**Session 31**

**Assignment 3**

**Problem Statements:**

**Part1:**

HBase is a schema less database, what does it mean?

* In traditional RDBMS database the schema of the table is predefined and cannot be modified easily.
* This schema is defined while making of the table i.e. the datatype of the columns.
* But in case of the HBASE, the schema is mentioned at time of loading the data. Also the data stored in HBASE is using the bytes hence it is independent of datatype.
* In this way even if new data comes into the database, HBASE can handle it smoothly without any error. This is one of the advantage of HBASE.
* Moreover HBase doesn't have the concept of fixed columns schema. It is a wide column store and has column families which are roughly equivalent to tables.
* The column names can be completely different and the number of columns can vary by row – so you can have a table with billions of rows and could have rows with any number of columns.
* Without HBase you can’t do table joins and so is definitely more of a “schema-less” nature. Hence HBase is a schema-less database. “Schema-less” doesn’t mean the data doesn’t have structure and data models can help us to design, rationalize, and communicate about data.
* For example, key columns can be identified, columns which would require indexing can be identified, and the granularity and relationships of the data can be understood.

**Part 2:**

What is the minimum number of column family every HBase table should have?

* HBase currently does not do well with anything above two or three column families so the number of column families in the schema should be kept low (i.e. set to 1).
* Currently, flushing and compactions are done on a per Region basis so if one column family is carrying the bulk of the data bringing on flushes, the adjacent families will also be flushed though the amount of data they carry is small.
* When many column families the flushing and compaction interaction can make for a bunch of needless input or output loading.
* Try to make the schema done with one column family if can. Only introduce a second and third column family in the case where data access is usually column scoped i.e. querying one column family or the other but usually not both at the one time.

**Part 3:**

What is the benefit of using connection pool in HBase?

* Database connections are often expensive to create because of the overhead of creating a new connection and initialization.
* In turn, connection session initialization often requires time consuming processing to perform user authentication, establish transactional contexts and establish other aspects of the session that are required for subsequent database usage.
* Also the ongoing management of the connections that are established are also one of the limiting factor for creating and destroying connections each time.
* Hence the connections are already created and never disconnected and the connections are given to each client in tandem manner.
* Also time needed for forming new connection is more hence by connection pooling this time can be saved.