Windows PowerShell Get-Help on Cmdlet 'New-SqlAvailabilityGroup'

PS:\>Get-HELP New-SqlAvailabilityGroup -Full

NAME

New-SqlAvailabilityGroup

SYNOPSIS

Creates an availability group.

SYNTAX

New-SqlAvailabilityGroup [-Name] <String> [-InputObject] <Server> [-AccessToken <PSObject>] [-AutomatedBackupPreference {Primary | SecondaryOnly | Secondary | None | Unknown}] -AvailabilityReplica <AvailabilityReplica[]> [-BasicAvailabilityGroup] [-ClusterType {Wsfc | None | External}] [-ContainedAvailabilityGroup] [-Database <String[]>] [-DatabaseHealthTrigger] [-DtcSupportEnabled] [-Encrypt {Mandatory | Optional | Strict}] [-FailureConditionLevel {OnServerDown | OnServerUnresponsive | OnCriticalServerErrors | OnModerateServerErrors | OnAnyQualifiedFailureCondition | Unknown}] [-HealthCheckTimeout <Int32>] [-HostNameInCertificate <String>] [-ProgressAction <ActionPreference>] [-RequiredSynchronizedSecondariesToCommit <Int32>] [-ReuseSystemDatabases] [-Script] [-TrustServerCertificate] [-Confirm] [-WhatIf] [<CommonParameters>]

New-SqlAvailabilityGroup [-Name] <String> [[-Path] <String>] [-AccessToken <PSObject>] [-AutomatedBackupPreference {Primary | SecondaryOnly | Secondary | None |

Unknown}] -AvailabilityReplica <AvailabilityReplica[]> [-BasicAvailabilityGroup] [-ClusterType {Wsfc | None | External}] [-ContainedAvailabilityGroup] [-Database

<String[]>] [-DatabaseHealthTrigger] [-DtcSupportEnabled] [-Encrypt {Mandatory | Optional | Strict}] [-FailureConditionLevel {OnServerDown | OnServerUnresponsive |

OnCriticalServerErrors | OnModerateServerErrors | OnAnyQualifiedFailureCondition | Unknown}] [-HealthCheckTimeout <Int32>] [-HostNameInCertificate <String>]

[-ProgressAction <ActionPreference>] [-RequiredSynchronizedSecondariesToCommit <Int32>] [-ReuseSystemDatabases] [-Script] [-TrustServerCertificate] [-Confirm]

[-WhatIf] [<CommonParameters>]

DESCRIPTION

The New-SqlAvailabilityGroup cmdlet creates an availability group in Always On Availability Groups.

The InputObject or Path parameter specifies the server that hosts the initial primary replica.

> `Module requirements: version 21+ on PowerShell 5.1; version 22+ on PowerShell 7.x.`

PARAMETERS

-AccessToken <PSObject>

The access token used to authenticate to SQL Server, as an alternative to user/password or Windows Authentication.

This can be used, for example, to connect to `SQL Azure DB` and `SQL Azure Managed Instance` using a `Service Principal` or a `Managed Identity`.

The parameter to use can be either a string representing the token or a `PSAccessToken` object as returned by running `Get-AzAccessToken -ResourceUrl

https://database.windows.net`.

> This parameter is new in v22 of the module.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AutomatedBackupPreference <AvailabilityGroupAutomatedBackupPreference>

Specifies the automated backup preference for the availability group.

