

• DB구현 •

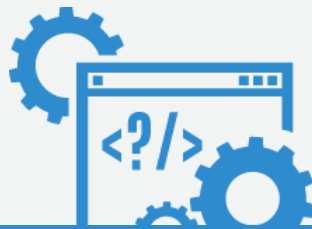
인덱스의 사용





학습내용

- ❖ 인덱스의 이해
- ❖ 인덱스의 관리
- ❖ 인덱스의 사용 효과



학습목표

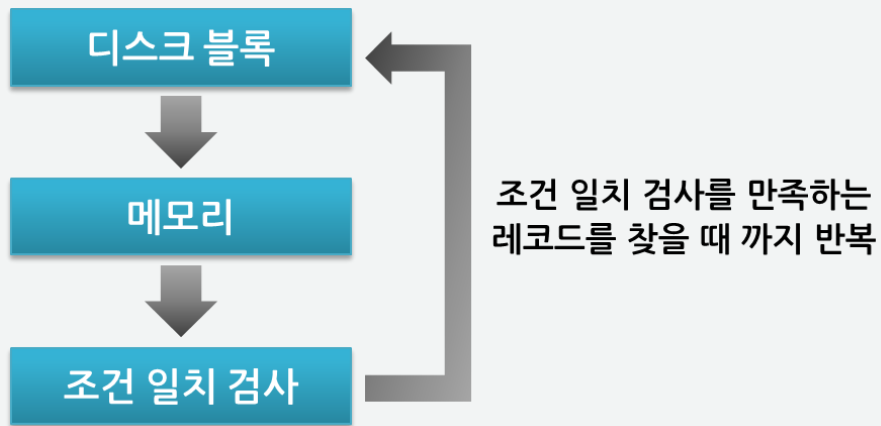
- ❖ 인덱스의 기능과 특징을 설명할 수 있다.
- ❖ 인덱스의 구조와 동작과정을 파악하고 DDL 구문을 사용하여 관리할 수 있다.
- ❖ MySQL의 인덱스의 효과를 입증할 수 있다.

인덱스의 이해

인덱스의 개념

▶ SELECT 질의 실행

- 조건을 만족하는 레코드를 탐색



학번	이름	소속학과
2012012	홍길동	컴퓨터과학
2012034	정도전	역사학
2013019	이순신	전기공학
2013030	장보고	컴퓨터과학
2013044	도철수	국문학
2014001	김영희	영문학
2014004	피천득	기계공학
2014017	안창호	물리학
2014055	소지섭	컴퓨터과학
2014225	차두리	기계공학

N개

인덱스의 이해

인덱스의 개념

▶ SELECT 질의 실행 예

학번	이름	소속학과
2012012	홍길동	컴퓨터과학
2012034	정도전	역사학
2013019	이순신	전기공학
2013030	장보고	컴퓨터과학
2013044	도철수	국문학
2014001	김영희	영문학
2014004	피천득	기계공학
2014017	안창호	물리학
2014055	소지섭	컴퓨터과학
2014225	차두리	기계공학

- “SELECT * FROM 학생 WHERE 학번 = ‘2014225’” 질의 실행
- 필요한 데이터 검색

- **최선** : 1 번
- **최악** : N 번
- **평균** : N/2 번

인덱스의 이해

인덱스의 개념

➤ 인덱스의 정의

인덱스란?

데이터베이스에 존재하는 많은 양의 데이터 중에 사용자가 원하는 특정 레코드만을 찾기 위해 사용하는 별도의 자료 구조

- “데이터 검색 시, 데이터베이스의 응답시간을 단축
- 순차 탐색 : 인덱스가 정의되지 않은 컬럼 검색
- InnoDB 엔진에서는 일반적으로 B+-트리 인덱스 적용

➤ 자동 생성 인덱스

- 제약 조건에 의해서 자동으로 생성되는 인덱스
- PRIMARY KEY, UNIQUE, FOREIGN KEY 제약 조건은 인덱스를 자동 생성

```
CREATE TABLE 강의 (  
    강의번호      INT          AUTO_INCREMENT PRIMARY KEY,  
    강의이름      VARCHAR(20) NOT NULL UNIQUE,  
    강의시간      VARCHAR(20) NOT NULL,  
    인원제한      INT          DEFAULT 50,  
    교수번호      CHAR(4)      NOT NULL,  
    FOREIGN KEY (작성자ID) REFERENCES 사용자(ID)  
)
```

인덱스의 이해

인덱스의 동작과정

인덱스의 사용

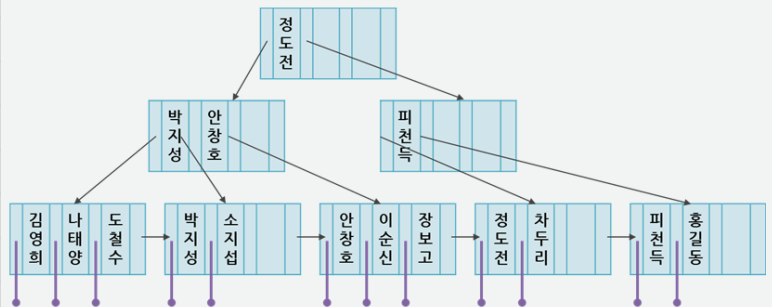
- DBMS는 인덱스를 통하여 레코드가 디스크 저장장치 또는 메모리의 어느 블록에 저장되어 있는지 파악하고, 해당 블록을 읽어 들임

