Calculate the number of employees corresponding to each skill from the table 'employee' which is loaded in the Demo.

Dataset:
employee.csv

contents of file:
```
101,Amit,HADOOP:HIVE:SPARK:BIG-DATA
102,Sumit,HIVE:OOZIE:HADOOP:SPARK:STORM
103,Rohit,KAFKA:CASSANDRA:HBASE
```

creating table using csv data:

```
hive> show databases;
OK
acadgild_db
custom
default
Time taken: 2.058 seconds, Fetched: 3 row(s)
hive> use custom
    > ;
OK
Time taken: 0.116 seconds
hive> show tables
    > ;
OK
temperature_data
temperature_data_vw
Time taken: 0.444 seconds, Fetched: 2 row(s)
hive> CREATE TABLE employee
    > (
    > id INT,
    > name STRING,
    > skills ARRAY<STRING>
    > )
    > ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY ','
    > COLLECTION ITEMS TERMINATED BY ':';
OK
Time taken: 1.662 seconds
```

Loading employee.csv data to table:

```
hive> LOAD DATA LOCAL INPATH '/Users/Disha/employee.csv' INTO TABLE employee;
Loading data to table custom.employee
OK
Time taken: 2.496 seconds
hive> select * from employee
    > ;
OK
101     Amit    ["HADOOP","HIVE","SPARK","BIG-DATA"]
102     Sumit   ["HIVE","OOZIE","HADOOP","SPARK","STORM"]
103     Rohit   ["KAFKA","CASSANDRA","HBASE"]
Time taken: 2.034 seconds, Fetched: 3 row(s)
```

Query to retrieve the number of employees corresponding to each skill from the table 'employee' **:**

```
hive> select skill, count(name) FROM employee LATERAL VIEW explode(skills) skill_set AS skill GROUP BY skill;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spa
rk, tez) or using Hive 1.X releases.
Query ID = Disha_20170917235508_7d27998e-f1ca-4b51-a987-67c8b7c7b046
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Job running in-process (local Hadoop)
2017-09-17 23:55:12,554 Stage-1 map = 100%,  reduce = 0%
2017-09-17 23:55:13,562 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local637477713_0001
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 216 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
BIG-DATA        1
CASSANDRA       1
HADOOP  2
HBASE   1
HIVE    2
KAFKA   1
OOZIE   1
SPARK   2
STORM   1
Time taken: 4.958 seconds, Fetched: 9 row(s)
```