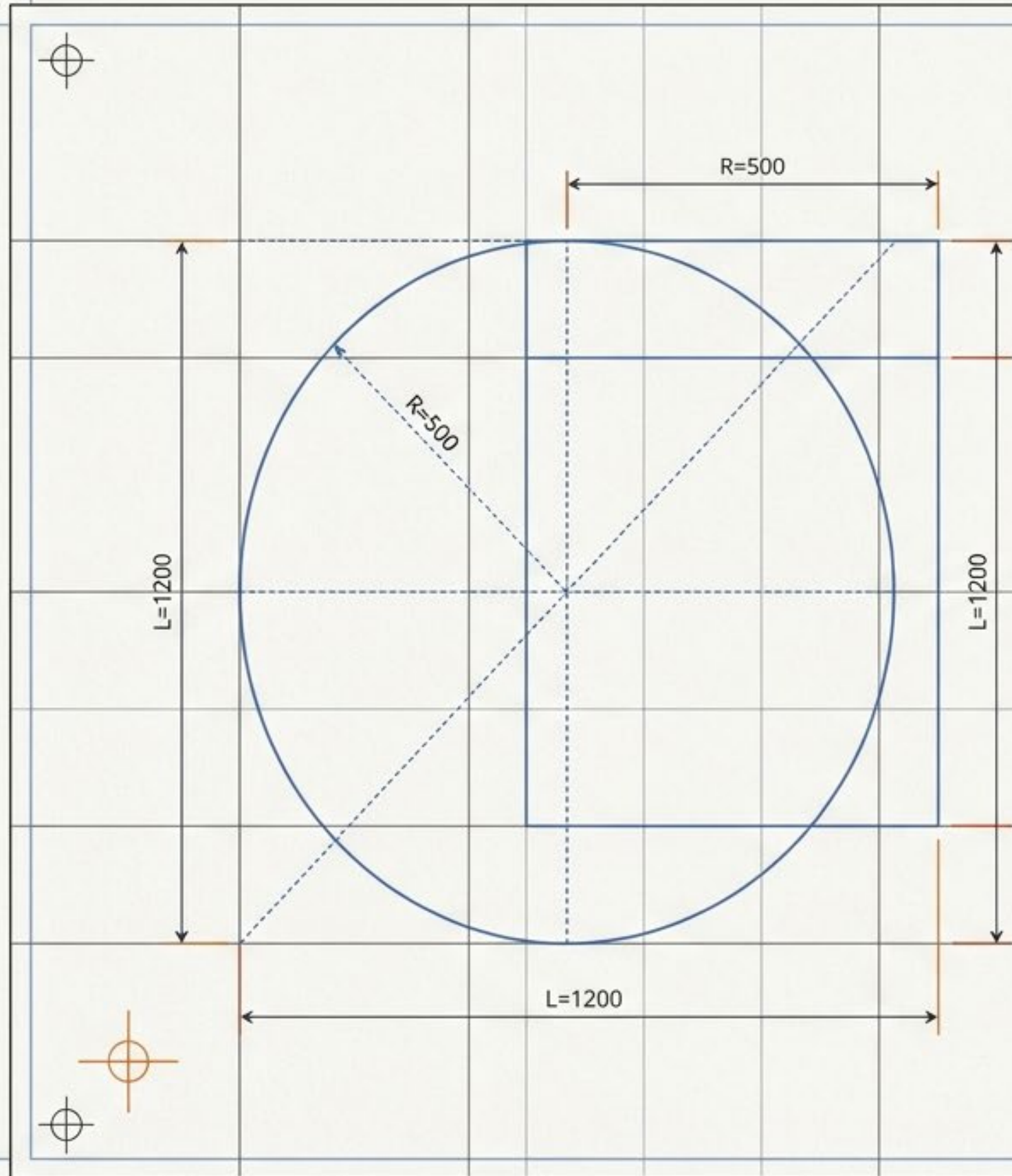


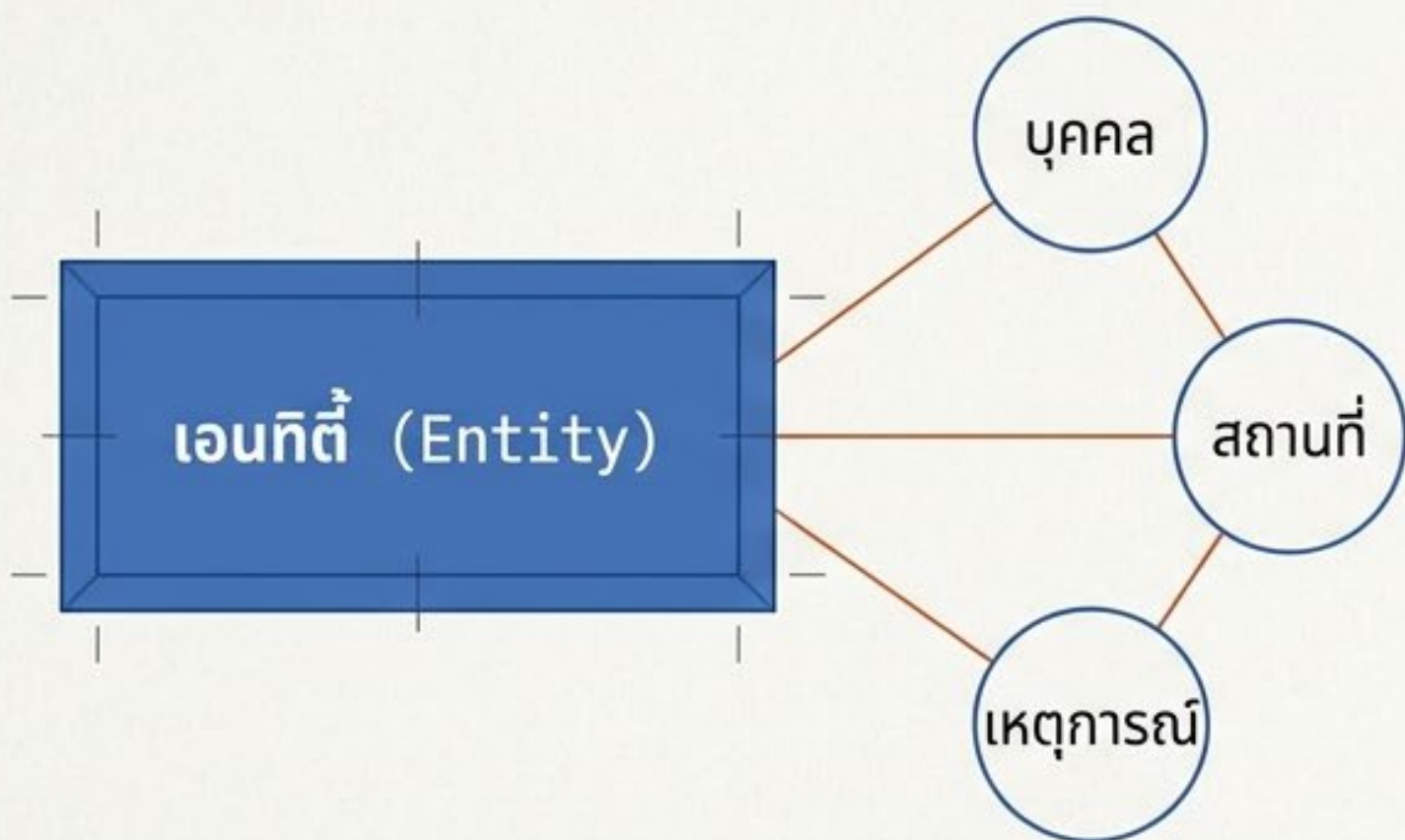


# สถาปัตยกรรมข้อมูล: พิมพ์เขียวฐานข้อมูล เชิงสัมพันธ์

Data Architecture: The  
Relational Database Blueprint

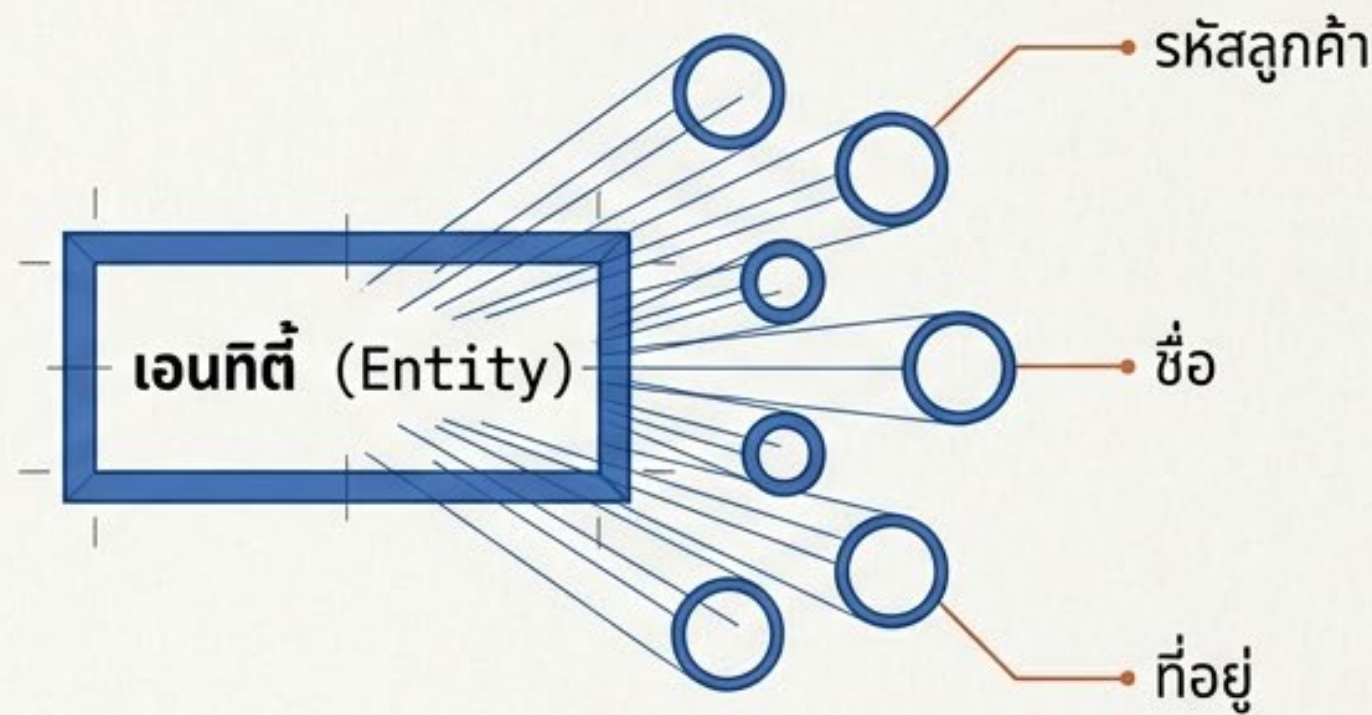
คู่มือฉบับภาพสำหรับการออกแบบและโครงสร้างระบบฐานข้อมูล

# รากฐานของแบบจำลองข้อมูล (The Foundations of Data Modeling)



**เอนทิตี:** บุคคล สถานที่ สิ่งของ หรือเหตุการณ์  
(เช่น ลูกค้า, เทียบิน)

