**1. Explain what is checksum and the importance of checksum and how hadoop performs checksum?**

Answer-

Checksum is a method of checking the data integrity but it does not correct the data if there is loss in the data. The checksum is steps that are performed on the data and get a numerical value this same steps are done after transmission and the earlier value is matched with the present calculated value if the values get matched then there is no loss of data, if different then there is loss of data.

In case of hadoop it uses CRC-32(Cyclic Redundancy Check), HDFS transparently checksums all data written to it and by default verifies checksums when reading data. A separate checksum is created for every io.bytes.per.checksum bytes of data.

The default is 512 bytes, and because a CRC-32 checksum is 4 bytes long, the storage overhead is less than 1%.

**2. Explain the anatomy of file write to HDFS for writing a file the hdfs client calls a function called as create which when, this is called the DFS(Distributed file system )?**

Answer-

RPC call to create a new file to name node. The name node performs all the check about whether the file already exists and if the client has the permissions to write or not. Once all the checks are done then name node creates a new record about this new file and also DFS creates DataFileSystem to write a file finally in response to create hdfs client receives this fsddataoutput Stream. Once the writing has been finished then the close command is called by the client which flushes all the data from dataoutputStream to the datanode pipeline. Each of the datanode stores and sends it to other for replication.

**3. Explain how hdfs handles when write fails?**

Answer-

Suppose that while writing the blocks in the data nodes one of the writing to the data node fails. Immediately the pipeline will be closed and the data nodes which are working fine are given a new identity. This change is conveyed to Name node. Once this is conveyed the failed data node is having partial data of the block that data is removed because it may cause confusion if the data node gets repaired and come back live. The remaining blocks in the queue are moved to the data queue and they are written into the two data nodes(considered replication factor as 3)once done the hdfs recognizes that block has been under replicated and hence generates another cop of the block in some fully functional data node.