HBase demo

While setting up HBase, HBase can actually run in two different modes that is your standalone mode. By default, HBase runs in standalone mode. HBase does not use HDFS but it uses the local file system and it's going to be running in standalone mode. It runs all the HBase daemons and a local ZooKeeper in the same JVM.

Remember, we had discussed about a ZooKeeper services which will be used up by HBase in order to actually be in sync with all of its RegionalServers. However, while running HBase in standalone mode, it doesn't depend upon an external ZooKeeper service. A ZooKeeper service also will be running locally into one of the JVMs.

The other mode of HBase obviously is going to be the Distributed mode. In Pseudo-distributed mode, all the daemons run on a single machine.

However, all of these daemons are individually going to be separately running on a JVM or in a separate container. And HBase is completely written up using JAVA. So obviously it makes sense to actually think of the HBase services or HBase daemons running in individual JVMs when it comes to Pseudo-distributed mode.

However, in Pseudo-distributed mode, it does depend upon the HDFS or optionally it can also make use of the local file system but in Fully-distributed mode, your HBase only is supposed to use HDFS. It doesn't use the local file system. And all these daemons will be spread across multiple nodes in the cluster. It's not going to be on a single node. Let's quickly have a hands on session of all the commands what we actually discussed in HBase.

Typically, HBase is just going to have four different set of operations that can be performed that is Get, Put, Scan and Delete and we shall see all those things in action right now.

I have logged into the Jigsaw lab using your putty, where the HBase services are actually being installed and then it's already running in a fully distributed mode. I recommend you not to waste time trying to install the HBase on your VMs. The VMs, we should be shared with you would not have HBase installed in it and more ever when you are actually trying to experiment with the database system like HBase, it's better to run it in a fully distributed mode.

So in my terminal of my home directory in the Jigsaw lab, you just have to type in the

command called HBase space shell and press the enter key and wait for the HBase shell to actually appear. Now this is where we are going to type in all the commands to get started working with HBase.

So unlike the normal databases, we don't have the concept of a container or a name space organization called as a database over here. Typically, it's just going to be flag tables. So, you want to see what are all the existing tables in HBase, the command for that would be list? Just press the enter key.

So these are all the existing tables in our HBase database that is we have a table by name boko, company, customer and which is already existing, created by someone.

Now, in order to actually check if a table by a particular name is existing, say for example, I want to check if a table by the name 'company' is existing. Here, in this case, I obviously know that by browsing through the list of the tables that the table by name 'company' is existing but in the case where we have a lot of tables in HBase, if you want to explicitly query that a table of a particular name exists, the command for that would be exists followed by a space and name of the table which you want to check and press the

enter key and it says table company does exist.

Suppose let's actually try to check if a table which is not existing. Exists, I would try to cook up some name x, c, f, g something which I know it's not existing. It says the following table x, c, f, g, g does not exist. Now, this is actually a quick way to check the existence of a table.

Suppose I want to check what are the contents of a existing table in HBase. I know that the two tables, one table by the name 'employee', other table by the name 'employees', actually is existing in the list of tables in my HBase. Now, I want to check what are the contents or what are the rows which actually is present in the employees table.

The command for that would be scan, scan space within single codes, you are supposed to mention the name of the table. I am interested in knowing what are contents of the employees table out here. Here, it actually gives out some information. At this point of time, I would request you not to worry too much about what exactly this means.

Very soon in the demo part, we are going to get started to see how we create a table and how we actually insert a data into the cells

and all these things actually would make sense. For time being just know that to check or to list the contents of an existing table, the command would be scan followed by within single code, you have to mention the name of the table to see what exactly are the contents. In a very brief way, you can just conclude that it has four rows present in it. That's it.

Now let's see how do we delete an existing table. Just now, I just try to scan the contents of the employees table. I now want to delete the existing table. The command for that would be as follows. First, we have to disable the table, disable employees and then just press enter key after that you just have to enter the command called drop employees. This would actually end up deleting the table and all the data associated with that. Now, if you actually run the command called exists employees, let us see what would be the result. It says the table employees does not exist.

Let's check the existence of a table by the name company by using the exists command. Yes, there thus exist a table by the name company. Now we have seen how do we delete the existing table by using the option called as disabling the table and dropping the table. Now, how do we delete just a data in

the table but retain the table for future use. For that, we need to first see what are the contents of the table company. So, it has ten rows.

Now, I just want to delete the rows or simply delete the data which is present in the table but retain the table as it is. The command for that would be Truncate. Truncate space name of the table company. See what's happening over here. It's disabling the table internally, it's dropping the table and then it's trying to recreate the table.

Now, if you want to check if the table by the name company is already existing, we can actually again use the command called "exists". It says the table company does exist. However, if we try to query and see what exactly are the contents of the table company, eventually, we would not get any rows out of it. It says zero rows.

Let's quickly check if the table by the name employees exist already. So it says, a table by the name employees does not exist. Now, how do we create a table? The command is quite simple and straight forward. It's just create followed by the table name and whatever comes after the table name is going to be the column families. Here, I am trying to create a table by the name employees consisting of two

column families. Let me name the first column family as basic underscore info and I am going to separate it by a comma to mention another column family as personal info.

Now, we have to be very careful while trying to design the column families in a table that is you cannot add more number of column families after you create a table. However, each of these column families, basically we have two column families, basic info and personal info can contain any number of columns and we can dynamically add columns during the run time especially while trying to insert the data but once you create a table with two column families, you cannot modify it. Either you have to delete the entire table and re-create a new one or nothing else can be done.

So, pretty much, we have to be careful in terms of design prospective like how many column families we need to actually have in our original table. That's it and to press enter key, a table by the name employees will be created. To cross check that we can use the exists command on the employees table, says table thus exist.

Since we have just created the table and not entered any kind of data values into it, we can

try to scan it and see that no rows will be return.

How do we check the schema of this table like how do we know how many column families are actually present inside this table? The command would be describe, describe employees. So it says the table employees is going to be having two column families, one of them is going to be called as basic info along with it, you have a bunch of other information and then the second column family name as personal underscore info is mentioned over here.

If we note carefully that the number of versions that is we have discussed about this in our previous videos like how many versions is HBase going to store by default. Here is the information we have. It says the versions is going to be three, three versions is going to be stored. So we shall discuss about what exactly is meant by three versions in the upcoming videos.