

4장 데이터의 추가, 삭제, 갱신

190727

조윤준

• 목차

- INSERT 명령으로 데이터 추가하기
- DELETE 명령으로 데이터 삭제하기
- UPDATE 명령으로 데이터 변경하기
- 논리삭제와 물리삭제

INSERT

• INSERT

INSERT INTO 테이블명 VALUES (값 1, 값2, ...)

- SELECT 명령과 같은 질의를 하면 데이터베이스 서버가 클라이언트로 결과를 반환하는 형식으로 처리
- 데이터를 추가할 경우에는 이와 반대로 클라이언트에서 서버로 데이터를 전송하는 형식을 취하며 서버 측은 전송 받은 데이터를 데이터베이스에 저장

• INSERT로 행 추가하기

- RDBMS에서는 INSERT 명령을 사용해 테이블의 행 단위로 데이터를 추가

no	a	b
----	---	---

Ex. INSERT INTO sample41 VALUES (1, 'ABC', '2014-01-25');

no	a	b
1	ABC	2014-01-25

• 값을 저장할 열 지정하기

- INSERT INTO 테이블명 (열1, 열2, ...) VALUES (값1, 값2, ...)

Ex. INSERT INTO sample41(a, no) VALUES ('XYZ', 2);

no	a	b
1	ABC	2014-01-25
2	XYZ	NULL

- DESC

Ex. DESC sample41;

Field	Type	Null	Key	Default	Extra
no	int(11)	NO		NULL	
a	varchar(30)	YES		NULL	
b	date	YES		NULL	

• NOT NULL 제약

- 행을 추가할 때 NULL로 두고 싶을 경우에는 VALUES 구에서 NULL로 값을 지정할 수 있음

Ex. INSERT INTO sample41(no, a, b) VALUES (NULL, NULL, NULL);

- 이 명령어를 실행하면 **에러 발생**

→ no 열에 대해 NULL 값을 허용하지 않는 **NOT NULL 제약이 걸려 있음**

Ex. INSERT INTO sample41(no, a, b) VALUES (3, NULL, NULL);

→ 이러면 가능!

• DEFAULT

- DESC 명령으로 열 구성을 살펴보면 **Default**라는 항목을 찾을 수 있음
- Default는 명시적으로 값을 지정하지 않았을 경우 사용하는 초기값

Ex. INSERT INTO sample411 (no, d) VALUES (2, DEFAULT);

→ Default 값으로 insert 됨

Ex. INSERT INTO sample411 (no) VALUES (2);

→ 열을 지정하지 않으면 default 값으로 행이 추가됨

DELETE

• DELETE

DELETE FROM 테이블명 WHERE 조건식

- RDBMS에서 데이터를 삭제할 경우에는 행 단위로 DELETE 명령을 수행

Ex. DELETE FROM sample41;

- WHERE 구를 생략할 경우에는 모든 행을 대상으로 동작
- 지정할 경우에는 해당 조건식에 맞는 행만 삭제 대상이 됨

• DELETE 명령구

Ex. DELETE FROM sample41 WHERE no=3;

→ no 열의 값이 3인 행이 삭제된 것을 확인 가능

- SELECT 명령과 같이 열을 지정할 수 없음

Ex. DELETE FROM sample41 WHERE no=1 OR no=2;

→ no 열의 값이 1 또는 2인 행이 삭제된 것을 확인 가능

UPDATE

• UPDATE

UPDATE 테이블명 SET 열1=값1, 열2=값2, ... WHERE 조건식

- RDBMS에서는 UPDATE 명령으로 데이터를 갱신할 수 있음
- UPDATE 명령은 테이블의 셀 값을 갱신하는 명령
- UPDATE 명령에서는 SET 구를 사용하여 갱신할 열과 값을 지정
- 갱신해야 할 열과 값이 여러 개일 경우에는 콤마(,)로 구분하여 리스트 형식으로 지정

• UPDATE 예시

Ex. UPDATE sample41 SET b = '2014-09-07' WHERE no=2;

Ex. UPDATE sample41 SET no=no+1;

Ex. UPDATE sample41 SET a='xxx', b='2014-01-01' WHERE no=2;

Ex1. UPDATE sample41 SET no=no+1, a=no;

Ex2. UPDATE sample41 SET a=no, no=no+1;

Ex. UPDATE sample41 SET a=NULL;

물리삭제와 논리삭제

• 물리삭제와 논리삭제

- **물리삭제**는 SQL의 DELETE 명령을 사용해 직접 데이터를 삭제하는 방식

→ 삭제 대상 데이터는 필요 없는 데이터이므로 DELETE 명령을 실행해서 테이블에서 삭제

- **논리삭제**는 테이블에 '삭제플래그' 열을 이용하여 데이터가 삭제된 것처럼 가장하는 방식

→ UPDATE 명령을 이용해 '삭제플래그'의 값을 1로 갱신

→ 실제 테이블 안에 데이터는 남아있지만, 참조할 때에는 '삭제플래그'가 삭제로 설정된 행을 제외하는 SELECT 명령을 실행 → 결과적으로 해당 행이 삭제된 것처럼 보임

• 논리삭제

- 논리삭제는 테이블에 '삭제플래그' 열을 이용하여 데이터가 삭제된 것처럼 가장하는 방식
 - UPDATE 명령을 이용해 '삭제플래그'의 값을 1로 갱신
 - 실제 테이블 안에 데이터는 남아있지만, 참조할 때에는 '삭제플래그'가 삭제로 설정된 행을 제외하는 SELECT 명령을 실행 → 결과적으로 해당 행이 삭제된 것처럼 보임

no	a	삭제플래그
1	ABC	0
2	DEF	0
3	GHI	0
4	JKL	0

UPDATE로 삭제플래그 열의 값을 1로 갱신

no	a	삭제플래그
1	ABC	0
2	DEF	0
3	GHI	1
4	JKL	0

SELECT FROM xxx WHERE 삭제플래그 != 1로 검색

끝 😊