

The background of the slide is a solid orange color with a subtle network diagram pattern of white lines and dots. In the bottom right corner, there is a large, faint, stylized logo consisting of a circle and a chevron shape.

Creating a Rock Sandbox

How to take a snapshot of a production environment to create a sandbox for testing and training.

Creating/Updating a Rock Development Environment

Often is helpful to have a Rock sandbox or development environment that is regularly updated from production to have a copy of the most recent code and data from production. When doing so, it is important to make sure that the development environment is safe-guarded from sending communications to actual production recipients. The following steps can be followed to create or update a development environment.

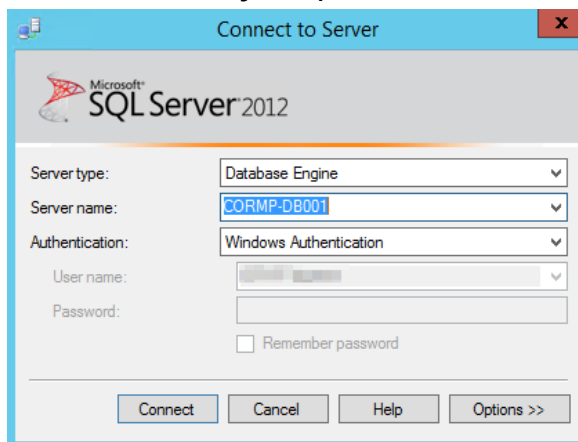
Backup the Current Production Database

A "copy-only" backup of the current Rock production database is needed so that it can be copied to, and restored on a development server. Depending on your production database size and server configuration, it is possible that the backup may take several hours to complete. Make sure to allow enough lead time for this.

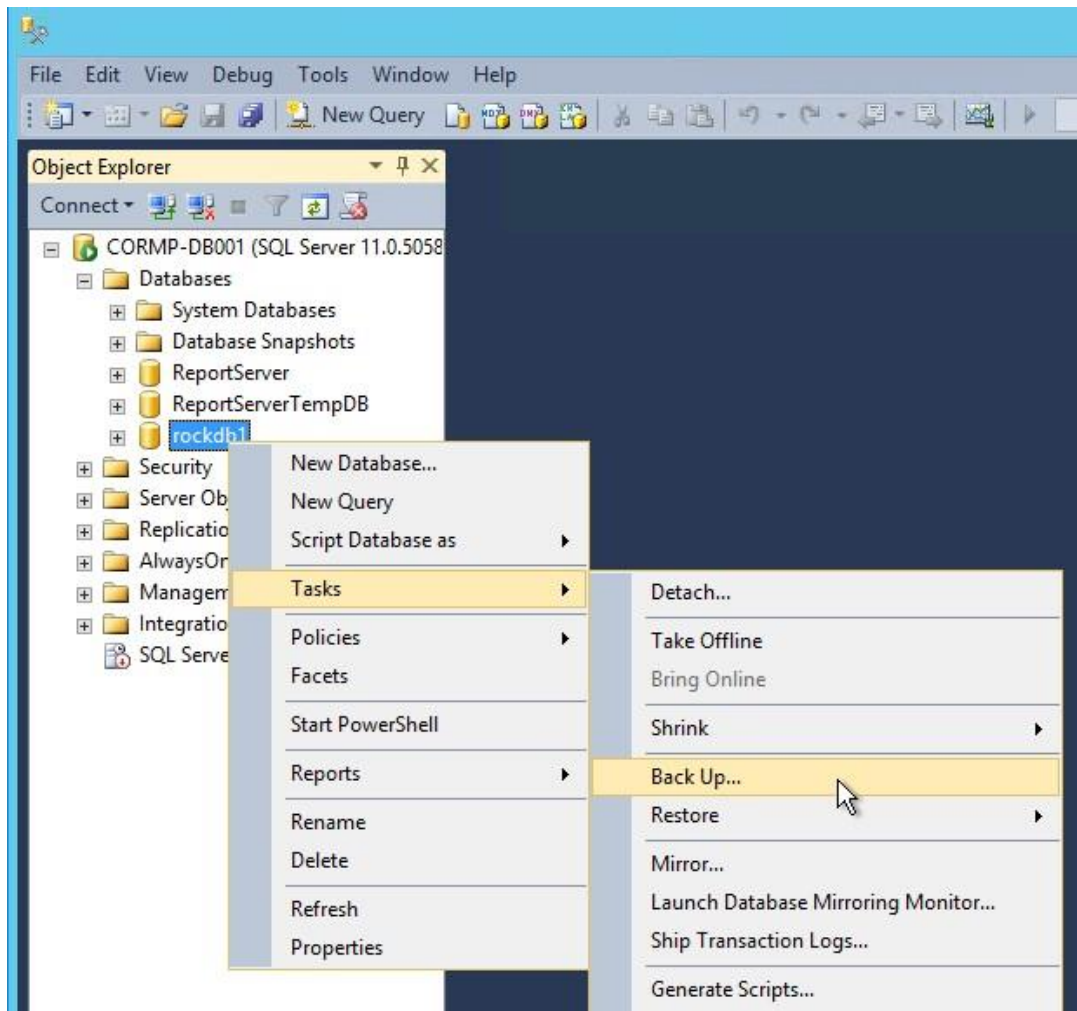
1. Connect to the production SQL Server and start the SQL Server Management Studio (SSMS) application,