검색기 파일에서 레코드를 찾는데 사용되는 컬럼이나 컬럼의 집합

인덱스의 사용

학번	이름	소속학과
2012012	홍길동	컴퓨터과학
2012034	정도전	역사학
2013019	이순신	전기공학
2013030	장보고	컴퓨터과학
2013044	도철수	국문학
2014001	김영희	영문학
2014004	피천득	기계공학
2014017	안창호	물리학
2014055	소지섭	컴퓨터과학
2014225	차두리	기계공학

⋮

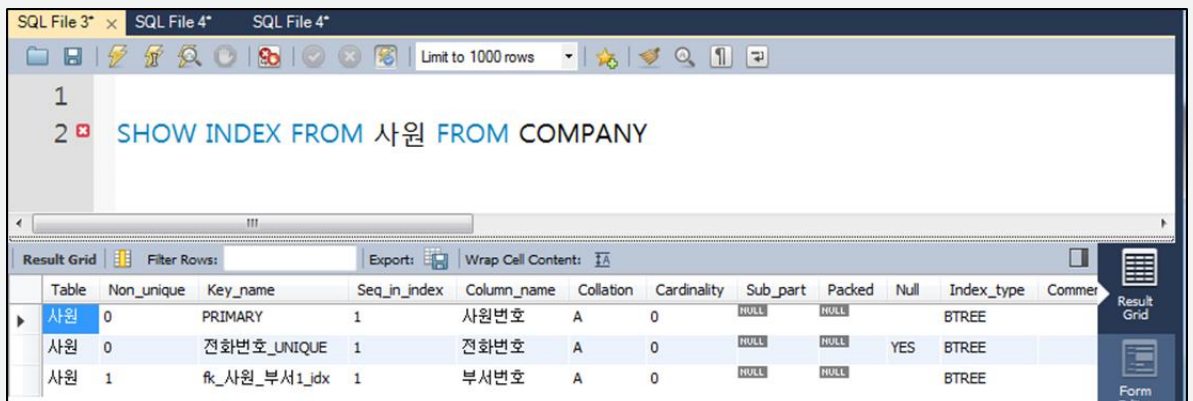


인덱스 관리

DDL 구문을 이용한 인덱스 관리

➤ 인덱스의 확인

SHOW INDEX FROM 테이블이름 [FROM 스키마이름]



SQL File 3* x SQL File 4* SQL File 4*

1
2 **SHOW INDEX FROM 사원 FROM COMPANY**

Result Grid

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment
▶ 사원	0	PRIMARY	1	사원번호	A	0	NULL	NULL		BTREE	
사원	0	전화번호_UNIQUE	1	전화번호	A	0	NULL	NULL	YES	BTREE	
사원	1	fk_사원_부서1_idx	1	부서번호	A	0	NULL	NULL		BTREE	

인덱스의 확인 예

➤ 인덱스 생성 구문 형식

CREATE INDEX 인덱스이름
ON 테이블이름 (컬럼명1, 컬럼명2, ..., 컬럼명n)

➤ 인덱스 생성 구문 예

CREATE INDEX idx_사원_1
ON 사원 (사원이름)

