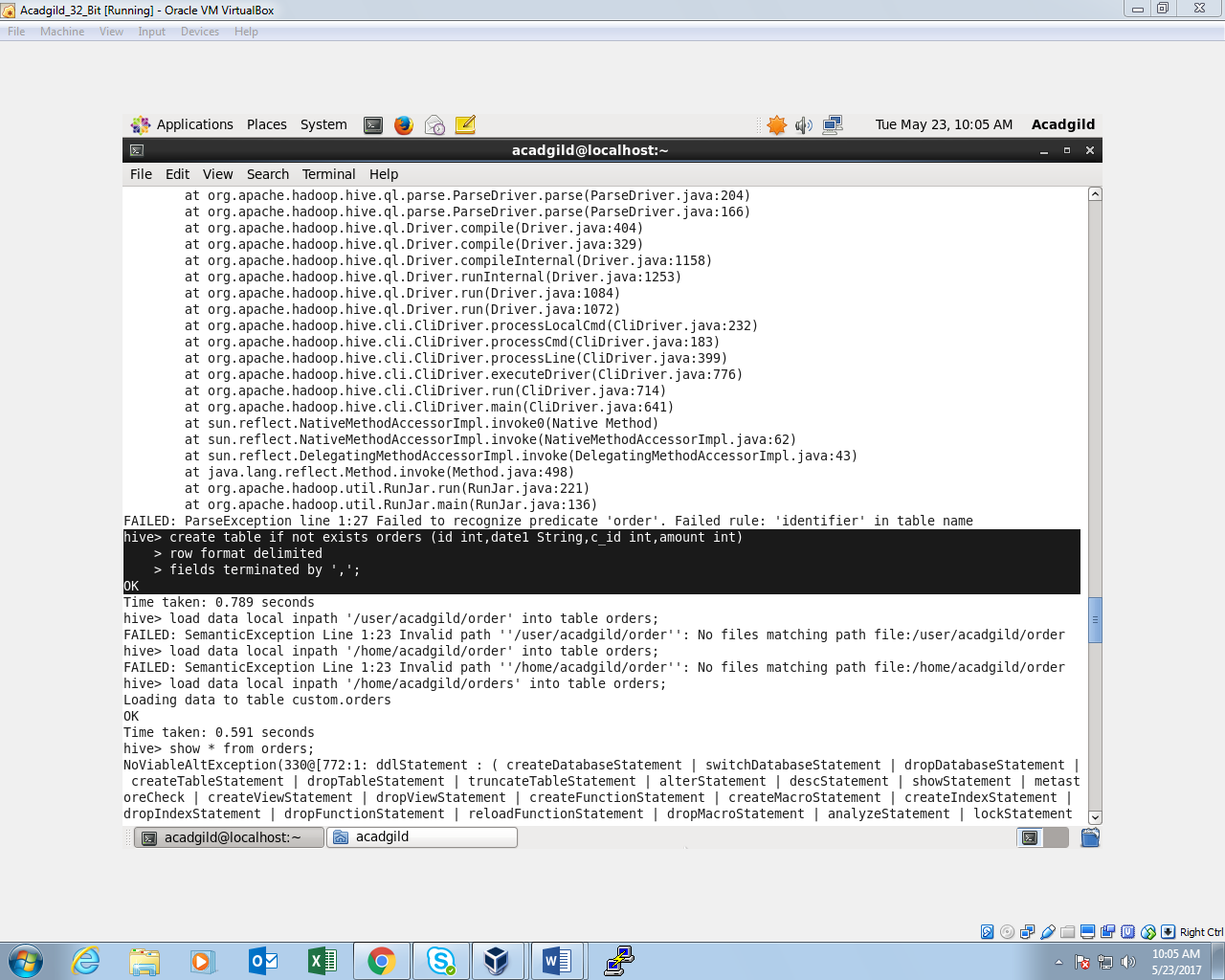
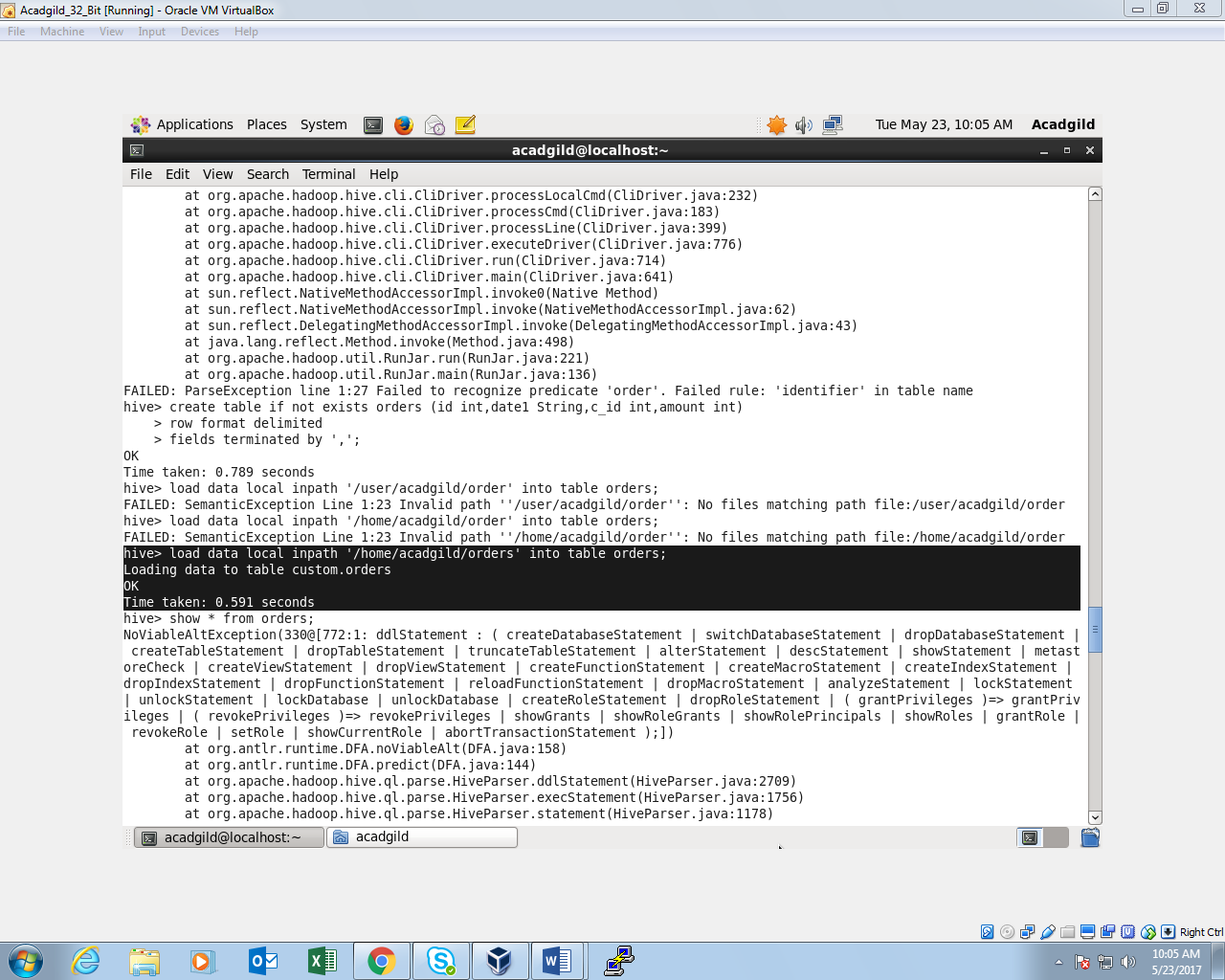
