

# การออกแบบฐานข้อมูลให้มีประสิทธิภาพ

---

## หลักการของการนอร์มัลไลซ์ (Normalize)

- เป็นวิธีการที่ทำให้ข้อมูลนำไปใช้งานได้ง่าย
- ลดความซ้ำซ้อนกันของข้อมูล ซึ่งทำให้เกิดปัญหาในการแก้ไข เพิ่มและลบข้อมูลทำได้ยาก

# ฟังก์ชันการขึ้นต่อกัน (Functional Dependency)

คือการแสดงให้เห็นว่าแอตทริบิวต์ต่าง ๆ มีความสัมพันธ์กัน  
อย่างไร โดยเมื่อทราบค่าแอตทริบิวต์ **A** แล้วบอกค่าแอตทริ  
บิวต์ **B,C** .. ได้

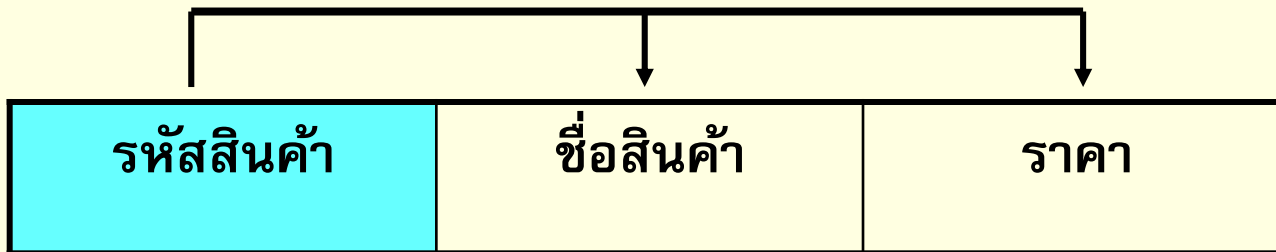
แสดงว่า **B,C** มีฟังก์ชันการขึ้นต่อกันกับแอตทริบิวต์  
**A** หรือกล่าวง่าย ๆ ว่า **B,C** ขึ้นกับ **A**



# ฟังก์ชันการขึ้นต่อกัน (Functional Dependency)

รหัสสินค้า	ชื่อสินค้า	ราคา
1111	ปากกา	15
2222	ดินสอ	10

ไดอะแกรมการขึ้นต่อกัน (Dependency Diagram)



สัญลักษณ์แบบข้อความ

รหัสสินค้า  $\longrightarrow$  ชื่อสินค้า, ราคา

# ชนิดของฟังก์ชันการขึ้นต่อกัน

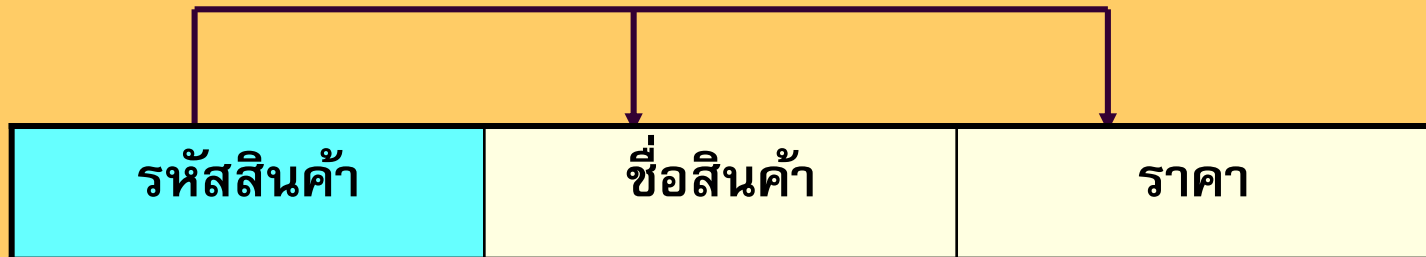
ในที่นี้กล่าวถึง 3 ลักษณะ

- ▶ การขึ้นต่อกันแบบทั้งหมด (Full dependency)
- ▶ การขึ้นต่อกันแบบบางส่วน (Partial dependency)
- ▶ การขึ้นต่อกันแบบทรานซิทีฟ (Transitive dependency)

