

ปฏิบัติการที่ 8 MySQL Views and MySQL Transaction

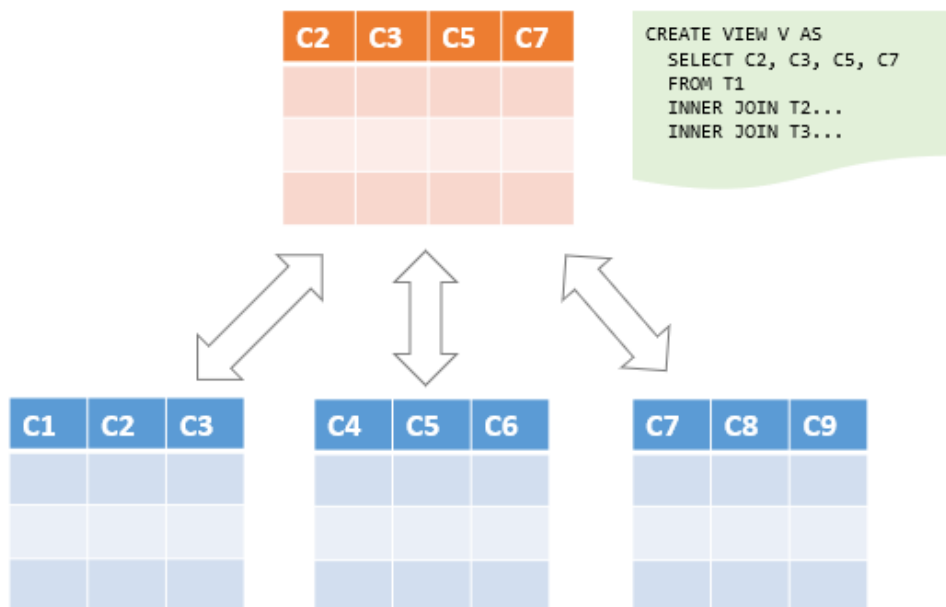
จุดประสงค์การปฏิบัติการ

- นักศึกษาสามารถสร้างและจัดการ View ได้
- นักศึกษาสามารถจัดการ Transaction ได้

ไฟล์ที่จำเป็น : <https://bit.ly/3cp72IA>

1. MySQL Views

เป็นการสร้างโครงสร้างข้อมูลใหม่โดยโครงสร้างวิว จะเป็นโครงสร้างที่ไม่ได้จัดเก็บข้อมูลจริง เป็นการนำข้อมูลในตารางมาแสดง เป็นเพียงตารางเสมือน เป็นการป้องกันผู้ใช้งานฐานข้อมูลไม่เห็นโครงสร้างข้อมูลทั้งหมดของฐานข้อมูล โดยสามารถกำหนดให้แสดงข้อมูลเพียงบางส่วน หรือแสดงข้อมูลหลาย ๆ ตารางในวิวเดียวทำให้ผู้ใช้งานเข้าใจข้อมูลได้ง่ายขึ้น อีกทั้งยังสามารถกำหนดสิทธิให้ผู้ใช้งานบางท่านสามารถ



- คำสั่งสร้าง View

```
CREATE [OR REPLACE] VIEW [db_name.]view_name [(column_list)]  
AS  
select-statement;
```

ตัวอย่างการสร้าง View

```
CREATE VIEW account_customer_view AS
SELECT
    a.account_number, a.branch_name, c.customer_name, c.customer_city
FROM
    account a INNER JOIN depositor d INNER JOIN customer c
ON
    a.account_number = d.account_number
AND d.customer_name = c.customer_name
```

เรียกใช้งาน View

```
SELECT * FROM account_customer_view
```

ผลลัพธ์

Result Grid				
		Filter Rows:	Search	Export:
	account_nu...	branch_name	customer_name	customer_city
▶ 1		SUT	Som	Korat
3		Mall	Dang	Korat

- คำสั่งเปลี่ยนชื่อ View

```
RENAME TABLE original_view_name
TO new_view_name;
```

ตัวอย่างเปลี่ยนชื่อ View

```
RENAME TABLE account_customer_view
TO new_account_customer_view;
```

ผลลัพธ์

Action Output			
	Time	Action	Response
✓ 1	22:04:10	RENAME TABLE account_customer_view TO new_account_customer_view	0 row(s) aff

- คำสั่งดู Views ในระบบ

```
SHOW FULL TABLES
[{FROM | IN } database_name]
WHERE table_type = 'VIEW';
```

ตัวอย่างดู Views ในระบบ

```
SHOW FULL TABLES IN db_pre_lab8
WHERE table_type='VIEW';
```

ผลลัพธ์

Result Grid			Filter Rows: <input type="text" value="Search"/>	Export:
Tables_in_db_pre_lab8-2	Table_type			
▶ new_account_customer_view	VIEW			

