**Session 29**

**Assignment 3**

**Problem Statement:**

Explain Brief of the following in brief

* Hive UDF
* Hive UDAF
* Hive UDTF
* Thrift server
* **Hive UDF:**

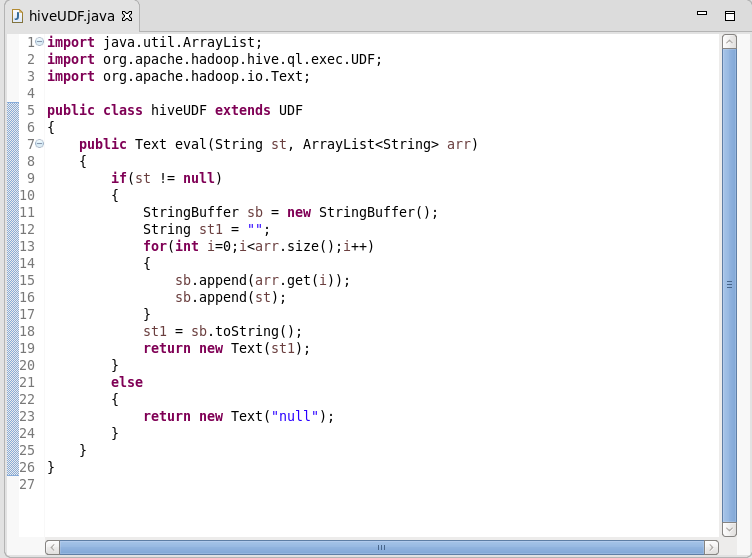
UDF is User Defined Function that works on a single row in a table and produces a single row as output. When used in queries, it can be called once for each row in the result set. It is one to one relationship between input and output of a function.

**User Defined Functions (UDFs):**

* Extends the functionality of Hive by writing functions that can be evaluated in HiveQL.
* Customize serializer and/or deserializer, which provides a way to either deserialize a custom file format stored in HDFS.
* Customizes mappers and/or reducers, which allows to add custom map or reduce steps into the Hive query.
* These map or reduce steps can be written in any programming language, not just in Java.
* Since the Hadoop framework is written mostly in Java language, naturally most of the Hadoop developers prefer Java to write the UDFs.
* However, Apache has also made it easy for non-Java developers to be able to work on Hadoop which is done using the Hadoop Streaming Interface.

**Example:**

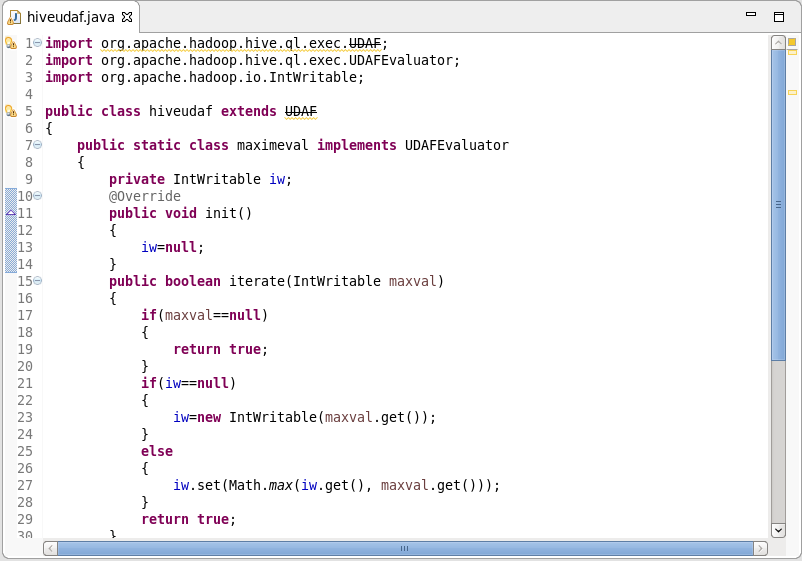
A java code is written to concatenate the given string or strings. This UDF will accept two arguments, one string and one array of string. It will return a single string where all the elements of the string array are concatenated along with the string (separator).