➤ 인덱스 삭제 구문 형식

DROP INDEX 인덱스이름 ON 테이블이름

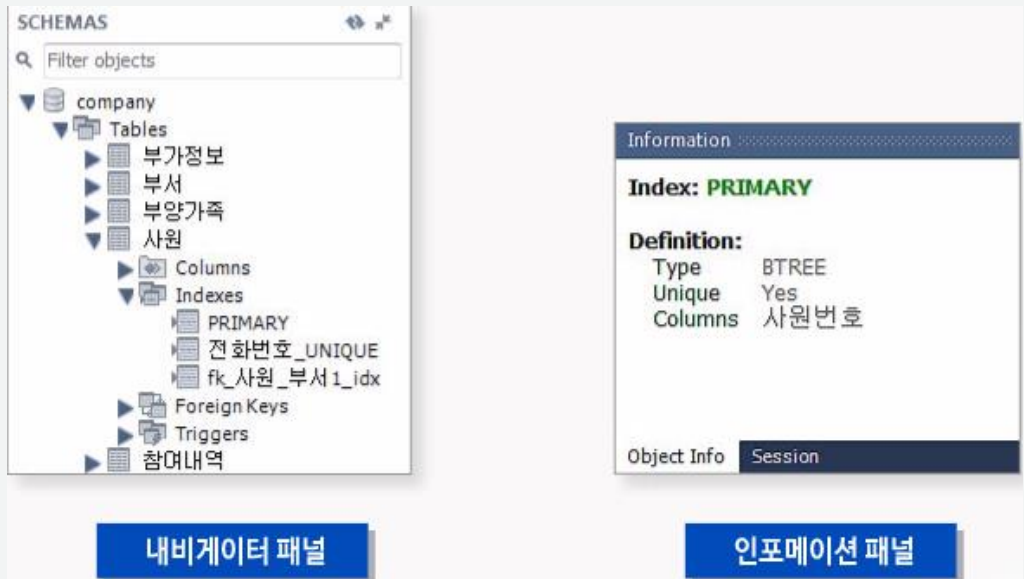
➤ 인덱스 삭제 구문 예

DROP INDEX idx_사원_1 ON 사원

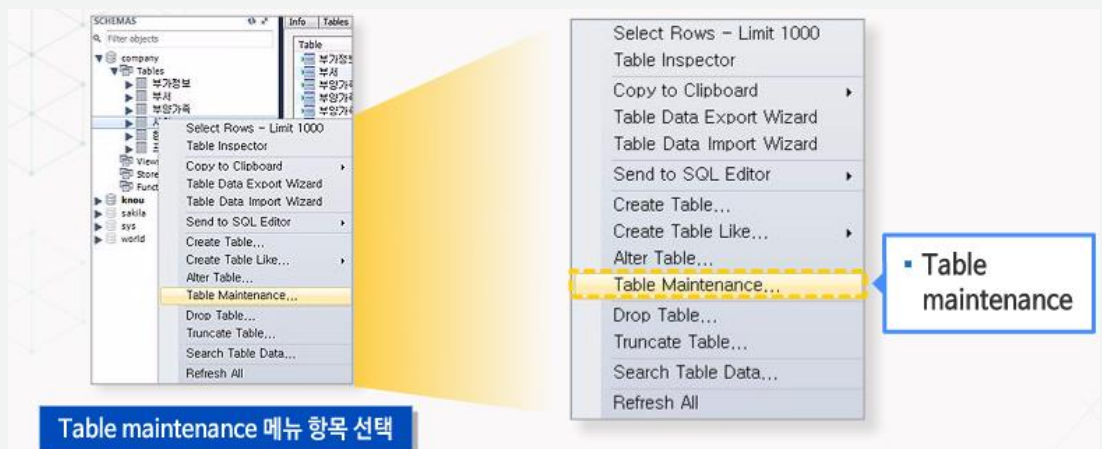
인덱스 관리

GUI 방식을 이용한 인덱스 관리

➤ 인덱스 확인



- GUI 방식을 이용한 인덱스 관리에서 내비게이터 패널과 인포메이션 패널을 통해 인덱스를 확인할 수 있습니다.



- MySQL Workbench를 이용한 인덱스 관리에서 Table maintenance 메뉴 항목을 선택하면 Table maintenance 화면에서 인덱스를 확인할 수 있습니다.



인덱스 관리



GUI 방식을 이용한 인덱스 관리

➤ 인덱스 확인

Table	Name	Unique	Inde...	Index Comment
부가정보	PRIMARY	Yes	BTREE	
부서	PRIMARY	Yes	BTREE	
부양가족	PRIMARY	Yes	BTREE	
부양가족	PRIMARY	Yes	BTREE	
부양가족	fk_부양가족_직원 1_idx	No	BTREE	
직원	PRIMARY	Yes	BTREE	
직원	건 화번호_UNIQUE	Yes	BTREE	
직원	fk_직원_부서 1_idx	No	BTREE	
참여내역	PRIMARY	Yes	BTREE	
참여내역	PRIMARY	Yes	BTREE	
참여내역	fk_프로젝트_has_직원_직원 1_idx	No	BTREE	
참여내역	fk_프로젝트_has_직원_프로젝트 ...	No	BTREE	
프로젝트	PRIMARY	Yes	BTREE	

