

[암호화 테이블의 컬럼에 걸린 인덱스값 확인]

-- (0) dbcc 결과 메시지 화면 출력

```
DBCC TRACEON (3604)
GO
```

-- (1) 테이블생성

```
CREATE TABLE MEMBER_TABLE (
  ID          INT,
  JUMIN       VARCHAR(50),
  AGE         INT,
  [NAME]      VARCHAR(100)
)
GO
```

-- (2) JUMIN 컬럼 에 non클러스터 인덱스 생성

```
CREATE NONCLUSTERED INDEX NL_JUMIN ON MEMBER_TABLE (JUMIN)
GO
```

-- (3) 입력

```
INSERT INTO MEMBER_TABLE VALUES (1, '7A0A0A8A0A1A-A1A1A2A3A4A5A6A', 50, '홍길동')
INSERT INTO MEMBER_TABLE VALUES (2, '7A0A0A8A0A1A-A1A1A2A3A4A5A7A', 20, '강감찬')
INSERT INTO MEMBER_TABLE VALUES (3, '7A0A0A8A0A1A-A1A1A2A3A4A5A8A', 30, '유관순')
INSERT INTO MEMBER_TABLE VALUES (4, '7A0A0A8A0A1A-A1A1A2A3A4A5A9A', 40, '이순신')
GO
```

-- (4) 조회

```
SELECT * FROM MEMBER_TABLE
```

> 두번째 컬럼 ("JUMIN") 값이 , 암호화 된 값으로 입력 되었다고 가정 합니다.

결과		메시지		
	ID	JUMIN	AGE	NAME
1	1	7A0A0A8A0A1A-A1A1A2A3A4A5A6A	50	홍길동
2	2	7A0A0A8A0A1A-A1A1A2A3A4A5A7A	20	강감찬
3	3	7A0A0A8A0A1A-A1A1A2A3A4A5A8A	30	유관순
4	4	7A0A0A8A0A1A-A1A1A2A3A4A5A9A	40	이순신

-- (5) 테이블 ("MEMBER_TABLE") 에 걸린 인덱스 조회

```
SELECT object_name (ID) AS "오브젝트_이름 (TABLE,VIEW)" , indid as 인덱스_아이디, name as 인덱스_이름
FROM sysindexes WHERE id = object_id('MEMBER_TABLE')
```

결과		메시지	
	오브젝트_이름 (TABLE,VIEW)	인덱스_아이디	인덱스_이름
1	MEMBER_TABLE	0	NULL
2	MEMBER_TABLE	2	NL_JUMIN

> 위 Step (2) 번에서 생성한 인덱스 ("NL_JUMIN") 확인

-- (6) 인덱스 ("NL_JUMIN") 에 물리적 저장정보 확인

```
dbcc ind (master , 'MEMBER_TABLE' , 2)
```

go

- > 인자 1 : 오브젝트가 포함된 데이터베이스 명
- > 인자 2 : 테이블 명 (OR 뷰 명)
- > 인자 3 : 인덱스 아이디 (값 0 은 항상 제외 시킴 - Heap)

결과

메시지

	PageFID	PagePID	IAMFID	IAMPID	ObjectID
1	1	370	NULL	NULL	343672272
2	1 \	364 /	1	370	343672272

-- (7) INDEX : NL_JUMIN 저장 구조 조회

```
dbcc page (master , 1 , 364 , 3)
```

go

결과

메시지

	Field	PageId	Row	Level	JUMIN (key)	HEAP RID (key)	KeyHashValue
1	1	364	0	0	7A0A0A8A0A1A-A1A1A2A3A4A5A6A	0x4801000001000000	(a601b31cae30)
2	1	364	1	0	7A0A0A8A0A1A-A1A1A2A3A4A5A7A	0x4801000001000100	(a601cc4677c6)
3	1	364	2	0	7A0A0A8A0A1A-A1A1A2A3A4A5A8A	0x4801000001000200	(a601824e511c)
4	1	364	3	0	7A0A0A8A0A1A-A1A1A2A3A4A5A9A	0x4801000001000300	(a601fd1488ea)

