Session 8: ADVANCED HIVE

Assignment 8.2

Student Name: Subham Vishal

Course: Big Data Hadoop & Spark Training

Assignment 8.2– 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.

Contents

Introduction 2

Dataset 2

Prerequisites 2

Create Database and Table 2

HIVE QL 2

HIVE QL 3

HIVE UDF Java code 5

HIVE UDF CONCAT_WS function 5

HIVE QL 6

Required Output 6

Introduction

In this assignment we are going to write HIVE UDF using Java in order to achieve the CONCAT_WS function. For example,

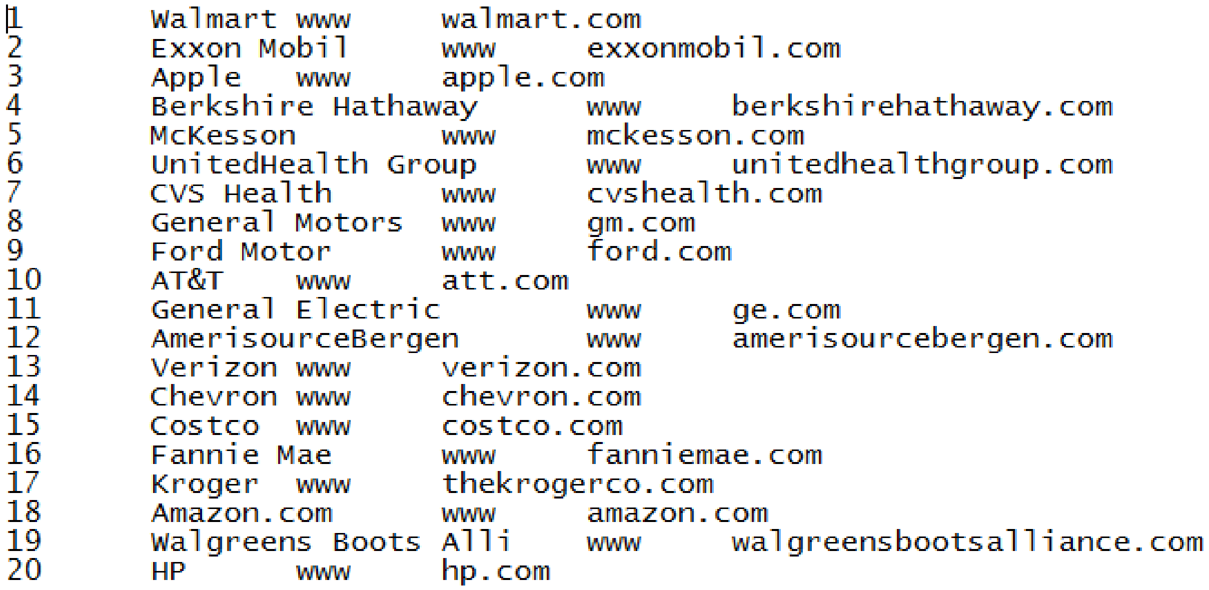
We have fortune 20 companies list and its company website URL, but the 'www' and the remaining domain are separated. In our output we try to achieve the output as below,

1 walmart www.walmart.com

Dataset

The below data contains, the column name as,

Rank, company_name, website, protocal.



```
1       Walmart www     walmart.com
2       Exxon Mobil     www     exxonmobil.com
3       Apple   www     apple.com
4       Berkshire Hathaway      www     berkshirehathaway.com
5       McKesson        www     mckesson.com
6       UnitedHealth Group      www     unitedhealthgroup.com
7       CVS Health      www     cvshealth.com
8       General Motors  www     gm.com
9       Ford Motor      www     ford.com
10      AT&T    www     att.com
11      General Electric        www     ge.com
12      AmerisourceBergen       www     amerisourcebergen.com
13      Verizon www     verizon.com
14      Chevron www     chevron.com
15      Costco  www     costco.com
16      Fannie Mae      www     fanniemae.com
17      Kroger  www     thekrogerco.com
18      Amazon.com      www     amazon.com
19      Walgreens Boots Alli    www     walgreensbootsalliance.com
20      HP      www     hp.com
```

