

แบบฝึกปฏิบัติการที่ 3

Data Definition Language (DDL)

วัตถุประสงค์

1. ฝึกปฏิบัติการสร้างฐาน (Create)
2. ฝึกปฏิบัติการลบฐาน (Drop)
3. ฝึกปฏิบัติการข้อมูลสร้างตาราง (Create) เพิ่ม ลบ แก้ไข เปลี่ยนชื่อ column (Alter)
4. ฝึกปฏิบัติในการเพิ่ม (Insert) ลบ (Delete) แก้ไข (Update)

คำสั่ง ในการสร้างฐานข้อมูล

```
Create Table table_name ( column_name_1 Data_Type_Name [ Default_Value]

                                [column_constraint], ...

                                column_name_n Data_Type_Name [ Default_Value]

                                [column_constraint]

[, PRIMARY KEY (column_name_x)]

[, FOREIGN KEY (column_name_x) REFERENCES Table_name_ref [ON DELETE , ON
UPDATE] {DELETE, CASCADE} ];
```

table_name :: ชื่อตารางที่ต้องการแก้ไขรายการข้อมูล

column_name_1, column_name_2, ... , column_name_n :: ชื่อคอลัมน์ที่ต้องการเป็นหัวตาราง

Data_Type_Name : ชื่อของ Data type ที่ต้องการจะเก็บในฐานข้อมูล เช่น int , char, varchar , date เป็นต้น

Column constraint : เงื่อนไขของ Column เช่น NULL, NOT NULL, UNIQUE เป็นต้น

PRIMARY KEY : Column ที่ประกาศนี้จะต้องมีค่าไม่ซ้ำกันทั้ง Table

FOREIGN KEY : Column ที่อ้างอิงไปยังอีก Column ต่าง Table กัน โดยสามารถกำหนดได้ว่า เมื่อมีการลบ หรือ แก้ไข Table ที่อ้างอิง สามารถลบและแก้ไขใน table นี้ได้ด้วยคำสั่ง CASCADE ได้ แต่หากไม่กำหนดหากมีการลบหรือแก้ไขใน Table ที่อ้างอิงก็จะไม่ยอมให้ลบหรือแก้ไขใดๆ

ตัวอย่าง สร้าง Table Employees ตามคุณสมบัติที่กำหนดให้

```
Person_id (char(3))
RateHour (int)
Person_name (varchar(100))
address (varchar(100))
BirthDate (date)
```

```
CREATE TABLE `persons` (
  `Person_id` char(3) NOT NULL,
  `RateHour` int(2) DEFAULT NULL,
  `Person_name` varchar(100) DEFAULT NULL,
  `address` varchar(100) DEFAULT NULL,
  `BirthDate` date DEFAULT NULL,
  PRIMARY KEY (`Person_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

คำสั่งลบตาราง

```
DROP TABLE table_name;
```

ตัวอย่าง drop table student ;

คำสั่งแก้ไขตารางโครงสร้าง ตาราง (ALTER)

```
ALTER TABLE table-Name {

    ADD COLUMN column-definition |

    ADD CONSTRAINT clause |

    DROP [ COLUMN ] column-name [ CASCADE | RESTRICT ]

    DROP { PRIMARY KEY | FOREIGN KEY constraint-name | UNIQUE

        constraint-name | CHECK constraint-name | CONSTRAINT constraint-name }
```

1 เพิ่ม column

```
ALTER TABLE table_name ADD column_name (char(1));
```

2 ลบ column

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

ตัวอย่าง ALTER TABLE student DROP advisor;

3 เพิ่ม table constraint.

```
ALTER TABLE table_name ADD constraintName column_name
```

ตัวอย่าง ALTER TABLE Client ADD prefNoRooms PRooms;

4 Drop Constraint

```
ALTER TABLE table_name DROP CONSTRAINT constraint_name
```

ตัวอย่าง ALTER TABLE student DROP CONSTRAINT StaffNotHandlingTooMuch

5 Set a default for a column.

```
ALTER TABLE table_name ALTER column_name SET DEFAULT value ;
```

ตัวอย่าง ALTER TABLE Staff ALTER sex SET DEFAULT 'F';

6 Drop a default for a column. (constraint)

ALTER TABLE table_name ALTER column_name DROP DEFAULT;

ตัวอย่าง ALTER TABLE Staff ALTER position DROP DEFAULT;

7 Rename table

ALTER TABLE Table_old_name RENAME Table_new_name

ALTER TABLE Table_old_name TO Table_new_name

ตัวอย่าง ALTER TABLE employee TO employees

8 Modify column ex data type

ALTER TABLE table_name MODIFY COLUMN column_name Number;

ตัวอย่าง ALTER TABLE employee MODIFY COLUMN age int(3);

9 Rename column

ALTER TABLE Table_name RENAME COLUMN old_col_name TO new_col_name

ตัวอย่าง ALTER TABLE employee RENAME COLUMN fname TO firstname

คำสั่ง Insert

```
INSERT INTO table_name (column_name_1, column_name_2, ... , column_name_n)
VALUES (value_1, value_2, ... , value_n)
```

column_name_1, column_name_2, ... , column_name_n :: ชื่อคอลัมน์ที่ต้องการบันทึก

value_1, value_2, ... , value_n :: ค่าของคอลัมน์ที่ต้องการบันทึก

หมายเหตุ

ค่าของ column ต้องระวังหากค่าที่ต้องการจะแก้ไขตัวหนังสือ (String) ให้กำหนดอยู่ในเครื่องหมาย ' ' หาก เป็นตัวเลขไม่ต้อง Number เช่น value = 'Test' หรือ value = 10

ตัวอย่าง

