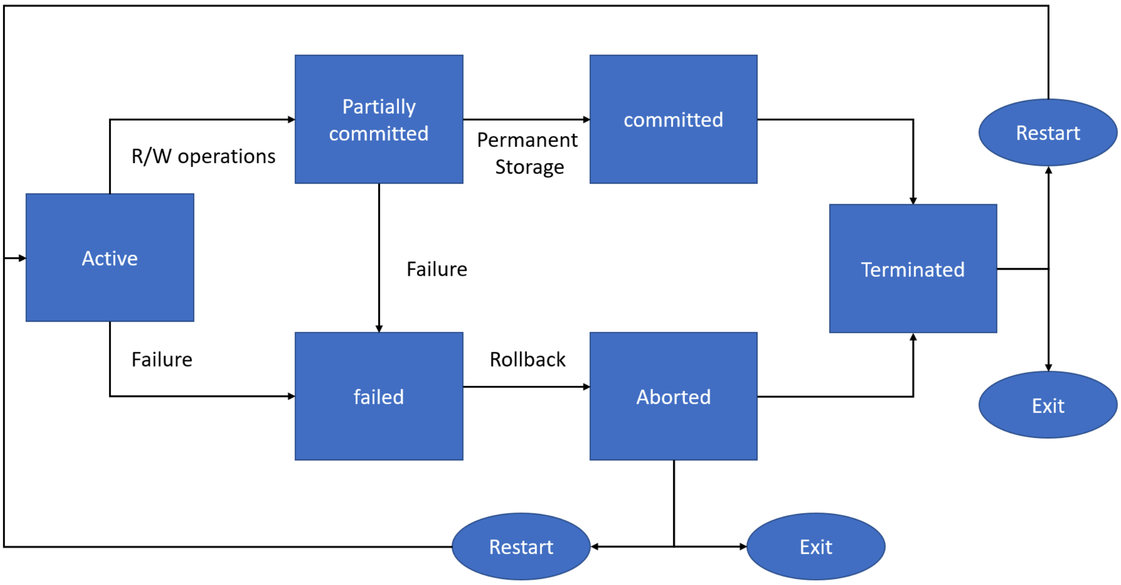
STATES OF TRANSACTION

1. A Transaction state shows the current and upcoming condition or state of a transaction in DBMS.
2. A transaction goes through so many states in its whole lifespan.

Transaction states are as follows-

1. Active state
2. Partially committed state
3. Committed state
4. Failed state
5. Aborted state
6. Terminated state



ACTIVE STATE

1. This is the first state of every transaction.
2. Here the transaction is being executed.
3. A transaction is called in an active state as long as its instructions are getting executed.
4. All the changes made by the transaction now are stored in the buffer in main memory.
5. If all the ‘read and write’ operations are performed without any error then it goes to the “partially committed state”.
6. If any instruction fails, it goes to the “failed state”.

7. For example, Insertion or deletion or updating a record is done here. But all the records are still not saved to the database.

PARTIALLY COMMITTED

1. In the partially committed state, a transaction executes its final operation, but the data is still not saved to the database.
2. After the last instruction of transaction has executed, it enters into a partially committed state.
3. After entering this state, the transaction is considered to be partially committed.
4. It is not considered fully committed because all the changes made by the transaction are still stored in the buffer in main memory.
5. After completion of all the read and write operation the changes are made in main memory or local buffer.
6. If the changes are made permanent on the Database, then the state will change to "committed state".
7. in case of failure, it will go to the "failed state".
8. In the total mark calculation example, a final display of the total marks step is executed in this state.

COMMITTED

1. A transaction is said to be in a committed state if it executes all its operations successfully. In this state, all the effects are now permanently saved on the database system.
2. After all the changes made by the transaction have been successfully stored into the database, it enters into a committed state.
3. It is the state when the changes are made permanent on the Data Base and the transaction is complete and therefore terminated in the "terminated state".
4. Now, the transaction is considered to be fully committed.

FAILED STATE

1. If any of the checks made by the database recovery system fails, then the transaction is said to be in the failed state.
2. When a transaction is getting executed in the active state or partially committed state and some failure occurs due to which it becomes impossible to continue the execution, it enters into a failed state.
3. When any instruction of the transaction fails, it goes to the "failed state" or if failure occurs in making a permanent change of data on Database.

4. In the example of total mark calculation, if the database is not able to fire a query to fetch the marks, then the transaction will fail to execute.

ABORTED

1. If any of the checks fail and the transaction has reached a failed state then the database recovery system will make sure that the database is in its previous consistent state.
2. If not then it will abort or roll back the transaction to bring the database into a consistent state.
3. If the transaction fails in the middle of the transaction, then before executing the transaction, all the executed transactions are rolled back to its consistent state.
4. After aborting the transaction, the database recovery module will select one of the two operations:
    1) Re-start the transaction
    2) Kill the transaction

TERMINATED STATE

1. If there isn't any roll-back or the transaction comes from the "committed state".
2. Then the system is consistent and ready for new transaction and the old transaction is terminated.