Prerequisites

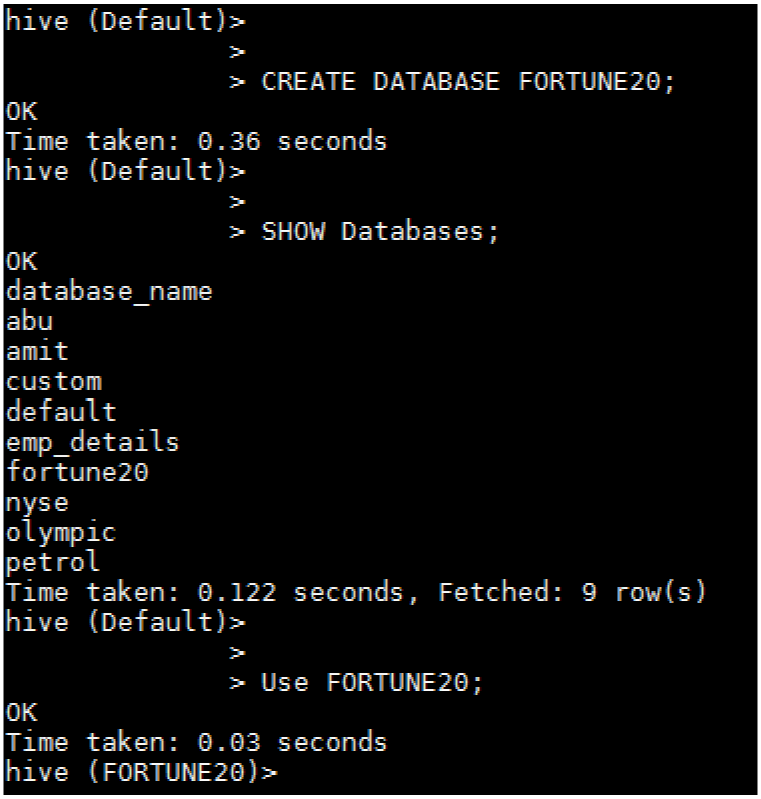
Create Database and Table

Create Database FORTUNE20

HIVE QL

CREATE DATABASE FORTUNE20

Use FORTUNE20;



```
hive (Default)>
              >
              > CREATE DATABASE FORTUNE20;
OK
Time taken: 0.36 seconds
hive (Default)>
              >
              > SHOW Databases;
OK
database_name
abu
amit
custom
default
emp_details
fortune20
nyse
olympic
petrol
Time taken: 0.122 seconds, Fetched: 9 row(s)
hive (Default)>
              >
              > Use FORTUNE20;
OK
Time taken: 0.03 seconds
hive (FORTUNE20)>
```

