รายงาน
เรื่อง การวิเคราะห์และจัดการข้อมูลด้วย MySQL: กรณีศึกษา 3 ระบบ
จัดทำโดย

นางสาวจุฑามาศ วงศ์สุโท 6521601776

นายชนาธิป เกี่ยวนุกูล 6621600178

นางสาวสิรภัทร วัตรบุตร 6621600500

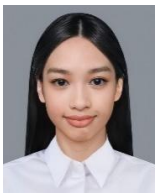
นายธนภัทร ชูสุวรรณ 6521601857

นางสาวชุติมณฑน์ อ่วมดีสุด 6621600216

นางสาวสุภัทรา ไชยา 6621600526

นายคุณากร กัลยาวุฒิพงศ์ 6621600143

นางสาวนภัสสร วัฏฏานนท์ 6621600321


นางสาวอัสวาณี อาริง 6621604700

นางสาวนวัตมน เกิดศรีเพ็ง 6621600330

เสนอ

อ.ดร.ศศิน เทียนดี

รายงานเล่มนี้เป็นส่วนหนึ่งของวิชา ระบบฐานข้อมูลเบื้องต้น 01418221-65

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ปีการศึกษา 2567

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน

ระบบการสั่งซื้อสินค้าออนไลน์

1.ทำความเข้าใจระบบ

1. ลูกค้าจะต้องทำการสมัครสมาชิกเพื่อเลือกซื้อสินค้า โดยขั้นตอนนี้จะมีการเก็บข้อมูล เลขบัตรประชาชน ชื่อ นามสกุล เพศ วัน/เดือน/ปีเกิด เบอร์โทร อีเมล เลขบัตรATM และที่อยู่ที่จัด-ส่ง
2. ผู้ขายจะต้องทำการลงทะเบียนร้านค้าเพื่อขายสินค้า โดยขั้นตอนนี้จะมีการเก็บข้อมูล ชื่อร้าน เลขบัตรประชาชน พิกัดร้าน เบอร์โทร และอีเมล
3. ผู้ขายทำการเพิ่มสินค้าเข้าระบบ โดยขั้นตอนนี้จะมีการเก็บข้อมูล ชื่อสินค้า ประเภท ราคา วัน/เดือน/ปีที่นำเข้า และวัน/เดือน/ปีที่หมดอายุของสินค้า
4. ลูกค้าทำการสั่งซื้อสินค้า และในการสั่งซื้อสินค้าจะมีส่วนลดให้ลูกค้าเลือกใช้ โดยขั้นตอนนี้จะมีการเก็บข้อมูล ไอดีลูกค้า รหัสสินค้า รหัสส่วนลด จำนวนสินค้า ยอดรวม ประเภทการจัดส่ง ราคาจัดส่ง วัน/เดือน/ปีที่สั่งซื้อ และเวลาที่สั่งซื้อ
5. หลังจากการสั่งซื้อสินค้า ลูกค้าจะต้องทำการชำระเงิน โดยขั้นตอนนี้จะมีการเก็บข้อมูล ไอดีคำสั่งซื้อ วัน/เดือน/ปีที่ชำระเงิน เวลาที่ชำระเงิน และสถานะการชำระเงิน
6. หลังจากที่ลูกค้าทำการชำระเงิน ผู้ขายจะทำการจัดส่งสินค้า ตามที่อยู่ที่ลูกค้ากรอกไว้ โดยขั้นตอนนี้จะเก็บข้อมูล รหัสการชำระเงิน ไอดีที่อยู่จัดส่ง วัน/เดือน/ปีที่จัดส่ง เวลาที่จัดส่ง บริษัทขนส่ง และสถานะการจัดส่ง
7. หลังจากลูกค้าได้รับสินค้า ลูกค้าจะต้องทำการรีวิวสินค้า โดยขั้นตอนนี้จะเก็บข้อมูล หมายเลขพัสดุ รูปภาพ รายละเอียดการรีวิว คะแนนคุณภาพสินค้า คะแนนบริการจากผู้ขาย และคะแนนผู้ให้บริการขนส่ง
8. ลูกค้าสามารถขอเงินคืน หรือ ส่งสินค้าคืน โดยจะต้องบอกเหตุผลการคืนสินค้าหรือขอคืนเงิน โดยขั้นตอนนี้จะเก็บข้อมูล หมายเลขพัสดุ วัน/เดือน/ปีที่ยกเลิก เวลาที่ยกเลิก เหตุผลการคืนเงิน/คืนสินค้า วัน/เดือน/ปีที่คืนเงิน เวลาที่คืนเงิน วัน/เดือน/ปีที่คืนสินค้า และเวลาที่คืนสินค้า

ข้อกำหนด

- ลูกค้าสมัครได้ 1 แอคเคาต์เท่านั้น
- ลูกค้าสามารถเพิ่มที่อยู่จัดส่งได้หลายที่อยู่
- ลูกค้าจะต้องทำการชำระเงินหลังการสั่งซื้อสินค้าภายใน 24 ชั่วโมงเท่านั้น หลังจากนั้นการสั่งซื้อสินค้าจะถูกยกเลิก
- การคืนสินค้าต้องคืนภายใน 3 วันหลังจากวันที่ได้รับสินค้า
- การคืนเงินระบบจะทำการคืนเงินลูกค้าภายใน 3 วันทำการ

2. สร้าง Entity Relational Diagram

ตารางบัตรประชาชน

เลขบัตรประชาชน	ชื่อ	นามสกุล	เพศ	วันเกิด	เดือนเกิด	ปีเกิด
1 7299 00123 45 6	สมพงษ์	ใจดี	M	12	5	2536
1 7299 00234 56 7	สมร	ปรีชา	F	10	1	2544
1 7299 00345 67 8	สมศรี	ประเสริฐ	F	9	3	2548
1 7299 00456 78 9	สมฤทัย	คงทำ	F	25	6	2545
1 7299 00567 89 0	สมหมาย	ทรงทำ	M	26	9	2538
1 7494 00578 12 3	สมพงษ์	เลี้ยงถนอม	M	1	10	2547
1 7647 00155 98 2	สมพร	เลี้ยงถนอม	F	3	4	2512
1 7630 01789 54 1	ลูกเกด	สิริธรณ์	F	7	5	2555
1 7424 00055 88 4	แวมไพร์	ทไวไลท์	F	5	8	2539
1 7889 00050 14 7	สิงหา	ใจดี	M	27	6	2511

ตารางบัตรประชาชน ประกอบด้วย Attributes ดังนี้

เลขบัตรประชาชน : ใช้สำหรับการยืนยันตัวตนบุคคล ใช้ทำธุรกรรมทางการเงิน และ ค้นหาบัตรประชาชนของคน 1 คน

ชื่อ-นามสกุล : เพื่อให้ทราบว่าลูกค้าที่ทำการสั่งซื้อสินค้า ชื่อ-นามสกุลอะไร กรณีที่ต้องติดต่อก็สามารถดึงข้อมูลไปอ้างอิงได้

เพศ : สำหรับระบุเพศสภาพ เพื่อกรณีที่ต้องการทราบกลุ่มลูกค้าว่าเป็นเพศหญิงหรือชาย

วัน/เดือน/ปีเกิด : สำหรับการระบุอายุของบุคคล เพื่อให้ทราบเกณฑ์อายุของลูกค้าสำหรับการนำไปทำสถิติ

ตารางบัตรATM

เลขบัตรATM	ไอดีลูกค้า	ชื่อ	นามสกุล	ประเภท	ธนาคาร	เดือนหมดอายุบัตร	ปีที่หมดอายุบัตร
1234 5678 9012 3456	MB2548	สมพงษ์	ใจดี	เครดิต	กสิกรไทย	9	28
9876 5432 1098 7654	MB2549	สมร	ปรีชา	เดบิต	กรุงไทย	11	29
4567 8901 2345 6789	MB2550	สมศรี	ประเสริฐ	เดบิต	กรุงเทพ	1	26
6543 2109 8765 4321	MB2551	สมฤทัย	คงทำ	เครดิต	ออมสิน	12	25
1111 2222 3333 4444	MB2552	สมหมาย	ทรงทำ	เครดิต	กรุงไทย	5	26

ตารางบัตรATM ประกอบด้วย Attributes ดังนี้

เลขบัตร ATM : ใช้สำหรับการทำธุรกรรมทางการเงิน และ ค้นหาข้อมูลของบัตรATM 1 บัญชี

ไอดีลูกค้า : เพื่อให้ทราบว่าบัตรATM นี้ถูกใช้ทำธุรกรรมโดยลูกค้าคนไหน

ชื่อ-นามสกุล : เพื่อให้ทราบว่าบัตรที่ลูกค้าทำการชำระเงินมีเจ้าของบัญชี ชื่อ-นามสกุลอะไร กรณีที่ชื่อบัตรATM ไม่ตรงกับชื่อบัตรประชาชน

ประเภทบัตร ATM : ระบุประเภทของบัตร โดยบัตรATM ที่สามารถใช้มำธุรกรรมออนไลน์จะมี 2 ประเภท คือ บัตรเดบิต และ บัตรเครดิต

ธนาคาร : ระบุธนาคารผู้ออกบัตร กรณีที่ต้องมีการติดต่อกับธนาคารเจ้าของบัตร

เดือน/ปีที่หมดอายุบัตร : สำหรับการตรวจสอบตามเงื่อนไขที่กำหนด กรณีที่มีบัตรหมดอายุ

ตารางลูกค้า

ไอดีลูกค้า	เลขบัตรประชาชน	เบอร์โทร	อีเมล
MB2548	1 7299 00123 45 6	1-2345-6789	Somp@gmail.com
MB2549	1 7299 00234 56 7	2-3456-7891	Samorn@gmail.com
MB2550	1 7299 00345 67 8	3-4567-8912	Somsri@gmail.com
MB2551	1 7299 00456 78 9	4-5678-9123	Smrueth@gmail.com
MB2552	1 7299 00567 89 0	5-6789-1234	Sommhai@gmail.com

ตารางลูกค้า ประกอบด้วย Attributes ดังนี้

ไอดีลูกค้า : สำหรับการค้นหาข้อมูลของลูกค้า 1 คน

เลขบัตรประชาชน : เพื่อบอกว่าลูกค้าคนนี้ มีรายละเอียดข้อมูลอะไรบ้าง เช่น ชื่อ นามสกุล ตามข้อมูลที่เก็บในตารางบัตรประชาชน

เบอร์โทร : ใช้ในการติดต่อสื่อสารกับลูกค้า เช่น การแจ้งข้อมูลข่าวสาร การเสนอโปรโมชั่น หรือการแก้ไขปัญหา

อีเมล : ใช้ในการติดต่อสื่อสารกับลูกค้า เช่น การส่งจดหมายข่าว การแจ้งข้อมูลโปรโมชั่น หรือการยืนยันการสั่งซื้อ

ตารางสินค้า

รหัสสินค้า	ชื่อร้านค้า	ชื่อสินค้า	ประเภท	ราคา	วันที่นำเข้า
SA0001	ShopEase	ดีโด้	เครื่องดื่ม	10.00	15
SA0002	TrendzyMall	กาโตะ	เครื่องดื่ม	12.00	7
SA0003	QuickBuy	เลย์	ขนมคบเคี้ยว	20.00	2
SA0004	EzyMart	เถ้าแก่น้อย	ขนมคบเคี้ยว	10.00	21
SA0005	BestPick	โดโซะ	ขนมคบเคี้ยว	25.00	27

เดือนที่นำเข้า	ปีที่นำเข้า	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ
2	2565	16	5	2568
1	2567	8	1	2570
6	2567	3	6	2570
10	2567	22	10	2570
9	2568	28	9	2571

ตารางสินค้า ประกอบด้วย Attributes ดังนี้

รหัสสินค้า : สำหรับการค้นหาข้อมูลของสินค้า 1 สินค้า

ชื่อร้านค้า : เพื่อให้ทราบว่าผู้ขายคนไหนเป็นคนขายสินค้ารายการนี้

ชื่อสินค้า : เพื่อให้ทราบชื่อของสินค้าแต่ละรายการ

ประเภท : สำหรับการจัดหมวดหมู่ของสินค้า ทำให้ง่ายต่อการค้นหา และจัดประเภท

ราคา : ใช้สำหรับการติดตามราคาของสินค้าแต่ละรายการ ทำให้ง่ายต่อการปรับปรุงราคา

วัน/เดือน/ปีที่นำเข้า : ใช้ติดตามวันที่นำเข้าของสินค้า ทำให้ทราบถึงอายุของสินค้า

วัน/เดือน/ปีที่หมดอายุ : ใช้สำหรับการตรวจสอบสินค้าที่ใกล้หมดอายุ เพื่อป้องกันการจำหน่ายสินค้าหมดอายุ

ตารางผู้ขาย

ชื่อร้านค้า	เลขบัตรประชาชน	พิกัด	เบอร์โทร	อีเมล
ShopEase	1 7494 00578 12 3	13.7465°N,100.5340°E	8-5514-3659	Shopease@gmail.com
TrendzyMall	1 7647 00155 98 2	13.7460°N,100.5296°E	8-7567-5312	Trendzymall@gmai.com
QuickBuy	1 7630 01789 54 1	13.6687°N,100.6042°E	9-8012-0893	Quickbuy@gmail.com
EzyMart	1 7424 00055 88 4	13.7370°N,100.5460°E	6-6413-9278	Ezymart@gmail.com
BestPick	1 7889 00050 14 7	13.7292°N,100.7750°E	6-6019-8723	Bestpick@gmail.com

ตารางผู้ขาย ประกอบด้วย Attributes ดังนี้

ชื่อร้านค้า : สำหรับการค้นหาข้อมูลผู้ขายของร้านค้านั้น 1 ร้านค้า

เลขบัตรประชาชน : เพื่อบอกว่าผู้ขายคนนี้ มีรายละเอียดข้อมูลอะไรบ้าง เช่น ชื่อ นามสกุล ตามข้อมูลที่เก็บในตารางบัตรประชาชน

พิกัด : ระบุตำแหน่งที่ตั้งของผู้ขาย เพื่อให้ทราบว่าร้านค้ามีหน้าร้านจริง กรณีต้องการติดต่อ หรือ หาข้อมูลเกี่ยวกับร้าน

เบอร์โทร : ใช้ในการติดต่อสื่อสารกับผู้ขาย เช่น การแจ้งข้อมูลข่าวสาร การประสานงาน และการแก้ไขปัญหา

อีเมล : ใช้ในการติดต่อสื่อสารกับผู้ขาย เช่น การส่งจดหมายข่าว การส่งข้อมูล และการแจ้งเตือน

ตารางส่วนลด

รหัสส่วนลด	ชื่อส่วนลด	ประเภทส่วนลด	มูลค่าของส่วนลด	จำนวนครั้งที่ใช้งานได้	วันที่เริ่มใช้ส่วนลด
DC0001	ซื้อครบ 300 บาท	ส่วนลด ร้านโค้ดคุ้ม	50.00	1000	7
DC0002	ซื้อครบ 150 บาท	ส่วนลดร้านค้า	100.00	500	7
DC0003	ซื้อครบ 100 บาท	ส่วนลด ร้านโค้ดคุ้ม	10.00	550	7
DC0004	ซื้อครบ 200 บาท	ส่วนลดร้านค้า	20.00	1000	8
DC0005	จัดส่งฟรี	จัดส่งฟรี	30.00	555	8

เดือนที่เริ่มใช้ส่วนลด	ปีที่เริ่มใช้ส่วนลด	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ
3	2568	14	3	2568
6	2568	14	6	2568
7	2568	14	7	2568
1	2568	15	1	2568
1	2568	15	1	2568

ตารางส่วนลด ประกอบด้วย Attributes ดังนี้

รหัสส่วนลด : สำหรับการค้นหาข้อมูลของส่วนลด 1 รายการ

ชื่อส่วนลด : เพื่อให้ทราบชื่อของส่วนลดแต่ละรายการ

ประเภทส่วนลด : สำหรับการจัดหมวดหมู่ส่วนลด ทำให้ง่ายต่อการค้นหา และจัดประเภท

มูลค่าของส่วนลด : ระบุมูลค่าของส่วนลด ทำให้ทราบถึงจำนวนเงินที่ลดได้ของส่วนลดรายการนั้น

จำนวนครั้งที่ใช้งานได้ : จำกัดจำนวนครั้งที่สามารถใช้ส่วนลดได้ ช่วยในการควบคุมการใช้งานส่วนลด

วัน/เดือน/ปีที่เริ่มใช้ส่วนลด : ติดตามวันที่เริ่มใช้ส่วนลด ทำให้ลูกค้าทราบว่าช่วงเวลาที่ส่วนลดมีผล

วัน/เดือน/ปีที่หมดอายุ : ใช้ตรวจสอบส่วนลดที่หมดอายุ และป้องกันการใช้งานส่วนลดหมดอายุ

ตารางการสั่งซื้อสินค้า

ไอดีคำสั่งซื้อ	ไอดีลูกค้า	รหัสสินค้า	รหัสส่วนลด	จำนวน	ยอดรวม	ประเภทการจัดส่ง
SM0001	MB2548	SA0001	DC0001	30	300.00	การจัดส่งด่วน
SM0002	MB2549	SA0002	DC0002	15	180.00	การจัดส่งแบบพัสดุลงทะเบียน
SM0003	MB2550	SA0003	DC0003	10	200.00	การจัดส่งด่วน
SM0004	MB2551	SA0004	DC0004	23	230.00	การจัดส่งระหว่างประเทศ
SM0005	MB2552	SA0005	DC0005	10	250.00	การจัดส่งแบบพัสดุลงทะเบียน
SM0006	MB2551	SA0005	DC0004	10	250.00	การจัดส่งแบบพัสดุลงทะเบียน

ราคาจัดส่ง	ราคาสุทธิ	วันที่สั่งซื้อ	เดือนที่สั่งซื้อ	ปีที่สั่งซื้อ	เวลาที่สั่งซื้อ	สถานะการสั่งซื้อ
50	250.00	11	1	2568	7:12:00	สำเร็จ
40	120.00	14	1	2568	6:00:00	สำเร็จ
50	240.00	24	1	2568	14:09:36	สำเร็จ
80	290.00	9	2	2568	11:16:48	สำเร็จ
40	260.00	14	2	2568	11:31:12	สำเร็จ
40	260.00	14	2	2568	12:31:12	ไม่สำเร็จ

ตารางการสั่งซื้อสินค้า ประกอบด้วย Attributes ดังนี้

ไอดีคำสั่งซื้อ : สำหรับการค้นหาข้อมูลของคำสั่งซื้อ 1 คำสั่งซื้อ

ไอดีลูกค้า : เพื่อให้ทราบว่าคำสั่งซื้อนี้ ถูกสั่งซื้อโดยลูกค้าคนไหน

รหัสสินค้า : เพื่อให้ทราบว่าสินค้าไหนถูกสั่งซื้อในคำสั่งซื้อนี้

รหัสส่วนลด : เพื่อให้ทราบว่าคำสั่งซื้อนี้ มีการใช้ส่วนลดอะไร

จำนวน : ระบุจำนวนสินค้าที่ลูกค้าสั่งซื้อ

ยอดรวม : เป็นยอดรวมของสินค้าตามจำนวนของสินค้าที่ลูกค้าสั่งซื้อ

ประเภทการจัดส่ง : ระบุวิธีการจัดส่งที่ลูกค้าเลือก เพื่อใช้ในการจัดส่งสินค้าและการคำนวณค่าจัดส่ง

ราคาจัดส่ง : ระบุราคาค่าจัดส่งตามประเภทที่ลูกค้าเลือก

ราคาสุทธิ : ใช้คำนวณราคาสินค้าที่ลูกค้าต้องจ่ายหลังจากหักส่วนลดและค่าจัดส่ง เพื่อตรวจสอบยอดเงินที่ลูกค้าจ่ายได้

วัน/เดือน/ปีที่สั่งซื้อ : ใช้ในการระบุวัน เดือน ปีที่มีการสั่งซื้อ เพื่อสามารถนำไปวิเคราะห์ตามช่วงเวลา และติดตามคำสั่งซื้อได้

เวลาที่สั่งซื้อ : ใช้ในการระบุเวลาที่มีการสั่งซื้อ เพื่อสามารถนำไปวิเคราะห์ตามช่วงเวลาและติดตามคำสั่งซื้อได้

สถานะการสั่งซื้อ : ใช้ติดตามสถานะของคำสั่งซื้อ คือ สำเร็จ และ ไม่สำเร็จ

ตารางการชำระเงิน

รหัสการชำระเงิน	ไอดีคำสั่งซื้อ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระเงิน	สถานะการชำระเงิน
PM0001	SM0001	11	1	2568	7:12:00	สำเร็จ
PM0002	SM0002	14	1	2568	0:00:00	สำเร็จ
PM0003	SM0003	24	1	2568	6:00:00	สำเร็จ
PM0004	SM0004	9	2	2568	8:38:24	สำเร็จ
PM0005	SM0005	14	2	2568	14:09:36	สำเร็จ
PM0006	SM0006	NULL	NULL	NULL	NULL	ไม่สำเร็จ

ตารางการชำระเงิน ประกอบด้วย Attributes ดังนี้

รหัสการชำระเงิน : สำหรับการค้นหาข้อมูลของการชำระเงิน 1 การชำระเงิน

ไอดีคำสั่งซื้อ : เพื่อให้ทราบว่าการชำระเงินนี้ เป็นการชำระเงินของไอดีคำสั่งซื้อไหน
วัน/เดือน/ปีที่ชำระเงิน : ใช้ในการระบุวัน เดือน ปีที่ทำการชำระเงิน โดยลูกค้าจะต้องทำการชำระเงินหลังการสั่งซื้อสินค้าภายใน 24 ชั่วโมงเท่านั้น หลังจากนั้นการสั่งซื้อสินค้าจะถูกยกเลิก
เวลาที่ชำระเงิน : ใช้ในการระบุเวลาที่ทำการชำระเงิน โดยลูกค้าจะต้องทำการชำระเงินหลังการสั่งซื้อสินค้าภายใน 24 ชั่วโมงเท่านั้น หลังจากนั้นการสั่งซื้อสินค้าจะถูกยกเลิก
สถานะการชำระเงิน : ใช้ติดตามสถานะของการชำระเงิน คือ สำเร็จ และ ไม่สำเร็จ

ตารางการจัดส่ง

หมายเลขพัสดุ	รหัสการชำระเงิน	ไอดีที่อยู่จัดส่ง	วันที่จัดส่ง	เดือนที่จัดส่ง	ปีที่จัดส่ง	เวลาที่จัดส่ง	บริษัท	สถานะการจัดส่ง
1Z12345E1512345676	PM0001	AD0001	12	1	2568	00:00:00	Kerry Express	จัดส่งสำเร็จ
EA123456789TH	PM0002	AD0002	15	1	2568	07:12:00	Thai Post	กำลังจัดส่ง
JT123456789TH	PM0003	AD0003	NULL	NULL	NULL	NULL	NULL	ไม่สำเร็จ
NV123456789	PM0004	AD0004	10	2	2568	07:12:00	Ninja Van	จัดส่งสำเร็จ
JV123456789TH	PM0005	AD0005	15	2	2568	07:12:00	J&T Express	จัดส่งสำเร็จ

ตารางการจัดส่ง ประกอบด้วย Attributes ดังนี้
หมายเลขพัสดุ : สำหรับการค้นหาข้อมูลของการจัดส่ง 1 ครั้ง
รหัสการชำระเงิน : เพื่อให้ทราบว่าการจัดส่งนี้ เป็นการจัดส่งของการชำระเงินหมายเลขไหน
ไอดีที่อยู่จัดส่ง : เพื่อให้ทราบว่าการจัดส่งนี้ เป็นการจัดส่งไปยังที่อยู่การจัดส่งไหน
วัน/เดือน/ปีที่จัดส่ง : ใช้ในการระบุวัน เดือน ปีที่ทำการจัดส่งสินค้า เพื่อสามารถนำไปวิเคราะห์ตามช่วงเวลาและติดตามการจัดส่งได้
เวลาที่จัดส่ง :ใช้ในการระบุเวลาที่ทำการจัดส่งสินค้า เพื่อสามารถนำไปวิเคราะห์ตามช่วงเวลาและติดตามการจัดส่งได้
บริษัท : ระบุบริษัทที่ทำการจัดส่ง กรณีที่ต้องมีการติดต่อกับบริษัทการจัดส่งสินค้า
สถานะการจัดส่ง : ใช้ติดตามสถานะของการจัดส่ง คือ จัดส่งสำเร็จ กำลังจัดส่ง และ ไม่สำเร็จ

ตารางที่อยู่จัดส่ง

ไอดีที่อยู่จัดส่ง	บ้านเลขที่	หมู่	ซอย	ตำบล/แขวง	อำเภอ/เขต	จังหวัด	รหัสไปรษณีย์
AD0001	20/1	4	2	บางแก้ว	บางพลี	สมุทรปราการ	10540
AD0002	10/29	7	1	ปากเกร็ด	ปากเกร็ด	นนทบุรี	11120
AD0003	134/24	9	5	เชียงใหม่	เชียงใหม่	เชียงใหม่	50000
AD0004	9/10	1	4	ท่าใหม่	ท่าใหม่	จันทบุรี	22120
AD0005	205/3	3	13	ท่าแร้ง	บางเขน	กรุงเทพมหานคร	10220

ตารางที่อยู่จัดส่ง ประกอบด้วย Attributes ดังนี้
ไอดีที่อยู่จัดส่ง : สำหรับการค้นหาข้อมูลของที่อยู่จัดส่ง 1 ที่อยู่จัดส่ง
บ้านเลขที่ : ระบุหมายเลขบ้าน เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง
หมู่ : ระบุหมายเลขหมู่บ้าน เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง
ซอย : ระบุซอยหรือถนน เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง
ตำบล/แขวง : ระบุตำบล/แขวง เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง
อำเภอ/เขต : ระบุอำเภอ/เขต เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง
จังหวัด : ระบุจังหวัด เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง
รหัสไปรษณีย์ : ระบุหมายเลขรหัสไปรษณีย์ เพื่อระบุที่อยู่จัดส่งที่เฉพาะเจาะจงของลูกค้าให้สามารถจัดส่งได้ถูกต้อง

ตารางการรีวิวสินค้า

ไอดีรีวิว	หมายเลขพัสดุ	Pathรูปภาพ	รายละเอียด
RW0001	1Z12345E1512345676	NULL	NULL
RW0002	EA123456789TH	C:\Users\User\Pictures\dowload\guide-for-food-photography-wedio.jpg	NULL
RW0003	JT123456789TH	NULL	NULL
RW0004	NV123456789	NULL	แม่ค้าตอบดีมาก
RW0005	JV123456789TH	C:\Users\User\Pictures\dowload\OIP.jpg	ขนมอร่อย

คุณภาพสินค้า	บริการจากผู้ขาย	ผู้ให้บริการขนส่ง
NULL	85	80
NULL	80	85
50	60	65
NULL	NULL	NULL
95	95	95

ตารางการรีวิวสินค้า ประกอบด้วย Attributes ดังนี้

ไอดีการรีวิว : สำหรับการค้นหาข้อมูลของการรีวิวสินค้า 1 การรีวิว

หมายเลขพัสดุ : เพื่อให้ทราบว่าการรีวิวนี้ เป็นการรีวิวของหมายเลขพัสดุหมายเลขไหน

Path รูปภาพ : ระบุที่อยู่ของรูปภาพที่ลูกค้าแนบมาในการรีวิวสินค้า เช่น รูปสินค้าที่ได้รับ เพื่อเป็นภาพประกอบการรีวิว

รายละเอียด : ใช้ในการเก็บข้อมูลที่ลูกค้าระบุเกี่ยวกับสินค้า เช่น ความพึงพอใจ หรือข้อเสนอแนะต่างๆ

คุณภาพสินค้า : เพื่อให้ลูกค้าให้คะแนนเกี่ยวกับคุณภาพของสินค้าที่ได้รับ และสามารถนำไปวิเคราะห์รวมถึงปรับปรุงคุณภาพสินค้าได้

บริการจากผู้ขาย : เพื่อให้ลูกค้าให้คะแนนเกี่ยวกับบริการจากผู้ขาย และสามารถนำไปวิเคราะห์รวมถึงปรับปรุงการบริการของผู้ขายได้

ผู้ให้บริการขนส่ง : เพื่อให้ลูกค้าให้คะแนนเกี่ยวกับผู้ให้บริการขนส่ง และสามารถนำไปวิเคราะห์รวมถึงปรับปรุงการขนส่งและผู้บริการขนส่งได้

ตารางการคืนเงิน/คืนสินค้า

ไอดีการคืนเงิน/คืนสินค้า	หมายเลขพัสดุ	วันที่ยกเลิก	เดือนที่ยกเลิก	ปีที่ยกเลิก	เวลาที่ยกเลิก	เหตุผลการคืนเงิน
CC0001	1Z12345E1512345676	15	1	2568	07:12:00	ได้สินค้าไม่ครบ
CC0002	EA123456789TH	16	1	2568	08:38:24	ยกเลิกการสั่ง
CC0003	JT123456789TH	17	1	2568	00:28:48	ยกเลิกการสั่ง
CC0004	NV123456789	18	2	2568	10:48:00	ได้สินค้าไม่ครบ
CC0005	JV123456789TH	19	2	2568	08:38:24	เปลี่ยนสินค้า

วันที่คืนเงิน	เดือนที่คืนเงิน	ปีที่คืนเงิน	เวลาที่คืนเงิน	วันที่คืนสินค้า	เดือนที่คืนสินค้า	ปีที่คืนสินค้า	เวลาที่คืนสินค้า
15	1	2568	09:36:00	16	1	2568	09:48:00
16	1	2568	09:00:00	17	1	2568	10:50:00
17	1	2568	08:24:00	NULL	NULL	NULL	NULL
18	2	2568	10:05:00	19	2	2568	11:27:00
19	2	2568	08:50:00	20	2	2568	09:22:00

ตารางการคืนเงิน/คืนสินค้า ประกอบด้วย Attributes ดังนี้

ไอดีการคืนเงิน/คืนสินค้า : สำหรับการค้นหาข้อมูลของการคืนเงินหรือการคืนสินค้า 1 ครั้ง

หมายเลขพัสดุ : เพื่อให้ทราบว่าการคืนเงินหรือคืนสินค้านี้ เป็นการคืนเงินหรือคืนสินค้าของหมายเลขพัสดุหมายเลขไหน

วัน/เดือน/ปีที่ยกเลิก : ใช้ในการระบุวัน เดือน ปีที่ยกเลิกสินค้า เพื่อบันทึกข้อมูลและสามารถตรวจสอบระยะเวลาที่เกิดการยกเลิกสินค้า
เวลาที่ยกเลิก : ระบุเวลาที่ทำการยกเลิกหรือคืนสินค้า เพื่อบันทึกข้อมูลและสามารถตรวจสอบระยะเวลาที่เกิดการยกเลิกสินค้า
เหตุผลการคืนเงิน : ใช้ระบุเหตุผลที่ลูกค้าทำการคืนเงิน เช่น ได้สินค้าไม่ครบ ต้องการยกเลิกการสั่งซื้อ หรือต้องการเปลี่ยนสินค้า
วัน/เดือน/ปีที่คืนเงิน : ใช้เพื่อระบุวันที่ทำการคืนเงิน เพื่อบันทึกและตรวจสอบสถานะการคืนเงิน โดยระบบจะทำการคืนเงินลูกค้าภายใน 3 วันหลังจากที่ลูกค้ายกเลิกสินค้า
เวลาที่คืนเงิน : ใช้เพื่อระบุเวลาที่ทำการคืนเงิน เพื่อบันทึกและตรวจสอบสถานะการคืนเงิน โดยระบบจะทำการคืนเงินลูกค้าภายใน 3 วันหลังจากที่ลูกค้ายกเลิกสินค้า
วัน/เดือน/ปีที่คืนสินค้า : ใช้เพื่อระบุวันที่ทำการคืนสินค้า เพื่อบันทึกและตรวจสอบการคืนสินค้า โดยจะต้องคืนสินค้าภายใน 3 วันหลังจากวันที่ได้รับสินค้า
เวลาที่คืนสินค้า : ใช้เพื่อระบุเวลาที่ทำการคืนสินค้า เพื่อบันทึกและตรวจสอบการคืนสินค้า โดยจะต้องคืนสินค้าภายใน 3 วันหลังจากวันที่ได้รับสินค้า

การวิเคราะห์หา Entity **และ** Attributes

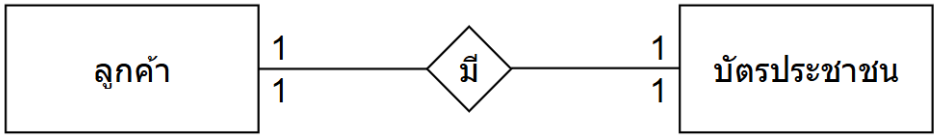
- ลูกค้า
- บัตรประชาชน
- บัตรเอทีเอ็ม
- ผู้ขาย
- สินค้า
- การสั่งซื้อสินค้า
- ส่วนลด
- การชำระเงิน
- การจัดส่ง
- ที่อยู่จัดส่ง
- การรีวิวสินค้า
- การคืนเงิน/คืนสินค้า