> 테이블의 컬럼 값 그대로 암호화 되어 INDEX 가 저장된 상태로 확인됨

- > 인자 1 : 오브젝트가 포함된 데이터베이스 명
- > 인자 2 : 오브젝트가 포함된 파일 ID
- > 인자 3 : 오브젝트가 포함된 페이지 ID (IAMPID 가 NULL 이 아닌 행의 PagePID 를 선택)
- > 인자 4 : 조회 옵션 - 3 (컬럼형태로 보기)

[스키마 바인딩 뷰 에 걸린 인덱스값 확인]

SchemaBinding View (스키마 바인딩 뷰)

테이블 등을 대상으로 뷰를 만들며 , 보여지는 값 (평문) 에 대하여 인덱스를 생성 가능
따라서 뷰를 통한 조회 시에 , 복호화 과정 없이 평문 인덱스를 통하여 고속 조회 가능

생성시에 다수의 제약 사항들이 있음.

-- (1) 복호화 평선으로 가정한, 평선 생성

```
CREATE FUNCTION DBO.DEC_JUMIN (@JUMIN VARCHAR(100))
RETURNS VARCHAR(100)
WITH SCHEMABINDING
AS
BEGIN
    RETURN ( REPLACE (@JUMIN , 'A' , '' ) )
END
GO
```

> 기능 : 단순히 문자열에 포함된 "A" 를 공백으로 대체함

-- (2) "MEMBER_TABLE" 테이블에 대하여 , SCHEMABINDING VIEW 생성

```
CREATE VIEW DBO.SCHEMABINDING_VIEW_MEMBER_TABLE
WITH SCHEMABINDING
AS
SELECT ID, dbo.DEC_JUMIN (JUMIN) AS JUMIN , AGE , NAME FROM DBO.MEMBER_TABLE
GO
```

> 앞서 생성한 테이블 ("MEMBER_TABLE") 에 대해서 스키마 바인딩 뷰 생성

> 기능 : 주민번호는 복호화 평선 ("dbo.DEC_JUMIN") 을 통하여 평문 형태로 보임

-- (3) 테이블 및 스키마바인딩뷰 조회

(a) 테이블 조회

결과		메시지		
	ID	JUMIN	AGE	NAME
1	1	7A0A0A8A0A1A-A1A1A2A3A4A5A6A	50	홍길동
2	2	7A0A0A8A0A1A-A1A1A2A3A4A5A7A	20	강감찬
3	3	7A0A0A8A0A1A-A1A1A2A3A4A5A8A	30	유관순
4	4	7A0A0A8A0A1A-A1A1A2A3A4A5A9A	40	이순신

(b) 스키마 바인딩 뷰 조회

결과		메시지		
	ID	JUMIN	AGE	NAME
1	1	700801-1123456	50	홍길동
2	2	700801-1123457	20	강감찬
3	3	700801-1123458	30	유관순
4	4	700801-1123459	40	이순신

-- (4) 스키마바인딩뷰의 주민번호(평균) 컬럼에 인덱스생성

```
CREATE UNIQUE CLUSTERED INDEX SCHEMA_INDEX ON SCHEMABINDING_VIEW_MEMBER_TABLE (JUMIN)
GO
```

-- (5) 스키마바인딩뷰("SCHEMABINDING_VIEW_MEMBER_TABLE") 에 걸린 인덱스조회

```
SELECT object_name (ID) AS "오브젝트_이름(TABLE,VIEW)" , indid as 인덱스_아이디, name as 인덱스_이름
FROM sysindexes WHERE id = object_id('SCHEMABINDING_VIEW_MEMBER_TABLE')
```

결과		메시지	
	오브젝트_이름(TABLE,VIEW)	인덱스_아이디	인덱스_이름
1	SCHEMABINDING_VIEW_MEMBER_TABLE	1	SCHEMA_INDEX

