BACKUP DATABASE [PRODUCT\_DATABASE] TO DISK = N'D:\BACKUPS\BackupDiff.BAK' WITH DIFFERENTIAL , DESCRIPTION = N'This is a differential backup for Product\_Database', RETAINDAYS = 14, NOFORMAT, INIT, NAME = N'PRODUCT\_DATABASE-Full Database Backup', SKIP, NOREWIND, NOUNLOAD, COMPRESSION, STATS = 10, CHECKSUM, CONTINUE\_AFTER\_ERROR

GO

declare @backupSetId as int

select @backupSetId = position from msdb..backupset where database\_name=N'PRODUCT\_DATABASE' and backup\_set\_id=(select max(backup\_set\_id) from msdb..backupset where database\_name=N'PRODUCT\_DATABASE' )

if @backupSetId is null begin raiserror(N'Verify failed. Backup information for database ''PRODUCT\_DATABASE'' not found.', 16, 1) end

RESTORE VERIFYONLY FROM DISK = N'D:\BACKUPS\BackupDiff.BAK' WITH FILE = @backupSetId, NOUNLOAD, NOREWIND

GO

BACKUP LOG [PRODUCT\_DATABASE] TO DISK = N'D:\Backups\LogBackup.trn' WITH FORMAT, INIT, MEDIANAME = N'Log Backup for Product\_Database', NAME = N'PRODUCT\_DATABASE-Full Database Backup', SKIP, NOREWIND, NOUNLOAD, STATS = 10, CHECKSUM, CONTINUE\_AFTER\_ERROR

GO

declare @backupSetId as int

select @backupSetId = position from msdb..backupset where database\_name=N'PRODUCT\_DATABASE' and backup\_set\_id=(select max(backup\_set\_id) from msdb..backupset where database\_name=N'PRODUCT\_DATABASE' )

if @backupSetId is null begin raiserror(N'Verify failed. Backup information for database ''PRODUCT\_DATABASE'' not found.', 16, 1) end

RESTORE VERIFYONLY FROM DISK = N'D:\Backups\LogBackup.trn' WITH FILE = @backupSetId, NOUNLOAD, NOREWIND

GO

BACKUP DATABASE [PRODUCT\_DATABASE]

FILEGROUP = N'PRIMARY' TO DISK = N'D:\BACKUPS\File\_FIlegroupBackup.BAK'

WITH COPY\_ONLY, NOFORMAT, NOINIT, NAME = N'PRODUCT\_DATABASE-Full Database Backup', SKIP, NOREWIND, NOUNLOAD, STATS = 10

GO

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* FOR PRACTICE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

-- 1. FULL DATABASE BACKUP:

BACKUP DATABASE [PRODUCT\_DATABASE] TO DISK = 'E:\PRODUCTDB\_BACKUP.bak' WITH FORMAT

-- 2. DIFFERENTIAL DATABASE BACKUP :

BACKUP DATABASE [PRODUCT\_DATABASE] TO DISK = 'E:\PRODUCTDB\_BACKUP.bak' WITH FORMAT, DIFFERENTIAL

-- 3. FULL FILEGROUP BACKUP :

BACKUP DATABASE [PRODUCT\_DATABASE] FILEGROUP = 'PRIMARY' TO DISK = 'E:\PRODUCTDB\_BACKUP.bak' WITH FORMAT

-- 4. DIFFERENTIAL FILEGROUP BACKUP :

BACKUP DATABASE [PRODUCT\_DATABASE] FILEGROUP = 'PRIMARY' TO DISK = 'E:\PRODUCTDB\_BACKUP.bak' WITH FORMAT, DIFFERENTIAL

-- 5. FULL FILE BACKUP :

BACKUP DATABASE [PRODUCT\_DATABASE] FILE = 'PRODUCT DATABASE' TO DISK = 'E:\PRODUCTDB\_BACKUP.bak' WITH FORMAT

-- 6. DIFFERENTIAL FILE BACKUP :

BACKUP DATABASE [PRODUCT\_DATABASE] FILE = 'PRODUCT DATABASE' TO DISK = 'E:\PRODUCTDB\_BACKUP.bak' WITH FORMAT, DIFFERENTIAL

-- 7. REGULAR LOG BACKUP :

BACKUP LOG [PRODUCT\_DATABASE] TO DISK = 'E:\PRODUCTDB\_BACKUP.trn' WITH FORMAT

-- 8. COPY ONLY BACKUP :

BACKUP DATABASE [PRODUCT\_DATABASE] TO DISK = 'E:\PRODUCTDB\_BACKUP\_COPY.bak' WITH NOFORMAT, COPY\_ONLY

-- 9. MIRRORED BACKUP

BACKUP DATABASE [PRODUCT\_DATABASE] TO

DISK = 'E:\PRODUCTDB\_BACKUP\_COPY1.bak'

MIRROR TO

DISK = 'D:\PRODUCTDB\_BACKUP\_COPY2.bak'

WITH FORMAT

-- 10. SPLIT BACKUP -- BELOW TWO BACKUP FILES FORM ONE BACKUP MEDIA FAMILY

BACKUP DATABASE [PRODUCT\_DATABASE] TO

DISK = 'E:\PRODUCTDB\_BACKUP\_PART1.bak'

,

DISK = 'D:\PRODUCTDB\_BACKUP\_PART2.bak'

WITH FORMAT

-- 11. PARTIAL BACKUP

BACKUP DATABASE [PRODUCT\_DATABASE]