Key Attributes

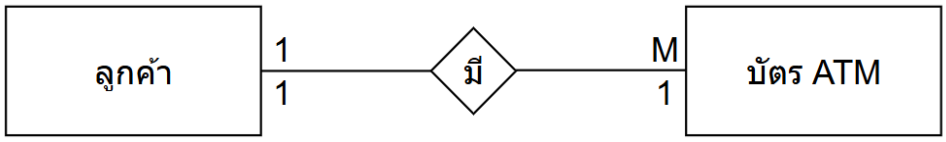
- ไอดีลูกค้า
- เลขบัตรประชาชน
- เลขบัตรเอทีเอ็ม
- ชื่อร้านค้า
- ไอดีสินค้า
- ไอดีคำสั่งซื้อ
- ไอดีส่วนลด
- ไอดีชำระเงิน
- หมายเลขพัสดุ
- ไอดีที่อยู่จัดส่ง
- ไอดีรีวิว
- ไอดีการคืนเงิน

วิเคราะห์หาความสัมพันธ์ระหว่าง Entity

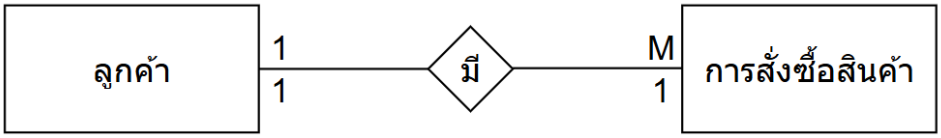
- **ลูกค้า กับ บัตรประชาชน**
  - ลูกค้า 1 คน มี บัตรประชาชน 1 ใบ
  - บัตรประชาชน 1 ใบ ถือครองโดยลูกค้า 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1:1



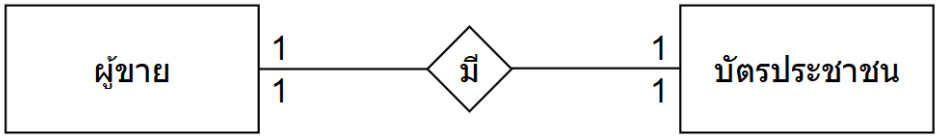
- **ลูกค้า กับ บัตรเอทีเอ็ม**
  - ลูกค้า 1 คน มีบัตรเอทีเอ็มหลายใบ
  - บัตรเอทีเอ็ม 1 ใบ ถือครองโดยลูกค้า 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1:M



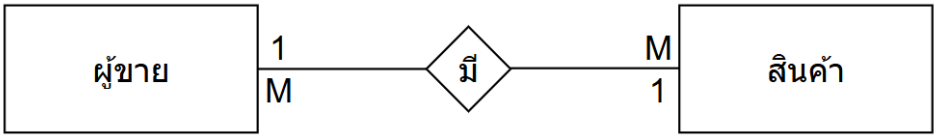
- **ลูกค้า กับ การสั่งซื้อสินค้า**
  - ลูกค้า 1 คน มีการสั่งซื้อสินค้าหลายครั้ง
  - การสั่งซื้อสินค้า 1 ครั้ง ทำโดยลูกค้า 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1: M



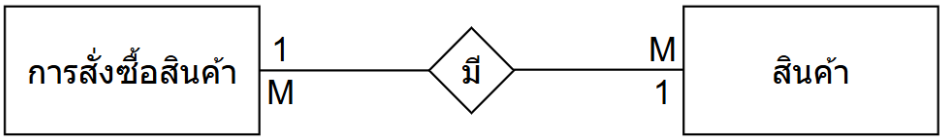
- **ผู้ขาย กับ บัตรประชาชน**
  - ผู้ขาย 1 คน มี บัตรประชาชน 1 ใบ
  - บัตรประชาชน 1 ใบ ถือครองโดยผู้ขาย 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1:1



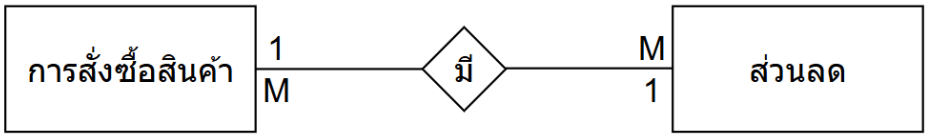
- **ผู้ขาย กับ สินค้า**
  - ผู้ขาย 1 คน ขายสินค้าได้หลายอย่าง
  - สินค้า 1 อย่าง ขายโดยผู้ขายหลายคน
  - ดังนั้น มีความสัมพันธ์แบบ M:M



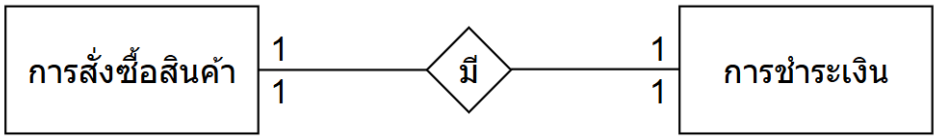
- **การสั่งซื้อสินค้า กับ สินค้า**
  - การสั่งซื้อสินค้า 1 ครั้ง สั่งซื้อสินค้าหลายอย่าง
  - สินค้า 1 อย่าง มีการสั่งซื้อสินค้าหลายครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ M:M



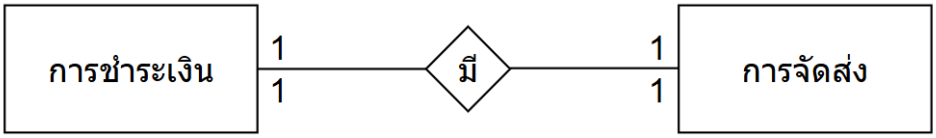
- **การสั่งซื้อสินค้า กับ ส่วนลด**
  - การสั่งซื้อสินค้า 1 ครั้ง ใช้ส่วนลดได้หลายส่วนลด
  - ส่วนลด 1 ส่วนลด ใช้ได้กับการสั่งซื้อสินค้าหลายครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ M:M



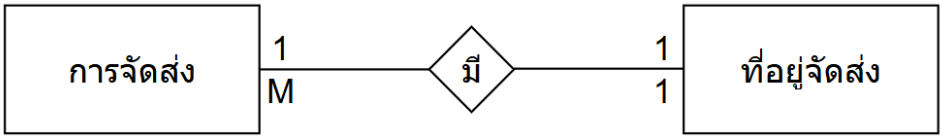
- **การสั่งซื้อสินค้า กับ การชำระเงิน**
  - การสั่งซื้อสินค้า 1 ครั้ง มีการชำระเงิน 1 ครั้ง
  - การชำระเงิน 1 ครั้ง มีการสั่งซื้อสินค้า 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1:1



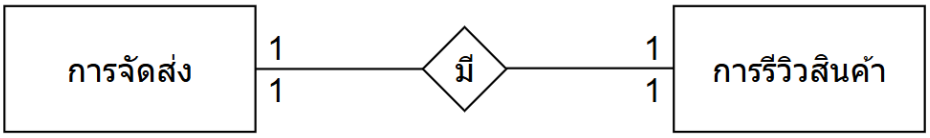
- **การชำระเงิน กับ การจัดส่ง**
  - การชำระเงิน 1 ครั้ง มีการจัดส่ง 1 การจัดส่ง
  - การจัดส่ง 1 ครั้ง มีการชำระเงิน 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1:1



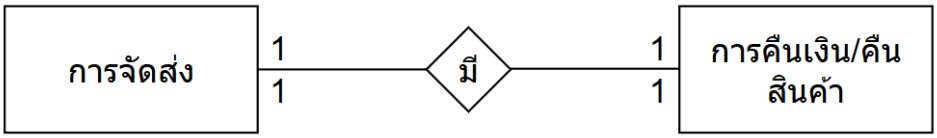
- **การจัดส่ง กับ ที่อยู่จัดส่ง**
  - การจัดส่ง 1 ครั้ง ส่งที่อยู่จัดส่ง 1 ที่อยู่
  - ที่อยู่จัดส่ง 1 ที่อยู่ ถูกจัดส่งได้หลายครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1:M



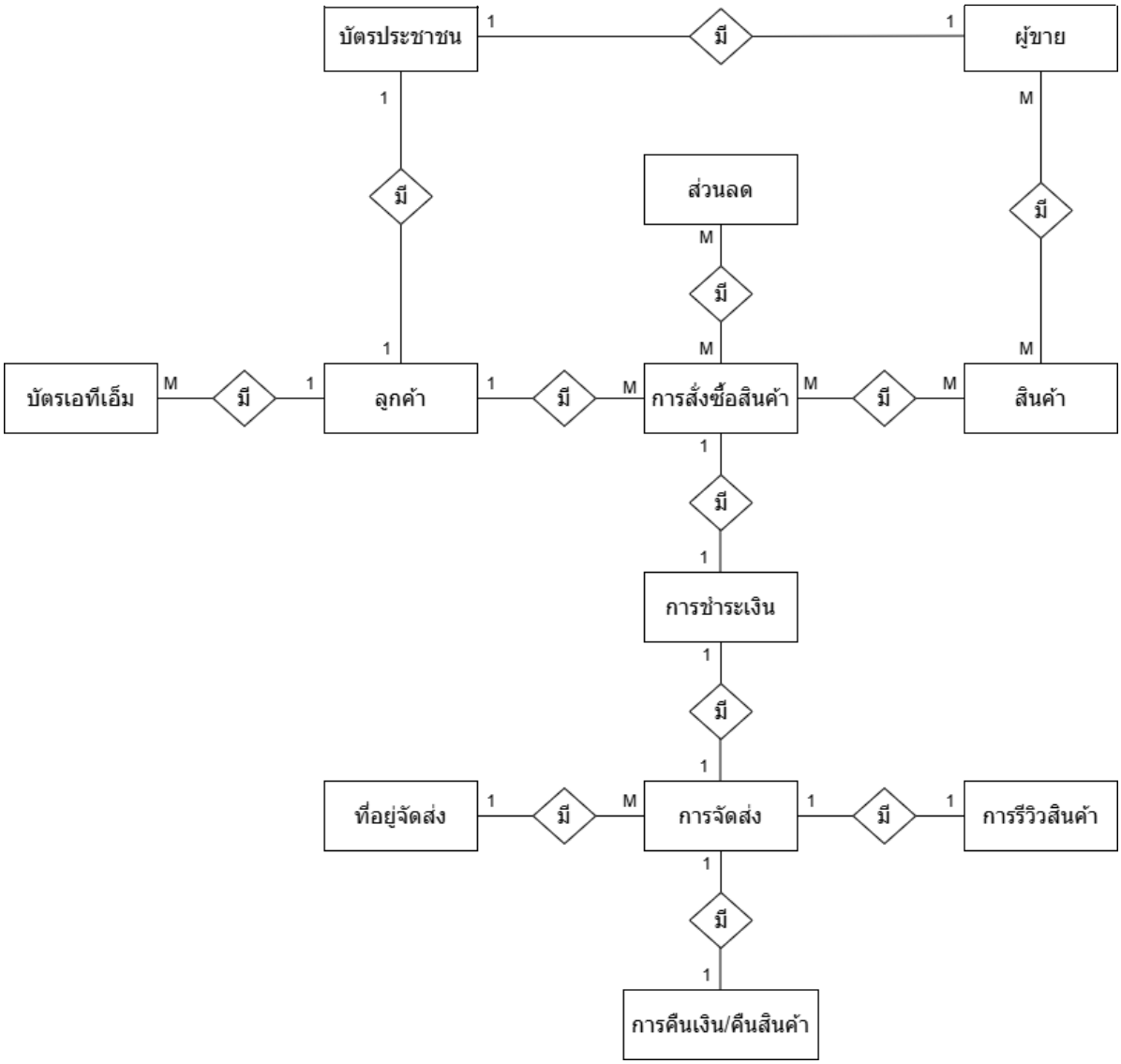
- **การจัดส่ง กับ การรีวิวสินค้า**
  - การจัดส่ง 1ครั้ง มีการรีวิวสินค้า 1 การรีวิว
  - การรีวิวสินค้า 1 ครั้ง มีการจัดส่ง 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1:1



- **การจัดส่ง กับ การคืนเงิน/คืนสินค้า**
  - การจัดส่ง 1 ครั้ง มีการคืนเงิน/คืนสินค้า 1 ครั้ง
  - การคืนเงิน/คืนสินค้า 1 ครั้ง มีการจัดส่ง 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1:1



Entity Relationship Diagram



3. การแปลงโมเดลเชิงแนวคิด ER-Diagram เป็น Relational Database Model

ตารางบัตรประชาชน

เลขบัตรประชาชน	ชื่อ	นามสกุล	เพศ	วันเกิด	เดือนเกิด	ปีเกิด

ตารางบัตรATM

เลขบัตรATM	ไอดีลูกค้า	ชื่อ	นามสกุล	ประเภท	ธนาคาร	เดือนหมดอายุบัตร	ปีที่หมดอายุบัตร

ตารางลูกค้า

ไอดีลูกค้า	เลขบัตรประชาชน	เบอร์โทร	อีเมล

ตารางสินค้า

รหัสสินค้า	ชื่อร้านค้า	ชื่อสินค้า	ประเภท	ราคา	วันที่นำเข้า	เดือนที่นำเข้า
ปีที่นำเข้า	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ			

ตารางส่วนลด

รหัสส่วนลด	ชื่อส่วนลด	ประเภทส่วนลด	มูลค่าของส่วนลด	จำนวนครั้งที่ใช้งานได้	วันที่เริ่มใช้ส่วนลด
เดือนที่เริ่มใช้ส่วนลด	ปีที่เริ่มใช้ส่วนลด	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ	

ตารางผู้ขาย

ชื่อร้านค้า	เลขบัตรประชาชน	พิกัด	เบอร์โทร	อีเมล

ตารางการสั่งซื้อสินค้า

ไอดีคำสั่งซื้อ	ไอดีลูกค้า	รหัสสินค้า	รหัสส่วนลด	จำนวน	ยอดรวม	ประเภทการจัดส่ง
ราคาจัดส่ง	ราคาสุทธิ	วันที่สั่งซื้อ	เดือนที่สั่งซื้อ	ปีที่สั่งซื้อ	เวลาที่สั่งซื้อ	สถานะการสั่งซื้อ

ตารางการชำระเงิน

รหัสการชำระเงิน	ไอดีคำสั่งซื้อ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระเงิน	สถานะการชำระเงิน

ตารางการจัดส่ง

หมายเลขพัสดุ	รหัสการชำระเงิน	ไอดีที่อยู่จัดส่ง	วันที่จัดส่ง	เดือนที่จัดส่ง	ปีที่จัดส่ง	เวลาที่จัดส่ง	บริษัท	สถานะการจัดส่

ตารางที่อยู่จัดส่ง

ไอดีที่อยู่จัดส่ง	บ้านเลขที่	หมู่	ซอย	ตำบล/แขวง	อำเภอ/เขต	จังหวัด	รหัสไปรษณีย

ตารางการรีวิวสินค้า

ไอดีรีวิว	หมายเลขพัสดุ	Pathรูปภาพ	รายละเอียด	คุณภาพสินค้า	บริการจากผู้ขาย	ผู้ให้บริการขนส่ง

ตารางการคืนเงิน/คืนสินค้า

ไอดีการคืนเงิน/คืนสินค้า	หมายเลขพัสดุ	วันที่ยกเลิก	เดือนที่ยกเลิก	ปีที่ยกเลิก	เวลาที่ยกเลิก	เหตุผลการคืนเงิน	วันที่คืนเงิน
เดือนที่คืนเงิน	ปีที่คืนเงิน	เวลาที่คืนเงิน	วันที่คืนสินค้า	เดือนที่คืนสินค้า	ปีที่คืนสินค้า	เวลาที่คืนสินค้า	

4.การทำบรรทัดฐาน(Normalization)

ตารางบัตรประชาชน

เลขบัตรประชาชน	ชื่อ	นามสกุล	เพศ	วันเกิด	เดือนเกิด	ปีเกิด
1 7299 00123 45 6	สมพงษ์	ใจดี	M	12	5	2536
1 7299 00234 56 7	สมร	ปรีชา	F	10	1	2544
1 7299 00345 67 8	สมศรี	ประเสริฐ	F	9	3	2548
1 7299 00456 78 9	สมฤทัย	คงทำ	F	25	6	2545
1 7299 00567 89 0	สมหมาย	ทรงทำ	M	26	9	2538
1 7494 00578 12 3	สมพงษ์	เลี้ยงถนอม	M	1	10	2547
1 7647 00155 98 2	สมพร	เลี้ยงถนอม	F	3	4	2512
1 7630 01789 54 1	ลูกเกด	สิริธรณ์	F	7	5	2555
1 7424 00055 88 4	แวมไพร์	ทไวไลท์	F	5	8	2539
1 7889 00050 14 7	สิงหา	ใจดี	M	27	6	2511

1NF : ตารางนี้มี เลขบัตรประชาชน เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางบัตร ATM

เลขบัตรATM	ไอดีลูกค้า	ชื่อ	นามสกุล	ประเภท	ธนาคาร	เดือนหมดอายุบัตร	ปีที่หมดอายุบัตร
1234 5678 9012 3456	MB2548	สมพงษ์	ใจดี	เครดิต	กสิกรไทย	9	28
9876 5432 1098 7654	MB2549	สมร	ปรีชา	เดบิต	กรุงไทย	11	29
4567 8901 2345 6789	MB2550	สมศรี	ประเสริฐ	เดบิต	กรุงเทพ	1	26
6543 2109 8765 4321	MB2551	สมฤทัย	คงทำ	เครดิต	ออมสิน	12	25
1111 2222 3333 4444	MB2552	สมหมาย	ทรงทำ	เครดิต	กรุงไทย	5	26

1NF : ตารางนี้มี เลขบัตร ATM เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางลูกค้า

ไอดีลูกค้า	เลขบัตรประชาชน	เบอร์โทร	อีเมล
MB2548	1 7299 00123 45 6	1-2345-6789	Somp@gmail.com
MB2549	1 7299 00234 56 7	2-3456-7891	Samorn@gmail.com
MB2550	1 7299 00345 67 8	3-4567-8912	Somsri@gmail.com
MB2551	1 7299 00456 78 9	4-5678-9123	Smrueth@gmail.com
MB2552	1 7299 00567 89 0	5-6789-1234	Sommhai@gmail.com

1NF : ตารางนี้มี ไอดีลูกค้า เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางสินค้า

รหัสสินค้า	ชื่อร้านค้า	ชื่อสินค้า	ประเภท	ราคา	วันที่นำเข้า
SA0001	ShopEase	ดีโด้	เครื่องดื่ม	10.00	15
SA0002	TrendzyMall	กาโตะ	เครื่องดื่ม	12.00	7
SA0003	QuickBuy	เลย์	ขนมคบเคี้ยว	20.00	2
SA0004	EzyMart	เถ้าแก่น้อย	ขนมคบเคี้ยว	10.00	21
SA0005	BestPick	โดโซะ	ขนมคบเคี้ยว	25.00	27

เดือนที่นำเข้า	ปีที่นำเข้า	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ
2	2565	16	5	2568
1	2567	8	1	2570
6	2567	3	6	2570
10	2567	22	10	2570
9	2568	28	9	2571

1NF : ตารางนี้มี รหัสสินค้า เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางส่วนลด

รหัสส่วนลด	ชื่อส่วนลด	ประเภทส่วนลด	มูลค่าของส่วนลด	จำนวนครั้งที่ใช้งานได้	วันที่เริ่มใช้ส่วนลด
DC0001	ซื้อครบ 300 บาท	ส่วนลด ร้านโค้ดคุ้ม	50.00	1000	7
DC0002	ซื้อครบ 150 บาท	ส่วนลดร้านค้า	100.00	500	7
DC0003	ซื้อครบ 100 บาท	ส่วนลด ร้านโค้ดคุ้ม	10.00	550	7
DC0004	ซื้อครบ 200 บาท	ส่วนลดร้านค้า	20.00	1000	8
DC0005	จัดส่งฟรี	จัดส่งฟรี	30.00	555	8

เดือนที่เริ่มใช้ส่วนลด	ปีที่เริ่มใช้ส่วนลด	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ
3	2568	14	3	2568
6	2568	14	6	2568
7	2568	14	7	2568
1	2568	15	1	2568
1	2568	15	1	2568

1NF : ตารางนี้มี รหัสส่วนลด เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางผู้ขาย

ชื่อร้านค้า	เลขบัตรประชาชน	พิกัด	เบอร์โทร	อีเมล
ShopEase	1 7494 00578 12 3	13.7465°N,100.5340°E	8-5514-3659	Shopease@gmail.com
TrendzyMall	1 7647 00155 98 2	13.7460°N,100.5296°E	8-7567-5312	Trendzymall@gmai.com
QuickBuy	1 7630 01789 54 1	13.6687°N,100.6042°E	9-8012-0893	Quickbuy@gmail.com
EzyMart	1 7424 00055 88 4	13.7370°N,100.5460°E	6-6413-9278	Ezymart@gmail.com
BestPick	1 7889 00050 14 7	13.7292°N,100.7750°E	6-6019-8723	Bestpick@gmail.com

1NF : ตารางนี้มี ชื่อร้านค้า เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการสั่งซื้อสินค้า

ไอดีคำสั่งซื้อ	ไอดีลูกค้า	รหัสสินค้า	รหัสส่วนลด	จำนวน	ยอดรวม	ประเภทการจัดส่ง
SM0001	MB2548	SA0001	DC0001	30	300.00	การจัดส่งด่วน
SM0002	MB2549	SA0002	DC0002	15	180.00	การจัดส่งแบบพัสดุลงทะเบียน
SM0003	MB2550	SA0003	DC0003	10	200.00	การจัดส่งด่วน
SM0004	MB2551	SA0004	DC0004	23	230.00	การจัดส่งระหว่างประเทศ
SM0005	MB2552	SA0005	DC0005	10	250.00	การจัดส่งแบบพัสดุลงทะเบียน
SM0006	MB2551	SA0005	DC0004	10	250.00	การจัดส่งแบบพัสดุลงทะเบียน

ราคาจัดส่ง	ราคาสุทธิ	วันที่สั่งซื้อ	เดือนที่สั่งซื้อ	ปีที่สั่งซื้อ	เวลาที่สั่งซื้อ	สถานะการสั่งซื้อ
50	250.00	11	1	2568	7:12:00	สำเร็จ
40	120.00	14	1	2568	6:00:00	สำเร็จ
50	240.00	24	1	2568	14:09:36	สำเร็จ
80	290.00	9	2	2568	11:16:48	สำเร็จ
40	260.00	14	2	2568	11:31:12	สำเร็จ
40	260.00	14	2	2568	12:31:12	ไม่สำเร็จ

1NF : ตารางนี้มี ไอดีคำสั่งซื้อ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการชำระเงิน

รหัสการชำระเงิน	ไอดีคำสั่งซื้อ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระเงิน	สถานะการชำระเงิน
PM0001	SM0001	11	1	2568	7:12:00	สำเร็จ
PM0002	SM0002	14	1	2568	0:00:00	สำเร็จ
PM0003	SM0003	24	1	2568	6:00:00	สำเร็จ
PM0004	SM0004	9	2	2568	8:38:24	สำเร็จ
PM0005	SM0005	14	2	2568	14:09:36	สำเร็จ
PM0006	SM0006	NULL	NULL	NULL	NULL	ไม่สำเร็จ

1NF : ตารางนี้มี รหัสการชำระเงิน เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการจัดส่ง

หมายเลขพัสดุ	รหัสการชำระเงิน	ไอดีที่อยู่จัดส่ง	วันที่จัดส่ง	เดือนที่จัดส่ง	ปีที่จัดส่ง	เวลาที่จัดส่ง	บริษัท	สถานะการจัดส่ง
1Z12345E1512345676	PM0001	AD0001	12	1	2568	00:00:00	Kerry Express	จัดส่งสำเร็จ
EA123456789TH	PM0002	AD0002	15	1	2568	07:12:00	Thai Post	กำลังจัดส่ง
JT123456789TH	PM0003	AD0003	NULL	NULL	NULL	NULL	NULL	ไม่สำเร็จ
NV123456789	PM0004	AD0004	10	2	2568	07:12:00	Ninja Van	จัดส่งสำเร็จ
JV123456789TH	PM0005	AD0005	15	2	2568	07:12:00	J&T Express	จัดส่งสำเร็จ

1NF : ตารางนี้มี หมายเลขพัสดุ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางที่อยู่จัดส่ง

ไอดีที่อยู่จัดส่ง	บ้านเลขที่	หมู่	ซอย	ตำบล/แขวง	อำเภอ/เขต	จังหวัด	รหัสไปรษณีย์
AD0001	20/1	4	2	บางแก้ว	บางพลี	สมุทรปราการ	10540
AD0002	10/29	7	1	ปากเกร็ด	ปากเกร็ด	นนทบุรี	11120
AD0003	134/24	9	5	เชียงใหม่	เชียงใหม่	เชียงใหม่	50000
AD0004	9/10	1	4	ท่าใหม่	ท่าใหม่	จันทบุรี	22120
AD0005	205/3	3	13	ท่าแร้ง	บางเขน	กรุงเทพมหานคร	10220

1NF : ตารางนี้มี ไอดีที่อยู่จัดส่ง เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว แต่มีการขึ้นต่อกันแบบ Transitive โดย รหัสไปรษณีย์ จะขึ้นอยู่กับ ตำบล/แขวง อำเภอ/เขต และ จังหวัด , ตำบล/แขวง ขึ้นอยู่กับ อำเภอ/เขต , อำเภอ/เขต ขึ้นอยู่กับ จังหวัด

แยก Attributes ที่มีการขึ้นต่อกันแบบ Transitive มาสร้างเป็นตารางใหม่ โดยให้ รหัสตำบล/แขวง เป็นคีย์หลัก (PK) ของตารางตำบล/แขวง , รหัสอำเภอ/เขต เป็นคีย์หลัก (PK) ของตารางอำเภอ/เขต , รหัสจังหวัด เป็นคีย์หลัก (PK) ของตารางจังหวัด และเก็บรหัสตำบล/แขวงไว้ในตารางเดิม เพื่อใช้เป็นคีย์นอกเชื่อมโยงกับตารางใหม่

ตารางที่อยู่จัดส่ง

ไอดีที่อยู่จัดส่ง	รหัสตำบล/แขวง	บ้านเลขที่	หมู่	ซอย
AD0001	A00001	20/1	4	2
AD0002	A00002	10/29	7	1
AD0003	A00003	134/24	9	5
AD0004	A00004	9/10	1	4
AD0005	A00005	205/3	3	13

ตารางตำบล/แขวง

รหัสตำบล/แขวง	รหัสอำเภอ/เขต	ตำบล/แขวง	รหัสไปรษณีย์
A00001	B00001	บางแก้ว	10540
A00002	B00002	ปากเกร็ด	11120
A00003	B00003	เชียงใหม่	50000
A00004	B00004	ท่าใหม่	22120
A00005	B00005	ท่าแร้ง	10220

ตารางอำเภอ/เขต

รหัสอำเภอ/เขต	รหัสจังหวัด	อำเภอ/เขต
B00001	C00001	บางพลี
B00002	C00002	ปากเกร็ด
B00003	C00003	เชียงใหม่
B00004	C00004	ท่าใหม่
B00005	C00005	บางเขน

ตารางจังหวัด

รหัสจังหวัด	จังหวัด
C00001	สมุทรปราการ
C00002	นนทบุรี
C00003	เชียงใหม่
C00004	จันทบุรี
C00005	กรุงเทพมหานคร

ตารางการรีวิวสินค้า

ไอดีรีวิว	หมายเลขพัสดุ	Pathรูปภาพ	รายละเอียด	คุณภาพสินค้า	บริการจากผู้ขาย	ผู้ให้บริการขนส่ง
RW0001	1Z12345E1512345676	NULL	NULL	NULL	85	80
RW0002	EA123456789TH	C:\Users\User\Pictures\dowload\guide-for-food-photography-wedio.jpg	NULL	NULL	80	85
RW0003	JT123456789TH	NULL	NULL	50	60	65
RW0004	NV123456789	NULL	แม่ค้าตอบดีมาก	NULL	NULL	NULL
RW0005	JV123456789TH	C:\Users\User\Pictures\dowload\OIP.jpg	ขนมอร่อย	95	95	95

1NF : ตารางนี้มี ไอดีรีวิว เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการคืนเงิน/คืนสินค้า

ไอดีการคืนเงิน/คืนสินค้า	หมายเลขพัสดุ	วันที่ยกเลิก	เดือนที่ยกเลิก	ปีที่ยกเลิก	เวลาที่ยกเลิก	เหตุผลการคืนเงิน	วันที่คืนเงิน
CC0001	1Z12345E1512345676	15	1	2568	7:12:00	ได้สินค้าไม่ครบ	15
CC0002	EA123456789TH	16	1	2568	8:38:24	ยกเลิกการสั่ง	16
CC0003	JT123456789TH	17	1	2568	0:28:48	ยกเลิกการสั่ง	17
CC0004	NV123456789	18	2	2568	10:48:00	ได้สินค้าไม่ครบ	18
CC0005	JV123456789TH	19	2	2568	8:38:24	เปลี่ยนสินค้า	19

เดือนที่คืนเงิน	ปีที่คืนเงิน	เวลาที่คืนเงิน	วันที่คืนสินค้า	เดือนที่คืนสินค้า	ปีที่คืนสินค้า	เวลาที่คืนสินค้า
1	2568	9:36:00	16	1	2568	9:48:00
1	2568	9:00:00	17	1	2568	10:50:00
1	2568	8:24:00	NULL	NULL	NULL	NULL
2	2568	10:05:00	19	2	2568	11:27:00
2	2568	8:50:00	20	2	2568	9:22:00

1NF : ตารางนี้มี ไอดีการคืนเงิน/คืนสินค้า เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุกAttributesอื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว แต่มีการขึ้นต่อกันแบบ Transitive โดย วันที่คืนสินค้า เดือนที่คืนสินค้า ปีที่คืนสินค้า เวลาที่คืนสินค้า จะขึ้นอยู่กับ หมายเลขพัสดุ

แยก Attributesที่มีการขึ้นต่อกันแบบ Transitive มาสร้างเป็นตารางใหม่ โดยให้ หมายเลขพัสดุเป็นคีย์หลัก (PK) ของตารางใหม่ และยังเก็บ หมายเลขพัสดุไว้ในตารางเดิม เพื่อใช้เป็นคีย์นอกเชื่อมโยงกับตารางใหม่

ตารางคืนเงิน

ไอดีการคืน	หมายเลขพัสดุ	วันที่ยกเลิก	เดือนที่ยกเลิก	ปีที่ยกเลิก	เวลาที่ยกเลิก	เหตุผลการคืนเงิน	วันที่คืนเงิน	เดือนที่คืนเงิน	ปีที่คืนเงิน	เวลาที่คืนเงิน
CC0001	1Z12345E1512345676	15	1	2568	7:12:00	ได้สินค้าไม่ครบ	15	1	2568	9:36:00
CC0002	EA123456789TH	16	1	2568	8:38:24	เปลี่ยนสินค้า	16	1	2568	9:00:00
CC0003	JT123456789TH	17	1	2568	0:28:48	ยกเลิกการสั่ง	17	1	2568	8:24:00
CC0004	NV123456789	18	2	2568	10:48:00	ได้สินค้าไม่ครบ	18	2	2568	10:05:00
CC0005	JV123456789TH	19	2	2568	8:38:24	เปลี่ยนสินค้า	19	2	2568	8:50:00

ตารางคืนสินค้า

หมายเลขพัสดุ	วันที่คืนสินค้า	เดือนที่คืนสินค้า	ปีที่คืนสินค้า	เวลาที่คืนสินค้า
1Z12345E1512345676	16	1	2568	9:48:00
EA123456789TH	17	1	2568	10:50:00
JT123456789TH	NULL	NULL	NULL	NULL
NV123456789	19	2	2568	11:27:00
JV123456789TH	20	2	2568	9:22:00

New Relational Database Model

ตารางบัตรประชาชน

เลขบัตรประชาชน	ชื่อ	นามสกุล	เพศ	วันเกิด	เดือนเกิด	ปีเกิด

ตารางบัตรATM

เลขบัตรATM	ไอดีลูกค้า	ชื่อ	นามสกุล	ประเภทบัตร	ธนาคาร	เดือนหมดอายุบัตร	ปีที่หมดอายุบัตร

ตารางลูกค้า

ไอดีลูกค้า	เลขบัตรประชาชน	เบอร์โทร	อีเมล

ตารางสินค้า

รหัสสินค้า	ชื่อร้านค้า	ชื่อสินค้า	ประเภท	ราคา	วันที่นำเข้า
เดือนที่นำเข้า	ปีที่นำเข้า	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ	

ตารางส่วนลด

รหัสส่วนลด	ชื่อส่วนลด	ประเภทส่วนลด	มูลค่าของส่วนลด	จำนวนครั้งที่ใช้งานได้	วันที่เริ่มใช้ส่วนลด
เดือนที่เริ่มใช้ส่วนลด	ปีที่เริ่มใช้ส่วนลด	วันหมดอายุ	เดือนที่หมดอายุ	ปีที่หมดอายุ	

ตารางผู้ขาย

ชื่อร้านค้า	เลขบัตรประชาชน	พิกัด	เบอร์โทร	อีเมล

ตารางการสั่งซื้อสินค้า

ไอดีคำสั่งซื้อ	ไอดีลูกค้า	รหัสสินค้า	รหัสส่วนลด	จำนวน	ยอดรวม	ประเภทการจัดส่ง
ราคาจัดส่ง	ราคาสุทธิ	วันที่สั่งซื้อ	เดือนที่สั่งซื้อ	ปีที่สั่งซื้อ	เวลาที่สั่งซื้อ	สถานะการสั่งซื้อ