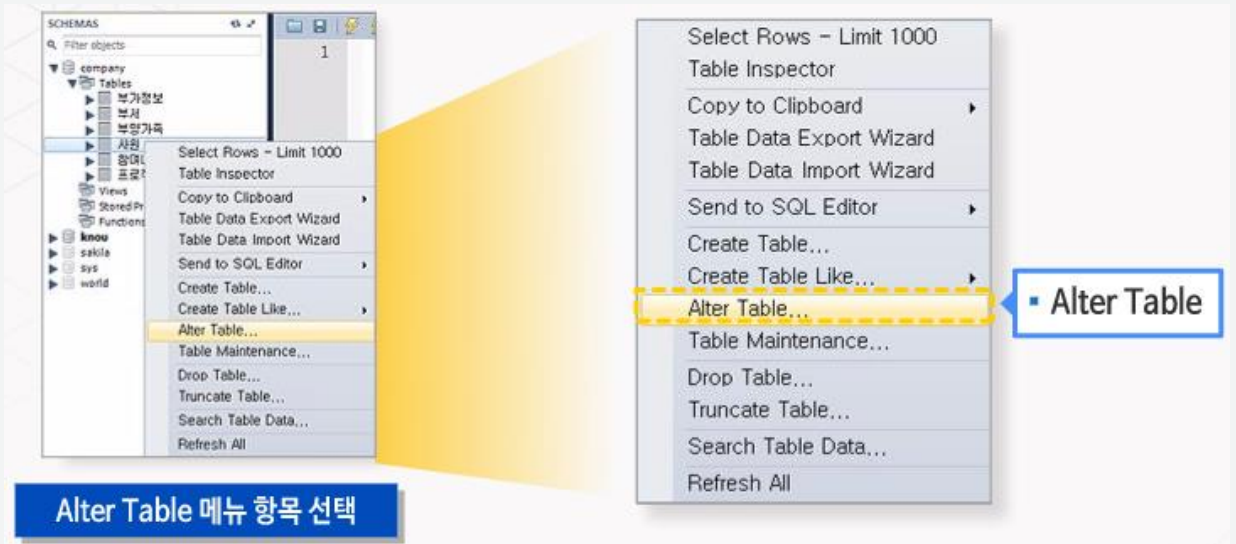
Table maintenance 화면

- 인덱스를 생성하기 위해서는 Alter Table 메뉴 항목을 선택합니다.

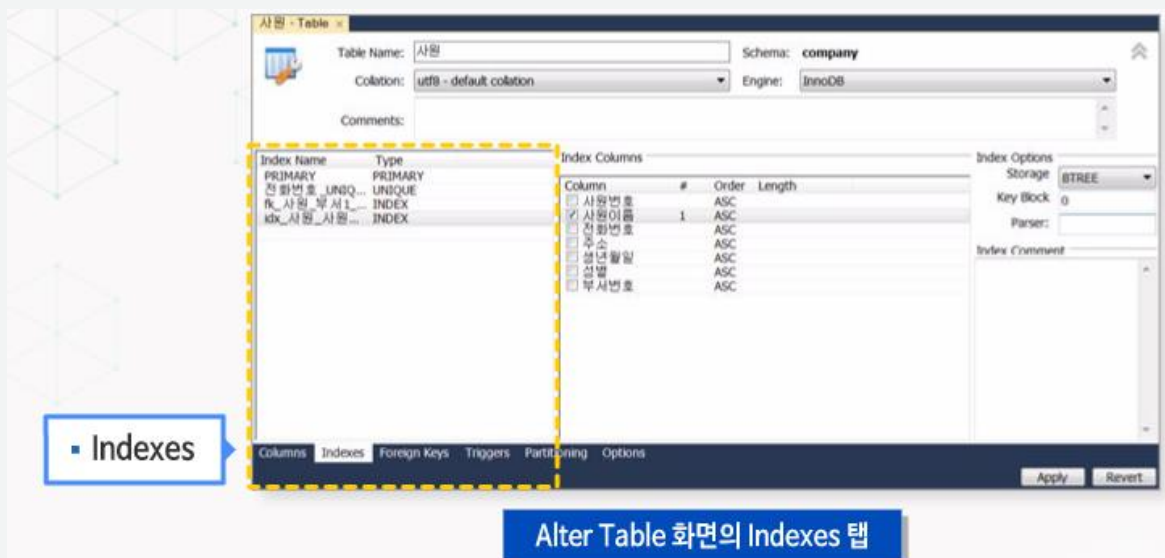
인덱스 관리

GUI 방식을 이용한 인덱스 관리

인덱스 생성



- Alter Table 화면에서 인덱스 탭을 확인 할 수 있습니다.

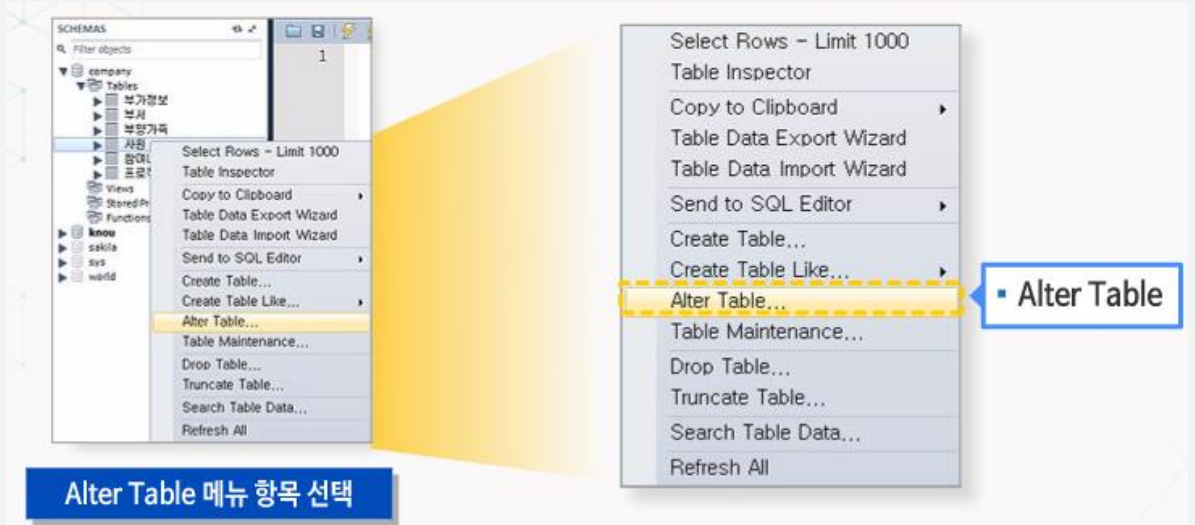


- 인덱스를 삭제하기 위해서는 인덱스를 생성했을 경우와 같은 방법으로 실행합니다.

