Building a Simple Query Processor that Evaluates a SQL Query in Select- From-Where

The idea behind this work is to process the SQL query in the same way SQL Server does, but without using SQL.

Let say we have been given below query.

```
Select Fname, Lname, SSN, Dno, Dname
From Employee, Department
Where Dno = 5 and Dno = DNumber;
```

There are multiple different ways to evaluate the above query. The query string in a text is parsed, built, and transformed to a query execution steps below (shown in Relational Algebra) in a tree structure.

STEP1: EMPS_DNO5 ← $\sigma_{dno=5}$ (EMPLOYEE)

STEP2: EMP_DEPT_DNO5 ← (EMPS_DNO5 $\bowtie_{DNO=DNUMBER}$ DEPARTMENT)

STEP3: EMP_DEPT_ MGR_DEPENDENT
← $\pi_{FNAME, LNAME, SSN, DNO, DNAME}$ (EMP_DEPT_DNO5)

But, for the simplicity, we assume that out query execution step is given in a text file (**inputFile.txt**) where the actual given query is transformed by SQL optimizer to the following query execution steps in a list of lines of Operations Steps in text below.

```
Selection InTable_Name Selection_Conditions OutTab_Name1
Nested Loop Join LTable RTable Join_Conditions OutTab_Name2
Projection InTable_Name2 ProjectColumn_List OutTab_Name3
```

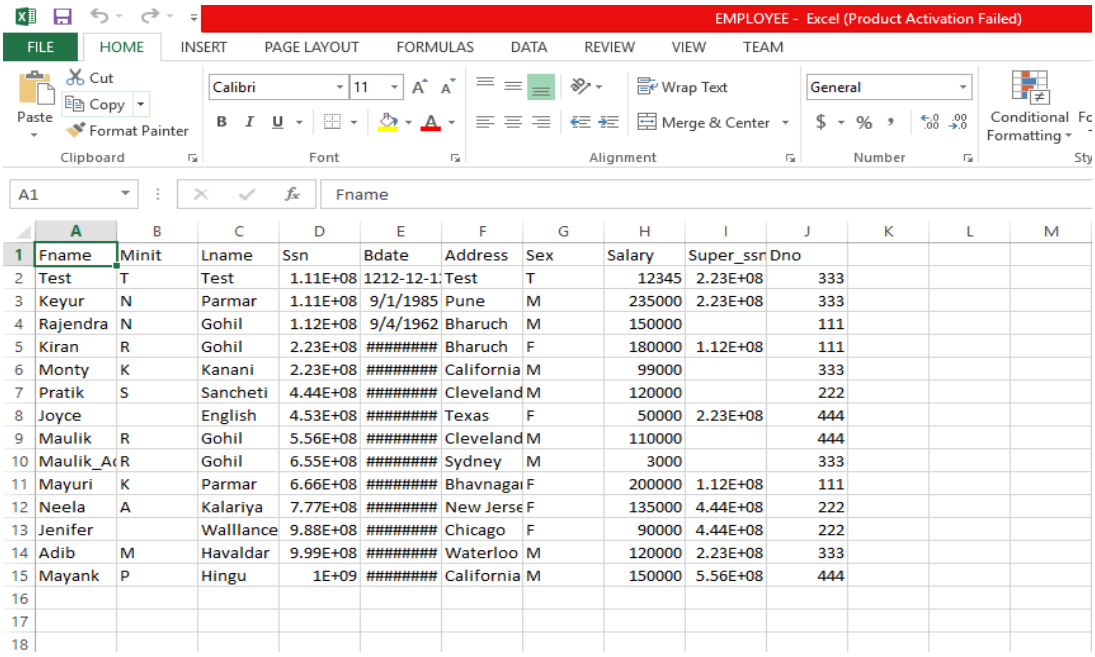
So, here I have implemented a query evaluator that takes a query execution steps (Query execution plan) in format from an input file and performs the three relational operators in the sequence (from top to bottom) as written in the input file to evaluate the given query at beginning of the page.

Process which I followed to get the required output using JAVA program.