## แอตทริบิวต์ (Attribute)



**แอตทริบิวต์:** คุณลักษณะของเอนทิตี

## แบบจำลองเชิงแนวคิด (Conceptual Data Models)

การแสดงผลภาพรวมของเอนทิตีและความสัมพันธ์  
ไม่ขึ้นกับระบบจัดการฐานข้อมูล (DBMS) ใดๆ

## แบบจำลองเพื่อการนำไปใช้ (Implementation Data Models)

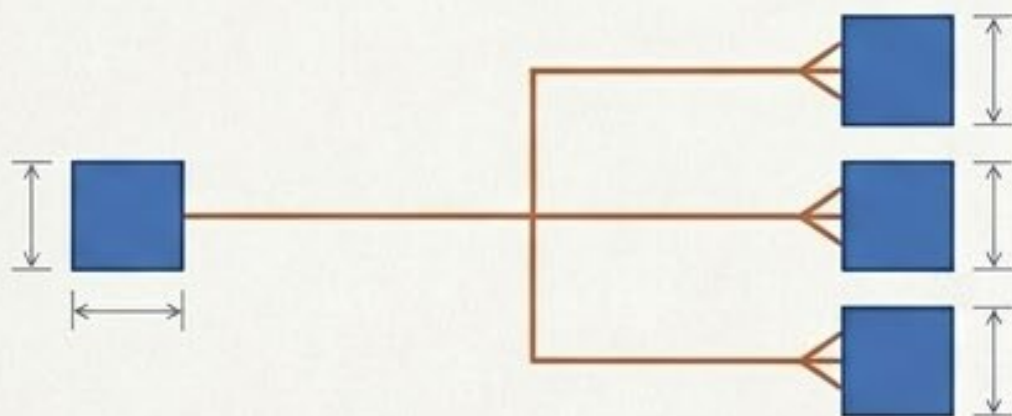
โครงสร้างที่อิงกับระบบฐานข้อมูลที่ใช้  
(เช่น ฐานข้อมูลเชิงสัมพันธ์)