Create Table Fortune_company

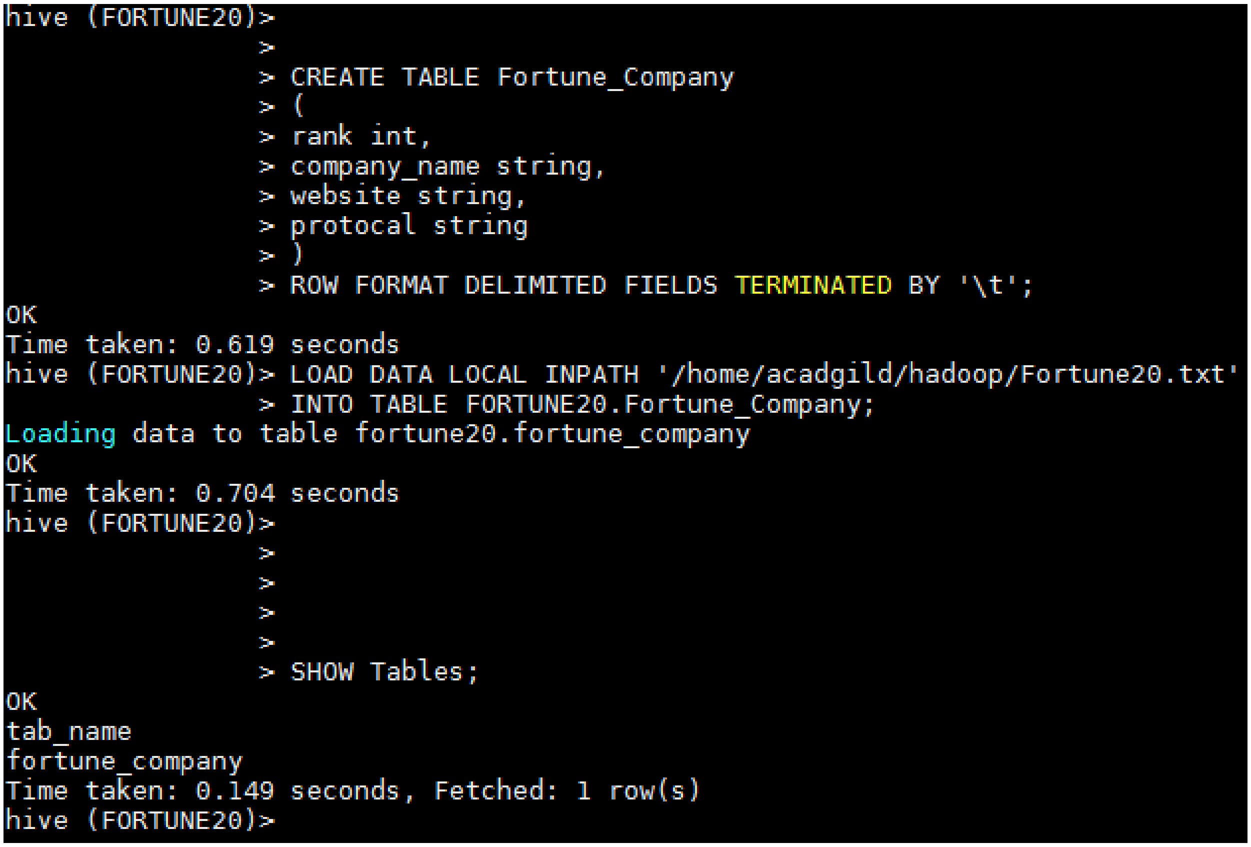
HIVE QL

CREATE TABLE fortune_company(rank int, company_name string,website string, protocal string)

ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t';

LOAD DATA LOCAL INPATH '/home/acadgild/hadoop/fortune20.txt’

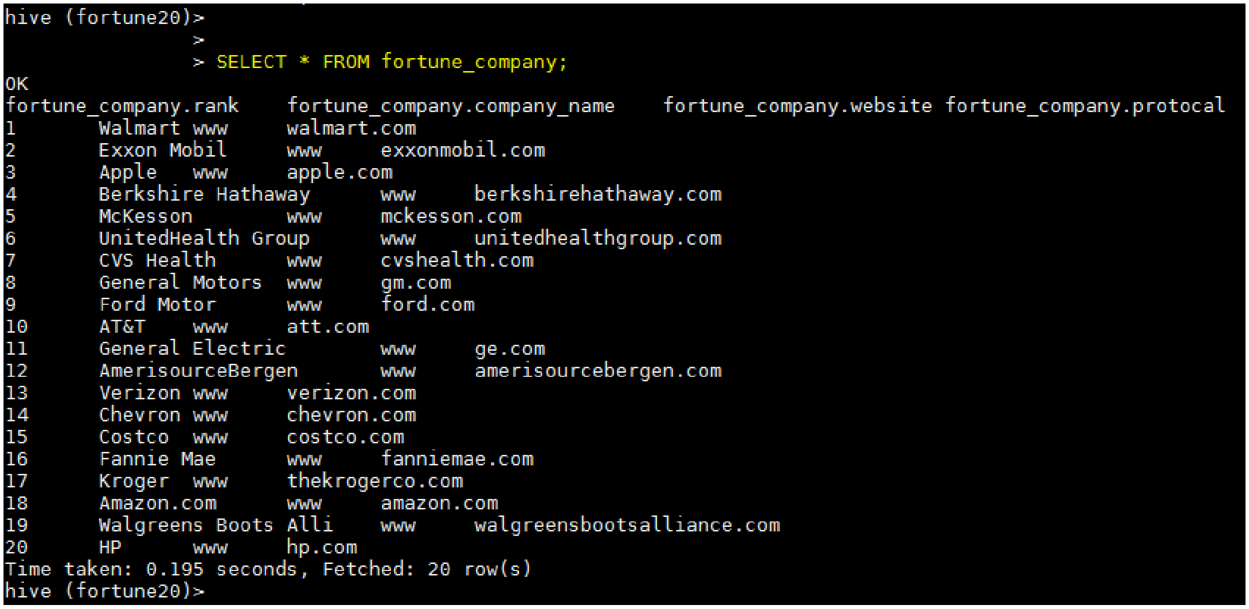
INTO TABLE fortune20.fortune_company;



```
hive (FORTUNE20)>
                > 
                > CREATE TABLE Fortune_Company
                > (
                > rank int,
                > company_name string,
                > website string,
                > protocal string
                > )
                > ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t';
OK
Time taken: 0.619 seconds
hive (FORTUNE20)> LOAD DATA LOCAL INPATH '/home/acadgild/hadoop/Fortune20.txt'
                > INTO TABLE FORTUNE20.Fortune_Company;
Loading data to table fortune20.fortune_company
OK
Time taken: 0.704 seconds
hive (FORTUNE20)>
                > 
                > 
                > 
                > 
                > SHOW Tables;
OK
tab_name
fortune_company
Time taken: 0.149 seconds, Fetched: 1 row(s)
hive (FORTUNE20)>
```

