

[Database] 테이블 생성(CREATE TABLE) 및 제약조건(CONSTRAINT)

노트북: [TIL-MY]

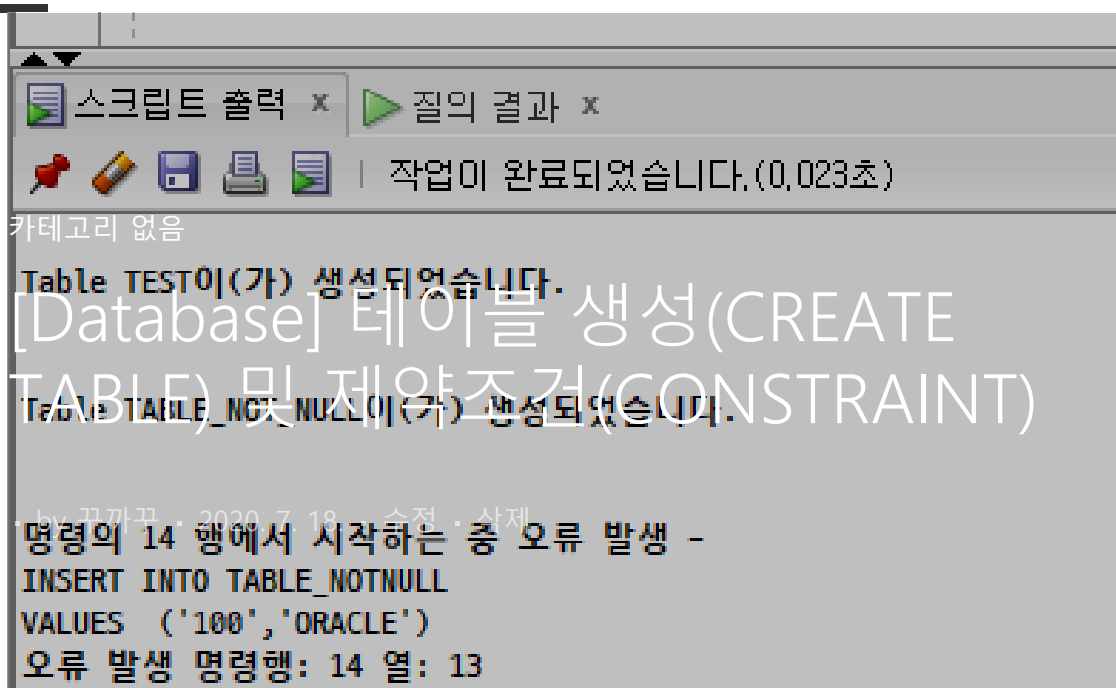
만든 날짜: 2020-07-19 오후 4:46

URL: <https://continuous-development.tistory.com/28>

나무늘보의 개발 블로그

홈

태그



테이블 생성하기

```
CREATE TABLE table_name (column_name datatype [DEFAULT expr] [column_constraint], [table_constraint ])
```

[분류 전체보기](#)

[Python](#)

[Database](#)

[ASP.NET](#)

[Algorithm](#)

[Deep learning](#)

column_constraint - 컬럼수준에서 제약조건을 거는 것 / 각 컬럼마다 적는다.

table_constraint - 컬럼을 다 만들고 마지막에 테이블 수준에서 제약조건을 건다.

table_name, column_name - 테이블 이름 지정, 컬럼 이름 지정

datatype - 컬럼의 데이터 타입, 크기 지정

DEFAULT expr - 해당 컬럼에 적용될 자동 기본값

테이블 생성 조건

테이블 및 컬럼 이름

문자로 시작, 30자 이하

영문대/소문자, 숫자, 특수문자, 한글만 포함가능

중복되는 이름은 사용 할 수 없음

예약 키워드는 사용할 수 없음

AWS

ETC..

공지사항

글 보실 때 주의사항

: 최근글 : 인기글

[Data
테...



2020.07.18

[Data
Sub...



2020.07.17

[Data
SEP...



2020.07.17

[Data
Self...



2020.07.17

[Data
IN...



2020.07.16

최근댓글

태그

Oracle SQL,

OUTER JOIN 함

수,

MINUS함수,

예제

```
CREATE TABLE TEST(  
    ID NUMBER(5),  
    NAME CHAR(10),  
    ADDRESS VARCHAR2(50)  
);
```

제약조건(CONSTRAINT)

테이블을 만들때는 제약조건을 넣을수 있다..

WQQ2

NOT NULL - 해당 컬럼에 NULL이 들어가선 안된다.

UNIQUE - 해당 컬럼 또는 컬럼 조합 값이 유일하도록 한다.

PRIMARY KEY - 각 행을 유일하게 식별할 수 있도록 한다.

REFERENCES TABLE - 외래키로서 부모키와 관계를 가진다. 이걸 부모키에 의존하는 데이터이거나 NULL 값만 허용 한다.

