CSE441: DATABASE SYSTEMS
ASSIGNMENT 1

In this assignment, you are supposed write a mini-sql engine which will run a subset of SQL Queries using **command line interface**.

Programming Languages Allowed : Java and C/C++

Dataset:

1. csv files for tables.
    a. If a file is : *File1.csv*, the table name would be File1.
    b. There will be no tab-separation or space-separation, so you are not required to handle it but you have to make sure to take care of both csv file type cases: the one where values are in double quotes and the one where values are without quotes.
2. All the elements in files would be **only INTEGERS**.
3. A file named: **metadata.txt**(note the extension) would be given to you which will have the following structure for each table:

<begin_table>
<table_name>
<attribute1>
....
<attributeN>
<end_table>

Type of Queries: You'll be presented with the following set of queries:

1. Select all records : Select * from table_name;
2. Aggregate functions: Simple aggregate functions on a single column. Sum, average, max and min. They will be very trivial given that the data is only numbers: select max(col1) from table1;

3. Project Columns(could be any number of columns) from one or more tables : Select col1, col2 from table_name;
4. Select/project with distinct from one table : select distinct(col1), distinct(col2) from table_name;
5. Select with where from one or more tables: select col1,col2 from table1,table2 where col1 = 10 AND col2 = 20;
    a. In the where queries, there would be a maximum of one AND/OR operator with no NOT operators.
6. Projection of one or more(including all the columns) from two tables with one join condition :
    a. select * from table1, table2 where table1.col1=table2.col2;
    b. select col1,col2 from table1,table2 where table1.col1=table2.col2;
    c. NO REPETITION OF COLUMNS – THE JOINING COLUMN SHOULD BE PRINTED ONLY ONCE.
7. **IMPORTANT:**
    a. ERROR HANDLING: 10% marks will be for error handling.
    b. For the above queries, please note all the permutations and combinations of SQL that MySQL permits, specially when it comes to multiple tables. What is mentioned above are examples of what the queries could be.

8. **Parser: You can use pre-built parsers for SQL queries**
9. **Format of Input**
    1. Command lines input such that: {compiled files} “SQL Query”. Here SQL Query would be a command line argument. Example :
        a. For C++ it will be – ./a.out “select * from table_name where condition”
        b. For Java it will be – java classfile.class “select * from table_name where condition”

10. Format of Output:

<Table1.column1, Table1.colum2….TableN.columnM>
Row1
Row2
…….
RowN

11. Sample Input and Sample Output:

A small test set will be available here soon.

12. Deliverables:

1. Java/C++ Source Code files.
2. Compiled Java/C++ files.
3. Bash script as your roll number.
4. Keep all of this in a folder Roll-Number and zip it. ONLY ZIP FILES.
5. UPLOAD ONLY ZIP FILE ON MOODLE.
6. **Copying in the Assignments can lead from a *Zero* in the current assignment to an F in the course.**

DEADLINE: 9:00 pm, 28th August,2015.