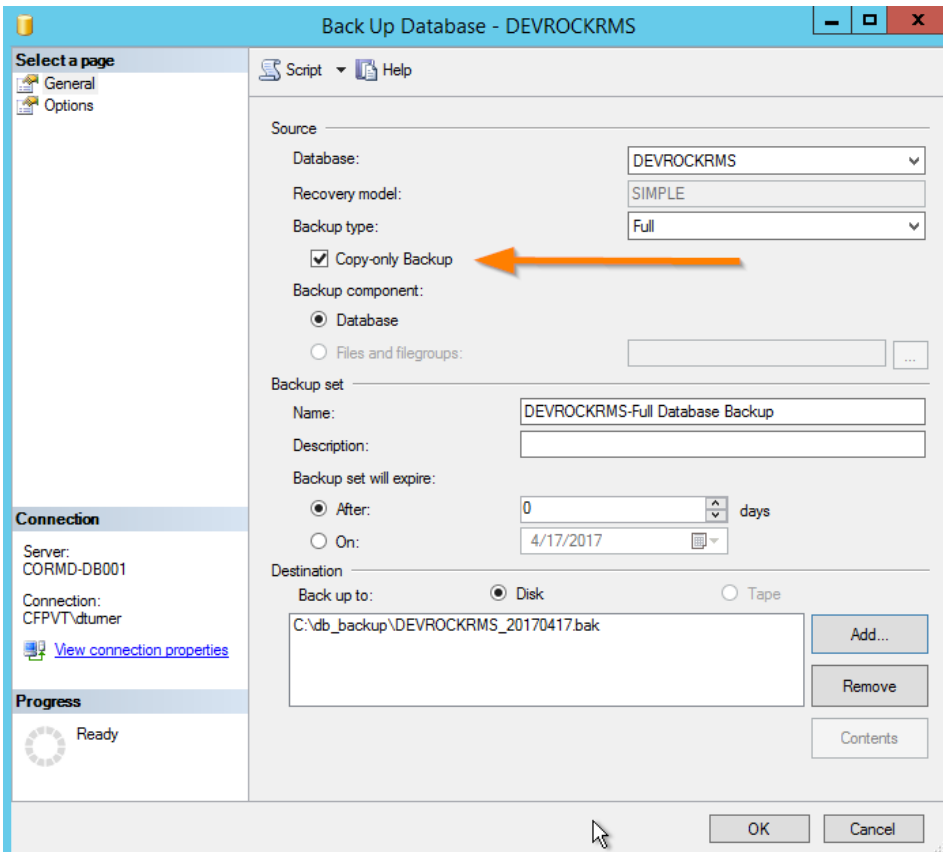
and connect to your production server



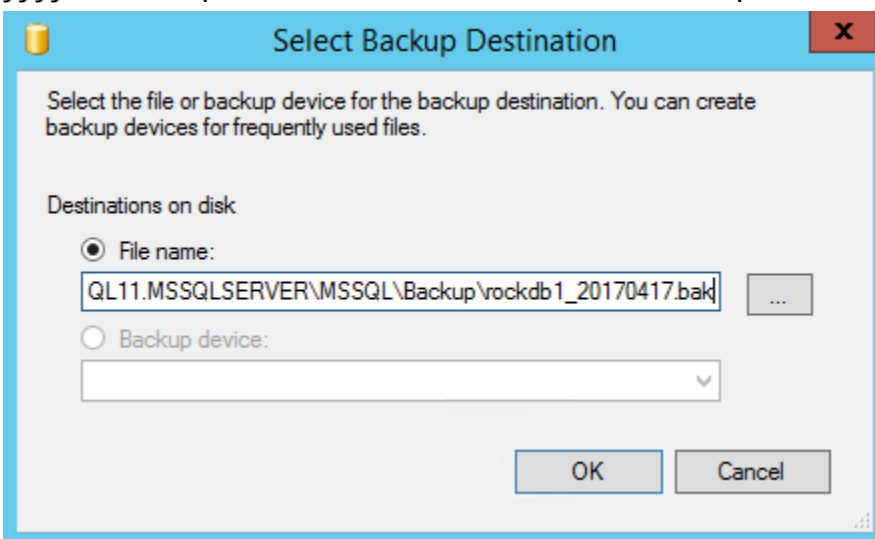
2. Using the SSMS object explorer, expand the Databases folder and right-click on your production Rock database. Use the menu to select *Tasks > Back Up...*



3. From the Back Up Database dialog, select the 'Copy-only Backup' checkbox.



4. In the Destination area, make sure the *Disk* radio button is selected. If a location is already listed in the Destination box, select it, then click *Remove* to remove it.
5. Click the *Add* button to select a location for the backup file. The filename should use a pattern like: *database_yyyymmdd.bak* where *yyymmdd* represents the current date. For example:



6. Once you've entered the destination file, click *OK* on the destination dialog, and then click *OK* on the *Back Up Database* window.

7. Wait for the backup to complete. This may take a while, but you should see the progress indicator continue to increase and a message will appear when the backup has been completed.