ตารางการชำระเงิน

รหัสการชำระเงิน	ไอดีคำสั่งซื้อ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระเงิน	สถานะการชำระเงิน

ตารางการจัดส่ง

หมายเลขพัสดุ	รหัสการชำระเงิน	ไอดีที่อยู่จัดส่ง	วันที่จัดส่ง	เดือนที่จัดส่ง	ปีที่จัดส่ง	เวลาที่จัดส่ง	บริษัท	สถานะการจัดส่

ตารางที่อยู่จัดส่ง

ไอดีที่อยู่จัดส่ง	รหัสตำบล/แขวง	บ้านเลขที่	หมู่	ซอย

ตารางตำบล/แขวง

รหัสตำบล/แขวง	รหัสอำเภอ/เขต	ตำบล/แขวง	รหัสไปรษณีย์

ตารางอำเภอ/เขต

รหัสอำเภอ/เขต	รหัสจังหวัด	อำเภอ/เขต

ตารางจังหวัด

รหัสจังหวัด	จังหวัด

ตารางการรีวิวสินค้า

ไอดีรีวิว	หมายเลขพัสดุ	Pathรูปภาพ	รายละเอียด

คุณภาพสินค้า	บริการจากผู้ขาย	ผู้ให้บริการขนส่ง

ตารางการคืนเงิน

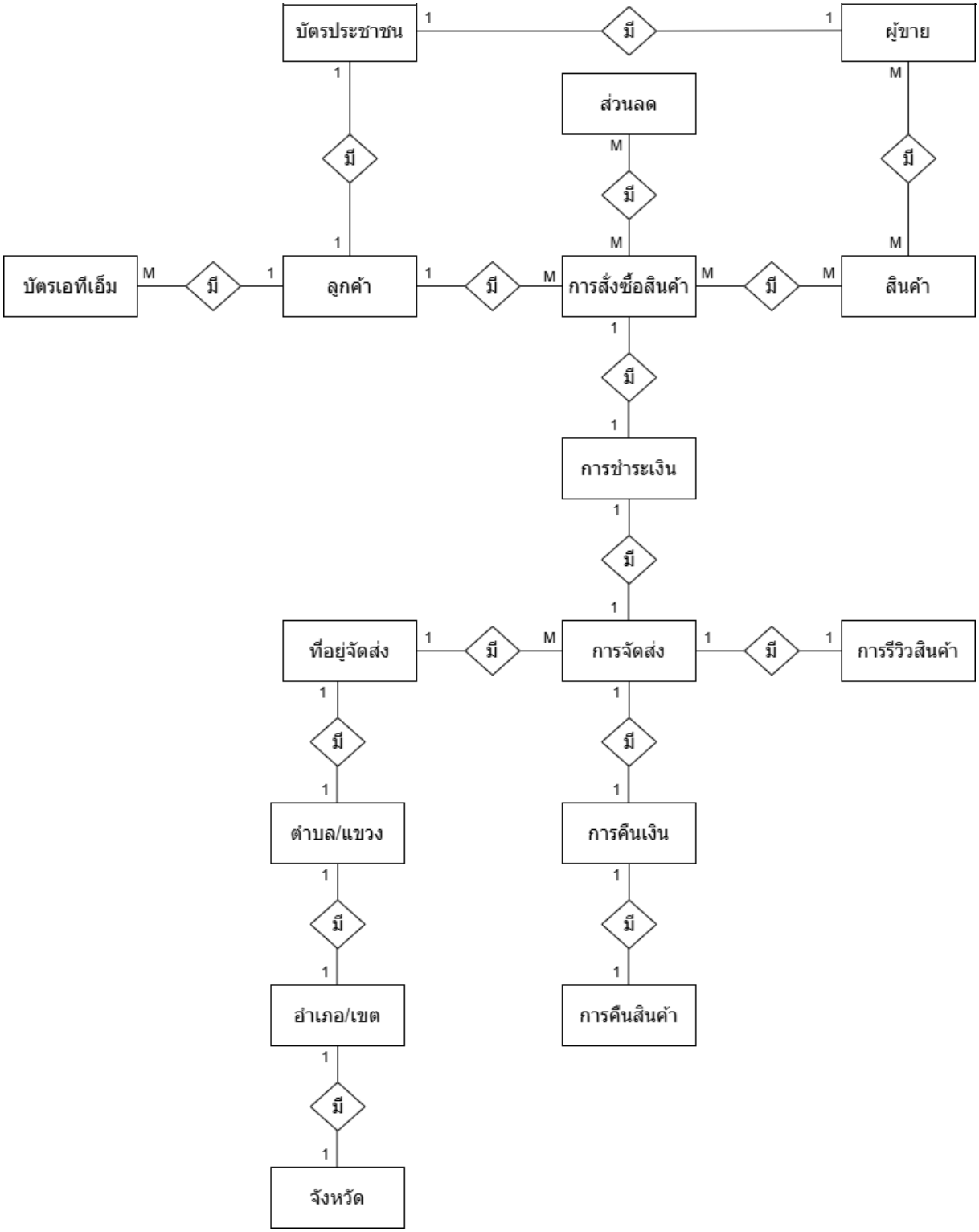
ไอดีการคืน	หมายเลขพัสดุ	วันที่ยกเลิก	เดือนที่ยกเลิก	ปีที่ยกเลิก	เวลาที่ยกเลิก	เหตุผลการคืนเงิน

วันที่คืนเงิน	เดือนที่คืนเงิน	ปีที่คืนเงิน	เวลาที่คืนเงิน

ตารางการคืนสินค้า

หมายเลขพัสดุ	วันที่คืนสินค้า	เดือนที่คืนสินค้า	ปีที่คืนสินค้า	เวลาที่คืนสินค้า

New Entity Relationship Diagram



5.พจนานุกรมข้อมูล (Data Dictionary)

ตารางบัตรประชาชน

Attributes	Data Type	Description
ID_card (เลขบัตรประชาชน)	CHAR(13)	เลขบัตรประจำตัวประชาชน 13 หลัก
First_name (ชื่อ)	VARCHAR(20)	ชื่อจริงบุคคล
Last_name (นามสกุล)	VARCHAR(20)	นามสกุลบุคคล
Gender(เพศ)	CHAR(1)	ระบุเพศ M-ชาย F-หญิง
Birth_day (วันเกิด)	TINYINT	วันที่เกิด (1-31)
Birth_month (เดือนเกิด)	TINYINT	เดือนที่เกิด (1-12)
Birth_year (ปีเกิด)	SMALLINT	ปีพุทธศักราชที่เกิด

ตารางบัตร ATM

Attributes	Data Type	Description
ID_ATM (เลขบัตรเอทีเอ็ม)	VARCHAR(16)	หมายเลขบัตร ATM 16 หลัก
First_name (ชื่อ)	VARCHAR(20)	ชื่อจริงเจ้าของบัตร ATM
Last_name (นามสกุล)	VARCHAR(20)	นามสกุลเจ้าของบัตร ATM
Card_type (ประเภทบัตร)	VARCHAR(20)	ประเภทของบัตร Debit - เดบิต Credit - เครดิต
Bank (ธนาคาร)	VARCHAR(100)	ชื่อธนาคารที่ออกบัตร
expiration_month (เดือนบัตรหมดอายุ)	TINYINT	เดือนที่หมดอายุของบัตร (1-12)
expiration_year (ปีที่บัตรหมดอายุ)	SMALLINT	ปีคริสต์ศักราชที่บัตรหมดอายุ
ID_Customer (ไอดีลูกค้า)	CHAR(6)	รหัสประจำตัวลูกค้า 6 หลัก

ตารางลูกค้า

Attributes	Data Type	Description
ID_Customer (ไอดีลูกค้า)	CHAR(6)	รหัสประจำตัวลูกค้า 6 หลัก
Phone_number (เบอร์โทร)	CHAR(12)	หมายเลขโทรศัพท์ลูกค้า 9 หลัก
Email (อีเมล)	VARCHAR(40)	อีเมลลูกค้า
ID_card (เลขบัตรประชาชน)	CHAR(13)	เลขบัตรประจำตัวประชาชน 13 หลัก

ตารางสินค้า

Attributes	Data Type	Description
Product_code (รหัสสินค้า)	CHAR(6)	รหัสประจำสินค้า 6 หลัก
Product_name (ชื่อสินค้า)	VARCHAR(50)	ชื่อของสินค้า
Product_type (ประเภท)	VARCHAR(50)	หมวดหมู่ของสินค้า A
Price (ราคา)	DECIMAL(8,2)	ราคาของสินค้า ที่มีจำนวนทั้งหมด 8 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 6 หลัก เช่น 4853.98
Import_day (วันที่นำเข้า)	TINYINT	วันที่นำเข้าสินค้า (1-31)
Import_month (เดือนที่นำเข้า)	TINYINT	เดือนที่นำเข้าสินค้า (1-12)
Import_year (ปีที่นำเข้า)	SMALLINT	ปีพุทธศักราชที่นำเข้าสินค้า
Expire_day (วันหมดอายุ)	TINYINT	วันที่หมดอายุของสินค้า (1-31)
Expire_month (เดือนที่หมดอายุ)	TINYINT	เดือนที่หมดอายุของงสินค้า (1-12)
Expire_year (ปีที่หมดอายุ)	SMALLINT	ปีพุทธศักราชที่หมดอายุของสินค้า
Shop_name (ชื่อร้านค้า)	VARCHAR(50)	ชื่อของร้านค้า

ตารางส่วนลด

Attributes	Data Type	Description
Discount_code (รหัสส่วนลด)	CHAR(6)	รหัสประจำส่วนลด 6 หลัก
Discount_name (ชื่อส่วนลด)	VARCHAR(50)	ชื่อของส่วนลด
Discount_type (ประเภทส่วนลด)	VARCHAR(50)	หมวดหมู่ของส่วนลด
Discount_value (มูลค่าของส่วนลด)	DECIMAL(8,2)	มูลค่าของส่วนลดที่มีจำนวนทั้งหมด 8 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 6 หลัก เช่น 4853.98
Usage_limit (จำนวนครั้งที่ใช้งานได้)	INT	จำนวนครั้งที่สามารถใช้ส่วนลด
Start_day (วันที่เริ่มใช้ส่วนลด)	TINYINT	วันที่สามารถเริ่มใช้ส่วนลด (1-31)
Start_month (เดือนที่เริ่มใช้ส่วนลด)	TINYINT	เดือนที่สามารถเริ่มใช้ส่วนลด (1-12)
Start_year (ปีที่เริ่มใช้ส่วนลด)	SMALLINT	ปีพุทธศักราชที่สามารถเริ่มใช้ส่วนลด
End_day (วันหมดอายุ)	TINYINT	วันที่หมดอายุของส่วนลด (1-31)
End_month (เดือนที่หมดอายุ)	TINYINT	เดือนที่หมดอายุของส่วนลด (1-12)
End_year (ปีที่หมดอายุ)	SMALLINT	ปีพุทธศักราชที่หมดอายุส่วนลด

ตารางผู้ขาย

Attributes	Data Type	Description
Shop_name (ชื่อร้านค้า)	VARCHAR(50)	ชื่อของร้านค้า
Location (พิกัด)	VARCHAR(100)	ตำแหน่งของร้านค้า
Phone_number (เบอร์โทร)	CHAR(12)	หมายเลขโทรศัพท์ 9 หลักของร้านค้า
Email (อีเมล)	VARCHAR(40)	อีเมลของร้านค้า
ID_card (เลขบัตรประชาชน)	CHAR(13)	หมายเลขประจำตัวประชาชนของผู้ขาย

ตารางการสั่งซื้อสินค้า

Attributes	Data Type	Description
Order_ID (ไอดีคำสั่งซื้อ)	CHAR(6)	รหัสคำสั่งซื้อ 6 หลัก
Quantity (จำนวน)	INT	จำนวนสินค้าที่สั่งซื้อ
Total_amount (ยอดรวม)	DECIMAL(8, 2)	ราคารวมสินค้าที่สั่งซื้อก่อนหักส่วนลด ที่มีจำนวนทั้งหมด 8 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 6 หลัก เช่น 4853.98
Shipping_type (ประเภทการจัดส่ง)	VARCHAR(50)	รูปแบบการจัดส่ง - การจัดส่งด่วน (Express Delivery) - การจัดส่งแบบพัสดุลงทะเบียน (Registered Parcel Delivery) - การจัดส่งระหว่างประเทศ (International Delivery)
Net_amount (ราคาสุทธิ)	DECIMAL(8, 2)	ราคาสุทธิหลังหักส่วนลด ที่มีจำนวนทั้งหมด 8 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 6 หลัก เช่น 4853.98
Shipping_price (ราคาจัดส่ง)	DECIMAL(8, 2)	ค่าจัดส่งสินค้า ที่มีจำนวนทั้งหมด 8 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 6 หลัก เช่น 4853.98
Order_day (วันที่สั่งซื้อ)	TINYINT	วันที่ทำการสั่งซื้อ (1-31)
Order_month (เดือนที่สั่งซื้อ)	TINYINT	เดือนที่ทำการสั่งซื้อ (1-12)
Order_year (ปีที่สั่งซื้อ)	SMALLINT	ปีพุทธศักราชที่ทำการสั่งซื้อ
Order_time (เวลาที่สั่งซื้อ)	TIME	เวลาที่ทำการสั่งซื้อ
ID_Customer (ไอดีลูกค้า)	CHAR(6)	รหัสประจำตัวลูกค้า 6 หลัก
Product_code (รหัสสินค้า)	CHAR(6)	รหัสประจำสินค้า 6 หลัก
Discount_code (รหัสส่วนลด)	CHAR(6)	รหัสประจำส่วนลด 6 หลัก
status_order (สถานะการสั่งซื้อ)	VARCHAR(20)	สถานะการสั่งซื้อ กรณีที่ไม่ชำระเงินภายใน 24 ชั่วโมงการสั่งซื้อจะถูกยกเลิก - สำเร็จ - ไม่สำเร็จ

ตารางการชำระเงิน

Attributes	Data Type	Description
ID_Payment (รหัสการชำระเงิน)	CHAR(6)	รหัสการชำระเงิน 6 หลัก
Payment_Date (วันที่ชำระเงิน)	TINYINT	วันที่ทำการชำระเงิน (1-31)
Payment_Month (เดือนที่ชำระเงิน)	TINYINT	เดือนที่ทำการชำระเงิน (1-12)
Payment_Year (ปีที่ชำระเงิน)	SMALLINT	ปีพุทธศักราชที่ทำการชำระเงิน
Payment_Time (เวลาที่ชำระเงิน)	TIME	เวลาที่ทำการชำระเงิน
Order_ID (ไอดีคำสั่งซื้อ)	CHAR(6)	รหัสคำสั่งซื้อ 6 หลัก
Payment_Status (สถานะการชำระเงิน)	VARCHAR(10)	สถานะการชำระเงิน - สำเร็จ - ไม่สำเร็จ

ตารางการจัดส่ง

Attributes	Data Type	Description
ID_Shipping (หมายเลขพัสดุ)	VARCHAR(20)	หมายเลขติดตามพัสดุของบริษัทขนส่ง
Shipping_Date (วันที่จัดส่ง)	TINYINT	วันที่เริ่มทำการจัดส่ง (1-31)
Shipping_Month (เดือนที่จัดส่ง)	TINYINT	เดือนที่เริ่มทำการจัดส่ง (1-12)
Shipping_Year (ปีที่จัดส่ง)	SMALLINT	ปีพุทธศักราชที่เริ่มทำการจัดส่ง
Shipping_Time (เวลาที่จัดส่ง)	TIME	เวลาที่เริ่มทำการจัดส่ง
Company (บริษัท)	VARCHAR(20)	บริษัทขนส่ง
ID_Payment (รหัสการชำระเงิน)	CHAR(6)	รหัสการชำระเงิน 6 หลัก
Shipping_address_ID (ไอดีที่อยู่จัดส่ง)	CHAR(6)	รหัสอ้างอิงที่อยู่สำหรับการจัดส่งของลูกค้า 6 หลัก
Shipping_Status (สถานะการจัดส่ง)	VARCHAR(20)	สถานะการจัดส่ง In Transit - กำลังจัดส่ง Delivered – จัดส่งสำเร็จ Failed - จัดส่งไม่สำเร็จ

ตารางที่อยู่จัดส่ง

Attributes	Data Type	Description
Shipping_address_ID (ไอดีที่อยู่จัดส่ง)	CHAR(6)	รหัสอ้างอิงที่อยู่สำหรับการจัดส่งของลูกค้า 6 หลัก
House_number (บ้านเลขที่)	VARCHAR(20)	เลขที่บ้าน/เลขที่อาคาร
Swine (หมู่)	VARCHAR(10)	เลขที่หมู่บ้าน
Alley (ซอย)	VARCHAR(20)	ชื่อซอย (ถ้ามี)
Subdistrict_code (รหัสตำบล/แขวง)	CHAR(6)	รหัสตำบล/แขวง 6 หลัก

ตารางตำบล/แขวง

Attributes	Data Type	Description
Subdistrict_code (รหัสตำบล/แขวง)	CHAR(6)	รหัสตำบล/แขวง 6 หลัก
Subdistrict_name (ตำบล/แขวง)	VARCHAR(50)	ชื่อตำบล (สำหรับพื้นที่ต่างจังหวัด) หรือแขวง (สำหรับพื้นที่กรุงเทพฯ)
Postal_code (รหัสไปรษณีย์)	CHAR(5)	รหัสไปรษณีย์ 5 หลัก
District_code (รหัสอำเภอ/เขต)	CHAR(6)	รหัสอำเภอ/เขต 6 หลัก

ตารางอำเภอ/เขต

Attributes	Data Type	Description
District_code (รหัสอำเภอ/เขต)	CHAR(6)	รหัสอำเภอ/เขต 6 หลัก
District_name (อำเภอ/เขต)	VARCHAR(50)	ชื่ออำเภอ(สำหรับพื้นที่ต่างจังหวัด) หรือเขต (สำหรับพื้นที่กรุงเทพฯ)
Province_code (รหัสจังหวัด)	CHAR(6)	รหัสจังหวัด 6 หลัก

ตารางจังหวัด

Attributes	Data Type	Description
Province_code (รหัสจังหวัด)	CHAR(6)	รหัสจังหวัด 6 หลัก
Province_name (จังหวัด)	VARCHAR(50)	ชื่อจังหวัด

ตารางการรีวิวสินค้า

Attributes	Data Type	Description
Review_ID (ไอดีการรีวิว)	CHAR(6)	รหัสไอดีการรีวิว 6 หลัก
Image_path (Path รูปภาพ)	BLOB	รูปภาพการรีวิว
Review_details (รายละเอียด)	TEXT	รายละเอียดการรีวิว
Product_quality (คุณภาพสินค้า)	DECIMAL(5,2)	เปอร์เซ็นคะแนนของคุณภาพสินค้า ที่มีจำนวนทั้งหมด 5 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 3 หลัก เช่น 99.99%
Seller_service (บริการจากผู้ขาย)	DECIMAL(5,2)	เปอร์เซ็นคะแนนของการบริการจากผู้ขายที่มีจำนวนทั้งหมด 5 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 3 หลัก เช่น 99.99%
Delivery_service (ผู้ให้บริการขนส่ง)	DECIMAL(5,2)	เปอร์เซ็นคะแนนของผู้ให้บริการขนส่งที่มีจำนวนทั้งหมด 5 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 3 หลัก เช่น 99.99%
ID_Shipping (หมายเลขพัสดุ)	VARCHAR(20)	หมายเลขติดตามพัสดุของบริษัทขนส่ง

ตารางการคืนเงิน

Attributes	Data Type	Description
ID_Refund (ไอดีการคืนเงิน)	CHAR(6)	รหัสการคืนเงิน 6 หลัก
Cancel_Date (วันที่ยกเลิก)	TINYINT	วันที่ยกเลิกคำสั่งซื้อ (1-31)
Cancel_Month (เดือนที่ยกเลิก)	TINYINT	เดือนที่ยกเลิกคำสั่งซื้อ (1-12)
Cancel_Year (ปีที่ยกเลิก)	SMALLINT	ปีพุทธศักราชที่ยกเลิกคำสั่งซื้อ
Cancel_Time (เวลาที่ยกเลิก)	TIME	เวลาที่ยกเลิกคำสั่งซื้อ
Reason_Refund (เหตุผลการยกเลิก)	VARCHAR(100)	เหตุผลที่คืนเงิน
Ref_Date (วันที่คืนเงิน)	TINYINT	วันที่ทำการคืนเงิน (1-31)
Ref_Month (เดือนที่คืนเงิน)	TINYINT	เดือนที่ทำการคืนเงิน (1-12)
Ref_Year (ปีที่คืนเงิน)	SMALLINT	ปีพุทธศักราชที่ทำการคืนเงิน
Ref_Time (เวลาที่คืนเงิน)	TIME	เวลาที่ทำการคืนเงิน
ID_Shipping (หมายเลขพัสดุ)	VARCHAR(20)	หมายเลขติดตามพัสดุของบริษัทขนส่ง

ตารางการคืนสินค้า

Attributes	Data Type	Description
ID_Shipping (หมายเลขพัสดุ)	VARCHAR(20)	หมายเลขติดตามพัสดุของบริษัทขนส่ง
RetPro_Date (วันที่คืนสินค้า)	TINYINT	วันที่คืนสินค้าให้กับลูกค้า (1-31)
RetPro_Month (เดือนที่คืนสินค้า)	TINYINT	เดือนที่คืนสินค้าให้กับลูกค้า (1-12)
RetPro_Year (ปีที่คืนสินค้า)	SMALLINT	ปีพุทธศักราชที่คืนสินค้าให้กับลูกค้า
RetPro_Time (เวลาที่คืนสินค้า)	TIME	เวลาที่คืนสินค้าให้กับลูกค้า

6.การสร้างฐานข้อมูลด้วยภาษา SQL โดยใช้ MySQL Workbench

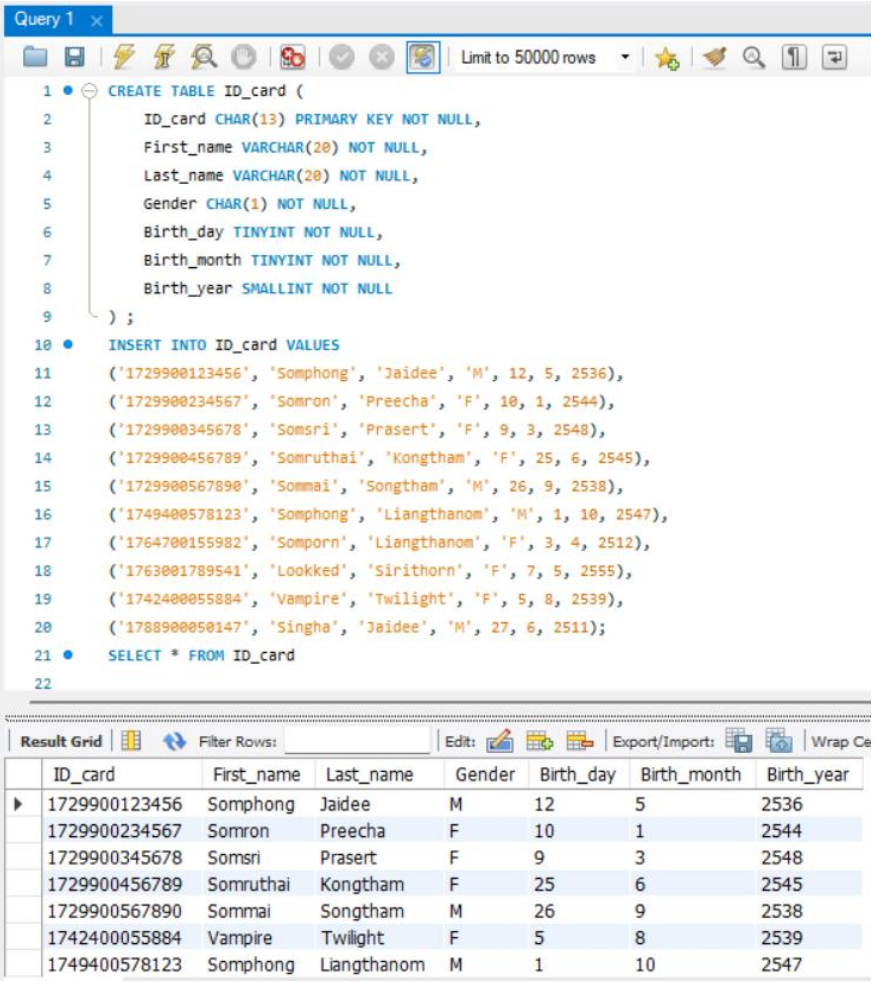
การสร้างตาราง **บัตรประชาชน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางบัตรประชาชนขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Card เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM ID_Card แสดงผลของตารางที่ได้ผ่านการ Query



ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางบัตรประชาชน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_birth_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนเกิด(Birth_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่เกิด (Birth _day) อยู่ในช่วง 1 ถึง 31 หรือไม่

  หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

  1. ตรวจสอบว่าเป็นเดือนที่เกิด (Birth _month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เกิด ไม่เกิน 29

```
1  ALTER TABLE ID_card
2  ADD CONSTRAINT chk_birth_date
3  CHECK (
4      Birth_month BETWEEN 1 AND 12 AND
5      Birth_day BETWEEN 1 AND 31 AND
6      (
7          (Birth_month IN (1, 3, 5, 7, 8, 10, 12)) OR
8          (Birth_month IN (4, 6, 9, 11) AND Birth_day <= 30) OR
9          (Birth_month = 2 AND Birth_day <= 29)
10     )
11 );
```

ทดสอบการใส่ค่า โดยใส่วันที่ 30 เดือน 2 ซึ่งโดยที่กำหนดไว้ เดือนที่ 2 สามารถเพิ่มค่าได้ไม่เกิน 29 วันทำให้ไม่สามารถเพิ่มค่าได้

```
1  INSERT INTO id_card VALUES ('1505000789123', 'Decha', 'Kengkla', 'M', 30, 2, 2551);
```

Output

Action Output

#	Time	Action	Message
1	18:04:03	INSERT INTO id_card VALUES ('1505000789123', 'Decha', 'Kengkla', 'M', 30, 2, 2551)	Error Code: 3819. Check constraint 'chk_birth_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางบัตรประชาชน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_card_gender

CHECK เพศ(Gender) กำหนดค่า 2 ค่า คือ M - Male ผู้ชาย และ F - Female ผู้หญิง

```
1  ALTER TABLE id_card
2  ADD CONSTRAINT chk_card_gender
3  CHECK (Gender IN ('M', 'F'));
```

ทดลองใส่ค่าเป็น MF ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร M,F เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
5  INSERT INTO ID_card VALUES ('1303000987654', 'Wilai', 'Madee', 'T', 8, 7, 2546);
```

Output

Action Output

#	Time	Action	Message
1	21:43:49	INSERT INTO ID_card VALUES ('1303000987654', 'Wilai', 'Madee', 'T', 8, 7, 2546)	Error Code: 3819. Check constraint 'chk_card_gender' is violated.

การสร้างตาราง **ลูกค้า** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตาราง ลูกค้า ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_ Customer เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_card เป็น FK

และอ้างอิงจาก ตาราง id_card

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Customers แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Customers (
    ID_Customer CHAR(6) PRIMARY KEY NOT NULL,
     ID_card CHAR(13) UNIQUE NOT NULL,
    Phone_number VARCHAR(11) NOT NULL,
    Email VARCHAR(40) NOT NULL,
    FOREIGN KEY ( ID_card) REFERENCES id_card(ID_card)
);
INSERT INTO customers VALUES
('MB2548', '1729900123456', '1-2345-6789', 'Somp@gmail.com'),
('MB2549', '1729900234567', '2-3456-7891', 'Samorn@gmail.com'),
('MB2550', '1729900345678', '3-4567-8912', 'Somsri@gmail.com'),
('MB2551', '1729900456789', '4-5678-9123', 'Smrueth@gmail.com'),
('MB2552', '1729900567890', '5-6789-1234', 'Sommhai@gmail.com');
SELECT * FROM Customers
```

ID_Customer	ID_card	Phone_number	Email
MB2548	1729900123456	1-2345-6789	Somp@gmail.com
MB2549	1729900234567	2-3456-7891	Samorn@gmail.com
MB2550	1729900345678	3-4567-8912	Somsri@gmail.com
MB2551	1729900456789	4-5678-9123	Smrueth@gmail.com
MB2552	1729900567890	5-6789-1234	Sommhai@gmail.com
NULL	NULL	NULL	NULL

การสร้างตาราง **บัตร** ATM โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางบัตรATMขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_ATM เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Customer เป็น FK และอ้างอิงจากตาราง customers

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM ATM แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE ATM (
    ID_ATM VARCHAR(16) PRIMARY KEY NOT NULL,
    ID_Customer CHAR(6) NOT NULL,
    First_name VARCHAR(20) NOT NULL,
    Last_name VARCHAR(20) NOT NULL,
    Card_type VARCHAR(20) NOT NULL,
    Bank VARCHAR(100) NOT NULL,
    expiration_month TINYINT NOT NULL,
    expiration_year SMALLINT NOT NULL,
    FOREIGN KEY (ID_Customer) REFERENCES customers(ID_Customer)
);
INSERT INTO ATM VALUES
('1234567890123456', 'MB2548', 'Somphong', 'Jaidee', 'Credit', 'Kasikorn Thai', 9, 28),
('9876543210987654', 'MB2549', 'Somron', 'Preecha', 'Debit', 'Krung Thai', 11, 29),
('4567890123456789', 'MB2550', 'Somsri', 'Prasert', 'Debit', 'Bangkok', 1, 26),
('8543210987654321', 'MB2551', 'Somruthai', 'Kongtham', 'Credit', 'Government Savings Bank', 12, 25),
('1111222233334444', 'MB2552', 'Sommai', 'Songtham', 'Credit', 'Krung Thai', 5, 26);
SELECT * FROM ATM;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ID_ATM	ID_Customer	First_name	Last_name	Card_type	Bank	expiration_month	expiration_year
1111222233334444	MB2552	Sommai	Songtham	Credit	Krung Thai	5	26
1234567890123456	MB2548	Somphong	Jaidee	Credit	Kasikorn Thai	9	28
4567890123456789	MB2550	Somsri	Prasert	Debit	Bangkok	1	26
8543210987654321	MB2551	Somruthai	Kongtham	Credit	Government Savings Bank	12	25
9876543210987654	MB2549	Somron	Preecha	Debit	Krung Thai	11	29
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางบัตรATM

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_atm_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนหมดอายุบัตร(expiration_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าปีหมดอายุบัตร(expiration_year) มากกว่าหรือเท่ากับปีปัจจุบัน ปี 2025(25) หรือไม่

```sql
ALTER TABLE atm
ADD CONSTRAINT chk_atm_date
CHECK (
    expiration_month BETWEEN 1 AND 12
    AND expiration_year >= 25
);
```

ทดลองเพิ่มค่าโดยใส่ เดือนที่ 5 ปีที่ 24 ซึ่งปีที่หมดอายุบัตร(expiration_year) กำหนดไว้ว่าต้องมากกว่าหรือเท่ากับปี 25 จึงทำให้ไม่สามารถเพิ่มค่าได้

```sql
INSERT INTO ATM VALUES
('11112222333355555', 'MB2552', 'Sommai', 'Songtham', 'Credit', 'Krung Thai', 5, 24);
```

Output

Action Output

#	Time	Action	Message
1	22:52:04	INSERT INTO ATM VALUES ('11112222333355555', 'MB2552', 'Sommai', 'Songtham', 'Credit', 'Krung Thai', 5, ...	Error Code: 1406. Data too long for column 'ID_ATM' at row 1

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางบัตรATM

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_card_type

CHECK ประเภทของบัตรเอทีเอ็ม(Card_type) มี 2 ประเภท คือ Debit, Credit

```sql
ALTER TABLE atm
ADD CONSTRAINT chk_card_type
CHECK (Card_type IN ('Debit', 'Credit'));
```

ทดลองใส่ค่าเป็นคำว่า Credit_card ซึ่งไม่ตรงกับเงื่อนไขที่กำหนดไว้ ทำให้ไม่สามารถเพิ่มค่าได้

```sql
INSERT INTO ATM VALUES ('1234567877777777', 'MB2548', 'Somphong', 'Jaidee', 'Credit_card', 'Kasikorn Thai', 9, 28);
```

Output

Action Output

#	Time	Action	Message
1	21:14:56	INSERT INTO ATM VALUES ('1234567877777777', 'MB2548', 'Somphong', 'Jaidee', 'Credit_card', 'Kasikorn Th...	Error Code: 3819. Check constraint 'chk_card_type' is violated.