# การขึ้นต่อกันแบบทั้งหมด (Full Dependency)

- ▶ ทุกแอตทริบิวต์ที่ไม่ใช่คีย์ขึ้นกับคีย์หลักทั้งหมด

รหัสสินค้า	→	ชื่อสินค้า
รหัสสินค้า	→	ราคา
รหัสสินค้า	→	ชื่อสินค้า, ราคา

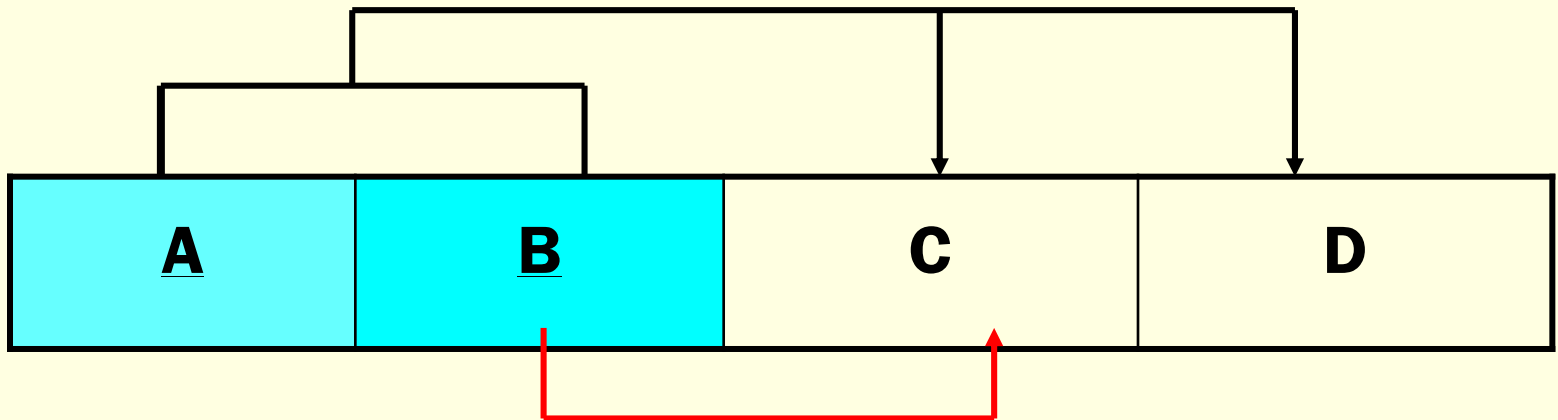


# การขึ้นต่อกันแบบบางส่วน (Partial dependency)

- ▶ แอตทริบิวต์ที่ไม่ใช่คีย์ขึ้นอยู่กับบางส่วนของคีย์หลัก
- ▶ มักเกิดกับการใช้แอตทริบิวต์ร่วมกันสองตัวเป็นคีย์หลัก

