

New-DatabaseDocumentation

New-DatabaseDocumentation

Generates documentation for the validated database schema produced by the Invoke-DlmDatabaseValidation cmdlet.

Syntax

```
New-DatabaseDocumentation [-InputObject] <IProject> [-TemporaryDatabaseServer <DatabaseServerConnection>] [-FilterPath <string>] [-IgnoreParserErrors] [-SQLCompareOptions <string>] [-SQLDataCompareOptions <string>] [<CommonParameters>]

New-DatabaseDocumentation [-InputObject] <IProject> [-TemporaryDatabase <DatabaseConnection>] [-FilterPath <string>] [-IgnoreParserErrors] [-SQLCompareOptions <string>] [-SQLDataCompareOptions <string>] [<CommonParameters>]
```

Description

The New-DatabaseDocumentation cmdlet takes the output of the Invoke-DatabaseBuild cmdlet and creates documentation for that schema. It creates a SchemaDocumentation object that represents the documentation for a validated project.

You can use the New-DatabaseBuildArtifact cmdlet to include this documentation in the database build artifact it creates.

In order to generate the documentation, the cmdlet creates a temporary copy of the database. By default, it uses LocalDB for this. Alternatively, you can use the TemporaryDatabaseServer parameter to specify a SQL Server instance for the temporary database. This is useful if your database uses features that aren't supported by LocalDB, such as Full-Text Search.

If you don't want to use LocalDB but don't have permission to create a database on the SQL Server instance, you can use the TemporaryDatabase parameter to specify an existing database.

Parameters

-InputObject <RedGate.Versioning.Automation.Compare.Domain.Projects.IProject>

The output object of Invoke-DatabaseBuild cmdlet, which represents the validated database schema.

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

-TemporaryDatabaseServer <RedGate.Versioning.Automation.Compare.SchemaSources.DatabaseServerConnection>

The connection string for the temporary database server used for documentation. For example, 'Data Source=TempServer01'.

By default, LocalDB is used for the temporary database. However there may be some features in your database that aren't supported by LocalDB (for example, Full-Text Search). In this case, or if LocalDB is not present, use this parameter to specify an alternative SQL Server instance for the temporary database.

Using this option, SQL Change Automation will create a temporary, randomly-named database on the specified SQL Server instance.

You can't use this parameter in addition to the TemporaryDatabase parameter.

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

Accept Wildcard Characters	false
----------------------------	-------

-TemporaryDatabase <RedGate.Versioning.Automation.Compare.SchemaSources.DatabaseConnection>

The details of the temporary database used for documentation. This can be:

- a database connection object that contains connection details for a database. See New-DatabaseConnection for details.
- a database connection string. For example, 'Data Source=TempServer01;Initial Catalog=TempDatabase01'.

By default, LocalDB is used for the temporary database. If you don't want to use LocalDB and don't have permission to create a database on the SQL Server instance, use this option to specify an existing database to use for the temporary copy of the database.

If you use this parameter, all existing data on the temporary database will be lost.

You can't use this parameter in addition to the TemporaryDatabaseServer parameter.

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

-FilterPath <System.String>

The path to a .scpf filter file.

Overrides any Filter.scpf file present in the InputObject schema with an alternative filter file to be used when documenting this schema.

This parameter will be ignored if the value specified is either \$null or empty.

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

-IgnoreParserErrors <System.Management.Automation.SwitchParameter>

Whether the SQL Compare engine should ignore parser errors.

This parameter will be ignored if the InputObject is a database connection.

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

-SQLCompareOptions <System.String>

Specifies the SQL Compare options to use when creating the script for validation. A default set of options is listed below. To include additional options, specify a comma-delimited list of the option names (eg 'IgnoreComments, ObjectExistenceChecks'). To turn off a default option, precede the option name with a minus sign (eg '-ForceColumnOrder').

This parameter will be ignored if the value specified is \$null or empty.

By default, the following Compare options are used:

- ConsiderNextFilegroupInPartitionSchemes
- DecryptPost2KEncryptedObjects
- DoNotOutputCommentHeader
- ForceColumnOrder
- IgnoreCertificatesAndCryptoKeys
- IgnoreDatabaseAndServerName
- IgnoreUsersPermissionsAndRoleMemberships
- IgnoreUserProperties
- IgnoreWhiteSpace
- IgnoreWithElementOrder
- IncludeDependencies
- ThrowOnFileParseFailed
- UseCompatibilityLevel

For more information about SQL Compare options, see <http://www.red-gate.com/sca/ps/help/compareoptions>.

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

-SQLDataCompareOptions <System.String>

Specifies the SQL Data Compare options to use when creating the script for validation. To include additional options, specify a comma-delimited list of the option names (eg 'DisableAndReenableDDLTriggers, CompressTemporaryFiles').

Aliases	None
Required?	false
Position?	named
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.Versioning.Automation.Compare.Domain.Projects.IProject**

The output object of Invoke-DatabaseBuild cmdlet, which represents the validated database schema.

Return values

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

- **RedGate.Versioning.Automation.Compare.Documentation.SchemaDocumentation**

Examples

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

```
$project = "C:\Work\project\project.sqlproj"
$validatedProject = $project | Invoke-DatabaseBuild
$documentation = $validatedProject | New-DatabaseDocumentation
$buildArtifact = $validatedProject | New-DatabaseBuildArtifact -PackageId MyDatabase -PackageVersion 1.0.0 -
Documentation $documentation
```

This example shows how to create database documentation for a validated SQL Change Automation project using the New-DatabaseDocumentation cmdlet. The New-DatabaseBuildArtifact cmdlet includes this documentation in the database build artifact it creates.

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

```
$scriptsFolder = "C:\Work\scripts"
$validatedProject = $scriptsFolder | Invoke-DatabaseBuild
$documentation = $validatedProject | New-DatabaseDocumentation
$buildArtifact = $validatedProject | New-DatabaseBuildArtifact -PackageId MyDatabase -PackageVersion 1.0.0 -
Documentation $documentation
```

This example shows how to create database documentation for a validated SQL Source Control project using the New-DatabaseDocumentation cmdlet. The New-DatabaseBuildArtifact cmdlet includes this documentation in the database build artifact it creates.