การสร้างตาราง **ส่วนลด** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตาราง ส่วนลด ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Discount_code เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Discount แสดงผลของตารางที่ได้ผ่านการ Query

```
Query 1
Limit to 50000 rows
    Discount_type VARCHAR(50) NOT NULL,
    Discount_value DECIMAL(8, 2) NOT NULL,
    Usage_limit INT NOT NULL,
    Start_day TINYINT NOT NULL,
    Start_month TINYINT NOT NULL,
    Start_year SMALLINT NOT NULL,
    End_day TINYINT NOT NULL,
    End_month TINYINT NOT NULL,
    End_year SMALLINT NOT NULL
);
INSERT INTO Discount VALUES
('DC0001', 'Buy 500 Baht', 'Shop Discount', 50.00, 1000, 7, 3, 2568, 14, 3, 2568),
('DC0002', 'Buy 150 Baht', 'Store Discount', 100.00, 500, 7, 6, 2568, 14, 6, 2568),
('DC0003', 'Buy 100 Baht', 'Shop Discount', 10.00, 550, 7, 7, 2568, 14, 7, 2568),
('DC0004', 'Buy 200 Baht', 'Store Discount', 20.00, 1000, 8, 1, 2568, 15, 1, 2568),
('DC0005', 'Free Shipping', 'Free Shipping', 30.00, 555, 8, 1, 2568, 15, 1, 2568);
SELECT * FROM Discount;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Discount_code	Discount_name	Discount_type	Discount_value	Usage_limit	Start_day	Start_month	Start_year	End_day	End_month	End_year
DC0001	Buy 500 Baht	Shop Discount	50.00	1000	7	3	2568	14	3	2568
DC0002	Buy 150 Baht	Store Discount	100.00	500	7	6	2568	14	6	2568
DC0003	Buy 100 Baht	Shop Discount	10.00	550	7	7	2568	14	7	2568
DC0004	Buy 200 Baht	Store Discount	20.00	1000	8	1	2568	15	1	2568
DC0005	Free Shipping	Free Shipping	30.00	555	8	1	2568	15	1	2568
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางส่วนลด

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_discount_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่เริ่มใช้ส่วนลด (Start_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่เริ่มใช้ส่วนลด (Start_day) อยู่ในช่วง 1 ถึง 31หรือไม่

หากวันที่เริ่มใช้ส่วนลด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่เริ่มใช้ส่วนลด (Start_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเดือนที่เริ่มใช้ส่วนลด (Start_month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเดือนที่เริ่มใช้ส่วนลด (Start_month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่า วันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่หมดอายุของส่วนลด(End_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าที่วันที่หมดอายุของส่วนลด (End_day) อยู่ในช่วง 1 ถึง 31

หากวันสุดท้ายที่สามารถใช้ส่วนลด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่หมดอายุของส่วนลด (End_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเดือนที่หมดอายุของส่วนลด (End_month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเดือนที่หมดอายุของส่วนลด (End_month เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าปีหมดอายุของส่วนลด (End_year) จะต้องมากกว่าหรือเท่ากับปีที่เริ่มใช้ส่วนลด (Start_year)
- หรือถ้าปีหมดอายุของส่วนลด (End_year) เท่ากับปีที่เริ่มใช้ส่วนลด (Start_year) เดือนที่หมดอายุของส่วนลด (End_month) จะต้องมากกว่าหรือเท่ากับเดือนที่เริ่มใช้ส่วนลด (Start_month)
- หรือถ้าปีหมดอายุของส่วนลด (End_year) เท่ากับปีที่เริ่มใช้ส่วนลด (Start_year) และ เดือนที่หมดอายุของส่วนลด (End_month) เท่ากับเดือนที่เริ่มใช้ส่วนลด (Start_month) วันที่หมดอายุของส่วนลด (End_day) จะต้องมากกว่าหรือเท่ากับวันที่เริ่มใช้ส่วนลด (Start_day)

```
1   ALTER TABLE discount
2   ADD CONSTRAINT chk_discount_date
3   CHECK (
4       Start_month BETWEEN 1 AND 12 AND
5       Start_day BETWEEN 1 AND 31 AND
6       (
7           (Start_month IN (1, 3, 5, 7, 8, 10, 12)) OR
8           (Start_month IN (4, 6, 9, 11) AND Start_day <= 30) OR
9           (Start_month = 2 AND Start_day <= 29)
10      ) AND
11      End_month BETWEEN 1 AND 12 AND
12      End_day BETWEEN 1 AND 31 AND
13      (
14          (End_month IN (1, 3, 5, 7, 8, 10, 12)) OR
15          (End_month IN (4, 6, 9, 11) AND End_day <= 30) OR
16          (End_month = 2 AND End_day <= 29)
17      )AND
18      End_year >= Start_year OR
19      End_year = Start_year AND End_month >= Start_month OR
20      End_year = Start_year AND End_month = Start_month AND  End_day > Start_day
21  );
```

ทดสอบการเพิ่มค่า โดย วันเริ่มใช้ส่วนลดคือวันที่ 8 เดือน 1 ปี 2568 วันหมดอายุของส่วนลด คือ วันที่ 15 เดือน 1 ปี 2565 ในเงื่อนไขกำหนดไว้ว่า ปีที่หมดอายุของส่วนลด จะต้องมากกว่าหรือเท่ากับปีที่เริ่มใช้ส่วนลด จึงทำให้ไม่สามารถเพิ่มค่าได้

```
24 •  INSERT INTO discount VALUES ('DC0006', 'Free Shipping', 'Free Shipping', 30.00, 555, 8, 1, 2568, 15, 1, 2565);
25
26
```

Output

Action Output

#	Time	Action	Message
1	18:46:55	ALTER TABLE discount ADD CONSTRAINT chk_discount_date CHECK (Start_month BETWEEN 1 AND 12 ...	Error Code: 3822. Duplicate check constraint name 'chk_discount_date'.

การสร้างตาราง **ผู้ขาย** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตาราง ผู้ขาย ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Shop_name เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_card เป็น FK

และอ้างอิงจากตาราง id_card

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Shop แสดงผลของตารางที่ได้ผ่านการ Query

Query 1

```
CREATE TABLE Sale (
    Shop_name VARCHAR(50) PRIMARY KEY NOT NULL,
    ID_card CHAR(13) UNIQUE NOT NULL,
    Location VARCHAR(100) NOT NULL,
    Phone_number VARCHAR(12) NOT NULL,
    Email VARCHAR(40) NOT NULL,
    FOREIGN KEY (ID_card) REFERENCES id_card(ID_card)
);
INSERT INTO Sale VALUES
('ShopEase', '1749400578123', '13.7465°N,100.5340°E', '8-5514-3659', 'Shopease@gmail.com'),
('TrendzyMall', '1764700155982', '13.7460°N,100.5296°E', '8-7567-5312', 'Trendzymall@gmail.com'),
('QuickBuy', '1763001789541', '13.6687°N,100.6042°E', '9-8012-0893', 'Quickbuy@gmail.com'),
('EzyMart', '1742400055884', '13.7370°N,100.5460°E', '6-6413-9278', 'Ezymart@gmail.com'),
('BestPick', '1788900050147', '13.7292°N,100.7750°E', '6-6019-8723', 'Bestpick@gmail.com');
SELECT * FROM Sale;
```

Result Grid

Shop_name	ID_card	Location	Phone_number	Email
BestPick	1788900050147	13.7292°N,100.7750°E	6-6019-8723	Bestpick@gmail.com
EzyMart	1742400055884	13.7370°N,100.5460°E	6-6413-9278	Ezymart@gmail.com
QuickBuy	1763001789541	13.6687°N,100.6042°E	9-8012-0893	Quickbuy@gmail.com
ShopEase	1749400578123	13.7465°N,100.5340°E	8-5514-3659	Shopease@gmail.com
TrendzyMall	1764700155982	13.7460°N,100.5296°E	8-7567-5312	Trendzymall@gmail.com
NULL	NULL	NULL	NULL	NULL

การสร้างตาราง **สินค้า** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางสินค้าขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Product_code เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Shop_name เป็น FK และอ้างอิงจากตาราง sale

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Product แสดงผลของตารางที่ได้ผ่านการ Query

ONLINE _shopping_system*

```
CREATE TABLE Product (
    Product_code CHAR(6) PRIMARY KEY NOT NULL,
    Shop_name VARCHAR(50) NOT NULL,
    Product_name VARCHAR(50) NOT NULL,
    Product_type VARCHAR(50) NOT NULL,
    Price DECIMAL(8,2) NOT NULL,
    Import_day TINYINT NOT NULL,
    Import_month TINYINT NOT NULL,
    Import_year SMALLINT NOT NULL,
    Expire_day TINYINT NOT NULL,
    Expire_month TINYINT NOT NULL,
    Expire_year SMALLINT NOT NULL,
   FOREIGN KEY (Shop_name) REFERENCES sale(Shop_name)
);
INSERT INTO Product VALUES
('SA0001', 'ShopEase', 'D-D', 'Beverage', 10.00, 15, 2, 2565, 16, 5, 2568),
('SA0002', 'TrendzyMall', 'Kato', 'Beverage', 12.00, 7, 1, 2567, 8, 5, 2570),
('SA0003', 'QuickBuy', 'Lay', 'Snack', 20.00, 2, 6, 2567, 3, 6, 2570),
('SA0004', 'EzyMart', 'Tao Kae Noi', 'Snack', 10.00, 21, 10, 2567, 22, 10, 2570),
('SA0005', 'BestPick', 'Kozo', 'Snack', 25.00, 27, 9, 2568, 28, 9, 2571);
SELECT * FROM Product
```

Result Grid

Product_code	Shop_name	Product_name	Product_type	Price	Import_day	Import_month	Import_year	Expire_day	Expire_month	Expire_year
SA0001	ShopEase	D-D	Beverage	10.00	15	2	2565	16	5	2568
SA0002	TrendzyMall	Kato	Beverage	12.00	7	1	2567	8	5	2570
SA0003	QuickBuy	Lay	Snack	20.00	2	6	2567	3	6	2570
SA0004	EzyMart	Tao Kae Noi	Snack	10.00	21	10	2567	22	10	2570
SA0005	BestPick	Kozo	Snack	25.00	27	9	2568	28	9	2571

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางสินค้า

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_product_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่เริ่มทำการนำเข้าสินค้า (import_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่เริ่มทำการนำเข้าสินค้า (Import_ day) อยู่ในช่วง 1 ถึง 31หรือไม่

หากวันที่เริ่มใช้ส่วนลด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่เริ่มทำการนำเข้าสินค้า (import_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเดือนที่เริ่มทำการนำเข้าสินค้า (import_month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเดือนที่เริ่มทำการนำเข้าสินค้า (import_month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่สินค้าหมดอายุ (Expire_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าที่วันที่สินค้าหมดอายุ (Expire _day) อยู่ในช่วง 1 ถึง 31หากวันสุดท้ายที่สามารถใช้ส่วนลด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่สินค้าหมดอายุ (Expire_Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเดือนที่สินค้าหมดอายุ (Expire_Month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเดือนที่สินค้าหมดอายุ (Expire_Month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าปีที่สินค้าหมดอายุ (Expire_year) จะต้องมากกว่าหรือเท่ากับปีที่เริ่มทำการนำเข้าสินค้า (import_yaer)
- หรือถ้าปีที่สินค้าหมดอายุ (Expire_year) เท่ากับปีที่เริ่มทำการนำเข้าสินค้า (import_yaer) และ เดือนที่สินค้าหมดอายุ (Expire_Month) มากกว่า เดือนที่เริ่มทำการนำเข้าสินค้า (import_month)
- หรือถ้าปีที่สินค้าหมดอายุ (Expire_year) ต้องเท่ากับปีที่เริ่มทำการนำเข้าสินค้า (import_yaer) และ เดือนที่สินค้าหมดอายุ (Expire_Month) ต้องเท่ากับเดือนที่เริ่มทำการนำเข้าสินค้า (import_month) และ วันที่สินค้าหมดอายุ (Expire _day) ต้องเท่ากับวันที่เริ่มทำการนำเข้าสินค้า (Import_ day)

```sql
ALTER TABLE product
ADD CONSTRAINT chk_product_date
CHECK (
    Import_month BETWEEN 1 AND 12 AND
    Import_day BETWEEN 1 AND 31 AND
    (
        (Import_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (Import_month IN (4, 6, 9, 11) AND Import_day <= 30) OR
        (Import_month = 2 AND Import_day <= 29)
    ) AND
    Expire_month BETWEEN 1 AND 12 AND
    Expire_day BETWEEN 1 AND 31 AND
    (
        (Expire_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (Expire_month IN (4, 6, 9, 11) AND Expire_day <= 30) OR
        (Expire_month = 2 AND Expire_day <= 29)
    )AND
    Expire_year >= Import_year OR
    Expire_year = Import_year AND Expire_month > Import_month OR
    Expire_year = Import_year AND Expire_month = Import_month AND Expire_day > Import_day
);
```

ทดสอบการเพิ่มค่า โดย วันนำเข้าสินค้าวันที่ 27 เดือน 9 ปี 2568 วันหมดอายุสินค้า คือ วันที่ 27 เดือน 9 ปี 2568 ในเงื่อนไขกำหนดไว้ว่า ถ้า ปี และ เดือนของการนำเข้าสินค้าและวันหมดอายุสินค้าเท่ากัน วันหมดอายุสินค้าต้องมากกว่าวันที่นำเข้าสินค้า

```
24 •  INSERT INTO product VALUES ('SA0006', 'BestPick', 'Kozo', 'Snack', 25.00, 27, 9, 2568, 27, 9, 2568);
25
```

Output

Action Output

#	Time	Action	Message
1	18:37:24	INSERT INTO product VALUES ('SA0006', 'BestPick', 'Kozo', 'Snack', 25.00, 27, 9, 2568, 27, 9, 2568)	Error Code: 1062. Duplicate entry 'SA0006' for key 'product.PRIMARY'

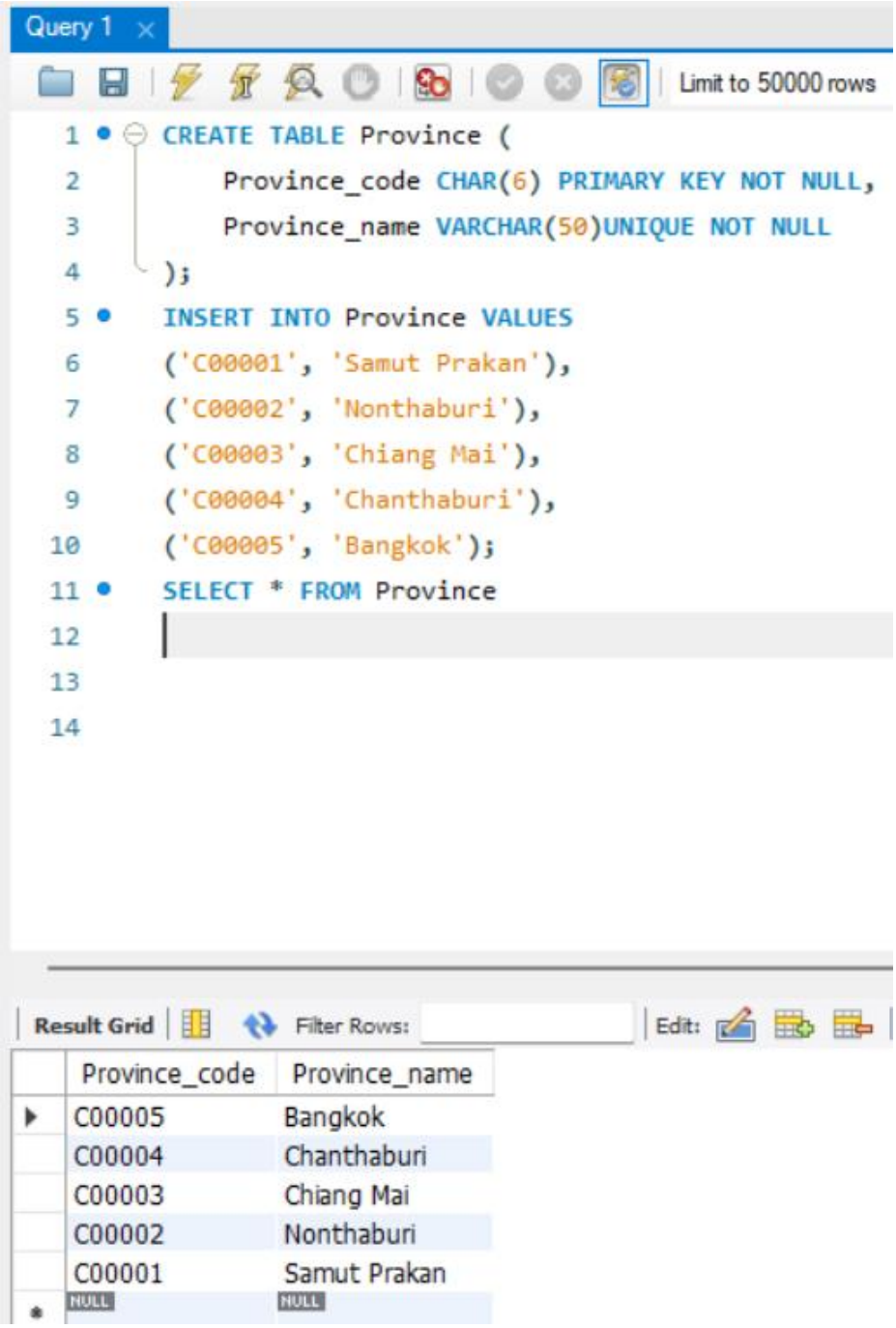
การสร้างตาราง **จังหวัด** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางจังหวัดขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Province _code เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Province แสดงผลของตารางที่ได้ผ่านการ Query



การสร้างตาราง **อำเภอ/เขต** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางอำเภอ/เขตขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ District_code เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Province_code เป็น FK และอ้างอิงจากตาราง Province

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM District แสดงผลของตารางที่ได้ผ่านการ Query

Query 1

Limit to 50000 rows

```
CREATE TABLE District (
    District_code CHAR(6) PRIMARY KEY NOT NULL,
    Province_code CHAR(6) UNIQUE NOT NULL,
    District_name VARCHAR(50) NOT NULL,
    FOREIGN KEY (Province_code) REFERENCES Province (Province_code)
);
INSERT INTO District VALUES
('B00001', 'C00001', 'Bang Phli'),
('B00002', 'C00002', 'Pak Kret'),
('B00003', 'C00003', 'Chiang Mai'),
('B00004', 'C00004', 'Tha Mai'),
('B00005', 'C00005', 'Bang Khen');
SELECT * FROM District
```

Result Grid | Filter Rows: | Edit: | Export/Import:

District_code	Province_code	District_name
B00001	C00001	Bang Phli
B00002	C00002	Pak Kret
B00003	C00003	Chiang Mai
B00004	C00004	Tha Mai
B00005	C00005	Bang Khen
NULL	NULL	NULL

การสร้างตาราง **ตำบล/แขวง** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางตำบล/แขวงขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Subdistrict _code เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ District_code เป็น FK และอ้างอิงจากตาราง District

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Subdistrict แสดงผลของตารางที่ได้ผ่านการ Query

Query 1

Limit to 50000 rows

```
CREATE TABLE Subdistrict (
    Subdistrict_code CHAR(6) PRIMARY KEY NOT NULL,
    District_code CHAR(6) UNIQUE NOT NULL,
    Subdistrict_name VARCHAR(50) NOT NULL,
    Postal_code CHAR(5) NOT NULL,
    FOREIGN KEY (District_code) REFERENCES District (District_code)
);
INSERT INTO Subdistrict VALUES
('A00001', 'B00001', 'Bang Kaeo', '10540'),
('A00002', 'B00002', 'Pak Kret', '11120'),
('A00003', 'B00003', 'Chiang Mai', '50000'),
('A00004', 'B00004', 'Tha Mai', '22120'),
('A00005', 'B00005', 'Tha Raeng', '10220');
SELECT * FROM Subdistrict
```

Result Grid | Filter Rows: | Edit: | Export/Import:

Subdistrict_code	District_code	Subdistrict_name	Postal_code
A00001	B00001	Bang Kaeo	10540
A00002	B00002	Pak Kret	11120
A00003	B00003	Chiang Mai	50000
A00004	B00004	Tha Mai	22120
A00005	B00005	Tha Raeng	10220
NULL	NULL	NULL	NULL

การสร้างตาราง **ที่อยู่จัดส่ง** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางที่อยู่จัดส่งขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Shipping_address_ID เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Subdistrict_code เป็น FK และอ้างอิงจากตาราง Subdistrict

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Shipping_address แสดงผลของตารางที่ได้ผ่านการ Query

Query 1

Limit to 50000 rows

```
CREATE TABLE Shipping_address (
    Shipping_address_ID VARCHAR(10) PRIMARY KEY NOT NULL,
     Subdistrict_code CHAR(6) UNIQUE NOT NULL,
    House_number VARCHAR(20) NOT NULL,
    swine VARCHAR(10) ,
    alley VARCHAR(20) ,
    FOREIGN KEY (Subdistrict_code) REFERENCES Subdistrict (Subdistrict_code)
);
INSERT INTO Shipping_address  VALUES
('AD0001', 'A00001', '20/1', '4', '2'),
('AD0002', 'A00002', '10/29', '7', '1'),
('AD0003', '00003', '134/24', '9', '5'),
('AD0004', 'A00004', '9/10', '1', '4'),
('AD0005', 'A00005', '205/3', '3', '13');
SELECT * FROM Shipping_address
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Co

Shipping_address_ID	Subdistrict_code	House_number	swine	alley
AD0001	A00001	20/1	4	2
AD0002	A00002	10/29	7	1
AD0003	A00003	134/24	9	5
AD0004	A00004	9/10	1	4
AD0005	A00005	205/3	3	13
NULL	NULL	NULL	NULL	NULL

การสร้างตาราง **การสั่งซื้อสินค้า** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการสั่งซื้อสินค้าขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Order_ID เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Order_ID เป็น FK และอ้างอิงจากตาราง Orders

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Product_code เป็น FK และอ้างอิงจากตาราง Product

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Discount_code เป็น FK และอ้างอิงจากตาราง Discount

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Orders แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Orders (
Order_ID CHAR(6) PRIMARY KEY NOT NULL,
ID_Customer CHAR(6) NOT NULL,
Product_code VARCHAR(6) NOT NULL,
Discount_code VARCHAR(6) NOT NULL,
Quantity INT NOT NULL,
Total_amount DECIMAL(8, 2) NOT NULL,
Shipping_type VARCHAR(50) NOT NULL,
Shipping_price DECIMAL(8, 2) NOT NULL,
Net_amount DECIMAL(8, 2) NOT NULL,
Order_day TINYINT NOT NULL,
Order_month TINYINT NOT NULL,
Order_year SMALLINT NOT NULL,
Order_time TIME NOT NULL,
status_order VARCHAR(20) NOT NULL,
FOREIGN KEY (ID_Customer) REFERENCES Customers (ID_Customer),
FOREIGN KEY (Product_code) REFERENCES Product(Product_code),
FOREIGN KEY (Discount_code) REFERENCES Discount (Discount_code)
) ;
INSERT INTO Orders VALUES
('SM0001', 'MB2548', 'SA0001', 'DC0001', 30, 250.00, 'Express Delivery', 50.00, 250.00, 11, 1, 2568, '7:12:00',สำเร็จ),
('SM0002', 'MB2549', 'SA0002', 'DC0002', 15, 120.00, 'Registered Parcel Delivery', 40.00, 120.00, 14, 1, 2568, '6:00:00',สำเร็จ),
('SM0003', 'MB2550', 'SA0003', 'DC0003', 10, 240.00, 'Express Delivery', 50.00, 240.00, 24, 1, 2568, '14:09:36',สำเร็จ),
('SM0004', 'MB2551', 'SA0004', 'DC0004', 23, 290.00, 'International Delivery', 80.00, 290.00, 9, 2, 2568, '11:16:48',สำเร็จ),
('SM0005', 'MB2552', 'SA0005', 'DC0005', 10, 260.00, 'Registered Parcel Delivery', 40.00, 260.00, 14, 2, 2568, '11:31:12',สำเร็จ),
('SM0006', 'MB2551', 'SA0005','DC0004', 10, 260.00, 'Registered Parcel Delivery', 40.00, 210.00, 14, 2, 2568, '12:31:12',ไม่สำเร็จ);
SELECT * FROM Orders;
```

Order_ID	ID_Customer	Product_code	Discount_code	Quantity	Total_amount	Shipping_type	Shipping_price	Net_amount	Order_day	Order_month	Order_year	Order_time	status_order
SM0001	MB2548	SA0001	DC0001	30	250.00	Express Delivery	50.00	250.00	11	1	2568	07:12:00	สำเร็จ
SM0002	MB2549	SA0002	DC0002	15	120.00	Registered Parcel Delivery	40.00	120.00	14	1	2568	06:00:00	สำเร็จ
SM0003	MB2550	SA0003	DC0003	10	240.00	Express Delivery	50.00	240.00	24	1	2568	14:09:36	สำเร็จ
SM0004	MB2551	SA0004	DC0004	23	290.00	International Delivery	80.00	290.00	9	2	2568	11:16:48	สำเร็จ
SM0005	MB2552	SA0005	DC0005	10	260.00	Registered Parcel Delivery	40.00	260.00	14	2	2568	11:31:12	สำเร็จ
SM0006	MB2551	SA0005	DC0004	10	260.00	Registered Parcel Delivery	40.00	210.00	14	2	2568	12:31:12	ไม่สำเร็จ

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการสั่งซื้อ

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_orders_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่สั่งซื้อ (Order_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่สั่งซื้อ (Order_day) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่สั่งซื้อ (Order_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่สั่งซื้อ (Order_month) ไม่เกิน 29 วัน

- กำหนดให้ปีที่สั่งซื้อ ต้องมากกว่าหรือเท่ากับปี 2568

```
ALTER TABLE orders
ADD CONSTRAINT chk_order_date
CHECK (
    Order_month BETWEEN 1 AND 12 AND
    Order_day BETWEEN 1 AND 31 AND
    (
        (Order_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (Order_month IN (4, 6, 9, 11) AND Order_day <= 30) OR
        (Order_month = 2 AND Order_day <= 29)
    ) AND
    Order_year >= 2568
);
```

ทดสอบการเพิ่มค่า โดย วันที่ 31 เดือน 6 ปี 2568 ในเงื่อนไขกำหนดไว้ว่า ถ้าเดือนที่คืนสินค้าคือเดือนที่ 6 วันที่คืนสินค้าจะสามารถเพิ่มค่าได้ไม่เกิน 30 วัน ทำให้ไม่สามารถเพิ่มข้อมูลนี้ได้

```
INSERT INTO orders VALUES ('SM0007', 'MB2559', 'SA0005','DC0004', 10, 260.00, 'Registered Parcel Delivery', 40.00, 210.00, 31, 6, 2568, '12:31:12','ไม่สำเร็จ');
```

Output

Action Output

#	Time	Action	Message
1	19:10:57	INSERT INTO orders VALUES ('SM0007', 'MB2559', 'SA0005','DC0004', 10, 260.00, 'Registered Parcel Deliver...	Error Code: 3819. Check constraint 'chk_order_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการสั่งซื้อ

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Shipping_type และ chk_status_order

CHECK ประเภทการจัดส่ง(Shipping_type) กำหนดให้มี 3 ประเภท คือ การจัดส่งด่วน(Express Delivery), การจัดส่งแบบพัสดุลงทะเบียน(Registered Parcel Delivery), การจัดส่งระหว่างประเทศ (International Delivery) และ สถานการณ์สั่งซื้อ(status_order) 2 สถานะ คือ สำเร็จ และ ไม่สำเร็จ

```
ALTER TABLE orders
ADD CONSTRAINT chk_Shipping_type
CHECK (Shipping_type IN ('Express Delivery', 'Registered Parcel Delivery', 'International Delivery')),
ADD CONSTRAINT chk_status_order
CHECK (status_order IN ('สำเร็จ', 'ไม่สำเร็จ'));
```

ทดลองเพิ่มค่า ใส่ค่าตรงประเภทการจัดส่งแบบ Express ซึ่งไม่ตรงกับเงื่อนไขที่กำหนดไว้ ทำให้ไม่สามารถเพิ่มค่าได้

```
19 •  INSERT INTO Orders VALUES ('SM0007', 'MB2550', 'SA0001', 'DC0001', 30, 250.00, 'Express', 50.00, 250.00, 11, 1, 2568, '7:12:00',ส่งสิ้น);
20
```

Output

Action Output

#	Time	Action	Message
1	21:20:30	INSERT INTO Orders VALUES ('SM0007', 'MB2550', 'SA0001', 'DC0001', 30, 250.00, 'Express', 50.00, 250.00, ...	Error Code: 3819. Check constraint 'chk_Shipping_type' is violated.

ทดลองเพิ่มค่า ใส่ค่าตรงสถานะการสั่งซื้อเป็น ยกเลิก ซึ่งไม่ตรงกับเงื่อนไขที่กำหนดไว้ ทำให้ไม่สามารถเพิ่มค่าได้

```
22 •  INSERT INTO Orders VALUES ('SM0007', 'MB2550', 'SA0001', 'DC0001', 30, 250.00, 'Express Delivery', 50.00, 250.00, 11, 1, 2568, '7:12:00',ยกเลิก);
23
```

Output

Action Output

#	Time	Action	Message
1	22:19:37	INSERT INTO Orders VALUES ('SM0007', 'MB2550', 'SA0001', 'DC0001', 30, 250.00, 'Express Delivery', 50.00,...	Error Code: 3819. Check constraint 'chk_status_order' is violated.

การสร้างตาราง **การชำระเงิน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการชำระเงินขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Payment เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Order_ID เป็น FK และอ้างอิงจากตาราง Orders

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Payment แสดงผลของตารางที่ได้ผ่านการ Query

ONLINE _shopping_system*

Limit to 50000 rows

```
CREATE TABLE Payment (
    ID_Payment CHAR(6) PRIMARY KEY NOT NULL,
    Order_ID CHAR(6) UNIQUE NOT NULL,
    Payment_Date TINYINT,
    Payment_Month TINYINT,
    Payment_Year SMALLINT,
    Payment_Time TIME,
    Payment_Status VARCHAR(10),
   FOREIGN KEY (Order_ID) REFERENCES     Orders (Order_ID)
);
INSERT INTO Payment VALUES
('PM0001',    'SM0001',    11,    1,    2568,    '7:12:00',    'สำเร็จ'),
('PM0002',    'SM0002',    14,    1,    2568,    '12:00:00',    'สำเร็จ'),
('PM0003',    'SM0003',    24,    1,    2568,    '6:00:00',    'สำเร็จ'),
('PM0004',    'SM0004',    9,    2,    2568,    '8:38:24', 'สำเร็จ'),
('PM0005',    'SM0005',    14,    2,    2568,    '2:09:36',    'สำเร็จ'),
('PM0006',    'SM0006',    NULL,NULL,    NULL,    NULL,    'ไม่สำเร็จ');
 SELECT * FROM Payment
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ID_Payment	Order_ID	Payment_Date	Payment_Month	Payment_Year	Payment_Time	Payment_Status
PM0001	SM0001	11	1	2568	07:12:00	สำเร็จ
PM0002	SM0002	14	1	2568	12:00:00	สำเร็จ
PM0003	SM0003	24	1	2568	06:00:00	สำเร็จ
PM0004	SM0004	9	2	2568	08:38:24	สำเร็จ
PM0005	SM0005	14	2	2568	02:09:36	สำเร็จ
PM0006	SM0006	NULL	NULL	NULL	NULL	ไม่สำเร็จ

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_payment_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ชำระเงิน (Payment _month) อยู่ในช่วง 1 ถึง 12

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ชำระเงิน (Payment _date) อยู่ในช่วง 1 ถึง 31 หรือไม่

  หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

  1. ตรวจสอบว่าเป็นเดือนที่ชำระเงิน (Payment _month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
  2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
  3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ชำระเงิน (Payment _date) ไม่เกิน 29 วัน

```
ALTER TABLE payment
ADD CONSTRAINT chk_payment_date
CHECK (
    Payment_month BETWEEN 1 AND 12 AND
    Payment_date BETWEEN 1 AND 31 AND
    (
        (Payment_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (Payment_month IN (4, 6, 9, 11) AND Payment_date <= 30) OR
        (Payment_month = 2 AND Payment_date <= 29)
    ) AND
    Payment_year >= 2568
);
```

ทดสอบการเพิ่มค่า โดย วันเริ่มทำการชำระเงินคือวันที่ 14 เดือน 2 ปี 2565 ในเงื่อนไขกำหนดไว้ว่า ปีที่ทำการชำระเงิน จะต้องมากกว่าหรือเท่ากับปี 2568 จึงทำให้ไม่สามารถเพิ่มค่าได้

```
INSERT INTO payment VALUES ('PM0007',    'SM0005',    14,    2,    2565,    '2:09:36',    'สำเร็จ');
```

Output

Action Output

#	Time	Action	Message
1	18:59:10	INSERT INTO payment VALUES ('PM0007', 'SM0005', 14, 2, 2565, '2:09:36', 'สำเร็จ')	Error Code: 3819. Check constraint 'chk_payment_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_payment_status

CHECK สถานะการชำระเงิน(Payment_Status) กำหนด 2 ค่า คือ สำเร็จ และ ไม่สำเร็จ

```
23 •  ALTER TABLE payment
24    ADD CONSTRAINT chk_payment_status
25    CHECK (Payment_Status IN ('สำเร็จ', 'ไม่สำเร็จ'));
```

ทดลองเพิ่มค่า โดยใส่ค่าของสถานการณ์ชำระเงินเป็น ยกเลิก ซึ่งไม่ตรงกับเงื่อนไขที่กำหนดทำให้ไม่สามารถเพิ่มค่าได้

```
27 •  INSERT INTO Payment VALUES ('PM0007', 'SM0001', 11, 1, 2568,'7:12:00','ยกเลิก');
28
```

Output

Action Output

#	Time	Action	Message
1	21:22:12	INSERT INTO Payment VALUES ('PM0007', 'SM0001', 11, 1, 2568,'7:12:00','ยกเลิก')	Error Code: 3819. Check constraint 'chk_payment_status' is violated.