FD1 : A,B → C, D

FD2 : B → C

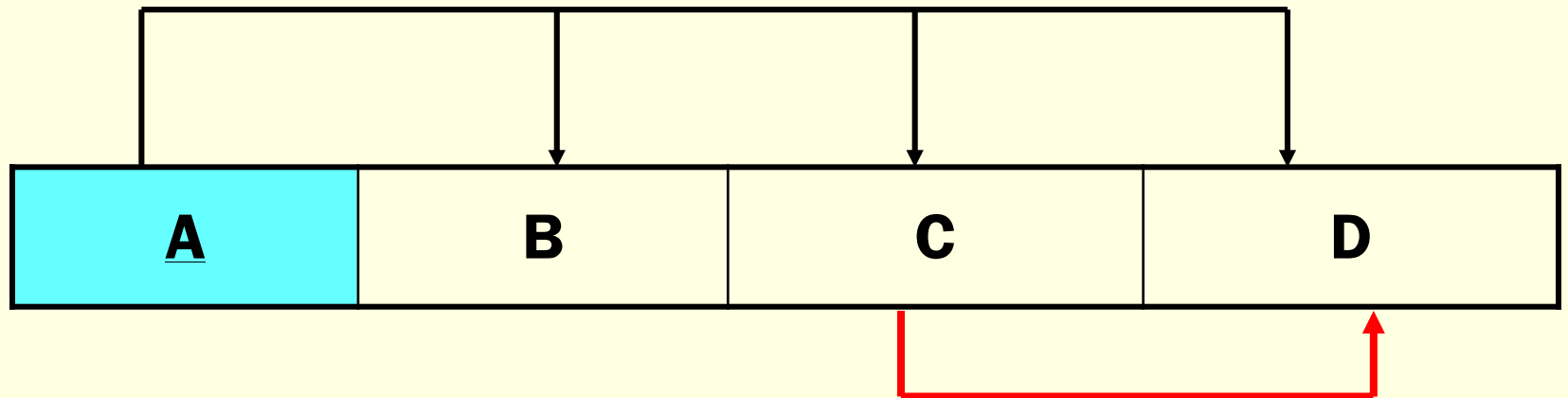


# การขึ้นต่อกันแบบทรานซิทีฟ (Transitive Dependency)

▶ แอตทริบิวต์ที่ไม่ใช่คีย์หลัก สามารถระบุค่าแอตทริบิวต์อื่น

FD1 : A → B,C,D

FD2 : C → D



# เทคนิคในการออกแบบตารางหรือรีเลชัน

เทคนิค 1 : ชื่อและค่าของแอตทริบิวต์ในแต่ละตารางต้องชัดเจน

เทคนิค 2 : ควรออกแบบให้ค่าของแอตทริบิวต์เป็นค่า **Null** น้อยที่สุด หรือไม่มีเลย

เทคนิค 3 : ไม่ออกแบบให้มีค่าแอตทริบิวต์ซ้ำซ้อนกัน

เทคนิค 4 : ถ้ามีการแตกตารางเดิมออกให้เป็นหลายตาราง เมื่อทำการ **Join** กัน ค่าของแอตทริบิวต์ทุกตัวต้องถูกต้องเหมือนเดิม จำนวนรายการเท่าเดิม



# แนวคิดเกี่ยวกับนอร์มัลไลเซชัน

นอร์มัลไลเซชัน (Normalization) เป็นกระบวนการวิเคราะห์ความสัมพันธ์ระหว่างแอตทริบิวต์ในตารางข้อมูล เพื่อตรวจสอบและแก้ไขปัญหาด้านความซ้ำซ้อนของข้อมูล ด้วยการแตกตารางข้อมูลที่มีความซ้ำซ้อนของข้อมูลออกเป็นตารางย่อย ๆ เพื่อขจัดความซ้ำซ้อนของข้อมูลให้เหลือน้อยที่สุด หรือจนกระทั่งไม่มีความซ้ำซ้อนเลย

## วัตถุประสงค์ของนอร์มัลไลเซชัน

**ลดความซ้ำซ้อนของข้อมูล** หากมีข้อมูลซ้ำซ้อนกัน เมื่อมีการแก้ไขเปลี่ยนแปลงข้อมูลจะต้องทำการเปลี่ยนแปลงแก้ไขให้ครบทุกแห่ง มิฉะนั้นอาจทำให้ข้อมูลไม่ถูกต้องหรือไม่สอดคล้องกันได้