# สถาปัตยกรรมการเชื่อมโยง 3 รูปแบบ (The 3 Connection Architectures)



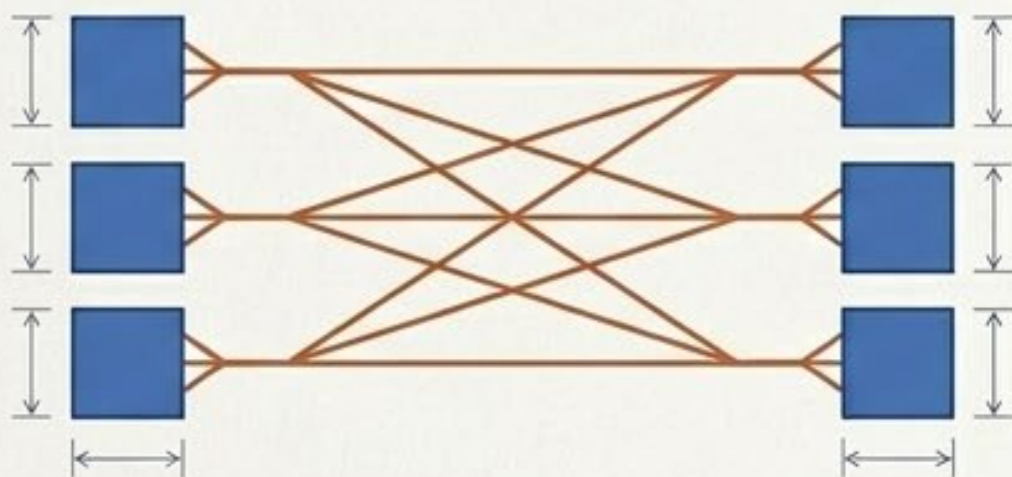
## หนึ่งต่อหนึ่ง (1:1)

รหัสนักศึกษา 1 รหัส  
อ้างอิงถึงนักศึกษาได้ 1 คนเท่านั้น



## หนึ่งต่อกลุ่ม (1:M)

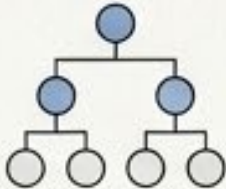
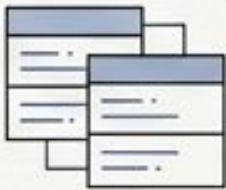
1 คณะ (เช่น คณะวิศวกรรมศาสตร์)  
มีได้หลายสาขาวิชา



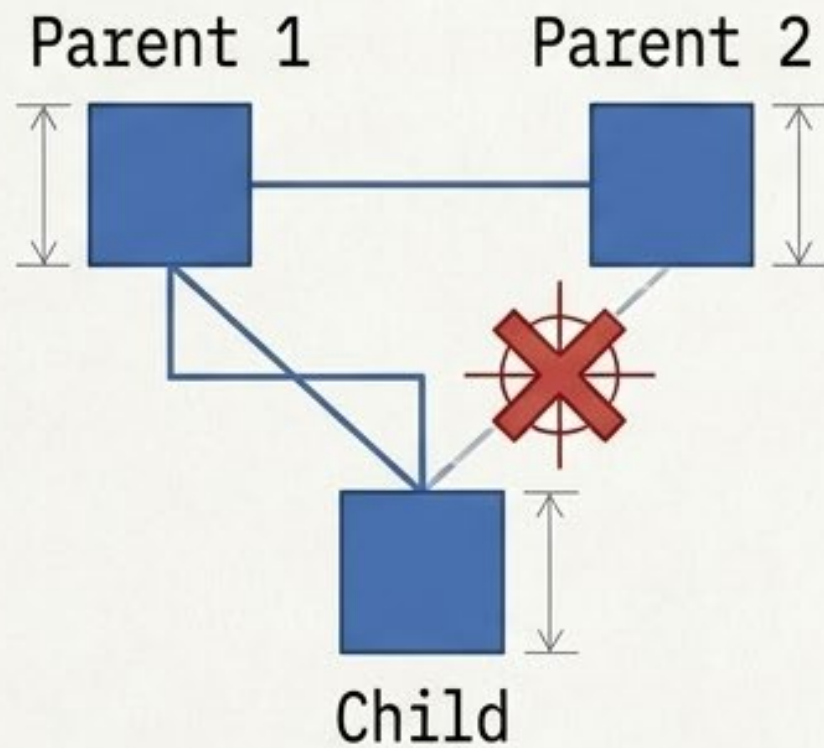
## กลุ่มต่อกลุ่ม (M:N)