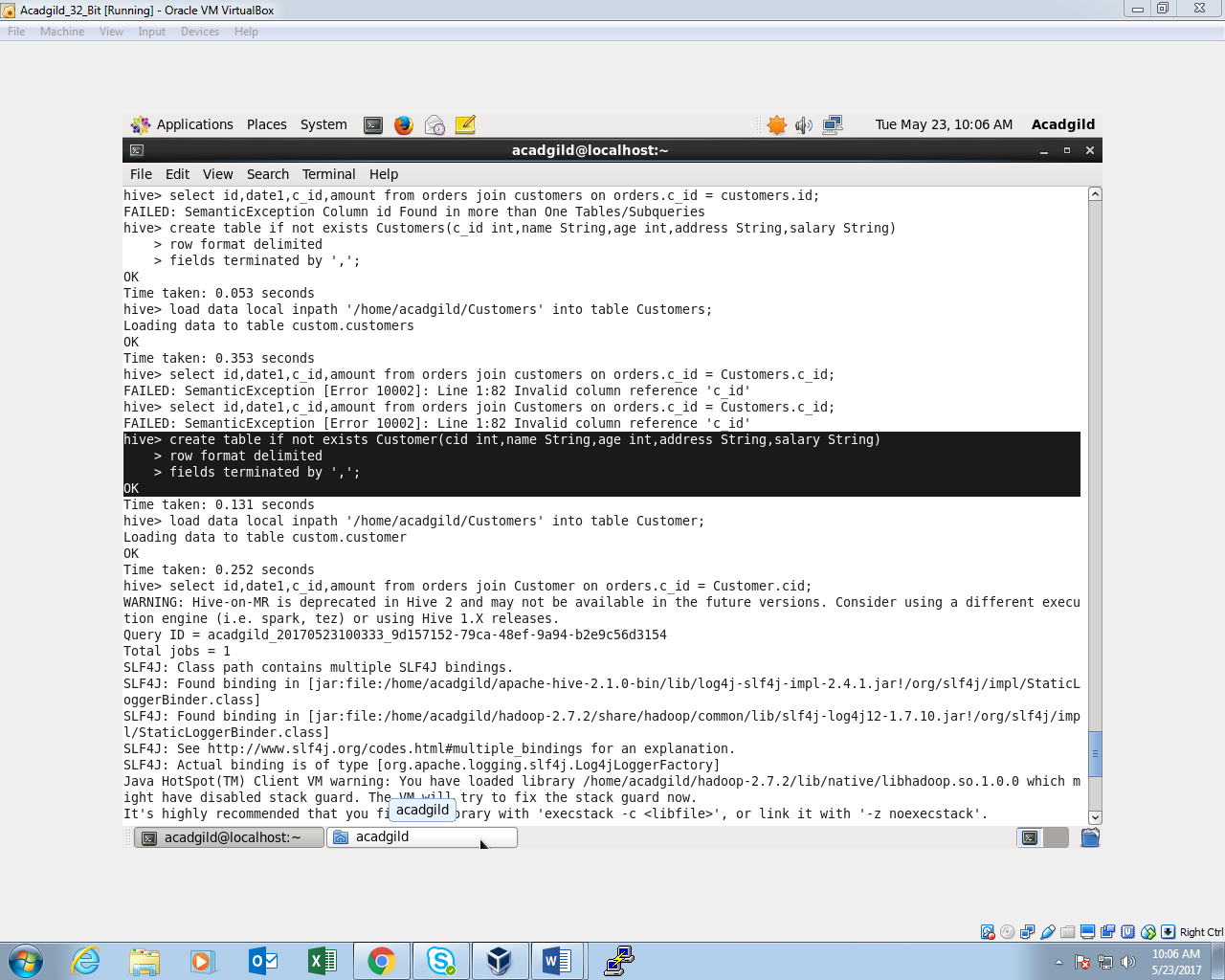
**Explain the following in brief by using the below datasets and their uses.**