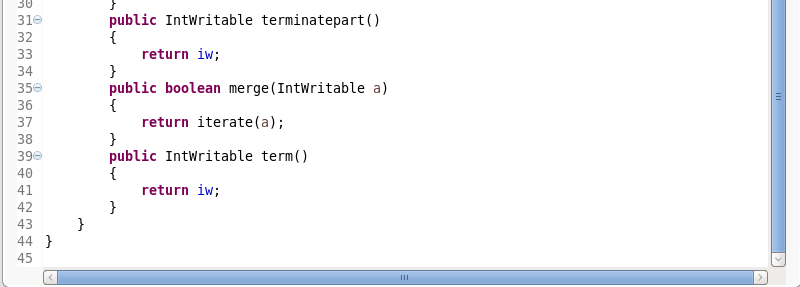


* **Hive UDAF:**
* User Defined Aggregation Functions (UDAFs) are an exceptional way to integrate advanced data-processing into Hive.
* User defined aggregate functions works on more than one row and gives single row as output.
* Aggregate functions perform a calculation on a set of values and return a single value.
* An aggregate function is more difficult to write than a regular UDF.
* Values are aggregated in chunks (potentially across many tasks), so the implementation has to be capable of combining partial aggregations into a final result.
* This function is typically used in the “group by” case.
* Hive allows us to define our own UDAFs.

**Example:**

A java code is written to give the maximum value from a selected column. Here when the code is executed with a column from a table then it will return the maximum value from that particular column.



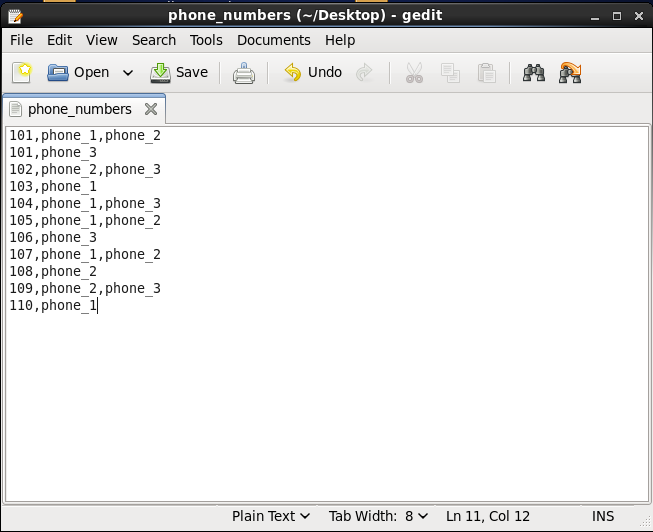


* **Hive UDTF:**
* UDTF is a User Defined Table Generating Function that works on one row as input and returns multiple rows (a table) as output. So here the relation in one to many.
* This is exactly opposite to the UDAF.
* UDTF can be used to split a column into multiple column as well. Here an alias "AS" clause is mandatory.
* This is more similar to hive built-in Explode () function.

**Example:**

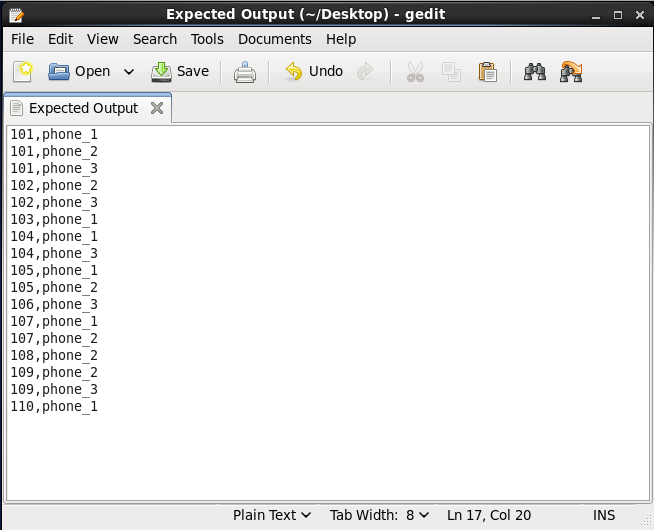
**Before using UDTF:**

The data in the file will not be in an arranged manner before using the Hive UDTF on this file.



**After using UDTF:**

If Hive UDTF is used on the above data file, then the file will be converted like the file shown below.



* **Thrift Server:**
* The Apache Thrift software framework, for scalable cross-language services development, combines a software stack with a code generation engine to build services that work efficiently and seamlessly between C++, Java, Python, PHP, Ruby, Erlang, Perl, Haskell, C#, Cocoa, JavaScript, Node.js, Smalltalk, OCaml and Delphi and other languages.
* Thrift is an RPC framework for building cross-platform services. Its stack consists of 4 layers: Server, Transport, Protocol, and Processor.
* When you query any hive tables or database, in background automatically your requests is transferred between hive service and hive server
* When you want to create your own service or project you can use thrift protocols which will help you in creating layers, think this as you are creating your user defined functions using libraries, so in that case libraries will be thrift.