<https://www.youtube.com/watch?v=XjVrId0YJSs&list=PLsxbzAdvhorg3kKYqVXEvOGreyGvW-k-i>

LOAD CSV file into PIG

**Step 1:**

Register piggybank

**Step 2: LOAD**

A = LOAD '/home/training/Desktop/Hive/hive partitioning/Consumer\_Complaints (1).csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage as (date\_recieved,Product, Sub\_Product,Issue,Sub\_Issue,Narrative,Company\_Public\_response,Company,State,Zipcode,Submitted\_via,Date\_sent\_to\_company,Company\_responsetoconsumer,Timely\_response,consumer\_dispute,Complaint\_ID);

**Total records:**

B = foreach (GROUP A all) generate COUNT(A);

(151114)

**Step 3: Generate required columns:**

C = FOREACH A GENERATE date\_recieved,Product, Sub\_Product,Company,State,Zipcode,Submitted\_via,Company\_responsetoconsumer,Complaint\_ID;

**Loading pre-processed data from pig to hive using HCatalog.**

**First create table in HIVE**

CREATE EXTERNAL TABLE CCA(date\_recieved STRING,Product STRING,Sub\_Product STRING,Company STRING,State STRING ,Zipcode INT, Submitted\_via STRING, Company\_responsetoconsumer STRING,Complaint\_ID INT)

ROW FORMAT DELIMITED fields terminated by ‘,’

location ‘/home/training/Desktop’;

**STORE PIG RELATION INTO HDFS**

grunt> STORE C INTO 'hdfs://localhost:54310/user/hive/warehouse/CCA' USING PigStorage(',');

**LOAD DATA FROM HDFS TO HIVE TABLE;**

hive> LOAD DATA INPATH 'hdfs://localhost:54310/user/hive/warehouse/CCA' INTO TABLE CCA;

CREATING PARTITION TABLE

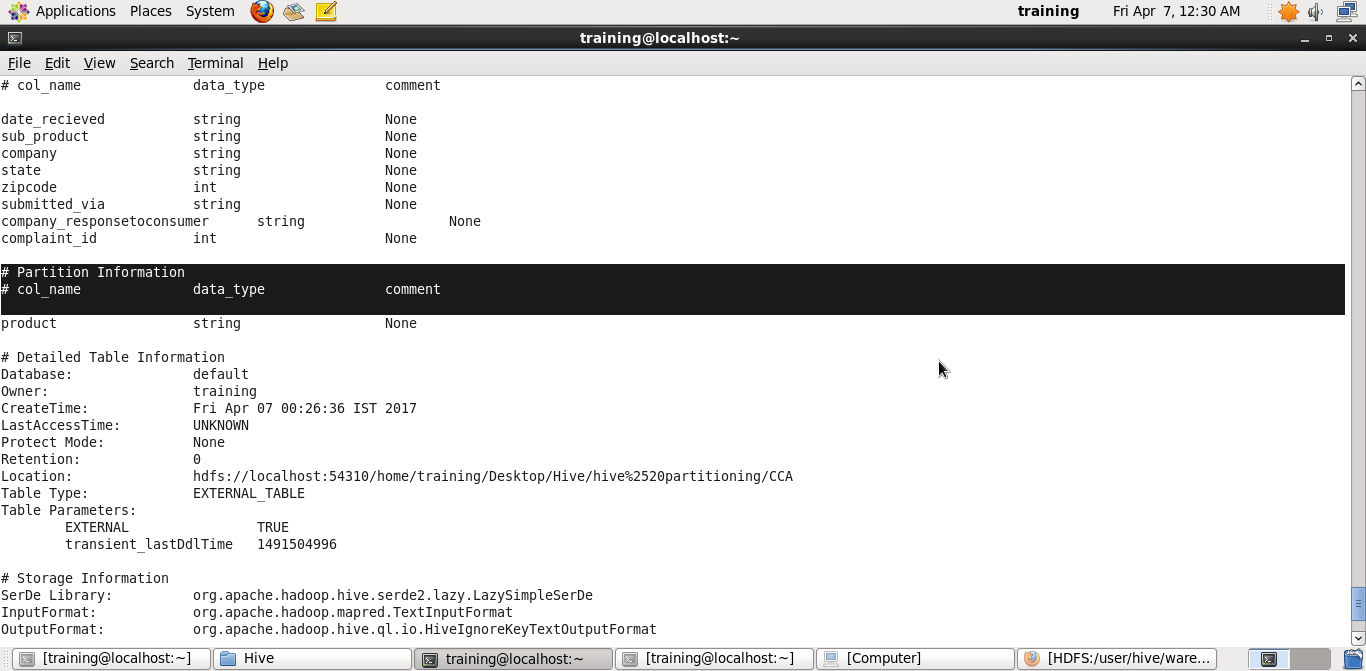
CREATE EXTERNAL TABLE **CCA\_STATIC\_PARTION**(date\_recieved STRING,Sub\_Product STRING,Company STRING,State STRING ,Zipcode INT, Submitted\_via STRING, Company\_responsetoconsumer STRING,Complaint\_ID INT) PARTITIONED BY(**PRODUCT** STRING)

> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

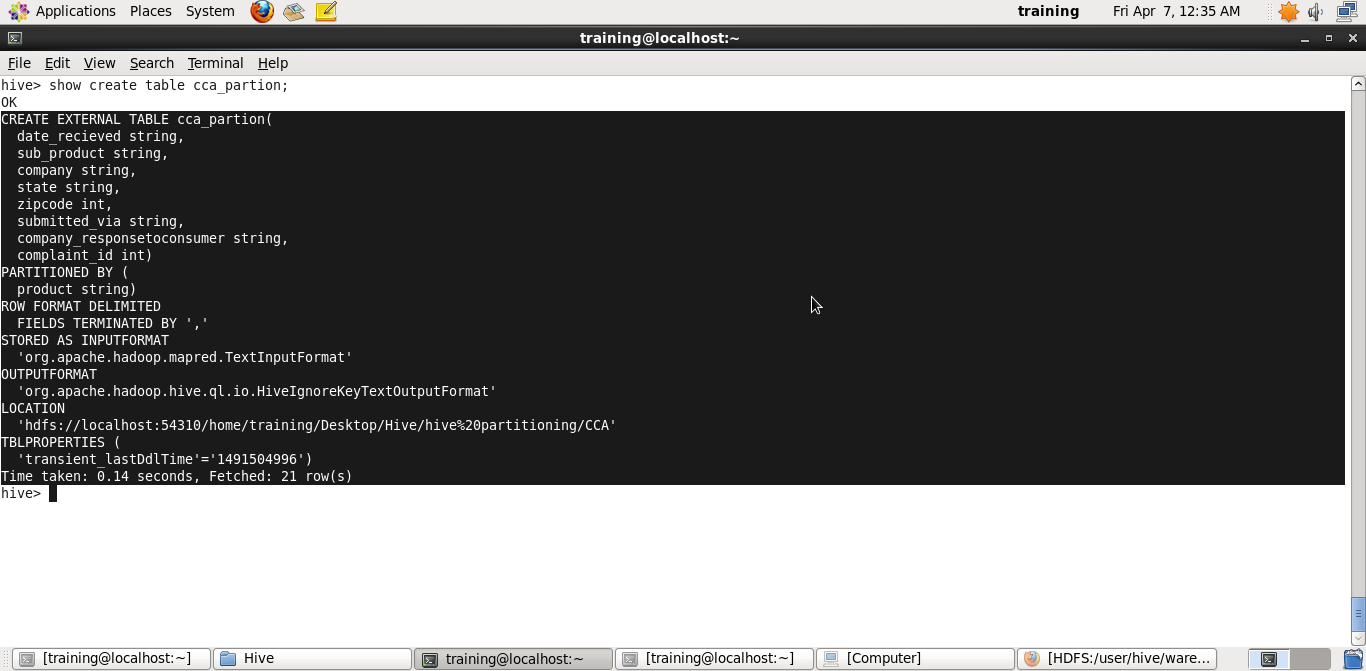
> STORED AS TEXTFILE

> LOCATION ' /user/hive/warehouse/CCA/Static\_Partition/';

**desc formatted CCA\_PARTION;**



**show create table cca\_partion;**



INSERTING DATA INTO PARTIONED TABLE – **STATIC INSERT**

hive (default)> INSERT OVERWRITE TABLE cca\_static\_partion PARTITION(Product='Consumer loan') SELECT date\_recieved,Sub\_Product,COMPANY,STATE,ZIPCODE,Submitted\_via,Company\_responsetoconsumer,Complaint\_ID FROM CCA WHERE Product='Consumer loan';

Drawbacks of static insert:

If we go for the above approach , if we have 50 partitions we need to do the insert statement 50 times. That is a tedious task.