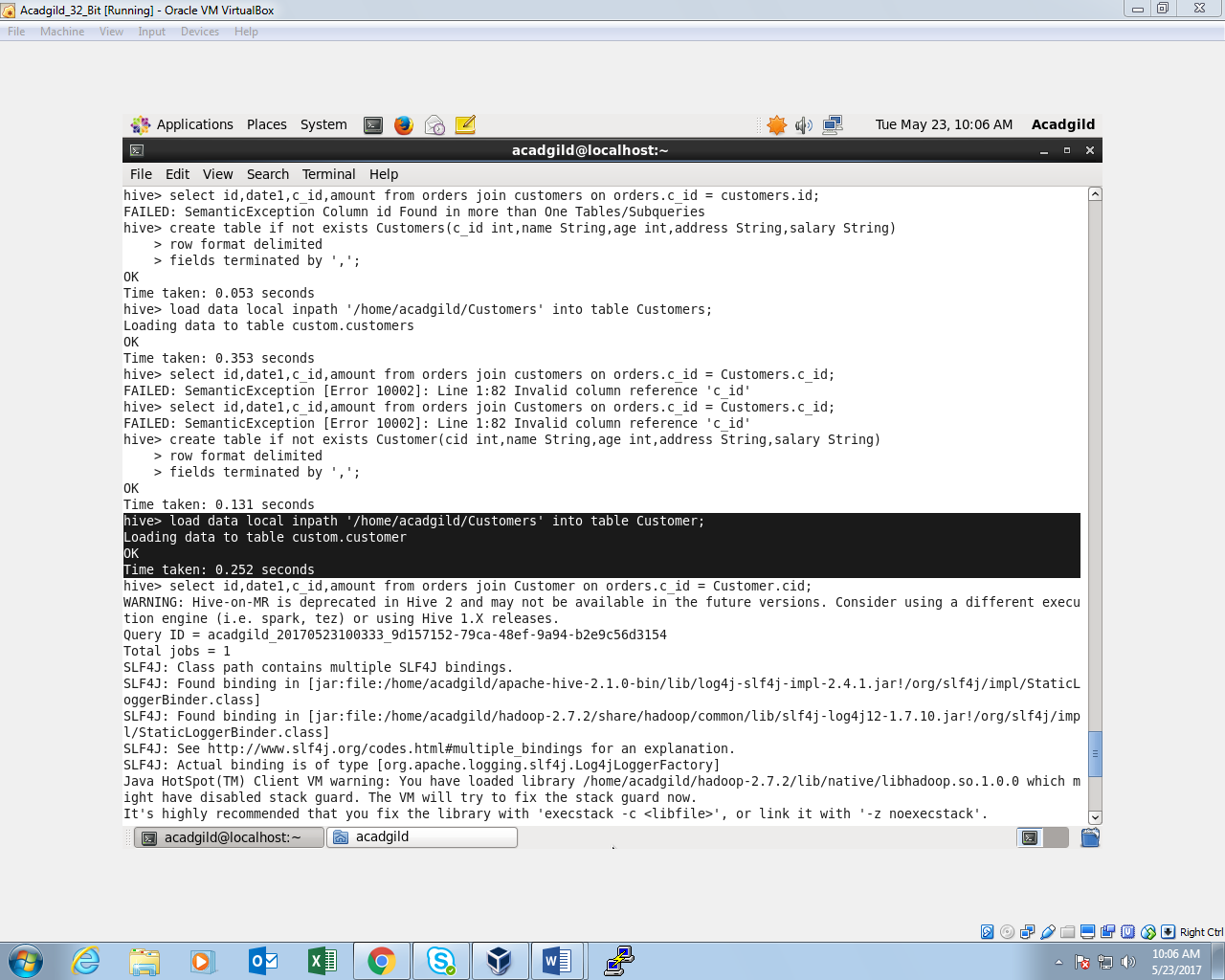
‘orders’ table is created and data is loaded into it.