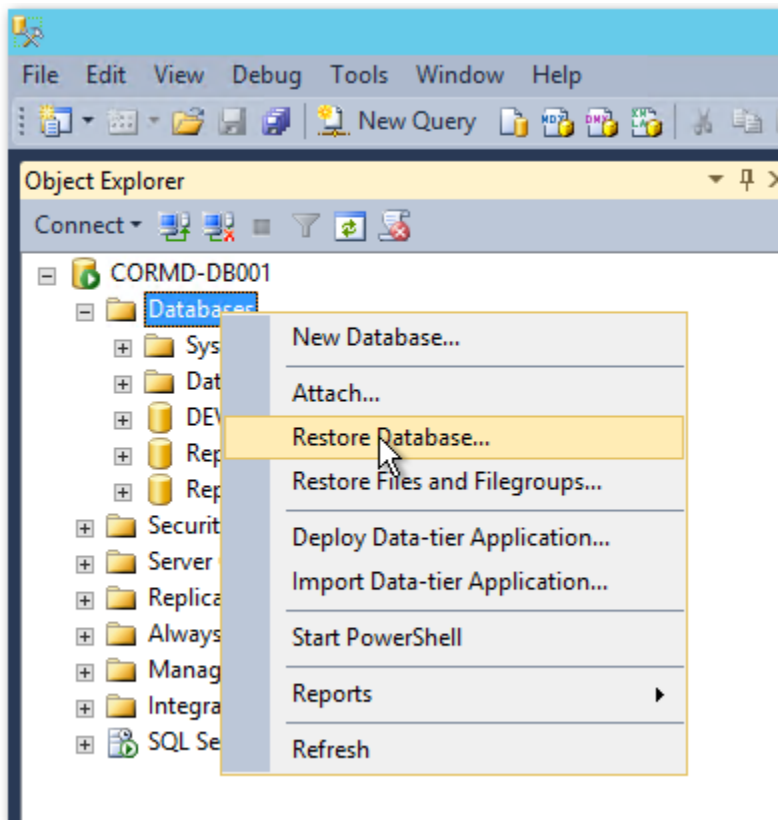
Copy Production Database to Development Server

Now that the backup has been completed, it should be copied to the development server.

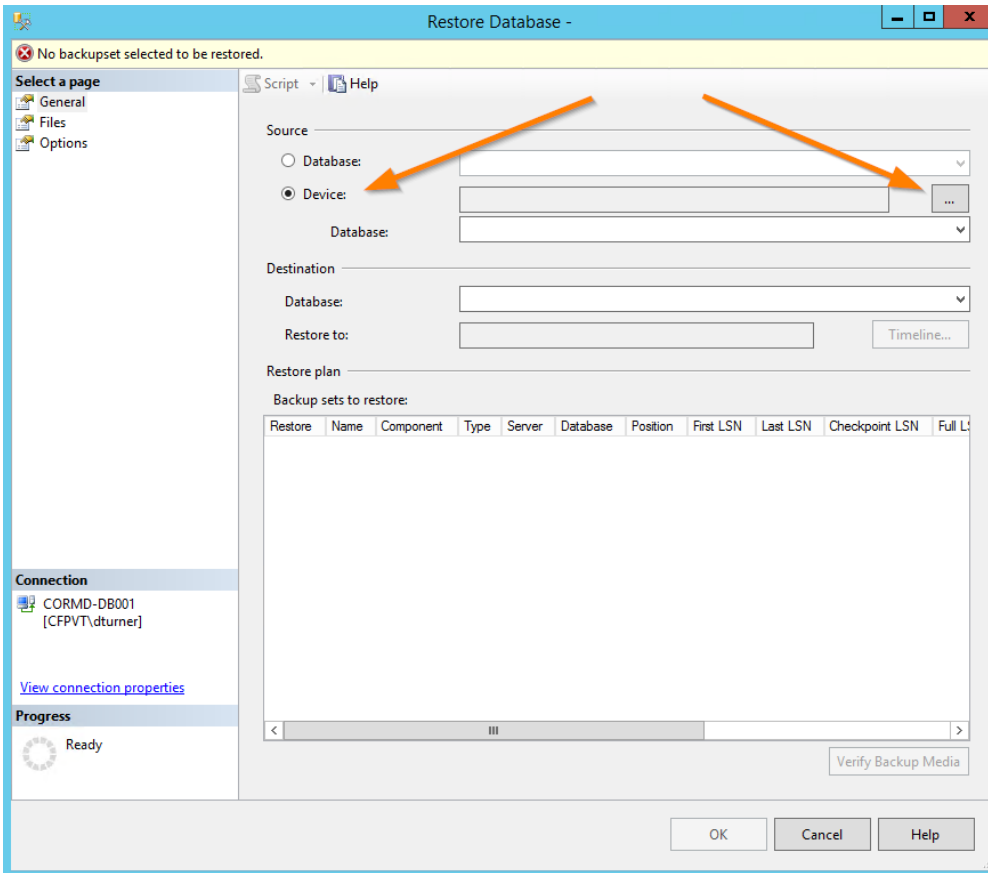
Restore Database to Development Server

After the production database has been copied to the development server, it then needs to be restored.

