## Create Hive-Managed Tables

**<Command to create the Hive tables>**

1. create database if not exists cab\_rides\_data;
2. use cab\_rides\_data;
3. Command to create clickStreamData table and load data from HDFS:

create table if not exists clickStreamData(

customer\_id int,

app\_version string,

os\_version string,

lat double,

lon double,

page\_id string,

button\_id string,

is\_button\_click string,

is\_page\_view string,

is\_scroll\_up string,

is\_scroll\_down string,

`timestamp` timestamp)

row format delimited fields terminated by ',' lines

terminated by '\n' stored as textfile

tblproperties("skip.header.line.count"="1");

1. create table if not exists bookingsData(

booking\_id string,

customer\_id int,

driver\_id int,

customer\_app\_version string,

customer\_phone\_os\_version string,

pickup\_lat double,

pickup\_lon double,

drop\_lat double,

drop\_lon double,

pickup\_timestamp timestamp,

drop\_timestamp timestamp,

trip\_fare double,

tip\_amount double,

currency\_code string,

cab\_color string,

cab\_registration\_no string,

customer\_rating\_by\_driver int,

rating\_by\_customer int,

passenger\_count int)

row format delimited fields terminated by ',' lines

terminated by '\n' stored as textfile;

1. create table if not exists testAggregateData( `date` string, no\_of\_bookings int)

row format delimited fields terminated by ',' lines terminated by '\n' stored as textfile;

**<Command to load the data into Hive tables>**

1.load data inpath '/user/hadoop/clickStream\_flatten\_data/' into table clickStreamData;

2. load data inpath '/user/hadoop/bookings-data/' into table bookingsData;

3. load data inpath '/user/hadoop/bookings\_aggregate\_data/' into table testAggregateData;