The acceptable values for this parameter are:

- Primary. Specifies that the backups always occur on the primary replica. This option supports the use of features not available when backup runs on a secondary replica, such as differential backups. - SecondaryOnly. Specifies that backups are never performed on primary replicas. If the primary replica is the only replica online, the backup does not occur. - Secondary. Specifies that backups occur on secondary replicas, unless the primary replica is the only replica online. Then the backup occurs on the primary replica. - None. Specifies that the primary or secondary status is not taken into account when deciding which replica performs backups. Instead, backup priority and online status determine which replica performs backups.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AvailabilityReplica <AvailabilityReplica[]>

Specifies an array of availability replicas that this cmdlet includes in the availability group. To obtain an AvailabilityReplica , use the

New-SqlAvailabilityReplica cmdlet. Specify the AsTemplate parameter.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-BasicAvailabilityGroup [<SwitchParameter>]

Specifies whether to create an `advanced` (default) or a `basic` availability group.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-ClusterType <AvailabilityGroupClusterType>

The type of cluster backing the AG. Possible values are:

- Wsfc. The AG will be integrated in Windows Server Failover Cluster. This is how AGs in SQL Server 2016 and below are created. This is the default. - None. The AG will be cluster-independent.

- External. The AG will be managed by a cluster manager that is not a Windows Server Failover Cluster, like Pacemaker on Linux.

This is supported in SQL Server 2017 and above. When targeting SQL Server on Linux, you must specify this value or an error will occour.

Note: An exception will be thrown if the -ClusterType parameter is used when the target server is SQL Server 2016 and below.

Required? false

Position? named

Default value Wsfc

Accept pipeline input? False

Accept wildcard characters? false

-ContainedAvailabilityGroup [<SwitchParameter>]

Used to create a contained availability group. This option is used to create an availability group with its own `master` and `msdb` databases, which are kept in

sync across the set of replicas in the availability group. This parameter may be used with its companion -ReuseSystemDatabases .

> This parameter is allowed only when the target SQL Server supports Contained Availability Groups (SQL 2022 and above). Trying to use is against versions of SQL

that do not support Contained Availability Groups would cause the cmdlet to throw an error.

> This parameter is only available in version 22+ of the module.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Database <String[]>

Specifies an array of local, read/write user databases. These databases must use the full recovery model and must not use AUTO_CLOSE. These databases cannot

belong to another availability group and cannot be configured for database mirroring. You must specify a value for this parameter.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-DatabaseHealthTrigger [<SwitchParameter>]

Specifies whether to trigger an automatic failover of the availability group if any user database replica within an availability group encounters a database

failure condition.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-DtcSupportEnabled [<SwitchParameter>]

Specifies whether databases within an availability group register with MSDTC at the instance-level (default) or at the per-database level.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Encrypt <String>

The encryption type to use when connecting to SQL Server.

This value maps to the `Encrypt` property `SqlConnectionEncryptOption` on the SqlConnection object of the Microsoft.Data.SqlClient driver.

In v22 of the module, the default is `Optional` (for compatibility with v21). In v23+ of the module, the default value will be 'Mandatory', which may create a

breaking change for existing scripts.

> This parameter is new in v22 of the module.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-FailureConditionLevel <AvailabilityGroupFailureConditionLevel>

Specifies the automatic failover behavior of the availability group. The acceptable values for this parameter are:

- OnServerDown. Failover or restart if the SQL Server service stops.

- OnServerUnresponsive. Failover or restart if any condition of lower value is satisfied, plus when the SQL Server

service is connected to the cluster and the HealthCheckTimeout threshold is exceeded, or if the availability replica currently

in primary role is in a failed state. - OnCriticalServerError. Failover or restart if any condition of lower value is satisfied, plus when an internal

critical Server error occurs, which include out of memory condition, serious write-access

violation, or too much dumping. - OnModerateServerError. Failover or restart if any condition of lower value is satisfied, plus if a moderate Server

error occurs, which includes persistent out of memory condition. - OnAnyQualifiedFailureConditions. Failover or restart if any

condition of lower value is satisfied, plus if a qualifying failure condition occurs, which includes engine worker thread

exhaustion and unsolvable deadlock detected.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-HealthCheckTimeout <Int32>

Specifies the length of time, in milliseconds, after which Always On availability groups declare an unresponsive server to be unhealthy.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-HostNameInCertificate <String>

The host name to be used in validating the SQL Server TLS/SSL certificate. You must pass this parameter if your SQL Server instance is enabled for Force

Encryption and you want to connect to an instance using hostname/shortname. If this parameter is omitted then passing the Fully Qualified Domain Name (FQDN) to

-ServerInstance is necessary to connect to a SQL Server instance enabled for Force Encryption.

