ASSIGNMENT 8.2

Problem Statement:

Write a hive UDF that implements functionality of string concat_ws (string SEP, array<string>).

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 array are separated by the SEP.

Solution:

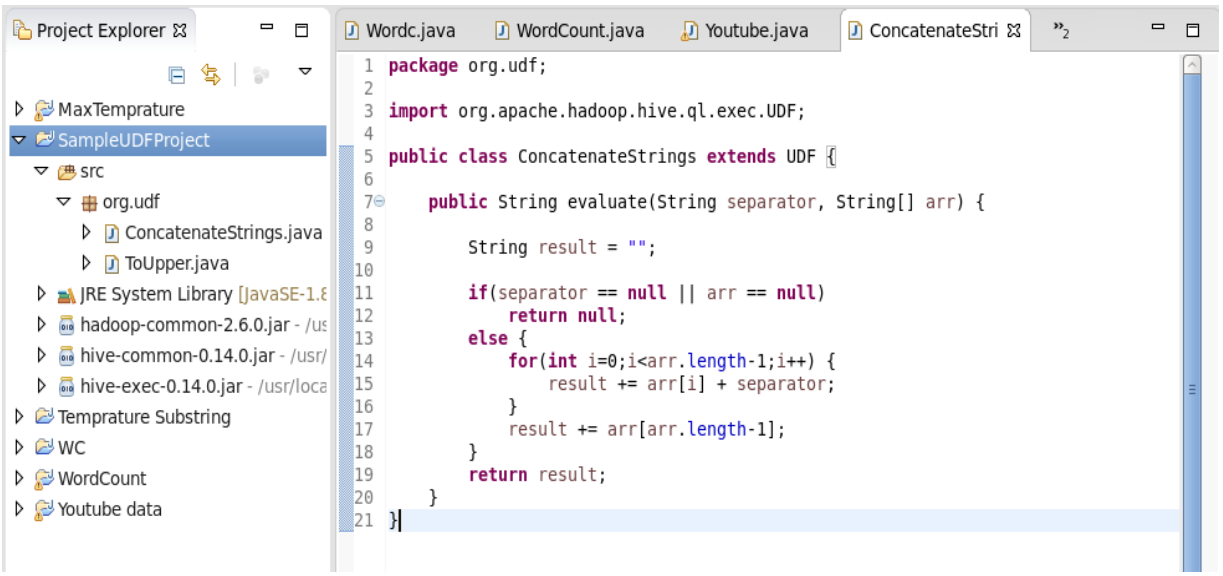
Step 1: Write Java code to implement the functionality of concatenating two or more strings.

We need to extend the class, 'UDF' and override evaluate() method from this class. Here is the code that I have written to implement this functionality:

ConcatenateStrings.java

```
package org.udf;

import org.apache.hadoop.hive.ql.exec.UDF;

public class ConcatenateStrings extends UDF {

	public String evaluate(String separator, String[] arr) {

		String result = "";

		if(separator == null || arr == null)
			return null;
		else {
			for(int i=0;i<arr.length-1;i++) {
				result += arr[i] + separator;
			}
			result += arr[arr.length-1];
		}
	return result;
	}
}
```

After checking for null input values, we are concatenating the separator character to the end of every string element from given array, storing it in a result variable and we finally append the last array element to the result and return it.



Step 2: Export this project as .jar file and add the same in Hive.

Here is the command to add the jar file in Hive:

ADD JAR /home/acadgild/workspace/myUDF.jar;

```
hive> ADD JAR /home/acadgild/workspace/SampleUDFProject.jar;
Added [/home/acadgild/workspace/SampleUDFProject.jar] to class path
Added resources: [/home/acadgild/workspace/SampleUDFProject.jar]
```

Step 3: Create a temporary function by referring to the Java class shown above.

Here is the command to do the same:

CREATE TEMPORARY FUNCTION concat AS 'org.udf.ConcatenateStrings';

```
hive> CREATE TEMPORARY FUNCTION concat AS 'org.udf.ConcatenateStrings';
OK
Time taken: 0.833 seconds
hive>
```

Step 4: Call this function by passing few fields of a table along with a separator character.

Let's see the contents of the table, 'smartphones_data':

```
hive> SELECT * FROM smartphones_data;
OK
101     Samsung S7 Edge 30000
102     Apple   iPhone X        89000
103     Motorola        G5 Plus 15000
101     Samsung S7 Edge 30000
102     Apple   iPhone X        89000
103     Motorola        G5 Plus 15000
Time taken: 0.845 seconds, Fetched: 6 row(s)
```

We can append phone manufacturer field with phone model leaving a space in between them.

Here is the query:

SELECT concat(' ', manufacturer, phone_model)

FROM smartphones_data;

Output:

```
hive> SELECT concat(' ', manufacturer, phone_model)
    > FROM smartphones_data;
OK
Samsung S7 Edge
Apple iPhone X
Motorola G5 Plus
Samsung S7 Edge
Apple iPhone X
Motorola G5 Plus
Time taken: 1.204 seconds, Fetched: 6 row(s)
hive>
```

Let's add the price as well but this time with comma as a separator:

SELECT concat(',', manufacturer, phone_model, phone_price)

FROM smartphones_data;

Output:

```
hive> SELECT concat(',', manufacturer, phone_model, phone_price)
    > FROM smartphones_data;
OK
Samsung,S7 Edge,30000
Apple,iPhone X,89000
Motorola,G5 Plus,15000
Samsung,S7 Edge,30000
Apple,iPhone X,89000
Motorola,G5 Plus,15000
Time taken: 0.678 seconds, Fetched: 6 row(s)
```

We can add any number of columns from a table of type String to concatenate by simply calling the registered function in a Hive query that fetches data such as SELECT query.