Viewing the data in the table fortune_company,

SELECT * FROM fortune_company;



```
hive (fortune20)>
                > 
                > SELECT * FROM fortune_company;
OK
fortune_company.rank    fortune_company.company_name    fortune_company.website fortune_company.protocal
1       Walmart www     walmart.com
2       Exxon Mobil     www     exxonmobil.com
3       Apple   www     apple.com
4       Berkshire Hathaway      www     berkshirehathaway.com
5       McKesson        www     mckesson.com
6       UnitedHealth Group      www     unitedhealthgroup.com
7       CVS Health      www     cvshealth.com
8       General Motors  www     gm.com
9       Ford Motor      www     ford.com
10      AT&T    www     att.com
11      General Electric        www     ge.com
12      AmerisourceBergen       www     amerisourcebergen.com
13      Verizon www     verizon.com
14      Chevron www     chevron.com
15      Costco  www     costco.com
16      Fannie Mae      www     fanniemae.com
17      Kroger  www     thekrogerco.com
18      Amazon.com      www     amazon.com
19      Walgreens Boots Alli    www     walgreensbootsalliance.com
20      HP      www     hp.com
Time taken: 0.195 seconds, Fetched: 20 row(s)
hive (fortune20)>
```

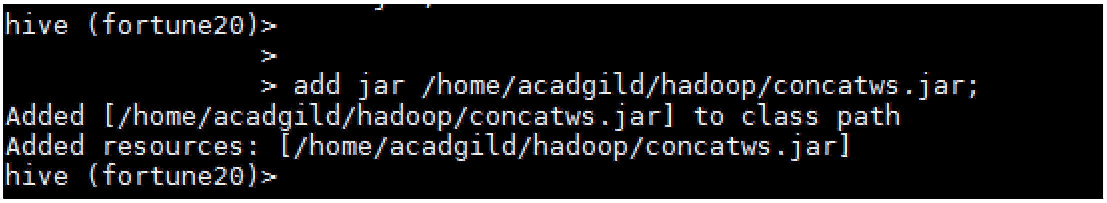
HIVE UDF Java code

```
package concatws;

import org.apache.hadoop.hive.ql.exec.UDF;
import org.apache.hadoop.hive.ql.exec.Description;
@Description(name = "concatws", value = "_FUNC_(string SEP, array<string>) -
RETURN_TYPE(STRING)\n" + "Description: Concatenate two strings, separated by the
seperator",
extended = "Example:\n"
             + "  > SELECT CONCAT_WS (website,’.’,protocal) FROM src;\n"
             + "www.walmart.com")

public class concatws extends UDF

{
      public String evaluate(String param1, String[] param2)

{
      String Output = "";
      if(param1==null && param2==null)
      {
            return null;
      }
      for(int i = 0; i < param2.length; i++)
      {
            Output+= param2[i];
      }
      return(param1.concat(Output));
}
}
```

After that we are adding JAR created from the JAVA class which is defining the UDF using below syntax-

HIVE UDF CONCAT_WS function

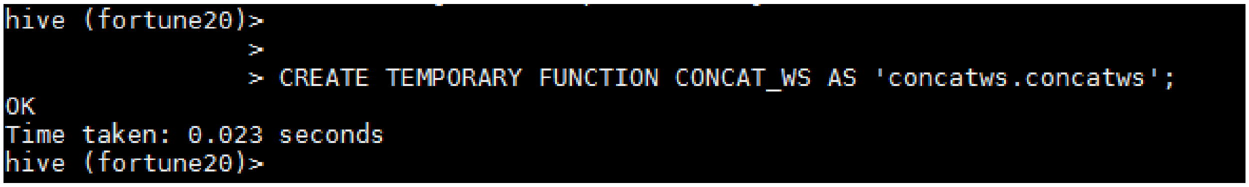
add jar /home/acadgild/hadoop/concatws.jar;



```
hive (fortune20)>
                 >
                 > add jar /home/acadgild/hadoop/concatws.jar;
Added [/home/acadgild/hadoop/concatws.jar] to class path
Added resources: [/home/acadgild/hadoop/concatws.jar]
hive (fortune20)>
```

After that we are creating a temporary function “CONCAT_WS” using below syntax-

CREATE TEMPORARY FUNCTION CONCAT_WS AS 'concatws.concatws';