การสร้างตาราง **การจัดส่ง** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการจัดส่งขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Shipping เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Payment เป็น FK และอ้างอิงจากตาราง Payment

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Shipping_address_ID เป็น FK และอ้างอิงจากตาราง Shipping_address

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Shipping แสดงผลของตารางที่ได้ผ่านการ Query

ONLINE _shopping_system*

```
CREATE TABLE Shipping (
    ID_Shipping VARCHAR(20) PRIMARY KEY,
    ID_Payment CHAR(6) UNIQUE NOT NULL,
    Shipping_address_ID VARCHAR(6) NOT NULL,
    Shipping_Date TINYINT ,
    Shipping_Month TINYINT ,
    Shipping_Year SMALLINT ,
    Shipping_Time TIME ,
    Company VARCHAR(20) ,
    Shipping_Status VARCHAR(20) NOT NULL ,
    FOREIGN KEY ( ID_Payment) REFERENCES     Payment ( ID_Payment),
    FOREIGN KEY (Shipping_address_ID) REFERENCES Shipping_address (Shipping_address_ID)
);
INSERT INTO Shipping  VALUES
('1Z12345E1512345676', 'PM0001', 'AD0001', 12, 1, 2568, '0:00:00', 'Kerry Express', 'Delivered'),
('EA123456789TH', 'PM0002', 'AD0002', 15, 1, 2568, '7:12:00', 'Thai Post', 'In Transit'),
('JT123456789TH', 'PM0003', 'AD0003', NULL, NULL, NULL, NULL, NULL, 'Failed'),
('NV123456789', 'PM0004', 'AD0004', 10, 2, 2568, '7:12:00', 'Ninja Van', 'Delivered'),
('JV123456789TH', 'PM0005', 'AD0005', 15, 2, 2568, '7:12:00', 'J&T Express', 'Delivered');
SELECT * FROM Shipping;
```

Result Grid

ID_Shipping	ID_Payment	Shipping_address_ID	Shipping_Date	Shipping_Month	Shipping_Year	Shipping_Time	Company	Shipping_Status
1Z12345E1512345676	PM0001	AD0001	12	1	2568	00:00:00	Kerry Express	Delivered
EA123456789TH	PM0002	AD0002	15	1	2568	07:12:00	Thai Post	In Transit
JT123456789TH	PM0003	AD0003	NULL	NULL	NULL	NULL	NULL	Failed
JV123456789TH	PM0005	AD0005	15	2	2568	07:12:00	J&T Express	Delivered
NV123456789	PM0004	AD0004	10	2	2568	07:12:00	Ninja Van	Delivered

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจัดส่ง

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_shipping_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่จัดส่ง (Shipping_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่จัดส่ง (Shipping_Date) อยู่ในช่วง 1 ถึง 31

หากวันที่จัดส่ง ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่จัดส่ง (Shipping_Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเดือนที่จัดส่ง (Shipping_Month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเดือนที่จัดส่ง (Shipping_Month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าปีที่จัดส่ง (Shipping_year) จะต้องมากกว่าหรือเท่ากับปีปัจจุบัน คือ ปี 2568

```
1   ALTER TABLE shipping
2   ADD CONSTRAINT chk_shipping_date
3   CHECK (
4       Shipping_Month BETWEEN 1 AND 12 AND
5       Shipping_Date BETWEEN 1 AND 31 AND
6       (
7           (Shipping_Month IN (1, 3, 5, 7, 8, 10, 12)) OR
8           (Shipping_Month IN (4, 6, 9, 11) AND Shipping_Date <= 30) OR
9           (Shipping_Month = 2 AND Shipping_Date <= 29)
10      ) AND
11      Shipping_Year >= 2568
12  );
```

ทดสอบการเพิ่มค่า โดย วันที่จัดส่งคือวันที่ 30 เดือน 2 ปี 2568 ในเงื่อนไขกำหนดไว้ว่า ถ้าเดือนที่จัดส่งคือ เดือนที่ 2 วันที่จัดส่งจะสามารถเพิ่มค่าได้ไม่เกิน 29 วัน ทำให้ไม่สามารถเพิ่มข้อมูลนี้ได้

```
16 •  INSERT INTO shipping VALUES ('TV123456789TH', 'PM0006', 'AD0006', 30, 2, 2568, '7:12:00', 'J&T Express', 'Delivered');
17
```

Output

Action Output

#	Time	Action	Message
1	19:03:09	INSERT INTO shipping VALUES ('TV123456789TH', 'PM0006', 'AD0006', 30, 2, 2568, '7:12:00', 'J&T Express',...	Error Code: 3819. Check constraint 'chk_shipping_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจัดส่ง

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Shipping_Status

CHECK สถานะการจัดส่ง(Shipping_Status) มี 3 สถานะ คือ กำลังจัดส่ง(In Transit), จัดส่งสำเร็จ(Delivered), จัดส่งไม่สำเร็จ(Failed)

```
29 •  ALTER TABLE Shipping
30    ADD CONSTRAINT chk_Shipping_Status
31    CHECK (Shipping_Status IN ('In Transit', 'Delivered', 'Failed'));
```

ทดลองเพิ่มค่า โดยใส่ค่าสถานะการจัดส่งมีสถานะ ว่า ยกเลิกแล้ว ซึ่งไม่ตรงกับเงื่อนไขที่กำหนด ทำให้ไม่สามารถเพิ่มค่าได้

```
33 •  INSERT INTO Shipping  VALUES ('1ZTH345E1512345676', 'PM0007', 'AD0001', 12, 3, 2568, '0:00:00', 'Kerry Express', 'ยกเลิกแล้ว');
34
```

Output

Action Output

#	Time	Action	Message
1	21:30:43	INSERT INTO Shipping VALUES ('1ZTH345E1512345676', 'PM0007', 'AD0001', 12, 3, 2568, '0:00:00', 'Kerry ...	Error Code: 3819. Check constraint 'chk_Shipping_Status' is violated.

การสร้างตาราง **การรีวิวสินค้า** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการรีวิวสินค้าขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Review_ID เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Shipping เป็น FK และอ้างอิงจากตาราง Shipping

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Review แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Review (
    Review_ID VARCHAR(10) PRIMARY KEY NOT NULL,
    ID_Shipping VARCHAR(50) UNIQUE NOT NULL,
    Image_path VARCHAR(255) ,
    Review_details VARCHAR(255) ,
    Product_quality TINYINT,
    Seller_service TINYINT,
    Delivery_service TINYINT,
    FOREIGN KEY (ID_Shipping) REFERENCES Shipping (ID_Shipping)
);
INSERT INTO Review  VALUES
('RW0001', '1Z12345E1512345676', NULL, NULL, NULL, 85, 80),
('RW0002', 'EA123456789TH', 'C:\\download guide-for-food-photography-wedio.jpg', NULL, NULL, 80, 85),
('RW0003', 'JT123456789TH', NULL, NULL, 50, 60, 65),
('RW0004', 'NV123456789', NULL, 'Good seller response', NULL, NULL, NULL),
('RW0005', 'JV123456789TH', 'C:\\Users\\User\\Pictures\\download\\OIP.jpg', 'Delicious snack', 95, 95, 95);
SELECT * FROM Review;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Review_ID	ID_Shipping	Image_path	Review_details	Product_quality	Seller_service	Delivery_service
RW0001	1Z12345E1512345676	NULL	NULL	NULL	85	80
RW0002	EA123456789TH	C:\download guide-for-food-photography-...	NULL	NULL	80	85
RW0003	JT123456789TH	NULL	NULL	50	60	65
RW0004	NV123456789	NULL	Good seller response	NULL	NULL	NULL
RW0005	JV123456789TH	C:\Users\User\Pictures\download\OIP.jpg	Delicious snack	95	95	95
NULL	NULL	NULL	NULL	NULL	NULL	NULL

การสร้างตาราง **การคืนเงิน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการคืนเงินขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Refund เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Shipping เป็น FK และอ้างอิงจากตาราง Shipping

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Refund_Money แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Refund_Money (
2      ID_Refund CHAR(20) PRIMARY KEY NOT NULL,
3      ID_Shipping VARCHAR(20) UNIQUE NOT NULL,
4      Cancel_Date TINYINT NOT NULL,
5      Cancel_Month TINYINT NOT NULL,
6      Cancel_Year SMALLINT NOT NULL,
7      Cancel_Time TIME NOT NULL,
8      Reason_Refund VARCHAR(100) NOT NULL,
9      Ref_Date TINYINT NOT NULL,
10     Ref_Month TINYINT NOT NULL,
11     Ref_Year SMALLINT NOT NULL,
12     Ref_Time TIME NOT NULL,
13     FOREIGN KEY (ID_Shipping) REFERENCES Shipping (ID_Shipping)
14 );
15 INSERT INTO Refund_Money VALUES
16 ('CC0001', '1Z12345E1512345676', 15, 1, 2568, '7:12:00', 'Incomplete product', 15, 1, 2568, '9:36:00'),
17 ('CC0002', 'EA123456789TH', 16, 1, 2568, '8:38:24', 'Product change', 16, 1, 2568, '9:00:00'),
18 ('CC0003', 'JT123456789TH', 17, 1, 2568, '12:28:48', 'Order cancellation', 17, 1, 2568, '8:24:00'),
19 ('CC0004', 'NV123456789', 18, 2, 2568, '10:48:00', 'Incomplete product', 18, 2, 2568, '10:05:00'),
20 ('CC0005', 'JV123456789TH', 19, 2, 2568, '8:38:24', 'Product change', 19, 2, 2568, '8:50:00');
21 SELECT * FROM Refund_Money
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ID_Refund	ID_Shipping	Cancel_Date	Cancel_Month	Cancel_Year	Cancel_Time	Reason_Refund	Ref_Date	Ref_Month	Ref_Year	Ref_Time
CC0001	1Z12345E1512345676	15	1	2568	07:12:00	Incomplete product	15	1	2568	09:36:00
CC0002	EA123456789TH	16	1	2568	08:38:24	Product change	16	1	2568	09:00:00
CC0003	JT123456789TH	17	1	2568	12:28:48	Order cancellation	17	1	2568	08:24:00
CC0004	NV123456789	18	2	2568	10:48:00	Incomplete product	18	2	2568	10:05:00
CC0005	JV123456789TH	19	2	2568	08:38:24	Product change	19	2	2568	08:50:00

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางส่วนลด

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_refund_money_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่ทำการยกเลิกสินค้า (Cancel_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการยกเลิกสินค้า (Cancel_Date) อยู่ในช่วง 1 ถึง 31หรือไม่

หากวันที่ทำการยกเลิกสินค้า ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่ทำการยกเลิกสินค้า (Cancel_Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเดือนที่ทำการยกเลิกสินค้า (Cancel_Month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเดือนที่ทำการยกเลิกสินค้า (Cancel_Month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่ทำการคืนเงิน (Ref_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าที่วันที่ทำการคืนเงิน (Ref_day) อยู่ในช่วง 1 ถึง 31

หากวันที่ทำการคืนเงิน ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่ทำการคืนเงิน (Ref_Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเดือนที่ทำการคืนเงิน (Ref_Month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเดือนที่ทำการคืนเงิน (Ref_Month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าปีที่ทำการคืนเงิน (Ref_Month) จะต้องเท่ากับปีที่ทำการยกเลิกสินค้า (Cancel_Years)
- และ ปีที่ทำการคืนเงิน (Ref_Month) ต้องเท่ากับปีที่ทำการยกเลิกสินค้า (Cancel_Month)
- และ วันที่ทำการคืนเงิน (Ref_Day) ต้องน้อยกว่าหรือเท่ากับวันที่ทำการยกเลิกสินค้า (Cancel_Date) และต้องคืนเงินภายใน 3 วัน

```
1  ALTER TABLE refund_money
2  ADD CONSTRAINT chk_refund_money_date
3  CHECK (
4      Cancel_Month BETWEEN 1 AND 12 AND
5      Cancel_Date BETWEEN 1 AND 31 AND
6      (
7          (Cancel_Month IN (1, 3, 5, 7, 8, 10, 12)) OR
8          (Cancel_Month IN (4, 6, 9, 11) AND Cancel_Date <= 30) OR
9          (Cancel_Month = 2 AND Cancel_Date <= 29)
10     ) AND
11     Ref_Month BETWEEN 1 AND 12 AND
12     Ref_Date BETWEEN 1 AND 31 AND
13     (
14         (Ref_Month IN (1, 3, 5, 7, 8, 10, 12)) OR
15         (Ref_Month IN (4, 6, 9, 11) AND Ref_Date <= 30) OR
16         (Ref_Month = 2 AND Ref_Date <= 29)
17     ) AND
18     Ref_Year = Cancel_Year AND
19     Ref_Month = Cancel_Month AND
20     Ref_Date <= Cancel_Date + 3
21 );
```

ทดสอบการเพิ่มค่า โดยวันที่ทำการยกเลิกสินค้า คือวันที่ 15 เดือน 1 ปี 2568 และวันที่คืนเงิน คือวันที่ 19 เดือน 1 ปี 2568 ในเงื่อนไขกำหนดไว้ว่า ปีที่คืนเงินต้องเท่ากับปีที่ยกเลิกสินค้า และ เดือนที่คืนเงินต้องเท่ากับเดือนที่ยกเลิกสินค้า และวันที่คืนเงิน จะต้องน้อยกว่าหรือเท่ากับ วันที่ยกเลิกสินค้า และจะคืนเงินภายใน 3 วัน ซึ่งในข้อมูลทำการคืนเงินหลังจากวันที่ยกเลิกสินค้า 4 วัน ทำให้ไม่สามารถเพิ่มข้อมูลนี้ได้

```
INSERT INTO Refund_Money VALUES
('CC0006', '1Z12345E151234567TH', 15, 1, 2568, '7:12:00', 'Incomplete product', 19, 1, 2568, '9:36:00');
```

Output

Action Output

#	Time	Action	Message
1	20:33:18	INSERT INTO Refund_Money VALUES ('CC0006', '1Z12345E151234567TH', 15, 1, 2568, '7:12:00', 'Incomple...	Error Code: 3819. Check constraint 'chk_refund_money_date' is violated.

การสร้างตาราง **การคืนสินค้า** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการคืนสินค้าขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Shipping เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Return_Product แสดงผลของตารางที่ได้ผ่านการ Query

ONLINE _shopping_system*

Limit to 50000 rows

```
CREATE TABLE Return_Product (
    ID_Shipping VARCHAR(20) PRIMARY KEY NOT NULL,
    RetPro_Date TINYINT,
    RetPro_Month TINYINT,
    RetPro_Year SMALLINT,
    RetPro_Time TIME
);

INSERT INTO Return_Product VALUES
('1Z12345E1512345676',    16,    1, 2568, '9:48:00'),
('EA123456789TH',    17,    1,    2568,    '10:50:00'),
('JT123456789TH',    NULL,    NULL,    NULL,    NULL),
('NV123456789',    19,    2,    2568,    '11:27:00'),
('JV123456789TH',    20,    2,    2568,    '9:22:00');
 SELECT * FROM Return_Product
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap C

ID_Shipping	RetPro_Date	RetPro_Month	RetPro_Year	RetPro_Time
1Z12345E1512345676	16	1	2568	09:48:00
EA123456789TH	17	1	2568	10:50:00
JT123456789TH	NULL	NULL	NULL	NULL
JV123456789TH	20	2	2568	09:22:00
NV123456789	19	2	2568	11:27:00

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการคืนสินค้า

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_return_product_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าเดือนที่คืนสินค้า (RetPro_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่คืนสินค้า (RetPro _Date) อยู่ในช่วง 1 ถึง 31

หากวันที่คืนสินค้า ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเดือนที่คืนสินค้า (RetPro_Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเดือนที่คืนสินค้า (RetPro_Month) เป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเดือนที่คืนสินค้า (RetPro_Month) เป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่า วันที่เริ่มใช้ส่วนลดไม่เกิน 29 วัน

```
1   ALTER TABLE return_product
2   ADD CONSTRAINT chk_return_product_date
3   CHECK (
4       RetPro_Month BETWEEN 1 AND 12 AND
5       RetPro_Date BETWEEN 1 AND 31 AND
6       (
7           (RetPro_Month IN (1, 3, 5, 7, 8, 10, 12)) OR
8           (RetPro_Month IN (4, 6, 9, 11) AND RetPro_Date <= 30) OR
9           (RetPro_Month = 2 AND RetPro_Date <= 29)
10      )
11  );
```

ทดสอบการเพิ่มค่า โดยวันที่คืนสินค้า คือวันที่ 31 เดือน 6 ปี 2568 ในเงื่อนไขกำหนดไว้ว่า ถ้าเดือนที่คืนสินค้าคือ เดือนที่ 6 วันที่คืนสินค้าจะสามารถเพิ่มค่าได้ไม่เกิน 30 วัน ทำให้ไม่สามารถเพิ่มข้อมูลนี้ได้

```
13  INSERT INTO return_product VALUES ('TV123456789TH',   31,   6,   2568,   '9:22:00');
14
15
```

Output

Action Output

#	Time	Action	Message
1	19:40:09	INSERT INTO return_product VALUES ('TV123456789TH', 31, 6, 2568, '9:22:00')	Error Code: 3819. Check constraint 'chk_return_product_date' is violated.

ตัวอย่างโจทย์ SQL

1. แสดงรายชื่อลูกค้าที่ซื้อประเภทเครื่องดื่มและชื่อเครื่องดื่มที่ซื้อ โดยเรียงจากจำนวนที่ซื้อจากมากไปน้อย

คำสั่ง SQL

```
SELECT First_name,Last_name,Product_name,Quantity
FROM id_card,customers,orders,product
WHERE id_card.ID_card = customers.ID_card
AND customers.ID_Customer = orders.ID_Customer
AND orders.Product_code = product.Product_code
AND product.Product_type = "Beverage"
ORDER BY Quantity desc
```

```
162 •   SELECT First_name,Last_name,Product_name,Quantity
163     FROM id_card,customers,orders,product
164     WHERE id_card.ID_card = customers.ID_card
165     AND customers.ID_Customer = orders.ID_Customer
166     AND orders.Product_code = product.Product_code
167     AND product.Product_type = "Beverage"
168     ORDER BY Quantity desc;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conter

First_name	Last_name	Product_name	Quantity
Somphong	Jaidee	D-D	30
Somron	Preecha	Kato	15

คำอธิบาย

ทำการค้นหา **ชื่อ** (First_name) **นามสกุล** (Last_name) **ชื่อสินค้า** (Product_name) **จำนวนสินค้า**(Quantity)

จาก ตารางบัตรประชาชน(id_card) ตารางลูกค้า(customers) ตารางการสั่งซื้อสินค้า(orders) ตารางสินค้า(product)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**ลูกค้า** กับ **บัตรประชาชน** ด้วยแอตทริบิวต์ ID_card
- ตาราง**ลูกค้า** กับ **การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ **สินค้า** ด้วยแอตทริบิวต์ Product_code

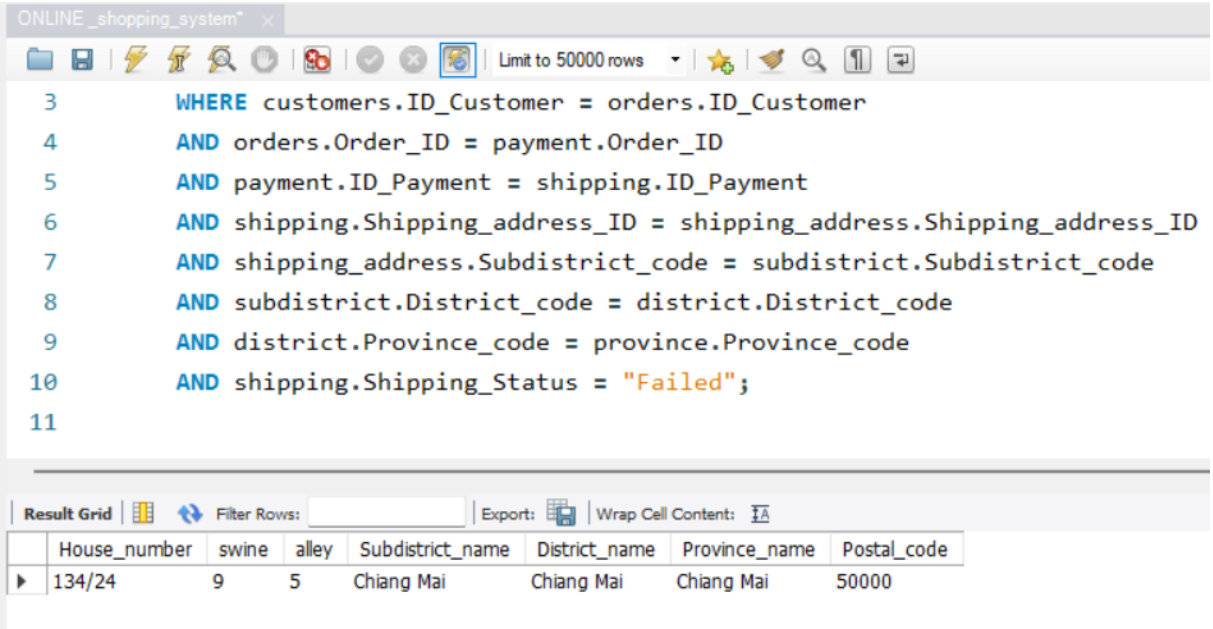
โดยเลือกเฉพาะ สินค้าที่ Product_type มีค่า Beverage (เครื่องดื่ม) และเรียงลำดับตามจำนวนสินค้าที่มาก(Quantity)

เพื่อให้ทราบความนิยมของสินค้าประเภทเครื่องดื่มว่าสินค้าใดที่มียอดขายสูงและเป็นที่นิยม

2. แสดงที่อยู่ของลูกค้าที่มีการจัดส่งไม่สำเร็จ

คำสั่ง SQL

```
SELECT House_number,swine,alley,Subdistrict_name,District_name,Province_name
,Postal_code
FROM customers,orders,payment,shipping,shipping_address,subdistrict,district
,province
WHERE customers.ID_Customer = orders.ID_Customer
AND orders.Order_ID = payment.Order_ID
AND payment.ID_Payment = shipping.ID_Payment
AND shipping.Shipping_address_ID = shipping_address.Shipping_address_ID
AND shipping_address.Subdistrict_code = subdistrict.Subdistrict_code
AND subdistrict.District_code = district.District_code
AND district.Province_code = province.Province_code
AND shipping.Shipping_Status = "Failed";
```



คำอธิบาย

ทำการค้นหา **บ้านเลขที่**(House_number) **หมู่**(swine) **ซอย**(alley) **ตำบล/แขวง**(Subdistrict_name) **อำเภอ/เขต**(District_name) **จังหวัด**(Province_name) **รหัสไปรษณีย์**(Postal_code)

จาก ตารางลูกค้า(customers) ตารางการสั่งซื้อสินค้า(orders) ตารางการจัดส่ง(shipping) ตารางที่อยู่จัดส่ง(shipping_address) ตารางตำบล/แขวง(subdistrict) ตารางอำเภอ/เขต(district) ตารางจังหวัด(province)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**ลูกค้า** กับ **การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ Order_ID
- ตาราง**การชำระเงิน** กับ **การจัดส่ง** ด้วยแอตทริบิวต์ ID_Payment
- ตาราง**การจัดส่ง** กับ **ที่อยู่จัดส่ง** ด้วยแอตทริบิวต์ Shipping_address_ID
- ตาราง**ที่อยู่จัดส่ง** กับ **ตำบล/แขวง** ด้วยแอตทริบิวต์ Subdistrict_code
- ตาราง**ตำบล/แขวง** กับ **อำเภอ/เขต** ด้วยแอตทริบิวต์ District_code
- ตาราง**อำเภอ/เขต** กับ **จังหวัด** ด้วยแอตทริบิวต์ Province_code

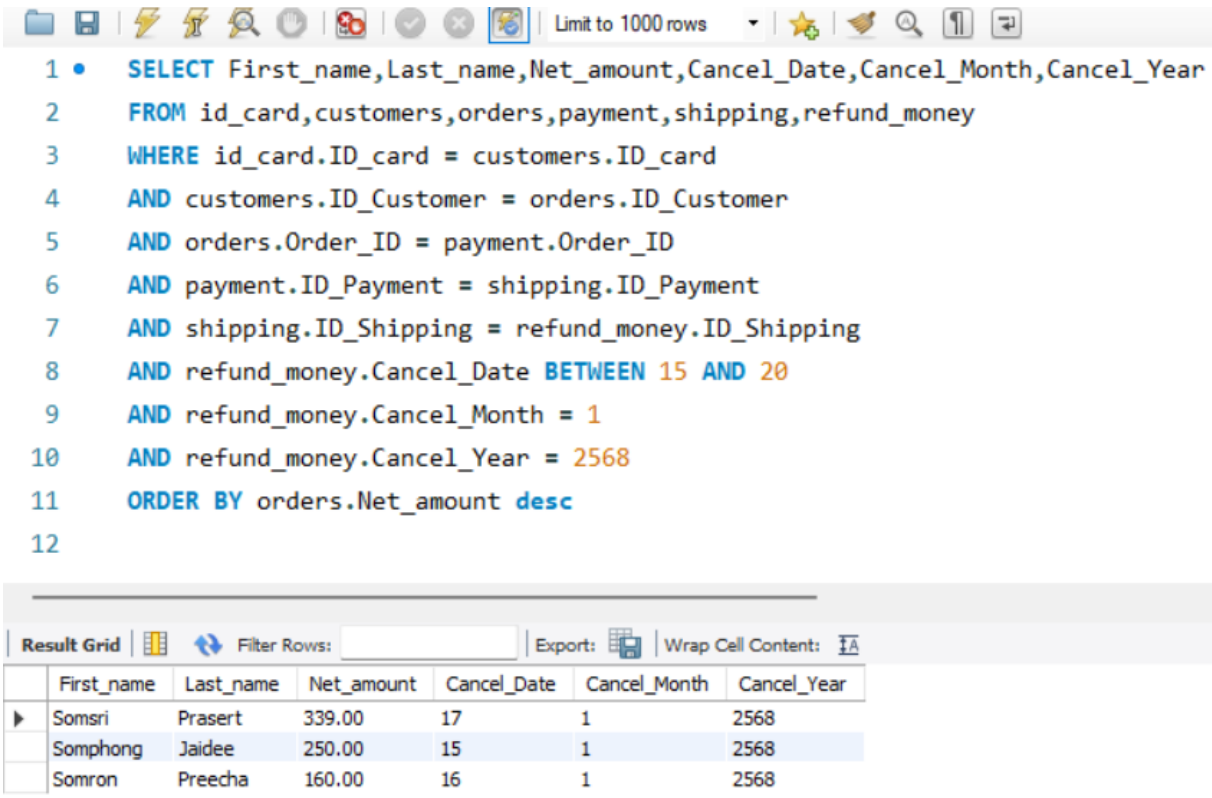
โดยเลือกเฉพาะ การจัดส่ง ที่ Shipping_Status มีค่า Failed (ไม่สำเร็จ)

เพื่อหาข้อมูลที่เกิดการจัดส่งไม่สำเร็จ และสืบค้นว่ามีการจัดส่งสินค้าใดติดค้างหรือไม่ หรือว่าของถึงมือลูกค้าหรือไม่

3. แสดงรายชื่อลูกค้าที่ทำการขอคืนเงิน ระหว่างวันที่ 15/01/68 – 20/01/68 โดยเรียงจากราคาสุทธิจากมากไปน้อย

คำสั่ง SQL

```
SELECT First_name,Last_name,Net_amount,Cancel_Date,Cancel_Month,Cancel_Year
FROM id_card,customers,orders,payment,shipping,refund_money
WHERE id_card.ID_card = customers.ID_card
AND customers.ID_Customer = orders.ID_Customer
AND orders.Order_ID = payment.Order_ID
AND payment.ID_Payment = shipping.ID_Payment
AND shipping.ID_Shipping = refund_money.ID_Shipping
AND refund_money.Cancel_Date BETWEEN 15 AND 20
AND refund_money.Cancel_Month = 1
AND refund_money.Cancel_Year = 2568
ORDER BY orders.Net_amount desc
```



First_name	Last_name	Net_amount	Cancel_Date	Cancel_Month	Cancel_Year
Somsri	Prasert	339.00	17	1	2568
Somphong	Jaidee	250.00	15	1	2568
Somron	Preecha	160.00	16	1	2568

คำอธิบาย

ทำการค้นหา **ชื่อ**(First_name) **นามสกุล**(Last_name) **ราคาสุทธิ**(Net_amount), **วันที่ยกเลิก**(Cancel_Date),**เดือนที่ยกเลิก**(Cancel_Month),**ปีที่ยกเลิก**(Cancel_Year)

จาก ตารางบัตรประชาชน(id_card) ตารางลูกค้า(customers) ตารางการสั่งซื้อสินค้า(orders) ตารางการชำระเงิน(payment) ตารางจัดส่ง(shipping) ตารางการคืนเงิน(refund_money)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**บัตรประชาชน** กับ **ลูกค้า** ด้วยแอตทริบิวต์ ID_card
- ตาราง**ลูกค้า** กับ **การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ Order_ID
- ตาราง**การชำระเงิน** กับ **การจัดส่ง** ด้วยแอตทริบิวต์ ID_Payment
- ตาราง**การจัดส่ง** กับ **การขอคืนเงิน** ด้วยแอตทริบิวต์ ID_Shipping

โดยเลือกเฉพาะลูกค้าที่ขอคืนเงินระหว่างวันที่ 15/01/68 – 20/01/68เรียงจากราคาสุทธิจากมากไปน้อย ด้วยคำสั่ง(desc)

เพื่อให้ทราบจำนวนเงินที่ระบบต้องทำการคืนเงินแก่ลูกค้า และเพื่อหาหลักฐานเวลาการขอคืนและให้รู้ระยะเวลาที่ลูกค้าต้องได้เงินคืน

4. แสดงลูกค้าที่สั่งซื้อสินค้าจากหมวดหมู่ขนมขบเคี้ยวและมีช่วงอายุที่ 20 – 30ปี

คำสั่ง SQL

```
SELECT Product_type,Gender,(YEAR(CURDATE()) + 543 - id_card.Birth_year)
AS  Age
FROM id_card,customers,orders,product
WHERE id_card.ID_card = customers.ID_card
AND customers.ID_Customer = orders.ID_Customer
AND orders.Product_code = product.Product_code
AND product.Product_type = 'snack'
AND (YEAR(CURDATE()) + 543 - id_card.Birth_year) BETWEEN 20 AND 30;
```

```
103 •  SELECT Product_type,Gender,(YEAR(CURDATE()) + 543 - id_card.Birth_year) AS Age
104    FROM id_card,customers,orders,product
105    WHERE id_card.ID_card = customers.ID_card
106    AND customers.ID_Customer = orders.ID_Customer
107    AND orders.Product_code = product.Product_code
108    AND product.Product_type = 'snack'
109    AND (YEAR(CURDATE()) + 543 - id_card.Birth_year) BETWEEN 20 AND 30;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_type	Gender	Age
Snack	F	20
Snack	F	23
Snack	M	30
Snack	F	23

คำอธิบาย

ทำการค้นหา ปีเกิด (Birth_year) ประเภทสินค้า (Product_type) และอายุ (Age) โดยคำสั่ง (YEAR(CURDATE()) + 543 - id_card.Birth_year)

จาก ตารางบัตรประชาชน(id_card) ตารางลูกค้า(customers) ตารางการสั่งซื้อสินค้า(orders) ตารางสินค้า(product)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**บัตรประชาชน** กับ **ลูกค้า** ด้วยแอตทริบิวต์ ID_card
- ตาราง**ลูกค้า** กับ **การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ **สินค้า** ด้วยแอตทริบิวต์ Product_code

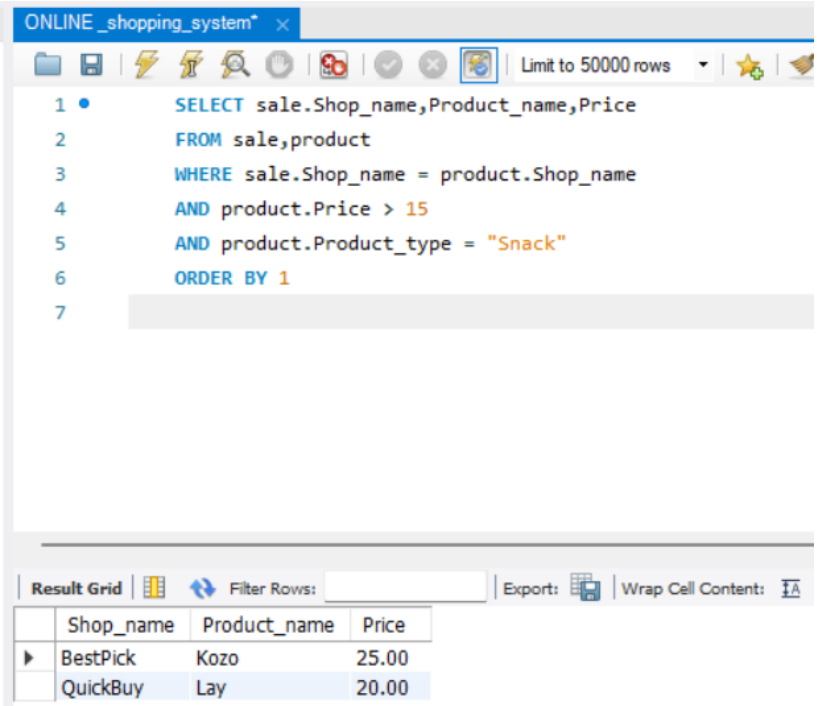
โดยเลือกเฉพาะ ลูกค้าที่มีอายุ 20 –30 ปี และซื้อสินค้าประเภทขนมขบเคี้ยว

เพื่อวิเคราะห์กลุ่มลูกค้าตามหมวดหมู่สินค้าและช่วงอายุ

5. แสดงชื่อร้านค้าที่ขายสินค้าประเภทขนมคบเคี้ยวที่มีราคามากกว่า 15 บาท โดยเรียงลำดับจากชื่อของร้านค้าตามตัวอักษร

คำสั่ง SQL

