웹 브라우저 포렌식

IE/MSEdge, Chrome, Firefox

2022-07-04 양봉열

LOGPRESSO

배경

엔드포인트 공격은 어디에서 주로 시작될까?

피싱	• 이메일, 문자 메시지, SNS 등을 통해 소셜 엔지니어링 공격 • 인증을 유도하여 계정과 암호를 탈취하거나 트로이 목마 설치
드라이브-바이 컴프로마이즈	• 취약한 상태에서 악성 웹사이트 방문으로 감염
원격 서비스	• VPN, Citrix, 팀뷰어, VNC 등 외부에 노출된 다양한 원격 제어 서비스를 통해 침투 • 다크웹에서 유출 정보를 구매하거나 일부 자산 공격 성공 후 계정과 암호를 얻어내어 활용
이동식 장치	• 윈도우 10 이전의 구형 시스템은 USB 연결 후 자동 실행 등을 통해 감염될 수 있음
공급망 공격	• 상대적으로 취약한 공급망 체계를 공격하여 개발 도구나 패키지에 악성코드를 삽입할 수 있음

추모의 시간


故 Internet Explorer
1995. 8. 17 ~ 2022. 6.15
He was a good tool to
download other browsers.

웹 브라우저 아티팩트

- 웹 사이트 접속 기록
- 쿠키 파일 메타데이터
- 캐시 파일 메타데이터
- 파일 다운로드 이력

IE10-11
ESE database

- 키워드 검색 기록
- 북마크
- 웹 사이트 접속 기록
- 쿠키
- 파일 다운로드 이력

SQLite database

- 북마크
- 웹 사이트 접속 기록
- 쿠키
- 파일 다운로드 이력

SQLite database

오늘의 주제

브라우저 아티팩트 자체에 대한 내용은 많이 알려져 있음
그러나 데이터베이스 구조에 대한 자료는 거의 없음

ESE DB 구조

SQLite DB 구조

ESE 데이터베이스

30년을 이어온 제트블루 엔진

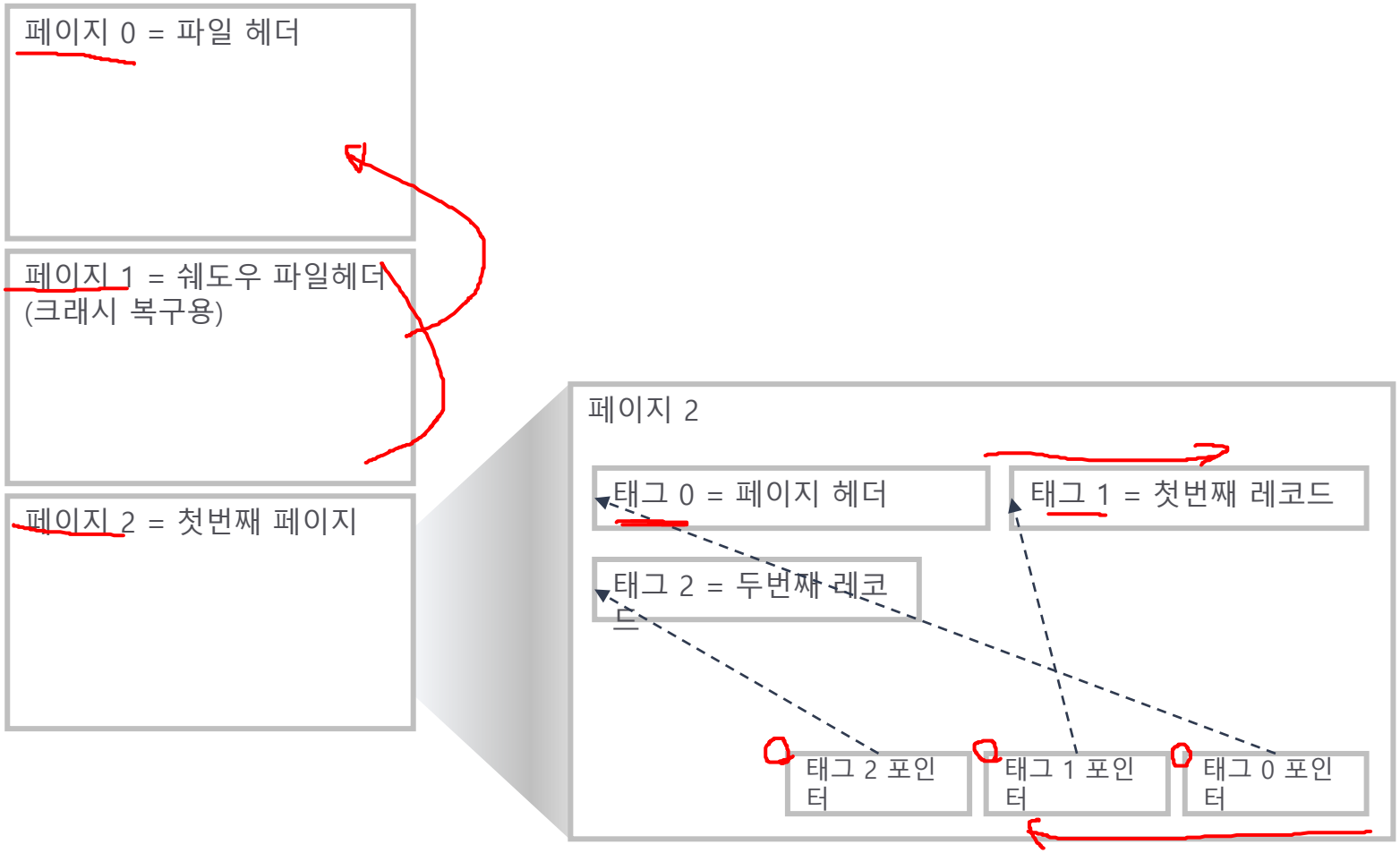
ESE 데이터베이스

ESE DB는 윈도우 애플리케이션에서 흔히 사용되는 스토리지 엔진

개요	• Extensible Storage Engine (JET Blue) • 마이크로소프트에서 1992년 개발, ISAM 파일
활용	• 웹 브라우저 (IE, MSEdge) • 익스체인지 서버, 액티브 디렉터리, 윈도우 서치 등 다양한 응용의 데이터 기반
도구	• Nirsoft ESEDatabaseView
코드	• 2021년 봄에 전체 코드가 MIT 라이선스로 github에 공개됨 • https://github.com/microsoft/Extensible-Storage-Engine

ESE DB 구조

DB는 B+트리 페이지로 나누어지고, 각 페이지는 태그로 구성됩니다.



카탈로그

카탈로그는 4번 페이지에 존재 5 x 페이지 크기 하고,
테이블, 인덱스, 컬럼 정의 등 모든 객체에 대한 메타정보를 관리합니다.

ESEDatabaseView: E:₩logsample₩forensic₩ie₩WebCacheV01.dat

File Edit View Options Help

FDP = Father Data Page
테이블별 최상위 페이지 번호

MSysObjects [Table ID = 2, 27 Columns]

ObjidTable	Type	Id	ColtypOrPgnoFDP	SpaceUsage	Flags	PagesOrLocale	RootFlag	RecordOffset	LCMapFlags	KeyMost	Name	Stats
2	1	2	4	80	-1073741824	20	255				MSysObjects	
2	2	1	4	4	1	1252	42	4			ObjidTable	
2	2	2	3	2	1	1252	42	4			Type	
2	2	3	4	4	1	1252	42	4			Id	
2	2	4	4	4	1	1252	42	4			ColtypOrPgnoFDP	
2	2	5	4	4	1	1252	42	4			SpaceUsage	
2	2	6	4	4	1	1252	42	4			Flags	
2	2	7	4	4	1	1252	42	4			PagesOrLocale	
2	2	8	1	1	0	1252	42	4			RootFlag	
2	2	9	3	2	0	1252	42	4			RecordOffset	
2	2	10	4	4	0	1252	42	4			LCMapFlags	
2	2	11	17	2	0	1252	42	4			KeyMost	
2	2	128	10	255	1	1252					Name	
2	2	129	9	255	0	1252					Stats	
2	2	130	10	255	0	1252					TemplateTable	
2	2	131	9	255	0	1252					DefaultValue	
2	2	132	9	255	0	1252					KeyFldIDs	
2	2	133	9	255	0	1252					VarSegMac	
2	2	134	9	255	0	1252					ConditionalColumns	
2	2	135	9	255	0	1252					TupleLimits	
2	2	136	9	255	0	1252					Version	
2	2	137	9	255	0	1252					SortID	
2	2	256	11	0	0	1252					CallbackData	
2	2	257	11	0	0	1252					CallbackDependencies	
2	2	258	11	0	0	1252					SeparateLV	

타입 1
테이블

타입 2
컬럼

타입 3
인덱스

타입 4
긴 값 Long Value

타입 5
콜백

파일 헤더

WebCacheV01.dat 파일 헤더를 아래와 같이 확인할 수 있습니다.