นักศึกษา 1 คนลงทะเบียนได้หลายวิชา  
และ 1 วิชา มีนักศึกษาหลายคน

# เมทริกซ์วิวัฒนาการแบบจำลองฐานข้อมูล (Database Evolution Diagnostic)

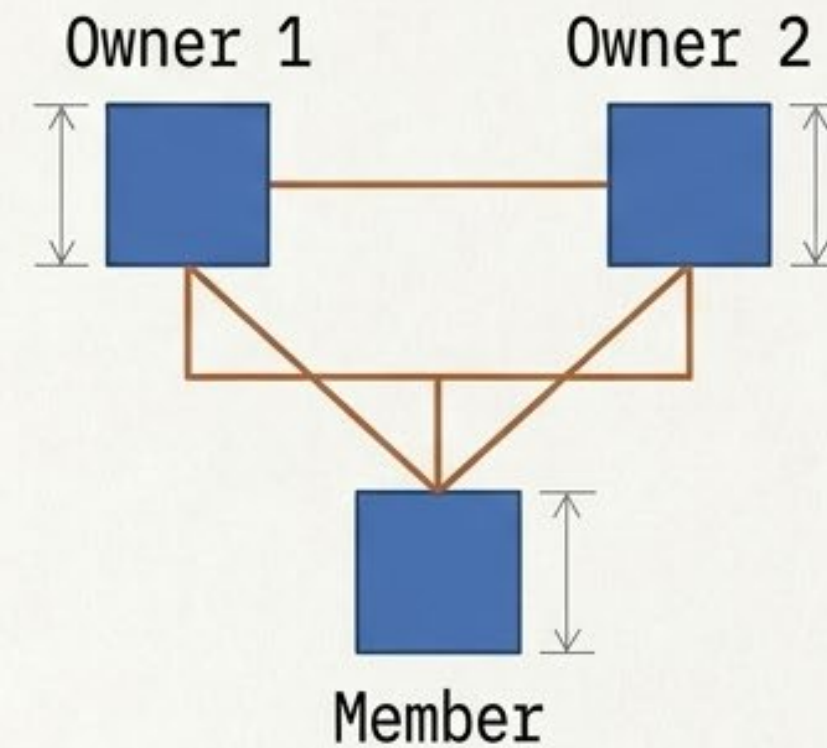
| รูปแบบ (Type)                       | โครงสร้าง (Structure)                                                                |                               | จุดเด่น (Core Strength)                         | ข้อจำกัด (Core Weakness)                 |
|-------------------------------------|--------------------------------------------------------------------------------------|-------------------------------|-------------------------------------------------|------------------------------------------|
| แบบลำดับชั้น<br>(Hierarchical)      |    | ต้นไม้<br>(Top-Down)          | ข้อมูลมีความคงสภาพสูง<br>(Data Integrity)       | ไม่รองรับความสัมพันธ์ M:N                |
| เครือข่าย<br>(Network)              |    | กราฟ/เซต<br>(Owner/Member)    | ยืดหยุ่นกว่าแบบลำดับชั้น<br>รองรับ M:N          | โครงสร้างซับซ้อน<br>ผูกติดกับแอปพลิเคชัน |
| เชิงสัมพันธ์<br>(Relational)        |  | ตาราง 2 มิติ<br>(Tables)      | เป็นอิสระทางโครงสร้าง,<br>ใช้ SQL               | ต้องการทรัพยากรระบบสูง                   |
| เชิงวัตถุ<br>(Object-Oriented)      |  | วัตถุ & คลาส                  | จัดการข้อมูลซับซ้อน<br>(กราฟิก/เสียง), Reusable | ขาดมาตรฐานที่ชัดเจน                      |
| มิติโตเมนชั้น<br>(Multidimensional) |  | ลูกบาศก์ข้อมูล<br>(Data Cube) | วิเคราะห์คลังข้อมูลได้หลายมิติ                  | ต้นทุนสูง<br>เหมาะกับธุรกิจขนาดใหญ่      |

# การปลดล็อกข้อจำกัดทางโครงสร้าง (Unlocking Structural Limitations)



## Hierarchical Model

โหนดลูกมีโหนดพ่อได้เพียงโหนดเดียว  
ขาดความยืดหยุ่นหากข้อมูลมีความ  
สัมพันธ์ที่ซับซ้อน



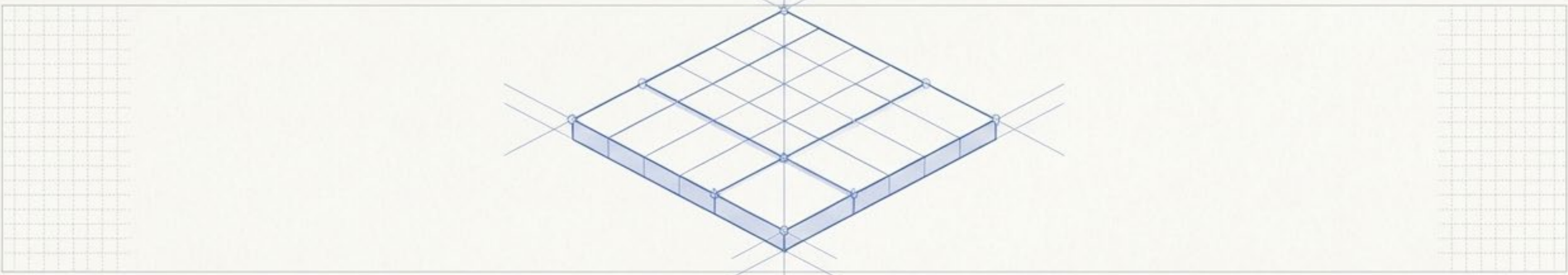
## Network Model

แต่ละโหนดเชื่อมโยงกันได้อย่างอิสระ (หลายพ่อได้)  
เกิดความเป็นอิสระทางโครงสร้างเบื้องต้น  
ก่อนนำไปสู่ยุคของตารางเชิงสัมพันธ์

