**Views**

In mySql, a view refers to a virtual table based on the results of a mySQL statements.It comprises rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database. We can add mySql functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table.

mySql CREATE VIEW Syntax

*CREATE VIEW view\_name AS  
SELECT column\_name(s)  
FROM table\_name  
WHERE condition*

The application contains majorly one view named “invoiceview”



The view “invoiceview” creates a view for the user to extract data from all the tables in the database dbteragigs to get an overall view of the invoice entries into database.It has join statements for header\_mstr, supplier\_mstr, invoice\_mstr, detailinfo, summary\_mstr and address\_mstr.

**Stored Procedures**

Stored Procedures lets us define an API for databases. Reusing this API then becomes easier in multiple applications and programming languages. This technique avoids duplicating database code, saving time and effort when you make updates due to schema changes, tune the performance of queries, or add new database operations for logging, security, and so on.

Create Procedure Syntax

*DELIMITER //*

*CREATE PROCEDURE procedureName(parameters)*

*BEGIN*

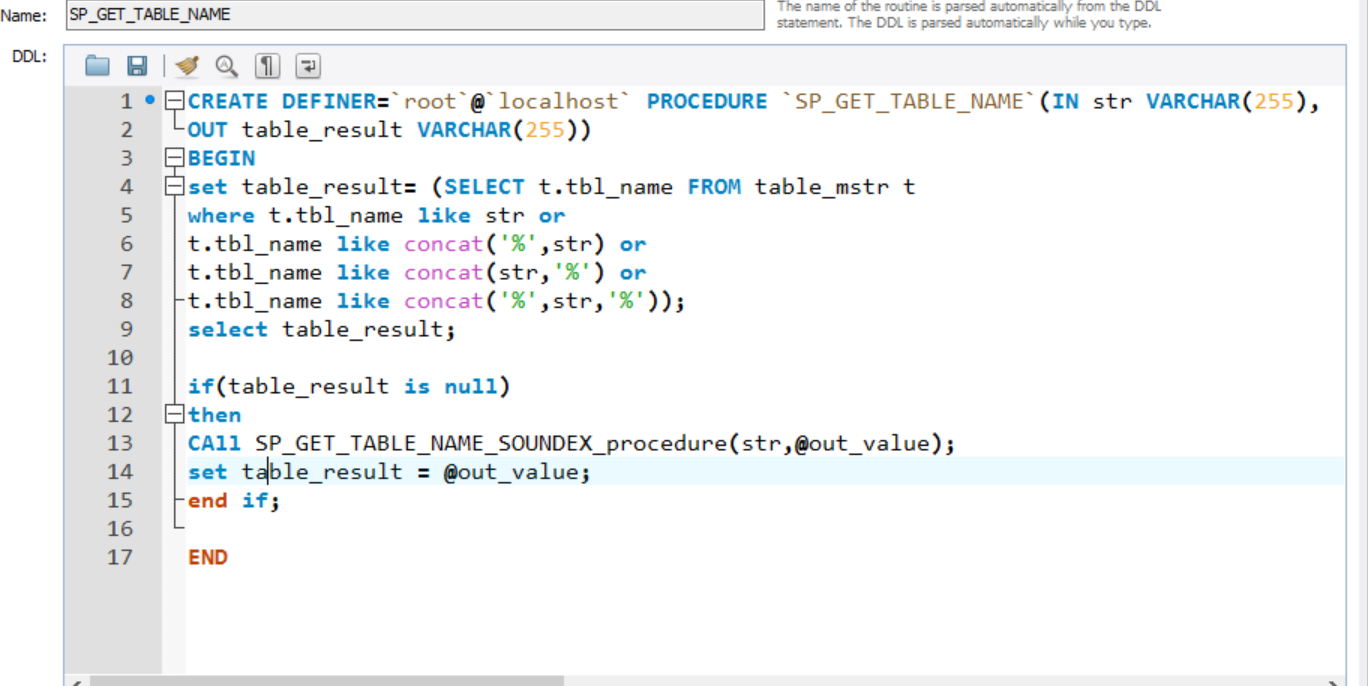
*//Queries*

*END //*

*DELIMITER ;*

The project includes 3 routines for fuzzy pattern matching required for extracting table names and column names and also creating new columns if they do not exist.

The following stored procedure is called through Java code for extracting table names through pattern matching-



The following steps -

SP\_GET\_TABLE\_NAME takes in the level-2 tag name as a varchar parameter from the file format and output parameter which results in the table name after pattern Matching.