WebCacheV01.dat

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00000000  00 77 04 C6 EF CD AB 89 20 06 00 00 00 00 00 00  .w.ÆïÍ«‰ .......
00000010  15 51 BF 00 00 00 00 00 F7 E4 0F B9 1C 0D 03 19  .Q¿.....÷ä.¹....
00000020  08 73 43 02 00 00 00 00 00 00 00 00 00 00 00 00  .sC.............
00000030  00 00 00 00 02 00 00 00 2E 01 14 00 4B 21 00 00  ............K!..
00000040  0A 09 07 0C 06 79 21 02 39 0D 07 0C 06 79 81 02  .....y!.9....y..
00000050  68 02 16 00 4B 21 00 00 00 00 00 00 00 00 00 00  h...K!..........
```

오프셋	크기	값	설명
4	4	EF CD AB 89	시그니처
8	4	20 06 00 00	파일 포맷 버전

```
000000D0  00 00 00 00 AE 14 00 00 0A 00 00 00 00 00 00 00  ....®...........
000000E0  BB 47 00 00 00 00 00 00 50 00 00 00 00 80 00 00  »G......P....€..
000000F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

오프셋	크기	값	설명
216	4	0A 00 00 00	NT 메이저 버전
220	4	00 00 00 00	NT 마이너 버전
224	4	BB 47 00 00	빌드 1909 해당
236	4	00 08 00 00	페이지 크기

http://forensic-proof.com/wp-content/uploads/2011/07/Extensible-Storage-Engine-ESE-Database-File-EDB-format.pdf

페이지 헤더

MSysObjects 카탈로그에 해당하는 페이지 4 헤더

```
00028000  97 F3 CA 4C 9F C1 60 3E C0 49 B3 00 00 00 00 00  —óÊLŸÁ`>ÀI³.....
00028010  00 00 00 00 00 00 00 00 02 00 00 00 3E 7E 00 00  ............>~..
00028020  5F 01 11 00 05 28 01 00 41 1D 30 BF DE D0 DE D0  _....(..A.0¿ÞÐÞÐ
00028030  70 74 A8 D4 D6 C3 29 3C 24 7B 2D 1D C1 E9 3E 16  pt¨ÔÖÃ)<${-.Áé>.
00028040  04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00028050  14 00 00 00 01 00 00 00 01 00 00 00 05 00 00 00  ................
00028060  00 00 7B 04 00 00 0D 00 7F 80 00 00 1E 7F 80 02  ..{.......€....€.
```

오프셋	크기	값	설명
16	4	00 00 00 00	(리프) 왼쪽 페이지
20	4	00 00 00 00	(리프) 오른쪽 페이지
24	4	02 00 00 00	부모 데이터 페이지
34	2	11 00	페이지 태그 수
36	2	05 28 01 00	페이지 플래그

페이지 플래그

부모

0001 0010 1000 0000 0101

루트

(0x012805)

값	설명
0x0001	루트 페이지
0x0002	리프 페이지
0x0004	부모 페이지
0x0008	빈 페이지
0x0020	스페이스 트리 페이지
0x0040	인덱스 페이지
0x0080	긴 값 페이지

레코드

헤더 | 고정 길이 컬럼 배열 | 가변 길이 컬럼 배열 | 태그 배열

```
000705D0  61 6C 65 4E 61 6D 65 07 A0 06 00 03 7F 80 00 00  aleName. ....€..
000705E0  02 0A 84 27 00 02 00 00 00 03 00 02 00 00 00 04  ..„'............
000705F0  00 00 00 50 00 00 00 31 00 01 00 00 00 00 00 2A  ...P...1.......*
00070600  2A 2A 01 04 03 00 80 FD 02 00 02 80 02 80 02 80  **....€ý...€.€.€
00070610  0E 00 49 64 00 00 01 00 00 00 02 00 00 00 03 00  ..Id............
```

오프셋	크기	값	설명
0	2	07 A0	공통 페이지 키
2	2	06 00	로컬 페이지 키 크기
4	6	03 7F 80 00 00 02	로컬 페이지 키 (앞에서 6바이트 지정)
10	1	0A	마지막 고정 길이 데이터 타입 번호
11	1	84	마지막 가변 길이 데이터 타입 번호 0x84 - 0x80 + 1 = 가변 타입 5개
12	2	27 00	페이지 키 이후의 위치 기준으로 가변 길이 컬럼 배열 시작 오프셋 0x705E1 + 0x27 = 0x70608
14	35	02 00 .. 80 FD	고정 길이 컬럼 값 배열
49	10	02 00 02 80 02 80 02 80 0E 00	가변 타입별 길이 (5개 x 2바이트)
59	2	49 64	Id 텍스트 값 (2바이트)

컬럼 타입

타입 번호	컬럼 이름	타입	바이트	값
1	ObjidTable	Long	4	02 00 00 00
2	Type	Short	2	03 00
3	Id	Long	4	02 00 00 00
4	ColtypOrPgnoFDP	Long	4	04 00 00 00
5	SpaceUsage	Long	4	50 00 00 00
6	Flags	Long	4	31 00 01 00
7	PagesOrLocale	Long	4	00 00 00 00
8	RootFlag	Bit	1	2A
9	RecordOffset	Short	2	2A 2A
10	LCMapFlags	Long	4	01 04 03 00
11	KeyMost	Short	2	80 FD

```
000705D0  61 6C 65 4E 61 6D 65 07 A0 06 00 03 7F 80 00 00  aleName. ....€..
000705E0  02 0A 84 27 00 02 00 00 00 03 00 02 00 00 00 04  ..„'............
000705F0  00 00 00 50 00 00 00 31 00 01 00 00 00 00 00 2A  ...P...1.......*
00070600  2A 2A 01 04 03 00 80 FD 02 00 02 80 02 80 02 80  **....€ý...€.€.€
00070610  0E 00 49 64 00 00 01 00 00 00 02 00 00 00 03 00  ..Id............
```

MSysObjects 카탈로그의 고정 길이 컬럼 예

번호	타입	설명
0	NIL	유효하지 않은 컬럼 타입
1	BIT	불린, 고정 1바이트
2	Unsigned Byte	부호 없는 8비트 정수
3	Short	부호 있는 16비트 정수
4	Long	부호 있는 32비트 정수
5	Currency	통화 64비트 정수
6	IEEE Single	32비트 부동소수점 실수
7	IEEE Double	64비트 부동소수점 실수
8	DateTime	윈도우 FILETIME 시각
9	Binary	255바이트 이하 바이너리
10	Text	255바이트 이하 텍스트
11	Long Binary	최대 2GB 바이너리
12	Long Text	최대 2GB 텍스트
13	SLV	Super Large Value
14	Unsigned Long	부호 없는 32비트 정수
15	Long Long	부호 있는 64비트 정수
16	GUID	128비트 GUID
17	Unsigned Short	부호 없는 16비트 정수

MSysObjects [Table ID = 2, 27 Columns]

ObjidTable	Id	ColtypOrPgnoFDP	SpaceUsage	Flags	PagesOrLocale	RootFlag	RecordOffset	LCMapFlags	KeyMost	Name
2	3	2	4	80	65585	0	42	10794	197633	Id