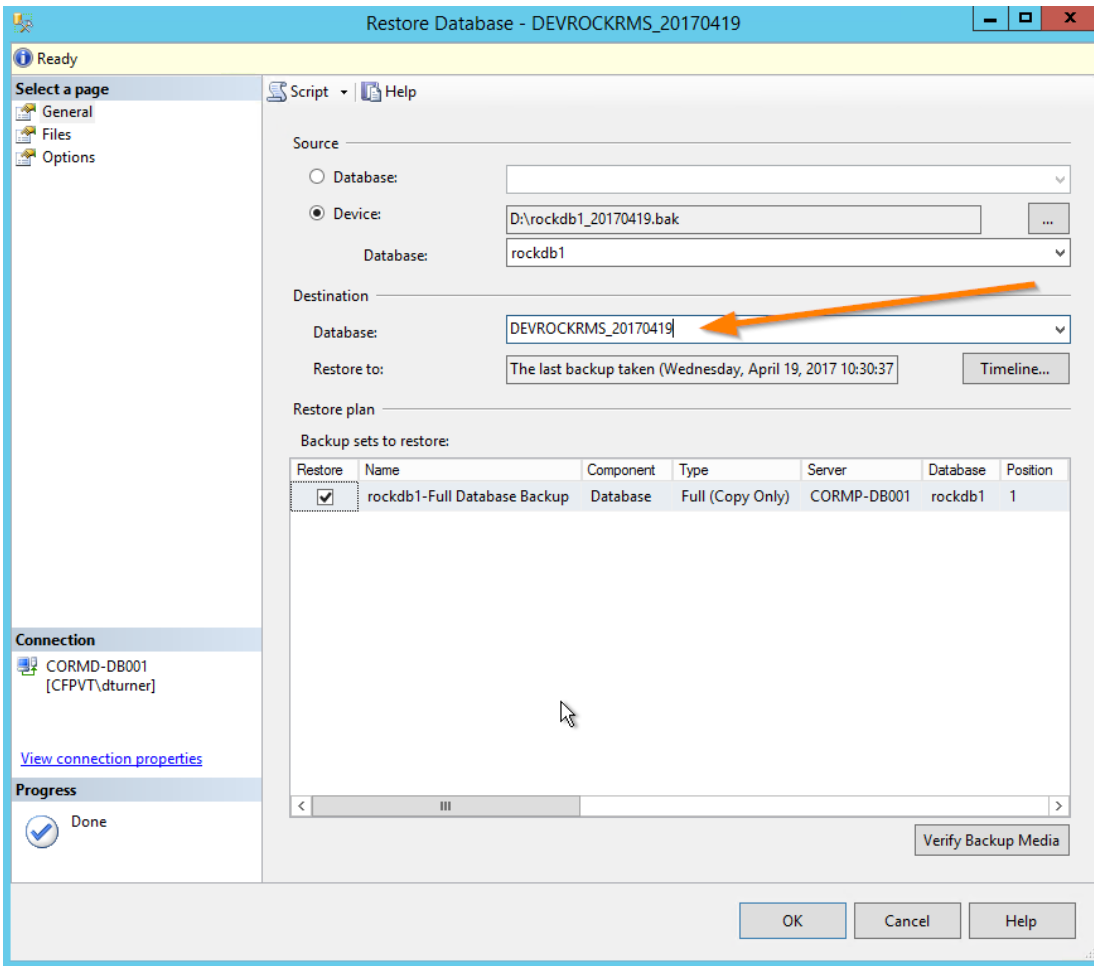
1. Connect to the development server and open the SSMS application. Right-Click the *Database* node, and select the *Restore Database* option.



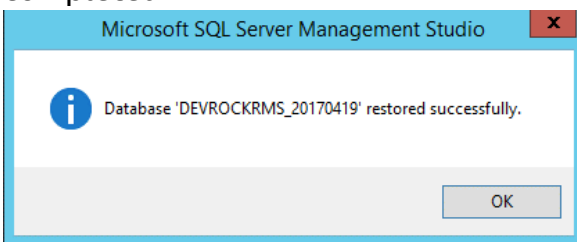
2. In the *Restore Database* dialog, click the *Device* radio button under *Source*.
3. Then click the button with three dots to the right of *Device* to select the backup file.



4. Navigate to the location of the production backup that was copied from the production server.
5. After selecting the backup file, change the name of the Database that will be created. Use a pattern like *DEVROCKRMS_yyyymmdd* where *yyymmdd* represents the current date.



6. After selecting the backup and changing the database name, click *OK* to start the restore. The progress of the restore will be displayed in the upper-right corner of the dialog and a message will indicate when the restore has completed.



Run Post-Deploy Script

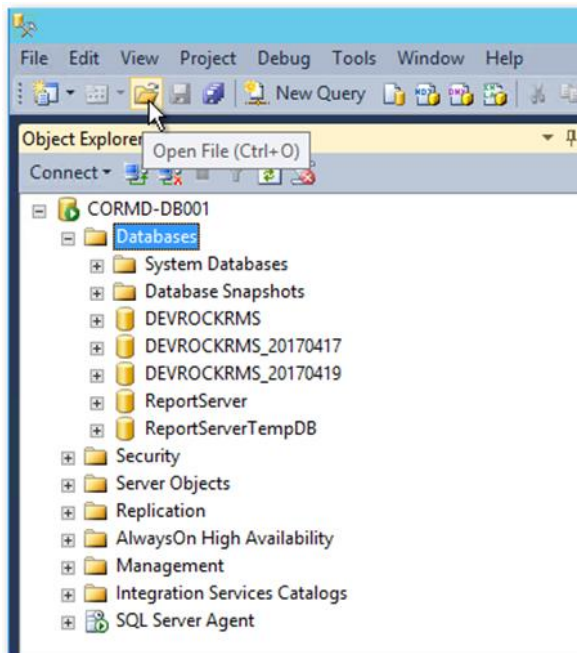
After the production database has been restored to development, it needs to be updated to work in a development environment. This requires the following changes...

- Updating all pages/sites to not require SSL since a certificate may not exist in development environment.