1. First of all, I created two required csv file EMPLOYEE and DEPARTMENT.
2. I have made a Text file named “inputFile” for the Query Processing Plan.
3. From that input file, each row will be read by my JAVA program and split each part to get the Source and Destination Table, which operation to perform and which condition to apply. I stored all these values in the different variables.
4. So there will be a while loop which runs until there is no further row in the inputFile and according to the operation, there is IF condition for each of them.
5. Every time, the output of the first operation is stored in the new CSV file and that file is used for the further operations.
6. I have attached source CSV files (EMPLOYEE, DEPARTMENT, inputFile) as well as the output of each operations (EMPS_DNO5, EMP_DEPT_DNO5, EMP_DEPT_MGR_DEPENDENT). Moreover I have also attached the Java project which contains the source code for the Query Processing.
7. In my DEPARTMENT table, there is only four Department having Dnumber = 111, 222, 333, 444. So *I have done all the Calculation choosing the* ***Dnumber = 444***

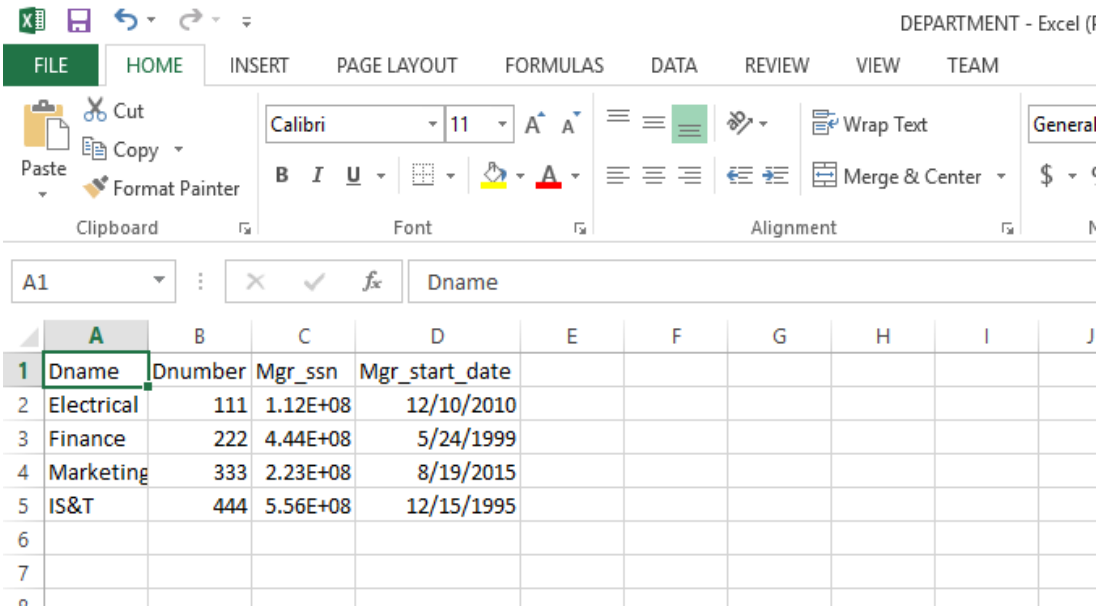
CONTENT of the each file.

EMPLOYEE



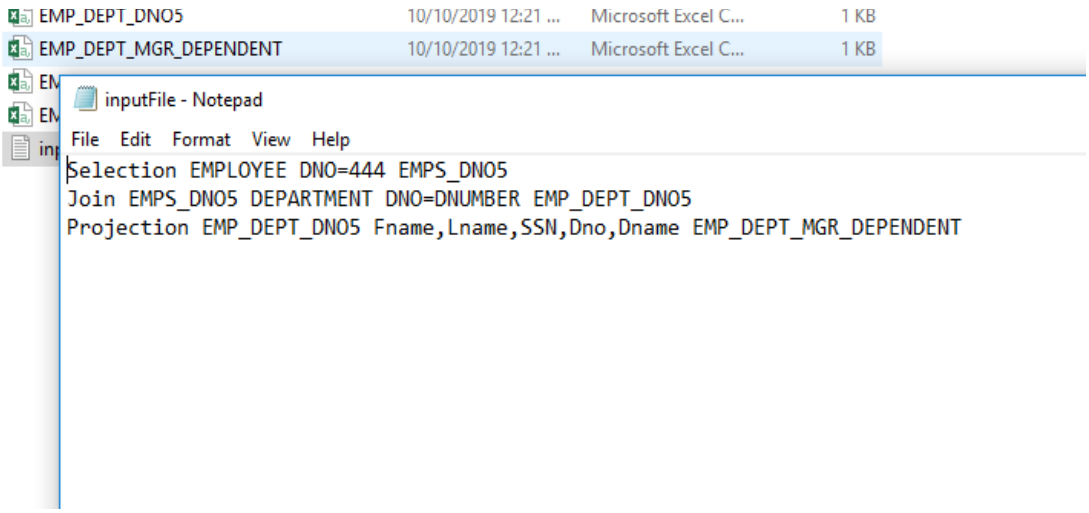
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno			
2	Test	T	Test	1.11E+08	1212-12-1	Test	T	12345	2.23E+08	333			
3	Keyur	N	Parmar	1.11E+08	9/1/1985	Pune	M	235000	2.23E+08	333			
4	Rajendra	N	Gohil	1.12E+08	9/4/1962	Bharuch	M	150000		111			
5	Kiran	R	Gohil	2.23E+08	########	Bharuch	F	180000	1.12E+08	111			
6	Monty	K	Kanani	2.23E+08	########	California	M	99000		333			
7	Pratik	S	Sancheti	4.44E+08	########	Cleveland	M	120000		222			
8	Joyce		English	4.53E+08	########	Texas	F	50000	2.23E+08	444			
9	Maulik	R	Gohil	5.56E+08	########	Cleveland	M	110000		444			
10	Maulik_A	R	Gohil	6.55E+08	########	Sydney	M	3000		333			
11	Mayuri	K	Parmar	6.66E+08	########	Bhavnaga	F	200000	1.12E+08	111			
12	Neela	A	Kalariya	7.77E+08	########	New Jerse	F	135000	4.44E+08	222			
13	Jenifer		Walllance	9.88E+08	########	Chicago	F	90000	4.44E+08	222			
14	Adib	M	Havaldar	9.99E+08	########	Waterloo	M	120000	2.23E+08	333			
15	Mayank	P	Hingu	1E+09	########	California	M	150000	5.56E+08	444			
16													
17													
18													

