**Hive Data Manipulations**

Hive DML language is used to insert and modify the data in hive.

There are multiple ways to modify data in Hive:

* [LOAD](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-Loadingfilesintotables)
* INSERT
  + [into Hive tables from queries](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-InsertingdataintoHiveTablesfromqueries)
  + [into directories from queries](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-Writingdataintothefilesystemfromqueries)
  + [into Hive tables from SQL](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-InsertingintotablesfromSQL)
* [UPDATE](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-Update)
* [DELETE](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-Delete)
* [MERGE](https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML#LanguageManualDML-Merge)

### Loading files into tables:

Hive does not do any transformation while loading data into tables. Load operations are currently pure copy/move operations that move datafiles into locations corresponding to Hive tables.

*Syntax:*

|  |
| --- |
| LOAD DATA [LOCAL] INPATH 'filepath' [OVERWRITE] INTO TABLE Acad1 [PARTITION (partcol1=val1, partcol2=val2 ...)] |

LOAD DATA LOCAL INPATH './usr/Desktop/kv1.txt' OVERWRITE INTO TABLE Employee;

**Inserting data into Hive Tables from queries**

Query Results can be inserted into tables by using the insert clause.

* Each row listed in the VALUES clause is inserted into table Acad1.
* If the table being inserted into supports [ACID](https://cwiki.apache.org/confluence/display/Hive/Hive+Transactions) and a transaction manager that supports ACID is in use, this operation will be auto-committed upon successful completion.
* Configuration Values to Set for INSERT, UPDATE, DELETE
* In addition to the new parameters listed above, some existing parameters need to be set to support INSERT ... VALUES, UPDATE, and DELETE.

| Configuration key | Must be set to |
| --- | --- |
| [hive.support.concurrency](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties#ConfigurationProperties-hive.support.concurrency) | true (default is false) |
| [hive.enforce.bucketing](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties#ConfigurationProperties-hive.enforce.bucketing) | true (default is false) (Not required as of [Hive 2.0](https://issues.apache.org/jira/browse/HIVE-12331)) |
| [hive.exec.dynamic.partition.mode](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties#ConfigurationProperties-hive.exec.dynamic.partition.mode) | nonstrict (default is strict) |
| [hive.txn.manager](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties#ConfigurationProperties-hive.txn.manager) | org.apache.hadoop.hive.ql.lockmgr.DbTxnManager |

Standard syntax:

INSERT OVERWRITE TABLE Acad11 [PARTITION (partcol1=val1, partcol2=val2 ...) [IF NOT EXISTS]] select\_statement1 FROM from\_statement;

INSERT INTO TABLE Acad11 [PARTITION (partcol1=val1, partcol2=val2 ...)] select\_statement1 FROM from\_statement;

*Examples:*

|  |
| --- |
| CREATE TABLE students (name VARCHAR(64), age INT, gpa DECIMAL(3, 2))    CLUSTERED BY (age) INTO 2 BUCKETS STORED AS ORC;    INSERT INTO TABLE students    VALUES ('fred flintstone', 35, 1.28), ('barney rubble', 32, 2.32);      CREATE TABLE pageviews (userid VARCHAR(64), link STRING, came\_from STRING)    PARTITIONED BY (datestamp STRING) CLUSTERED BY (userid) INTO 256 BUCKETS STORED AS ORC;    INSERT INTO TABLE pageviews PARTITION (datestamp = '2014-09-23')    VALUES ('jsmith', 'mail.com', 'sports.com'), ('jdoe', 'mail.com', null);    INSERT INTO TABLE pageviews PARTITION (datestamp)    VALUES ('tjohnson', 'sports.com', 'finance.com', '2014-09-23'), ('tlee', 'finance.com', null, '2014-09-21'); |

### Update:

* The referenced column must be a column of the table being updated.
* The value assigned must be an expression that Hive supports in the select clause.  Thus arithmetic operators, UDFs, casts, literals, etc. are supported.  Subqueries are not supported.
* Only rows that match the WHERE clause will be updated.
* Partitioning columns cannot be updated.
* Bucketing columns cannot be updated.
* In Hive 0.14, upon successful completion of this operation the changes will be auto-committed.

*Standard Syntax:*

UPDATE Acad1 SET column = value [, column = value ...] [WHERE expression]

##### *Note:*

* Vectorization will be turned off for update operations.  This is automatic and requires no action on the part of the user.  Non-update operations are not affected.  Updated tables can still be queried using vectorization.
* In version 0.14 it is recommended that you set [hive.optimize.sort.dynamic.partition](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties" \l "ConfigurationProperties-hive.optimize.sort.dynamic.partition)=false when doing updates, as this produces more efficient execution plans.

### Delete:

*Syntax*

|  |
| --- |
| *Standard Syntax:*  DELETE FROM Acad1 [WHERE expression] |

* Only rows that match the WHERE clause will be deleted.
* In Hive 0.14, upon successful completion of this operation the changes will be auto-committed.

##### Note:

* Vectorization will be turned off for delete operations.  This is automatic and requires no action on the part of the user.  Non-delete operations are not affected.  Tables with deleted data can still be queried using vectorization.
* In version 0.14 it is recommended that you set [hive.optimize.sort.dynamic.partition](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties" \l "ConfigurationProperties-hive.optimize.sort.dynamic.partition)=false when doing deletes, as this produces more efficient execution plans.

### Merge:

Standard Syntax:

MERGE INTO <target table> AS T USING <source expression/table> AS S

ON <boolean expression1>

WHEN MATCHED [AND <boolean expression2>] THEN UPDATE SET <set clause list>

WHEN MATCHED [AND <boolean expression3>] THEN DELETE

WHEN NOT MATCHED [AND <boolean expression4>] THEN INSERT VALUES<value list>

* [Merge](https://en.wikipedia.org/wiki/Merge_(SQL)) allows actions to be performed on a target table based on the results of a join with a source table.
* In Hive 2.2, upon successful completion of this operation the changes will be auto-committed.

##### *Performance Note*

SQL Standard requires that an error is raised if the ON clause is such that more than 1 row in source matches a row in target.  This check is computationally expensive and may affect the overall runtime of a MERGE statement significantly.  [hive.merge.cardinality.check](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties#ConfigurationProperties-hive.merge.cardinality.check)=false may be used to disable the check at your own risk.  If the check is disabled, but the statement has such a cross join effect, it may lead to data corruption.