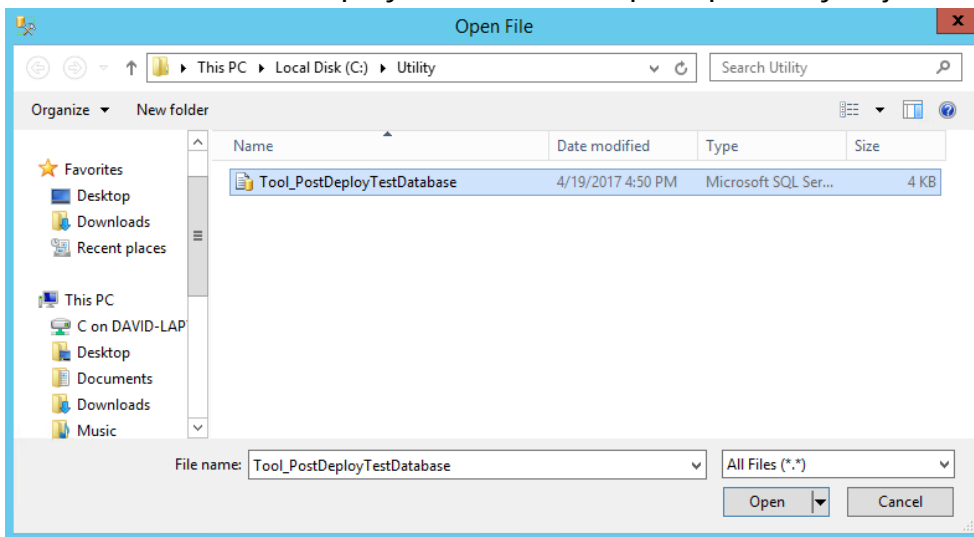
- Updating every person's email address to have a @safety.netz domain (which is invalid) so that they do not inadvertently receive any notifications from the development environment.
- Updating all jobs to be inactive and changing the SMTP Transport settings to further prevent notifications from being sent.
- Adding html to each page so that it is obvious to user that they are on a development environment.

Fortunately, there is a SQL script that can be used to make these updates.

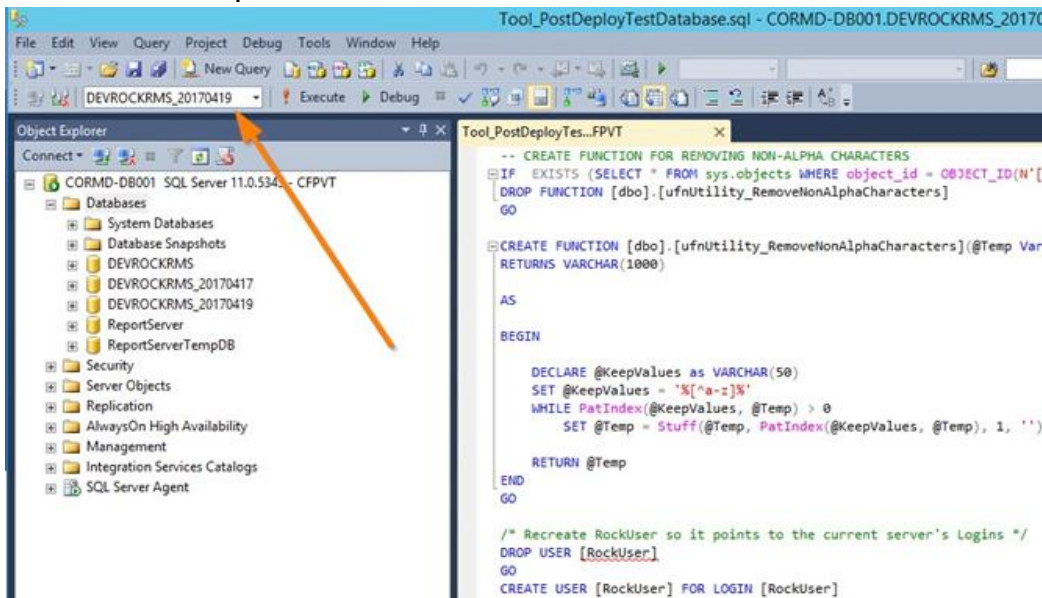
1. Copy the following script to the development server:
https://raw.githubusercontent.com/SparkDevNetwork/Rock/develop/DevTools/Sql/Tool_PostDeployTestDatabase.sql
2. On the develop SQL Server, using SSMS, select the Open File option from the toolbar.



