

제5장 SQL



5.2 DDL(Data Definition Language)

- **CREATE, ALTER, DROP**

CREATE	DOMAIN	도메인을 생성
	TABLE	테이블을 생성
	VIEW	뷰를 생성
	INDEX	인덱스를 생성. SQL2의 표준이 아님
ALTER	TABLE	테이블의 구조를 변경
DROP	DOMAIN	도메인을 제거
	TABLE	테이블을 제거
	VIEW	뷰를 제거
	INDEX	인덱스를 제거. SQL2의 표준이 아님

5.2 DDL(Data Definition Language)

CREATE TABLE Statement

CREATE TABLE 구문

```
CREATE TABLE table_name  
(  
column_name1 data_type(size),  
column_name2 data_type(size),  
column_name3 data_type(size),  
....  
);
```

CREATE TABLE 예제

```
CREATE TABLE Persons  
(  
  PersonID int,  
  LastName varchar(255),  
  FirstName varchar(255),  
  Address varchar(255),  
  City varchar(255)  
);
```

5.2 DDL(Data Definition Language)

SQL 제약조건(Constraints)

CREATE TABLE 제약조건 구문

```
CREATE TABLE table_name
(
  column_name1 data_type(size), constraint_name,
  column_name2 data_type(size), constraint_name,
  column_name3 data_type(size), constraint_name,
  ....
);
```

제약 조건

NOT NULL - 열이 NULL 값을 저장할 수 없는 것을 지정

UNIQUE - 열의 값이 모든 행에 대하여 유일 해야함을 지정

PRIMARY KEY - 개체 무결성 지정(NOT NULL 과 UNIQUE)

FOREIGN KEY - 참조 무결성 지정

CHECK - 영역 무결성 지정(열의 값이 특정 조건을 만족하는지 확인)

DEFAULT - 열의 값이 지정되지 않을 경우 기본 값 지정

5.2 DDL(Data Definition Language)

NOT NULL Constraint

예) "Persons" 테이블의 "P_Id" 와 "LastName" 열은 널 값을 허용하지 않음을 지정

```
CREATE TABLE Persons
(
P_Id int NOT NULL ,
LastName varchar(255) NOT NULL ,
FirstName varchar(255),
Address varchar(255),
City varchar(255)
)
```

5.2 DDL(Data Definition Language)

UNIQUE Constraint – 단일 열

예) "Persons" 테이블의 "P_Id" 열을 기본키로 지정

MySQL:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL,
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255),
  UNIQUE (P_Id)
)
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL UNIQUE,
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255)
)
```

5.2 DDL(Data Definition Language)

UNIQUE Constraint – 다중 열

예) "Persons" 테이블의 "P_Id " 와 "LastName" 을 UNIQUE로 지정

MySQL/ SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
P_Id int NOT NULL,
LastName varchar(255) NOT NULL,
FirstName varchar(255),
Address varchar(255),
City varchar(255),
CONSTRAINT uc_PersonID UNIQUE (P_Id,LastName)
)
```

5.2 DDL(Data Definition Language)

PRIMARY KEY Constraint – 단일 열

예) "Persons" 테이블의 "P_Id" 열을 기본키로 지정

MySQL:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL,
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255),
  PRIMARY KEY (P_Id)
)
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL PRIMARY KEY,
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255)
)
```


5.2 DDL(Data Definition Language)

PRIMARY KEY Constraint – 복합 열

예) "Persons" 테이블의 "P_Id " 와 "LastName" 을 결합하여 기본키로 지정

MySQL/SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
```

```
(
```

```
P_Id int NOT NULL,
```

```
LastName varchar(255) NOT NULL,
```

```
FirstName varchar(255),
```

```
Address varchar(255),
```

```
City varchar(255),
```

```
CONSTRAINT pk_PersonID PRIMARY KEY (P_Id, LastName)
```

```
)
```