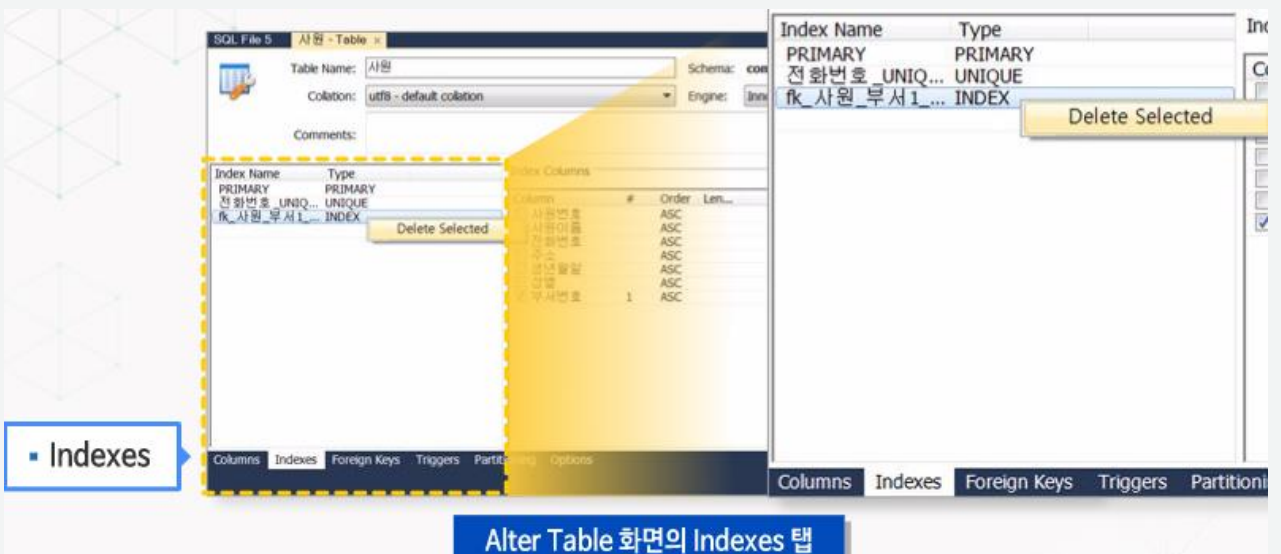
인덱스 관리

GUI 방식을 이용한 인덱스 관리

인덱스 삭제



- Alter Table 메뉴 항목을 선택 후 인덱스를 삭제할 수 있습니다.



- 인덱스에서 삭제항목을 선택 후 삭제합니다.

