

2. 데이터베이스에서 Null

먼저, 데이터베이스에서 Null은 다음과 같은 의미를 가진다.

모름, 미정 (unknown)	<ul style="list-style-type: none"> 존재하지만 누락된 경우(사람의 체중, 키 등) 존재하는지조차도 모르는 경우(유전자 정보)
해당사항 없음 (not applicable)	<ul style="list-style-type: none"> 미혼자의 결혼기념일 고졸자의 대학교 학사 정보 외국인의 한국 주민등록번호 등
유보, 보류 (withheld)	<ul style="list-style-type: none"> 의도적으로 유보한 값(withheld value) 알고 있지만 의도적으로 공개하지 않은 것

◆ null이 포함된 사칙 및 비교 연산

질의문	수행 결과	질의문	수행 결과
Select null + 1;	null	Select null > 1;	null
Select null - 1;	null	Select null < 1;	null
Select null * 1;	null	Select null = 1;	null
Select null / 1;	null	Select null <> 1;	null

- 널(null)과의 사칙 및 비교 연산 결과는 null이다.
- **Select Null Is Null;** -- 1(참) // null과의 비교는 Is를 사용해야 참(1)이 될 수 있고
- **Select Null Is Not Null;** -- 0(거짓) // null과의 비교에서 Is Not을 사용하면 거짓(0)이 된다.

◆ 논리 연산

A and B				A or B				not A	
A \ B	true	false	unknown	A \ B	true	false	unknown	A	not A
true	true	false	unknown	true	true	true	true	true	false
false	false	false	false	false	true	false	unknown	false	true
unknown	unknown	false	unknown	unknown	true	unknown	unknown	unknown	unknown

- 논리값으로 Null은 참(true)도 아니고 거짓(false)도 아닌 미정(unknown)이 된다.
- 다음은 MySQL에서 실행한 결과이다.(Null값 처리는 DBMS에 따라 약간의 차이가 있다)

질의문	수행 결과	질의문	수행 결과
Select (5 > 2) and (6 > 3);	1(true)	Select (5 > 2) or (6 < 3);	1(true)
Select (5 > 2) and (6 < 3);	0(false)	Select (5 > 2) or null;	1(true)
Select (5 > 2) and null;	null	Select (5 < 2) or null;	null
Select (5 < 2) and null;	0(false)	Select null or null;	null

[예제] 실습 - Null

〈실습 null〉

수강

학번	과목	점수
100	디비	90
200	소공	80
300	디비	90

↓
↓ Update 수강
↓ Set 점수 = 점수 + null
↓ Where 과목 = '디비';
↓

〈실행 결과〉

수강

학번	과목	점수
100	디비	null
200	소공	80
300	디비	null

'디비' 점수는 모두 null로 수정되었다.

◆ DBMS에서 null과 빈 문자열(empty string)

Oracle	• 오라클에서 빈 문자열 ""은 모두 null로 처리한다. • 즉, 오라클에서는 빈 문자열과 null은 같다.
MySQL	• 빈 문자열 ""을 null로 처리하지 않는다.
MariaDB	
SQLServer	• 즉, 빈 문자열과 null은 다르다.

- Insert Into 수강(학번, 과목) Values (500, Null);
- Insert Into 수강(학번, 과목) Values (600, ""); -- 빈 문자열

// 실습 : MariaDB에서

질의문	수행 결과	비고
Select Length(Null);	null	null의 문자열 길이는 null이다.
Select Length("");	0	빈 문자열의 문자열 길이는 0이다.
Select Length(' ');	1	빈 문자열이 아니다. 문자열 길이는 1이다.

기출문제 분석

1. 관계형 데이터 모델에서 무결성 제약조건(integrity constraints)에 대한 설명으로 옳지 않은 것은? [2010년 국가 7급]

- ① NOT NULL 제약조건은 자료를 입력할 때 속성 값을 반드시 요구한다.
- ② Primary Key 칼럼은 NULL 값을 가질 수 없다.
- ③ Unique 칼럼은 NULL 또는 유일한 값을 가질 수 있다.
- ④ Foreign Key 칼럼은 참조되는 속성 값이 반드시 존재해야 하고 NULL 값을 가질 수 없다.

☞ 무결성 제약조건(integrity constraints)

- Foreign Key 칼럼은 참조되는 속성 값이 반드시 존재해야 하고 NULL 값을 가질 수 없다.(*)
→ Foreign Key 칼럼은 NULL 값을 가질 수 있다.

◆ Unique 칼럼

- Unique은 유일한 값으로 존재해야 하는 것을 의미한다.(중복 배제)
→ Null 값에 대해서는 Unique 제약이 적용되지 않는다.(다수의 Null 값 존재 가능)
→ Null 값은 데이터로 인식하지 않는 원리이다.
- Unique 칼럼은 Null 또는 유일한 값을 가질 수 있다.
- Unique 칼럼은 Primary Key로 인식되는 것은 아니다.
- Unique 칼럼으로 참조 무결성 지정은 불가능하다.
- 하나의 테이블에 Unique 칼럼은 여러 개 존재할 수 있다.