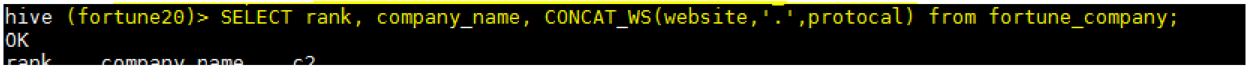


```
hive (fortune20)>
                 >
                 > CREATE TEMPORARY FUNCTION CONCAT_WS AS 'concatws.concatws';
OK
Time taken: 0.023 seconds
hive (fortune20)>
```

After that we run below query to take one column (company_name) input as String and another array(website,'.',protocal) as Array of Strings and concatenate them,

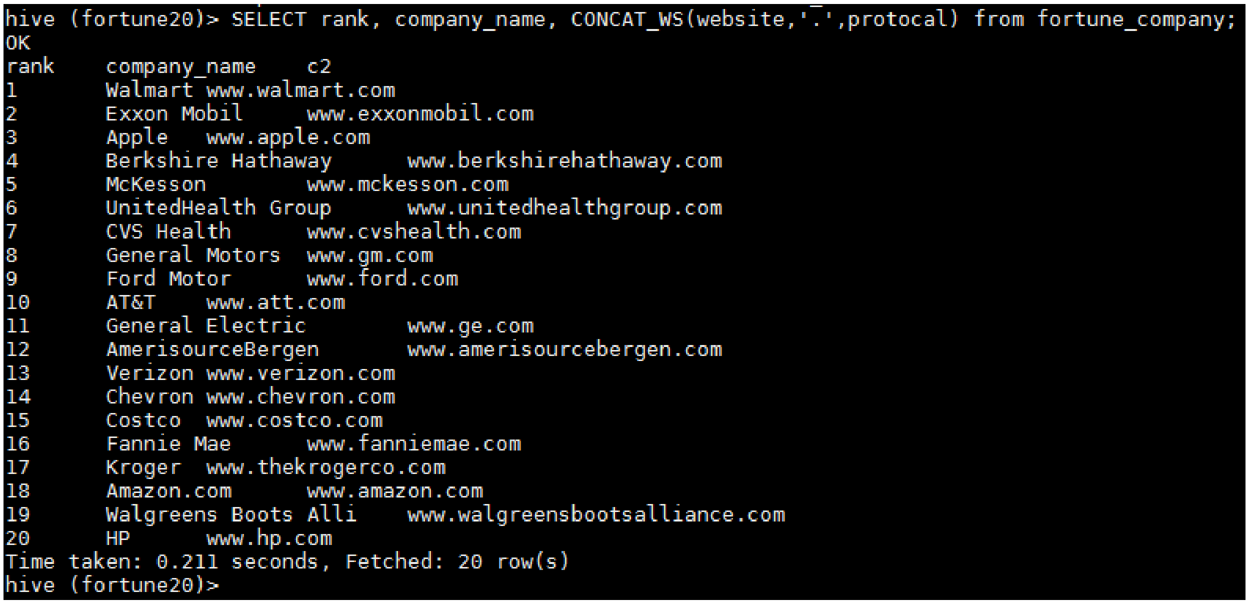
HIVE QL

SELECT rank, company_name, CONCAT_WS(website,'.',protocal) from fortune_company; SELECT rank, company_name, CONCAT_WS(website,'.',protocal) from fortune_company;



```
hive (fortune20)> SELECT rank, company_name, CONCAT_WS(website,'.',protocal) from fortune_company;
OK
rank    company_name    c2
```

Required Output



```
hive (fortune20)> SELECT rank, company_name, CONCAT_WS(website,'.',protocal) from fortune_company;
OK
rank    company_name    c2
1       Walmart www.walmart.com
2       Exxon Mobil     www.exxonmobil.com
3       Apple   www.apple.com
4       Berkshire Hathaway      www.berkshirehathaway.com
5       McKesson        www.mckesson.com
6       UnitedHealth Group      www.unitedhealthgroup.com
7       CVS Health      www.cvshealth.com
8       General Motors  www.gm.com
9       Ford Motor      www.ford.com
10      AT&T    www.att.com
11      General Electric        www.ge.com
12      AmerisourceBergen       www.amerisourcebergen.com
13      Verizon www.verizon.com
14      Chevron www.chevron.com
15      Costco  www.costco.com
16      Fannie Mae      www.fanniemae.com
17      Kroger  www.thekrogerco.com
18      Amazon.com      www.amazon.com
19      Walgreens Boots Alli    www.walgreensbootsalliance.com
20      HP      www.hp.com
Time taken: 0.211 seconds, Fetched: 20 row(s)
hive (fortune20)>
```