인덱스의 사용 효과

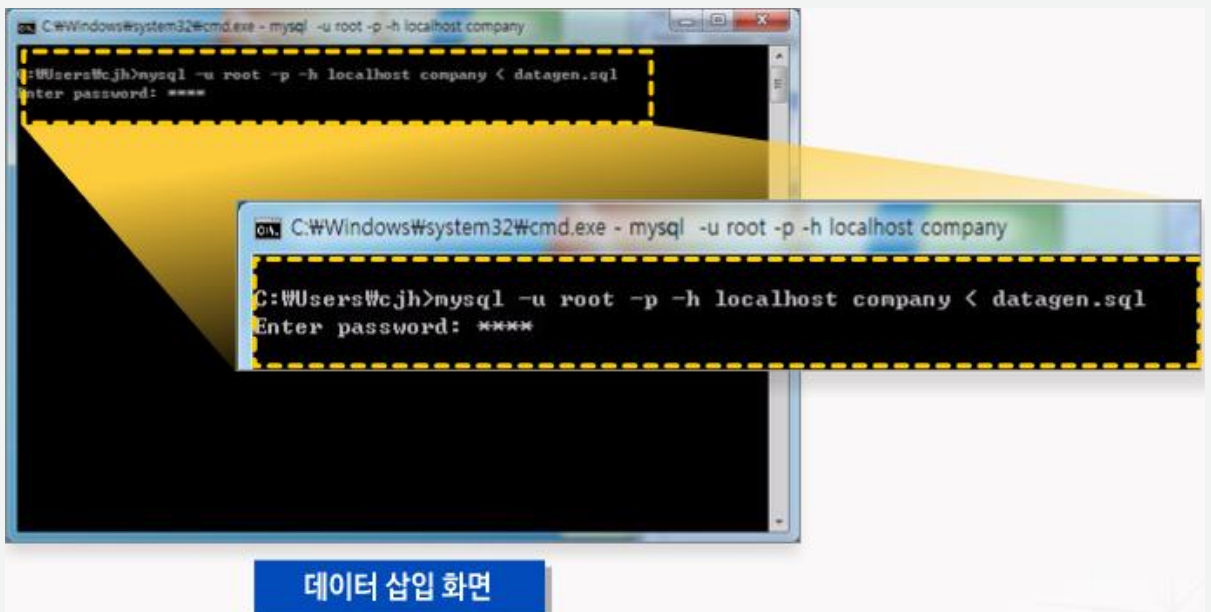
환경 구축

▶ 임시 테이블 생성

```
CREATE TABLE SAMPLE_TABLE  
(  
    STATIC_STRING CHAR(20),  
    STATIC_STRING_IDX CHAR(20),  
    DYNAMIC_STRING CHAR(30),  
    DYNAMIC_STRING_IDX CHAR(30),  
    INTEGER_NUMBER INT,  
    INTEGER_NUMBER_IDX INT,  
    FLOAT_NUMBER FLOAT,  
    FLOAT_NUMBER_IDX FLOAT  
)
```

- 인덱스 사용을 위해 환경을 구축합니다.

▶ 데이터 삽입



- 테이블을 생성 한 후 데이터를 삽입합니다.

인덱스의 사용 효과

검색 효과 비교

▶ 인덱스의 생성

```
CREATE INDEX idx_st_static_string  
ON SAMPLE_TABLE (static_string_idx)
```

```
SHOW INDEX FROM SAMPLE_TABLE
```

- 생성된 인덱스 확인 후 비인덱스 컬럼과 인덱스 컬럼을 각각 조회합니다.

▶ 비인덱스 컬럼 조회

Query 1

```
1 SELECT * FROM SAMPLE_TABLE  
2 WHERE STATIC_STRING LIKE 'ABX'  
3
```

Result Grid

STATIC_STRING	STATIC_STRING_IDX	DYNAMIC_STRING
AbU3AGQJshg0Chw	AwK9f9gHvD9gFDQe3z	j1qf
abLy2dW4eoQKNGTON	Q DMgB6Le5K26wp8U2	B2qyCT bD4ssCtp4DcJT2Q
ab4w4edogDE2oDCgB6EO	6vnxP3a02R120AX3e	ZOKgag cqy6P4Ww69812c
abP4cAHMCdMTeu0ko b	C6eW0Juxet43e94K0g	s26e8tq ed23b
ABhXvnbXlQ08phqy8PW	IDXgO72wYOG0ExOu pOK	hPM6uh7WTH8K3MAWQehSp

Duration / Fetch
0.063 sec / 0.000 sec

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	00:10:23	SELECT * FROM SAMPLE_TABLE WHERE S...	98 row(s) returned	0.063 sec / 0.000 sec

비인덱스 컬럼 조회 화면

- 인덱스의 사용 분석을 각각의 비인덱스 컬럼과 인덱스 컬럼을 조회하며 확인 할 수 있습니다.

인덱스의 사용 효과

검색 효과 비교

➤ 인덱스 컬럼 조회

The screenshot shows a database query execution window. The query is: `SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING_IDX LIKE 'ABX'`. The execution plan shows the use of the `STATIC_STRING_IDX` index. A callout box displays the execution duration: `Duration / Fetch 0.015 sec / 0.000 sec`. The result grid shows the first few rows of the query results.

인덱스 컬럼 조회 화면

- 인덱스의 사용 분석을 각각의 비인덱스 컬럼과 인덱스 컬럼을 조회하며 확인 할 수 있습니다.

인덱스의 사용 효과

검색 효과 비교

▶ 인덱스의 사용 분석

The screenshot shows a query window with the following SQL: `SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING LIKE 'ABX'`. The execution plan below the query shows a single step: a full table scan of `SAMPLE_TABLE`. A callout box highlights the `Duration / Fetch` as `0.063 sec / 0.000 sec`. The bottom status bar indicates that 95 rows were returned.