**INSERTING DATA INTO PARTIONED TABLE – Dynamic INSERT**

First create an external table for it :

CREATE EXTERNAL TABLE CCA\_DYNAMIC\_PARTITION2 (date\_recieved STRING,Sub\_Product STRING,Company STRING,State STRING ,Zipcode INT, Submitted\_via STRING, Company\_responsetoconsumer STRING,Complaint\_ID INT) PARTITIONED BY (Product string)

> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

> LOCATION '/user/hive/warehouse/CCA/';

**In order to achieve the same we need to set 4 things,**

1. **set hive.exec.dynamic.partition=true;**

This enable dynamic partitions, by default it is false.

1. **set hive.exec.dynamic.partition.mode=nonstrict;**

* By default strict – Because it may leads to so many number of partitions
* NonStrict- We are explicitly specifying no of partitions.
* If strict we cannot perform queries like – select \* from --------------(here we have to use where clause). i.e It restricts queries of partitioned tables without a **WHERE** clause.
* But for nonstrict mode we can perform.
* It restricts **ORDER BY** operation without a **LIMIT** clause ( since it uses a single reducer which can choke your processing if not handled properly

1. **set hive.exec.max.dynamic.partitions.pernode=3;**

The default value is 100, we have to modify the

same according to the possible no of partitions

1. **set hive.exec.max.created.files=150000;**

The default values is 100000 but for larger tables. It can exceed the default, so we may have to update the same.

If the table has only dynamic partition columns

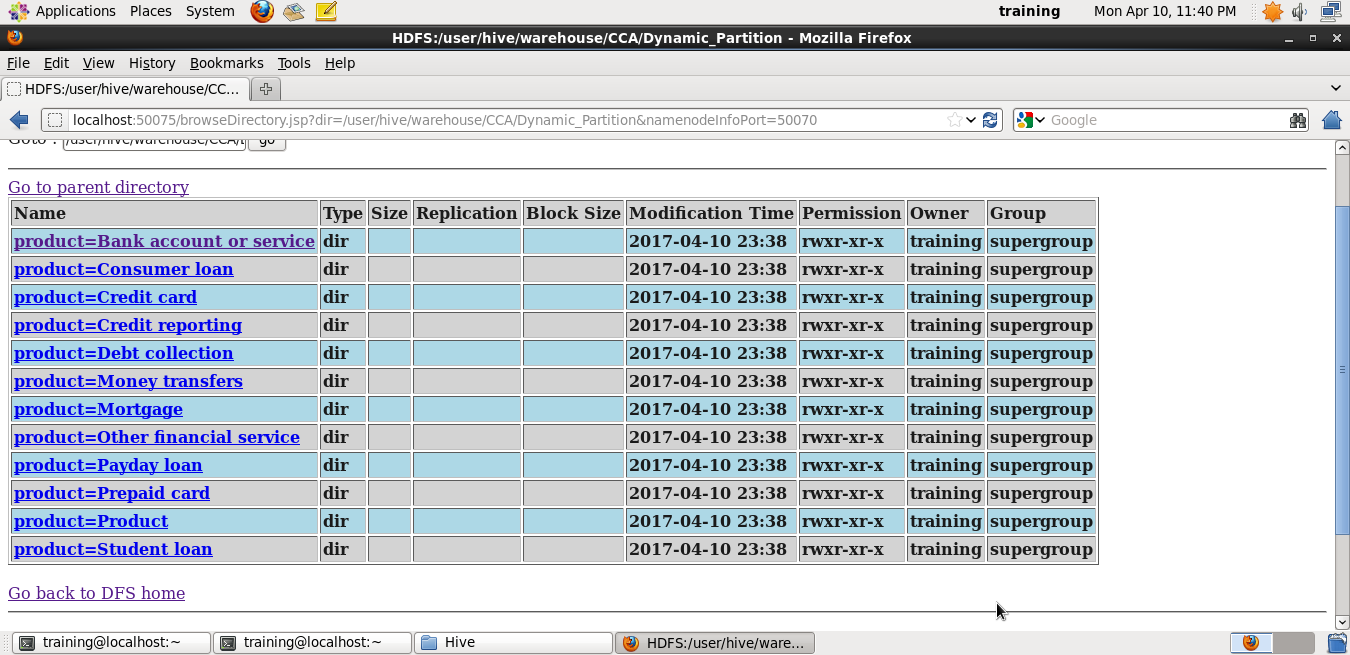
Change dynamic partition mode to non-strict:

By default strict: To prevent accidental creation of partition.

