

# Plex Server 이전 설치 (Library DB 이전)

Plex Library를 이전할 경우가 종종 있습니다.

Library양이 적다면 신규 생성시 별 문제가 되지 않지만 방대한 양의 자료를 가지고 있고,

정성스럽게 메타데이터를 관리하신 분들은 예전 자료를 그대로 백업 받고 새로운 장소에 그대로 셋팅을

원하실 경우가 있습니다.

이럴 경우 필요한 자료를 백업하고 이동하는 방법을 성명 드립니다.

제일 먼저 Plex 서비스를 중단하고 필요한 파일을 백업 합니다.

일단 백업을 해야하는 자료는 아래의 3개 입니다. (아래의 모든 경로는 Docker가 아닌 공식홈페이지에서 제공한 spk를 설치했을 경우 입니다. Docker를 사용하신다면 사용하신 설정에 맞게 경로를 바꾸셔야 합니다.)

- ✓ 1. Plex경로/Application Support/Plex Media Server/Media  
: 아트웍, 자막, 썸네일이 들어있는 디렉토리 입니다.

- ✓ 2. Plex경로/Application Support/Plex Media Server/Metadata  
: 메타데이터가 들어있는 디렉토리 입니다.

- 3. Plex경로/Application Support/Plex Media Server/Plug-in Support/Databases 중
  - ✓ 1) com.plexapp.plugins.library.db : 라이브러리의 위치, 각 자료의 저장위치와 위 1, 2번 자료와 매핑할 수 있는 데이터베이스 입니다.
  - 2) com.plexapp.plugins.library.blobs.db : 바이너리 데이터베이스 입니다.
  - 3) 나머지 확장자가 날짜로 되어 있는 db파일은 백업 파일로 필요 없는 파일 입니다.

3번의 경우 경로에서 파일 두개를 복사하여 백업을 하면 되는데 1~2번의 경우 파일의 갯수가 많아 복사나 이동이 어려울 수 있습니다.

이 경우 압축을 하고 다른 곳으로 이동 후 압축 해제하는 것을 추천 드립니다. 압축 해제 후 신규 Plex의 동일 위치에 파일을 복사하시면 됩니다.

- 1. 압축 명령 (경로에 공백이 있을 경우 ""로 묶으셔야 합니다.)

[DSM 7.x]

- 1) Databases 압축

```
tar -zcvf /volume2/Temp_Vol2/Databases.tar.gz "/volume1/PlexMediaServer/AppData/Plex Media Server/Plug-in Support/Databases"
```

## 2) Media 압축

```
tar -zcvf /volume2/Temp_Vol2/Media.tar.gz "/volume1/PlexMediaServer/AppData/Plex Media Server/Media"
```

## 3) Metadata 압축

```
tar -zcvf /volume2/Temp_Vol2/Metadata.tar.gz "/volume1/PlexMediaServer/AppData/Plex Media Server/Metadata"
```

[DSM 6.x]

## 1) Databases 압축

```
tar -zcvf /volume1/Media/Databases.tar.gz "/volume1/Plex/Library/Application Support/Plex Media Server/Plug-in Support/Databases"
```

## 2) Media 압축

```
tar -zcvf /volume1/Media/Media.tar.gz "/volume1/Plex/Library/Application Support/Plex Media Server/Media"
```

## 3) Metadata 압축

```
tar -zcvf /volume1/Media/Metadata.tar.gz "/volume1/Plex/Library/Application Support/Plex Media Server/Metadata"
```

## 2. 압축해제 ("-C"를 넣으면 다른 디렉토리에 압축을 해제 할수 있습니다. 없으면 동일 경로에 해제 됩니다.)

```
tar -xvf 압축파일명 -C 해제할 경로
```

이제 Database를 수정해야 합니다.

DB는 아래 "DB Browser for SQLite" (<https://sqlitebrowser.org/>)라는 프로그램을 다운받아 수정 가능합니다.