비인덱스 컬럼 조회 시 화면

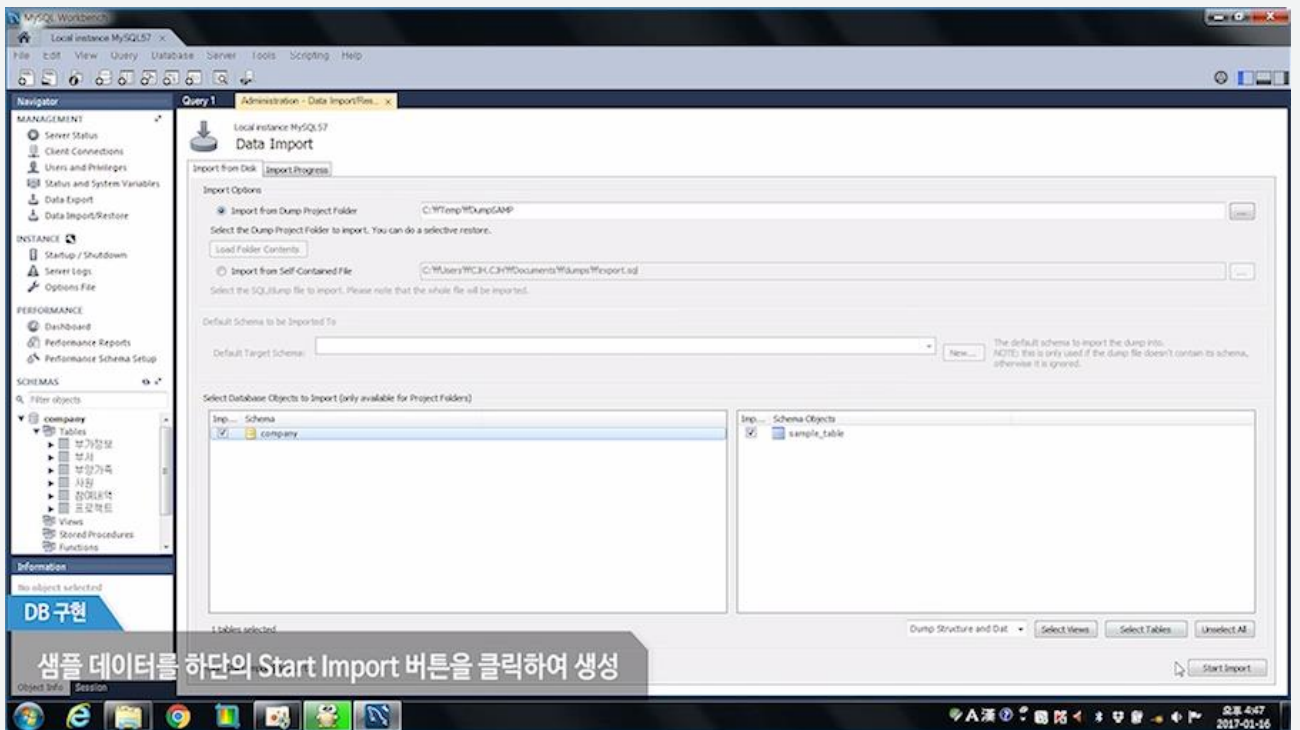
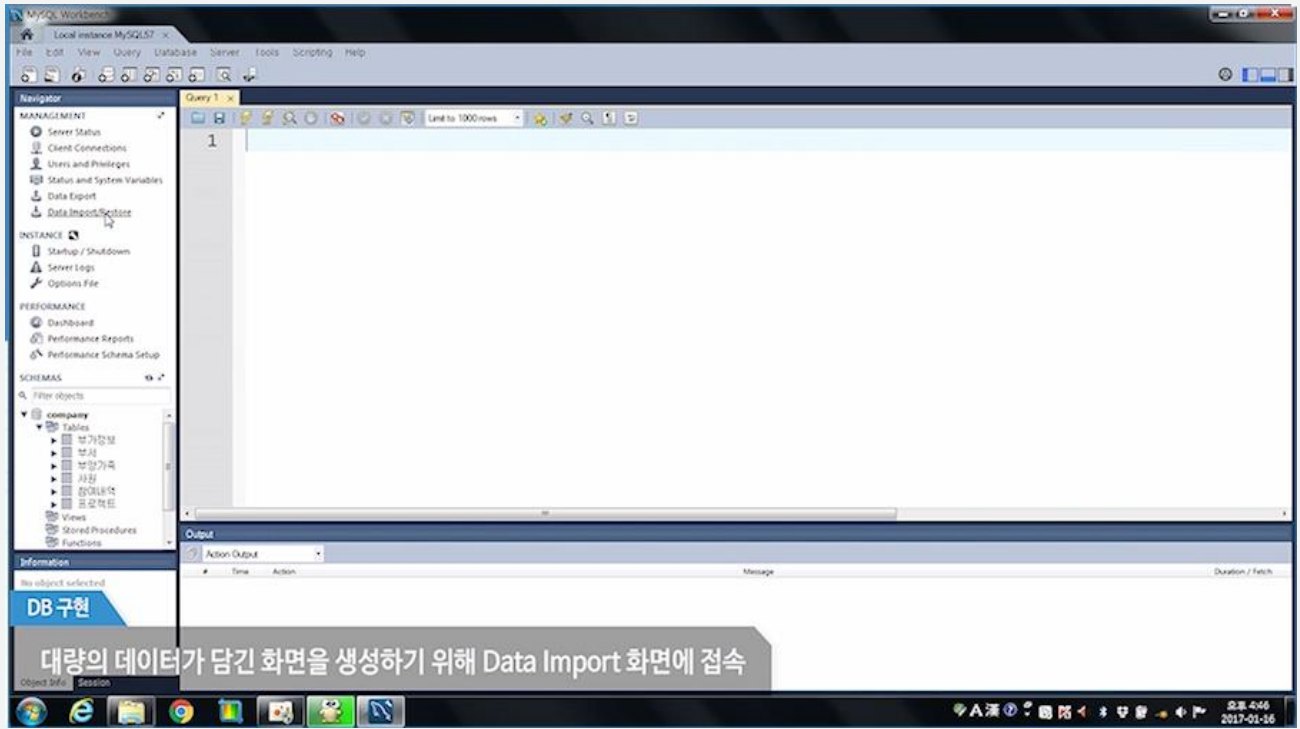
- 인덱스의 사용 분석을 각각의 비인덱스 컬럼과 인덱스 컬럼을 조회하며 확인 할 수 있습니다.