Insert

INSERT OVERWRITE TABLE cca\_dynamic\_partition2 PARTITION(Product) SELECT date\_recieved,Sub\_Product,COMPANY,STATE,ZIPCODE,Submitted\_via,Company\_responsetoconsumer,Complaint\_ID,Product FROM CCA;

Here Product ( i.e partitioning column) should be in last item.



**Partition by Company:**

CREATE EXTERNAL TABLE CCA\_DYNAMIC\_PARTITION\_Company2 (date\_recieved STRING, Product string, Sub\_Product STRING, State STRING ,Zipcode INT, Submitted\_via STRING, Company\_responsetoconsumer STRING,Complaint\_ID INT) PARTITIONED BY (Company string)

> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

> LOCATION '/user/hive/warehouse/CCA/CCA\_DYNAMIC\_PARTITION\_Company';

INSERT OVERWRITE TABLE CCA\_DYNAMIC\_PARTITION\_Company2 PARTITION(Company) SELECT date\_recieved,Product,Sub\_Product,STATE,ZIPCODE,Submitted\_via,Company\_responsetoconsumer,Complaint\_ID,Company FROM CCA;

Partition by Multiple column:

First Partition – Product.

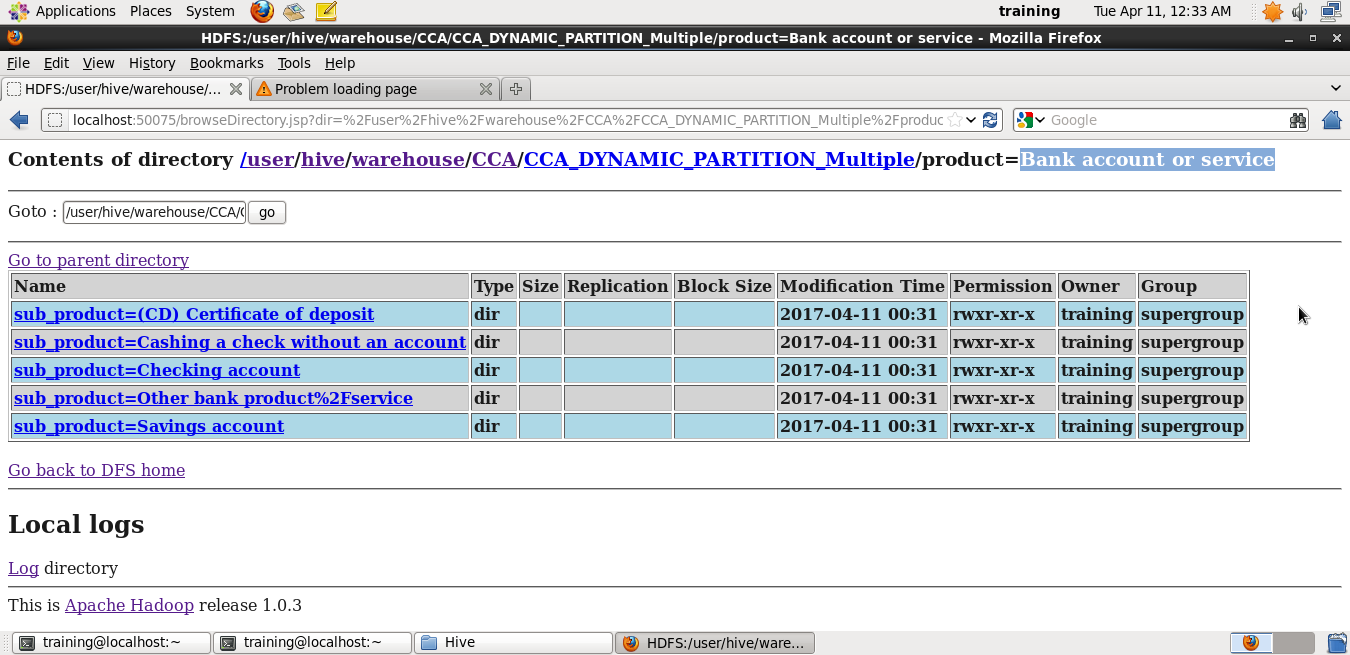
Under product: Partition by sub\_product.