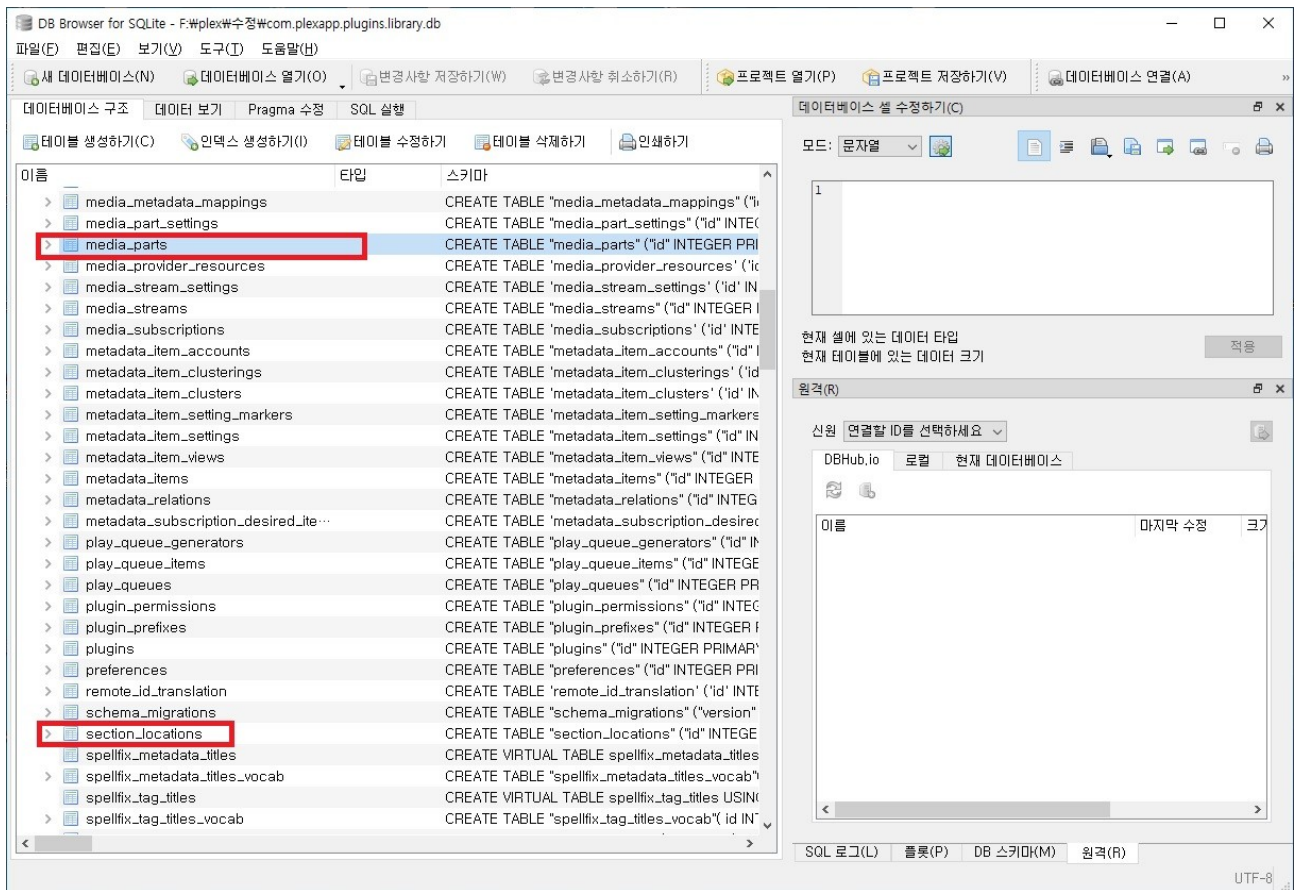
다운받은 후 열기를 통해 "com.plexapp.plugins.library.db" 파일을 열어 줍니다.

여기서 수정을 해야하는 DB table은 **"section\_locations"**와 **"media\_parts"** 테이블 입니다.

