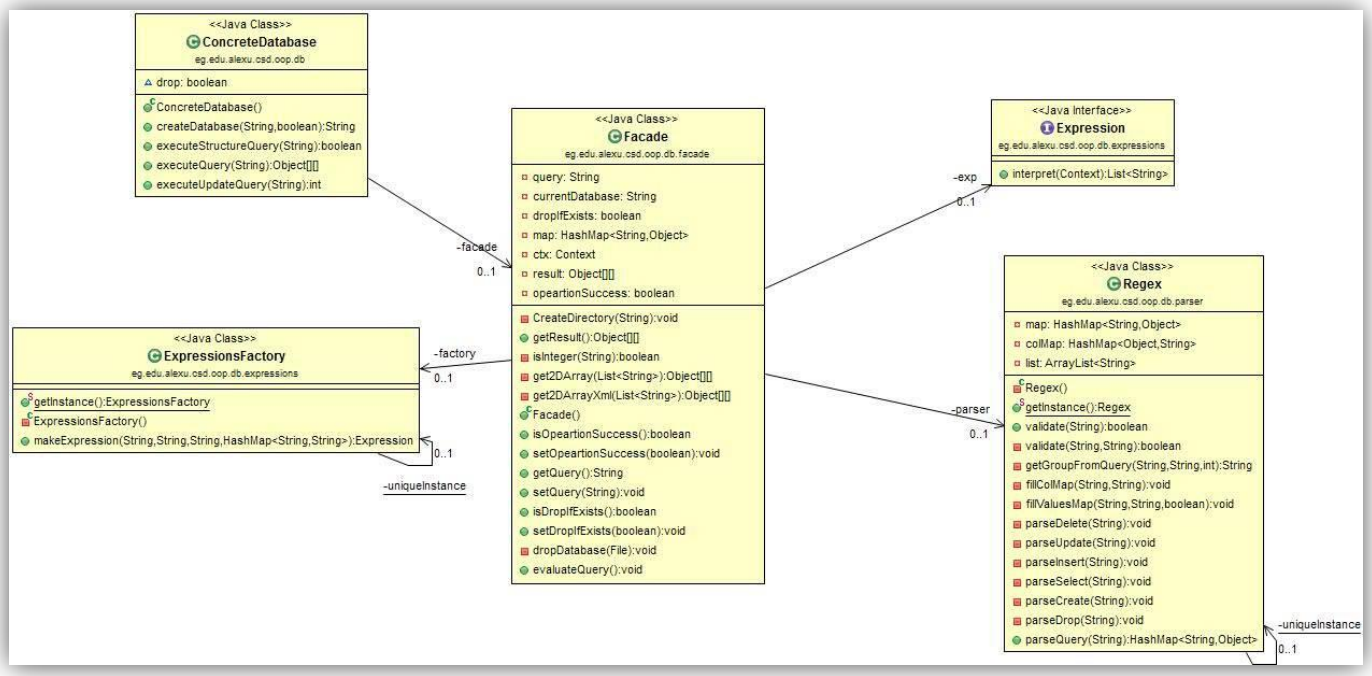
ConcreteDatabase: the class that implements the database Interface, it delegates all the work to the façade class.

XmlFile class: performs the parsing and writing operations of the xml file.

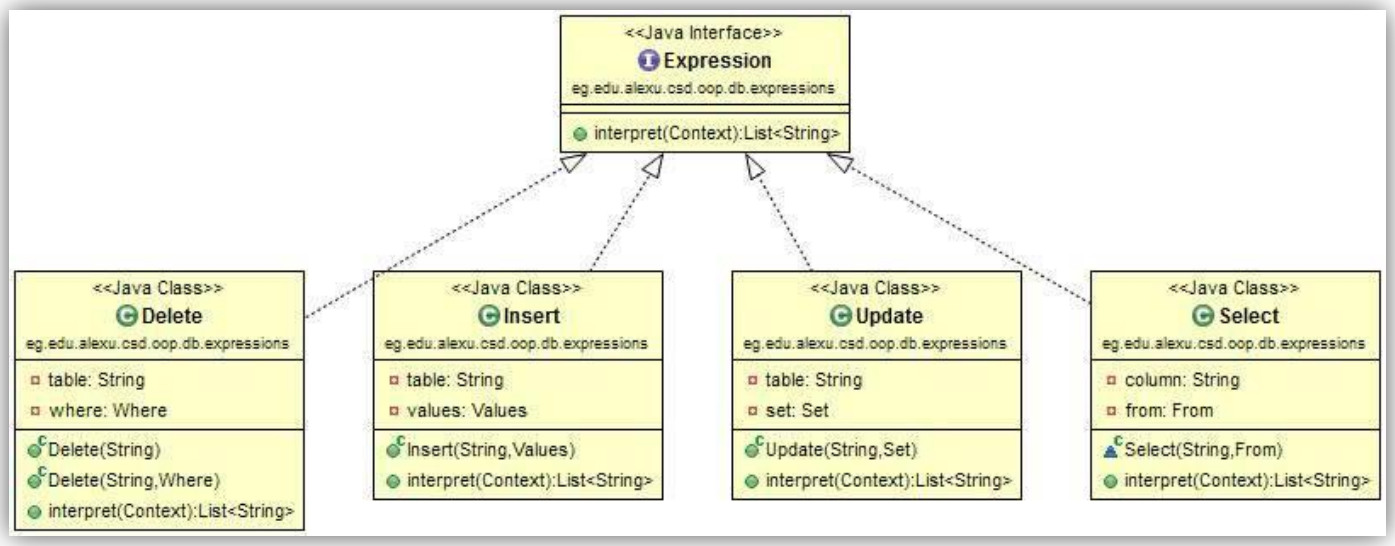
DTDFile class: performs the parsing and writing of DTD files that store the schema of each table stored in the database.