DEPARTMENT

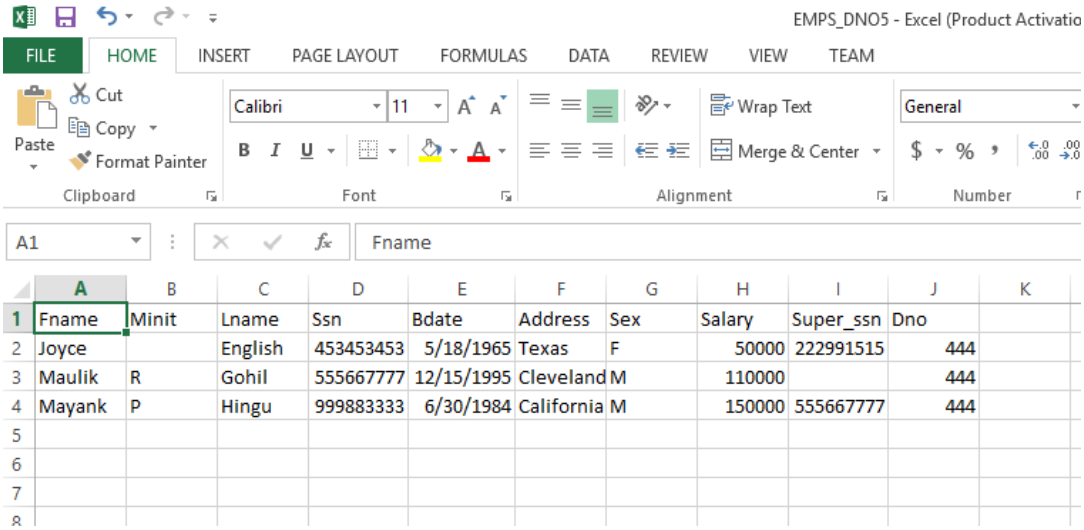


	A	B	C	D	E	F	G	H	I	J
1	Dname	Dnumber	Mgr_ssn	Mgr_start_date						
2	Electrical	111	1.12E+08	12/10/2010						
3	Finance	222	4.44E+08	5/24/1999						
4	Marketing	333	2.23E+08	8/19/2015						
5	IS&T	444	5.56E+08	12/15/1995						
6										
7										