**ประหยัดเนื้อที่ในการจัดเก็บข้อมูล** เมื่อข้อมูลที่จัดเก็บไม่ซ้ำซ้อนกัน จึงใช้เนื้อที่ในการจัดเก็บข้อมูลน้อยลง

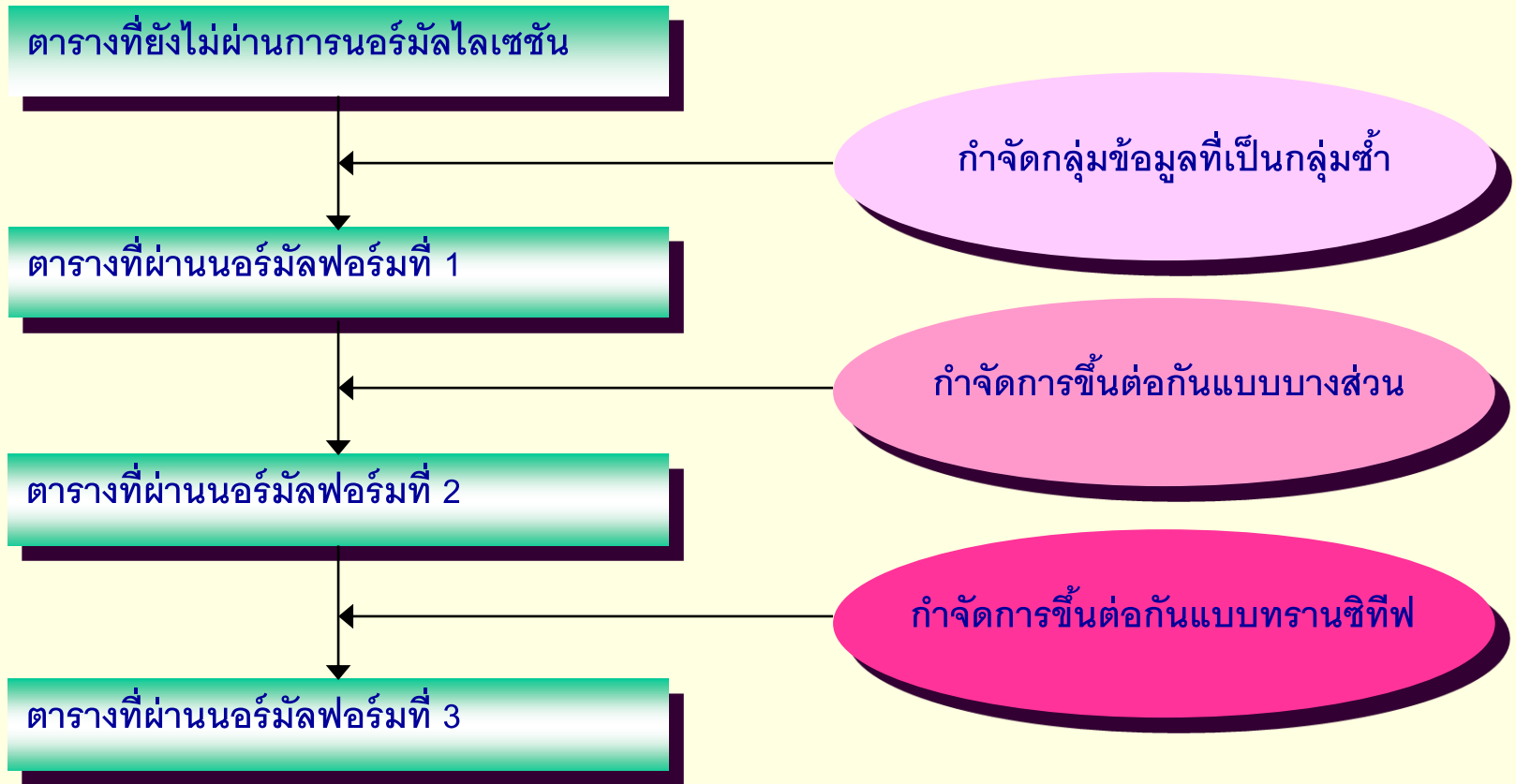
# กระบวนการนอร์มัลไลเซชัน

เป็นการดำเนินการนอร์มัลไลเซชันอย่างเป็นระดับขั้นตอน เพื่อจัดการให้ตารางข้อมูลมีคุณสมบัติอยู่ในนอร์มัลฟอร์มแบบต่าง ๆ ซึ่งประกอบด้วย 6 ระดับ คือ

- ▶ นอร์มัลฟอร์มที่ 1 หรือ 1NF (First Normal Form)
- ▶ นอร์มัลฟอร์มที่ 2 หรือ 2NF (Second Normal Form)
- ▶ นอร์มัลฟอร์มที่ 3 หรือ 3NF (Third Normal Form)
- ▶ นอร์มัลฟอร์มแบบบอยซ์คอดด์ หรือ BCNF (Boyce-Codd Normal Form)
- ▶ นอร์มัลฟอร์มที่ 4 หรือ 4NF (Fourth Normal Form)
- ▶ นอร์มัลฟอร์มที่ 5 หรือ 5NF (Fifth Normal Form)