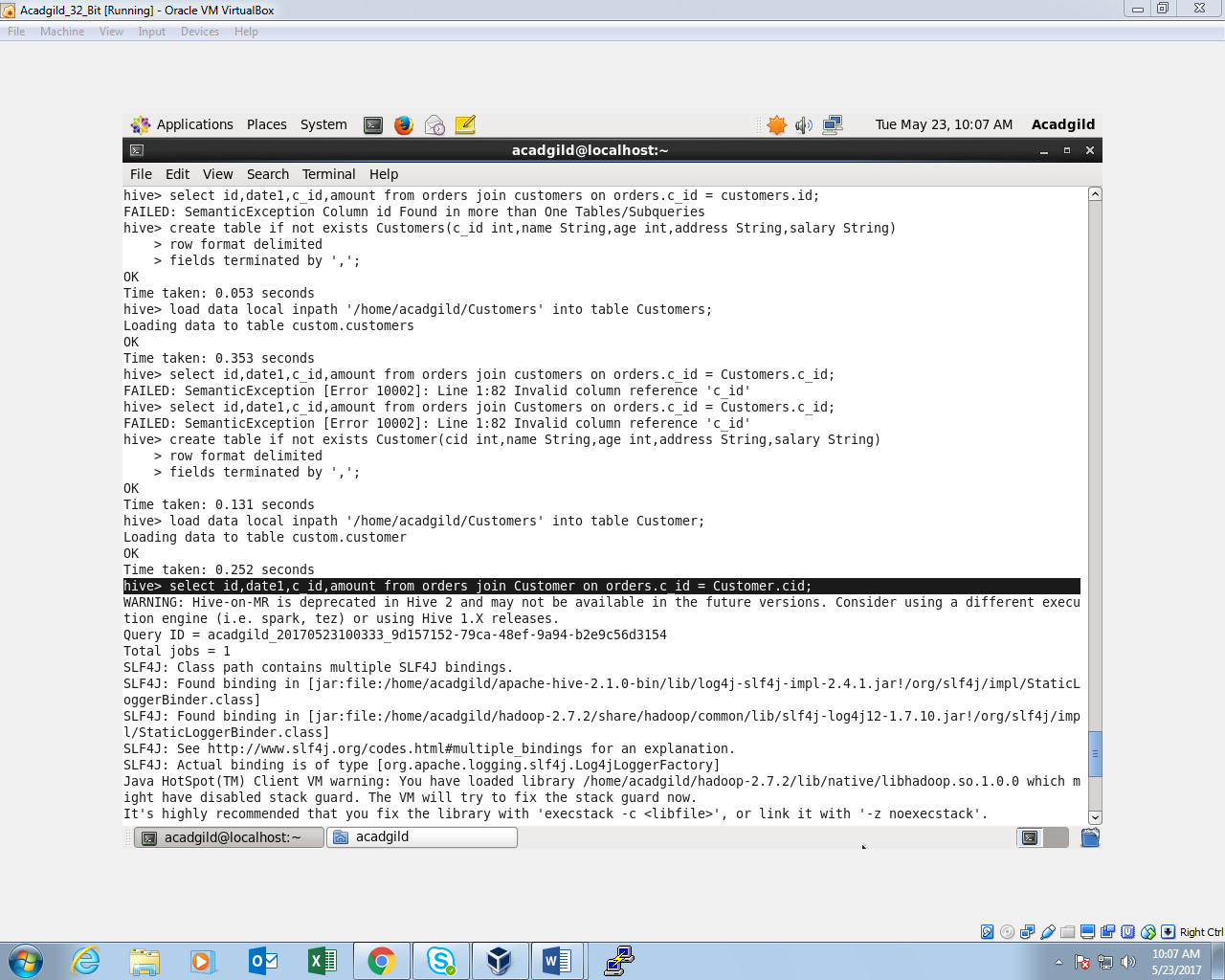
‘Customer’ table is created and dat is loaded into it.



