

New-SqlCloneImage

New-SqlCloneImage

Starts creating a new image from either a live dabase or backup.

Syntax

```
New-SqlCloneImage [-Name] <string> [-SqlServerInstance] <SqlServerInstanceResource> [-DatabaseName] <string> [-Destination] <ImageLocationResource> [<CommonParameters>]

New-SqlCloneImage [-Name] <string> [-BackupLocation] <BackupLocationResource> [-BackupFileName] <string[]> [-Destination] <ImageLocationResource> [-BackupPassword <string>] [<CommonParameters>]
```

Description

The New-SqlCloneImage cmdlet starts creating a new image from either a live dabase or backup and outputs details of the creation operation.

This cmdlet returns a OperationResource which can be passed to the Wait-SqlCloneOperation cmdlet.

Connect-SqlClone must be called before this cmdlet.

Parameters

-Name <System.String>

Specifies the name of the image.

Aliases	None
Required?	true
Position?	0
Default Value	None
Accept Pipeline Input	false
Accept Wildcard Characters	false

-SqlServerInstance <RedGate.SqlClone.Client.Api.Objects.SqlServerInstanceResource>

Specifies the SQL Server instance that the source database is on. The Get-SqlCloneSqlServerInstance cmdlet can be used to get a SqlServerInstanceResource.

You can't use this parameter in addition to the BackupLocation and BackupFileName parameters.

Aliases	None
Required?	true
Position?	1
Default Value	None
Accept Pipeline Input	true (ByValue)
Accept Wildcard Characters	false

-DatabaseName <System.String>

Specifies the name of the database to be used as the source for the image.

You can't use this parameter in addition to the BackupLocation and BackupFileName parameters.

Aliases	None
---------	------

Required?	true
Position?	2
Default Value	None
Accept Pipeline Input	false
Accept Wildcard Characters	false

-BackupLocation <RedGate.SqlClone.Client.Api.Objects.BackupLocationResource>

Specifies location of the backup files. The Get-SqlCloneBackupLocation cmdlet can be used to get a BackupLocationResource.

You can't use this parameter in addition to the SqlServerInstance and DatabaseName parameters.

Aliases	None
Required?	true
Position?	1
Default Value	None
Accept Pipeline Input	true (ByValue)
Accept Wildcard Characters	false

-BackupFileName <System.String[]>

Specifies the backup file names relative to the given backup location.

You can't use this parameter in addition to the SqlServerInstance and DatabaseName parameters.

Aliases	None
Required?	true
Position?	2
Default Value	None
Accept Pipeline Input	false
Accept Wildcard Characters	false

-BackupPassword <System.String>

Specifies the password for the backup, if the backup is encrypted.

Aliases	None
Required?	false
Position?	named
Default Value	None
Accept Pipeline Input	false
Accept Wildcard Characters	false

-Destination <RedGate.SqlClone.Client.Api.Objects.ImageLocationResource>

Specifies where the image will be stored. The Get-SqlCloneImageLocation cmdlet can be used to get an ImageLocationResource

Aliases	None
Required?	true
Position?	3
Default Value	None
Accept Pipeline Input	false
Accept Wildcard Characters	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see <http://technet.microsoft.com/en-us/library/hh847884.aspx>.

Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **RedGate.SqlClone.Client.Api.Objects.SqlServerInstanceResource**

Specifies the SQL Server instance that the source database is on. The Get-SqlCloneSqlServerInstance cmdlet can be used to get a SqlServerInstanceResource.

You can't use this parameter in addition to the BackupLocation and BackupFileName parameters.

- **RedGate.SqlClone.Client.Api.Objects.BackupLocationResource**

Specifies location of the backup files. The Get-SqlCloneBackupLocation cmdlet can be used to get a BackupLocationResource.

You can't use this parameter in addition to the SqlServerInstance and DatabaseName parameters.

Return values

The output type is the type of the objects that the cmdlet emits.

- **RedGate.SqlClone.Client.Api.Objects.OperationResource**

Examples

----- EXAMPLE 1 -----

```
Connect-SqlClone -ServerUrl 'http://sql-clone.example.com:14145'
$SqlServerInstance = Get-SqlCloneSqlServerInstance -MachineName WIN201601 -InstanceName SQL2014
$ImageDestination = Get-SqlCloneImageLocation -Path '\\red-gate\data-images'

$ImageOperation = New-SqlCloneImage -Name "AdventureWorks-$(Get-Date -Format yyyyMMddHHmmss)" `
    -SqlServerInstance $SqlServerInstance `
    -DatabaseName 'AdventureWorks' `
    -Destination $ImageDestination

Wait-SqlCloneOperation -Operation $ImageOperation
```

This example creates a new image from the 'AdventureWorks' database on WIN201601\SQL2014 and saves it under '\\red-gate\data-images' with a timestamp postfix.

----- EXAMPLE 2 -----

```
Connect-SqlClone -ServerUrl 'http://sql-clone.example.com:14145'
$BackupLocation = Get-SqlCloneBackupLocation -Path '\\red-gate\backups'
$ImageDestination = Get-SqlCloneImageLocation -Path '\\red-gate\data-images'

$ImageOperation = New-SqlCloneImage -Name "AdventureWorks-$(Get-Date -Format yyyyMMddHHmmss)" `
    -BackupLocation $BackupLocation `
    -BackupFileName @( 'AdventureWorks-201701012210.bak' ) `
    -Destination $ImageDestination

$ImageOperation | Wait-SqlCloneOperation
```

This example creates a new image from the backup file '\\red-gate\backups\AdventureWorks-201701012210.bak' and saves it under '\\red-gate\data-images' with a timestamp postfix.

----- EXAMPLE 3 -----

```
Connect-SqlClone -ServerUrl 'http://sql-clone.example.com:14145'
$BackupLocation = Get-SqlCloneBackupLocation -Path '\\red-gate\backups'
$ImageDestination = Get-SqlCloneImageLocation -Path '\\red-gate\data-images'

$ImageOperation = New-SqlCloneImage -Name "Production-$(Get-Date -Format yyyyMMddHHmmss)" `
    -BackupLocation $BackupLocation `
    -BackupFileName @('Production-201703201042.sqb') `
    -BackupPassword 'PasswordForSQLBackup' `
    -Destination $ImageDestination

$ImageOperation | Wait-SqlCloneOperation
```

This example creates a new image from the encrypted SQL Backup file '\\red-gate\backups\Production-201703201042.sqb' and saves it under '\\red-gate\data-images' with a timestamp postfix.