> This parameter is new in v22 of the module.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-InputObject <Server>

Specifies the instance of SQL Server that hosts the primary replica of the availability group that this cmdlet creates.

Required? true

Position? 2

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-Name <String>

Specifies the name of the availability group that this cmdlet creates.

Required? true

Position? 1

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Path <String>

Specifies the path of the instance of SQL Server that hosts the initial primary replica of the availability group that this cmdlet creates. If you do not specify

this parameter, this cmdlet uses current working location. If you specify a value, the path must currently exist.

Required? false

Position? 2

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ProgressAction <ActionPreference>

Determines how PowerShell responds to progress updates generated by a script, cmdlet, or provider, such as the progress bars generated by the Write-Progress

cmdlet. The Write-Progress cmdlet creates progress bars that show a command's status.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-RequiredSynchronizedSecondariesToCommit <Int32>

The number of synchronous commit secondaries that must be available to be able to commit on the primary.

If a `SYNCHRONOUS_COMMIT` secondary is disconnected from the primary for some time, the primary demotes it to `ASYNCHRONOUS_COMMIT` to avoid blocking commits. If

the primary then becomes unavailable and the user wishes to fail over to one of these secondaries, they may incur data loss. By setting

RequiredSynchronizedSecondariesToCommit to some number, the user can prevent the data loss since the primary will start blocking commits if too many secondaries

are demoted to `ASYNCHRONOUS_COMMIT`.

The default value of this setting is 0, which means the primary will never block commits. This is identical to the behavior before SQL Server 2017.

Required? false

Position? named

Default value 0

Accept pipeline input? False

Accept wildcard characters? false

-ReuseSystemDatabases [<SwitchParameter>]

This parameter causes the contained `master` and `msdb` databases from a prior version of the AG to be used in the creation of this new availability group.

> Trying to use this parameter without specifying -ContainedAvailabilityGroup is not allowed would cause the cmdlet to throw an error.

> This parameter is only available in version 22+ of the module.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Script [<SwitchParameter>]

Indicates that this cmdlet returns a Transact-SQL script that performs the task that this cmdlet performs.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-TrustServerCertificate [<SwitchParameter>]

Indicates whether the channel will be encrypted while bypassing walking the certificate chain to validate trust.

In v22 of the module, the default is `$true` (for compatibility with v21). In v23+ of the module, the default value will be '$false', which may create a breaking

change for existing scripts.

> This parameter is new in v22 of the module.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Confirm [<SwitchParameter>]

Prompts you for confirmation before running the cmdlet.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-WhatIf [<SwitchParameter>]

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug,

ErrorAction, ErrorVariable, WarningAction, WarningVariable,

OutBuffer, PipelineVariable, and OutVariable. For more information, see

about_CommonParameters (https:/go.microsoft.com/fwlink/?LinkID=113216).

INPUTS

Microsoft.SqlServer.Management.Smo.Server

You can pass a server instance to this cmdlet.

OUTPUTS

Microsoft.SqlServer.Management.Smo.AvailabilityGroup

This cmdlet returns an availability group.