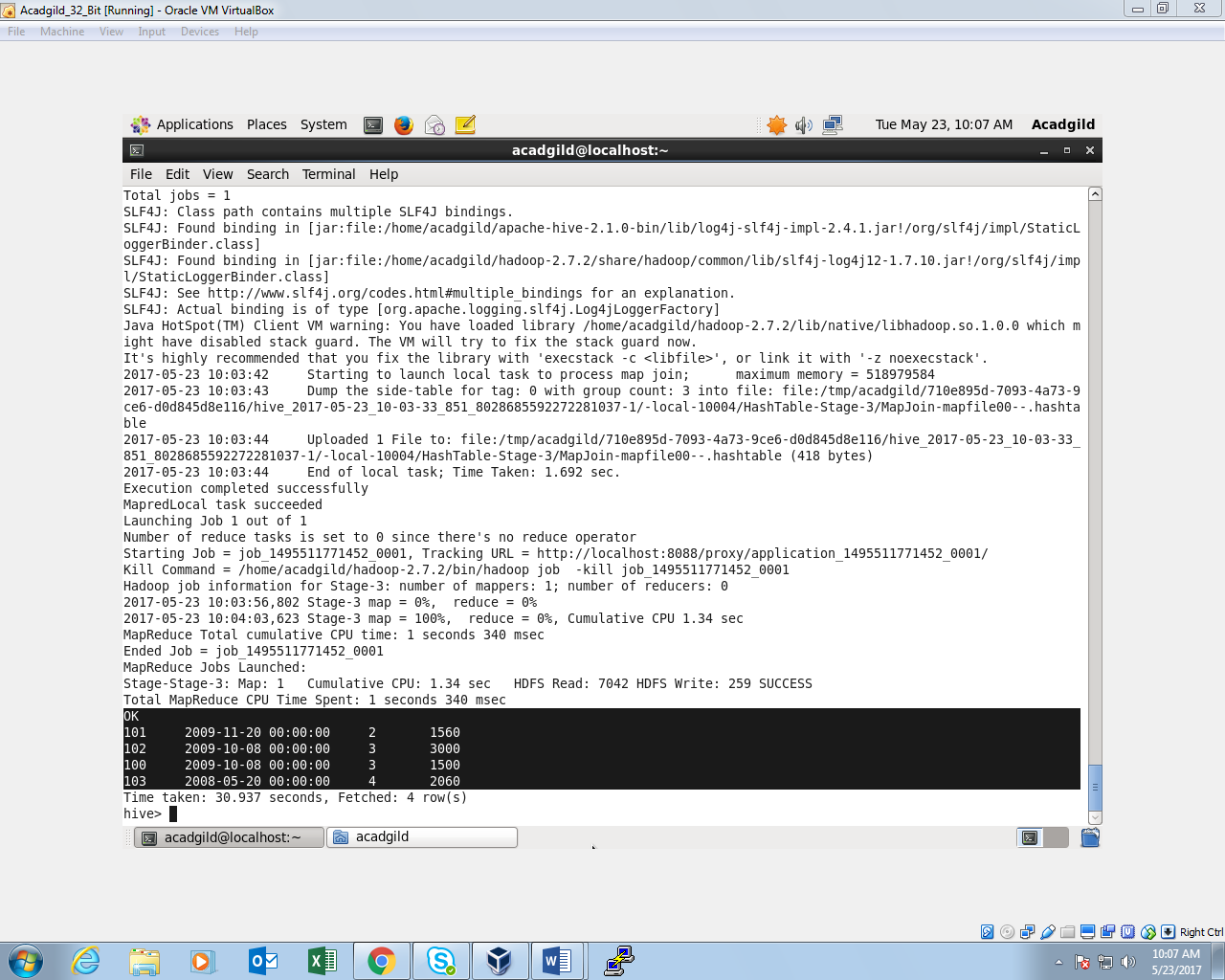


**JOIN :**

**COMMAND**



**OUTPUT:**

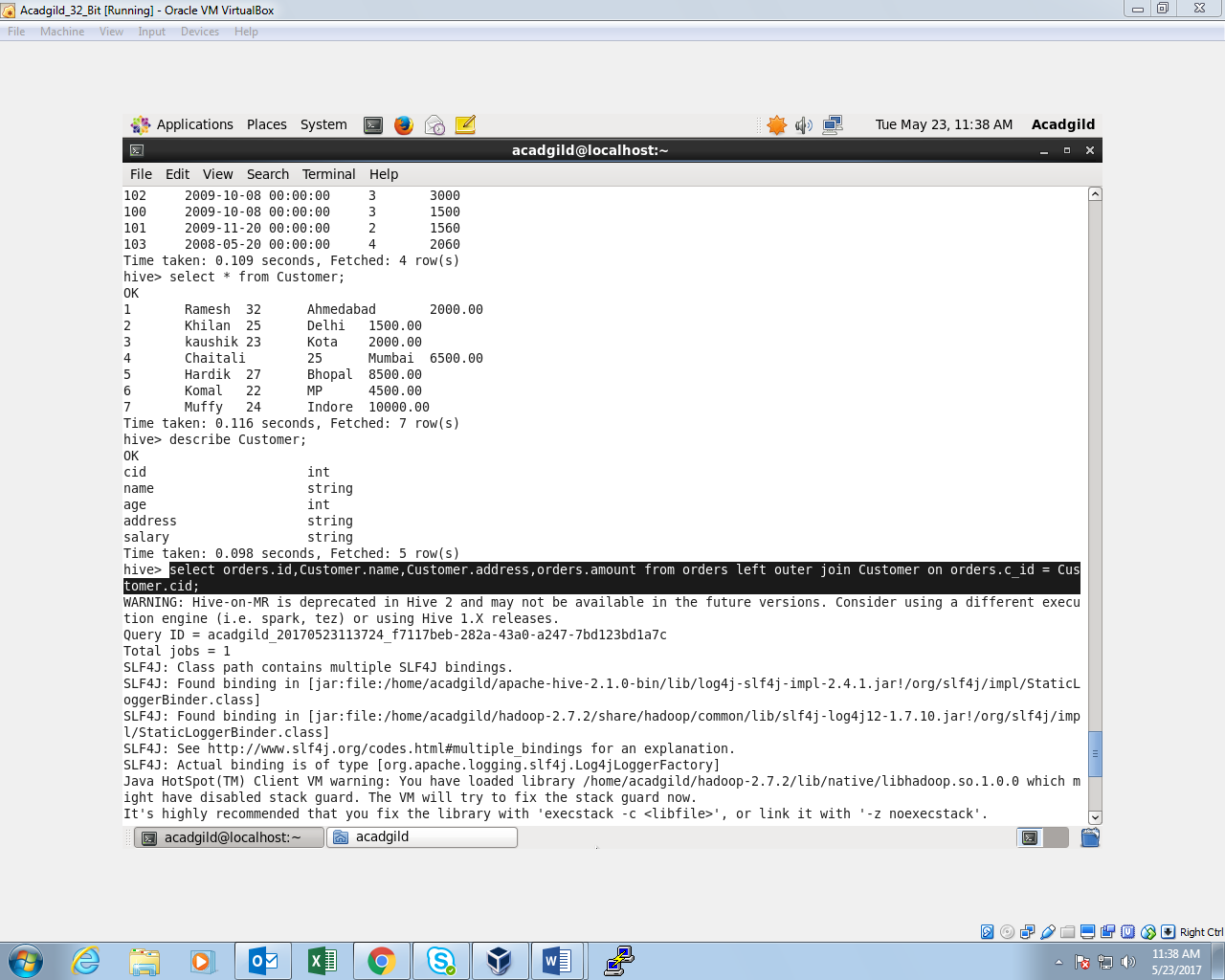


**USES OF JOIN:**

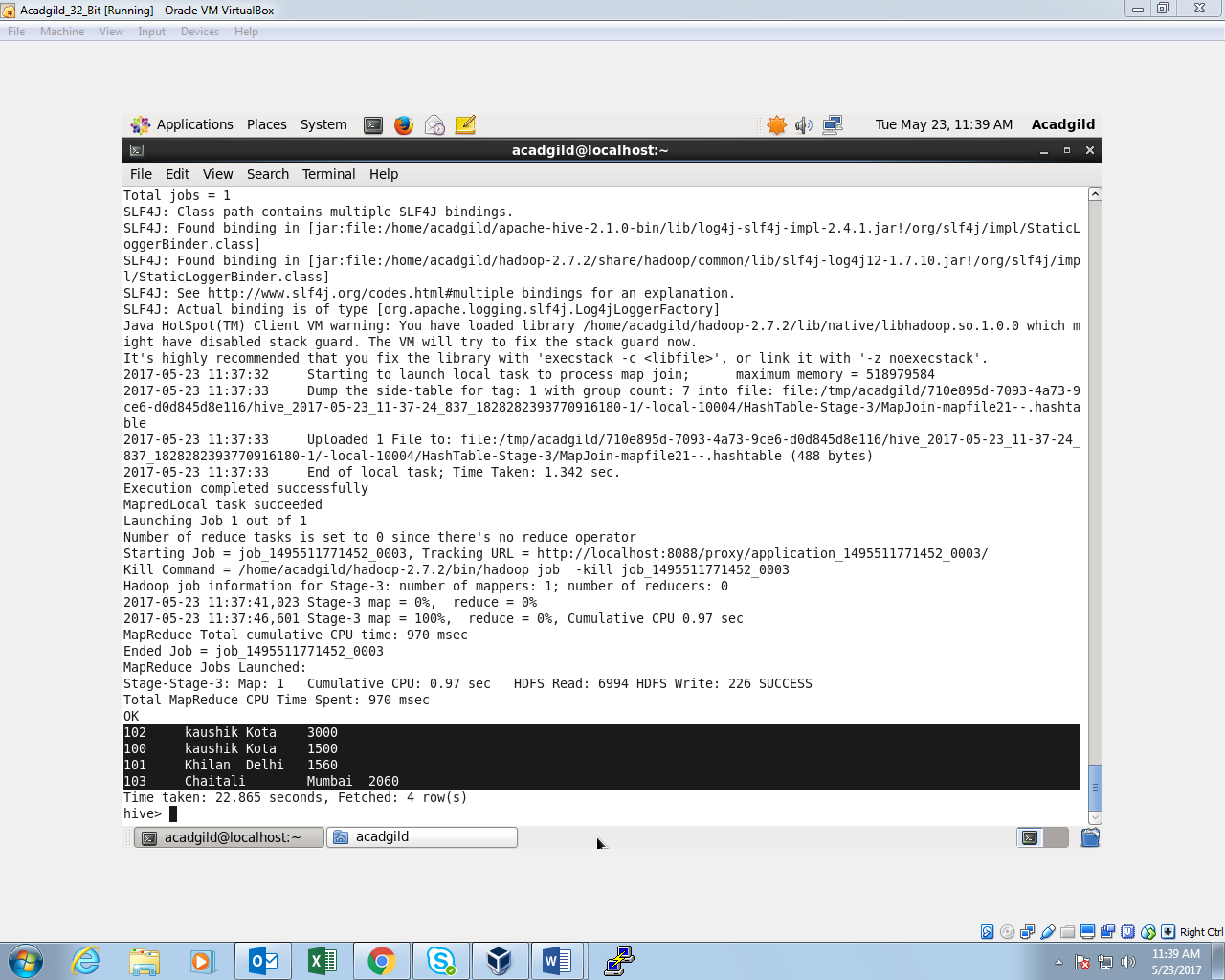
JOIN clause is used to combine and retrieve the records from multiple tables. JOIN is same as OUTER JOIN in SQL. A JOIN condition is to be raised using the primary keys and foreign keys of the tables.