◆ 무결성 제약조건 종류

Not Null	해당 칼럼 값으로 NULL을 허용하지 않는다.
Unique	테이블 내에서 해당 칼럼 값은 항상 유일한 값을 가진다.
Primary Key	NULL을 허용하지 않고, 중복된 값도 허용하지 않는다. Primary Key = Not Null + Unique
Check	범위나 조건을 지정하여 설정한 값만 허용한다.
Foreign Key	참조되는 테이블의 칼럼 값이 존재하면 허용한다. Foreign Key 칼럼은 NULL 값을 가질 수 있다.

4. 데이터베이스에서 널(null) 값이 적용되는 예로 적절하지 않은 것은? [2012년 국가 7급]

- ① A 남자 환자의 경우, 출산여부 항목
- ② B 신입생의 주소가 아직 파악되지 않은 경우, 주소 항목
- ③ 연봉이 동결된 C 사원의 경우, 연봉인상률 항목
- ④ D 제품에 대한 제품가격이 담당부서에서 아직 넘어오지 않은 경우, 제품가격 항목

☞ 데이터베이스에서 널(null)

모름, 미정(unknown)	<ul style="list-style-type: none"> • 존재하지만 누락된 경우(사람의 체중, 키 등) • 존재하는지조차도 모르는 경우(유전자 정보)
해당사항 없음(not applicable)	<ul style="list-style-type: none"> • 미혼자의 결혼기념일, 외국인의 한국 주민등록번호 등
유보, 보류(withheld)	<ul style="list-style-type: none"> • 의도적으로 유보한 값(withheld value) • 알고 있지만 의도적으로 공개하지 않은 것

- 연봉이 동결된 것은 인상률은 0이라는 것이다. 데이터베이스에서 0과 널(null)은 다르다.
- 참고로, C 언어에서는 null은 0이다. C와 데이터베이스는 다르다.

정답 : ③

5. 직원(사번, 이름, 입사년도, 부서) 테이블에 대한 SQL문 중 문법적으로 옳은 것은? [2016년 계리직]

- ① SELECT COUNT(부서) FROM 직원 GROUP 부서;
- ② SELECT * FROM 직원 WHERE 입사년도 IS NULL;
- ③ SELECT 이름, 입사년도 FROM 직원 WHERE 이름 = '최%';
- ④ SELECT 이름, 부서 FROM 직원 WHERE 입사년도 = (2014, 2015);

☞ SQL문 문법

- ① SELECT COUNT(부서) FROM 직원 GROUP 부서; (×)
→ SELECT COUNT(부서) FROM 직원 **Where** 부서 = '영업부'; (조건은 Where 사용)
- ② SELECT * FROM 직원 WHERE 입사년도 IS NULL; (○)
→ NULL 값을 비교할 때는 IS(또는 IS NOT)을 사용해야 한다.(정상적인 문법)
- ③ SELECT 이름, 입사년도 FROM 직원 WHERE 이름 = '최%'; (×)
→ SELECT 이름, 입사년도 FROM 직원 **WHERE** 이름 **Like** '최%';
- ④ SELECT 이름, 부서 FROM 직원 WHERE 입사년도 = (2014, 2015); (×)
→ SELECT 이름, 부서 FROM 직원 **WHERE** 입사년도 = 2014 **Or** 입사년도 = 2015;
→ 조건이 두 개이면 연산자 Or를 사용해야 한다.

정답 : ②

6. 다음 SQL 문장 중 구문이 옳은 것은? [2017년 국회 9급]

- ① Delete From Student, Enrol Where Sno = 100;
- ② Select Count(Distinct Cno) From Enrol Where Sno = 100;
- ③ Select Sno, Sname From Student Where Dept = Null;
- ④ Insert Student Into Values(100, '홍길동', 2, '전산과');
- ⑤ Select Dno, Sum(Sname) From Student Group By Dno Where Sum(Sname)<100;

♣ SQL 문장

- ① Delete From **Student, Enrol** → 테이블이 2개 기술되어 오류
Where Sno = 100;
 - ② Select Count(Distinct Cno) From Enrol Where Sno = 100; → ②번 문장은 정상
 - ③ Select Sno, Sname From Student
Where Dept = Null; → Null 비교는 Dept **Is** Null(연산자 =을 사용 안함)
 - ④ Insert Student **Into** Values(100, '홍길동', 2, '전산과');
→ Insert **Into** Student Values(100, '홍길동', 2, '전산과');
 - ⑤ Select Dno, Sum(Sname) From Student
Group By Dno
Where Sum(Sname)<100; → Group By에서는 조건은 **Having**을 사용
-

정답 : ②