태그된 컬럼 tagged column

가변 길이 컬럼 ID 범위: 128 <= 컬럼 ID < 256

태그된 컬럼 ID 범위: 256 <= 컬럼 ID < 65535

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0465FFF0  06 02 9B 1A 8D 02 A5 02 93 02 12 00 00 00 00 00  ..›...¥.“.......
04660000  C3 38 7D 6B 7D 1A 7D 1A 62 A7 A3 00 00 00 00 00  Ã8}k}.}.b§£.....
04660010  00 00 00 00 FC 00 00 00 28 00 00 00 92 26 41 00  ....ü...(...’&A.
04660020  4D 5E 2D 00 02 A8 01 00 24 70 87 4F 5A 7D 5A 7D  M^-..¨..$p‡OZ}Z}
04660030  72 23 19 B8 50 A4 50 A4 C3 B2 5F 15 13 64 EC 9B  r#.¸P¤P¤Ã²_..dì›
04660040  CB 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00  Ë...............
04660050  7F 92 29 81 61 2F 9F 69 1F 7F 80 00 00 00 00 00  .’).a/Ÿi..€.....
04660060  13 F2 05 A0 0D 00 13 BF 69 E3 7F 80 00 00 00 00  .ò. ...¿iã.€....
04660070  00 13 EB 11 7F 77 00 EB 13 00 00 00 00 00 00 12  ..ë..w.ë........
04660080  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E3  ...............ã
04660090  69 BF 13 61 81 29 12 00 00 00 00 00 00 00 00 00  i¿.a.)..........
046600A0  00 00 00 01 00 20 00 00 00 00 00 01 00 00 00 73  ..... .........s
046600B0  A3 57 05 73 48 D5 01 00 00 00 00 00 00 00 00 E2  £W.sHÕ.........â
046600C0  EE FF EC E2 93 D5 01 73 A3 57 05 73 48 D5 01 73  îÿìâ“Õ.s£W.sHÕ.s
046600D0  A3 57 05 73 48 D5 01 00 00 00 00 00 00 00 00 01  £W.sHÕ..........
046600E0  00 00 00 00 00 00 00 00 00 FE 00 01 08 00 04 01  .........þ......
046600F0  0D 00 05 D6 0A 00 00 01 79 00 00 00 75 00 00 00  ...Ö....y...u...
04660100  31 53 50 53 A1 14 02 00 00 00 00 00 C0 00 00 00  1SPS¡.......À...
04660110  00 00 00 46 11 00 00 00 17 00 00 00 00 13 00 00  ...F............
04660120  00 00 00 00 00 15 00 00 00 18 00 00 00 00 40 00  ..............@.
04660130  00 00 80 33 3F 75 BE 48 D5 01 11 00 00 00 0D 00  ..€3?u¾HÕ.......
04660140  00 00 00 13 00 00 00 00 00 00 00 11 00 00 00 09  ................
04660150  00 00 00 00 13 00 00 00 00 00 00 00 11 00 00 00  ................
04660160  06 00 00 00 00 13 00 00 00 00 00 00 00 00 00 00  ................
04660170  00 00 00 00 00 53 00 00 00 4F 00 00 00 31 53 50  .....S...O...1SP
04660180  53 A1 14 02 00 00 00 00 00 C0 00 00 00 00 00 00  S¡.......À......
04660190  46 11 00 00 00 21 00 00 00 00 13 00 00 00 00 00  F....!..........
046601A0  00 00 11 00 00 00 1C 00 00 00 00 03 00 00 00 00  ................
046601B0  00 00 00 11 00 00 00 20 00 00 00 00 03 00 00 00  ....... ........
046601C0  00 00 00 00 00 00 00 00 00 00 00 00 12 A0 00 00  ............ ..
046601D0  11 7F 77 00 F2 13 00 00 00 00 00 00 12 00 00 00  ..w.ò...........
046601E0  00 00 00 00 00 00 00 00 00 00 00 00 1F 69 9F 2F  .............iŸ/
```

오프셋	크기	값 (hex)	설명
0	1	11	마지막 고정 길이 데이터 타입 번호
1	1	7F	마지막 가변 길이 데이터 타입 번호 0x7F + 1 - 0x80 = 가변 타입 0개
2	2	77 00	페이지 키 이후의 위치 기준으로 가변 길이 컬럼 배열 시작 오프셋 0x4660073 + 0x77 = 46600EA
119	2	00 01	첫번째 태그 컬럼 ID 256
121	2	08 00	태그 컬럼 영역 시작 기준 첫번째 태그 값 오프셋
123	2	04 01	두번째 태그 컬럼 ID 260
125	2	0D 00	태그 컬럼 영역 시작 기준 두번째 태그 값 오프셋
127	1	05	Long Value 타입
128	4	D6 0A 00 00	Long Value ID **2774**
132	1	01	일반 Value 타입

가변 길이 컬럼 배열 값을 합산하면 태그 영역 오프셋 계산됨

긴 값 long value

MSysObjects 카탈로그에 LV (Long Value) 테이블에 대한 메타 정보 유지

MSysObjects [Table ID = 2, 27 Columns]

ObjidTable	Type	Id	ColtypOrPgnoFDP	SpaceUsage	Flags	PagesOrLocale	RootFlag	RecordOffset	LCMapFlags	KeyMost	Name
39	3	39	237	60	65583	0	42	10794	197633		HashEntryIdIndex
39	4	568	246	100	0	1					LV
40	1	40	247	60	0	10	255				Container_18
40	2	1	15	8	4	0	42	4			EntryId

MSysObjects [Table ID = 2, 27 Columns]

ObjidTable	Type	Id	ColtypOrPgnoFDP	SpaceUsage	Flags	PagesOrLocale	RootFlag	RecordOffset	LCMapFlags	KeyMost	Name
40	2	262	11	65536	0	0					Group
40	2	263	11	65536	0	0					ExtraData
40	3	40	247	60	65583	0	42	10794	197633		HashEntryIdIndex
40	4	68	248	100	0	1					LV
41	1	41	257	60	0	10	255				AppCacheEntryEx_4

Container_18 테이블의 Id=40

Id=68 테이블이 Container_18에 대한 LV 테이블이고, 루트 페이지 번호는 248

긴 값 long value

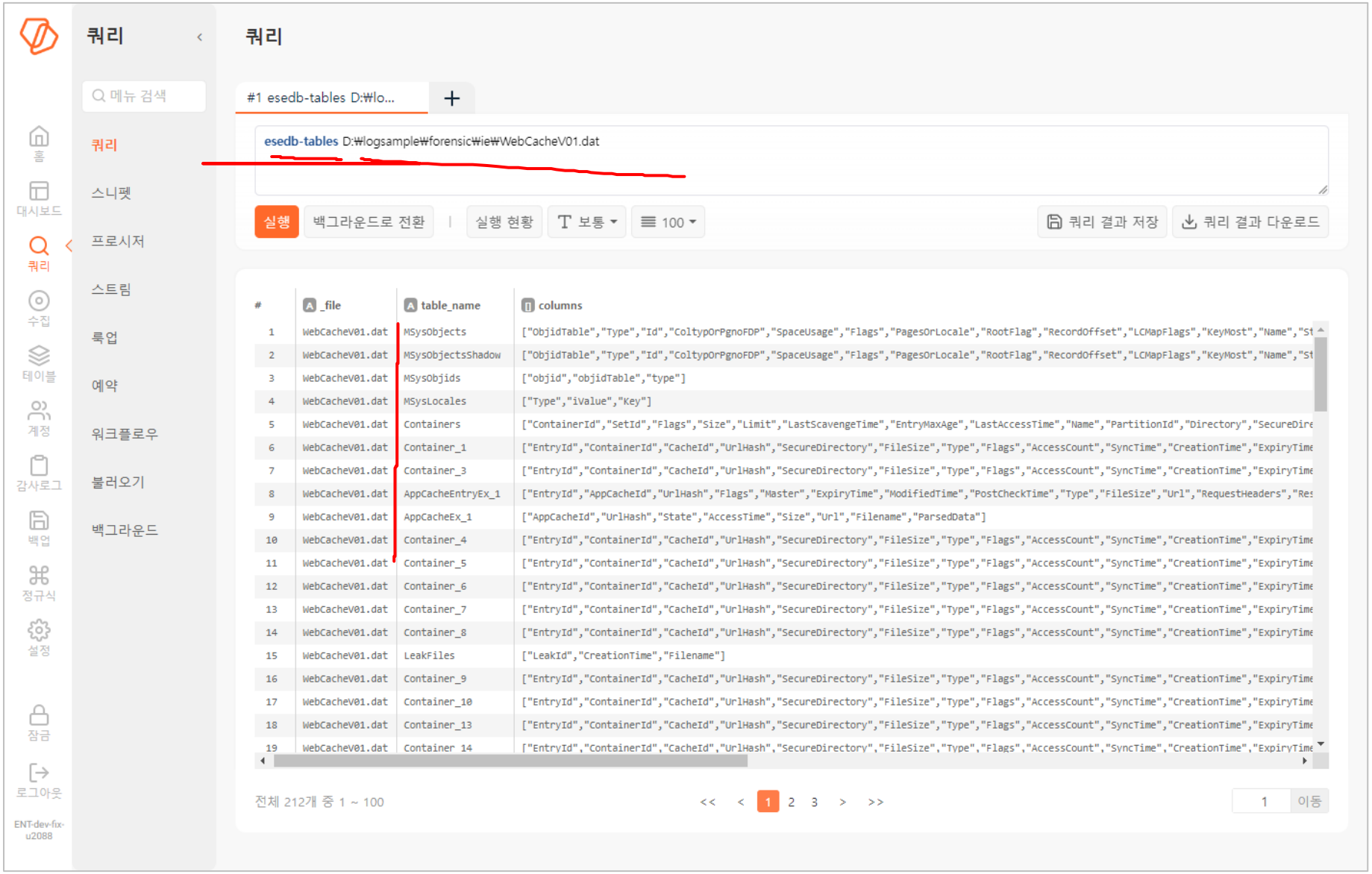
태그 1 위치 (페이지 헤더 80바이트 바로 뒷 부분)에 빅엔디안으로 첫번째 LV ID 기입
첫번째 LV ID 0xAD5의 MSB 3바이트에 태그의 D6 1바이트를 조합하면 LV ID 0xAD6
(**2774**)

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

05460000  C6 6E 65 84 EA DF 15 20 2F BC A0 00 00 00 00 00  Æne„êß. /¼ .....
05460010  8A 0A 00 00 62 06 00 00 44 00 00 00 EE 07 00 00  Š...b...D...î...
05460020  86 77 0F 00 82 28 01 00 FD 06 8D 28 45 8E BA 71  †w..‚(..ý..(EŽºq
05460030  10 08 86 5F 1E 46 E1 B9 AE 0D B5 B3 67 0C 67 0C  ..†_.Fá¹®.µ³g.g.
05460040  8B 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ‹...............
05460050  00 00 0A D5 00 00 3F AC 03 A0 05 00 D6 00 00 00  ...Õ..?¬. ..Ö...
05460060  00 56 00 69 00 73 00 69 00 74 00 65 00 64 00 3A  .V.i.s.i.t.e.d.:
05460070  00 20 00 78 00 65 00 72 00 61 00 70 00 68 00 40  . .x.e.r.a.p.h.@
05460080  00 68 00 74 00 74 00 70 00 73 00 3A 00 2F 00 2F  .h.t.t.p.s.:././
05460090  00 63 00 64 00 6E 00 2E 00 62 00 61 00 6E 00 6E  .c.d.n...b.a.n.n
054600A0  00 65 00 72 00 66 00 6C 00 6F 00 77 00 2E 00 63  .e.r.f.l.o.w...c
054600B0  00 6F 00 6D 00 2F 00 62 00 66 00 2D 00 62 00 61  .o.m./.b.f.-.b.a
054600C0  00 6E 00 6E 00 65 00 72 00 73 00 2F 00 35 00 63  .n.n.e.r.s./.5.c
054600D0  00 62 00 65 00 62 00 35 00 39 00 36 00 34 00 39  .b.e.b.5.9.6.4.9
054600E0  00 64 00 34 00 63 00 37 00 31 00 64 00 62 00 63  .d.4.c.7.1.d.b.c
054600F0  00 61 00 31 00 61 00 31 00 65 00 65 00 2E 00 65  .a.1.a.1.e.e...e
05460100  00 6A 00 79 00 53 00 6E 00 7A 00 76 00 76 00 32  .j.y.S.n.z.v.v.2
05460110  00 53 00 66 00 4B 00 2E 00 68 00 74 00 6D 00 6C  .S.f.K...h.t.m.l
05460120  00 3F 00 63 00 62 00 3D 00 36 00 33 00 36 00 39  .?.c.b.=.6.3.6.9
05460130  00 39 00 37 00 32 00 32 00 36 00 35 00 36 00 30  .9.7.2.2.6.5.6.0
05460140  00 36 00 34 00 35 00 34 00 39 00 36 00 26 00 63  .6.4.5.4.9.6.&.c
05460150  00 6C 00 69 00 63 00 6B 00 70 00 69 00 78 00 65  .l.i.c.k.p.i.x.e
05460160  00 6C 00 3D 00 25 00 32 00 46 00 25 00 32 00 46  .l.=.%.2.F.%.2.F
05460170  00 35 00 38 00 39 00 38 00 37 00 33 00 62 00 64  .5.8.9.8.7.3.b.d
05460180  00 35 00 61 00 34 00 65 00 38 00 37 00 34 00 39  .5.a.4.e.8.7.4.9
05460190  00 62 00 63 00 65 00 34 00 30 00 33 00 66 00 66  .b.c.e.4.0.3.f.f
054601A0  00 2E 00 74 00 72 00 61 00 63 00 6B 00 65 00 72  ...t.r.a.c.k.e.r
054601B0  00 2E 00 62 00 61 00 6E 00 6E 00 65 00 72 00 66  ...b.a.n.n.e.r.f
054601C0  00 6C 00 6F 00 77 00 2E 00 63 00 6F 00 6D 00 25  .l.o.w...c.o.m.%
054601D0  00 32 00 46 00 61 00 70 00 69 00 25 00 32 00 46  .2.F.a.p.i.%.2.F
054601E0  00 74 00 72 00 25 00 32 00 46 00 63 00 6C 00 69  .t.r.%.2.F.c.l.i
054601F0  00 63 00 6B 00 25 00 33 00 46 00 64 00 61 00 74  .c.k.%.3.F.d.a.t
```

