Brief of Denormalization

Denormalization is a DBMS optimization technique which we add redundant data specifically to the table to avoid complex and expensive join operation.
As we all know, normalization is necessary when we are creating a database. There are 3 to 5 tiers of normalization procedures to get rid of redundant data in the database. It not only saves disk space but also maintain the data integrity. But it also brings up disadvantages to a certain extent.
For instance, we have two tables, the first one is student table (student id, course taken, course id) with student id as the entry. The second one is course table (course id , teacher) with course id as entry. If we want to implement a query that will return the combination of the course taken from student table and teacher teaching the class from the course table, then we have to perform a join operation to do it.
Join operation could be expensive if we have too much data in the table to combine. That's the purpose of denormalization, to reduce the number of cost operation and simplify the query. Compared to normalization, denormalization won't maintain data integrity anymore, and insertion and update will be more expensive since we have more than one table to update. Also, it necessities more storage space.

Brief of Sharding and partitioning

Sharding is a method for distributing a single dataset across multiple databases which can be stored on multiple machines. This enables large datasets to be split in smaller chunks and stored in multiple data nodes.
User has to define the distribution strategy of collection content across the availability shards. The advantage of sharing is increase the performance of operation on humongous dataset by dividing the datasets Into multiple chunks and distributing them across different databases.Each chunk contains a range of values to a shard, so if the user only needs to query the divided smaller chunk that conation the data needed.
Ps: Each shard key will require an index or auto-build one, as part of the sharding command, if the collection is empty and the required index is missing.

Replica is used to improve the reliability of the system, by making copy of each chunk distributed to different databases.A change that is made in one subscription database is copied to the Publisher (merged), and the Publisher then replicates that change to the other Subscribers. On fixed Subscribers, this is a continuous process and you are not aware of it happening, synchronization usually occurs in a matter of seconds. The effect is that all users on fixed Subscribers have an almost identical view of the data even though they are working on different databases.