**Video Time for Students - 1 hour**

**0) HBASE is a NoSQL database**

**A) What is NOSQL and Why No SQL**

* Discussing what is SQL and data model used for SQL(RDBMS)
* Why popularity of SQL started to decline or for which use cases SQL databases were incompetent
* Hadoop/HDFS solved some of the challenges faced in SQL. But introduced new challenges
* What is a NoSQL database and how it solved the SQL and HDFS-related challenges

Consider to add

* See if you can find a case-study that highlights the above path (Ericsson-case study)
* Where NOSQL is used?

Some useful links for this topic is :

<https://datajobs.com/what-is-hadoop-and-nosq>l

<https://mapr.com/blog/hadoop-vs-nosql-whiteboard-walkthrough/>

**B) CAP Theorem**

* Discussing what is CAP theorem - Consistency, Availability, Partition
* Differentiate RDBMS and NoSQL using CAP theorem

For ATM - Consistency is more critical while on social media, availability is more important

Shankaran Karthikeyan had discussed about NoSql and CAP theorem in his video so we can use that - Use content from Shankaran’s video

**C) Introduction to HBase**

* Some real life industry applications where Hbase is used. Twitter? - add relevant links which can work as additional links to students (source)
* History of Hbase.
  + Who developed it
  + Like inspired by Google’s Big Table etc.
* Features of Hbase
  + Stores data column-wise -- give as additional readings
  + Stores data using a Key-value manner; data retrieval is fast
  + Used to store and retrieve data
  + Hbase is fault tolerant
  + Any more features - offer links for students to read.
* Row storage vs columnar storage - ***Siddharth does this as an SME***

<https://www.ericsson.com/research-blog/horizontal-scalability-hbase/>

**D) Data Model Used in Hbase**

* Discuss how data is stored in Hbase. The answer is table
* How is the table for Hbase different from SQL
  + In the table talk about rowkey, column family, column qualifiers etc
    - Show a lot of examples here and quizzes here
  + Schema related differences

**F) Common Operations performed on a HBase table (25% of student time on this section)**

* Install Hbase
* Discuss some important commands or operations which can be performed on a HBase table
  + Create table - create
  + Insert data - put
  + Retrieve data (use the 2-3 variants of this command) - get and scan
  + Select data

Consider to add

* Someone should do more research to see if there any more commands we can include here.
* <https://learnhbase.wordpress.com/2013/03/02/hbase-shell-commands/>
* Emphasise that this is for students to get comfortable

**G) Java representation of all operations (25% of student time on this section)**

* Build to teach Prof. Hota’s code

Consider to add

* Write one more Java file above and beyond what Prof. Hota has sent
  + Specially retrieve and delete (get and scan)

Some Hbase tutorials are:

ttps://intellipaat.com/tutorial/hbase-tutorial/installation/

<https://www.javatpoint.com/hbase-example>

<https://www.guru99.com/hbase-tutorials.html>

Prof. Hota’s code

1. MyFirstHBaseTable.java is a Java program to create Hbase table.  
In this program we create a table “user” with two field “id” and “name”.  
2. PopulateHBaseTable.java With the help of this program we insert record in to above table and display same records as output.

**H) Optional Student Projects**

* Come up with 3 projects that are skill appropriate for students to do as more practice in Hbase

**I) Architecture of HBase (Text/Optional)**

* Describe the master slave architecture
* Discuss regarding other important components like HMaster, Region server, Zookeeper, mem store, WAL etc