* Brushed up backup concept
* TRIGGERS
* After update trigger, After delete, After insert(practised on ssms)
* Instead of update, instead of insert, insert of delete.
* Store procedure functions
* Backup permissions: for taking permission we need permission of:
* System Admin (sa)
* DB owner
* DB creator
* DB Backup operator

**COMPRESSED BACKUP:**

* When ever we take backup it compresses the backup ,it will not affect the database and also saves the disk storage. Once it is restored it gets uncompressed
* We can use sp\_configure store procedure with backup compress default, 1 for enable and 0 to disable default compression of all backups in server
* Syntax: sp\_configure ‘backup\_compression\_default; ‘1’ Reconfigure with override.
* To set backup compression : backup database database\_name to disk=’path.bak’ with compression=1.

**TAIL LOG BACKUP:**

* This feature only works if we have taken full backup and full recovery/bulk recovery
* It contains the log-records which was has not been backed up by the last backup transaction.
* Consider we have made some changes after taking full backup then we will miss the changed logs to overcome this issue trail log backup is used tail log backup
* To avoid losing last transactions it’s better to take tail log backup with *Norecovery model*
* **SYNTAX:** BACKUP LOG DATABASE\_NAME To disk=’path.trn’, with norecovery
* After taking the tail log backup the database will move to restoring mode and no further transactions will b possible.
* Rules to be followed for tail log backup:
* Restore the full backup
* Restore the differential backup
* Restore the log transaction
* Restore the tail log backup with norecovery

**BACKUP OPTIONS:**

INTI/ NO INIT:

* When we take a backup with init then the backup stores the record history of backup in MSDB but it overwrites the existing record.
* Noinit will write the new record under existing record

SKIP/ NO SKIP:

* To skip the verification process of checking the backup whether it properly write a copy or not
* No skip will verify the verification process.

BACKUP HISTORY:

* We can get the information of the database who has taken the backup and what type of backup it is.
* Syntax: select \* from msdb.dbo.backupset.

CURSOR:

* A Cursor allow us to retrieve data from a result set in singleton fashion means row by row.
* Cursor are required when we need to update records in a database table one row at a time.
* Types of cursors:
  + - Implicit
    - Explicit
* Implicit:
* Default cursor of sql server database
* The cursor server is allocated by db server when we perform DML operations
* Explicit:
* These are created by users
* Fetch data from table in row-by-row manner
* Working with Explicit cursor:

Step1: Declare cursor object

Step2: Open cursor connection

Step3: Fetch data from cursor

Step4: close cursor connection

Step5: Deallocate the memory

**11/12/2020**