```
INSERT INTO persons (Person_id, RateHour, Person_name, address, BirthDate) VALUES ('001',1000,'นัชชา สีกัน','ท่าศาลา เมืองเหนือ','1975-01-03')
```

หากตำแหน่งของ value เรียงลำดับตามกับ column ในฐานข้อมูลจะสามารถใช้

```
Insert into Table_name VALUES (value_1, value_2, ... , value_n)
```

ตัวอย่าง

```
INSERT INTO `persons` VALUES ('001',1000,'นัชชา สีกัน','ท่าศาลา เมืองเหนือ','1975-01-03')
```

คำสั่ง Update

```
UPDATE table_name
```

```
SET column_name = value_1 [, column_name_2 = value_2 , ..., column_name_n = value_n]
```

```
[WHERE where clause];
```

table_name :: ชื่อตารางที่ต้องการแก้ไขรายการข้อมูล

column_name_1, column_name_2, ... , column_name_n :: ชื่อคอลัมน์ที่ต้องการแก้ไข

value_1, value_2, ... , value_n :: ค่าของคอลัมน์ที่ต้องการแก้ไข

where definition : เป็นเงื่อนไขที่จะเลือก row ที่ต้องการ

ตัวอย่าง ต้องการแก้ไขข้อมูลราคาชั่วโมงการทำงานเป็น 1000 บาทต่อ ชม. และเปลี่ยนนามสกุล

```
UPDATE persons
```

```
SET RateHour=1000, Person_name='นัชชา แต่งงานแล้ว'
```

WHERE Person_id='001';

หมายเหตุ

- ค่าของ column ต้องระวังหากค่าที่ต้องการจะแก้ไขตัวหนังสือ (String) ให้กำหนดอยู่ในเครื่องหมาย '' หากเป็นตัวเลขไม่ต้อง Number เช่น value = 'Test' หรือ value = 10
- where clause ต้องระวังหากค่าที่ต้องการเปรียบเทียบเป็นตัวหนังสือ (String) ให้กำหนดอยู่ในเครื่องหมาย '' หากเป็นตัวเลขไม่ต้อง Number เช่น value = 'Test' หรือ value = 10

คำสั่ง Delete

```
DELETE FROM Table_Name
```

```
[WHERE where definition];
```

table_name :: ชื่อตารางที่ต้องการแก้ไขรายการข้อมูล

where definition : เป็นเงื่อนไขที่จะเลือก row ที่ต้องการ

ตัวอย่าง ลบคนทีรหัส 001 เป็นต้น

```
DELETE FROM persons
```

```
WHERE Person_id='001';
```

แบบฝึกปฏิบัติ

สร้างฐานข้อมูล mysql> create database database_name;

หมายเหตุ database_name หมายถึงชื่อ ฐานข้อมูล

เรียกใช้ mysql> use database_name; (ทบทวนจากปฏิบัติการ 2)

1. ให้นักศึกษาสร้างตารางชื่อ Employee มีลักษณะของข้อมูลดังนี้

Person_id (char(3))
 RateHour (int)
 Person_name (varchar(100))
 address (varchar(100))
 BirthDate (date)

2. ให้แก้ไขชื่อ table เป็น Employees
3. ให้เพิ่มและแก้ไข column ต่างๆให้มีคุณสมบัติดังนี้

Employee_id (char(4))
 First_name (varchar(20))
 Last_name (varchar(25))
 Email (varchar(25))
 Phone_number (varchar(20))
 Hire_date (date)
 Job_id (varchar(10))
 Salary (float)
 Commision_pct (float)
 Manager_id (char(4))
 Department_id (int)
 Ratehour (int)
 Address (varchar(100))
 Birthdate (date)

4. ให้เพิ่มข้อมูลดังนี้ในตาราง

ABC Employee_id	ABC First_name	ABC Last_name	123 Department_id	123 Ratehour	ABC Address	Birthdate
1	Natcha	Srikun	1,000	1,000	Tasala Muangnear	1975-01-03
3	Panita	Narak	1,000	1,500	Samoon Maunglai	1974-01-06
4	Auschara	Bunchouy	1,000	2,500	Suksai Samosorn	1972-01-01
5	Natee	Deeharm	1,000	1,500	Silpakorn Tappitak	1979-05-01
6	SomSkul	Deeharm	1,000	1,000	Tasala Muangnear	1985-01-01
7	Pethan	Deeharm	1,000	1,000	Samoon Muangthai	1980-07-01
8	Aumnath	Khangkrang	1,000	3,000	Tasung Maungtong	1977-06-01

ส่งอาจารย์ พิมพ์คำสั่ง Select * from employees;

5. ให้เปลี่ยน Department_id จาก int เป็น char(1) นักศึกษาต้องคิดเองว่าจะทำอย่างไร
6. แก้ไขให้ Department_id ของ employee_id 1,3,4 มีรหัส Department เป็น 4 และคนที่ 5,6,7,8 เป็น 2
7. ลบคนที่ชื่อ Aumnath ทิ้ง
8. แสดงข้อมูลของพนักงานทั้งหมด ที่มีที่อยู่ที่ตำบล Tasala Mauangthai