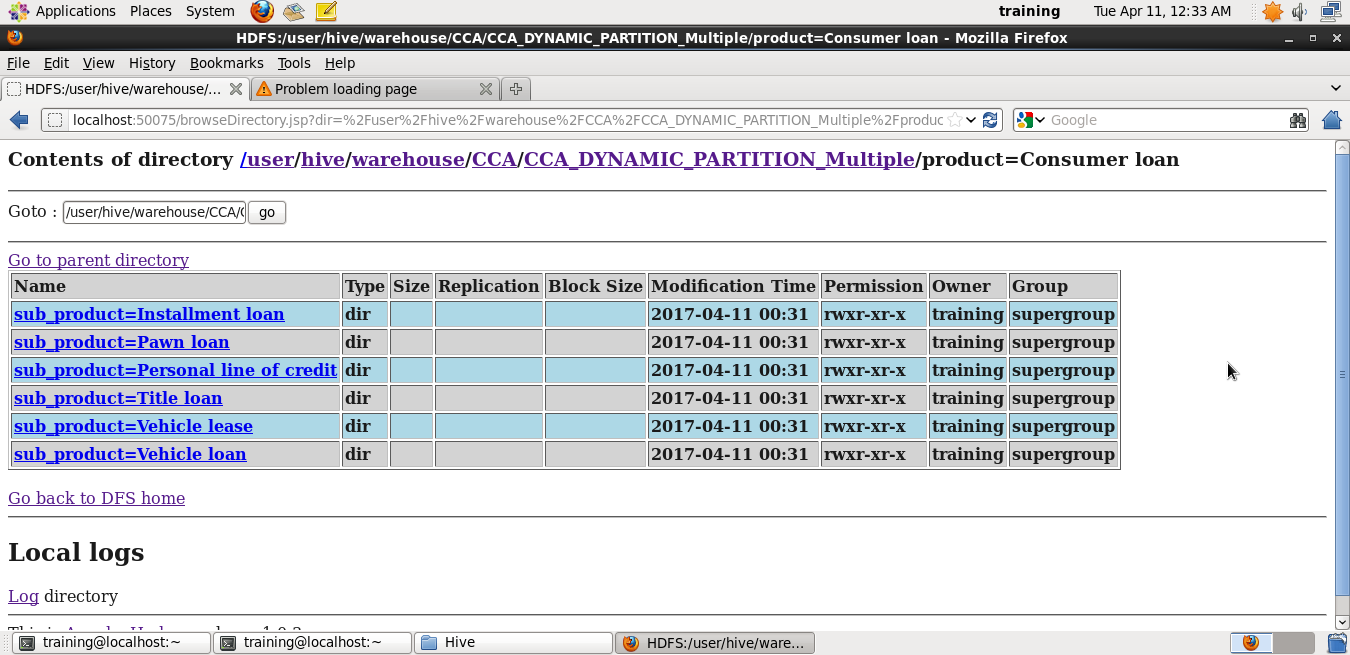
CREATE EXTERNAL TABLE CCA\_DYNAMIC\_PARTITION\_Multiple (date\_recieved STRING,Company STRING,State STRING ,Zipcode INT, Submitted\_via STRING, Company\_responsetoconsumer STRING,Complaint\_ID INT) PARTITIONED BY (Product string, Sub\_Product STRING)

> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

> LOCATION '/user/hive/warehouse/CCA/CCA\_DYNAMIC\_PARTITION\_Multiple';

INSERT OVERWRITE TABLE CCA\_DYNAMIC\_PARTITION\_Multiple PARTITION(Product, Sub\_Product) SELECT date\_recieved,COMPANY,STATE,ZIPCODE,Submitted\_via,Company\_responsetoconsumer,Complaint\_ID,Product, Sub\_Product FROM CCA;





To show how many partitions that a table had?

Show partitions cca\_dynamic\_partition\_multiple;