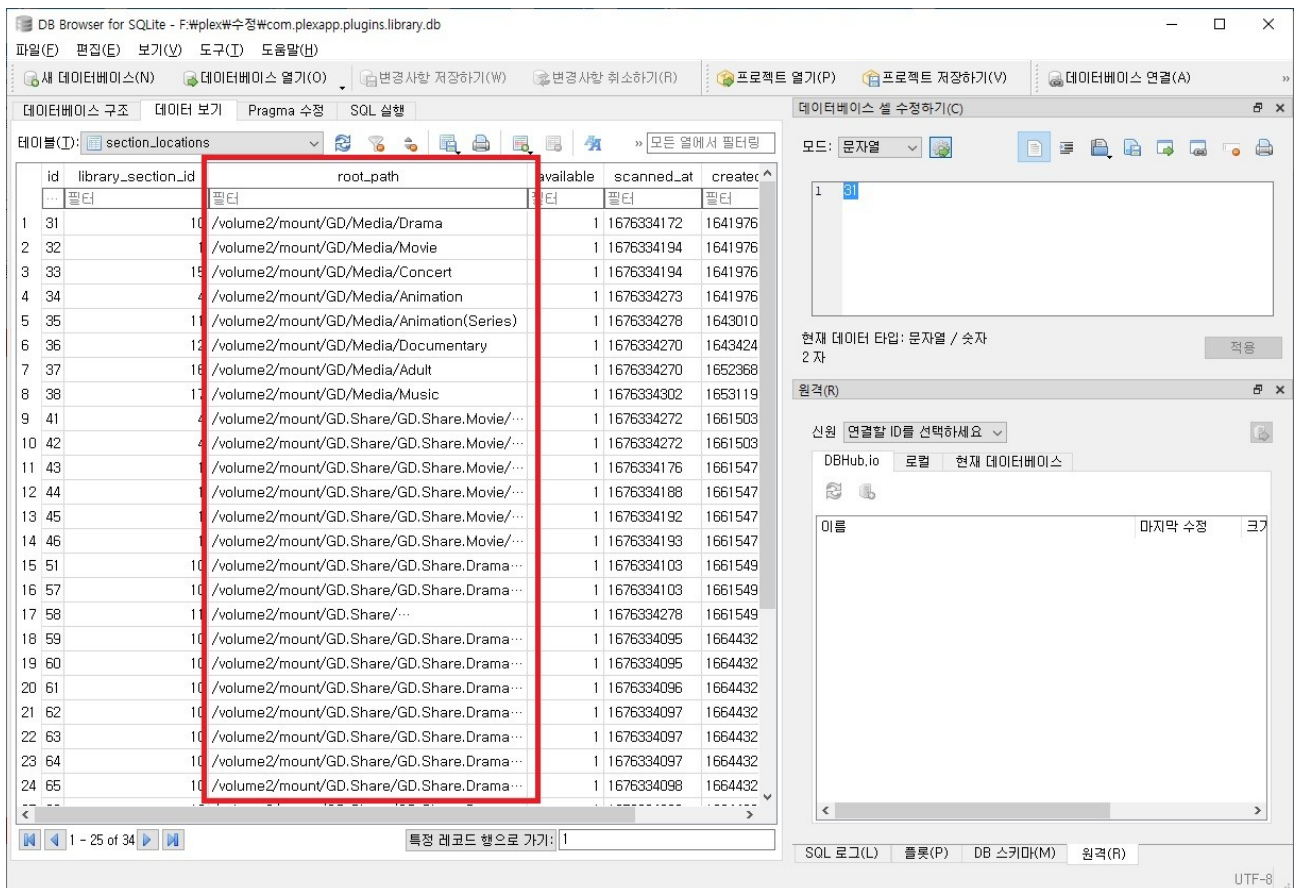
"section\_locations"는 등록된 영화, 드라마, 다큐 등등의 각각의 경로가 지정된 테이블이고,

"media\_parts"는 각각 파일들의 경로 및 메타데이터와 연결하기 위한 id 값을 가지고 있는 테이블 입니다.

[각 테이블들]



## [section\_locations 내용]



## [media\_parts 내용]

D8 Browser for SQLite - F:\plex\#수정\com.plexapp.plugins.library.db

파일(F) 편집(E) 보기(V) 도구(T) 도움말(H)

새 데이터베이스(N) 데이터베이스 열기(O) 변경사항 저장하기(W) 변경사항 취소하기(R) 프로젝트 열기(P) 프로젝트 저장하기(V) 데이터베이스 연결(A)

데이터베이스 구조 데이터 보기 Pragma 설정 SQL 실행

데이터베이스 설정하기(C)

모드: 문자열

현재 데이터 타입: 문자열 / 숫자 6 자

원격(R)