# กระบวนทัศน์เชิงสัมพันธ์ & ศิลาจารึกคำศัพท์

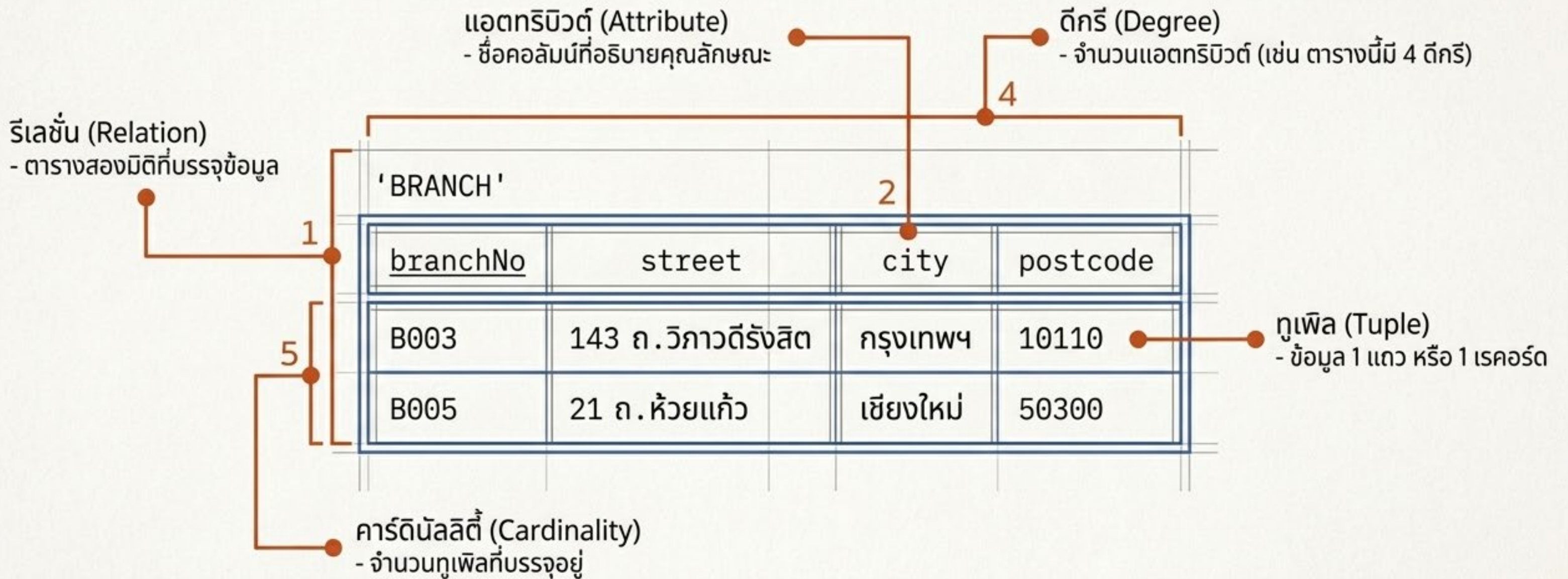
## (The Relational Paradigm & Terminology Rosetta Stone)

แบบจำลองที่นำเสนอมุมมองข้อมูลในลักษณะ "ตาราง" สื่อสารเข้าใจง่าย เชื่อมโยงข้อมูลผ่าน "คีย์" และมีความอิสระจากตัวโปรแกรม (Physical Data Independence).



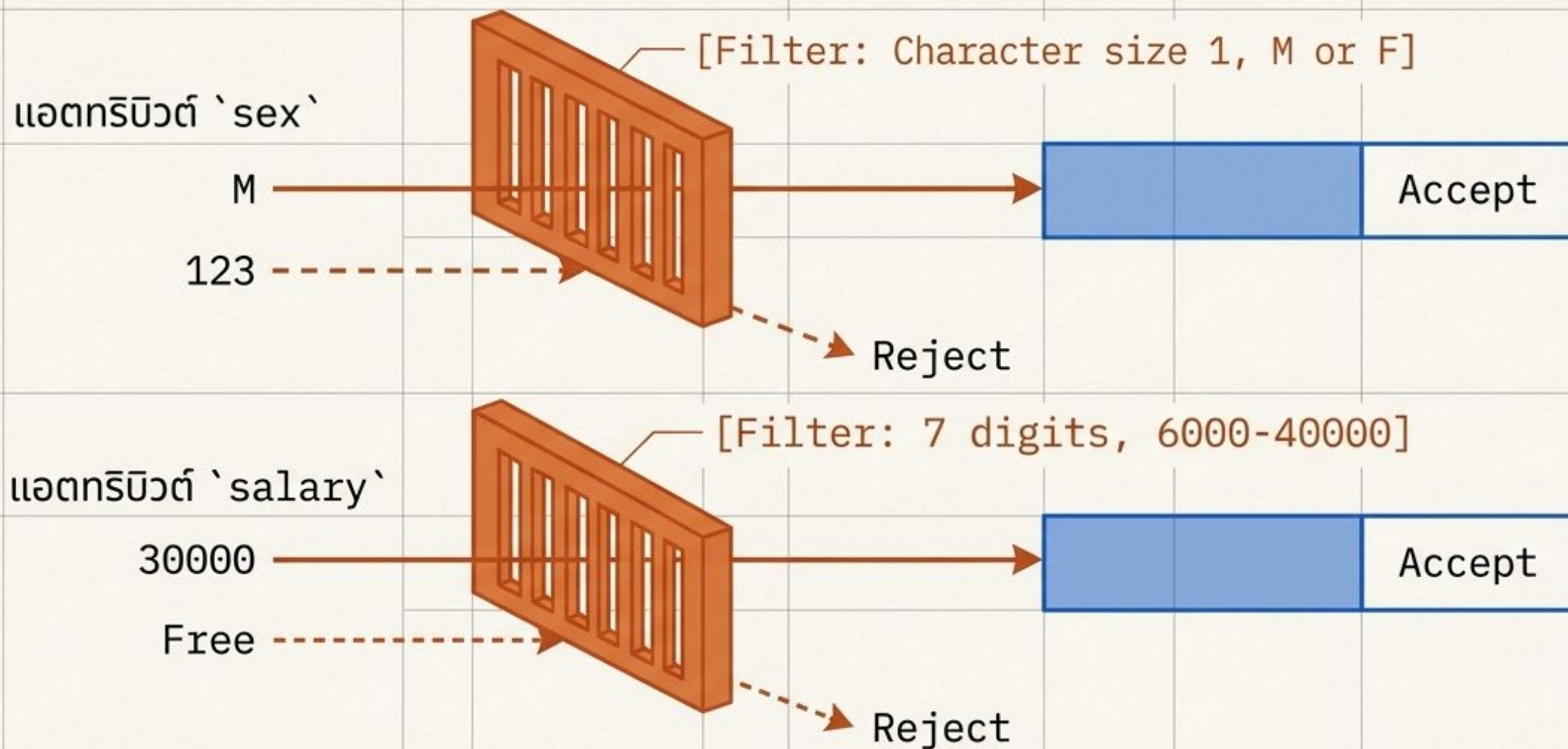
| ศัพท์วิชาการ (Formal)   | ศัพท์ฐานข้อมูลตาราง (Table) | ศัพท์ระบบไฟล์ (File) |
|-------------------------|-----------------------------|----------------------|
| รีเลชัน (Relation)      | ตาราง (Table)               | ไฟล์ (File)          |
| ทูเพิล (Tuple)          | แถว (Row)                   | ระเบียน (Record)     |
| แอตทริบิวต์ (Attribute) | คอลัมน์ (Column)            | ฟิลด์ (Field)        |

# กายวิภาคของรีเลชัน (Anatomy of a Relation)



# โดเมน: ผู้คุมกฎแห่งความถูกต้อง (The Domain Gatekeeper)

โดเมน (Domain) คือการกำหนดขอบเขตและชนิดข้อมูลของแต่ละแอตทริบิวต์ เพื่อป้องกันความผิดพลาดก่อนข้อมูลเข้าสู่ระบบ



