

# Restore a SQL Database Using PowerShell

---

Warning: this code is provided on a best effort basis and is not in any way officially supported or sanctioned by Cohesity. The code is intentionally kept simple to retain value as example code. The code in this repository is provided as-is and the author accepts no liability for damages resulting from its use.

This script demonstrates how to perform an restore of a SQL database. The script can restore the database to the original server, or a different server. It can overwrite the existing database or restore with a different database name.

## Warning!!

---

This script can overwrite production data if you ask it to. Make sure you know what you are doing and test thoroughly before using in production. The authors accept no liability for damages!!!

## Components

---

- restore-SQL.ps1: the main powershell script
- cohesity-api.ps1: the Cohesity REST API helper module

Place both files in a folder together and run the main script like so:

```
./restore-SQL.ps1 -vip mycluster -username admin -sourceServer sql2012 -sourceDB  
cohesitydb -targetServer sqldev01 -targetDB restoreTest -mdfFolder c:\sqldata -  
ndfFolder c:\sqldata\ndf -ldfFolder c:\sqldata\logs  
Connected!  
Restoring cohesitydb to sqldev01 as restoreTest
```

Note: if you want to restore multiple ndf files to separate locations, use the following parameter:

```
-ndfFolders @{ '*1.ndf'='E:\sqlrestore\ndf1'; '*2.ndf'='E:\sqlrestore\ndf2' }
```

If you want to replay logs to a point in time, use this parameter:

```
-logTime '2019-01-20 23:47:02'
```