CHECK - 해당 컬럼에 특정 조건을 항상 만족시키도록 한다(전처리)

selfjoin,
IN Operator,
SQL 테이블 생성,
CRATE TABLE,
selfjoin사용법,
SEP Operator,
selfjoin함수,
서브쿼리 사용법,
날짜함수,
SUBQUERY사용,
trunc 사용,
UNION함수,
AWS, 인스턴스,
Oracle,
self join 함수,
trunc 함수,
UNION ALL함수,
SQL,
self join 사용법,
SUBQUERY함수,
설정, 사용법,
INTERSECT함수,
INNER JOIN
OUTER JOIN,
Subquery 함수,
select trunc

전체 방문자

91

Today : 0

Yesterday : 1

제약조건을 설정 할 수있고 나머지는 테이블과 컬럼 레벨에서 다 가능하다.

제약 조건을 만들 때 컬럼부분에서 제약조건을 설정 하는 방법과 컬럼부분이 끝난뒤 맨 마지막에 제약조건을 설정하는 테이블 수준의 제약조건 설정을 하는 방법 2가지가 있다.

제약 조건을 통해 데이터가 들어올때 전처리를 한다.

NOT NULL 제약조건

```
SELECT *
FROM TEST;

CREATE TABLE TABLE_NOTNULL
( ID CHAR(3) NOT NULL,
  SNAME VARCHAR2(20));

INSERT INTO TABLE_NOTNULL
VALUES ('100','ORACLE');
```

스크립트 출력 x | 질의 결과 x

작업이 완료되었습니다. (0.023초)

Table TEST이(가) 생성되었습니다.

Table TABLE_NOT_NULL이(가) 생성되었습니다.

명령의 14 행에서 시작하는 중 오류 발생 -
INSERT INTO TABLE_NOTNULL
VALUES ('100','ORACLE')
오류 발생 명령행: 14 열: 13
오류 보고 -
SQL 오류: ORA-00942: table or view does not exist
00942. 00000 - "table or view does not exist"
*Cause:
*Action:

Table TABLE_NOTNULL이(가) 생성되었습니다.

1 행 이(가) 삽입되었습니다.

NOT NULL 조건에서 아래와 같이 NULL을 입력하면 오류가 뜬다.

```
INSERT INTO TABLE_NOTNULL
VALUES (NULL, 'ORACLE');
```

스크립트 출력 x | 질의 결과 x

작업이 완료되었습니다.(0,027초)

명령의 18 행에서 시작하는 중 오류 발생 -
INSERT INTO TABLE_NOTNULL
VALUES (NULL, 'ORACLE')
오류 보고 -
ORA-01400: cannot insert NULL into ("HR"."TABLE_NOTNULL"."ID")

여기 아래에서 오류가 난 이유는 위에서 말했듯이 NOT NULL 제약 조건은 테이블 레벨에서 할 수 없고 컬럼 레벨에서 해야 된다.

```
CREATE TABLE TABLE_NOTNULL2
( ID CHAR(3),
  SNAME VARCHAR2(20),
  CONSTRAINT TN2_ID_NN NOT NULL (ID));
```

스크립트 출력 x | 질의 결과 x

작업이 완료되었습니다.(0,026초)

명령의 21 행에서 시작하는 중 오류 발생 -
CREATE TABLE TABLE_NOTNULL2
(ID CHAR(3),
 SNAME VARCHAR2(20),
 CONSTRAINT TN2_ID_NN NOT NULL (ID))
오류 보고 -
ORA-00904: : invalid identifier
00904. 00000 - "%s: invalid identifier"
*Cause:
*Action:

REFERENCES 제약조건

```

CREATE TABLE TABLE_FK(
  ID      CHAR(3),
  SNAME   VARCHAR2(20),
  LID     CHAR(2) REFERENCES LOCATION (LOCATION_ID) );

INSERT INTO TABLE_FK
VALUES ('200','ORACLE','C1');

```

CHECK 제약조건 / PRIMARY KEY 제약 조건

입력되는 값을 체크하여 설정된 값이외의 값이 들어오면 오류와 함께 쿼리가 실행이 되지 않게 만든다.

변하는 값은 조건으로 사용할 수 없다.

```

--CHECK

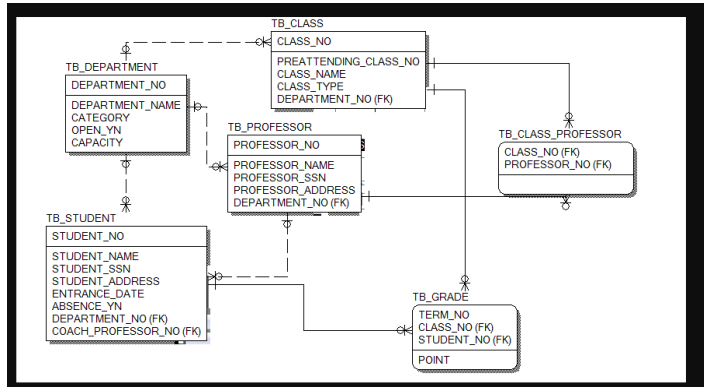
CREATE TABLE TABLE_CHECK(
  EMP_ID  CHAR(3) PRIMARY KEY,
  SALARY  NUMBER CHECK(SALARY > 0),
  MARRIAGE CHAR(1),
  CONSTRAINT CHK_MRG CHECK (MARRIAGE IN ('Y','N') ));

```

교차 테이블 생성하는 방법

관계형 데이터 베이스는 다대다를 구현할 수 없다.
다대다를 교차 테이블을 하나만들어서 풀어준다.

