?in using begin transaction statement and when

? In implicit Transaction mode the transaction begins automatically but at the end of transaction we have specify whether the transaction should be committed or rolled back using the COMMIT or ROLLBACK commands. If a user doesn't specify any thing at the end of the transaction then all the changes to the data are rolled back when the user disconnects.

We can the implicit transaction mode by SET IMPLICIT_TRANSACTIONS to ON In explicit transaction mode the user must mention when the transaction is going to start using BEGIN TRANSACTION statement and when the transaction is going to end with the COMMIT or ROLLBACK commands. In this mode we can use nested transactions. Duration of Explicit transactions exists only till the transaction end after that it automatically transaction mode changes to the original mode which it was before.

Exclusive lock is not compatible with any other locks or any locks are not compatible with an exclusive lock. An exclusive lock is used to reserve a page or row limited for a single transaction. We cannot set exclusive lock if there is any other processes holds a shared or exclusive lock on the resource. If we set exclusive lock for the row or a page then we cannot place any other lock on the same resource. It is used for DML statements which modify the resource.

When a data in a database is in a process of being changed by one user or program then at the same time if another user reads the same data from the database then the DBMS should prohibit access to the program or user until the modification is complete to ensure the data integrity. To enable multiple users access the data and modify the data in the database DBMS uses a mechanism called locking. To ensure the data integrity locks are used. When a database resource is locked by one process, another process is not permitted to change the locked data. So, Data can be locked at different levels in the database like page- level, row-level and Table-level. Row-level locking: Row-level locking means that the particular row which the user or application is using is locked and it is not used by any other application. Hence, the other applications can use remaining all other rows which belongs to the same page. So it increases the usage of database by multiple users simultaneously.

But locking the row-level requires more resources since for each row we require one lock and each page contains N number of rows so page level locking requires less memory and CPU. Page-level locking: If an application uses all the rows in a page then instead of locking the each and every row we can lock entire page by page locking which decreases the usage of resources.