Pattern Matching includes –

1) Keyword Matching

2) Phonetic/Sound Matching

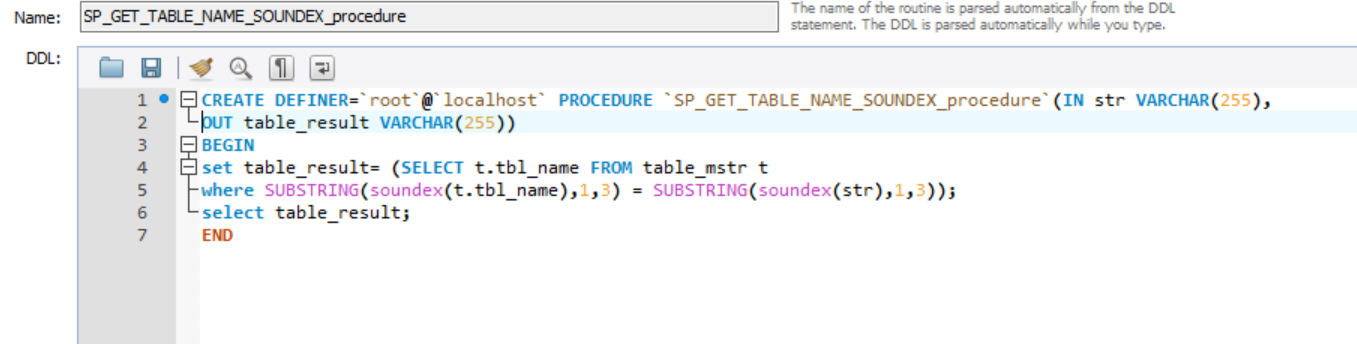
**1) Keyword Matching:-**

The MySQL LIKE condition allows wildcards to be used in the[WHERE clause](http://www.techonthenet.com/mysql/where.php) of a [SELECT](http://www.techonthenet.com/mysql/select.php), [INSERT](http://www.techonthenet.com/mysql/insert.php), [UPDATE](http://www.techonthenet.com/mysql/update.php), or [DELETE](http://www.techonthenet.com/mysql/delete.php) statement. This allows you to perform pattern matching.

For example, if string name that is, tag name is ‘header’ or ‘head’ ,this kind of pattern matching will extract the table ‘header\_mstr’.

**2) Phonetic/Sound matching:-**

Further in the stored procedure, if the compared string is null the following stored procedure is called **–**

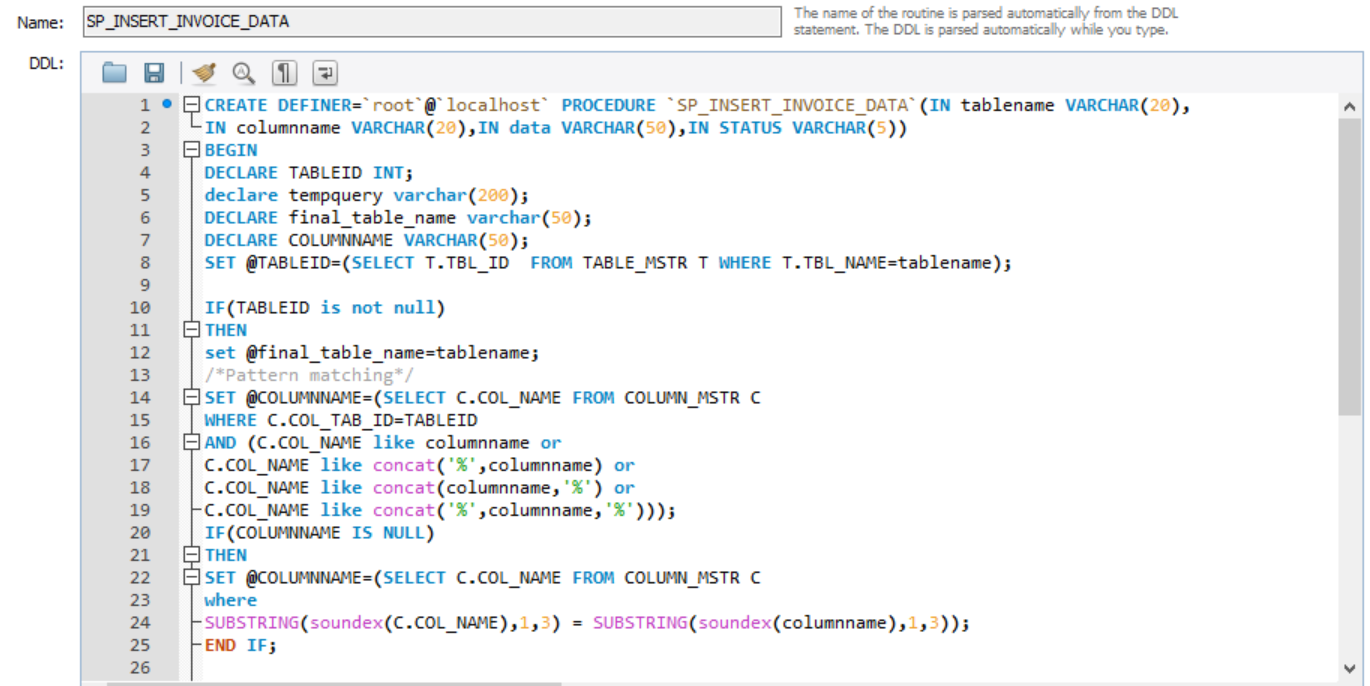


**Soundex** is a phonetic algorithm for indexing names by sound, as pronounced in English. The goal is for homophones to be encoded to the same representation so that they can be matched despite minor differences in spelling.

Soundex is a phonetic algorithm for indexing names by sound, as pronounced in English.  SOUNDEX codes from different strings can be compared to see how similar the strings sound when spoken. The first character of the code is the first character of the expression, converted to upper case. The second through fourth characters of the code are numbers that represent the letters in the expression. The letters A, E, I, O, U, H, W, and Y are ignored unless they are the first letter of the string.  All international alphabetic characters outside the A-Z range are treated as vowels.  Hence, two strings that sound almost the same should have identical soundex strings.  For instance, the words "Assistance" and "Assistants" both produce a soundex of “A223”.

In this stored procedure, substring of every sound is calculated and accordingly, returns sound matched string table name.

* The same logic code applies for column name extraction.



* Further, after pattern matching is done ,if the column name doesn’t exist in the table ,the table is altered to add a new column.