신원 연결할 ID를 선택하세요

DBHub.io 로컬 현재 데이터베이스

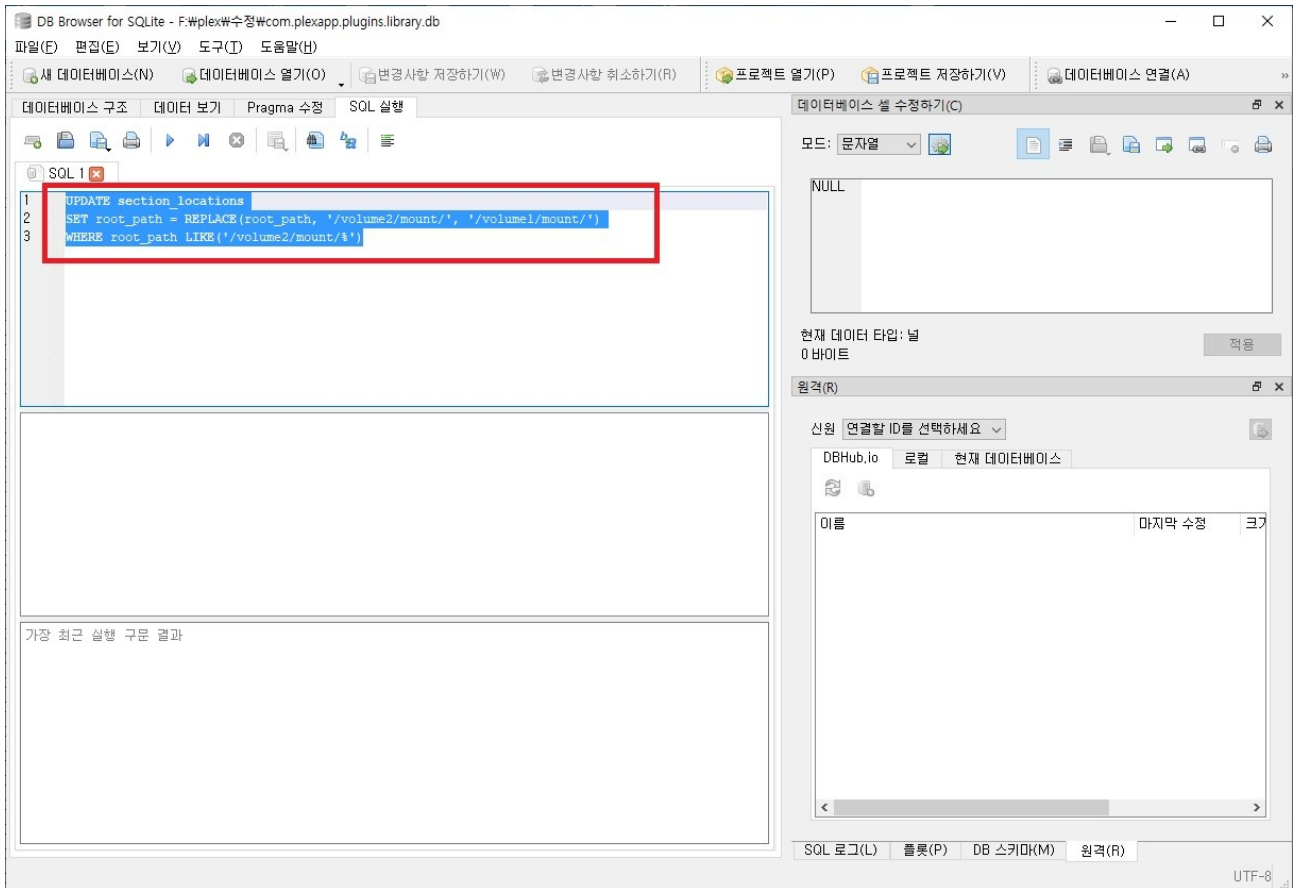
이름 마지막 수정 크기

SQL 로그(L) 플러그인(P) DB 스키마(M) 원격(R)

UTF-8

file_hash	file	index
31033	5c77	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E04.mp4
31034	3ad78	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E05.mp4
31035	1a	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E06.mp4
31036	237ec	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E07.mp4
31037	84b3	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E08.mp4
31038	818fb	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E09.mp4
31039	7e3a	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E10.mp4
31040	c3a9d	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E11.mp4
31041	5478e	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E12.mp4
31042	3a155	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E13.mp4
31043	cc38	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E14.mp4
31044	2974	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E15.mp4
31045	80d4	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E16.mp4
31046	e67dc	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E17.mp4
31047	5db73	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E18.mp4
31048	39f	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E19.mp4
31049	139e	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E20.mp4
31050	30ab8	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E21.mp4
31051	2bdd	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E22.mp4
31052	1acad6	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E23.mp4
31053	3f80	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E24.mp4
31054	3f3d1	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E25.mp4
31055	3961	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E26.mp4
31056	3ea63	/volume2/mount/GD/Media/Drama/신조협려 2014/신조협려 2014.S01E27.mp4

DB를 수정합니다. 수정은 SQL 탭으로 가서서 스크립트를 작성하고 "Run"을 하시면 됩니다.