3. Select the Tool_PostDeployTestDatabase.sql script that you just created.



4. After this script has been opened, make sure to select the database that you restored in the previous section.



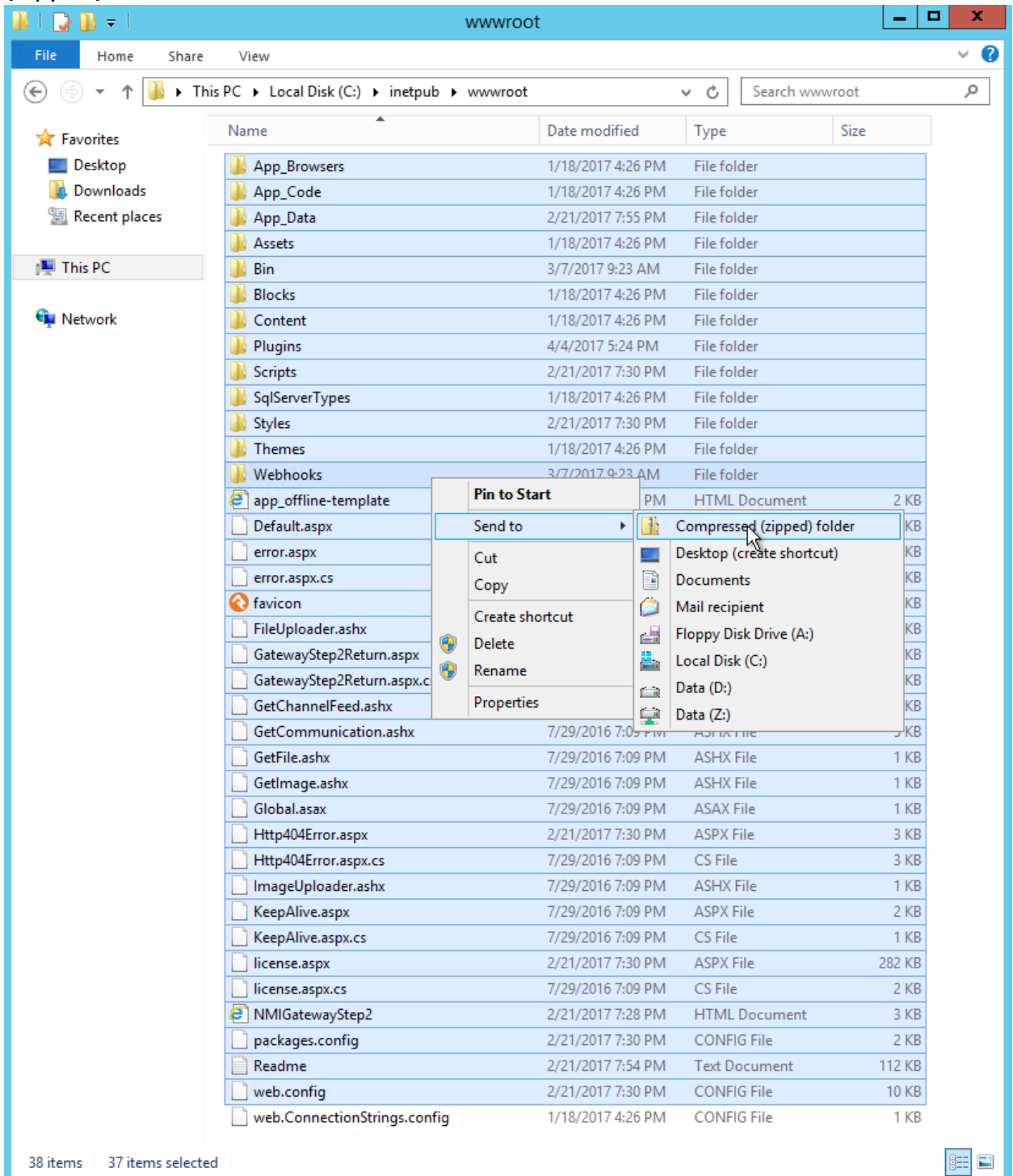
5. After opening the script, and selecting the correct database, click the *Execute* option from the toolbar to run the script.

Take Copy of Production Website

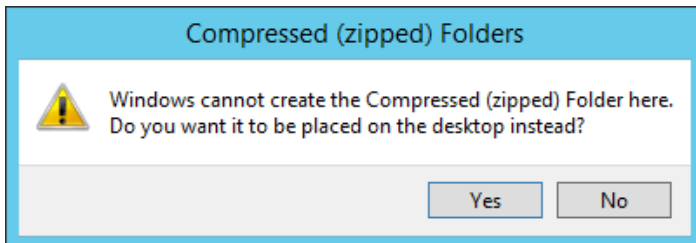
Now that there is an updated database, the Rock source code (website) should also be updated. The first step is to take a copy of the production source code.

1. Connection the production web server and navigate to the `C:\inetpub\wwwroot` folder (This is the typical location of the Rock website, but your installation may be in a different folder).
2. Select all of the folders and files except for the `web.ConnectionStrings.config` file.

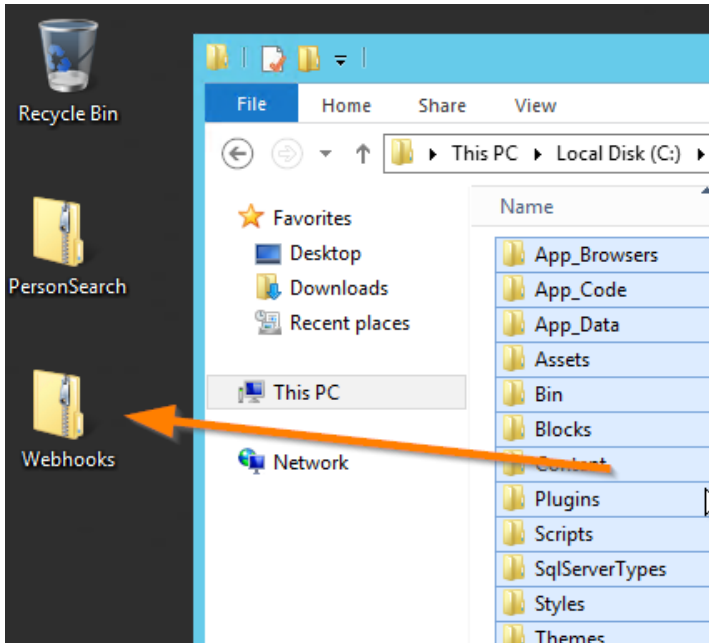
3. Right-Click the selected files and select the option to Send to a Compressed (zipped) folder.



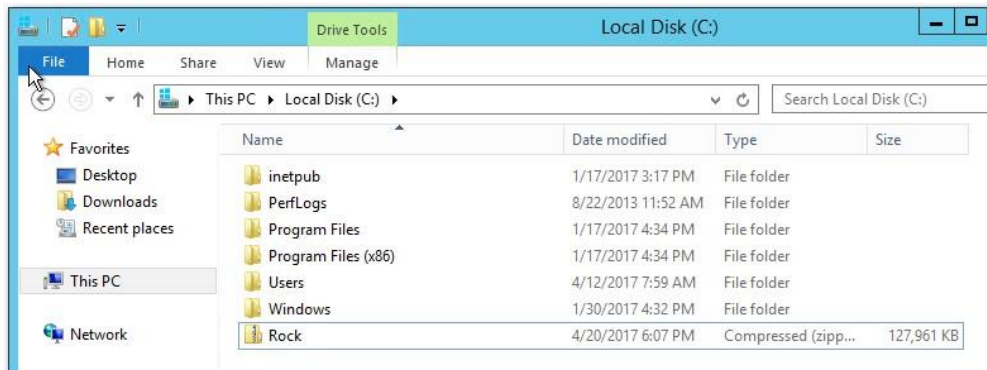
You may get a warning about not being able to create the file in the current location.