UML:

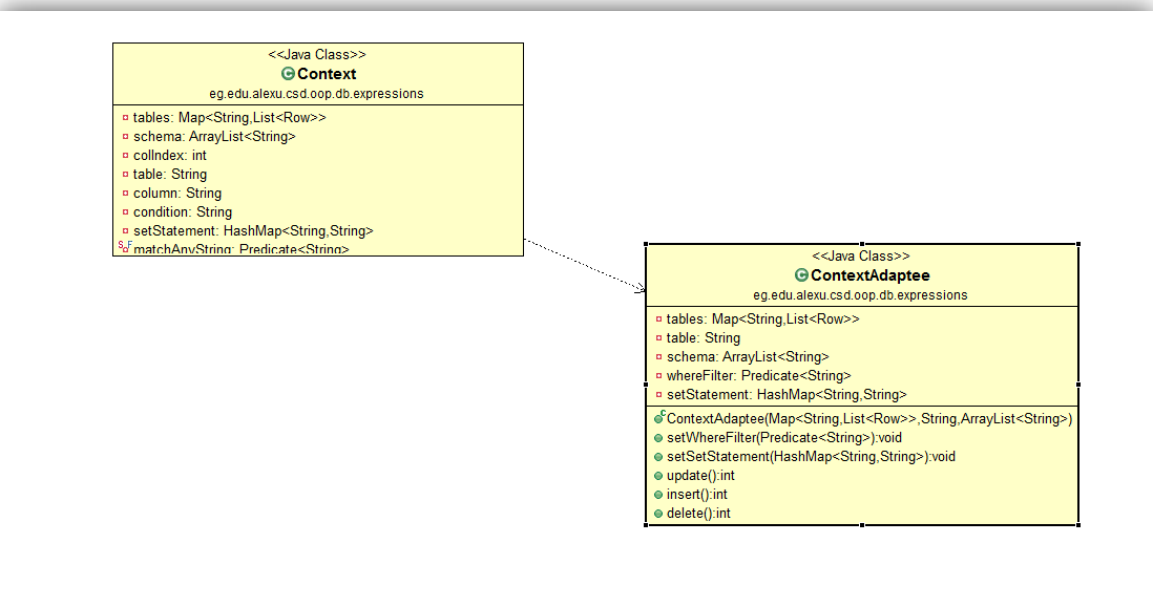
Façade:



Expression:



Adapter:



User Guide:

- The application is based on SQL queries.
- To start you need to create a new database.
- If the chosen name is an existing one you should choose whether you need to overwrite it or updating it.
- To move to another created database, create a database with the required name and set the flag to "false".

```
Enter a SQL query:
Create Database myDatabase
In case the entered name exists already
Enter:
'true' to create a new Database
'false' to use the existing Database

true
```

- To create table specify the names of the columns and the data type of each column (varchar or int).

```
Enter a SQL query:
Create table table1 (name varchar, surname varchar, id int, score int)
```

- To insert new rows in an existing table:

```
Enter a SQL query:
insert into table1 (name, surname, id, score) values (karim, mohamed, 4557, 130)
Number of updated rows: 1
Enter a SQL query:
insert into table1 values (youssef, abdallah, 4541, 135)
Number of updated rows: 1
Enter a SQL query:
insert into table1 (name, id, score) values (michael, 4100, 137)
Number of updated rows: 1
```

- To select from the table:

```
Enter a SQL query:
select * from table1
karim mohamed 4557 130
youssef abdallah 4541 135
michael null 4100 137
Enter a SQL query:
select name, surname from table1
karim mohamed
youssef abdallah
michael null
Enter a SQL query:
select name from table1 where id=4557 or id=4100
karim
michael
```

- To update rows in a table:

```
Enter a SQL query:
update table1 set score=150 where name=karim
Number of updated rows: 1
Enter a SQL query:
select score from table1 where name=karim
150
Enter a SQL query:
update table1 set score= 150
Number of updated rows: 3
Enter a SQL query:
select * from table1
karim mohamed 4557 150
youssef abdallah 4541 150
michael null 4100 150
```

- To delete a row from a table:

```
delete from table1 where name=karim
Number of updated rows: 1
Enter a SQL query:
select * from table1
youssef abdallah 4541 150
michael null 4100 150
```

- To drop a table or drop a database:

```
Drop Table table1
Enter a SQL query:
Drop Database myDatabase
```