(แต่ในที่นี้จะกล่าวถึงเพียง 3 ระดับแรกเท่านั้น)

# กระบวนการนอร์มัลไลเซชัน



## คุณสมบัติของนอร์มัลฟอร์มที่ 1 (First Normal Form)

---

ตารางข้อมูลใด ๆ มีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 1 ก็ต่อเมื่อ

- ▶ ค่าของแอตทริบิวต์ต่าง ๆ ในแต่ละตารางมีค่าของข้อมูลเป็นค่าเดี่ยว ๆ
- ▶ ข้อมูลในตารางนั้นต้องไม่มีข้อมูลที่เป็นกลุ่มซ้ำ

## ตารางข้อมูลที่ยังไม่ผ่านนอร์มัลฟอร์มที่ 1

SUPPLIER_ID	SUPPLIER_NAME	CITY	REGION	PRODUCT_ID	PRODUCT_NAME	PRODUCT_COLOR	QUANTITY
S01	STOU FURNITURE	กรุงเทพมหานคร	กลาง	P01	โต๊ะ	น้ำเตา	10
S01	STOU FURNITURE	กรุงเทพ	กลาง	P02	เก้าอี้	ขาว	40
S01	STOU FURNITURE	กรุงเทพ	กลาง	P03	ตู้	ดำ	20
S02	ศรีเฟอร์นิเจอร์	เชียงใหม่	เหนือ	P01	โต๊ะ	น้ำเตา	40
S02	ศรีเฟอร์นิเจอร์	เชียงใหม่	เหนือ	P04	เตียง	ฟ้า	5
S02	ศรีเฟอร์นิเจอร์	เชียงใหม่	เหนือ	P05	ชั้นวางหนังสือ	ครีม	15
S03	มสธ. เฟอร์นิเจอร์	สงขลา	ใต้	P01	โต๊ะ	น้ำเตา	20
S03	มสธ. เฟอร์นิเจอร์	สงขลา	ใต้	P02	เก้าอี้	ขาว	10
S03	มสธ. เฟอร์นิเจอร์	สงขลา	ใต้	P04	เตียง	ฟ้า	15
S04	สุโขทัยเฟอร์นิเจอร์	เชียงใหม่	เหนือ	P01	โต๊ะ	น้ำเตา	20
S05	ธรรมาเฟอร์นิเจอร์	กรุงเทพ	กลาง	P05	ชั้นวางหนังสือ	ครีม	30

## ผลของตารางข้อมูลที่ผ่านนอร์มัลฟอร์มที่ 1 แล้ว

SUPPLIER_ID	SUPPLIER_NAME	CITY	REGION
S01	STOU Furniture	กรุงเทพ	กลาง
S02	คริสตเฟอร์นิเจอร์	เชียงใหม่	เหนือ
S03	มสธ.เฟอร์นิเจอร์	สงขลา	ใต้
S04	สุโขทัยเฟอร์นิเจอร์	เชียงใหม่	เหนือ
S05	ธรรมาเฟอร์นิเจอร์	กรุงเทพ	กลาง

SUPPLIER_ID	PRODUCT_ID	PRODUCT_NAME	PRODUCT_COLOR	QUANTITY
S01	P01	โต๊ะ	น้ำตาล	10
S01	P02	เก้าอี้	ขาว	40
S01	P03	ตู้	ดำ	20
S02	P01	โต๊ะ	น้ำตาล	40
S02	P04	เตียง	ฟ้า	5
S02	P05	ชั้นวางหนังสือ	ครีม	15
S03	P01	โต๊ะ	น้ำตาล	20
S03	P02	เก้าอี้	ขาว	10
S03	P04	เตียง	ฟ้า	15
S04	P01	โต๊ะ	น้ำตาล	20
S05	P05	ชั้นวางหนังสือ	ครีม	50

