



## 5장 데이터베이스 개체

김철학

# 목차

---

1. Index
2. View
3. 저장 프로시저
4. 저장 함수

# 1. Index

- 인덱스<sup>Index</sup>는 데이터 검색 성능 향상을 위해 테이블 특정 컬럼 데이터를 B-Tree 구조로 생성한 데이터 구조
- 기본키<sup>Primary Key</sup>는 클러스터형 인덱스가 생성되고 고유키<sup>Unique Key</sup>는 보조 인덱스 생성
- 인덱스는 INSERT 작업이 빈번할 경우 인덱스 재정렬 및 페이지 분할로 성능 저하가 발생

인덱스 종류	설명
클러스터형 인덱스	<ul style="list-style-type: none"><li>• 테이블 자체의 물리적인 정렬 순서를 인덱스로 사용</li><li>• 데이터의 물리적 저장 순서와 인덱스가 동일하므로, 검색 작업이 매우 빠름</li><li>• 데이터 삽입, 수정, 삭제 시 인덱스 재정렬이 필요하여 성능 저하가 발생</li><li>• 테이블당 1개 생성, 기본 인덱스</li><li>• 기본키 생성</li></ul>
보조 인덱스	<ul style="list-style-type: none"><li>• 테이블의 물리적인 저장 순서와는 관계없이 별도의 데이터 구조로 인덱스를 생성</li><li>• 데이터 삽입, 수정, 삭제 시 인덱스 재정렬이 더 적게 필요</li><li>• 테이블당 여러 개 생성</li><li>• 고유키 생성</li></ul>

## 2. View

- 뷰<sup>View</sup>는 하나 이상의 테이블을 조회하는 SELECT 결과를 가진 가상 테이블 개체
- 뷰는 테이블의 특정 컬럼의 노출을 피하거나 자주 사용하는 복잡한 쿼리를 간편하게 사용할 목적



### 3. 저장 프로시저

---

- 저장 프로시저<sup>Stored Procedure</sup>는 자주 사용하는 쿼리를 모듈화해서 호출하는 방식으로 사용하는 개체
- 커서<sup>Cursor</sup>는 테이블 조회 결과를 한 행씩 처리하는 개체

```
1  USE sqlDB;
2  DROP PROCEDURE IF EXISTS userProc;
3  DELIMITER $$
4  CREATE PROCEDURE userProc()
5  BEGIN
6      SELECT * FROM userTbl; -- 스토어드 프로시저 내용
7  END $$
8  DELIMITER ;
9
10 CALL userProc();
```

## 4. 저장 함수

---

- 저장 함수 `Stored Function` 는 저장 프로시저와 비슷하지만 하나의 결과를 반환
- 저장 함수의 매개변수는 모두 입력이고 `SELECT`문으로 호출

```
DELIMITER $$  
CREATE FUNCTION 스토어드 함수이름( 파라미터 )  
    RETURNS 반환형식  
BEGIN  
  
    이 부분에 프로그래밍 코딩..  
    RETURN 반환값;  
  
END $$  
DELIMITER ;  
SELECT 스토어드_함수이름();
```