-- (6) 인덱스 ("SCHEMA_INDEX"에 물리적 저장정보 확인

```
dbcc ind (master , 'SCHEMABINDING_VIEW_MEMBER_TABLE', 1)
GO
```

- > 인자 1 : 오브젝트가 포함된 데이터베이스 명
- > 인자 2 : 테이블 명 (OR 뷰 명)
- > 인자 3 : 인덱스 아이디 (값 0 은 항상 제외 시킴 - Heap)

결과 메시지

	PageFID	PagePID	IAMFID	IAMPID	ObjectID
1	1	468	NULL	NULL	375672386
2	1	411	1	468	375672386

-- (7) INDEX : SCHEMA_INDEX 저장 구조 조회

```
dbcc page (master , 1 , 411 , 3)
```

결과 메시지

	Field	PageId	Row	Level	JUMIN (key)	HEAP RID (key)	KeyHashValue
1	1	364	0	0	7A0A0A8A0A1A-A1A1A2A3A4A5A6A	0x4801000001000000	(a601b31cae30)
2	1	364	1	0	7A0A0A8A0A1A-A1A1A2A3A4A5A7A	0x4801000001000100	(a601cc4677c6)
3	1	364	2	0	7A0A0A8A0A1A-A1A1A2A3A4A5A8A	0x4801000001000200	(a601824e511c)
4	1	364	3	0	7A0A0A8A0A1A-A1A1A2A3A4A5A9A	0x4801000001000300	(a601fd1488ea)

> 테이블의 컬럼 값 그대로 암호화 되어 있는 상태로 확인됨

- > 인자 1 : 오브젝트가 포함된 데이터베이스 명
- > 인자 2 : 오브젝트가 포함된 파일 ID
- > 인자 3 : 오브젝트가 포함된 페이지 ID (IAMPID 가 NULL 이 아닌 행의 PagePID 를 선택)
- > 인자 4 : 조회 옵션 - 3 (컬럼형태로 보기)

메시지

```
00000000: 30000c00 03000000 1e000000 0400f002 +0.....
00000010: 00230029 00373030 3830312d 31313233 +.#.).700801-1123
00000020: 343538c0 afb0fcbe f8+++++++458.....
```

Slot 2 Column 0 Offset 0x15 Length 14

JUMIN = 700801-1123458

Slot 2 Column 1 Offset 0x4 Length 4

ID = 3

Slot 2 Column 2 Offset 0x8 Length 4

AGE = 30

Slot 2 Column 3 Offset 0x23 Length 6

NAME = 유관순

Slot 3 Offset 0xdb Length 41

Record Type = PRIMARY_RECORD

Record Attributes = NULL_BITMAP VARIABLE_COLUMNS

> 스키바 바인딩 뷰의 , "JUMIN" 컬럼에 걸린 인덱스가 평문 상태로 저장됨이 확인됨

-- (8)SCHEMABINDING 텍스트를 포함하는 VIEW / FUNCTION 이 있는지 확인

SELECT [TEXT] FROM SYS COMMENTS WHERE TEXT LIKE '%SCHEMABINDING%'

결과 메시지

	TEXT
1	CREATE FUNCTION DBO.DEC_JUMIN (@JUMIN VARCHAR(100)) RETURNS VARCHAR(100) WITH SCHEMABINDI...
2	CREATE VIEW DBO.SCHEMABINDING_VIEW_MEMBER_TABLE WITH SCHEMABINDING AS SELECT ID,db0.DE...
3	dex(type_name(system_type_id), @numtypes) > 0 then convert(char(5),OdbcScale(system_type_id,scale)) el...
4	ect @dynamic_snapshot_view_definition = replace(@dynamic_snapshot_view_definition collate SQL_Latin1_Ge...

> SYS COMMENTS 시스템 테이블은 각 데이터베이스 별로 존재 하며, 해당 테이블이 있는 데이터베이스로 이동 및 조회