**Ass-31.3**

**Explain in brief**

**● Hbase is a schema less database, what does it mean?**

1. A schema-less database does not require conformation to a rigid schema that one is required to live up to through the life of a system.
2. Does not enforce data type limitations on individual values pertaining to one single column type
3. Models the business usage and not a database schema, application or product.
4. Can store structured and unstructured data.
5. Eliminates the need to introduce additional layers (ORM layer) to abstract the relational model and expose it in an object oriented format.

**● What is the minimum number of column family every Hbase table should have?**

1. The current HBase implementation does not handle a lot of column families well.

2. Basically you should try to stick with one and add a second if you have radically different access patterns.

3. There is a limit to the number of column families in HBase. There is one MemStore per Column Family, when one is full, they all flush.

4. The more you add column families there will be more MemStore created and Memstore flush will be more frequent. It will degrade the performance.

**● What is the benefit of using connection pool in Hbase?**

1. Creating connections to a server component from an application is a heavy weight operation and it is much pronounced when connecting to a database server.

2. That being the reason database connection pooling is used to reuse connection objects and HBase is no exception.

3. In HBase, data from meta table that stores details about region servers that can serve data for specific key ranges gets cached at the individual connection level that makes HBase connections much heavier.

4. So if there are region movements for balancing or if a region server fails, the meta data need to be refreshed for each connection object which is a performance overhead.

5. For these reasons, applications need to try to reuse connection objects created.