

New-DatabaseBuildArtifact

New-DatabaseBuildArtifact

Creates a build artifact from the validated project produced by the Invoke-DatabaseBuild cmdlet.

Syntax

```
New-DatabaseBuildArtifact [-InputObject] <IProject> -PackageId <string> -PackageVersion <string> [-PackageDescription <string>] [-Documentation <SchemaDocumentation>] [<CommonParameters>]
```

Description

The New-DatabaseBuildArtifact cmdlet takes a validated project from the output of the Invoke-DatabaseBuild cmdlet and creates a build artifact. It can also include the documentation for your database using the output of the New-DatabaseDocumentation cmdlet.

You can use the Export-DatabaseBuildArtifact cmdlet to export the build artifact.

Parameters

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

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

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

-PackageId <System.String>

The unique identifier for the package you are creating.

This mustn't contain more than 100 characters and must be a valid NuGet package identifier.

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

-PackageVersion <System.String>

The version of the package. This must be in a valid NuGet version format (eg 3.2.1, 3.2.1-beta1).

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

-PackageDescription <System.String>

The description of the package you are creating.

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

-Documentation <RedGate.Versioning.Automation.Compare.Documentation.SchemaDocumentation>

The documentation object created by New-DatabaseDocumentation to include in the database package.

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 project.

Return values

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

- **RedGate.Versioning.Automation.Compare.Domain.BuildArtifacts.IBuildArtifact**

Examples

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

```
$project = "C:\Work\project\project.sqlproj"
$validatedProject = $project | Invoke-DatabaseBuild
$buildArtifact = $validatedProject | New-DatabaseBuildArtifact -PackageId MyDatabase -PackageVersion 1.0.0
$buildArtifact | Export-DatabaseBuildArtifact -Path "C:\Export\buildArtifacts"
```

This example shows how to use the New-DatabaseBuildArtifact cmdlet to create a build artifact from a validated SQL Change Automation project.

First the Invoke-DatabaseBuild cmdlet validates the project. Then the New-DatabaseBuildArtifact cmdlet creates the build artifact. Finally the Export-DatabaseBuildArtifact cmdlet exports the build artifact to file.

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

```
$project = "C:\Work\scripts"
$validatedProject = $project | Invoke-DatabaseBuild
$buildArtifact = $validatedProject | New-DatabaseBuildArtifact -PackageId MyDatabase -PackageVersion 1.0.0
$buildArtifact | Export-DatabaseBuildArtifact -Path "C:\Export\buildArtifacts"
```

This example shows how to use the New-DatabaseBuildArtifact cmdlet to create a NuGet package from a validated SQL Source Control project.

First the Invoke-DatabaseBuild cmdlet validates the database schema. Then the New-DatabaseBuildArtifact cmdlet creates the build artifact. Finally the Export-DatabaseBuildArtifact cmdlet exports the build artifact to file.

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

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

This example shows how to use the New-DatabaseBuildArtifact cmdlet to create a NuGet build artifact that also includes database documentation.

The documentation is created by the New-DatabaseDocumentation cmdlet.