

ภาษา SQL ในการสร้างฐานข้อมูลและตาราง (ส่วน DDL)

1. สร้างฐานข้อมูล

รูปแบบประโยคการเขียนคำสั่ง (Syntax) ดังนี้

```
CREATE DATABASE databasename;
```

ตัวอย่าง CREATE DATABASE GeowebDB;

2. ลบฐานข้อมูล

รูปแบบประโยคการเขียนคำสั่งดังนี้

```
DROP DATABASE databasename;
```

ตัวอย่าง DROP DATABASE testDB;

3. สร้างตาราง

รูปแบบประโยคการเขียนคำสั่งดังนี้

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
);
```

table_name = ชื่อตารางที่ต้องการสร้าง
column1 datatype = ชื่อฟิลด์ที่ต้องการสร้าง
column2 datatype = ชื่อฟิลด์ที่ต้องการสร้าง
column3 datatype = ชื่อฟิลด์ที่ต้องการสร้าง

ตัวอย่าง มีความต้องการสร้างตารางเพื่อจัดเก็บข้อมูลเขตการปกครองระดับอำเภอ ได้ออกแบบแนวคิดในการจัดทำฐานข้อมูลดังนี้

อำเภอ(รหัสอำเภอ, ชื่ออำเภอ, รหัสจังหวัด)

Entity	Attribute	Data type	Length	Description	Constraint
Amphur	IdAmphur	Int	4	รหัสอำเภอ	Primary key
	Ampname	Text	40	ชื่ออำเภอ	
	IdProv	Int	2	รหัสจังหวัด	Foreign Key

เขียนคำสั่ง d11 เพื่อสร้างตารางและกำหนดค่าคีย์หลักและคีย์ Foreign ดังนี้

```
CREATE TABLE Amphur(  
    IdAmphur int(4),  
    Ampname Text(40),  
    IdProv int(2),  
);
```

4. การกำหนดฟิลด์ Primary key ในตาราง

รูปแบบประโยคการเขียนคำสั่งดังนี้

4.1 กรณีกำหนดใช้ 1 ฟิลด์

รูปแบบประโยค	ตัวอย่าง
<pre>CREATE TABLE table_name (column1 datatype, column2 datatype, column3 datatype, PRIMARY KEY (IDcolumn));</pre>	<pre>CREATE TABLE Amphur(IdAmphur int(4), Ampname Text(40), IdProv int(2), PRIMARY KEY (IdAmphur));</pre>

4.2 กรณีกำหนดใช้ 2 필ด์

รูปแบบประโยค

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    CONSTRAINT table_name PRIMARY KEY (ID,LastName)  
);
```

ตัวอย่าง

```
CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    CONSTRAINT PK_Person PRIMARY KEY (ID,LastName)  
);
```

5. การกำหนด Foreign Key

รูปแบบประโยคการเขียนคำสั่งดังนี้

1) กรณีทั่วไปของ MySQL:

```
CREATE TABLE Orders (  
    OrderID int NOT NULL,  
    OrderNumber int NOT NULL,  
    PersonID int,  
    PRIMARY KEY (OrderID),  
    FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)  
);
```

2) กรณีใช้ในโปรแกรมฐานข้อมูล SQL Server / Oracle / MS Access:

```
CREATE TABLE Orders (  
    OrderID int NOT NULL PRIMARY KEY,  
    OrderNumber int NOT NULL,  
    PersonID int FOREIGN KEY REFERENCES Persons(PersonID)  
);
```

ตัวอย่างการการเขียนคำสั่งที่นิยมใช้งานทั่วไป

```
CREATE TABLE Orders (  
    OrderID int NOT NULL,  
    OrderNumber int NOT NULL,  
    PersonID int,  
    PRIMARY KEY (OrderID),  
    CONSTRAINT FK_PersonOrder FOREIGN KEY (PersonID)  
    REFERENCES Persons(PersonID)  
);
```

6. การกำหนดลำดับเลขอัตโนมัติเรียงลำดับ (Auto Increment)

รูปแบบประโยคการเขียนคำสั่งดังนี้

1) รูปแบบประโยคใน MySQL

```
CREATE TABLE Persons (  
    ID int NOT NULL AUTO_INCREMENT,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    PRIMARY KEY (ID)  
);
```

2) รูปแบบประโยคใน MS Access

```
CREATE TABLE Persons (  
    ID Integer PRIMARY KEY AUTOINCREMENT,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int  
);
```

7. การปรับเปลี่ยนการออกแบบตารางหลังการสร้างด้วยคำสั่ง Create table (ALTER TABLE)

รูปแบบประโยคการเขียนคำสั่งดังนี้

```
ALTER TABLE table {ADD {COLUMN field type[(size)] [NOT NULL]  
[CONSTRAINT index] |  
ALTER COLUMN field type[(size)] |  
CONSTRAINT multifieldindex} |  
DROP {COLUMN field I CONSTRAINT indexname} }
```

1) กรณีเพิ่มฟิลด์ (Add Colmn)

```
ALTER TABLE table_name  
ADD column_name datatype;
```

2) กรณีลบฟิลด์ (DROP Colmn)

```
ALTER TABLE table_name  
DROP COLUMN column_name;
```