**LEFT OUTER JOIN:**

**COMMAND:**



**OUTPUT:**

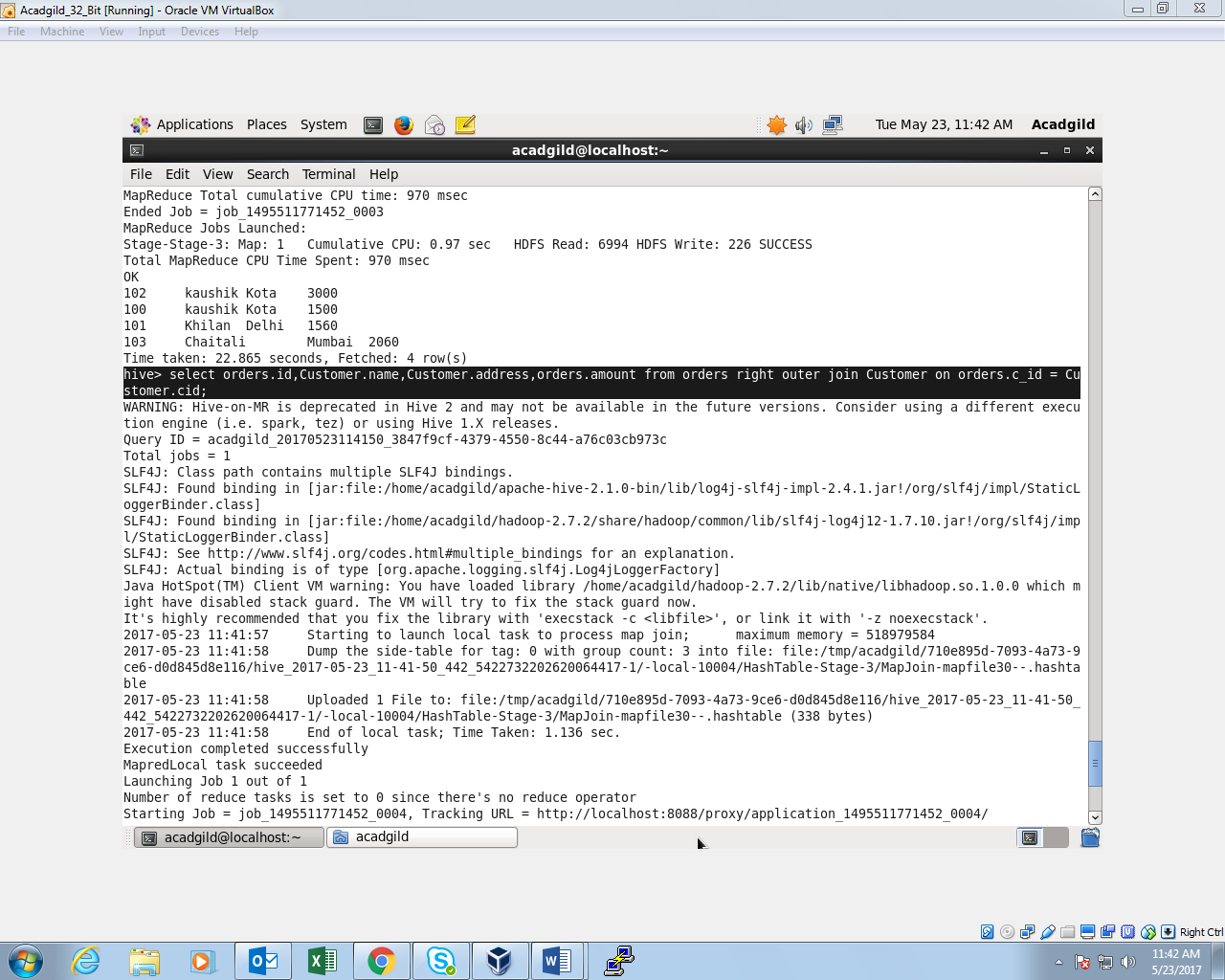


**USES OF LEFT OUTER JOIN :**

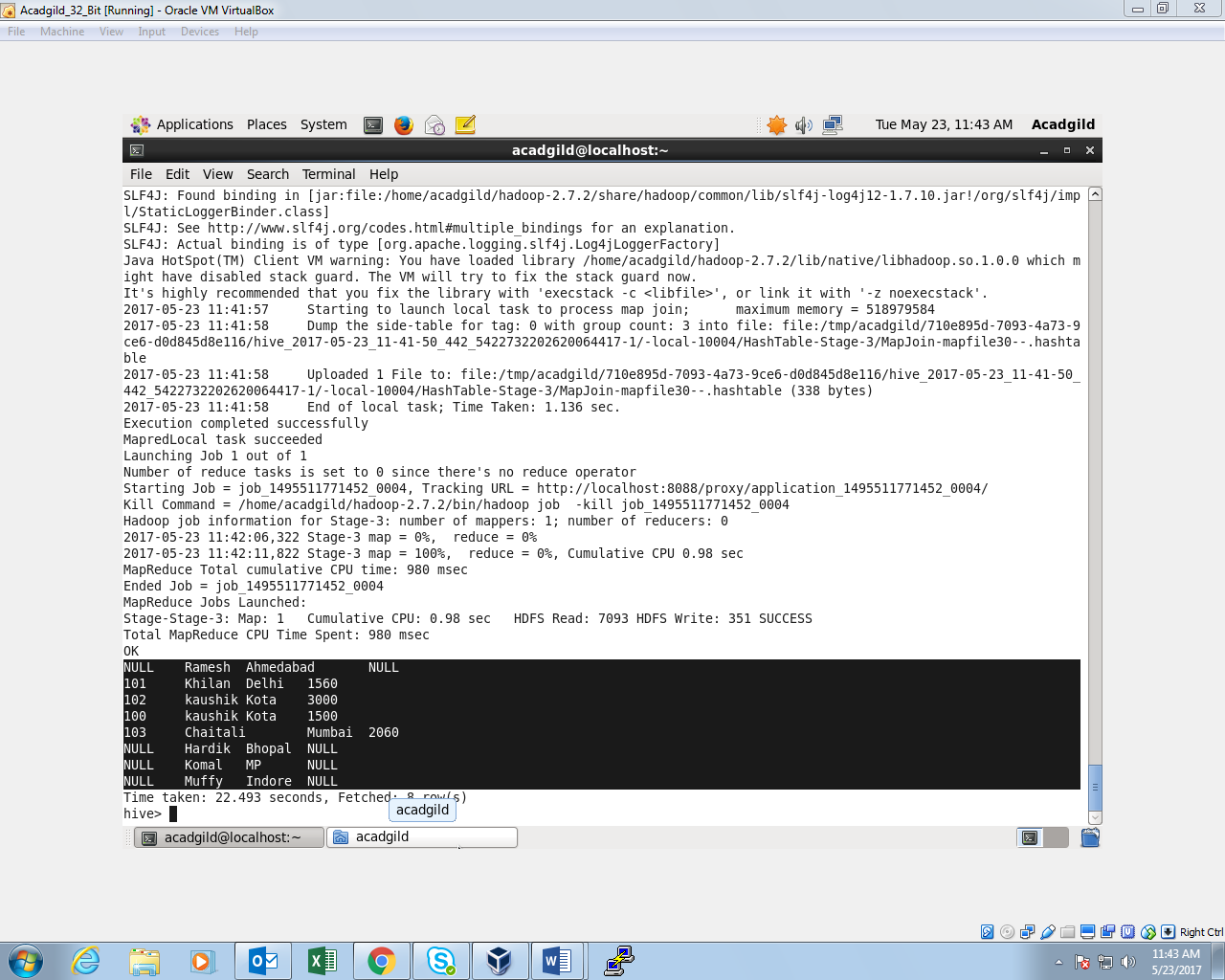
The HiveQL LEFT OUTER JOIN returns all the rows from the left table, even if there are no matches in the right table. This means, if the ON clause matches 0 (zero) records in the right table, the JOIN still returns a row in the result, but with NULL in each column from the right table.

A LEFT JOIN returns all the values from the left table, plus the matched values from the right table, or NULL in case of no matching JOIN predicate.