READ\_WRITE\_FILEGROUPS

TO DISK = 'E:\PRODUCTDB\_BACKUP\_PARTIAL.bak' WITH FORMAT, COMPRESSION, CHECKSUM

-- 12. HOT BACKUPS

BACKUP DATABASE [PRODUCT\_DATABASE] TO DISK = N'E:\PRODUCTDB\_BACKUPS.bak' WITH RETAINDAYS = 14, NOFORMAT, NOINIT, NAME = N'BankingDB-Full Database Backup', SKIP, NOREWIND, NOUNLOAD, COMPRESSION, STATS = 10, CHECKSUM, CONTINUE\_AFTER\_ERROR

GO

declare @backupSetId as int

select @backupSetId = position from msdb..backupset where database\_name=N'PRODUCT\_DATABASE' and backup\_set\_id=(select max(backup\_set\_id) from msdb..backupset where database\_name=N'BankingDB' )

if @backupSetId is null begin raiserror(N'Verify failed. Backup information for database ''[PRODUCT\_DATABASE]'' not found.', 16, 1) end

RESTORE VERIFYONLY FROM DISK = N'E:\PRODUCTDB\_BACKUPS.bak' WITH FILE = @backupSetId, NOUNLOAD, NOREWIND

GO

-----------------------------------------------------------------------------------------

--- BACKUP AUDITS

SELECT \* FROM MSDB.DBO.backupset -- CONTAINS AN ENTRY FOR EVERY BACKUP WE PERFORM ON THE SERVER

SELECT \* FROM MSDB.DBO.backupfile -- CONTAINS AN ENTRY FOR EVERY DATA FILE OR LOG FILE INVOVLED IN THE BACKUP

SELECT \* FROM MSDB.DBO.backupfilegroup -- CONTAINS AN ENTRY FOR EVERY FILEGROUP INVOVLED IN THE BACKUP

SELECT \* FROM MSDB.DBO.backupmediaset -- CONTAINS AN ENTRY FOR EVERY UNIQUE BACKUP (FILE) LOCATION

SELECT \* FROM MSDB.DBO.backupmediafamily -- CONTAINS AN ENTRY FOR EVERY BACKUP LOCATION & ITS PHYSICAL PATH

-- HOW TO USE ABOVE BACKUP AUDIT TABLES?

-- EXAMPLE: HOW TO AUDIT FOR LATEST FULL BACKUP?

SELECT TOP 1 B.DATABASE\_NAME, B.TYPE, B.backup\_finish\_date , M.physical\_device\_name AS BACKUP\_LOCATION

FROM MSDB.DBO.backupset AS B

INNER JOIN

MSDB.DBO.backupmediafamily AS M

ON

B.media\_set\_id = M.media\_set\_id

WHERE B.TYPE = 'D' AND B.DATABASE\_NAME = 'PRODUCT\_DATABASE'

ORDER BY B.backup\_finish\_date DESC

-- EXAMPLE: HOW TO AUDIT FOR LATEST DIFFERENTIAL BACKUP?

SELECT TOP 1 B.DATABASE\_NAME, B.TYPE, B.backup\_finish\_date , M.physical\_device\_name AS BACKUP\_LOCATION

FROM MSDB.DBO.backupset AS B

INNER JOIN

MSDB.DBO.backupmediafamily AS M

ON

B.media\_set\_id = M.media\_set\_id

WHERE B.TYPE = 'I' AND B.DATABASE\_NAME = 'PRODUCT\_DATABASE'

ORDER BY B.backup\_finish\_date DESC

-- EXAMPLE: HOW TO AUDIT FOR LOG BACKUPS AFTER A SPECIFIC DATE?

SELECT TOP 1 B.DATABASE\_NAME, B.TYPE, B.backup\_finish\_date , M.physical\_device\_name AS BACKUP\_LOCATION

FROM MSDB.DBO.backupset AS B

INNER JOIN

MSDB.DBO.backupmediafamily AS M

ON

B.media\_set\_id = M.media\_set\_id

WHERE B.TYPE = 'L' AND B.DATABASE\_NAME = 'PRODUCT\_DATABASE'

AND

backup\_finish\_date > '2018-10-23 18:19:16.000'

ORDER BY B.backup\_finish\_date DESC

-----------------------------------------------------------------------------------------

/\*

RECOVERY MODELS : A PROPERTY USING WHICH WE CAN CONTROL THE LEVEL OF DATABASE LOGGING OPERATIONS.

TYPES OF RECOVERY MODELS:

\*\*\* 1. FULL RECOVERY MODEL - EVERY TRANSACTION METADATA + ACTUAL DATA IS AUDITTED

2. SIMPLE RECOVERY MODEL - EVERY TRANSACTION METADATA AUDITTED

3. BULK LOGGED RECOVERY MODEL - EVERY DETAIL IS COMPLETELY LOGGED EXCEPT BULK OPERATIONS.

THESE BULK OPERATIONS ARE MINIMALLY LOGGED.

DEFAULT RECOVERY MODEL IS "FULL".

LOG BACKUPS ARE NOT POSSIBLE IN SIMPLE RECOVERY MODEL.

IN SIMPLE RECOVERY MODEL, DATABASE REQUIRES LESSER STORAGE SPACE.

\*\* IT IS RECOMMENDED TO PERFORM A NON BASE FULL BACKUP BEFORE CHANGING THE RECOVERY MODEL OF THE DATABASE.

\*\* TRANSACTION METADATA :: TRANSACTION TIMESTAMP, STATUS, KIND OF OPERATION, SESSION ID, LOGIN NAME....

\*/

ALTER DATABASE [PRODUCT\_DATABASE] SET RECOVERY FULL

ALTER DATABASE [PRODUCT\_DATABASE] SET RECOVERY SIMPLE

ALTER DATABASE [PRODUCT\_DATABASE] SET RECOVERY BULK\_LOGGED