NOTES

---------- Example 1: Create an Availability Group ----------

PS C:\> $PrimaryServer = Get-Item "SQLSERVER:\SQL\PrimaryServer\Instance22"

PS C:\> $SecondaryServer = Get-Item "SQLSERVER:\SQL\SecondaryServer\Instance22"

PS C:\> $PrimaryReplica = New-SqlAvailabilityReplica -Name "PrimaryServer\Instance22" -EndpointUrl "TCP://PrimaryServer.domain:5022" -FailoverMode "Automatic"

-AvailabilityMode "SynchronousCommit" -AsTemplate -Version ($PrimaryServer.Version)

PS C:\> $SecondaryReplica = New-SqlAvailabilityReplica -Name "SecondaryServer\Instance22" -EndpointUrl "TCP://SecondaryServer.domain:5022" -FailoverMode "Automatic"

-AvailabilityMode "SynchronousCommit" -AsTemplate -Version ($SecondaryServer.Version)

PS C:\> New-SqlAvailabilityGroup -InputObject $PrimaryServer -Name "MainAG" -AvailabilityReplica ($PrimaryReplica, $SecondaryReplica) -Database

@("Database01","Database02")

The first command gets an instance of SQL Server on the primary server, and then stores it in the $PrimaryServer variable.

The fourth command creates a replica that includes the secondary server instance by using New-SqlAvailabilityReplica, and then stores it in the $SecondaryReplica

variable.

The command specifies the version of the server instance by using the Version property of $SecondaryServer.

---- Example 2: Create an Availability Group backed by WSFC ----

Get server

PS C:\> cd 'SQLSERVER:\SQL\some-hostname\Default'

PS SQLSERVER:\SQL\some-hostname\Default> $server = Get-Item $PWD

Create primary replica

PS SQLSERVER:\SQL\some-hostname\Default> $primaryReplica = New-SqlAvailabilityReplica -Name 'some-hostname' -EndpointUrl 'tcp://some-hostname:5022' -FailoverMode

Automatic -AvailabilityMode SynchronousCommit -AsTemplate

Create AG with explicit cluster type WSFC (same as not passing -ClusterType, since it is the default value)

PS SQLSERVER:\SQL\some-hostname\Default> $server | New-SqlAvailabilityGroup -Name 'ag1' -AvailabilityReplica $PrimaryReplica -Script -ClusterType Wsfc

- Example 3: Create an availability groups cluster independent -

Get server

PS C:\> CD 'SQLSERVER:\SQL\some-hostname\Default'

PS SQLSERVER:\SQL\some-hostname\Default> $server = Get-Item $PWD

Create primary replica

PS SQLSERVER:\SQL\some-hostname\Default> $primaryReplica = New-SqlAvailabilityReplica -Name 'some-hostname' -EndpointUrl 'tcp://some-hostname:5022' -FailoverMode

Automatic -AvailabilityMode SynchronousCommit -AsTemplate

Create AG cluster group independent

PS SQLSERVER:\SQL\some-hostname\Default> $server | New-SqlAvailabilityGroup -Name 'ag1' -AvailabilityReplica $PrimaryReplica -Script -ClusterType None

Example 4: Create an availability groups setting the number of SYNCHRONOUS_COMMIT secondaries that must be available for transactions to commit on the primary

Get server

PS C:\> Get-Item 'SQLSERVER:\SQL\some-hostname\Default'

Create primary replica

PS C:\> $primaryReplica = New-SqlAvailabilityReplica -Name 'some-hostname' -EndpointUrl 'tcp://some-hostname:5022' -FailoverMode Automatic -AvailabilityMode

SynchronousCommit -AsTemplate

Create AG with RequiredSynchronizedSecondariesToCommit = 3

PS C:\> $server | New-SqlAvailabilityGroup -Name 'ag1' -AvailabilityReplica $PrimaryReplica -RequiredSynchronizedSecondariesToCommit 3

Note: An exception will be thrown if the -RequiredSynchronizedSecondariesToCommit parameter is used when the target server version is SQL Server 2016 or lower.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/sqlserver/new-sqlavailabilitygroup

Join-SqlAvailabilityGroup

New-SqlAvailabilityReplica

Remove-SqlAvailabilityGroup

Set-SqlAvailabilityGroup

Switch-SqlAvailabilityGroup

Test-SqlAvailabilityGroup