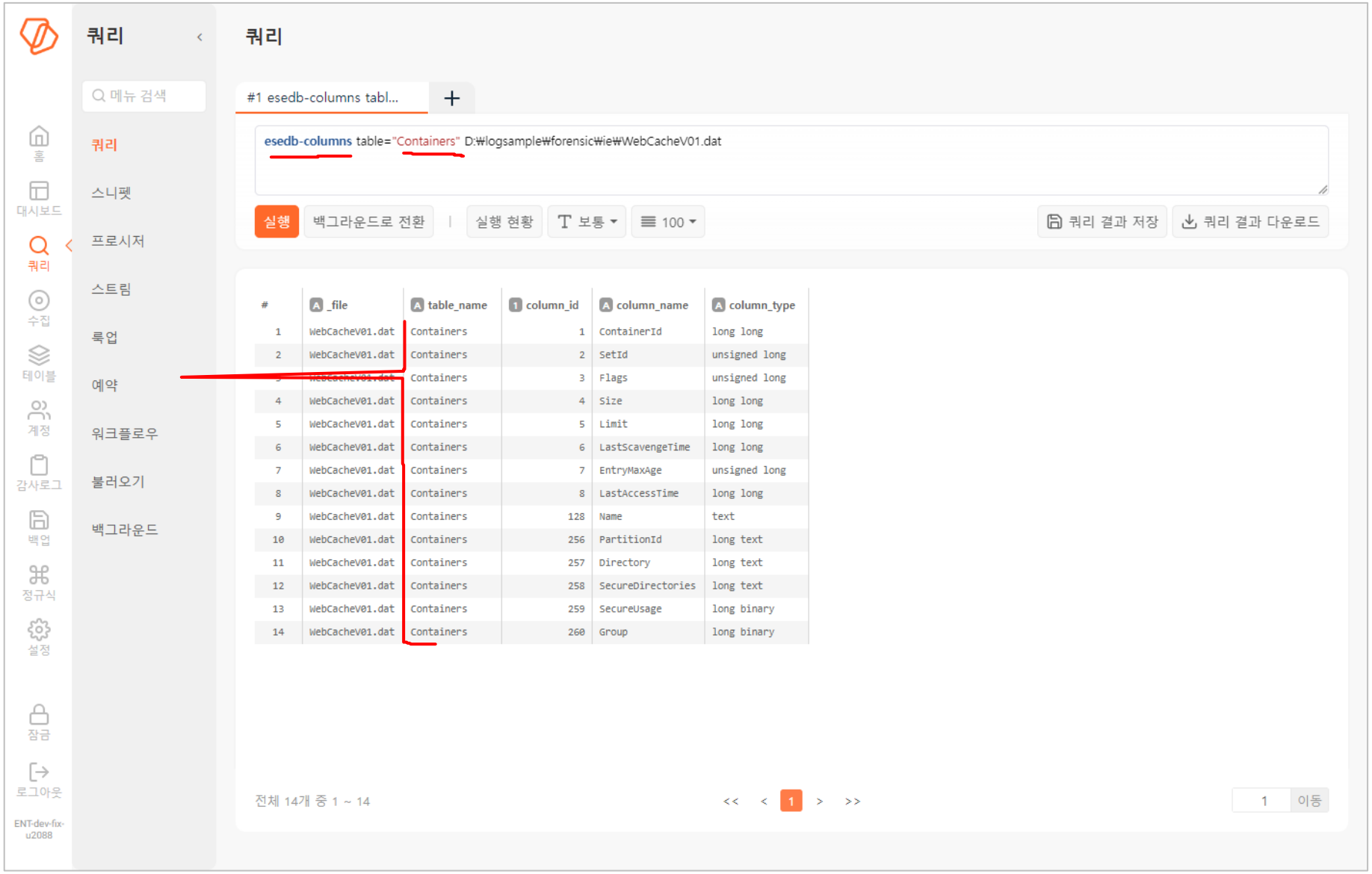
오프셋	크기	값 (hex)	설명
0	2	03 A0	
2	2	05 00	
4	5	D6 00 00 00 00	
9	..		Long Value 데이터 부
			여러 개의 태그로 연속될 수 있음

ESE DB - 테이블 목록 조회



#	_file	table_name	columns
1	WebCacheV01.dat	MSysObjects	["ObjidTable","Type","Id","ColtypOrPgnoFDP","SpaceUsage","Flags","PagesOrLocale","RootFlag","RecordOffset","LCMapFlags","KeyMost","Name","St
2	WebCacheV01.dat	MSysObjectsShadow	["ObjidTable","Type","Id","ColtypOrPgnoFDP","SpaceUsage","Flags","PagesOrLocale","RootFlag","RecordOffset","LCMapFlags","KeyMost","Name","St
3	WebCacheV01.dat	MSysObjids	["objid","objidTable","type"]
4	WebCacheV01.dat	MSysLocales	["Type","iValue","Key"]
5	WebCacheV01.dat	Containers	["ContainerId","SetId","Flags","Size","Limit","LastScavengeTime","EntryMaxAge","LastAccessTime","Name","PartitionId","Directory","SecureDire
6	WebCacheV01.dat	Container_1	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
7	WebCacheV01.dat	Container_3	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
8	WebCacheV01.dat	AppCacheEntryEx_1	["EntryId","AppCacheId","UrlHash","Flags","Master","ExpiryTime","ModifiedTime","PostCheckTime","Type","FileSize","Url","RequestHeaders","Res
9	WebCacheV01.dat	AppCacheEx_1	["AppCacheId","UrlHash","State","AccessTime","Size","Url","Filename","ParsedData"]
10	WebCacheV01.dat	Container_4	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
11	WebCacheV01.dat	Container_5	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
12	WebCacheV01.dat	Container_6	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
13	WebCacheV01.dat	Container_7	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
14	WebCacheV01.dat	Container_8	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
15	WebCacheV01.dat	LeakFiles	["LeakId","CreationTime","Filename"]
16	WebCacheV01.dat	Container_9	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
17	WebCacheV01.dat	Container_10	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
18	WebCacheV01.dat	Container_13	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime
19	WebCacheV01.dat	Container_14	["EntryId","ContainerId","CacheId","UrlHash","SecureDirectory","FileSize","Type","Flags","AccessCount","SyncTime","CreationTime","ExpiryTime

ESE DB - Containers 테이블 스키마



#	_file	table_name	column_id	column_name	column_type
1	WebCacheV01.dat	Containers	1	ContainerId	long long
2	WebCacheV01.dat	Containers	2	SetId	unsigned long
3	WebCacheV01.dat	Containers	3	Flags	unsigned long
4	WebCacheV01.dat	Containers	4	Size	long long
5	WebCacheV01.dat	Containers	5	Limit	long long
6	WebCacheV01.dat	Containers	6	LastScavengeTime	long long
7	WebCacheV01.dat	Containers	7	EntryMaxAge	unsigned long
8	WebCacheV01.dat	Containers	8	LastAccessTime	long long
9	WebCacheV01.dat	Containers	128	Name	text
10	WebCacheV01.dat	Containers	256	PartitionId	long text
11	WebCacheV01.dat	Containers	257	Directory	long text
12	WebCacheV01.dat	Containers	258	SecureDirectories	long text
13	WebCacheV01.dat	Containers	259	SecureUsage	long binary
14	WebCacheV01.dat	Containers	260	Group	long binary

ESE DB - IE History 메타데이터 조회

쿼리

메뉴 검색

쿼리
스니펫
프로시저
스트림
룩업
예약
워크플로우
불러오기
백그라운드

홈
대시보드
쿼리
수집
테이블
계정
감사로그
백업
정규식
설정
잠금
로그아웃
ENT-dev-fix-u2088

쿼리

#1 esedb-records table...