- คำสั่งลบ View

```
DROP VIEW [IF EXISTS] view_name1 [,view_name2]...;
```

ตัวอย่างลบ View

```
DROP VIEW new_account_customer_view;
```

ผลลัพธ์

Action Output			
	Time	Action	Response
✓ 1	22:08:06	DROP VIEW new_account_customer_view	0 row(s) aff

- คำสั่ง With Check Option

```
CREATE [OR REPLACE] VIEW view_name
AS
    select_statement
WITH CHECK OPTION;
```

ตัวอย่าง With Check Option

```
CREATE OR REPLACE VIEW customer_view AS
SELECT
    *
FROM
    customer
WHERE
    customer_city = 'Korat'
WITH CHECK OPTION;
```

ทดลองเพิ่มข้อมูล

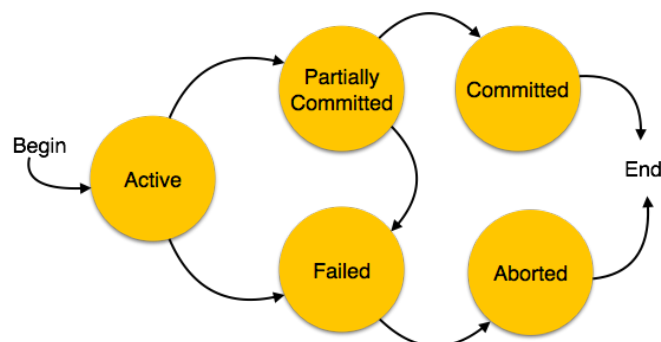
```
INSERT INTO customer_view
VALUES('Dan', 'Mitr-Pab', 'NakhonRatchasima');
```

ผลลัพธ์

Action Output			
	Time	Action	Response
✖ 1	22:13:59	INSERT INTO customer_view VALUES('Da...	Error Code: 1369. CHECK OPTION failed 'db_pre_lab8-2.customer_vie

2. MySQL Transaction

การใช้ Transaction บน Stored Procedure คือ การจัดการกับความถูกต้องในการทำงานของ SQL Statement ด้วย Transaction โดยสามารถที่จะทำการ เปิด SAVEPOINT เพื่อที่จะ ROLLBACK ไปยัง SAVEPOINT นั้นได้ ซึ่งประโยชน์ คือ จะสามารถควบคุมการทำงานของ Query ที่เกิดขึ้น ในกรณีที่มีการ Error หรือเกิดเงื่อนไขที่ไม่ต้องการ ทำให้สามารถทำการ ROLLBACK ย้อนกลับข้อมูลที่เกิดขึ้นระหว่างการทำงาน เช่น INSERT, UPDATE และ DELETE ให้กลับมายังจุดก่อนที่ Stored Procedure ที่ทำได้



- คำสั่งจัดการ Transaction

```
CREATE PROCEDURE stored_procedure_name
    ([IN | OUT | INOUT] parameter_name datatype[(length)])
BEGIN
    DECLARE errorStatus BOOLEAN DEFAULT FALSE;

    START TRANSACTION;
    BEGIN
        DECLARE EXIT HANDLER FOR SQLEXCEPTION SET errorStatus = TRUE;
        Statement;
    END;

    IF errorStatus = TRUE THEN
        ROLLBACK;
    ELSE
        COMMIT;
    END IF;
END
```

ตัวอย่างการจัดการ Transaction

```
CREATE PROCEDURE `INSERT_NEW_CUSTOMER` (
    IN account_number INT,
    IN branch_name CHAR(9),
    IN balance FLOAT,
    IN customer_name CHAR(9),
    IN customer_street VARCHAR(20),
    IN customer_city VARCHAR(20)
)
BEGIN
    DECLARE errorStatus BOOLEAN DEFAULT FALSE;

    START TRANSACTION;
    BEGIN
        DECLARE EXIT HANDLER FOR SQLEXCEPTION SET errorStatus = TRUE;
        INSERT INTO ACCOUNT VALUES (account_number, branch_name, balance);
        INSERT INTO CUSTOMER VALUES (customer_name , customer_street, customer_city);
        INSERT INTO DEPOSITOR VALUES (customer_name , account_number);
    END;

    IF errorStatus = TRUE THEN
        ROLLBACK;
    ELSE
        COMMIT;
    END IF;
END
```

ให้ทำการรัน Store Procedure

```
CALL `INSERT_NEW_CUSTOMER`(2,'SUT',200,'Dan', 'Mitr-Pab', 'Korat');
```

และ

```
CALL `INSERT_NEW_CUSTOMER`(3,'SUT',300,'Manee', 'Mitr-Pab', 'Korat');
```

สังเกตผลลัพธ์