```
SELECT sale.Shop_name,Product_name,Price
FROM sale,product
WHERE sale.Shop_name = product.Shop_name
AND product.Price > 15
AND product.Product_type = "Snack"
ORDER BY 1
```



คำอธิบาย

ทำการค้นหา **ชื่อร้าน** (Shop_name) **ชื่อสินค้า** (Product_name) และ **ราคาสินค้า** (Price)

จาก ตารางผู้ขาย (sale) และตารางสินค้า(Product)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**ผู้ขาย** กับ **สินค้า** ด้วยแอตทริบิวต์ Shop_name

โดยเลือกเฉพาะ สินค้าที่มีราคามากกว่า 15 บาท และอยู่ในประเภทขนมขบเคี้ยว และเรียงลำดับชื่อตามตัวอักษรด้วยคำสั่ง order by 1

จากโจทย์การเรียงลำดับเพื่อให้ค้นหาชื่อได้ง่าย และสามารถทราบว่าขนมร้านใดมีราคาเกินกว่าที่กำหนดเพื่อที่ในอนาคตจะเกิดการจัดส่วนลดสำหรับร้านค้า หรือ สินค้า

6. แสดงรายการสินค้าที่จัดส่งหลังจากวันที่ 31/01/68 และมีราคาสุทธิมากกว่า 200 บาท

คำสั่ง SQL

```
SELECT Product_name,Shipping_Date,Shipping_Month,Shipping_Year,Net_amount
FROM product,orders,payment,shipping
WHERE product.Product_code = orders.Product_code
AND orders.Order_ID = payment.Order_ID
AND payment.ID_Payment = shipping.ID_Payment
AND shipping.Shipping_Month > 1
AND shipping.Shipping_Year >= 2568
AND orders.Net_amount > 200
```

```
171 •   SELECT Product_name,Shipping_Date,Shipping_Month,Shipping_Year,Net_amount
172     FROM product,orders,payment,shipping
173     WHERE product.Product_code = orders.Product_code
174     AND orders.Order_ID = payment.Order_ID
175     AND payment.ID_Payment = shipping.ID_Payment
176     AND shipping.Shipping_Month > 1
177     AND shipping.Shipping_Year >= 2568
178     AND orders.Net_amount > 200
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_name	Shipping_Date	Shipping_Month	Shipping_Year	Net_amount
Kozo	15	2	2568	260.00
Tao Kae Noi	10	2	2568	290.00

คำอธิบาย

ทำการค้นหา **ชื่อสินค้า** (Product_name) **วันที่จัดส่ง**(Shipping_Date) **เดือนที่จัดส่ง** (Shipping_Month) **ปีที่จัดส่ง**(Shipping_Year) และ**ราคาสุทธิ**(Net_amount)

จาก ตารางสินค้า(Product) ตารางการสั่งซื้อสินค้า(orders) ตารางการชำระเงิน(payment) และตารางการจัดส่ง(shipping)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**สินค้า** กับ **การสั่งซื้อ** ด้วยแอตทริบิวต์ Product_code
- ตาราง**การสั่งซื้อ** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ Order_ID
- ตาราง**การชำระเงิน** กับ **การจัดส่ง** ด้วยแอตทริบิวต์ ID_Payment

โดยเลือกเฉพาะ วันที่จัดส่งที่มากกว่าวันที่ 1 จัดส่งในปี 2568 และมีราคาสินค้ามากกว่า 200 บาท โดยคำสั่ง orders.Net_amount > 200

การค้นหาวันที่การจัดส่ง เพื่อบ่งบอกได้ว่าช่วงเวลานี้จะมีสินค้าใดมีการจัดส่งบ้าง และทราบราคาเพื่อบอกมูลค่ากรณีที่เกิดความเสียหายของสินค้าระหว่างการจัดส่ง

7. แสดงชื่อลูกค้า และชื่อสินค้าที่สั่งซื้อ ที่มีการให้คะแนนการรีวิวคุณภาพสินค้า บริการจากผู้ขาย และผู้ให้บริการขนส่ง น้อยกว่า 70 คะแนน

คำสั่ง SQL

```
SELECT First_name,Last_name,Product_name,Product_quality,Seller_service,
Delivery_service
FROM id_card,customers,orders,product,payment,shipping,review
WHERE id_card.ID_card = customers.ID_card
AND customers.ID_Customer = orders.ID_Customer
AND orders.Product_code = product.Product_code
AND orders.Order_ID = payment.Order_ID
AND payment.ID_Payment = shipping.ID_Payment
AND shipping.ID_Shipping = review.ID_Shipping
AND review.Product_quality < 70
AND review.Seller_service < 70
AND review. Delivery_service < 70;
```

```
1   SELECT First_name,Last_name,Product_name,Product_quality,Seller_service,Delivery_service
2   FROM id_card,customers,orders,product,payment,shipping,review
3   WHERE id_card.ID_card = customers.ID_card
4   AND customers.ID_Customer = orders.ID_Customer
5   AND orders.Product_code = product.Product_code
6   AND orders.Order_ID = payment.Order_ID
7   AND payment.ID_Payment = shipping.ID_Payment
8   AND shipping.ID_Shipping = review.ID_Shipping
9   AND review.Product_quality < 70
10  AND review.Seller_service < 70
11  AND review. Delivery_service < 70;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

First_name	Last_name	Product_name	Product_quality	Seller_service	Delivery_service
Somsri	Prasert	Lay	50	60	65

คำอธิบาย

ทำการค้นหา **ชื่อ**(First_name) **นามสกุล**(Last_name) **ชื่อสินค้า**(Product_name) **คุณภาพสินค้า**(Product_quality) **บริการจากผู้ขาย**(Seller_service) และ**ผู้ให้บริการขนส่ง**(Delivery_service)

จาก ตารางบัตรประชาชน (id_card) ตารางลูกค้า(customer) ตารางการสั่งซื้อสินค้า(orders) ตารางสินค้า(Product) ตารางการชำระเงิน(payment) ตารางการจัดส่ง(shipping) และตารางการรีวิวสินค้า(review)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**บัตรประชาชน** กับ **ลูกค้า** ด้วยแอตทริบิวต์ ID_card
- ตาราง**ลูกค้า** กับ **การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ **สินค้า** ด้วยแอตทริบิวต์ Product_code
- ตาราง**การสั่งซื้อสินค้า** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ Order_ID
- ตาราง**การชำระเงิน** กับ **การจัดส่ง** ด้วยแอตทริบิวต์ ID_Payment
- ตาราง**การจัดส่ง** กับ **การรีวิวสินค้า** ด้วยแอตทริบิวต์ ID_Shipping

โดยเลือกเฉพาะ คุณภาพสินค้า บริการจากผู้ขาย ผู้ให้บริการขนส่ง ที่มีคะแนนการรีวิวน้อยกว่า 70 เพื่อดูผลตอบกลับจากลูกค้า โดยที่ร้านค้าสามารถนำไปพัฒนาการให้บริการและการเลือกขนส่งได้ในอนาคต

8.ค้นหาลูกค้าที่สั่งซื้อสินค้าในช่วงเวลาที่กำหนดและมีที่อยู่จัดส่งในจังหวัดที่กำหนด

คำสั่ง SQL

```
SELECT c.First_name, c.Last_name, o.Order_ID, o.Order_day, o.Order_month,
o.Order_year,  p.Province_name
FROM Customers cu
JOIN ID_card c ON cu.ID_card = c.ID_card
JOIN Orders o ON cu.ID_Customer = o.ID_Customer
JOIN Payment pay ON o.Order_ID = pay.Order_ID
JOIN Shipping s ON pay.ID_Payment = s.ID_Payment
JOIN Shipping_address sa ON s.Shipping_address_ID = sa.Shipping_address_ID
JOIN Subdistrict sub ON sa.Subdistrict_code = sub.Subdistrict_code
JOIN District d ON sub.District_code = d.District_code
JOIN Province p ON d.Province_code = p.Province_code
WHERE o.Order_day BETWEEN 1 AND 31
AND o.Order_month BETWEEN 1 AND 12
AND o.Order_year = 2568
AND p.Province_name = 'Bangkok';
```

```
143 • SELECT c.First_name, c.Last_name, o.Order_ID, o.Order_day, o.Order_month, o.Order_year, p.Province_name
144   FROM Customers cu
145   JOIN ID_card c ON cu.ID_card = c.ID_card
146   JOIN Orders o ON cu.ID_Customer = o.ID_Customer
147   JOIN Payment pay ON o.Order_ID = pay.Order_ID
148   JOIN Shipping s ON pay.ID_Payment = s.ID_Payment
149   JOIN Shipping_address sa ON s.Shipping_address_ID = sa.Shipping_address_ID
150   JOIN Subdistrict sub ON sa.Subdistrict_code = sub.Subdistrict_code
151   JOIN District d ON sub.District_code = d.District_code
152   JOIN Province p ON d.Province_code = p.Province_code
153   WHERE o.Order_day BETWEEN 1 AND 31
154   AND o.Order_month BETWEEN 1 AND 12
155   AND o.Order_year = 2568
156   AND p.Province_name = 'Bangkok';
157
158
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

First_name	Last_name	Order_ID	Order_day	Order_month	Order_year	Province_name
Sommai	Songtham	SM0005	14	2	2568	Bangkok

คำอธิบาย

ทำการค้นหา **ชื่อ** (First_name) **นามสกุล** (Last_name) **รหัสคำสั่งซื้อ** (Order_ID) **วันที่สั่งซื้อ** (Order_day) **เดือนที่สั่งซื้อ** (Order_month) **ปีที่สั่งซื้อ** (Order_year) และ **ชื่อจังหวัด** (Province_name)

จากตารางลูกค้า (Customers) ซึ่งถูกตั้งชื่อย่อว่า cu, ตารางบัตรประชาชน (ID_card) ซึ่งถูกตั้งชื่อย่อว่า c, ตารางการสั่งซื้อสินค้า (Orders) ซึ่งถูกตั้งชื่อย่อว่า o, ตารางการชำระเงิน (Payment) ซึ่งถูกตั้งชื่อย่อว่า pay, ตารางการจัดส่ง (Shipping) ซึ่งถูกตั้งชื่อย่อว่า s, ตารางที่อยู่จัดส่ง (Shipping_address) ซึ่งถูกตั้งชื่อย่อว่า sa, ตารางตำบล (Subdistrict) ซึ่งถูกตั้งชื่อย่อว่า sub, ตารางอำเภอ (District) ซึ่งถูกตั้งชื่อย่อว่า d, และตารางจังหวัด (Province) ซึ่งถูกตั้งชื่อย่อว่า p

โดย การเชื่อมตารางด้วยคำสั่งJOIN และ ON มีดังนี้:

- ตาราง**ลูกค้า** กับ ตาราง**บัตรประชาชน** ด้วยแอตทริบิวต์ ID_card
- ตาราง**ลูกค้า** กับ ตาราง**การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ ตาราง**การชำระเงิน** ด้วยแอตทริบิวต์ Order_ID
- ตาราง**การชำระเงิน** กับ ตาราง**การจัดส่ง** ด้วยแอตทริบิวต์ ID_Payment
- ตาราง**การจัดส่ง** กับ ตาราง**ที่อยู่จัดส่ง** ด้วยแอตทริบิวต์ Shipping_address_ID
- ตาราง**ที่อยู่จัดส่ง** กับ ตาราง**ตำบล** ด้วยแอตทริบิวต์ Subdistrict_code
- ตาราง**ตำบล** กับ ตาราง**อำเภอ** ด้วยแอตทริบิวต์ District_code
- ตาราง**อำเภอ** กับ ตาราง**จังหวัด** ด้วยแอตทริบิวต์ Province_code

โดยเลือกเฉพาะ คำสั่งซื้อที่มี วันที่สั่งซื้อ (Order_day) อยู่ในช่วงวันที่ 1 ถึง 31 เดือนที่สั่งซื้อ (Order_month) อยู่ในช่วงเดือน 1 ถึง 12 ปีที่สั่งซื้อ (Order_year) เท่ากับ 2568 และ ชื่อจังหวัด (Province_name) เท่ากับ 'Bangkok'

เพื่อวิเคราะห์กลุ่มลูกค้าตามพื้นที่ หรือ ใช้วางแผนการตลาด

9. แสดงชื่อร้านค้าและยอดรวมทั้งหมดที่ลูกค้าสั่งซื้อสินค้าภายในร้าน ในวันที่ 01/01/2568 – 31/01/2568 โดยเรียงลำดับจากชื่อของร้านค้าตามตัวอักษร

คำสั่ง SQL

```
SELECT sale.Shop_name, orders.Total_amount, orders.Order_day,
orders.Order_month,   orders.Order_year
FROM orders, product, sale
WHERE orders.Product_code = product.Product_code
AND product.Shop_name = sale.Shop_name
AND orders.Order_day BETWEEN  1 AND 31
AND orders.Order_month = 1
AND orders.Order_year = 2568
ORDER BY sale.Shop_name;
```

```
192 •   SELECT sale.Shop_name, orders.Total_amount, orders.Order_day,   orders.Order_month,    orders.Order_year
193     FROM orders, product, sale
194     WHERE orders.Product_code = product.Product_code
195     AND product.Shop_name = sale.Shop_name
196     AND orders.Order_day BETWEEN  1 AND 31
197     AND orders.Order_month = 1
198     AND orders.Order_year = 2568
199     ORDER BY sale.Shop_name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Shop_name	Total_amount	Order_day	Order_month	Order_year
QuickBuy	240.00	24	1	2568
ShopEase	250.00	11	1	2568
TrendzyMall	120.00	14	1	2568

คำอธิบาย

ทำการค้นหา **ชื่อร้านค้า**(Shop_name) **ยอดรวม**(Total_amount) **วันที่สั่งซื้อ**(Order_day) **เดือนที่สั่งซื้อ**(Order_month) และ **ปีที่สั่งซื้อ**(Order_year)

จากตารางการสั่งซื้อสินค้า(orders) ตารางสินค้า(product) ตารางผู้ขาย(sale)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**การสั่งซื้อสินค้า**กับ **สินค้า** ด้วยแอตทริบิวต์ Product_code
- ตาราง**สินค้า** กับ **ผู้ขาย** ด้วยแอตทริบิวต์ Shop_name

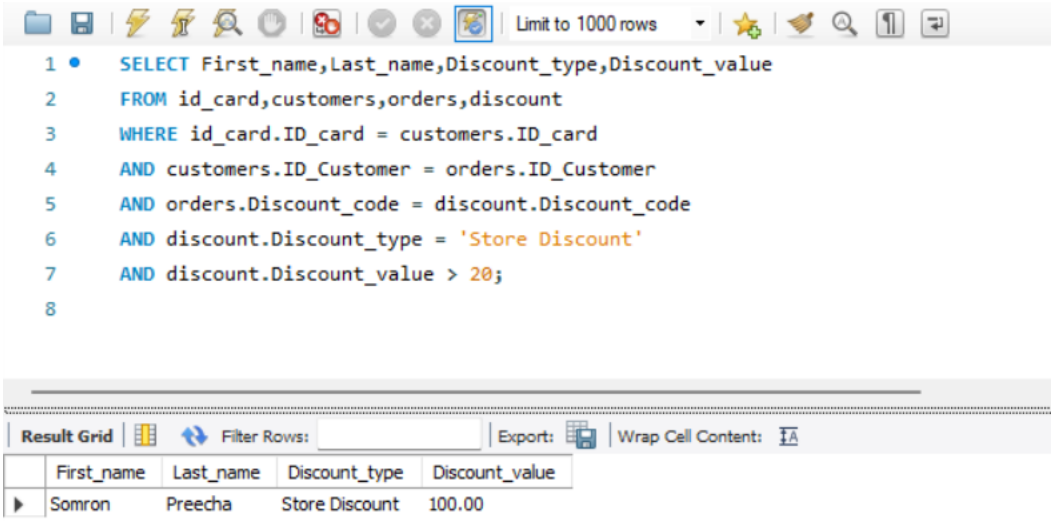
โดยเลือกเฉพาะ ลูกค้าสั่งซื้อสินค้าระหว่างวันที่ 01– 31 เดือนที่ 1 ปี 2568 และเรียงลำดับจากชื่อร้านค้าตามตัวอักษรด้วยคำสั่ง order by sale.shop_name

เพื่อเก็บสถิติการซื้อสินค้าของลูกค้า ในแต่ละเดือนลูกค้าซื้อสินค้าใดในร้านและนิยมซื้อช่วงไหน

10. แสดงชื่อลูกค้าที่ใช้ส่วนลดประเภท ส่วนลดร้านค้า ที่มีมูลค่าของส่วนลดมากกว่า 20 บาท

คำสั่ง SQL