## คุณสมบัติของนอร์มัลฟอร์มที่ 2 (Second Normal Form)

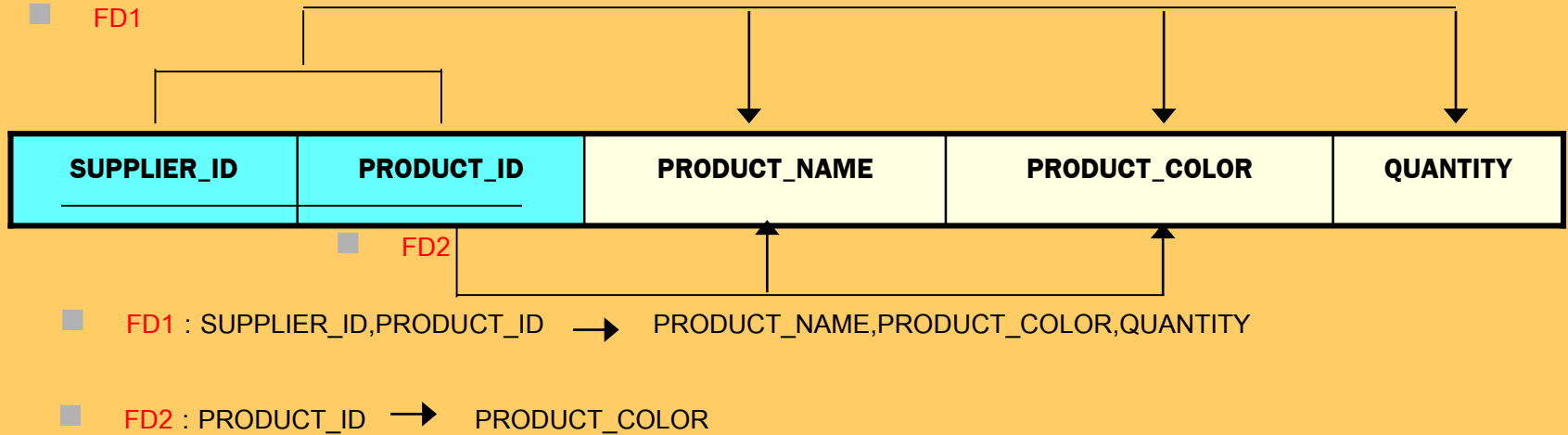
---

ตารางข้อมูลใด ๆ มีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 2 ก็ต่อเมื่อ

- ▶ ตารางข้อมูลนั้นมีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 1 แล้ว
- ▶ แอตทริบิวต์ที่ไม่ใช่คีย์หลัก ต้องขึ้นกับคีย์หลักอย่างแท้จริง หรือกล่าวได้ว่าแอตทริบิวต์ที่ไม่ใช่คีย์หลักจะต้องไม่ขึ้นกับส่วนใดส่วนหนึ่งของคีย์หลัก