If so, click Yes. This will create a new zip file (either in the current directory or on the desktop) and it will have a name that isn't obvious.



4. Copy this file from the production server to the development server. You may want to rename the zip file to *Rock.zip* as well.



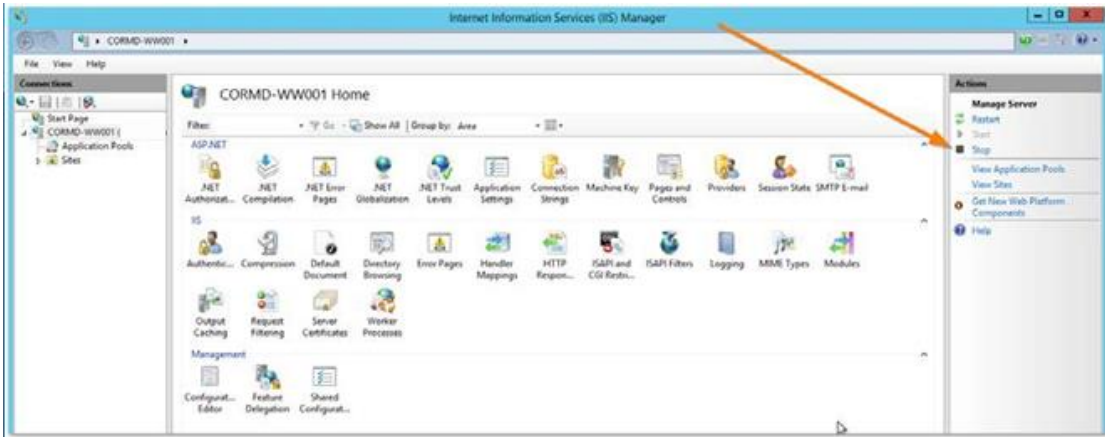
Update Development Website

The development server now needs to be updated with files from production

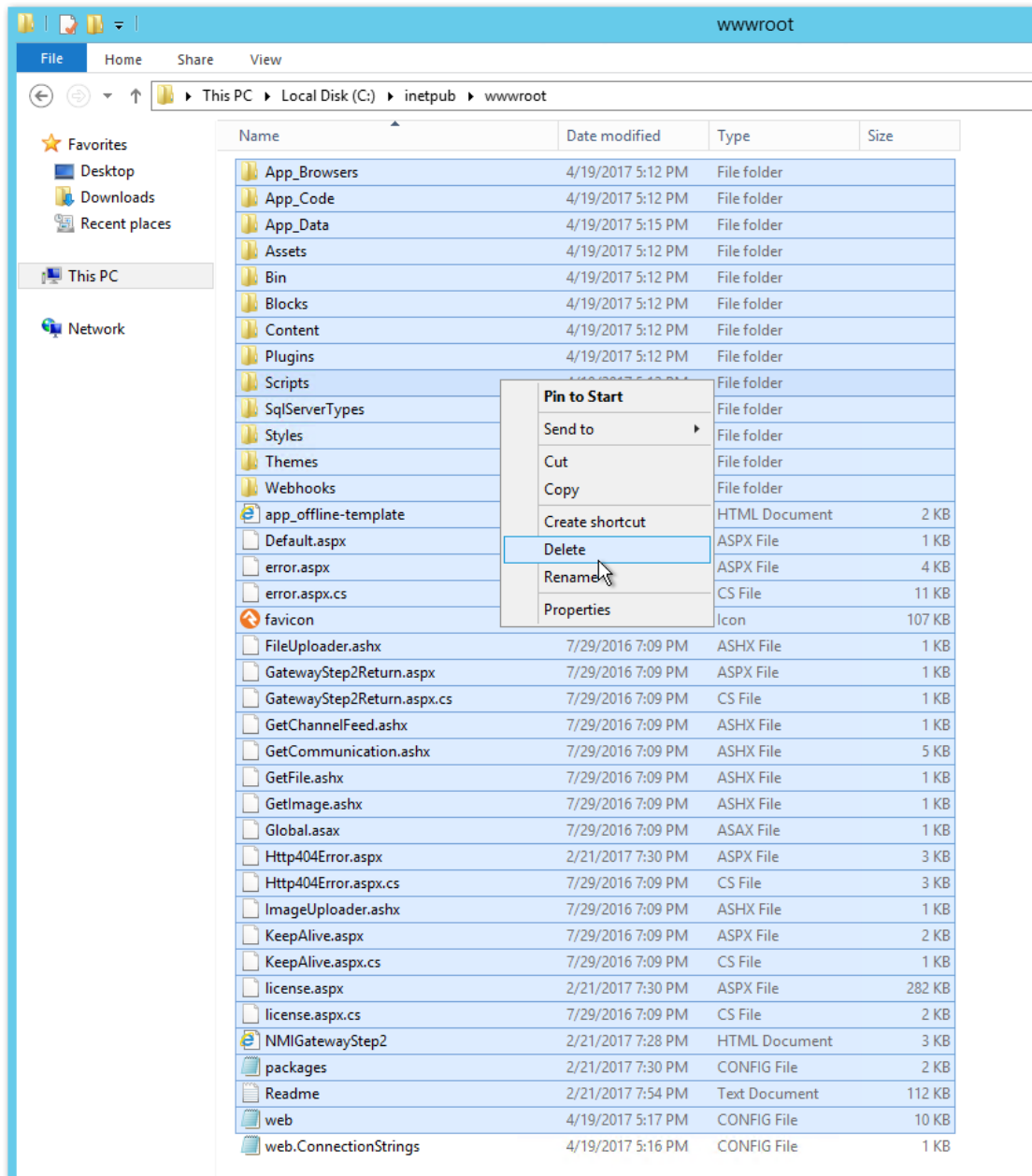
1. Connect to the Development web server and run the *Internet Information Services (IIS)* application on the development server.