# กฎข้อบังคับการก่อสร้าง: กฎ 12 ข้อของคอดด์ (The Building Code: Codd's 12 Rules)



1. Information:  
ทุกข้อมูลต้องอยู่ในรูปตาราง



2. Guaranteed Access:  
เข้าถึงผ่าน ชื่อตาราง +  
คีย์หลัก + คอลัมน์



3. Systematic Nulls:  
ค่าว่างถูกจัดการอย่างมีแบบแผน



4. Dynamic Catalog:  
เมทาดาตาค้นหาด้วยภาษา  
มาตรฐาน



5. Comprehensive Language:  
รองรับภาษาครอบคลุม  
นิยามและจัดการ



6. View Updating:  
วิวทุกชนิดสามารถอัปเดตกลับได้



7. High-Level Modification:  
รองรับแทรก, อัปเดต, ลบ  
ระดับสูง



8. Physical Independence:  
โครงสร้างกายภาพเปลี่ยน  
แอปไม่สะท้อน



9. Logical Independence:  
โครงสร้างตรรกะเปลี่ยน  
แอปไม่สะท้อน



10. Integrity Independence:  
กฎความคงสภาพอยู่ในระบบ  
ไม่ใช่แอป



11. Distribution  
Independence: ระบบทำงาน  
ได้แม้ข้อมูลกระจายตัว



12. Nonsubversion:  
ห้ามใช้ช่องทางระดับต่ำหลบ  
เลี่ยงกฎระบบ

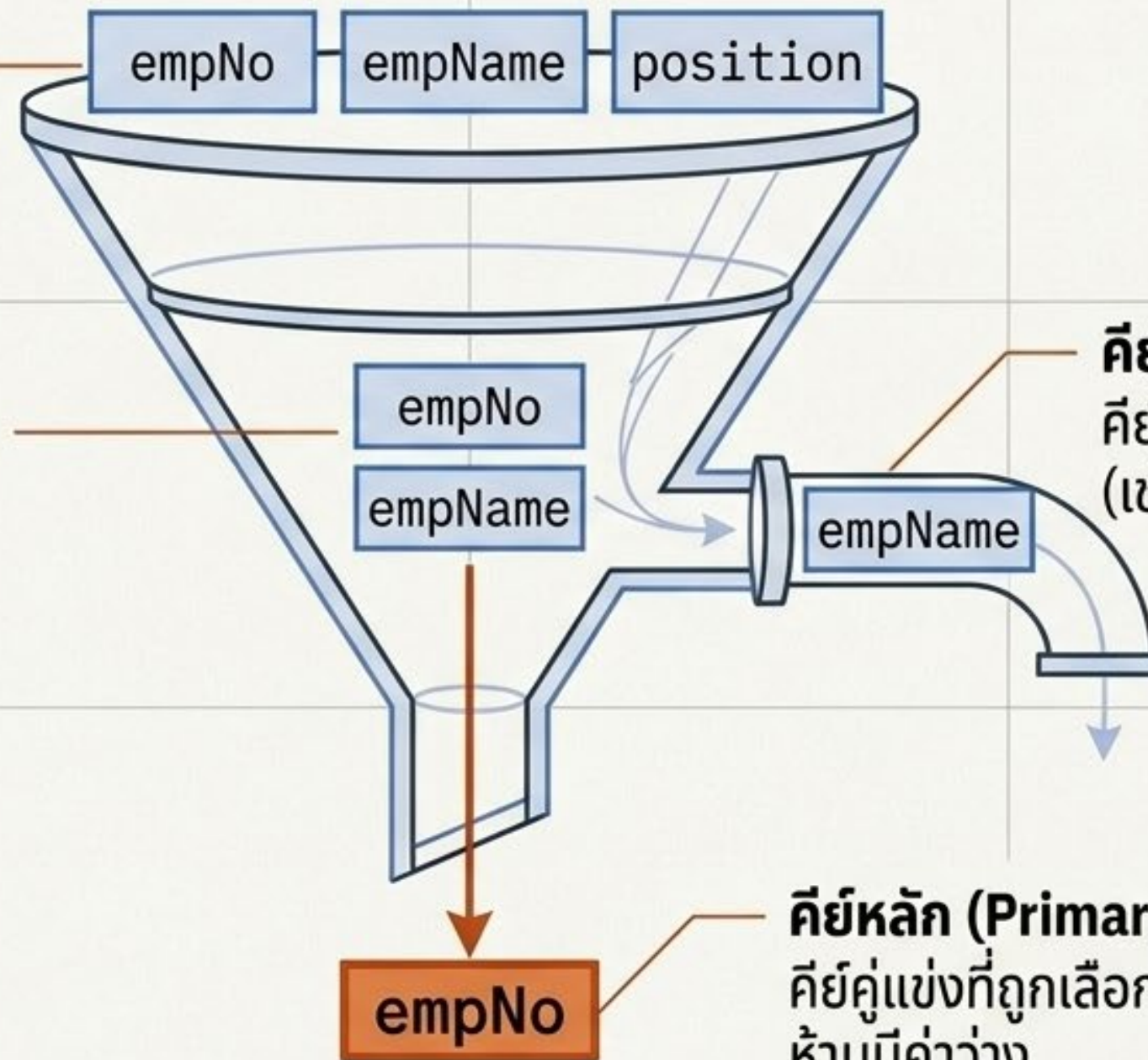
# กระบวนการกลั่นกรองคีย์ (The Key Filtration Funnel)