inputFile

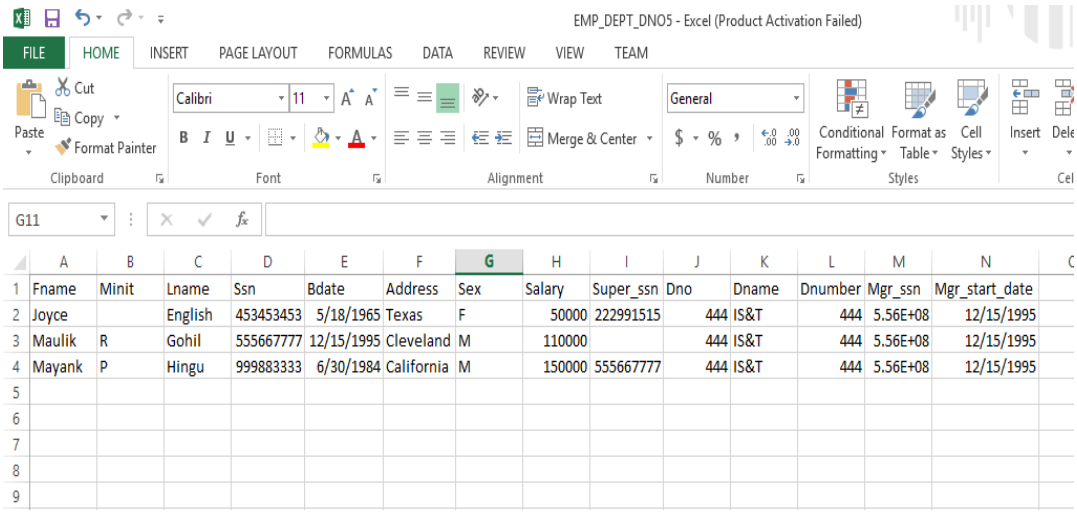


EMPS_DNO5



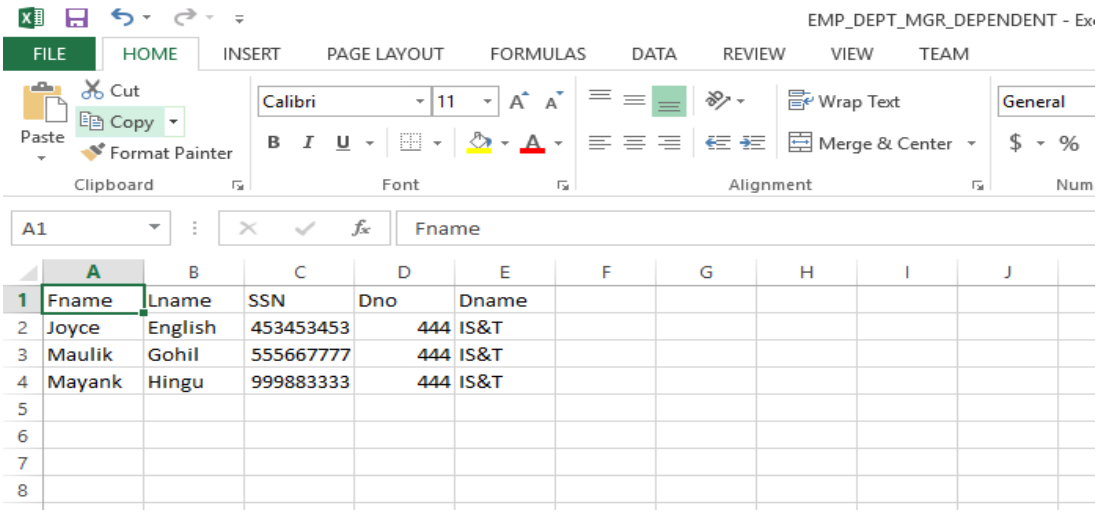
	A	B	C	D	E	F	G	H	I	J	K
1	Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno	
2	Joyce		English	453453453	5/18/1965	Texas	F	50000	222991515	444	
3	Maulik	R	Gohil	555667777	12/15/1995	Cleveland	M	110000		444	
4	Mayank	P	Hingu	999883333	6/30/1984	California	M	150000	555667777	444	
5											
6											
7											
8											

EMP_DEPT_DNO5



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno	Dname	Dnumber	Mgr_ssn	Mgr_start_date
2	Joyce		English	453453453	5/18/1965	Texas	F	50000	222991515	444	IS&T	444	5.56E+08	12/15/1995
3	Maulik	R	Gohil	555667777	12/15/1995	Cleveland	M	110000		444	IS&T	444	5.56E+08	12/15/1995
4	Mayank	P	Hingu	999883333	6/30/1984	California	M	150000	555667777	444	IS&T	444	5.56E+08	12/15/1995
5														
6														
7														
8														
9														

EMP_DEPT_MGR_DEPENDENT (FINAL OUTPUT)



	A	B	C	D	E	F	G	H	I	J
1	Fname	Lname	SSN	Dno	Dname					
2	Joyce	English	453453453	444	IS&T					
3	Maulik	Gohil	555667777	444	IS&T					
4	Mayank	Hingu	999883333	444	IS&T					
5										
6										
7										
8										