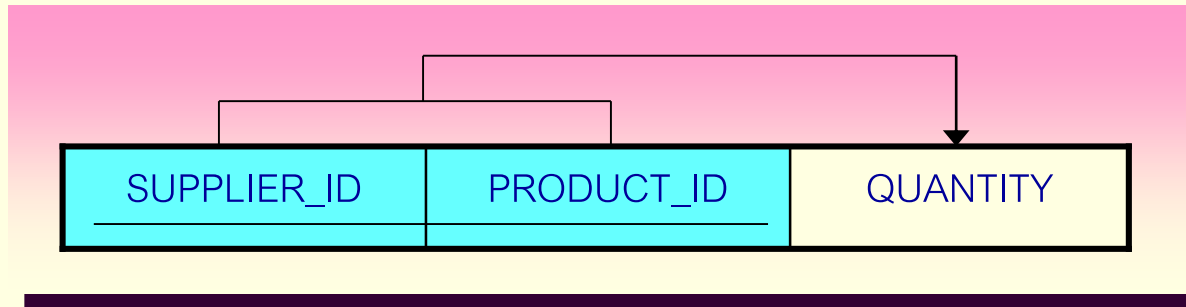


## ตารางข้อมูลที่ยังไม่ผ่านนอร์มัลฟอร์มที่ 2

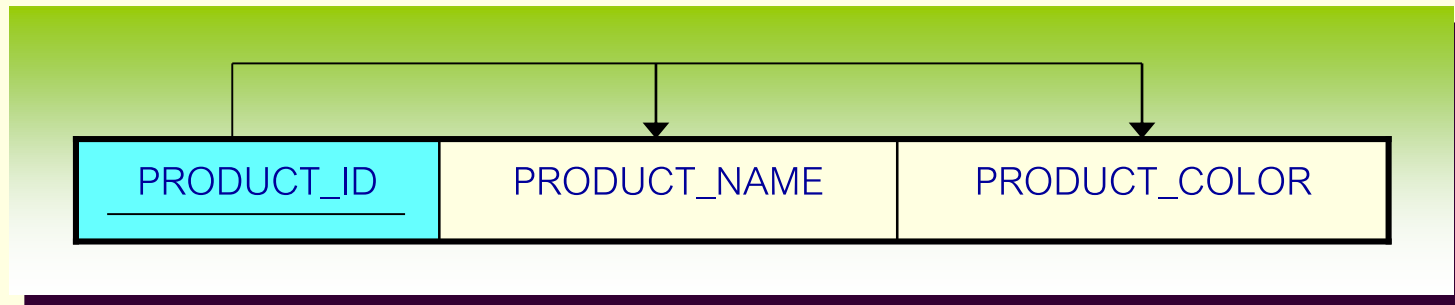


- คีย์หลักประกอบด้วย 2 แอตทริบิวต์ คือ SUPPLIER\_ID และ PRODUCT\_ID จะเห็นว่า
- ▶ QUANTITY เป็นแอตทริบิวต์ที่ขึ้นกับ SUPPLIER\_ID และ PRODUCT\_ID
- ▶ แต่ PRODUCT\_NAME และ PRODUCT\_COLOR เป็นแอตทริบิวต์ที่ขึ้นกับ PRODUCT\_ID เท่านั้น ไม่ได้ขึ้นกับ SUPPLIER\_ID เลย

## ■ ตารางข้อมูลผ่านนอร์มัลฟอร์มที่ 2 แล้ว



■ FD1 : SUPPLIER\_ID,PRODUCT\_ID → QUANTITY



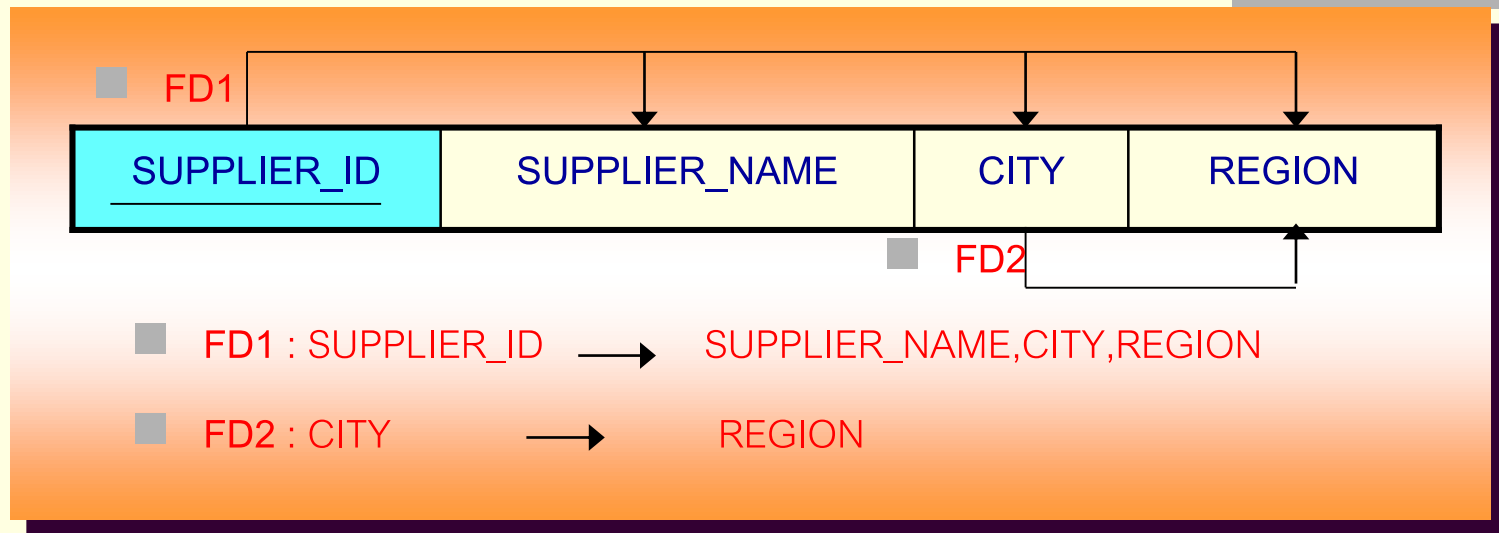
■ FD2 : PRODUCT\_ID → PRODUCT\_NAME,PRODUCT\_COLOR