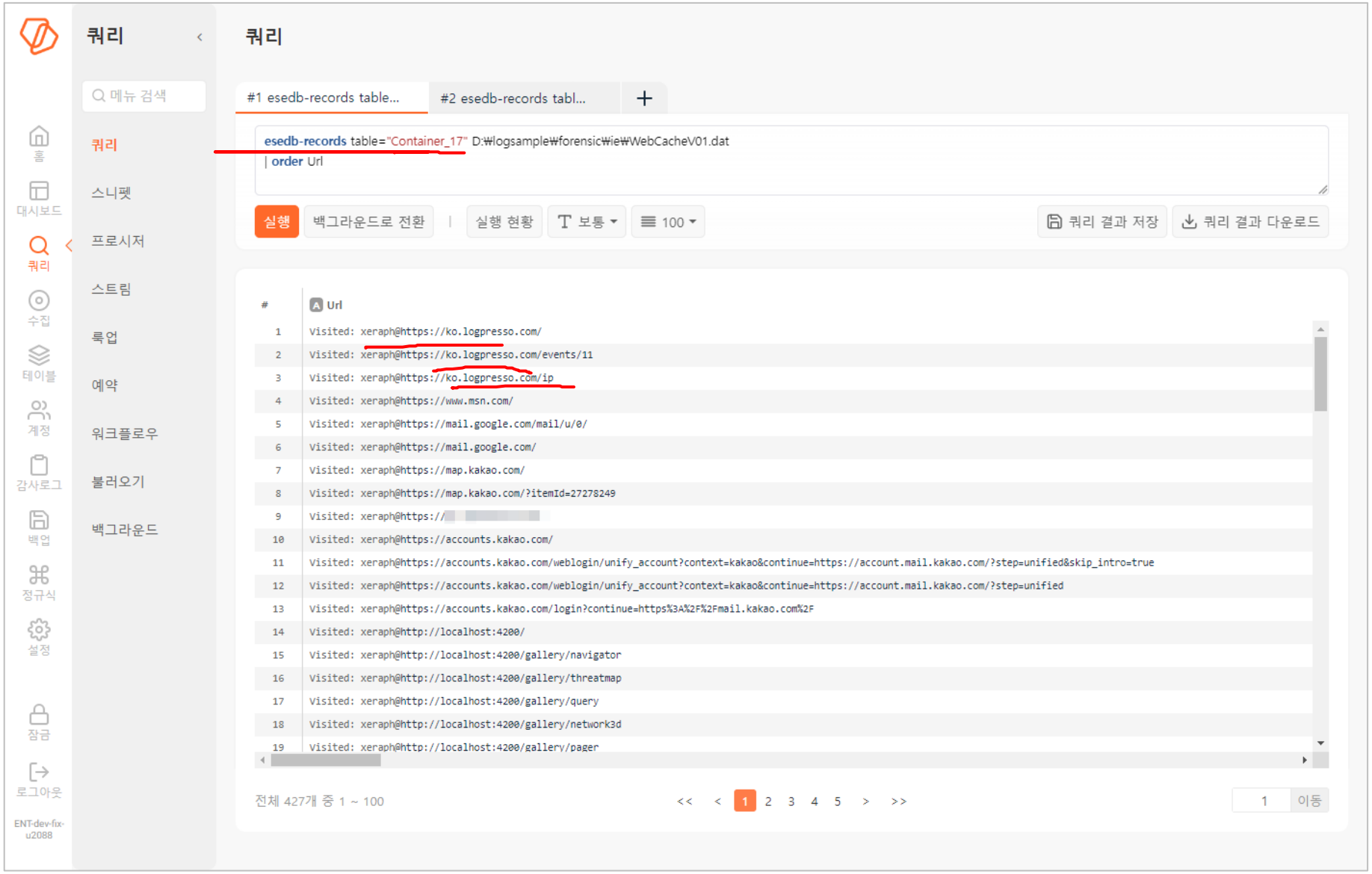
```
esedb-records table="Containers" D:₩logsample₩forensic₩ie₩WebCacheV01.dat | search Name == "History"
```

실행 | 백그라운드로 전환 | 실행 현황 | T 보통 | ≡ 100 | 쿼리 결과 저장 | 쿼리 결과 다운로드

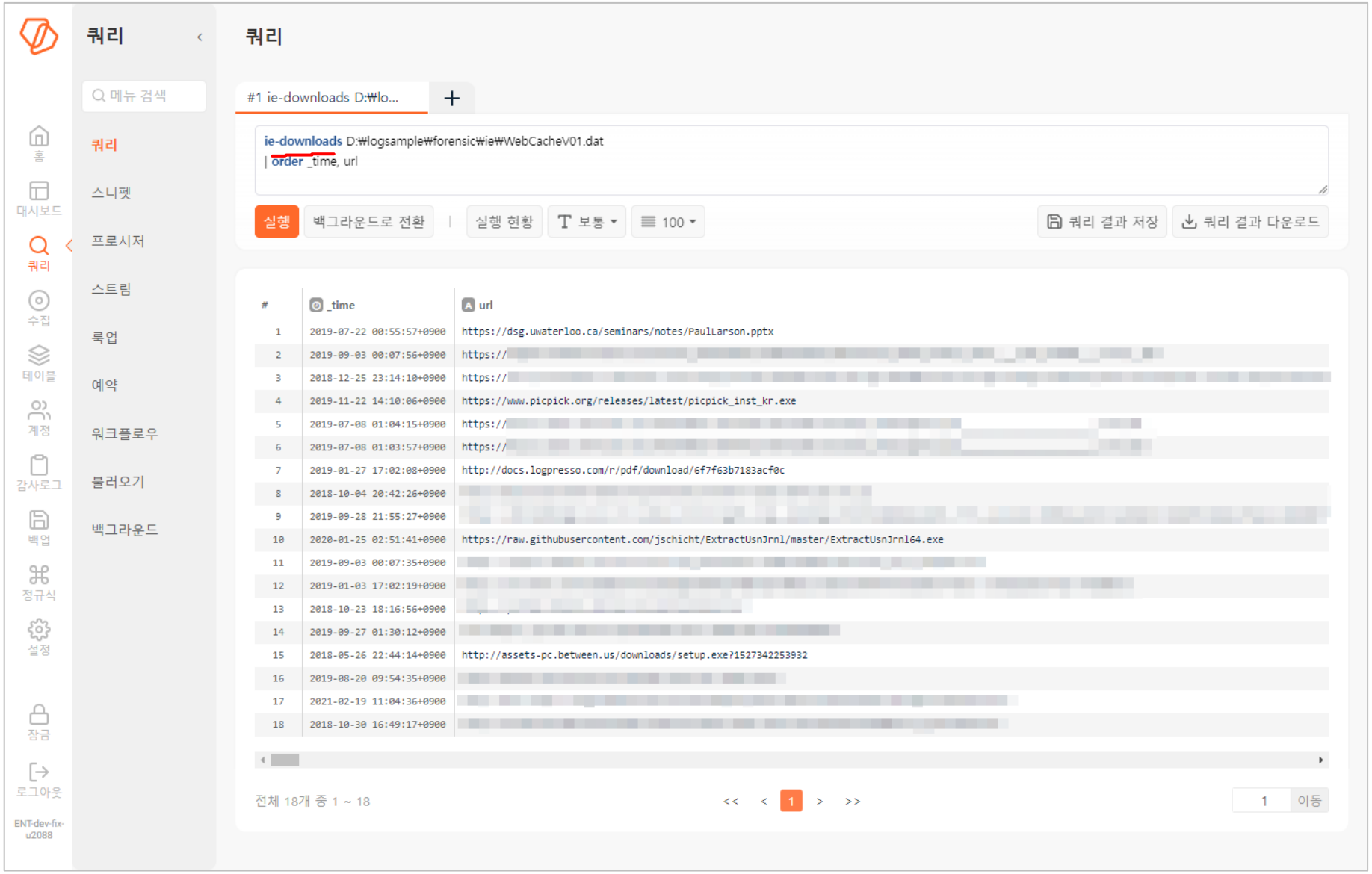
#	_file	_table	ContainerId	Directory	EntryMaxAge
1	WebCacheV01.dat	Containers	1	C:\Users\xeraph.HQ\AppData\Local\Microsoft\Windows\History\History.IE5\	0
2	WebCacheV01.dat	Containers	17	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\MicrosoftEdge\History\	0
3	WebCacheV01.dat	Containers	18	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!001\MicrosoftEdge\History\	0
4	WebCacheV01.dat	Containers	42	C:\Users\xeraph.HQ\AppData\Local\Microsoft\Windows\History\Low\History.IE5\	0
5	WebCacheV01.dat	Containers	47	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!121\MicrosoftEdge\History\	0
6	WebCacheV01.dat	Containers	58	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!002\MicrosoftEdge\History\	0
7	WebCacheV01.dat	Containers	298	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!003\MicrosoftEdge\History\	0
8	WebCacheV01.dat	Containers	475	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!005\MicrosoftEdge\History\	0
9	WebCacheV01.dat	Containers	517	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.windows.authhost.a_8wekyb3d8bbwe\AC\INetHistory\	0
10	WebCacheV01.dat	Containers	997	C:\Users\xeraph.HQ\AppData\Local\Packages\windows_ie_ac_001\AC\INetHistory\	0
11	WebCacheV01.dat	Containers	1254	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!006\MicrosoftEdge\History\	0
12	WebCacheV01.dat	Containers	2000	C:\Users\xeraph.HQ\AppData\Local\Packages\microsoft.microsoftedge_8wekyb3d8bbwe\AC\#!004\MicrosoftEdge\History\	0
13	WebCacheV01.dat	Containers	2360	C:\Users\xeraph.HQ\AppData\Local\Packages\Microsoft.DesktopAppInstaller_8wekyb3d8bbwe\AC\INetHistory\History.IE5\	0

