**Hive Operations**

**1. Hive Data Definitions :-**

Hive data definition language is a subset of hive SQL statements that described the data structure in hive by creating, deleting or altering schema objects such as database, tables, views, partitions and buckets. Most hive DDL statements starts with Keyword CREATE, DROP and ALTER.  It is perhaps closest to MySQL’s dialect, but with significant differences. Hive offers no support for row-level inserts, updates, and deletes. Hive doesn’t support transactions. Hive adds extensions to provide better performance in the context of Hadoop and to integrate with custom extensions and even external programs.

**2. Hive Data Manipulation :-**

In Hive data manipulation language parts is used to put data into tables and to extract data from tables to the filesystem. The ability to manipulate data is a critical capability in big data analysis. Manipulating data is the process of exchanging, moving, sorting, and transforming the data. This technique is used in many situations, such as cleaning data, searching patterns, creating trends, and so on. Hive offers various query statements, keywords, operators, and functions to carry out data manipulation.

**3. HiveQL Manipulation :-**

**1. Loading Data into Managed Tables :-**

Since Hive has no row-level insert, update, and delete operations, the only way to put data into an table is to use one of the “bulk” load operations. Or you can just write files in the correct directories by other means.

We saw an example of how to load data into a managed table in [Partitioned, Managed Tables](https://www.safaribooksonline.com/library/view/programming-hive/9781449326944/ch04.html#PartitionedManagedTables), which we repeat here with an addition, the use of the OVERWRITE keyword:

**LOAD** **DATA** **LOCAL** INPATH '${env:HOME}/california-employees'

OVERWRITE **INTO** **TABLE** employees

PARTITION (country = 'US', **state** = 'CA');  
 **2.Inserting Data into Tables from Queries :-**

**i. Dynamic Partition Inserts :-**

The user has to know which partition to insert into and only one partition can be inserted in one insert statement. If you want to load into multiple partitions, you have to use multi-insert statement as illustrated below.

|  |
| --- |
| FROM page\_view\_stg pvs  INSERT OVERWRITE TABLE page\_view PARTITION(dt='2008-06-08', country='US')         SELECT pvs.viewTime, pvs.userid, pvs.page\_url,  pvs.referrer\_url, null, null, pvs.ip WHERE pvs.country = 'US'  INSERT OVERWRITE TABLE page\_view PARTITION  (dt='2008-06-08', country='CA')         SELECT pvs.viewTime, pvs.userid, pvs.page\_url,  pvs.referrer\_url, null, null, pvs.ip WHERE pvs.country = 'CA'  INSERT OVERWRITE TABLE page\_view PARTITION  (dt='2008-06-08', country='UK')         SELECT pvs.viewTime, pvs.userid, pvs.page\_url,  pvs.referrer\_url, null, null, pvs.ip WHERE pvs.country = 'UK'; |

In order to load data into all country partitions in a particular day, you have to add an insert statement for each country in the input data. This is very inconvenient since you have to have the priori knowledge of the list of countries exist in the input data and create the partitions beforehand. If the list changed for another day, you have to modify your insert DML as well as the partition creation DDLs. It is also inefficient since each insert statement may be turned into a MapReduce Job.

3.Creating Tables and Loading Them in One Query.