```
SELECT First_name,Last_name,Discount_type,Discount_value
FROM id_card,customers,orders,discount
WHERE id_card.ID_card = customers.ID_card
AND customers.ID_Customer = orders.ID_Customer
AND orders.Discount_code = discount.Discount_code
AND discount.Discount_type = 'Store Discount'
AND discount.Discount_value > 20
```



คำอธิบาย

ทำการค้นหา **ชื่อ** (First_name) **นามสกุล** (Last_name) **ประเภทส่วนลด** (Discount_type) **มูลค่าของส่วนลด**(Discount_value)

จาก ตารางบัตรประชาชน(id_card) ตารางลูกค้า(customers) ตารางการสั่งซื้อสินค้า(orders) ตารางส่วนลด(discount)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**ลูกค้า** กับ **บัตรประชาชน** ด้วยแอตทริบิวต์ ID_card
- ตาราง**ลูกค้า** กับ **การสั่งซื้อสินค้า** ด้วยแอตทริบิวต์ ID_Customer
- ตาราง**การสั่งซื้อสินค้า** กับ **ส่วนลด** ด้วยแอตทริบิวต์ Discount_code

โดยเลือกเฉพาะ ลูกค้าที่ใช้ส่วนลดประเภท ส่วนลดร้านค้า(Store Discount) และมูลค่าของส่วนลด(Discount_value) มากกว่า 20 บาท

เพื่อดูจำนวนผู้ใช้ส่วนลด เพื่อดูการตอบรับการใช้ส่วนลดที่มีอยู่สามารถนำไปใช้ต่อรอบหน้าหรือเพิ่มมูลค่า เพื่อเพิ่มยอดขายหรือคำสั่งซื้อ

ระบบการจองที่จอดรถยนต์

1.ทำความเข้าใจระบบ

1. รถยนต์จะต้องมีข้อมูลรายละเอียดของรถยนต์ โดยขั้นตอนนี้จะมีการเก็บข้อมูล ป้ายทะเบียน รหัสเจ้าของรถยนต์ ยี่ห้อรถยนต์ รุ่นรถยนต์ สีรถ ประเภทรถยนต์
2. เจ้าของรถยนต์ต้องมีข้อมูลส่วนตัว โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสเจ้าของรถยนต์ ชื่อ นามสกุล เบอร์โทรศัพท์ อีเมล
3. เจ้าของรถยนต์จะต้องมีบัญชีที่ใช้สำหรับการชำระเงิน โดยขั้นตอนนี้จะมีการเก็บข้อมูล เลขบัญชี รหัสเจ้าของรถยนต์ ชื่อธนาคาร ชื่อบัญชี
4. เจ้าของรถยนต์ทำการจองที่จอดรถยนต์ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสการจองที่จอดรถยนต์ รหัสที่จอดรถยนต์ ป้ายทะเบียน วันที่จอง เดือนที่จอง ปีที่จอง เวลาที่จอง วันที่จอด เดือนที่จอด ปีที่จอด เวลาที่จอด สถานะการจอง
5. เมื่อจองเสร็จเรียบร้อยแล้ว จึงจะสามารถนำรถมาจอดได้ ซึ่งการจอดรถยนต์จะต้องนำมาจอดภายในที่จอดรถยนต์ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสการจอดรถยนต์ รหัสการจองที่จอดรถยนต์ วันที่จอด เดือนที่จอด ปีที่จอด เวลาที่จอด เวลาออก ค่าบริการ รหัสที่จอดรถยนต์ ประเภทที่จอดรถยนต์ อาคารจอดรถยนต์ ชั้น ค่าจอด/ชั่วโมง สถานะที่จอดรถยนต์
6. เมื่อทำการจอด เสร็จสิ้นแล้วจึงจะทำการชำระเงิน โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสการชำระเงิน รหัสการจอดรถยนต์ จำนวนเงินที่ชำระ วันที่ชำระเงิน เดือนที่ชำระเงิน ปีที่ชำชะเงิน เวลาที่ชำระเงิน
7. ซึ่งภายในที่จอดรถยนต์จะมีการรักษาความปลอดภัย โดยจะมีกล้องวงจรปิด และเจ้าหน้าที่ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสกล้องวงจรปิด รหัสที่จอดรถยนต์ ยี่ห้อกล้องวงจรปิด วันที่ตรวจสอบ เดือนที่ตรวจสอบ ปีที่ตรวจสอบ เวลาที่ตรวจสอบล่าสุด วันที่เก็บได้สูงสุด เส้นทางที่เก็บไฟล์ สถานะกล้องวงจรปิด รหัสเจ้าหน้าที่ รหัสที่จอดรถยนต์ ช่วงเวลาทำงาน ชื่อ นามสกุล ตำแหน่ง เบอร์โทรศัพท์ เวลาเริ่มงาน เวลาออกงาน
8. หลังจากที่มีการใช้งานที่จอดรถยนต์ เจ้าหน้าที่จะต้องมีการตรวจสอบที่จอดรถยนต์ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสการตรวจสอบ รหัสที่จอดรถยนต์ วันที่ตรวจสอบ เดือนที่ตรวจสอบ ปีที่ตรวจสอบ เวลาที่ตรวจสอบ ปัญหาที่พบ ความคืบหน้า

ข้อกำหนด

- กล้องวงจรปิด สามารถเก็บไฟล์ได้อย่างน้อย 90 วัน
- ระบบการจองที่จอดรถยนต์นี้จะเก็บข้อมูลในช่วงปี พ.ศ. 2565 ถึง พ.ศ. 2568
- ที่จอดรถยนต์มีทั้งหมด 4 อาคาร แต่ละอาคาร มี 7 ชั้น

2. สร้าง Entity Relational Diagram

ตารางรถยนต์

ป้ายทะเบียน	รหัสเจ้าของรถยนต์	ยี่ห้อรถยนต์	รุ่นรถยนต์	สีรถ	ประเภทรถยนต์
กข-1234	Q00123	Toyota	Corolla	ขาว	เก๋ง
ขค-5678	Q00124	Toyota	Hilux	ดำ	กระบะ
งง-9101	Q00125	Ford	Ranger	น้ำเงิน	กระบะ
จจ-2345	Q00126	Nissan	Almera	เทา	เก๋ง
ฉฉ-6789	Q00127	Mazda	CX-5	แดง	SUV
ดบ-1235	Q00127	Honda	Civic	เงิน	เก๋ง
กร-5679	Q00125	Isuzu	D-Max	ขาว	กระบะ
ลน-9102	Q00125	Mitsubishi	Pajero Sport	เทา	SUV

ตารางรถยนต์ประกอบด้วย Attributes ดังนี้

ป้ายทะเบียน : ใช้สำหรับระบุรถยนต์แต่ละคัน เพื่อนำไปค้นหาข้อมูลของรถยนต์

รหัสเจ้าของรถยนต์ : เพื่อให้ทราบว่าใครเป็นเจ้าของรถยนต์คันนี้ สามารถตรวจสอบชื่อผ่านรหัสเจ้าของรถยนต์ได้

ยี่ห้อรถยนต์ : เพื่อระบุยี่ห้อของรถยนต์

รุ่นรถยนต์ : เพื่อบอกรุ่นของรถยนต์

สีรถ : ใช้ระบุสีของรถยนต์

ประเภทรถยนต์ : เพื่อแยกประเภทรถยนต์ เช่น รถเก๋ง รถกระบะ SUV

ตารางเจ้าของรถยนต์

รหัสเจ้าของรถยนต์	ชื่อ	นามสกุล	เบอร์โทรศัพท์	อีเมล
O00123	สมชาย	แก้วใส	8-1234-5678	somchai@gmail.com
O00124	มานะ	ทองดี	8-9876-5432	mana@gmail.com
O00125	สายฝน	วิริยะ	8-2345-6789	saifon@gmail.com
O00126	อภิชัย	บุญมี	9-0456-7890	apichai@gmail.com
O00127	วิภา	รัตนวงศ์	9-5567-8901	wipa@gmail.com

ตารางเจ้าของรถยนต์ ประกอบด้วย Attributes ดังนี้

รหัสเจ้าของรถยนต์ : รหัสประจำตัวเจ้าของรถยนต์ ใช้สำหรับค้นหาข้อมูลของเจ้ารถยนต์แต่ละคน

ชื่อ-นามสกุล : เพื่อจัดเก็บข้อมูล ชื่อ-นามสกุล ของเจ้าของรถยนต์

เบอร์โทรศัพท์ : เพื่อใช้สำหรับติดต่อเจ้าของรถยนต์

อีเมล : เพื่อใช้สำหรับติดต่อเจ้าของรถยนต์ ในกรณีที่ต้องการส่งเป็นไฟล์ เช่น หลักฐานการเข้าจอด

ตารางบัญชี

เลขบัญชี	รหัสเจ้าของรถยนต์	ชื่อธนาคาร	ชื่อบัญชี
1234567890	O00123	กสิกรไทย	สมชาย แก้วใส
2345678901	O00124	ไทยพาณิชย์	มานะ ทองดี
3456789012	O00125	กรุงเทพ	สายฝน วิริยะ
4567890123	O00126	กรุงไทย	อภิชัย บุญมี
5678901234	O00127	กรุงศรี	วิภา รัตนวงศ์

ตารางบัญชีประกอบด้วย Attributes ดังนี้

เลขบัญชี : เพื่อระบุหมายเลขบัญชีธนาคารที่ใช้ในการชำระเงิน

รหัสเจ้าของรถยนต์ : ใช้เพื่อทราบว่าใครเป็นเจ้าของบัญชีนี้

ชื่อธนาคาร : เพื่อทราบว่าบัญชีนี้เป็นของธนาคารอะไร

ชื่อบัญชี : ชื่อที่ใช้สำหรับบัญชีธนาคาร

ตารางการชำระเงิน

รหัสการชำระเงิน	รหัสการจอดรถยนต์	จำนวนเงินที่ชำระ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระเงิน
P00001	A00001	30	2	1	2567	9:10:00
P00002	A00002	100	3	2	2567	11:25:00
P00003	A00003	200	3	2	2567	11:40:00
P00004	A00004	60	3	2	2567	11:10:00
P00005	A00005	50	4	2	2567	11:10:00
P00006	A00006	150	5	2	2567	11:40:00
P00007	A00007	150	5	3	2567	12:10:00
P00008	A00008	150	5	4	2567	13:25:00
P00009	A00009	100	5	4	2567	13:10:00

ตารางการชำระเงิน ประกอบด้วย Attributes ดังนี้

รหัสการชำระเงิน : เป็นรหัสอ้างอิงของแต่ละรายการชำระเงินแต่ละอัน

รหัสการจอดรถยนต์ : ใช้ตรวจสอบได้ว่าการจอดรถยนต์ไหนที่มาทำการชำระเงิน

จำนวนเงินที่ชำรถ : เพื่อบอกยอดเงินที่ชำระ

วัน-เดือน-ปี-เวลาที่ชำระเงิน : บันทึกวัน เดือน ปี และเวลาที่ทำรายการชำระเงิน เพื่อใช้ในการตรวจได้

ตารางการจองที่จอดรถยนต์

รหัสการจองที่จอดรถยนต์	รหัสที่จอดรถยนต์	ป้ายทะเบียน	วันที่จอง	เดือนที่จอง	ปีที่จอง	เวลาที่จอง	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	สถานะการจอง
B00001	P00101	กข-1234	1	1	2567	10:30:00	2	1	2567	8:00:00	สำเร็จ
B00002	P00102	ขค-5678	2	2	2567	14:00:00	3	2	2567	9:15:00	สำเร็จ
B00003	P00102	งง-9101	2	2	2567	9:45:00	3	2	2567	7:30:00	สำเร็จ
B00004	P00104	จจ-2345	2	2	2567	16:20:00	3	2	2567	10:00:00	สำเร็จ
B00005	P00105	ฉฉ-6789	2	2	2567	11:10:00	2	2	2567	8:45:00	ยกเลิก
B00006	P00103	ฉฉ-6789	3	2	2567	13:10:00	4	2	2567	10:00:00	สำเร็จ
B00007	P00103	ฉฉ-6789	4	2	2567	10:00:00	5	2	2567	8:30:00	สำเร็จ
B00008	P00107	ขค-5678	4	3	2567	11:30:00	5	3	2567	9:00:00	สำเร็จ
B00009	P00107	งง-9101	4	4	2567	13:00:00	5	4	2567	10:15:00	สำเร็จ
B00010	P00107	จจ-2345	4	4	2567	15:00:00	5	4	2567	11:00:00	สำเร็จ

ตารางการจองที่จอดรถยนต์ ประกอบด้วย Attributes ดังนี้

รหัสการจองที่จอดรถยนต์ : เพื่ออ้างอิงของแต่ละการจอง

รหัสที่จอดรถยนต์ : ใช้เพื่อเชื่อมโยงว่าเป็นการจองที่จอดรถยนต์ไหน

ป้ายทะเบียน : ใช้เพื่อรถบุรถยนต์ที่ทำการจองที่จอด

วัน-เดือน-ปี-เวลาที่จอง : เพื่อบันทึก วัน-เดือน-ปี-เวลาที่ทำการจอง

วัน-เดือน-ปี-เวลาที่จอด : เพื่อระบุ วัน-เดือน-ปี-เวลาที่จอด ใช้สำหรับตรวจสอบการทำรายการได้

สถานะการจอง : เพื่อบอกถึงสถานะการจองที่จอดว่าสำเร็จ หรือ ยกเลิก

ตารางการจอดรถยนต์

รหัสการจอดรถยนต์	รหัสการจองที่จอดรถยนต์	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	เวลาออก	ค่าบริการ
A00001	B00001	2	1	2567	8:00:00	9:00:00	30
A00002	B00002	3	2	2567	9:15:00	11:15:00	100
A00003	B00003	3	2	2567	7:30:00	11:30:00	200
A00004	B00004	3	2	2567	10:00:00	11:00:00	60
A00005	B00006	4	2	2567	10:00:00	11:00:00	50
A00006	B00007	5	2	2567	8:30:00	11:30:00	150
A00007	B00008	5	3	2567	9:00:00	12:00:00	150
A00008	B00009	5	4	2567	10:15:00	13:15:00	150
A00009	B00010	5	4	2567	11:00:00	13:00:00	100

ตารางการจอดรถยนต์ ประกอบด้วย Attributes ดังนี้

รหัสการจอดรถยนต์ : เพื่ออ้างอิงการจอดรถยนต์แต่ละการจอด

รหัสการจองที่จอดรถยนต์ : เพื่อระบุว่าการจอดรถยนต์นี้มาจากการจองรถยนต์อันไหน

วัน-เดือน-ปี-เวลาเข้า-เวลาออก : เพื่อบันทึกวัน-เดือน-ปี-เวลาเข้า-เวลาออก ใช้สำหรับการตรวจสอบได้

ค่าบริการ : บอกจำนวนเงินที่ต้องชำระ

ตารางที่จอดรถยนต์

รหัสที่จอดรถยนต์	ประเภทที่จอดรถยนต์	อาคารจอดรถยนต์	ชั้น	ค่าจอด / ชั่วโมง	สถานะที่จอดรถยนต์
P00101	ที่จอดสำหรับผู้พิการ	1	1	30.00	พร้อมใช้งาน
P00102	ที่จอดทั่วไป	1	1	50.00	พร้อมใช้งาน
P00103	ที่จอดทั่วไป	2	1	50.00	พร้อมใช้งาน
P00104	ที่จอดรถสำหรับรถไฟฟ้า	3	2	60.00	พร้อมใช้งาน
P00105	ที่จอดรถสำหรับรถไฟฟ้า	4	3	60.00	พร้อมใช้งาน
P00106	ที่จอดทั่วไป	2	1	50.00	พร้อมใช้งาน
P00107	ที่จอดทั่วไป	2	1	50.00	พร้อมใช้งาน
P00108	ที่จอดสำหรับผู้พิการ	2	1	30.00	ชำรุด
P00109	ที่จอดสำหรับผู้พิการ	4	1	30.00	ชำรุด

ตารางที่จอดรถยนต์ ประกอบด้วย Attributes ดังนี้

รหัสที่จอดรถยนต์ : เพื่ออ้างอิงการจอดแต่ละการจอด

ประเภทที่จอดรถยนต์ : เพื่อบอกว่าที่จอดรถนั้นเป็นประเภทอะไร เช่น ที่จอดรถทั่วไป ที่จอดรถสำหรับรถไฟฟ้า ที่จอดสำหรับผู้พิการ

อาคารจอดรถยนต์ : เพื่อระบุอาคารที่ตั้งของที่จอด

ชั้น : ระบุชั้นของที่จอดรถ

ค่าจอด / ชั่วโมง : บอกค่าบริการต่อชั่วโมง

สถานะที่จอดรถยนต์ : บอกถึงสถานะการจอดรถยนต์เช่น พร้อมใช้งาน หรือ ชำรุดอยู่

ตารางกล้องวงจรปิด

รหัสกล้องวงจรปิด	รหัสที่จอดรถยนต์	ยี่ห้อกล้องวงจรปิด	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบล่าสุด	จำนวนวันที่เก็บได้สูงสุด	เส้นทางที่เก็บไฟล์	สถานะกล้องวงจรปิด
C00001	P00101	Hikvision	11	1	2567	21:24:00	90	D:/CCTV/Camera1/	ใช้งานได้
C00002	P00102	Dahua	11	1	2567	12:48:00	90	D:/CCTV/Camera2/	ใช้งานได้
C00003	P00103	Panasonic	11	1	2567	9:43:00	90	D:/CCTV/Camera3/	ชำรุด
C00004	P00104	Panasonic	11	2	2567	15:53:00	120	D:/CCTV/Camera4/	ใช้งานได้
C00005	P00105	Sony	11	2	2567	16:02:00	90	D:/CCTV/Camera5/	ใช้งานได้
C00006	P00106	Hikvision	12	3	2567	10:30:00	90	D:/CCTV/Camera6/	ชำรุด
C00007	P00107	Hikvision	12	3	2567	11:15:00	90	D:/CCTV/Camera7/	ชำรุด
C00008	P00108	Hikvision	12	3	2567	13:00:00	120	D:/CCTV/Camera8/	ใช้งานได้
C00009	P00109	Panasonic	12	4	2567	14:45:00	90	D:/CCTV/Camera9/	ใช้งานได้

ตารางกล้องวงจรปิด ประกอบด้วย Attributes ดังนี้

รหัสกล้องวงจรปิด : เพื่ออ้างอิงกล้องวงจรปิดแต่ละตัว

รหัสที่จอดรถยนต์ : เพื่อบอกว่ากล้องวงจรปิดนี้มองเห็นที่จอดรถยนต์ที่ไหน

ยี่ห้อกล้องวงจรปิด : ระบุยี่ห้อกล้อง เผื่อติดต่อในการซ่อมบำรุงหรือเกี่ยวกับการใช้งาน

วัน-เดือน-ปี-เวลาที่ตรวจสอบ : เพื่อบอกวัน-เดือน-ปี-เวลาไหนที่ได้ทำการตรวจสอบกล้องวงจรปิดล่าสุด

จำนวนวันที่เก็บได้สูงสุด : ระบุระยะเวลาสูงสุดทีสามารถบันทึกวิดีโอได้

เส้นทางที่เก็บไฟล์ : เพื่อเข้าถึงตำแหน่งที่จัดเก็บวิดิโอได้ เผื่อต้องการดูกล้อง

สถานะกล้องวงจรปิด : บอกถึงสถานะกล้องว่าใช้งานได้หรือไม่ เช่น ใช้งานได้ ชำรุด

ตารางเจ้าหน้าที่

รหัสเจ้าหน้าที่	รหัสที่จอดรถยนต์	ช่วงเวลาทำงาน	ชื่อ	นามสกุล	ตำแหน่ง	เบอร์โทรศัพท์	เวลาเริ่มงาน	เวลาออกงาน
N00007	P00101	ช่วงที่ 1	สมพร	ขจรศักดิ์	ยาม	6-6778-6251	8:00:00	16:00:00
N00008	P00102	ช่วงที่ 1	สมจิตร	มิตรผล	ยาม	9-9123-4545	8:00:00	16:00:00
N00001	P00103	ช่วงที่ 1	สมรักษ์	มักสุข	หัวหน้ายาม	8-8578-9012	8:00:00	16:00:00
N00003	P00104	ช่วงที่ 2	สมถวิล	นิมิตรหมาย	ยาม	6-3337-2775	16:00:00	0:00:00
N00004	P00105	ช่วงที่ 3	สมปอง	ก้องเกียรติ	ยาม	8-5654-7895	0:00:00	8:00:00
N00009	P00106	ช่วงที่ 4	สมชาย	ตั้งใจจริง	เจ้าหน้าที่ทำความสะอาด	6-2345-6789	7:00:00	15:00:00
N00010	P00103	ช่วงที่ 1	สมหญิง	แข่ตั้ง	เจ้าหน้าที่ซ่อมบำรุง	7-7890-2345	8:00:00	16:00:00
N00011	P00104	ช่วงที่ 2	สมพงษ์	จันทร์เจริญ	พนักงานเก็บเงิน	9-5678-3456	16:00:00	0:00:00
N00012	P00109	ช่วงที่ 4	สมฤดี	เพียรดี	เจ้าหน้าที่ทำความสะอาด	5-3456-7890	7:00:00	7:00:00

ตารางเจ้าหน้าที่ ประกอบด้วย Attributes ดังนี้

รหัสที่จอดรถยนต์ : เป็นรหัสในการระบุที่จอดรถยนต์

ช่วงเวลาทำงาน : เพื่อใช้บอกกะ ในการทำงานของเจ้าหน้าที่แต่ละคน

ชื่อ-นามสกุล : เพื่อให้ทราบว่าเจ้าหน้าที่ชื่อ - นามสกุลอะไร

ตำแหน่ง : เพื่อระบุตำแหน่งงานของเจ้าหน้าที่

เบอร์โทรศัพท์ : เพื่อใช้ติดต่อเจ้าหน้าที่

เวลาเริ่ม-ออกงาน : เพื่อใช้ระบุเวลาที่เริ่มทำงาน และเลิกทำงาน ของเจ้าหน้าที่แต่ละคน

ตารางการตรวจสอบที่จอดรถยนต์

รหัสการตรวจสอบ	รหัสที่จอดรถยนต์	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบ	ปัญหาที่พบ	สถานะการซ่อมแซม
K00001	P00101	20	2	2567	8:00:00	ไฟไม่ติด	รอการซ่อมแซม
K00002	P00102	21	3	2567	9:15:00	สีเส้นไม่ชัด	ซ่อมเสร็จแล้ว
K00003	P00103	22	4	2567	7:30:00	NULL	NULL
K00004	P00104	23	5	2567	10:00:00	NULL	NULL
K00005	P00105	24	6	2567	8:45:00	กล้องมองไม่เห็น	กำลังซ่อมแซม
K00006	P00101	25	7	2567	12:00:00	ไฟไม่ติด	ซ่อมเสร็จแล้ว
K00007	P00101	26	7	2567	15:00:00	NULL	NULL
K00008	P00101	27	7	2567	14:30:00	กล้องบันทึกภาพไม่ได้	รอการซ่อมแซม
K00009	P00102	28	8	2567	16:00:00	ไฟไม่ติด	ซ่อมเสร็จแล้ว

ตารางการตรวจสอบที่จอดรถยนต์ ประกอบด้วย Attributes ดังนี้

รหัสการตรวจสอบ : เป็นรหัสในการอ้างอิงแต่ละการตรวจสอบ

รหัสที่จอดรถยนต์ : เป็นรหัสในการระบุที่จอดรถยนต์

วัน-เดือน-ปี-เวลาที่ตรวจสอบ : เพื่อบันทึกวัน เดือน ปี และเวลาที่ทำการตรวจสอบ

ปัญหาที่พบ : เพื่อใช้ระบุปัญหาที่พบ จากการตรวจสอบ

สถานะการซ่อมแซม : เพื่อใช้เป็นการติดตามการแก้ไขของปัญหา

การวิเคราะห์หา Entity **และ** Attributes

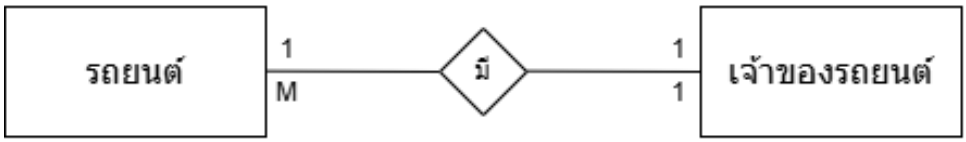
- รถยนต์
- เจ้าของรถยนต์
- บัญชี
- การชำระเงิน
- การจองที่จอดรถยนต์
- การจอดรถยนต์
- ที่จอดรถยนต์
- กล้องวงจรปิด
- เจ้าหน้าที่
- การตรวจสอบที่จอดรถยนต์

Key Attributes

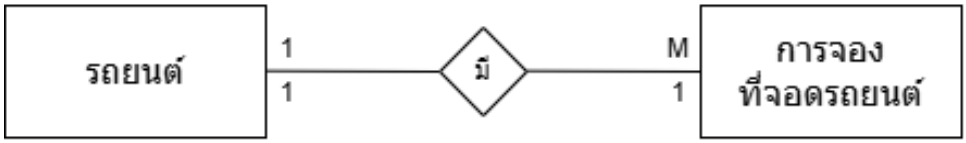
- ป้ายทะเบียน
- รหัสเจ้าของรถยนต์
- เลขบัญชี
- รหัสการชำระเงิน
- รหัสการจองที่จอดรถยนต์
- รหัสการจอดของรถยนต์
- รหัสที่จอดรถยนต์
- รหัสกล้องวงจรปิด
- รหัสเจ้าหน้าที่
- รหัสการตรวจสอบ

วิเคราะห์หาความสัมพันธ์ระหว่าง Entity

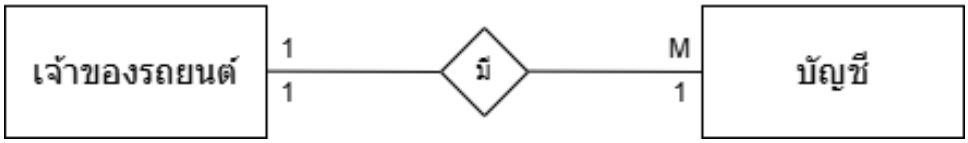
- **รถยนต์ กับ เจ้าของรถยนต์**
  - รถยนต์ 1 คัน มี เจ้าของรถยนต์ได้ 1 คน
  - เจ้าของรถยนต์ 1 คน มี รถยนต์ได้หลายคัน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M



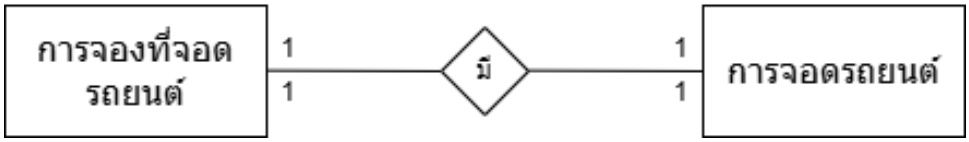
- **รถยนต์ กับ การจองที่จอดรถยนต์**
  - รถยนต์ 1 คัน มี การจองที่จอดรถยนต์ได้หลายครั้ง
  - การจองที่จอดรถยนต์ 1 ครั้ง ถูกจองได้จาก รถยนต์ 1 คัน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M



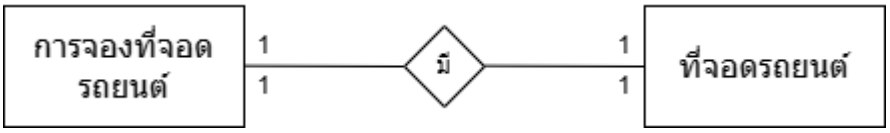
- **เจ้าของรถยนต์ กับ บัญชี**
  - เจ้าของรถยนต์ 1 คน มี บัญชีได้หลายบัญชี
  - บัญชี 1 บัญชี มี เจ้าของได้เพียง 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M



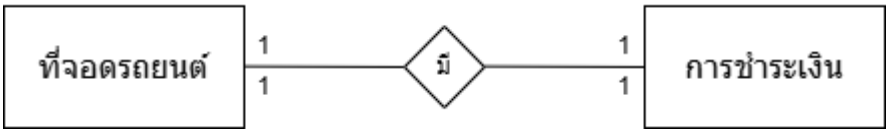
- **การจองที่จอดรถยนต์ กับ การจอดรถยนต์**
  - การจองที่จอดรถยนต์ 1 ครั้ง มี การจอดรถยนต์ได้ 1 คัน
  - การจอดรถยนต์ 1 คัน เกิดจาก การจองที่จอดรถยนต์ 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1 : 1



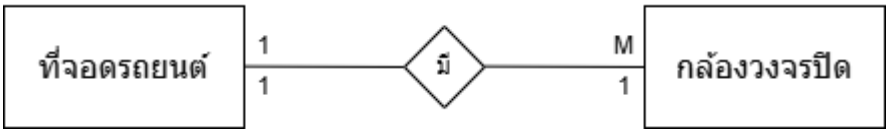
- **การจองที่จอดรถยนต์ กับ ที่จอดรถยนต์**
  - การจองที่จอดรถยนต์ 1 ครั้ง จะมี ที่จอดรถยนต์ 1 ที่
  - ที่จอดรถยนต์ 1 ที่ เกิดจาก การจองที่จอดรถยนต์ 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1 : 1



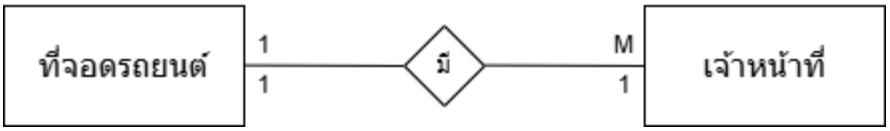
- **การจอดรถยนต์ กับ การชำระเงิน**
  - การจอดรถยนต์ 1 คัน มี การชำระเงิน 1 ครั้ง
  - การชำระเงิน 1 ครั้ง มี การจอดรถได้ 1 คัน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : 1



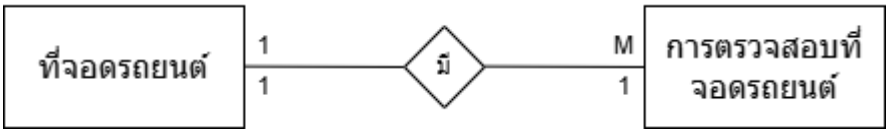
- **ที่จอดรถยนต์ กับ กล้องวงจรปิด**
  - ที่จอดรถยนต์ 1 ที่ มี กล้องวงจรปิดได้หลายตัว
  - กล้องวงจรปิด 1 ตัว มองเห็น ได้ 1 ที่จอดรถยนต์
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M



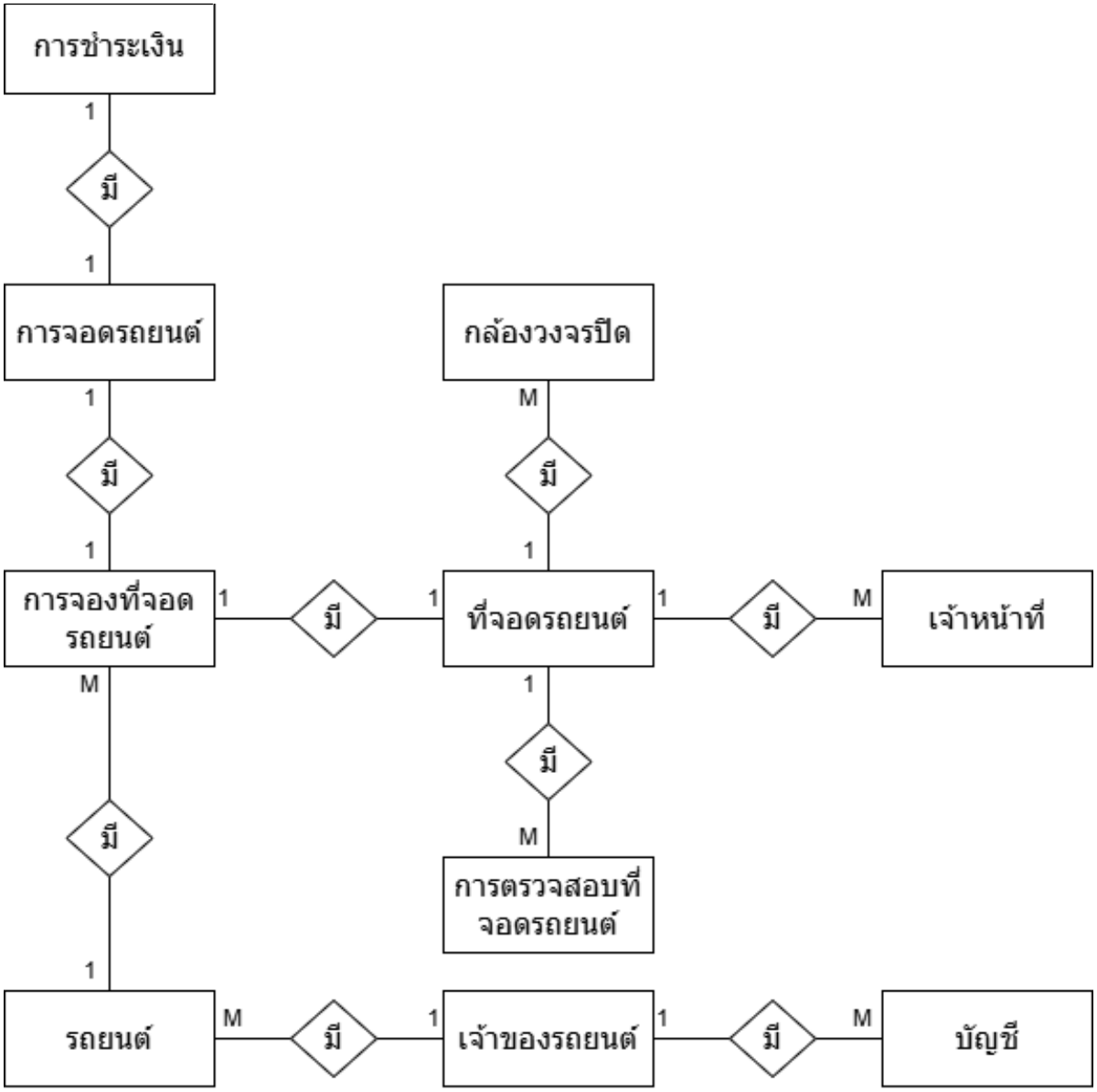
- **ที่จอดรถยนต์ กับ เจ้าหน้าที่**
  - ที่จอดรถยนต์ 1 ที่ มี เจ้าหน้าที่ได้หลายคน
  - เจ้าหน้าที่ 1 คน ดูแล ที่จอดรถยนต์ได้ 1 ที่
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M



- **ที่จอดรถยนต์ กับ การตรวจสอบที่จอดรถยนต์**
  - ที่จอดรถยนต์ 1 ที่ มี การตรวจสอบได้หลายครั้ง
  - การตรวจสอบ 1 ครั้ง มี ที่จอดรถยนต์ได้ 1 ที่
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M



Entity Relationship Diagram



3.การแปลงโมเดลเชิงแนวคิด ER-Diagram เป็น Relational Database Model

รถยนต์

ป้ายทะเบียน	รหัสเจ้าของรถยนต์	ยี่ห้อรถยนต์	รุ่นรถยนต์	สีรถ	ประเภทรถยนต์

เจ้าของรถยนต์

รหัสเจ้าของรถยนต์	ชื่อ	นามสกุล	เบอร์โทรศัพท์	อีเมล

บัญชี

เลขบัญชี	รหัสเจ้าของรถยนต์	ชื่อธนาคาร	ชื่อบัญชี

การชำระเงิน

รหัสการชำระเงิน	รหัสการจอดรถยนต์	จำนวนเงินที่ชำระ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระ

การจองที่จอดรถยนต์

รหัสการจองที่จอดรถยนต์	รหัสที่จอดรถยนต์	ป้ายทะเบียน	วันที่จอง	เดือนที่จอง	ปีที่จอง	เวลาที่จอง	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	สถานะการจอง

การจอดรถยนต์

รหัสการจอดรถยนต์	รหัสการจองที่จอดรถยนต์	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	เวลาออก	ค่าบริการ

ที่จอดรถยนต์

รหัสที่จอดรถยนต์	ประเภทที่จอดรถยนต์	อาคารจอดรถยนต์	ชั้น	ค่าจอด / ชั่วโมง	สถานะที่จอดรถยนต์

กล้องวงจรปิด

รหัสกล้องวงจรปิด	รหัสที่จอดรถยนต์	ยี่ห้อกล้องวงจรปิด	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบล่าสุด	จำนวนวันที่เก็บได้สูงสุด	เส้นทางที่เก็บไฟล์	สถานะกล้องวงจรปิด

เจ้าหน้าที่

รหัสเจ้าหน้าที่	รหัสที่จอดรถยนต์	ช่วงเวลาทำงาน	ชื่อ	นามสกุล	ตำแหน่ง	เบอร์โทรศัพท์	เวลาเริ่มงาน	เวลาออกงาน

การตรวจสอบที่จอดรถยนต์

รหัสการตรวจสอบ	รหัสที่จอดรถยนต์	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบ	ปัญหาที่พบ	สถานะการซ่อมแซม

4.การทำบรรทัดฐาน(Normalization)

ตารางรถยนต์

ป้ายทะเบียน	รหัสเจ้าของรถยนต์	ยี่ห้อรถยนต์	รุ่นรถยนต์	สีรถ	ประเภทรถยนต์
กข-1234	Q00123	Toyota	Corolla	ขาว	เก๋ง
ขค-5678	Q00124	Toyota	Hilux	ดำ	กระบะ
งง-9101	Q00125	Ford	Ranger	น้ำเงิน	กระบะ
จจ-2345	Q00126	Nissan	Almera	เทา	เก๋ง
ฉฉ-6789	Q00127	Mazda	CX-5	แดง	SUV
ดบ-1235	Q00127	Honda	Civic	เงิน	เก๋ง
กร-5679	Q00125	Isuzu	D-Max	ขาว	กระบะ
ลน-9102	Q00125	Mitsubishi	Pajero Sport	เทา	SUV

1NF : ตารางนี้มี ป้ายทะเบียน เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว แต่มีการขึ้นต่อกันแบบ Transitive โดย ยี่ห้อรถยนต์ และ ประเภทรถยนต์ จะขึ้นอยู่กับรุ่นรถยนต์

จึงแยก Attributes ที่มีการขึ้นต่อกันแบบ Transitive มาสร้างเป็นตารางใหม่ ชื่อ ตารางรุ่นรถยนต์ โดยให้ รุ่นรถยนต์ เป็นคีย์หลัก (PK) และเก็บรุ่นรถยนต์ในตารารถยนต์ เพื่อใช้เป็น FK เชื่อมโยงกับตารางกัน

ตารางรถยนต์

ป้ายทะเบียน	รหัสเจ้าของรถยนต์	รุ่นรถยนต์	สีรถ
กข-1234	Q00123	Corolla	ขาว
ขค-5678	Q00124	Hilux	ดำ
งง-9101	Q00125	Ranger	น้ำเงิน
จจ-2345	Q00126	Almera	เทา
ฉฉ-6789	Q00127	CX-5	แดง
ดบ-1235	Q00127	Civic	เงิน
กร-5679	Q00125	D-Max	ขาว
ลน-9102	Q00125	Pajero Sport	เทา

และตารางรุ่นรถยนต์

รุ่นรถยนต์	ยี่ห้อรถยนต์	ประเภทรถยนต์
Corolla	Toyota	เก๋ง
Hilux	Toyota	กระบะ
Ranger	Ford	กระบะ
Almera	Nissan	เก๋ง
CX-5	Mazda	SUV
Civic	Honda	เก๋ง
D-Max	Isuzu	กระบะ
Pajero Sport	Mitsubishi	SUV

ตารางเจ้าของรถยนต์

รหัสเจ้าของรถยนต์	ชื่อ	นามสกุล	เบอร์โทรศัพท์	อีเมล
O00123	สมชาย	แก้วใส	8-1234-5678	somchai@gmail.com
O00124	มานะ	ทองดี	8-9876-5432	mana@gmail.com
O00125	สายฝน	วิริยะ	8-2345-6789	saifon@gmail.com
O00126	อภิชัย	บุญมี	9-0456-7890	apichai@gmail.com
O00127	วิภา	รัตนวงศ์	9-5567-8901	wipa@gmail.com

1NF : ตารางนี้มี รหัสเจ้าของรถยนต์ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางบัญชี

เลขบัญชี	รหัสเจ้าของรถยนต์	ชื่อธนาคาร	ชื่อบัญชี
1234567890	O00123	กสิกรไทย	สมชาย แก้วใส
2345678901	O00124	ไทยพาณิชย์	มานะ ทองดี
3456789012	O00125	กรุงเทพ	สายฝน วิริยะ
4567890123	O00126	กรุงไทย	อภิชัย บุญมี
5678901234	O00127	กรุงศรี	วิภา รัตนวงศ์

1NF : ตารางนี้มี เลขบัญชีตารางนี้มี ป้ายทะเบียน เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการชำระเงิน

รหัสการชำระเงิน	รหัสการจอดรถยนต์	จำนวนเงินที่ชำระ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระเงิน
P00001	A00001	30	2	1	2567	9:10:00
P00002	A00002	100	3	2	2567	11:25:00
P00003	A00003	200	3	2	2567	11:40:00
P00004	A00004	60	3	2	2567	11:10:00
P00005	A00005	50	4	2	2567	11:10:00
P00006	A00006	150	5	2	2567	11:40:00
P00007	A00007	150	5	3	2567	12:10:00
P00008	A00008	150	5	4	2567	13:25:00
P00009	A00009	100	5	4	2567	13:10:00

1NF : ตารางนี้มี รหัสการชำระเงิน เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการจองที่จอดรถยนต์

รหัสการจองที่จอดรถยนต์	รหัสที่จอดรถยนต์	ป้ายทะเบียน	วันที่จอง	เดือนที่จอง	ปีที่จอง	เวลาที่จอง	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	สถานะการจอง
B00001	P00101	กข-1234	1	1	2567	10:30:00	2	1	2567	8:00:00	สำเร็จ
B00002	P00102	ขค-5678	2	2	2567	14:00:00	3	2	2567	9:15:00	สำเร็จ
B00003	P00102	งง-9101	2	2	2567	9:45:00	3	2	2567	7:30:00	สำเร็จ
B00004	P00104	จจ-2345	2	2	2567	16:20:00	3	2	2567	10:00:00	สำเร็จ
B00005	P00105	ฉฉ-6789	2	2	2567	11:10:00	2	2	2567	8:45:00	ยกเลิก
B00006	P00103	ฉฉ-6789	3	2	2567	13:10:00	4	2	2567	10:00:00	สำเร็จ
B00007	P00103	ฉฉ-6789	4	2	2567	10:00:00	5	2	2567	8:30:00	สำเร็จ
B00008	P00107	ขค-5678	4	3	2567	11:30:00	5	3	2567	9:00:00	สำเร็จ
B00009	P00107	งง-9101	4	4	2567	13:00:00	5	4	2567	10:15:00	สำเร็จ
B00010	P00107	จจ-2345	4	4	2567	15:00:00	5	4	2567	11:00:00	สำเร็จ

1NF : ตารางนี้มี รหัสการจองที่จอดรถ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการจอดรถยนต์

รหัสการจอดรถยนต์	รหัสการจองที่จอดรถยนต์	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	เวลาออก	ค่าบริการ
A00001	B00001	2	1	2567	8:00:00	9:00:00	30
A00002	B00002	3	2	2567	9:15:00	11:15:00	100
A00003	B00003	3	2	2567	7:30:00	11:30:00	200
A00004	B00004	3	2	2567	10:00:00	11:00:00	60
A00005	B00006	4	2	2567	10:00:00	11:00:00	50
A00006	B00007	5	2	2567	8:30:00	11:30:00	150
A00007	B00008	5	3	2567	9:00:00	12:00:00	150
A00008	B00009	5	4	2567	10:15:00	13:15:00	150
A00009	B00010	5	4	2567	11:00:00	13:00:00	100

1NF : ตารางนี้มี รหัสการจอดรถยนต์ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางที่จอดรถยนต์

รหัสที่จอดรถยนต์	ประเภทที่จอดรถยนต์	อาคารจอดรถยนต์	ชั้น	ค่าจอด / ชั่วโมง	สถานะที่จอดรถยนต์
P00101	ที่จอดสำหรับผู้พิการ	1	1	30.00	พร้อมใช้งาน
P00102	ที่จอดทั่วไป	1	1	50.00	พร้อมใช้งาน
P00103	ที่จอดทั่วไป	2	1	50.00	พร้อมใช้งาน
P00104	ที่จอดรถสำหรับรถไฟฟ้า	3	2	60.00	พร้อมใช้งาน
P00105	ที่จอดรถสำหรับรถไฟฟ้า	4	3	60.00	พร้อมใช้งาน
P00106	ที่จอดทั่วไป	2	1	50.00	พร้อมใช้งาน
P00107	ที่จอดทั่วไป	2	1	50.00	พร้อมใช้งาน
P00108	ที่จอดสำหรับผู้พิการ	2	1	30.00	ชำรุด
P00109	ที่จอดสำหรับผู้พิการ	4	1	30.00	ชำรุด

1NF : ตารางนี้มี รหัสที่จอดรถยนต์ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : อยู่ในรูปแบบ 1NF แล้ว ไม่มีการขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว แต่มีการขึ้นต่อกันแบบ Transitive โดย ค่าจอด จะขึ้นอยู่กับประเภทที่จอดรถยนต์

จึงแยก Attributes ที่มีการขึ้นต่อกันแบบ Transitive มาสร้างเป็นตารางใหม่ ชื่อตารางประภทที่จอดรถยนต์ โดยให้ ประเภทที่จอดรถยนต์ เป็นคีย์หลัก (PK) และเก็บประเภทที่จอดรถยนต์ในตารางที่จอดรถยนต์ เพื่อใช้เป็น FK เชื่อมโยงกับตารางกัน

ตารางที่จอดรถยนต์

รหัสที่จอดรถยนต์	ประเภทที่จอดรถยนต์	อาคารจอดรถยนต์	ชั้น	สถานะที่จอดรถยนต์
P00101	ที่จอดสำหรับผู้พิการ	1	1	พร้อมใช้งาน
P00102	ที่จอดทั่วไป	1	1	พร้อมใช้งาน
P00103	ที่จอดทั่วไป	2	1	พร้อมใช้งาน
P00104	ที่จอดรถสำหรับรถไฟฟ้า	3	2	พร้อมใช้งาน
P00105	ที่จอดรถสำหรับรถไฟฟ้า	4	3	พร้อมใช้งาน
P00106	ที่จอดทั่วไป	2	1	พร้อมใช้งาน
P00107	ที่จอดทั่วไป	2	1	พร้อมใช้งาน
P00108	ที่จอดสำหรับผู้พิการ	2	1	ชำรุด
P00109	ที่จอดสำหรับผู้พิการ	4	1	ชำรุด

และ ตารางประเภทที่จอดรถยนต์

ประเภทที่จอดรถยนต์	ค่าจอด / ชั่วโมง
ที่จอดสำหรับผู้พิการ	30.00
ที่จอดทั่วไป	50.00
ที่จอดรถสำหรับรถไฟฟ้า	60.00

ตารางกล้องวงจรปิด

รหัสกล้องวงจรปิด	รหัสที่จอดรถยนต์	ยี่ห้อกล้องวงจรปิด	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบล่าสุด	จำนวนวันที่เก็บได้สูงสุด	เส้นทางที่เก็บไฟล์	สถานะกล้องวงจรปิด
C00001	P00101	Hikvision	11	1	2567	21:24:00	90	D:/CCTV/Camera1/	ใช้งานได้
C00002	P00102	Dahua	11	1	2567	12:48:00	90	D:/CCTV/Camera2/	ใช้งานได้
C00003	P00103	Panasonic	11	1	2567	9:43:00	90	D:/CCTV/Camera3/	ชำรุด
C00004	P00104	Panasonic	11	2	2567	15:53:00	120	D:/CCTV/Camera4/	ใช้งานได้
C00005	P00105	Sony	11	2	2567	16:02:00	90	D:/CCTV/Camera5/	ใช้งานได้
C00006	P00106	Hikvision	12	3	2567	10:30:00	90	D:/CCTV/Camera6/	ชำรุด
C00007	P00107	Hikvision	12	3	2567	11:15:00	90	D:/CCTV/Camera7/	ชำรุด
C00008	P00108	Hikvision	12	3	2567	13:00:00	120	D:/CCTV/Camera8/	ใช้งานได้
C00009	P00109	Panasonic	12	4	2567	14:45:00	90	D:/CCTV/Camera9/	ใช้งานได้

1NF : ตารางนี้มี รหัสกล้องวงจรปิด เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางเจ้าหน้าที่

รหัสเจ้าหน้าที่	รหัสที่จอดรถยนต์	ช่วงเวลาทำงาน	ชื่อ	นามสกุล	ตำแหน่ง	เบอร์โทรศัพท์	เวลาเริ่มงาน	เวลาออกงาน
N00007	P00101	ช่วงที่ 1	สมพร	ขจรศักดิ์	ยาม	6-6778-6251	8:00:00	16:00:00
N00008	P00102	ช่วงที่ 1	สมจิตร	มิตรผล	ยาม	9-9123-4545	8:00:00	16:00:00
N00001	P00103	ช่วงที่ 1	สมรักษ์	มักสุข	หัวหน้ายาม	8-8578-9012	8:00:00	16:00:00
N00003	P00104	ช่วงที่ 2	สมถวิล	นิมิตรหมาย	ยาม	6-3337-2775	16:00:00	0:00:00
N00004	P00105	ช่วงที่ 3	สมปอง	ก้องเกียรติ	ยาม	8-5654-7895	0:00:00	8:00:00
N00009	P00106	ช่วงที่ 4	สมชาย	ตั้งใจจริง	เจ้าหน้าที่ทำความสะอาด	6-2345-6789	7:00:00	15:00:00
N00010	P00103	ช่วงที่ 1	สมหญิง	แซ่ตั้ง	เจ้าหน้าที่ซ่อมบำรุง	7-7890-2345	8:00:00	16:00:00
N00011	P00104	ช่วงที่ 2	สมพงษ์	จันทร์เจริญ	พนักงานเก็บเงิน	9-5678-3456	16:00:00	0:00:00
N00012	P00109	ช่วงที่ 4	สมฤดี	เพียรดี	เจ้าหน้าที่ทำความสะอาด	5-3456-7890	7:00:00	7:00:00

1NF : ตารางนี้มี รหัสเจ้าหน้าที่ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว แต่มีการขึ้นต่อกันแบบ Transitive โดย เวลาเริ่มงานและเวลาออกงาน จะขึ้นอยู่กับช่วงเวลาที่ทำงาน

จึงแยก Attributes ที่มีการขึ้นต่อกันแบบ Transitive มาสร้างเป็นตารางใหม่ ชื่อตารางช่วงเวลาทำงาน โดยให้ ช่วงเวลาที่ทำงาน เป็นคีย์หลัก (PK) และเก็บช่วงเวลาททำงาน ในตารางเจ้าหน้าที่เพื่อใช้เป็น FK เชื่อมโยงกับตารางกัน

ตารางเจ้าหน้าที่

รหัสเจ้าหน้าที่	รหัสที่จอดรถยนต์	ช่วงเวลาทำงาน	ชื่อ	นามสกุล	ตำแหน่ง	เบอร์โทรศัพท์
N00007	P00101	ช่วงที่ 1	สมพร	ขจรศักดิ์	ยาม	6-6778-6251
N00008	P00102	ช่วงที่ 1	สมจิตร	มิตรผล	ยาม	9-9123-4545
N00001	P00103	ช่วงที่ 1	สมรักษ์	มักสุข	หัวหน้ายาม	8-8578-9012
N00003	P00104	ช่วงที่ 2	สมถวิล	นิมิตรหมาย	ยาม	6-3337-2775
N00004	P00105	ช่วงที่ 3	สมปอง	ก้องเกียรติ	ยาม	8-5654-7895
N00009	P00106	ช่วงที่ 4	สมชาย	ตั้งใจจริง	เจ้าหน้าที่ทำความสะอาด	6-2345-6789
N00010	P00103	ช่วงที่ 1	สมหญิง	แซ่ตั้ง	เจ้าหน้าที่ซ่อมบำรุง	7-7890-2345
N00011	P00104	ช่วงที่ 2	สมพงษ์	จันทร์เจริญ	พนักงานเก็บเงิน	9-5678-3456
N00012	P00109	ช่วงที่ 4	สมฤดี	เพียรดี	เจ้าหน้าที่ทำความสะอาด	5-3456-7890

และตารางช่วงเวลาทำงาน

ช่วงเวลาที่ทำงาน	เวลาเริ่มงาน	เวลาออกงาน
ช่วงที่ 1	8:00:00	16:00:00
ช่วงที่ 2	16:00:00	0:00:00
ช่วงที่ 3	0:00:00	8:00:00
ช่วงที่ 4	7:00:00	15:00:00

ตารางการตรวจสอบที่จอดรถยนต์

รหัสการตรวจสอบ	รหัสที่จอดรถยนต์	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบ	ปัญหาที่พบ	สถานะการซ่อมแซม
K00001	P00101	20	2	2567	8:00:00	ไฟไม่ติด	รอการซ่อมแซม
K00002	P00102	21	3	2567	9:15:00	สีเส้นไม่ชัด	ซ่อมเสร็จแล้ว
K00003	P00103	22	4	2567	7:30:00	NULL	NULL
K00004	P00104	23	5	2567	10:00:00	NULL	NULL
K00005	P00105	24	6	2567	8:45:00	กล้องมองไม่เห็น	กำลังซ่อมแซม
K00006	P00101	25	7	2567	12:00:00	ไฟไม่ติด	ซ่อมเสร็จแล้ว
K00007	P00101	26	7	2567	15:00:00	NULL	NULL
K00008	P00101	27	7	2567	14:30:00	กล้องบันทึกภาพไม่ได้	รอการซ่อมแซม
K00009	P00102	28	8	2567	16:00:00	ไฟไม่ติด	ซ่อมเสร็จแล้ว

1NF : ตารางนี้มี รหัสการตรวจสอบ เป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

New Relational Database Model

รถยนต์

ป้ายทะเบียน	รหัสเจ้าของรถยนต์	รุ่นรถยนต์	สีรถ

รุ่นรถยนต์

รุ่นรถยนต์	ยี่ห้อรถยนต์	ประเภทรถยนต์

เจ้าของรถยนต์

รหัสเจ้าของรถยนต์	ชื่อ	นามสกุล	เบอร์โทรศัพท์	อีเมล

บัญชี

เลขบัญชี	รหัสเจ้าของรถยนต์	ชื่อธนาคาร	ชื่อบัญชี

การชำระเงิน

รหัสการชำระเงิน	รหัสการจอดรถยนต์	จำนวนเงินที่ชำระ	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน	เวลาที่ชำระ

การจองที่จอดรถยนต์

รหัสการจองที่จอดรถยนต์	รหัสที่จอดรถยนต์	ป้ายทะเบียน	วันที่จอง	เดือนที่จอง	ปีที่จอง	เวลาที่จอง	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	สถานะการจอง

การจอดรถยนต์

รหัสการจอดของรถยนต์	รหัสการจองที่จอดรถยนต์	วันที่จอด	เดือนที่จอด	ปีที่จอด	เวลาที่จอด	เวลาออก	ค่าบริการ

ที่จอดรถยนต์

รหัสที่จอดรถยนต์	ประเภทที่จอดรถยนต์	อาคารจอดรถยนต์	ชั้น	สถานะที่จอดรถยนต์

ประเภทที่จอดรถยนต์

ประเภทที่จอดรถยนต์	ค่าจอด / ชั่วโมง

กล้องวงจรปิด

รหัสกล้องวงจรปิด	รหัสที่จอดรถยนต์	ยี่ห้อกล้องวงจรปิด	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบล่าสุด	จำนวนวันที่เก็บได้สูงสุด	เส้นทางที่เก็บไฟล์	สถานะกล้องวงจรปิด

เจ้าหน้าที่

รหัสเจ้าหน้าที่	รหัสที่จอดรถยนต์	ช่วงเวลาทำงาน	ชื่อ	นามสกุล	ตำแหน่ง	เบอร์โทรศัพท์

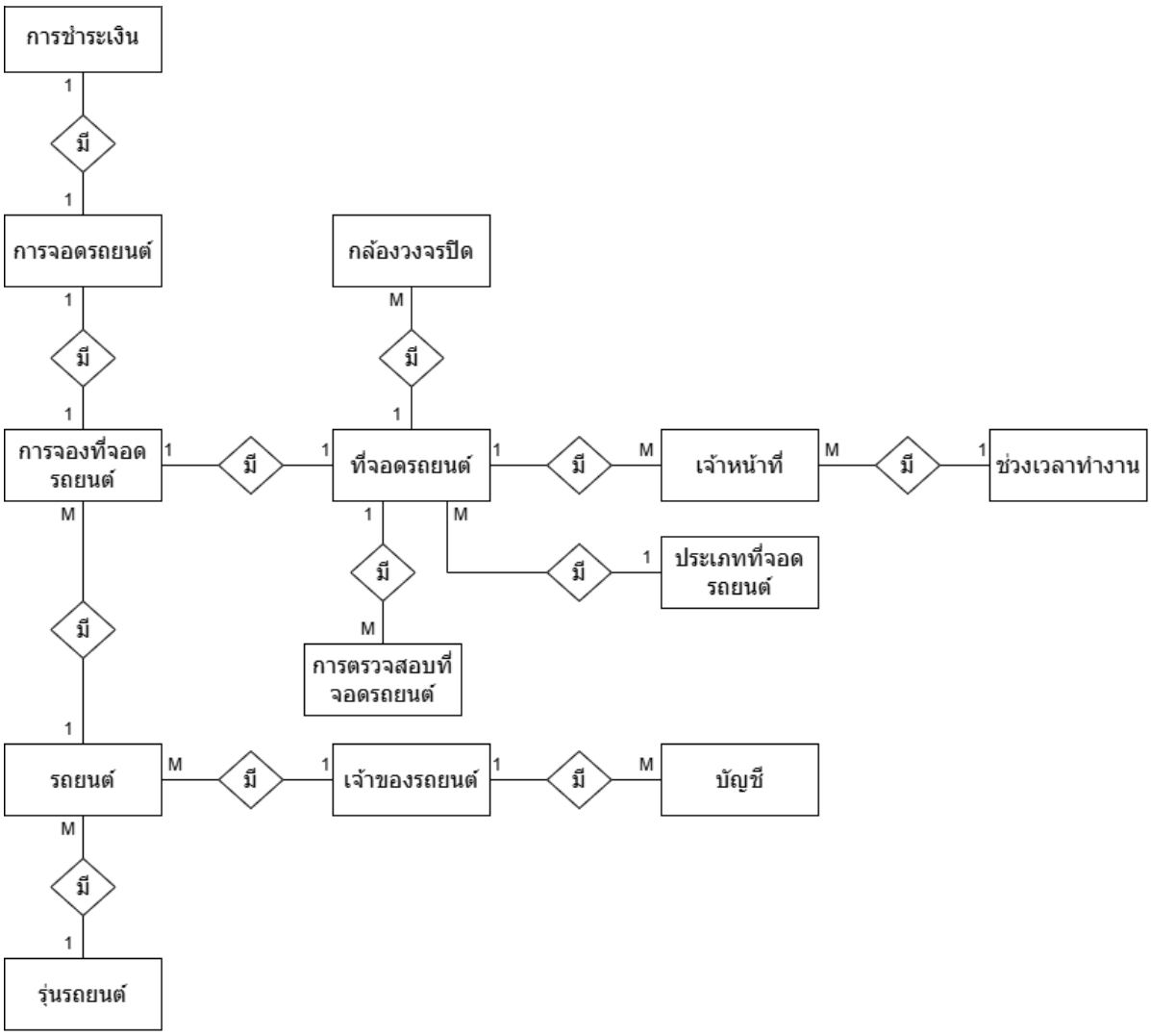
ช่วงเวลาทำงาน

ช่วงเวลาที่ทำงาน	เวลาเริ่มงาน	เวลาออกงาน

การตรวจสอบที่จอดรถยนต์

รหัสการตรวจสอบ	รหัสที่จอดรถยนต์	วันที่ตรวจสอบ	เดือนที่ตรวจสอบ	ปีที่ตรวจสอบ	เวลาที่ตรวจสอบ	ปัญหาที่พบ	สถานะการซ่อมแซม

New Entity Relationship Diagram



5.พจนานุกรมข้อมูล(Data Dictionary)

ตารางรถยนต์

Attributes	Data Type	Description
license_plate (ป้ายทะเบียน)	VARCHAR(10)	หมายเลขทะเบียนของรถยนต์
owner_id (รหัสเจ้าของรถยนต์)	CHAR(6)	รหัสที่ใช้ระบุเจ้าของรถยนต์
model_car (รุ่นรถยนต์)	VARCHAR(50)	ชื่อรุ่นรถยนต์ เช่น Corolla, Civic, Fortuner
colour_car (สีรถ)	VARCHAR(20)	ชื่อสีรถยนต์

ตารางรุ่นรถยนต์

Attributes	Data Type	Description
model (รุ่นรถยนต์)	VARCHAR(50)	ชื่อรุ่นรถยนต์ เช่น Corolla, Civic, Fortuner
brand (ยี่ห้อรถยนต์)	VARCHAR(50)	ชื่อยี่ห้อรถยนต์ เช่น Toyota, Honda, Ford
cartype (ประเภทรถยนต์)	VARCHAR(50)	ชื่อประเภทรถยนต์ เช่น รถเก๋ง, รถกระบะ, SUV

ตารางเจ้าของรถยนต์

Attributes	Data Type	Description
owner_id (รหัสเจ้าของรถยนต์)	CHAR(6)	รหัสอ้างอิงสำหรับเจ้าของรถยนต์
owner_name (ชื่อ)	VARCHAR(20)	ชื่อเจ้าของรถยนต์
owner_lastname (นามสกุล)	VARCHAR(20)	นามสกุลเจ้าของรถยนต์
owner_phonenumber (เบอร์โทรศัพท์)	CHAR(12)	หมายเลขโทรศัพท์เจ้าของรถยนต์ 9 หลัก
owner_gmail (อีเมล)	VARCHAR(100)	อีเมลเจ้าของรถยนต์ที่ติดต่อได้

ตารางบัญชี

Attributes	Data Type	Description
account_id (เลขบัญชี)	VARCHAR(20)	หมายเลขบัญชีที่ใช้ในการชำระเงิน
owner_id (รหัสเจ้าของรถยนต์)	CHAR(6)	รหัสที่ใช้ระบุเจ้าของรถยนต์ที่เป็นเจ้าของบัญชี
bankname (ชื่อธนาคาร)	VARCHAR(20)	ชื่อธนาคาร
account_name (ชื่อบัญชี)	VARCHAR(40)	ชื่อเจ้าของบัญชี

ตารางการชำระเงิน

Attributes	Data Type	Description
payment_id (รหัสการชำระเงิน)	CHAR(6)	รหัสอ้างอิงสำหรับการชำระเงิน
parking_id (รหัสการจอดรถยนต์)	CHAR(6)	รหัสที่ใช้ระบุการจอดรถ
amount (จำนวนเงินที่ชำระ)	DECIMAL(10,2)	จำนวนเงินที่ทำการชำระ ทศนิยม 2 ตำแหน่ง
pay_date (วันที่ชำระเงิน)	TINYINT	วันที่ทำการชำระเงิน
pay_month (เดือนที่ชำระเงิน)	TINYINT	เดือนที่ทำการชำระเงิน
pay_year (ปีที่ชำระเงิน)	SMALLINT	ปีที่ทำการชำระเงิน
pay_time (เวลาที่ชำระเงิน)	TIME	เวลาที่ทำการชำระเงิน

ตารางการจองที่จอดรถยนต์

Attributes	Data Type	Description
reservation_id (รหัสการจองที่จอดรถยนต์)	CHAR(6)	รหัสอ้างอิงสำหรับการจองที่จอดรถยนต์
carpark_id (รหัสที่จอดรถยนต์)	CHAR(6)	รหัสที่ใช้ระบุที่จอดรถยนต์
license_plate (ป้ายทะเบียน)	VARCHAR(10)	หมายเลขทะเบียนของรถยนต์
reservation_date (วันที่จอง)	TINYINT	วันที่จองที่จอดรถยนต์
reservation_month (เดือนที่จอง)	TINYINT	เดือนที่จองที่จอดรถยนต์
reservation_year (ปีที่จอง)	SMALLINT	ปีที่จองที่จอดรถยนต์
reservation_time (เวลาที่จอง)	TIME	เวลาที่จองที่จอดรถยนต์
park_date (วันที่จอด)	TINYINT	วันที่จอดที่จองไว้
park_month (เดือนที่จอด)	TINYINT	เดือนที่จอดที่จองไว้
park_year (ปีที่จอด)	SMALLINT	ปีที่จอดที่จองไว้
park_time (เวลาที่จอด)	TIME	เวลาที่จอดที่จองไว้
reservation_status (สถานะการจอง)	VARCHAR(20)	เก็บสถานะของการจองที่จอดรถยนต์ - สำเร็จ (Success) - ยกเลิก (Canceledl)

ตารางการจอดรถยนต์

Attributes	Data Type	Description
parking_id (รหัสการจอดรถยนต์)	CHAR(6)	รหัสอ้างอิงสำหรับการจอดของรถยนต์
reservation_id (รหัสการจองที่จอดรถยนต์)	CHAR(6)	รหัสที่ใช้ระบุการจองที่จอดรถยนต์
park_date (วันที่จอด)	TINYINT	วันที่เข้าจอดรถ
park_month (เดือนที่จอด)	TINYINT	เดือนที่เข้าจอดรถ
park_year (ปีที่จอด)	SMALLINT	ปีที่เข้าจอดรถ
come_time (เวลาที่จอด)	TIME	เวลาที่เข้าจอดรถ
out_time (เวลาออก)	TIME	เวลาที่ออกจากการจอดรถ
service_fee (ค่าบริการ)	DECIMAL(6,2)	ค่าบริการรวมของการจอดรถ

ตารางที่จอดรถยนต์

Attributes	Data Type	Description
carpark_id (รหัสที่จอดรถยนต์)	CHAR(6)	รหัสอ้างอิงสำหรับที่จอดรถยนต์
carpark_building (อาคารจอดรถยนต์)	TINYINT	หมายเลขอาคารที่จอด มีทั้งหมด 3 อาคาร
carpark_floor (ชั้น)	TINYINT	หมายเลขชั้นที่จอด มีตั้งหมด 7 ชั้น
carpark_type (ประเภทที่จอดรถยนต์)	VARCHAR(20)	ประเภทของที่จอดรถยนต์ เช่น - ที่จอดสำหรับผู้พิการ (Disabled Parking) - ที่จอดทั่วไป (General Parking) - ที่จอดรถสำหรับรถไฟฟ้า (EV Parking)
carpark_status (สถานะที่จอดรถยนต์)	VARCHAR(20)	เก็บสถานะของที่จอรถยนต์ - พร้อมใช้งาน (Ready to use) - ชำรุด (Out of Service)

ตารางประเภทที่จอดรถยนต์

Attributes	Data Type	Description
carpark_type (ประเภทที่จอดรถยนต์)	VARCHAR(50)	ประเภทของที่จอดรถยนต์ เช่น - ที่จอดสำหรับผู้พิการ (Disabled Parking) - ที่จอดทั่วไป (General Parking) - ที่จอดรถสำหรับรถไฟฟ้า (EV Parking)
carpark_price (ค่าจอด / ชั่วโมง)	DECIMAL(4,2)	ค่าจอดรถ ต่อชั่วโมง

ตารางกล้องวงจรปิด

Attributes	Data Type	Description
CCTV_id (รหัสกล้องวงจรปิด)	CHAR(6)	รหัสอ้างอิงสำหรับกล้องวงจรปิดแต่ละตัว
carpark_id (รหัสที่จอดรถยนต์)	CHAR(6)	รหัสที่ใช้ระบุที่จอดรถที่กล้องวงจรปิดมองเห็น
CCTV_ brand (ยี่ห้อกล้องวงจรปิด)	VARCHAR(50)	ชื่อยี่ห้อกล้องวงจรปิด
CCTV_ checkDay (วันที่ตรวจสอบ)	TINYINT	วันที่ทำการตรวจสอบ
CCTV_ checkMonth (เดือนที่ตรวจสอบ)	TINYINT	เดือนที่ตรวจสอบ
CCTV_ checkYear (ปีที่ตรวจสอบ)	SMALLINT	ปีที่ตรวจสอบ
CCTV_ checkTime (เวลาที่ตรวจสอบล่าสุด)	TIME	เวลาที่ตรวจสอบ
CCTV_maxDay (จำนวนวันที่เก็บได้สูงสุด)	TINYINT	จำนวนวันสูงสุดที่กล้องวงรปิดบันทึกวิดิโอก่อนถูกลบ
CCTV_path (เส้นทางที่เก็บไฟล์)	BLOB	เก็บ path ของโฟลเดอร์วิดีโอที่กล้องวงจรปิดบันทึก
CCTV_status (สถานะกล้องวงจรปิด)	VARCHAR(10)	เก็บสถานะของกล้องวงจรปิด - ใช้งานได้ (Active) - ชำรุด (Broken)

ตารางเจ้าหน้าที่

Attributes	Data Type	Description
officer_id (รหัสเจ้าหน้าที่)	CHAR(6)	รหัสอ้างอิงสำหรับเจ้าหน้าที่แต่ละคน
carpark_id (รหัสที่จอดรถยนต์)	CHAR(6)	รหัสที่จอดรถที่เจ้าหน้าที่รับผิดชอบ
officer_name (ชื่อ)	VARCHAR(20)	ชื่อของเจ้าหน้าที่
officer_lastname (นามสกุล)	VARCHAR(20)	นามสกุลของเจ้าหน้าที่
officer_position (ตำแหน่ง)	VARCHAR(50)	ตำแหน่งงานของเจ้าหน้าที่ เช่น ยาม, หัวหน้ายาม, พนักงานทำความสะอาด
officer_phonenumber (เบอร์โทรศัพท์)	CHAR(11)	หมายเลขโทรศัพท์เจ้าหน้าที่ 10 หลัก
officer_worktime (ช่วงเวลาทำงาน)	CHAR(10)	ช่วงเวลาทำงานของเจ้าหน้าที่

ตารางช่วงเวลาทำงาน

Attributes	Data Type	Description
Worktime (ช่วงเวลาทำงาน)	CHAR(10)	ช่วงเวลาทำงานของเจ้าหน้าที่
start_worktime (เวลาเริ่มงาน)	TIME	เวลาที่เจ้าหน้าเข้างาน
end_worktime (เวลาออกงาน)	TIME	เวลาที่เจ้าหน้าที่เลิกงาน

ตารางการตรวจสอบที่จอดรถยนต์

Attributes	Data Type	Description
inspection_id (รหัสการตรวจสอบ)	CHAR(6)	รหัสอ้างอิงสำหรับการตรวจสอบที่จอดรถแต่ละครั้ง
carpark_id (รหัสที่จอดรถยนต์)	CHAR(6)	รหัสของที่จอดรถที่ทำการตรวจสอบ
inspection_day (วันที่ตรวจสอบ)	TINYINT	วันที่ที่ทำการตรวจสอบ
inspection_month (เดือนที่ตรวจสอบ)	TINYINT	เดือนที่ทำการตรวจสอบ
inspection_year (ปีที่ตรวจสอบ)	SMALLINT	ปีที่ทำการตรวจสอบ
inspection_time (เวลาที่ตรวจสอบ)	TIME	เวลาที่ทำการตรวจสอบที่จอดรถยนต์
inspection_issue (ปัญหาที่พบ)	VARCHAR(100)	ปัญหาที่เจอจากการตรวจสอบที่จอดรถยนต์
inspection_Record (สถานะการซ่อมแซม)	VARCHAR(50)	สถานการณ์ซ่อมแซม - รอการซ่อมแซม (Waiting for repairs) - กำลังซ่อมแซม (Under repair) - ซ่อมเสร็จแล้ว (Repair completed)

6.การสร้างฐานข้อมูลด้วยภาษา SQL โดยใช้ MySQL Workbench

การสร้างตาราง **เจ้าของรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางเจ้าของรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ owner_id เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Owners แสดงผลของตารางที่ได้ผ่านการ Query