전체 13개 중 1 ~ 13 << < 1 > >> 1 이동

ESE DB - IE 웹사이트 방문 기록 조회



IE 다운로드 기록 조회



#	_time	url
1	2019-07-22 00:55:57+0900	https://dsg.uwaterloo.ca/seminars/notes/PaulLarson.pptx
2	2019-09-03 00:07:56+0900	https://
3	2018-12-25 23:14:10+0900	https://
4	2019-11-22 14:10:06+0900	https://www.picpick.org/releases/latest/picpick_inst_kr.exe
5	2019-07-08 01:04:15+0900	https://
6	2019-07-08 01:03:57+0900	https://
7	2019-01-27 17:02:08+0900	http://docs.logpresso.com/r/pdf/download/6f7f63b7183acf0c
8	2018-10-04 20:42:26+0900	
9	2019-09-28 21:55:27+0900	
10	2020-01-25 02:51:41+0900	https://raw.githubusercontent.com/jschicht/ExtractUsnJrnl/master/ExtractUsnJrnl64.exe
11	2019-09-03 00:07:35+0900	
12	2019-01-03 17:02:19+0900	
13	2018-10-23 18:16:56+0900	
14	2019-09-27 01:30:12+0900	
15	2018-05-26 22:44:14+0900	http://assets-pc.between.us/downloads/setup.exe?1527342253932
16	2019-08-20 09:54:35+0900	
17	2021-02-19 11:04:36+0900	
18	2018-10-30 16:49:17+0900	

SQLITE

초소형 SQL 데이터베이스 엔진

Dwayne Richard Hipp

- 인포믹스 부팅이 제대로 안 되어서 접속이 안 되는데 본인에게 전화오는게 짜증나서 DB를 만든 사람 (...)
- 파서 생성기 직접 개발 (Lemon)
- 형상관리 도구 직접 개발 (Fossil)
- 프로젝트 관리 도구 직접 개발 (CVSTrac)

57 events for the month beginning 2022-06-01 by user drh

2022-06-18

10:26 In the --query-invariants option of fuzzcheck, correctly deal with OOMs causing the return value of sqlite3_column_name() to be NULL. (**Leaf** check-in: eabbee4a user: drh tags: trunk)

2022-06-17

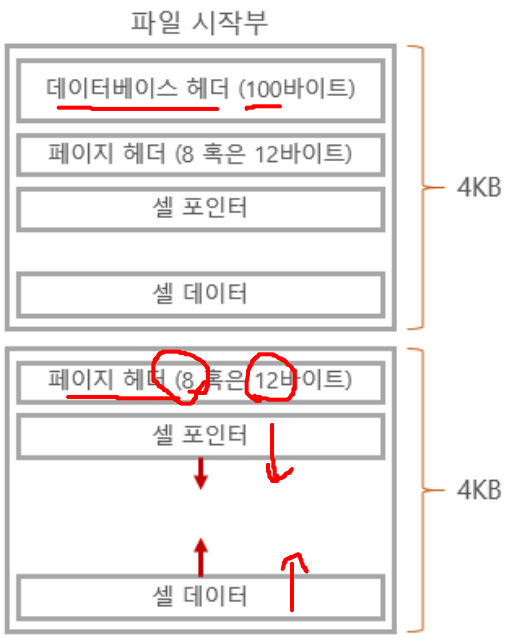
21:31 Fix the OP_Concat operator such that when concatenating a BLOB with an odd number of bytes on a database that is UTF16, the size of the resulting string is reduced to a multiple of two. (check-in: 5eb2c236 user: drh tags: trunk)

17:11 Omit the --query-invariants processing in fuzzcheck for queries that contain the implies_nonnull_row() test function. (check-in: 0602a084 user: drh tags: trunk)

https://changelog.com/podcast/201

SQLite 파일 구조

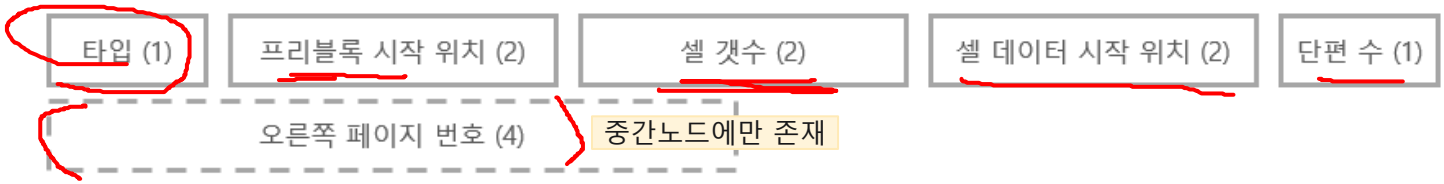
DB는 B+트리 페이지로 나누어지고, 파일 최초 100바이트는 데이터베이스 헤더 부분입니다.



오프셋	크기	설명
0	16	파일 식별용 헤더 SQLite format 3
16	2	페이지 크기 (단, 64K는 1로 표현)
18	1	파일 포맷 쓰기 버전 - 레거시 1, WAL 2
19	1	파일 포맷 읽기 버전 - 레거시 1, WAL 2
20	1	예약된 공간 (reserved) 크기 - 암호화 시 확장
21	1	최대 페이로드 분할 수 - 64 고정 (오버플로우 계산에 사용)
22	1	최소 페이로드 분할 수 - 32 고정 (오버플로우 계산에 사용)
23	1	리프 페이로드 분할 수 - 32 고정
24	4	파일 변경 카운터 - 캐시 무효화 처리에 활용
28	4	페이지 갯수
32	4	첫번째 프리 리스트 페이지 번호
36	4	전체 프리 리스트 페이지 번호
40	4	스키마 쿠키 - 스키마 변경 시 증가, 변경할 때마다 Prepared Statement 다시 컴파일
44	4	스키마 포맷 번호 • 1 (>= 3.0.0) 2004-06-18 배포 • 2 (>= 3.1.3) ADD COLUMN 지원하면서 가변 컬럼 갯수 지원 추가됨 (2005-02-20) • 3 (>= 3.1.4) ADD COLUMN 시 기본 값 정의 지원 추가됨 (2005-03-11) • 4 (>= 3.3.0) 직렬화 시 불린 타입 추가됨, 인덱스 정의 시 DESC 키워드 지원 (2006)
48	4	기본 페이지 캐시 크기
52	4	VACUUM 시 최대 페이지 번호
56	4	텍스트 인코딩: UTF-8 (1), UTF-16 (2), UTF-16BE (3)
60	4	user_version
64	4	증분 VACUUM 모드 (0 아닌 값이면 활성화)
68	4	응용 프로그램 ID
72	20	예약된 헤더 공간
92	4	version-valid-for
96	4	SQLITE 버전, 예를 들어 정수 값 3032001은 3.32.1 에 해당

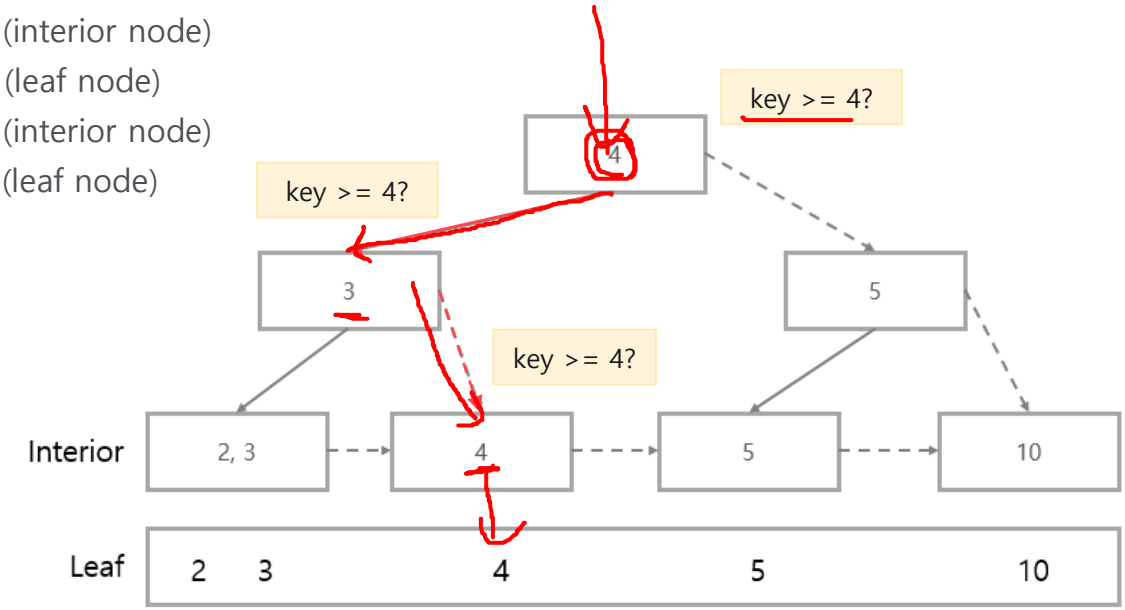
SQLite 페이지 헤더

DB는 B+트리 페이지로 나누어지고, 파일 최초 100바이트는 데이터베이스 헤더 부분입니다.



타입

- 0x05 - 테이블 중간 노드 (interior node)
- 0x0d - 테이블 말단 노드 (leaf node)
- 0x02 - 인덱스 중간 노드 (interior node)
- 0x0a - 인덱스 말단 노드 (leaf node)



SQLite 페이지 헤더

100바이트 (0x64) 데이터베이스 헤더 뒤에 페이지 헤더가 이어지는 것을 확인할 수 있습니다.

History

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	파일 시그니처
00000000	53	51	4C	69	74	65	20	66	6F	72	6D	61	74	20	33	00	SQLite format 3.
00000010	10	00	01	01	00	40	20	20	00	00	AD	B1	00	00	1B	E7	.....@ ...±...ç
00000020	00	00	0E	DF	00	00	00	05	00	00	00	43	00	00	00	04	...ß.....C....
00000030	00	00	00	00	00	00	00	00	00	00	00	01	00	00	00	00	.....
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	AD	B1	.....±
00000060	00	2E	43	C1	05	00	00	00	01	0F	FB	00	00	00	00	A1	..CÁ.....û....¡
00000070	0F	FB	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.û.....
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....

