**LOCKING**

* Locking is used to manage transaction concurrency
* Locks are in-memory structures (96 bytes) which have owners and types and the hash of the resource it should protect
* Locking is an essential part of the isolation requirement of databases
* When a transaction imposes a lock on an object, all other transactions that require access must wait until the resource is released

**LOCK MODES**

Exclusive (X)

* Imposed when transaction wants to modify data (DELETE, UPDATE, INSERT)
* Ensures a page or row is reserved exclusively for the transaction
* Imposed only if no other shared or exclusive lock exists on the target
* Only one per page or row

Shared (S)

* Also known as a Read lock
* Reserves page/row for reading only, any other transactions are prevented from modifying the record while this lock is active
* Can be imposed by >1 transaction over the same page/row. Hence, they share the ability to read data
* Will allow write operations but no DDL changes

Update (U)

* Similar to exclusive lock but more flexible
* Can be imposed on a record that already has a shared lock. Here the update lock imposes another shared lock on the target row
* When transaction holding update lock is ready to change the data it becomes an exclusive lock
* Asymmetrical regarding shared locks; U can be imposed on a record with S lock but not the reverse

Intent (I)

* Transaction informs another transaction of its intent to acquire a lock
* Ensure data modification is correctly executed in the hierarchy
* Three conversion (intent) locks:

1. Shared with intent exclusive (SIX)
2. Shared with intent update (SIU)
3. Update with intent exclusive (UIX)

* There are 3 regular intent locks:

1. Intent exclusive (IX)
2. Intent shared (IS)
3. Intent update (IU)

Schema (Sch.)

* Two types of schema locks:

1. Schema modification lock (Sch-M) – when a DDL statement is executed
2. Schema stability lock (Sch-S) – acquired when a schema dependent query is being compiled and executed and execution plan is being generated

Bulk Update (BU)

* Designed for bulk update operations when issued with a TABLOCK hint
* Once acquired other processes will not be able to access a table during bulk load execution