테이블이 두개일때 테이블을 하나 더만들어서 이런
다대다 관계를 풀어 준다.



여기서 TB_CLASS_PROFESSOR 과 같다. 이런경우에
외래키 이면서 기본키이다.

이런경우에는 컴퍼짓 키는 테이블의 조건으로 하고
각각의 외래키는 컬럼수준으로 한다.

```
CREATE TABLE ENROLL(  
    STU_ID NUMBER REFERENCES STUDENT (STU_ID),  
    SUBJECT_ID NUMBER REFERENCES SUBJECT (SUBJECT_ID),  
    PRIMARY KEY(STU_ID,SUBJECT_ID)  
);
```

스크립트 출력 x | 질의 결과 x

작업이 완료되었습니다. (0.03초)

Table SUBJECT이(가) 생성되었습니다.

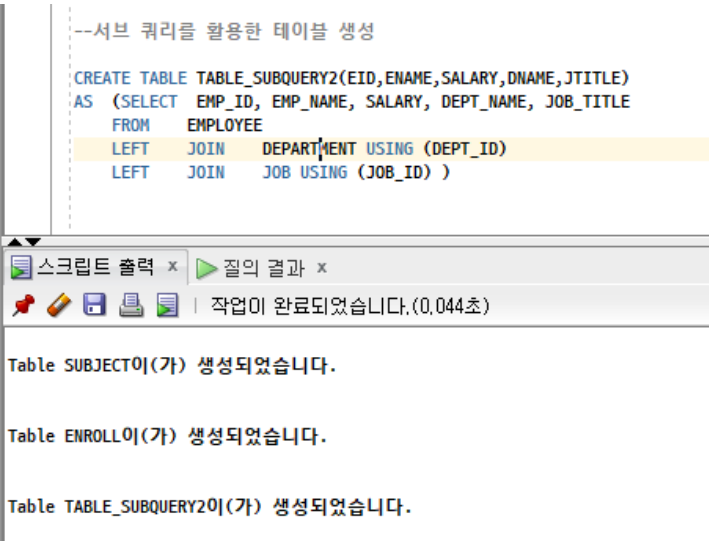
Table ENROLL이(가) 생성되었습니다.

이런 테이블의 쿼리다. 컬럼수준에서는 외래키를 잡고 테이블 수준에서 복합키를 잡으면 된다.

서브쿼리를 활용해 테이블을 생성

```
CREATE TABLE table_name [(column_name)] AS  
SUBQUERY
```

서브쿼리를 사용하기 때문 물리적인 메모리 공간을 차지하는 단점이 있긴하다.



EMPLOYEE 테이블에 데이터가 있었기 때문에 새로만든 서브쿼리 테이블에도 데이터가 들어간다.

--서브 쿼리를 활용한 테이블 생성

```
CREATE TABLE TABLE_SUBQUERY2(EID,ENAME,SALARY,DNAME,JTITLE)
AS (SELECT EMP_ID, EMP_NAME, SALARY, DEPT_NAME, JOB_TITLE
FROM EMPLOYEE
LEFT JOIN DEPARTMENT USING (DEPT_ID)
LEFT JOIN JOB USING (JOB_ID) );

SELECT
FROM TABLE_SUBQUERY2;
```

스크립트 출력 x						질의 결과 x					
SQL 인출된 모든 행: 22(0.004초)											
	EID	ENAME	SALARY	DNAME	JTITLE						
1	100	한선기	9000000	해외영업3팀	대표이사						
2	101	강중훈	5500000	해외영업3팀	부사장						
3	102	최만식	3600000	해외영업3팀	부사장						
4	104	안석규	3500000	기술지원팀	부장						
5	107	조재형	3800000	기술지원팀	부장						
6	144	김순이	3400000	해외영업1팀	부장						
7	103	정도연	2600000	기술지원팀	차장						
8	207	김술오	2500000	회계팀	차장						
9	208	이중기	2500000	회계팀	차장						
10	210	감우섭	2500000	회계팀	차장						
11	141	김예수	2100000	해외영업1팀	과장						
12	143	나승원	2300000	해외영업1팀	과장						
13	201	박하일	2600000	해외영업1팀	과장						
14	174	전우선	2090000	해외영업2팀	대리						

태그

constraint

CRATE TABLE

DDL

Oracle

SQL 테이블 생성

제약조건

테이블 생성

댓글 0