**RIGHT OUTER JOIN:**



**OUTPUT:**



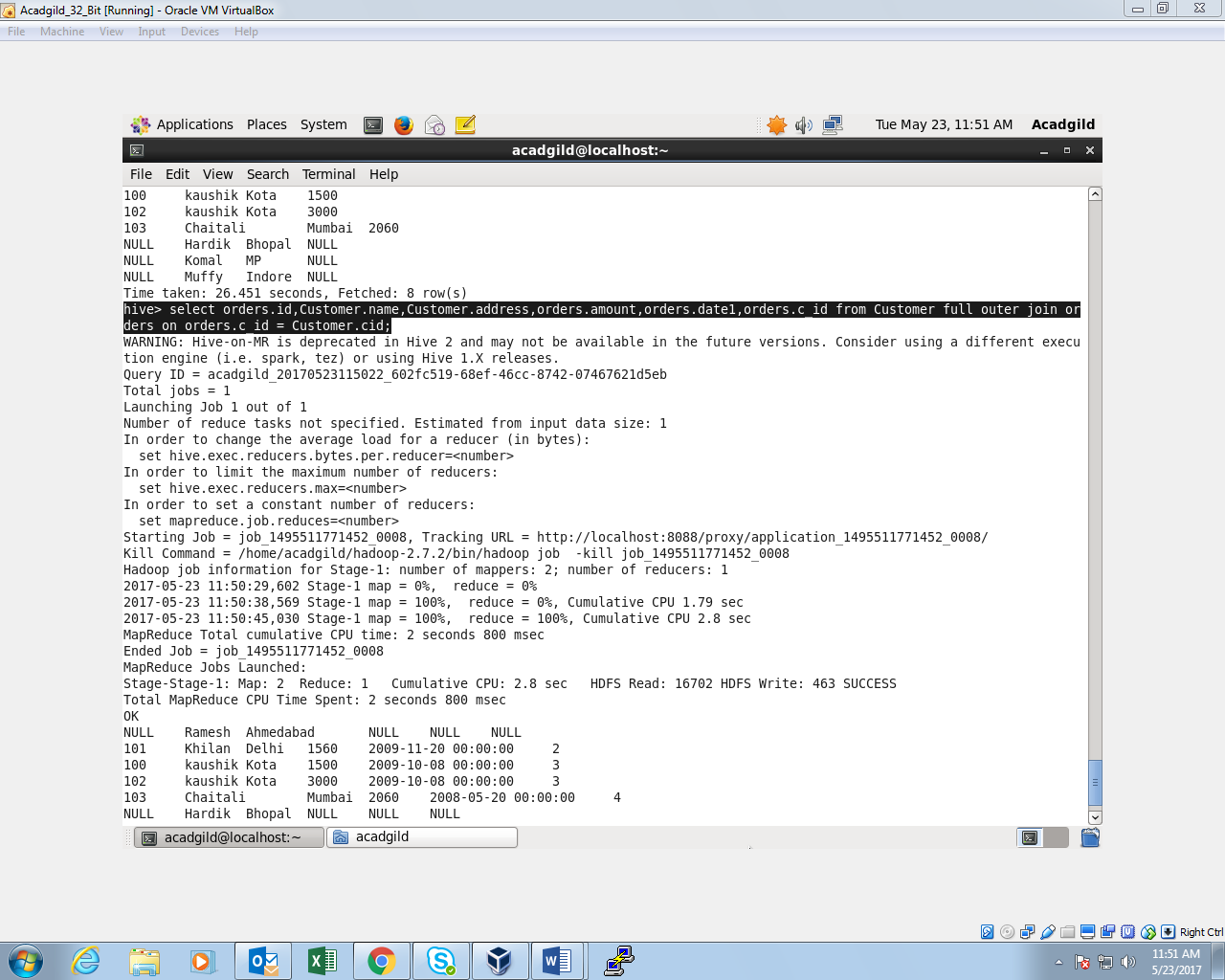
**USES:**

The HiveQL RIGHT OUTER JOIN returns all the rows from the right table, even if there are no matches in the left table. If the ON clause matches 0 (zero) records in the left table, the JOIN still returns a row in the result, but with NULL in each column from the left table.

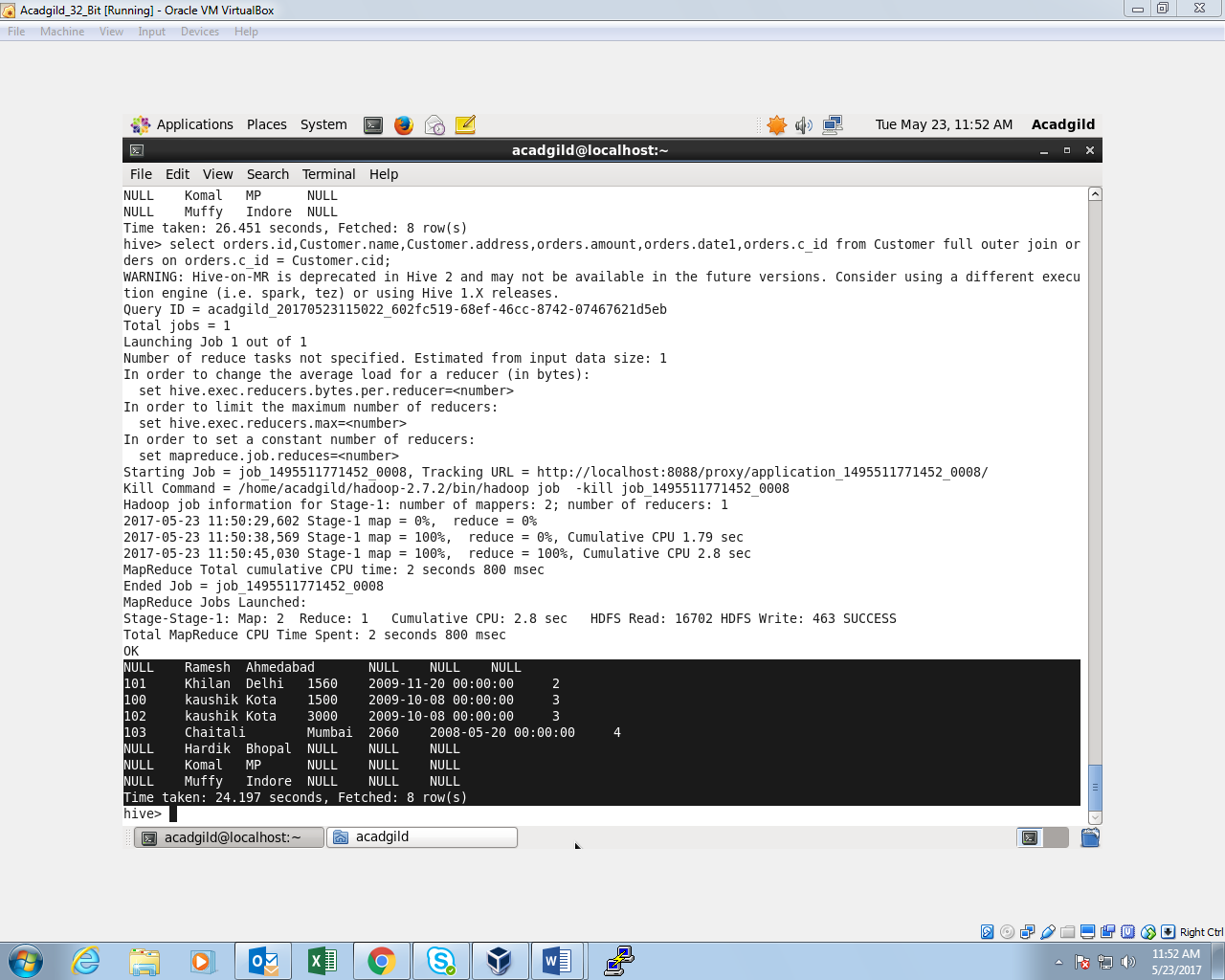
A RIGHT JOIN returns all the values from the right table, plus the matched values from the left table, or NULL in case of no matching join predicate.

**FULL OUTER JOIN:**

**COMMAND:**



**OUTPUT:**



**USES:**

The HiveQL FULL OUTER JOIN combines the records of both the left and the right outer tables that fulfil the JOIN condition. The joined table contains either all the records from both the tables, or fills in NULL values for missing matches on either side.