The screenshot shows a query window with the following SQL: `SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING_IDX LIKE 'ABX'`. The execution plan below the query shows a single step: an index scan on `STATIC_STRING_IDX` of `SAMPLE_TABLE`. A callout box highlights the `Duration / Fetch` as `0.015 sec / 0.000 sec`. The bottom status bar indicates that 95 rows were returned.

인덱스 컬럼 조회 시 화면

- 인덱스의 사용 분석을 각각의 비인덱스 컬럼과 인덱스 컬럼을 조회하며 확인 할 수 있습니다.

인덱스의 사용 효과



인덱스의 사용 효과

MySQL Workbench - Local instance MySQL57

Query 1: SHOW INDEX FROM SAMPLE_TABLE;

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	NULL	Index_type	Comment	Index_comment
sample_table	1	idx_float_num	1	Float_number_idx	A	103182			YES	BTREE		
sample_table	1	idx_static_str	1	Static_string_idx	A	899739			YES	BTREE		
sample_table	1	idx_dynamic_str	1	Dynamic_string_idx	A	903628			YES	BTREE		
sample_table	1	idx_integer_num	1	Integer_number_idx	A	80045			YES	BTREE		

Result 2: Output

Time	Action	Message	Duration / Fetch
17:01:55	SELECT COUNT(*) FROM SAMPLE_TABLE LIMIT 0, 1000	1 row(s) returned	0.733 sec / 0.000 sec
17:03:09	SHOW INDEX FROM SAMPLE_TABLE	4 row(s) returned	0.000 sec / 0.000 sec

DB 구현

데이터 검색을 통해 속도의 차이를 확인

MySQL Workbench - Local instance MySQL57

Query 1: SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING_IDX = '4JiCxWh10xwVUGbpB0vr';

STATIC_STRING	STATIC_STRING_IDX	DYNAMIC_STRING	DYNAMIC_STRING_IDX	INTEGER_NUMBER	INTEGER_NUMBER_IDX	FLOAT_NUMBER	FLOAT_NUMBER_IDX
4JiCxWh10xwVUGbpB0vr	4JiCxWh10xwVUGbpB0vr	4JiCxWh10xwVUGbpB0vr	4JiCxWh10xwVUGbpB0vr	96477	96477	0.60751	0.60751

Result 2: Output

Time	Action	Message	Duration / Fetch
17:01:55	SELECT COUNT(*) FROM SAMPLE_TABLE LIMIT 0, 1000	1 row(s) returned	0.733 sec / 0.000 sec
17:03:09	SHOW INDEX FROM SAMPLE_TABLE	4 row(s) returned	0.000 sec / 0.000 sec
17:03:42	SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING = '4JiCxWh10xwVUGbpB0vr' LIMIT 0, 1000	Error Code: 1146, Table 'company.sample' doesn't exist	0.000 sec / 0.000 sec
17:06:11	SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING_IDX = '4JiCxWh10xwVUGbpB0vr' LIMIT 0, 1000	1 row(s) returned	1.061 sec / 0.000 sec
17:06:11	SELECT * FROM SAMPLE_TABLE WHERE STATIC_STRING = '4JiCxWh10xwVUGbpB0vr' LIMIT 0, 1000	1 row(s) returned	0.016 sec / 0.000 sec

DB 구현

많은 양의 데이터가 존재하는 테이블에서 인덱스 차이의 속도 확인



핵심요약

인덱스의 이해

❖ 인덱스란?

- 데이터베이스에 존재하는 많은 양의 데이터 中 사용자가 원하는 특정 레코드만을 찾기 위해 사용하는 별도의 자료 구조
- 데이터 검색 시, 데이터베이스의 응답시간을 단축시키는 효과가 있음

❖ 자동 생성 인덱스

- 제약 조건에 의해서 자동으로 생성되는 인덱스
- PRIMARY KEY, UNIQUE, FOREIGN KEY 제약 조건이 생성된 컬럼에 대해서는 인덱스가 자동 생성됨



핵심요약

인덱스의 관리

- ❖ 인덱스는 데이터베이스 객체이기 때문에 생성, 수정, 삭제에 CREATE, ALTER, DROP 명령어로 관리할 수 있음
- ❖ 인덱스 관리의 종류
 - DDL 구문을 이용한 인덱스 관리
 - GUI 방식을 이용한 인덱스 관리
 - MySQL Workbench를 이용한 인덱스 관리
- ❖ MySQL Workbench
 - 인덱스를 GUI 방식으로 관리할 수 있는 메뉴 항목과 UI를 제공함

인덱스의 사용 효과

- ❖ 단축의 효과
 - 인덱스는 검색의 성능을 비약적으로 단축시키는 효과가 있음
- ❖ 인덱스의 사용 여부 분석
 - EXPLAIN 구문을 활용하여 검색 과정에서 인덱스의 사용 여부를 분석할 수 있음
- ❖ 인덱스 사용의 장·단점
 - 인덱스는 검색의 성능을 향상시키는 반면 데이터 입력, 수정, 삭제에는 성능이 감소함