5.2 DDL(Data Definition Language)

FOREIGN Key Constraint – 단일 열

예) "Orders" 테이블의 "P_Id" 열을 외래키로 지정

MySQL:

```
CREATE TABLE Orders
(
  O_Id int NOT NULL,
  OrderNo int NOT NULL,
  P_Id int,
  PRIMARY KEY (O_Id),
  FOREIGN KEY (P_Id) REFERENCES
  Persons(P_Id)
)
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Orders
(
  O_Id int NOT NULL PRIMARY KEY,
  OrderNo int NOT NULL,
  P_Id int FOREIGN KEY REFERENCES
  Persons(P_Id)
)
```

5.2 DDL(Data Definition Language)

FOREIGN Key Constraint – 단일 열

예) "Orders" 테이블의 "P_Id" 열을 외래키로 지정

MySQL/ SQL Server / Oracle / MS Access:

```
CREATE TABLE Orders
(
  O_Id int NOT NULL,
  OrderNo int NOT NULL,
  P_Id int,
  PRIMARY KEY (O_Id),
  CONSTRAINT fk_PerOrders FOREIGN KEY (P_Id) REFERENCES Persons(P_Id)
)
```

5.2 DDL(Data Definition Language)

CHECK Constraint – 단일 열

예) "Persons" 테이블의 "P_Id" 열의 값이 0보다 커야한다.

MySQL:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL,
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255),
  CHECK (P_Id>0)
)
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL CHECK (P_Id>0),
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255)
)
```

5.2 DDL(Data Definition Language)

CHECK Constraint – 다중 열

예) "P_Id" 열의 값이 0보다 크고, "City" 값은 "Seoul"이어야 한다.

MySQL/ SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
```

```
(
```

```
P_Id int NOT NULL,
```

```
LastName varchar(255) NOT NULL,
```

```
FirstName varchar(255),
```

```
Address varchar(255),
```

```
City varchar(255),
```

```
CONSTRAINT chk_Person CHECK (P_Id>0 AND City='Seoul')
```

```
)
```

5.2 DDL(Data Definition Language)

DEFAULT Constraint

예) "Persons" 테이블의 "City" 열의 기본값을 "Seoul"로 한다.

MySQL/ SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
  P_Id int NOT NULL,
  LastName varchar(255) NOT NULL,
  FirstName varchar(255),
  Address varchar(255),
  City varchar(255) DEFAULT 'Seoul'
)
```

5.2 DDL(Data Definition Language)

DROP/ ALTER TABLE Statement

DROP TABLE 구문

DROP TABLE table_name

ALTER TABLE 구문(기존 테이블에 열을 삽입, 삭제, 수정)

**ALTER TABLE table_name
ADD column_name datatype**

**ALTER TABLE table_name
DROP COLUMN column_name**

SQL Server / MS Access:

ALTER TABLE table_name
ALTER COLUMN column_name datatype

My SQL / Oracle:

ALTER TABLE table_name
MODIFY COLUMN column_name datatype

Oracle 10G and later:

ALTER TABLE table_name
MODIFY column_name datatype

5.2 DDL(Data Definition Language)

CREATE INDEX Statement

CREATE INDEX 구문(중복 허용)

```
CREATE INDEX index_name  
ON table_name (column_name)
```

CREATE UNIQUE INDEX 구문(중복 불허)

```
CREATE UNIQUE INDEX index_name  
ON table_name (column_name)
```

CREATE INDEX 예제

```
CREATE INDEX PIndex  
ON Persons (LastName)
```

```
CREATE INDEX PIndex  
ON Persons (LastName, FirstName)
```


5.2 DDL(Data Definition Language)

DROP INDEX Statement

DROP INDEX 구문

MS Access: DROP INDEX index_name ON table_name

MS SQL Server: DROP INDEX table_name.index_name

DB2/Oracle: DROP INDEX index_name

MySQL: ALTER TABLE table_name DROP INDEX index_name