<파일 헤더와 첫번째 페이지 헤더>

페이지 크기
4096 바이트

테이블
중간 노드 타입

셀 포인터
1개

오른쪽 자식
페이지 번호

셀 포인터
4091 (0xFFB)

SQLite 셀 구조

페이지 시작 주소 + 셀 포인터 값이 셀 데이터의 주소가 됩니다.

테이블 중간 노드 레이아웃

크기	설명
4	왼쪽 자식 페이지 번호
VarInt	왼쪽 자식 페이지의 최대 Row ID

테이블 말단 노드 레이아웃

크기	설명
VarInt	페이로드 크기 오버플로우 포함한 전체 크기이며, 현재 페이지의 크기는 별도 수식으로 계산해야 함
VarInt	Row ID
지정 길이	페이로드
4	오버플로우 페이지 번호

```
00000FE0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .................
00000FF0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 3E 0A  ............>.
00001000  0D 0F F2 00 04 0F 7E 00 0F D4 0F 8E 0F 7E 0F A2  ..ò...~..Ô.Ž.~.¢
00001010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .................
```

왼쪽 자식 페이지 번호 62 (0x3E)

왼쪽 자식 최대 Row ID 10 (0x0A)

말단 노드 타입

셀 포인터 8개

3992

3951

```
0003D000  0D 0B 1C 00 08 07 30 00 0F 98 0F 6F 07 30 0E AA  ......0.˜.o.0.ª
0003D010  0E 61 0D 33 0B E7 0A AC 0A AC 0A AC 09 F1 09 A8  .a.3.ç.¬.¬.¬.ñ.¨
0003D020  09 57 08 FD 08 A3 08 54 07 FF 07 69 06 F9 06 61  .W.ý.£.T.ÿ.i.ù.a
0003D030  05 D9 05 53 04 8C 04 4B 03 B3 00 55 00 00 00 00  .Ù.S.Œ.K.³.U....
0003D040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .................
```

첫번째 셀에 해당하는 0x3DF98 (0x3D000 + 0xF98) 로 이동

SQLite 페이로드 구조

데이터 직렬화 타입

타입 번호	길이	설명
0	0	NULL
1	1	1바이트로 인코딩된 정수
2	2	2바이트로 인코딩된 정수
3	3	3바이트로 인코딩된 정수
4	4	4바이트로 인코딩된 정수
5	6	6바이트로 인코딩된 정수
6	8	8바이트로 인코딩된 정수
8	0	정수 값 0 (불린 표현용)
9	0	정수 값 1 (불린 표현용)
12>= (짝수)	(N-12)/2	바이너리 배열
13>= (홀수)	(N-13)/2	문자열 (데이터베이스 헤더에 지정된 인코딩으로 해석)

페이로드 길이 102 (0x66)

Row ID 1

헤더 길이 7

문자열 (길이 5), 문자열 (길이 4), 문자열 (길이 4), 정수, 문자열 (길이 81) 타입

```
0003DF90  61 5F 31 6D 65 74 61 03 66 01 07 17 15 15 01 81  a_1meta.f.......
0003DFA0  2F 74 61 62 6C 65 6D 65 74 61 6D 65 74 61 02 43  /tablemetameta.C
0003DFB0  52 45 41 54 45 20 54 41 42 4C 45 20 6D 65 74 61  REATE TABLE meta
0003DFC0  28 6B 65 79 20 4C 4F 4E 47 56 41 52 43 48 41 52  (key LONGVARCHAR
0003DFD0  20 4E 4F 54 20 4E 55 4C 4C 20 55 4E 49 51 55 45   NOT NULL UNIQUE
0003DFE0  20 50 52 49 4D 41 52 59 20 4B 45 59 2C 20 76 61   PRIMARY KEY, va
0003DFF0  6C 75 65 20 4C 4F 4E 47 56 41 52 43 48 41 52 29  lue LONGVARCHAR)
0003E000  0D 00 00 00 0D 01 4A 00 0F 56 0B C7 0B B3 0B 09  ......J..V.Ç.³..
```

<페이지 62, 첫번째 레코드에 해당되는 meta 테이블 스키마>

루트 페이지 번호
파일 오프셋 (2-1) x 4096

SQLite 오버플로우 처리

페이지보다 큰 데이터는 대부분을 오버플로우 페이지에 기록하고,
현재 페이지에는 데이터 앞부분과 오버플로우 페이지 번호를 기록합니다.

테이블 말단 노드

- 가용 크기 (usable size) = 페이지 크기 - 예약 크기
  (암호화 확장되지 않으면 예약 크기가 0이므로, 대부분 페이지 크기)
- 최소 크기 (min local) = (가용 크기 - 12) * 32 / 255 - 23 = 4K의 경우 489가 됨
- 최대 크기 (max local) = 가용 크기 - 35 = 4K의 경우 4061 이 됨
  - 페이로드 크기가 이 최대 크기 이하이면 전체를 기록 가능함
- 기록 크기 (surplus) = 최소 크기 + (페이로드 크기 - 최소 크기) % (가용 크기 - 4)
  - 가용 크기 - 4가 되는 것은 오버플로우 페이지에서 첫 4바이트가 다음 오버플로우 번호를 기록하여 링크하기 때문
  - 그런데 이 기록 크기가 최대 크기를 초과하면 최소 크기를 현재 페이지에 기록하는 것으로 결정됨

테이블 중간 노드

- 모든 테이블 키 (Row ID)는 정수형이므로 절대로 오버플로우 되지 않음

SQLite 오버플로우 처리

페이지보다 큰 데이터는 대부분을 오버플로우 페이지에 기록하고,
현재 페이지에는 데이터 앞부분과 오버플로우 페이지 번호를 기록합니다.

```
0010C000  0D 00 00 00 0A 00 1F 00 0F 6B 0F 21 0E D5 05 C2  ........k.!.Õ.Â
0010C010  04 6C 01 6A 01 1E 00 D2 00 86 00 1F 00 00 00 64  .l.j...Ò.†.....d
0010C020  91 25 05 02 08 81 47 07 17 68 74 74 70 3A 2F 2F  '%....G..http://
0010C030  62 75 69 6C 64 2F 6A 6F 62 2F 6C 6F 67 70 72 65  build/job/logpre
```

