**Native Backup and Restore PowerShell Scripts**

**Backup-MT\_SQLDB**

Synopsis:

Performs Native backup of specified database on specified server.

Description:

Upon passing backupservername and database name, Function will perform backup of that database on specified server. Single or multiple databases can be passed for backup. If no value is specified, then backup of all databases will be taken. Detailed Log file will be saved in location *c:\temp\Backup-MT\_SQLDB\_\*.*

Syntax:

Backup-MT\_SQLDB -backupserver hostname\instancename -backupdatabase 'dbname1','dbname2' -Type 'FULL' -backuplocation '\\hostname\E$\backup\disk1\foldername

*Parameters*:

*Backupserver*(Mandatory)

hostname\instancename - source servername for backup.

*Backupdatabase*

database name for backup on source server.

*Type*

To specify type of backup. Type can be 'FULL', 'Log','Differential','Diff'

*Backuplocation*

Location for backup files(unc path only). If not specified default location will be used. If does not exits backup location will be created.

*Example 1:*

*Backup-MT\_SQLDB -backupserver hostname\instancename -backupdatabase'db1','db2','db3'*

Performs backup of databases db1,db2 and db3 on backup server hostname\instance

*Example 2:*

*Backup-MT\_SQLDB -backupserver hostname\instancename -backupdatabase 'db1','db2','db3' -Type log*

Performs log backup of databases db1,db2 and db3

*Example 3:*

*$backuplocation = "\\hostname\E$\backup\disk1\MSSQL11.Instancename\Foldername"*

*$dbs= ('db1','db2','db3')*

*Backup-MT\_SQLDB -backupserver hostname\instancename -backupdatabase $dbs -Type Full -backuplocation $backuplocation*

Performs full backup of all databases in array $dbs to $backuplocation

*Example 4:*

*$backuptable=@{}*

*$backuptable. \*\*\*\*TRY THIS\*\*\*\*\*\**

*$srvs= 'hostname1\instancename1','hostname2\instnacename2'*

*$srvs.GetEnumerator().foreach{Backup-MT\_SQLDB -backupserver $\_ }*

Backups all databases mentioned in array list $srvrs

*Example 5:*

*$srvs= 'hostname1\instancename1','hostname2\instnacename2'*

*$srvs.GetEnumerator().foreach{Backup-MT\_SQLDB -backupserver $\_ -backupdatabase 'sa'}*

Backups sa database on all servers in the array list $srvs

***Restore-MT\_SQLDB***

*Synopsis*:

Performs restore of database on specified server when correct backup files or location is provided

*Description*:

Upon passing correct backup location\file function restores database specified in backuplocation or backup files

Backup location should be a UNC Path - \\hostname\backuplocation. Restoring databases can be done with replace or recovery. Please check detailed log in location *c:\temp\Restore-MT\_SQLDB\_\*.txt*

*Parameters*:

*RestoreServername(mandatory)*

Parameter to store single value restore servername - hostname\instancename

*Restoredatabase*

Parameter to store array of source database name - 'databasename1','databasename2',...

*backuplocation*(mandatory)

Parameter to store location of backup files. - uncpath -\\hostname\location\_of\_file

*Recovery*(switch)

Enable this switch to restore database with full recovery -Recovery. If not mentioned default value restore with norecovery will be triggered.

*Replace*(switch)

Enable this switch to restore database with replace -Replace. If not mentioned default value restore with no replace will be triggered.

*Example 1:*

*Restore-MT\_SQLDB -Restoreservername hostname\instancename -restoredatabase 'db1' -backuplocation '\\servername\E$\backup\disk1\foldername\db1.bak'*

Restores database on single server with name db1 with backup file in mentioned in backuplocation parameter

*Example 2:*

*Restore-MT\_SQLDB-Restoreservername hostname\instancename -restoredatabase 'db1','db2' -backuplocation '\\servername\E$\Backup\Disk1\foldername\'*

Restore all backup files from foldername into db1, db2 with default options with no replace and norecovery.

*Example 3*

*Restore-MT\_SQLDB-Restoreservername hostname\instancename -restoredatabase 'db1','db2' -backuplocation '\\servername\E$\Backup\Disk1\foldername\' -replace -recovery*

Restore all backup files from foldername into db1, db2 with default options with replace and recovery

*Example 4*

*$backupinfo = Backup-MT\_SQLDB -backupserver hostname\instancename -backupdatabase'db1','db2','db3'*

*Restore-MT\_SQLDB -Restoreservername hostname\instancename -backuplocation $backupinfo.uncpath -replace -recovery*

Uses return value of backup-MT\_SQLDB command as input for restore. Restores all the databases that are in $backupinfo.uncpath with same name.

*Example 5:*

*$backupinfo = Backup-MT\_SQLDB -backupserver hostname\instancename -backupdatabase'db1','db2','db3'*

*$backupinfo.uncpath | Restore-MT\_SQLDB -Restoreservername hostname\instancename -Replace -Recovery*

Restores all databases present in location $backupinfo.uncpath with same name.

*Example 6:*

*Restoring same database with one backup file to multiple servers*

*$backuplocation="\\hostname\E$\backup\disk1\MSSQL11.Instancename\Foldername\filename.bak"*

*$srvs= 'hostname1\instancename1','hostname2\instnacename2'*

*$srvs.GetEnumerator().foreach{Restore-MT\_SQLDB -Restoreservername $\_ -backuplocation $backuplocation -recovery}*

*for with replace restore:*

*$srvs.GetEnumerator().foreach{Restore-MT\_SQLDB -Restoreservername $\_ -backuplocation $backuplocation -replace -recovery}*

*Notes*

If you want to restore multiple database with different recovery and replace options. Use seperate restore commands to match replace and recovery options.

example:

db1, db2 with replace and db3 with no recovery,

use:

Restore-MT\_SQLDB -restoreservername hostname\instancename -restoredatabase 'db1', 'db2' -Replace -recovery

Restore-MT\_SQLDB -restoreservername hostname\instancename -restoredatabase 'db3' -Replace

Working:

* Make sure dbtools is installed on server. Script uses dbtools to function
* Copy Backup-MT\_DDBoost\_DB.ps1,Restore-MT\_DDBoost.ps1 and Write-log.ps1 to same folder. Otherwise script throws an error.
* Open Powershell and navigate to above folder location where you copied all three files. Run all three files so that they are loaded into powershell.
* Please check detailed log in Please check detailed log in location c:\temp\Restore-MT\_SQLDB\_\*
* Follow above examples for any reference.