## คุณสมบัติของนอร์มัลฟอร์มที่ 3 (Third Normal Form)

ตารางข้อมูลใดๆ มีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 3 ก็ต่อเมื่อ

- ▶ ตารางข้อมูลนั้นมีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 2
- ▶ แอตทริบิวต์ที่ไม่ใช่คีย์ จะต้องไม่สามารถระบุค่าของแอตทริบิวต์อื่นที่ไม่ใช่คีย์หลักในตารางข้อมูลนั้น

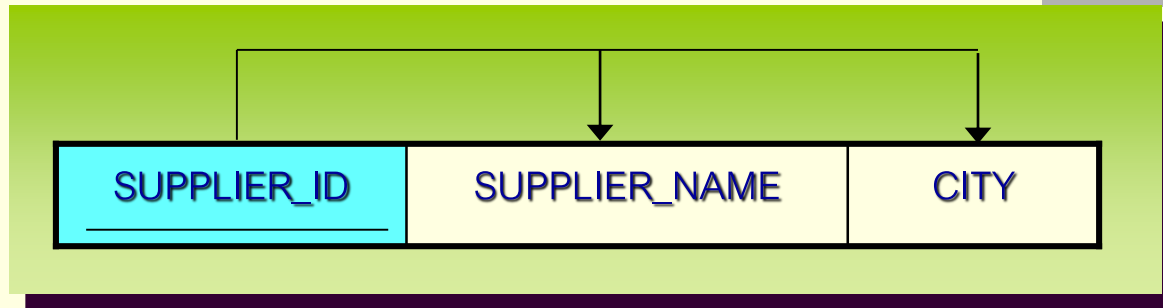
## ตารางข้อมูลที่ยังไม่ผ่านนอร์มัลฟอร์มที่ 3



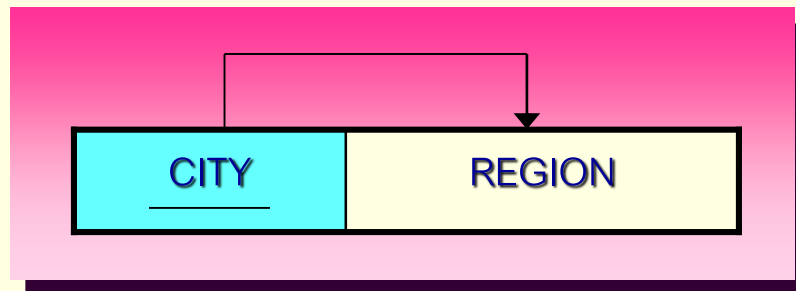
ตรวจสอบตารางว่ามีการขึ้นต่อกันแบบทรานซิทีฟหรือไม่ จะเห็นว่า

- ▶ ตารางนี้มีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 2
- ▶ แต่เมื่อทราบค่าของ CITY จะทำให้ทราบค่า REGION จึงยังไม่มีคุณสมบัติอยู่ในนอร์มัลฟอร์มที่ 3

- ตารางข้อมูลที่ผ่านมาในรูปแบบที่ 3 แล้ว



- **FD1** : SUPPLIER\_ID  $\longrightarrow$  SUPPLIER\_NAME,CITY



- **FD2** : CITY  $\longrightarrow$  REGION