<테이블 말단 페이지 269의 4번째 셀 데이터는 0x10C5C2>

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0010C5C0   34 37 D2 03 91 1F 06 02 08 81 A4 03 07 12 64 61  47Ò.'.....¤...da
0010C5D0   74 61 3A 69 6D 61 67 65 2F 6A 70 65 67 3B 62 61  ta:image/jpeg;ba
0010C5E0   73 65 36 34 2C 2F 39 6A 2F 34 41 41 51 53 6B 5A  se64,/9j/4AAQSkZ
```

- 페이로드 크기 0xD203 = 11010010 00000011 = 10499
- Row ID 0x911F = 10010001 00011111 = 2207
- 가용 크기 4096
- 최소 크기 489 = (4096 - 12) x 32 / 255 - 23
- 최대 크기 4061 = 4096 - 35
- 기록 크기 2315 = 489 + (10499 - 489) % (4096 - 4)
  - 최대 크기 넘지 않으므로 2315가 이 페이지에 분할되어 기록하는 페이로드 크기
- 노란 박스 헤더부
  - 2 = 2바이트 정수 => 0x0712 = 1810
  - 8 = 불린 false
  - 20995 = 10000001 10100100 00000011 = 문자열 값은 data:image/jpeg 로 시작

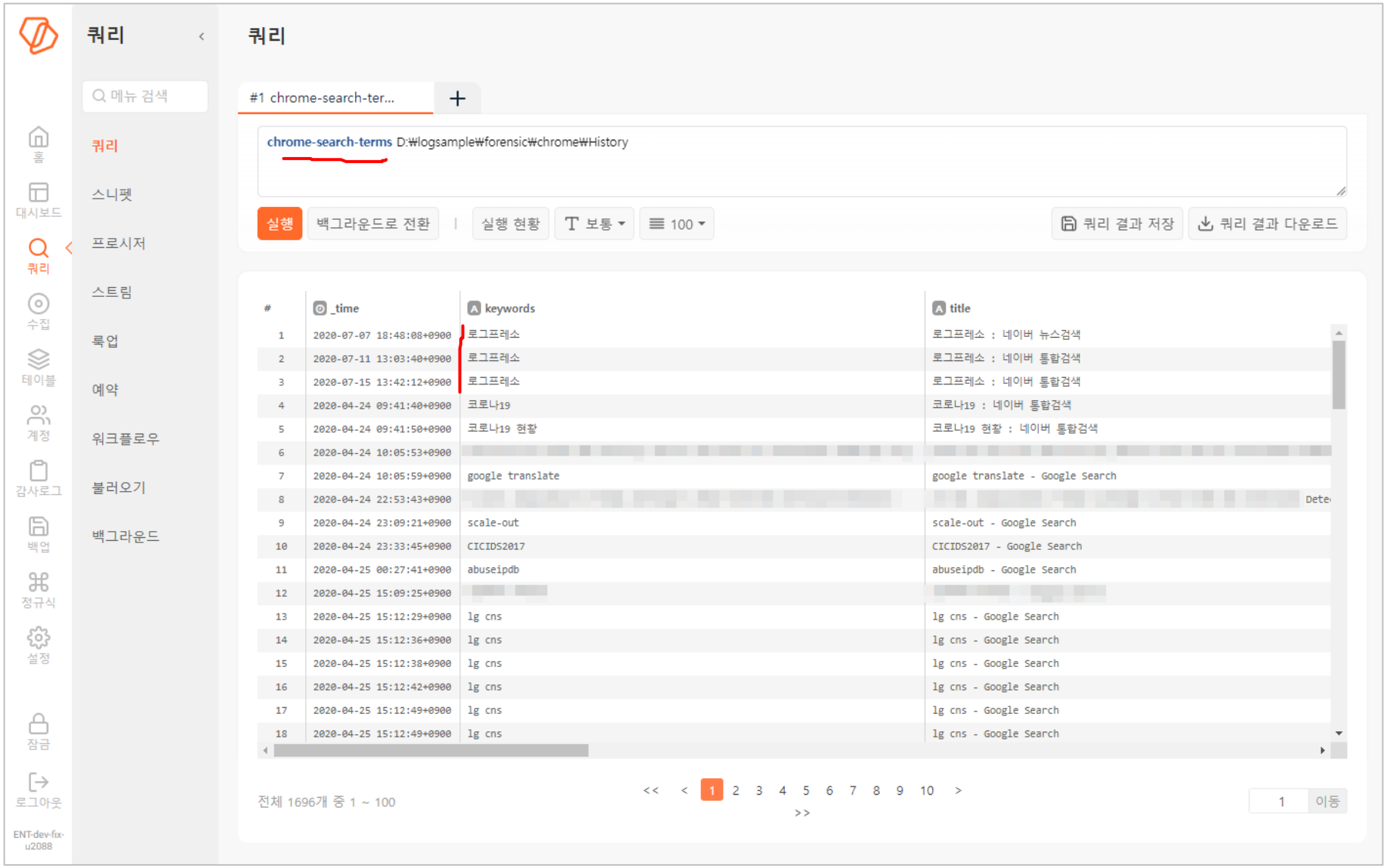
SQLite 스키마 정의

1페이지부터 페이지를 순회하면 스키마 레코드를 조회할 수 있습니다.

컬럼	설명
type	table 혹은 index
name	테이블 이름 혹은 인덱스 이름
table_name	테이블 이름
root_page	루트 페이지 번호
sql	테이블이나 인덱스에 대한 DDL

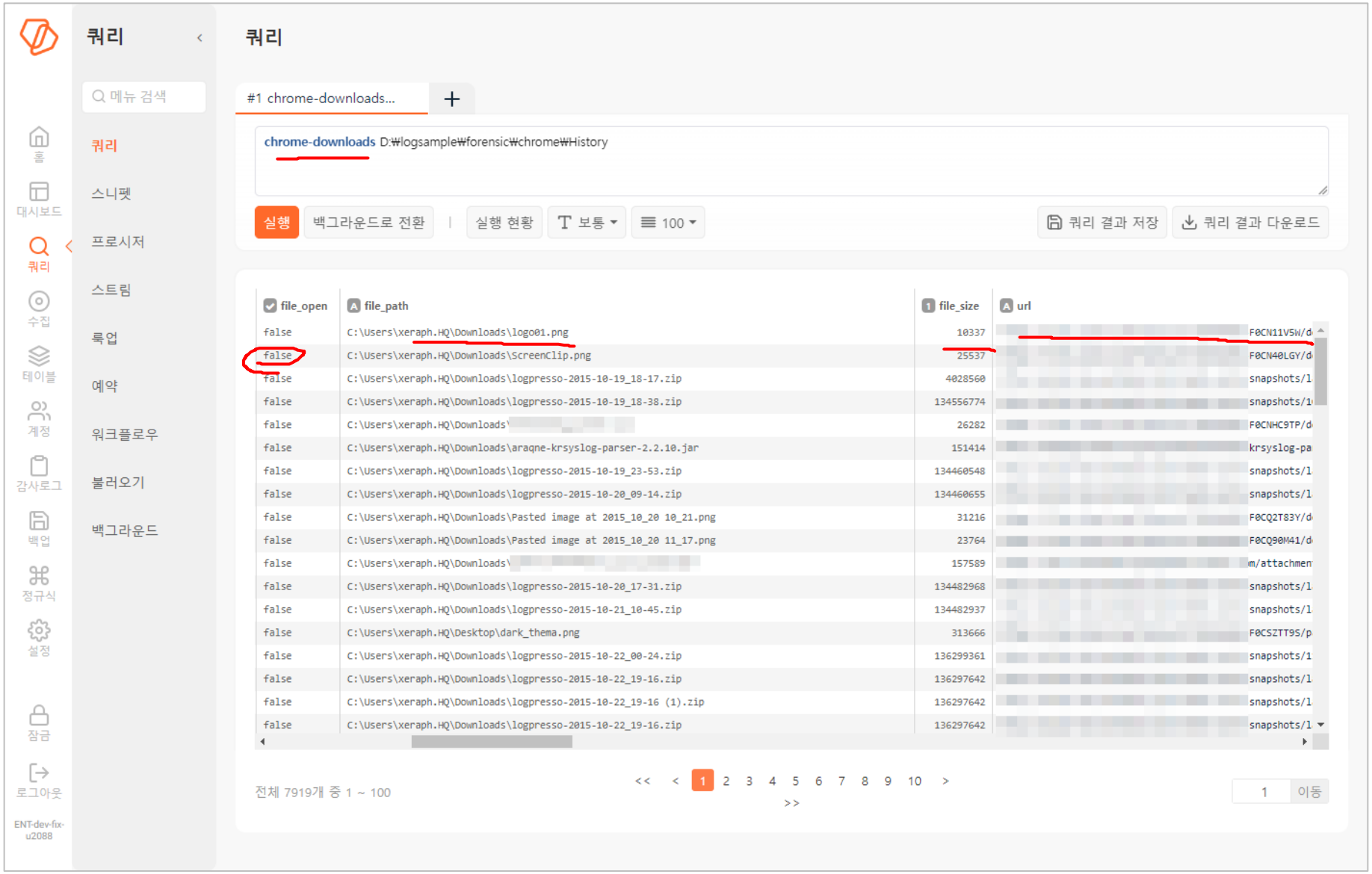
<sqlite_schema 테이블의 스키마>

응용 사례 - 크롬 검색어 조회



#	_time	keywords	title
1	2020-07-07 18:48:08+0900	로그프레소	로그프레소 : 네이버 뉴스검색
2	2020-07-11 13:03:40+0900	로그프레소	로그프레소 : 네이버 통합검색
3	2020-07-15 13:42:12+0900	로그프레소	로그프레소 : 네이버 통합검색
4	2020-04-24 09:41:40+0900	코로나19	코로나19 : 네이버 통합검색
5	2020-04-24 09:41:50+0900	코로나19 현황	코로나19 현황 : 네이버 통합검색
6	2020-04-24 10:05:53+0900		
7	2020-04-24 10:05:59+0900	google translate	google translate - Google Search
8	2020-04-24 22:53:43+0900		Dete
9	2020-04-24 23:09:21+0900	scale-out	scale-out - Google Search
10	2020-04-24 23:33:45+0900	CICIDS2017	CICIDS2017 - Google Search
11	2020-04-25 00:27:41+0900	abuseipdb	abuseipdb - Google Search
12	2020-04-25 15:09:25+0900		
13	2020-04-25 15:12:29+0900	lg cns	lg cns - Google Search
14	2020-04-25 15:12:36+0900	lg cns	lg cns - Google Search
15	2020-04-25 15:12:38+0900	lg cns	lg cns - Google Search
16	2020-04-25 15:12:42+0900	lg cns	lg cns - Google Search
17	2020-04-25 15:12:49+0900	lg cns	lg cns - Google Search
18	2020-04-25 15:12:49+0900	lg cns	lg cns - Google Search

응용 사례 - 크롬 다운로드 이력 조회



file_open	file_path	file_size	url
false	C:\Users\xeraph.HQ\Downloads\logo01.png	10337	F0CN11V5W/d
false	C:\Users\xeraph.HQ\Downloads\ScreenClip.png	25537	F0CN40LGY/d
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-19_18-17.zip	4028560	snapshots/l
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-19_18-38.zip	134556774	snapshots/1
false	C:\Users\xeraph.HQ\Downloads\	26282	F0CNHC9TP/d
false	C:\Users\xeraph.HQ\Downloads\araqne-krsyslog-parser-2.2.10.jar	151414	krsyslog-pa
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-19_23-53.zip	134460548	snapshots/l
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-20_09-14.zip	134460655	snapshots/l
false	C:\Users\xeraph.HQ\Downloads\Pasted image at 2015_10_20 10_21.png	31216	F0CQ2T83Y/d
false	C:\Users\xeraph.HQ\Downloads\Pasted image at 2015_10_20 11_17.png	23764	F0CQ90M41/d
false	C:\Users\xeraph.HQ\Downloads\	157589	m/attachmen
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-20_17-31.zip	134482968	snapshots/l
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-21_10-45.zip	134482937	snapshots/l
false	C:\Users\xeraph.HQ\Desktop\dark_thema.png	313666	F0CSZTT9S/p
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-22_00-24.zip	136299361	snapshots/1
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-22_19-16.zip	136297642	snapshots/l
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-22_19-16 (1).zip	136297642	snapshots/l
false	C:\Users\xeraph.HQ\Downloads\logpresso-2015-10-22_19-16.zip	136297642	snapshots/l

인재를 모십니다

로그프레소 채용 공고 페이지
https://career.logpresso.com/

- 데이터베이스 엔진 개발에 관심있는 분
  - 스토리지 및 인덱스 엔진 개발
  - 쿼리 엔진 및 최적화 플래너 개발

- 디지털 포렌식 툴 개발에 관심있는 분
  - 아티팩트 추출부터 자동 침해 분석까지 개발
  - 보안 기반 지식을 이용한 전문적 QA

Thank you

서울특별시 마포구 새창로 7 SNU장학빌딩 16층 (도화동 565)
02-6730-7249 contact@logpresso.com