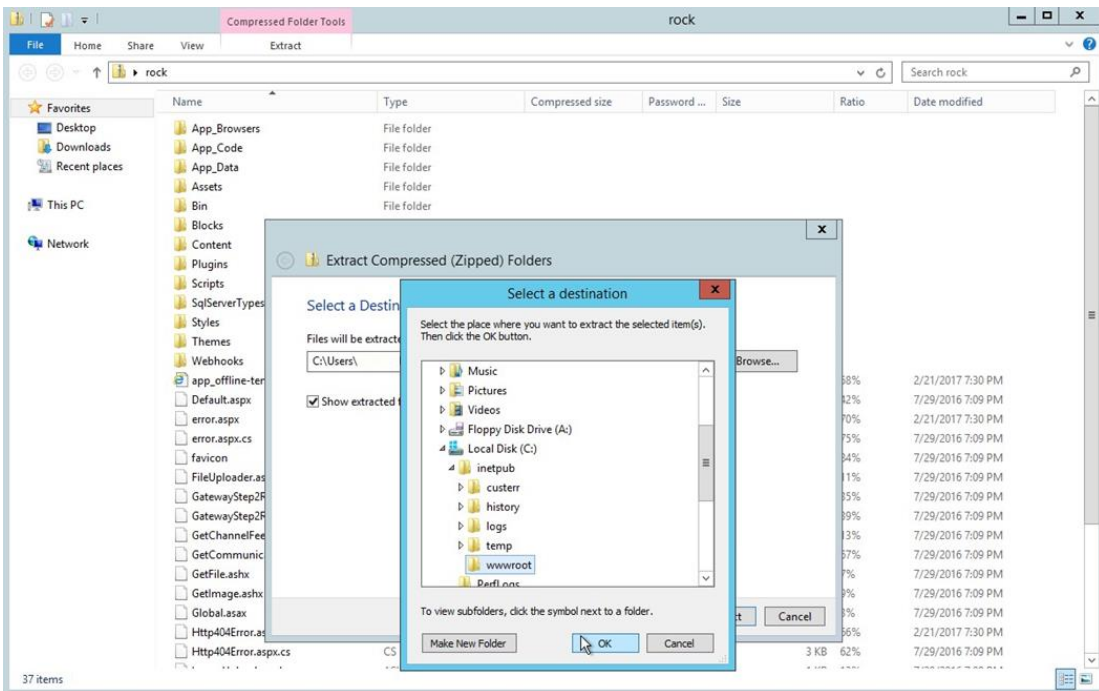
2. From IIS select the option to stop the server



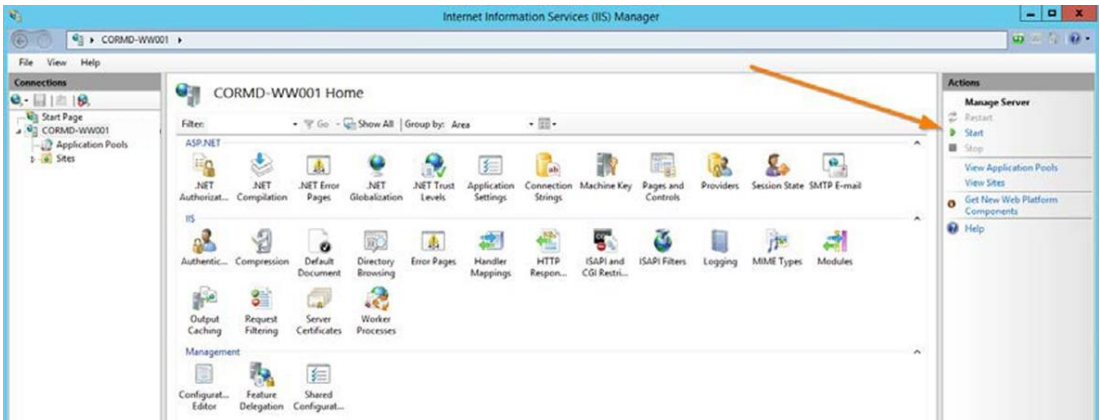
3. Switch to File Explorer and delete all of the files and folders except the *web.ConnectionStrings.config* file on the *C:\inetpub\wwwroot* folder (the same files you added to a zip file in step #4).



4. After the files have been deleted, double-click the *Rock.Zip* file that you copied from production, and extract all its files to the *C:\inetpub\wwwroot* folder.



5. Once the files have been extracted, open the destination folder (*C:\inetpub\wwwroot*).
6. Edit the *web.ConnectionStrings.config* file. If you do not have a default editor configured for these types of files, you can use Notepad to edit the file.
7. Update the Initial Catalog value to be the name of the new database that was restored from production. Save your changes.
8. Switch back to IIS and restart the web server.



9. After IIS has restarted the development web server, open a browser and navigate to the develop instance of Rock.