**ซูเปอร์คีย์ (Superkey)**  
กลุ่มแอตทริบิวต์ใดๆ ที่ระบุ  
ความแตกต่างของทูเพิลได้

**คีย์คู่แข่ง (Candidate Key)**  
ซูเปอร์คีย์ที่มีขนาดเล็กที่สุด  
ไม่มีซบเซตอื่นแล้ว

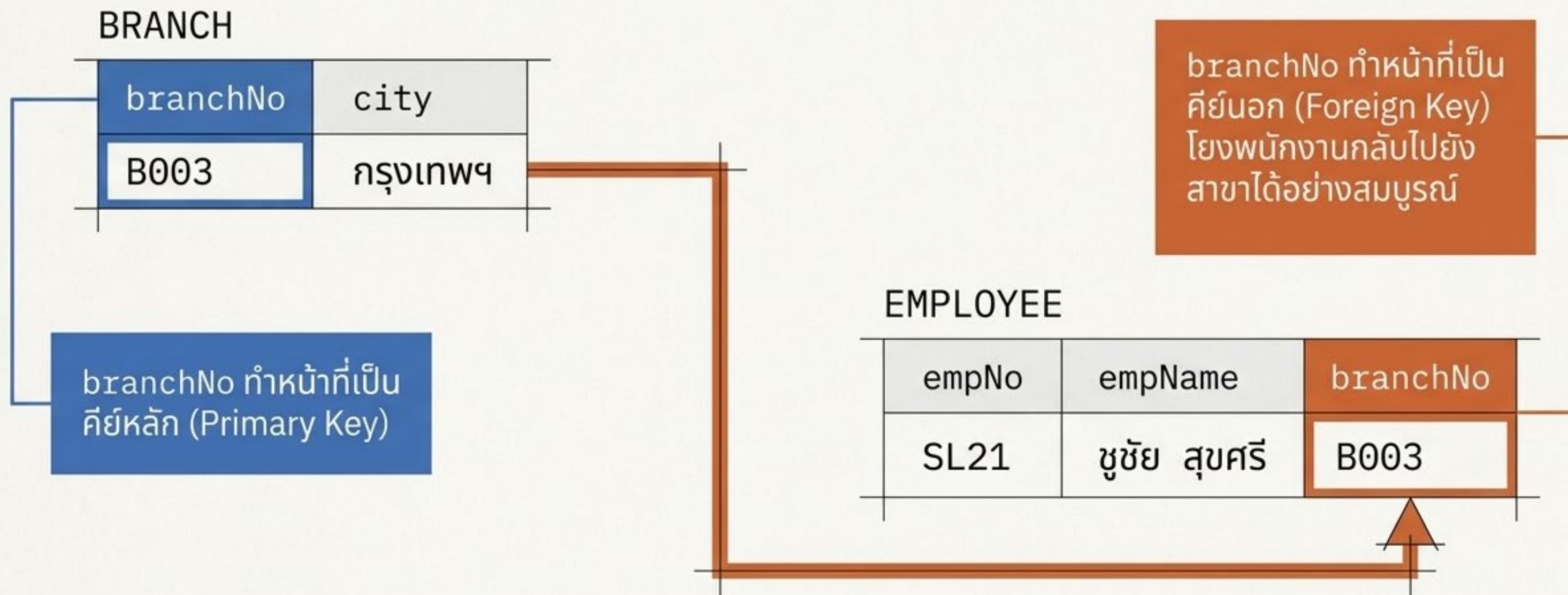
**คีย์รอง (Alternate Key)**  
คีย์คู่แข่งที่สอบตก ไม่ได้เป็นคีย์หลัก  
(เช่น ชื่ออาจซ้ำได้)

**คีย์หลัก (Primary Key)**  
คีย์คู่แข่งที่ถูกเลือกให้เป็นตัวแทนหลัก  
ห้ามมีค่าว่าง



# คีย์นอก: การเดินสายไฟเชื่อมต่อตาราง (Foreign Keys: Wiring the Database Together)

คีย์นอก (Foreign Key) คือแอตทริบิวต์ที่เป็นคีย์หลักในตารางหนึ่ง แต่ไปปรากฏในอีกตารางหนึ่งเพื่อใช้เชื่อมโยงความสัมพันธ์



# กรณีศึกษา: ระบบฐานข้อมูล DreamHome (Case Study: DreamHome Entities)

บริษัทตัวแทนบริการบ้านเช่าที่มีสาขาทั่วประเทศไทย ประกอบด้วยเอนทิตีหลักดังนี้:

**BRANCH (สาขา)**

PK: branchNo

**EMPLOYEE (พนักงาน)**

PK: empNo

**PROPERTY\_FOR\_RENT  
(บ้านเช่า)**

PK: propertyNo

**CLIENT (ลูกค้า)**

PK: clientNo

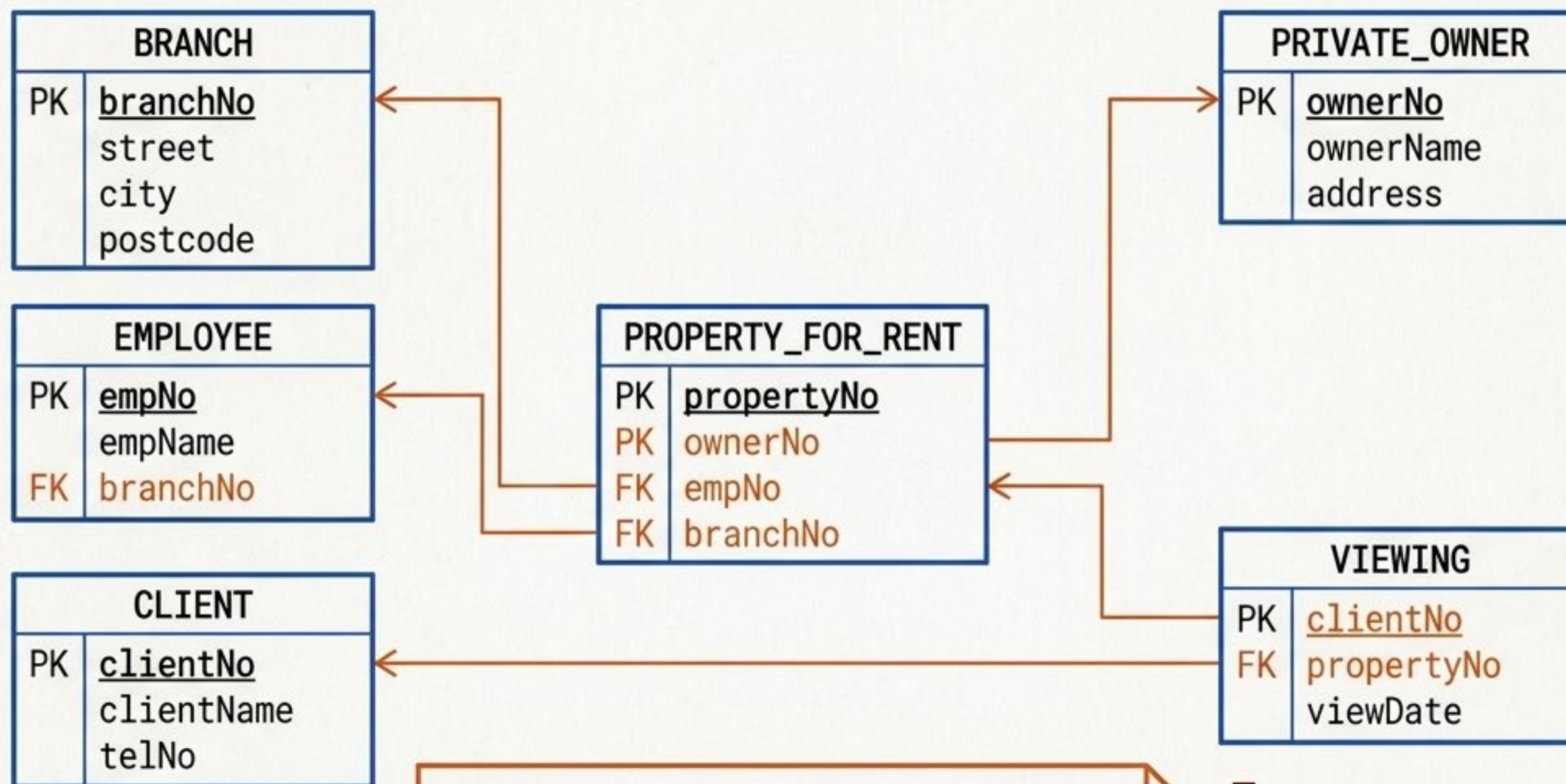
**PRIVATE\_OWNER  
(เจ้าของบ้าน)**

PK: ownerNo

**VIEWING (การเข้าชม)**

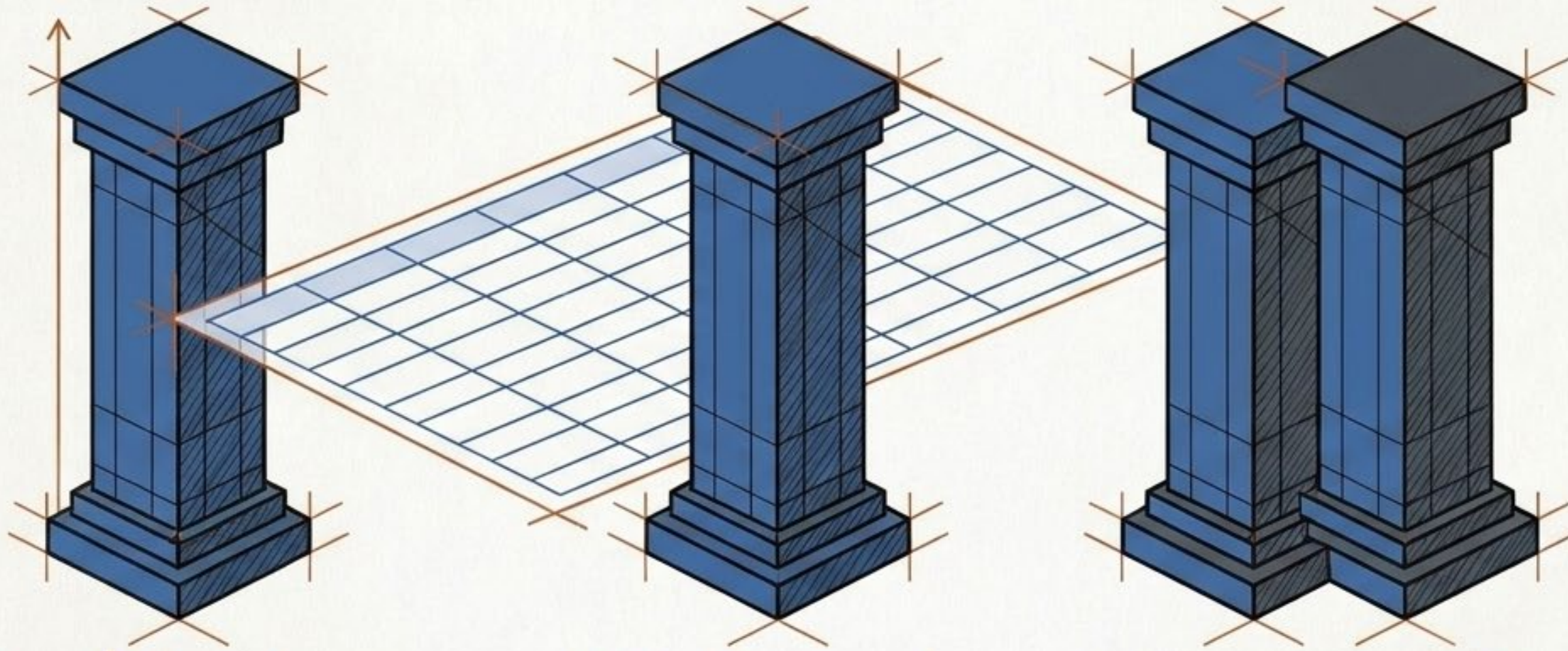
PK: clientNo, propertyNo  
(คีย์รวม - Composite Key)

# พิมพ์เขียวความสัมพันธ์ DreamHome (The Master Relational Blueprint)



เส้นสีส้มแสดงถึง **คีย์นอก** (Foreign Keys) ที่ทำหน้าที่ถักทอตารางอิสระเข้าด้วยกันเป็นระบบฐานข้อมูลเชิงสัมพันธ์ที่สมบูรณ์แบบ

# สถาปัตยกรรมที่เหนือกาลเวลา (The Timeless Architecture)



## ความอิสระที่ไร้รอยต่อ (Structural Independence)

การแยกโครงสร้างเชิงตรรกะออก  
จากกายภาพโดยสมบูรณ์

## ผู้พิทักษ์ความถูกต้อง (Robust Integrity)

การบังคับใช้ขอบเขตข้อมูล  
(Domains) และระบบคีย์ (Keys)  
ป้องกันความซ้ำซ้อน

## มาตรฐานคณิตศาสตร์ (Mathematical Foundation)

กฎของคอดด์และทฤษฎีเซตสร้างรากฐาน  
ที่แข็งแกร่ง ทำให้ฐานข้อมูลเชิงสัมพันธ์  
เป็นระบบที่มีประสิทธิภาพสูงสุด