"section\_locations" 수정 SQL 예제 입니다.

```
UPDATE section_locations
SET root_path = REPLACE(root_path, '/volume2/mount/', '/volume1/mount/')
WHERE root_path LIKE ('/volume2/mount/%')
```

위 SQL 쿼리에 대해 간단히 설정 드리면,

- (1) "section\_locations" 테이블에 대해 아래 조건으로 업데이트를 합니다.
- (2) "section\_locations" 테이블의 "root\_path" 열에서 "/volume2/mount/"로 시작하는 경로들을 모두 찾습니다.

여기서 뒤에 있는 "%"는 "/volume2/mount/" 뒤에 어떤 값이든 다 허용해서 찾는다는걸 의미 합니다.

- (3) 위에서 찾은 "root\_path"에 있는 값을 바꿔 줍니다. 위에 있는 샘플은 "/volume2/mount/" 값을 "/volume1/mount/" 값으로 바꾼다는 예제 입니다.
- (4) 아래의 media\_parts에 대한 SQL쿼리도 동일 원리 입니다.
- (5) 만일 update 쿼리를 즉시 반영하기 싫으시다면 update 쿼리 실행 이전에 BEGIN 명령을 실행 (트랜잭션 기능) --> UPDATE쿼리 실행 -> COMMIT or ROLLBACK 명령을 하시면 됩니다.

COMMIT을 하면 업데이트가 되고 ROLLBACK을 하면 업데이트를 취소하고 이전으로 돌아 갑니다.

다음으로 "media\_parts" 수정 SQL 예제 입니다.

```
UPDATE media_parts
SET file = REPLACE(file, '/volume2/mount/', '/volume1/mount/')
WHERE file LIKE ('/volume2/mount/%')
```

이 예제는 ....volume2/....를 ....volume1/.... 으로 바꾸는 예제 입니다. 각 상황에 맞게 수정하시면 됩니다.

수정 후 db를 저장하고 새로 이동할 동일 위치로 파일을 복사 합니다.

"com.plexapp.plugins.library.blobs.db"의 경우 별도 수정 없이 그냥 복사 후 이동하시면 됩니다.

파일들이 외부에서 넘어 왔으므로 변경된 plex 디렉토리 및 파일들의 소유권을 변경해 주셔야 합니다.

```
chown -R PlexMediaServer:PlexMediaServer AppData
```

끝으로 Plex 서비스를 실행합니다.

**[DB Schema] ★ = PK**



library_sections	section_locations	metadata_items	media_items
★id	★id	★id	★id
library_id	library_section_id	library_section_id	library_section_id
name	root_path	parent_id	section_location_id
name_sort	available	metadata_type	metadata_item_id
section_type	scanned_at	guid	type_id
language	created_at	media_item_count	width
agent	updated_at	title	height
scanner		title_sort	size
user_thumb_url	directories	original_title	duration
user_art_url	★id	studio	bitrate
user_theme_music_url	library_section_id	rating	container
public	parent_directory_id	rating_count	video_codec
created_at	path	tagline	audio_codec
updated_at	created_at	summary	display_aspect_ratio
scanned_at	updated_at	trivia	frames_per_second
display_secondary_level	deleted_at	quotes	audio_channels
user_fields		content_rating	interlaced
query_xml	media_streams	content_rating_age	source
query_type	★id	index	hints
uuid	stream_type_id	absolute_index	display_offset
changed_at	media_item_id	duration	settings
content_changed_at	url	user_thumb_url	created_at
	codec	user_art_url	updated_at
media_parts	language	user_banner_url	optimized_for_streaming
★id	created_at	user_music_url	deleted_at
media_item_id	updated_at	user_fields	media_analysis_version
directory_id	index	tags_genre	sample_aspect_ratio
hash	media_part_id	tags_collection	extra_data
open_subtitle_hash	channels	tags_director	proxy_type
file	bitrate	tags_writer	channel_id
index	url_index	tags_star	begins_at
size	default	originally_available_at	ends_at
duration	forced	available_at	color_trc
created_at	extra_data	expires_at	
updated_at		refreshed_at	
deleted_at	tags	year	
extra_data	★id	added_at	
	metadata_item_id	created_at	
taggings	tag	updated_at	
★id	tag_type	deleted_at	
metadata_item_id	user_thumb_url	tags_country	
tag_id	user_art_url	extra_data	
index	user_music_url	hash	
text	created_at	audience_rating	
time_offset	updated_at	changed_at	
end_time_offset	tag_value	resources_changed_at	
thumb_url	extra_data	remote	
created_at	key		
extra_data	parent_id		

🔄버전 #10

★생성 23 6월 2023 01:23:15, DH Choi

✎수정 9 4월 2024 04:19:19, Admin