```sql
CREATE TABLE Owners (
    owner_id CHAR (6) PRIMARY KEY,
    owner_name VARCHAR (20) NOT NULL,
    owner_lastname VARCHAR (20) NOT NULL,
    owner_phonenumber CHAR (12)  NOT NULL,
    owner_gmail VARCHAR (100) NOT NULL
);
INSERT INTO Owners
VALUES  ('Q00123', 'Somchai', 'Keawsai', '8-1234-5678', 'somchai@gmail.com'),
        ('Q00124', 'Mana', 'Tongdee', '8-9876-5432', 'mana@gmail.com'),
        ('Q00125', 'Saifon', 'Wiriya', '8-2345-6789', 'saifon@gmail.com'),
        ('Q00126', 'Apichai', 'Boonmee', '9-0456-7890', 'apichai@gmail.com'),
        ('Q00127', 'Wipa', 'Ratanawong', '9-5567-8901', 'wipa@gmail.com');
SELECT * FROM Owners ;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Cont

owner_id	owner_name	owner_lastname	owner_phonenumber	owner_gmail
Q00123	Somchai	Keawsai	8-1234-5678	somchai@gmail.com
Q00124	Mana	Tongdee	8-9876-5432	mana@gmail.com
Q00125	Saifon	Wiriya	8-2345-6789	saifon@gmail.com
Q00126	Apichai	Boonmee	9-0456-7890	apichai@gmail.com
Q00127	Wipa	Ratanawong	9-5567-8901	wipa@gmail.com

การสร้างตาราง **รุ่นรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางรุ่นรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ model เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Models แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Models (
2      model VARCHAR(50) PRIMARY KEY,
3      brand VARCHAR(50) NOT NULL,
4      cartype VARCHAR(50) NOT NULL
5  );
6  INSERT INTO Models
7  VALUES  ('Corolla', 'Toyota', 'Car'),
8          ('Hilux', 'Toyota', 'Truck'),
9          ('Ranger', 'Ford', 'Truck'),
10         ('Almera', 'Nissan', 'Car'),
11         ('CX-5', 'Mazda', 'SUV'),
12         ('Civic', 'Honda', 'Car'),
13         ('D-Max', 'Isuzu', 'Truck'),
14         ('Pajero Sport', 'Mitsubishi', 'SUV');
15 SELECT * FROM Models;
```

Result Grid | Filter Rows: | Edit:

model	brand	cartype
Almera	Nissan	Car
Civic	Honda	Car
Corolla	Toyota	Car
CX-5	Mazda	SUV
D-Max	Isuzu	Truck
Hilux	Toyota	Truck
Pajero Sport	Mitsubishi	SUV
Ranger	Ford	Truck

การสร้างตาราง **รถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ car_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ owner_id เป็น FK และอ้างอิงจากตาราง Owners

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ model_car เป็น FK และอ้างอิงจากตาราง Models

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM cars แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Cars (
2      license_plate VARCHAR(10) PRIMARY KEY,
3      owner_id CHAR(6),
4      model_car VARCHAR(50) NOT NULL,
5      colour_car VARCHAR(20) NOT NULL,
6      FOREIGN KEY (owner_id) REFERENCES Owners(owner_id),
7      FOREIGN KEY (model_car) REFERENCES Models (model)
8  );
9  INSERT INTO Cars
10 VALUES  ('กข-1234', 'Q00123', 'Corolla', 'ขาว'),
11         ('ขค-5678', 'Q00124', 'Hilux', 'ดำ'),
12         ('งง-9101', 'Q00125', 'Ranger', 'น้ำเงิน'),
13         ('จจ-2345', 'Q00126', 'Almera', 'เทา'),
14         ('ฉฉ-6789', 'Q00127', 'CX-5', 'แดง'),
15         ('ดบ-1235', 'Q00127', 'Civic', 'เงิน'),
16         ('กร-5679', 'Q00125', 'D-Max', 'ขาว'),
17         ('ลน-9102', 'Q00125', 'Pajero Sport', 'เทา');
18 SELECT * FROM Cars;
```

Result Grid | Filter Rows: | Edit: | Export/Impo

license_plate	owner_id	model_car	colour_car
กข-1234	Q00123	Corolla	White
กร-5679	Q00125	D-Max	ขาว
ขค-5678	Q00124	Hilux	Black
งง-9101	Q00125	Ranger	Blue
จจ-2345	Q00126	Almera	Grey
ฉฉ-6789	Q00127	CX-5	Red
ดบ-1235	Q00127	Civic	เงิน
ลน-9102	Q00125	Pajero Sport	เทา

การสร้างตาราง **บัญชี** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางบัญชีขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ account_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ owner_id เป็น FK และอ้างอิงจากตาราง Owners

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Accounts แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Accounts (
2      account_id VARCHAR(20) PRIMARY KEY,
3      owner_id CHAR(6) NOT NULL,
4      bankname VARCHAR(40) NOT NULL,
5      account_name VARCHAR(40) NOT NULL,
6      FOREIGN KEY (owner_id) REFERENCES Owners(owner_id)
7  );
8  INSERT INTO Accounts
9  VALUES  ('1234567890', 'Q00123', 'Kasikorn Bank', 'Somchai Keawsai'),
10         ('2345678901', 'Q00124', 'Siam Commercial Bank', 'Mana Tongdee'),
11         ('3456789012', 'Q00125', 'Bangkok Bank', 'Saifon Wiriya'),
12         ('4567890123', 'Q00126', 'Krung Thai Bank', 'Apichai Boonmee'),
13         ('5678901234', 'Q00127', 'Bank of Ayudhya', 'Wipa Ratanawong');
14
15 SELECT * FROM Accounts;
16
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Ce

account_id	owner_id	bankname	account_name
1234567890	Q00123	Kasikorn Bank	Somchai Keawsai
2345678901	Q00124	Siam Commercial Bank	Mana Tongdee
3456789012	Q00125	Bangkok Bank	Saifon Wiriya
4567890123	Q00126	Krung Thai Bank	Apichai Boonmee
5678901234	Q00127	Bank of Ayudhya	Wipa Ratanawong

การสร้างตาราง **ประเภทที่จอดรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างประเภทที่จอดรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ carpark_type เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM carpark_type แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE carpark_type(
2      carpark_type VARCHAR(20) PRIMARY KEY,
3      carpark_price DECIMAL(4,2) NOT NULL
4  );
5  INSERT INTO carpark_type
6  VALUES  ('Disabled Parking', 30.00),
7          ('General Parking', 50.00),
8          ('EV Parking', 60.00);
9
10 SELECT * FROM carpark_type;
```

Result Grid | Filter Rows: | Edit: | Expor

	carpark_type	carpark_price
▶	Disabled Parking	30.00
	EV Parking	60.00
	General Parking	50.00

การสร้างตาราง **ที่จอดรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางที่จอดรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ carpark_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ carpark_type เป็น FK และอ้างอิงจากตาราง carpark_type

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM car_parking แสดงผลของตารางที่ได้ผ่านการ Query

```
1   CREATE TABLE car_parking(
2       carpark_id CHAR(6) PRIMARY KEY,
3       carpark_type VARCHAR(20) NOT NULL,
4       carpark_building TINYINT NOT NULL,
5       carpark_floor TINYINT NOT NULL,
6       carpark_status VARCHAR(20) NOT NULL,
7       FOREIGN KEY (carpark_type) REFERENCES carpark_type (carpark_type)
8   );
9   INSERT INTO car_parking
10  VALUES  ('P00101', 'Disabled Parking', 1, 1, 'Ready To Use'),
11          ('P00102', 'General Parking', 1, 1, 'Ready To Use'),
12          ('P00103', 'General Parking', 2, 1, 'Ready To Use'),
13          ('P00104', 'EV Parking', 3, 2, 'Ready To Use'),
14          ('P00105', 'EV Parking', 4, 3, 'Ready To Use'),
15          ('P00106', 'General Parking', 2, 1, 'Ready To Use'),
16          ('P00107', 'General Parking', 2, 1, 'Ready To Use'),
17          ('P00108', 'Disabled Parking', 2, 1, 'Out of Service'),
18          ('P00109', 'Disabled Parking', 4, 1, 'Out of Service');
```

Result Grid | Filter Rows: | Edit: | Export/Import:

carpark_id	carpark_type	carpark_building	carpark_floor	carpark_status
P00101	Disabled Parking	1	1	Ready To Use
P00102	General Parking	1	1	Ready To Use
P00103	General Parking	2	1	Ready To Use
P00104	EV Parking	3	2	Ready To Use
P00105	EV Parking	4	3	Ready To Use
P00106	General Parking	2	1	Ready To Use
P00107	General Parking	2	1	Ready To Use
P00108	Disabled Parking	2	1	Out of Service
P00109	Disabled Parking	4	1	Out of Service

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางที่จอดรถยนต์
ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_carpark_building
CHECK อาคารที่จอดรถยนต์ (carpark_building) ไม่เกิน 4 อาคาร

```
ALTER TABLE car_parking
ADD CONSTRAINT chk_carpark_building
CHECK (carpark_building <= 4);
```

ทดลองใส่ค่า อาคารที่จอดรถยนต์ เป็น 5 จะไม่สามารถทำได้เนื่องจากอาคารที่จอดรถยนต์ ต้องน้องกว่าเท่ากับ 4

```
INSERT INTO car_parking
VALUES ('P00110', 'Disabled Parking', 5, 1, 'Ready to use ');
```

Output

Action Output

#	Time	Action	Message
1	17:24:44	INSERT INTO car_parking VALUES ('P00110', 'Disabled Parking', 5, 1, 'Ready to use ')	Error Code: 3819. Check constraint 'chk_carpark_building' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางที่จอดรถยนต์
ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_carpark_floor
CHECK ชั้นที่จอดรถยนต์ (carpark_floor) ไม่เกิน 7 ชั้น

```
ALTER TABLE car_parking
ADD CONSTRAINT chk_carpark_floor
CHECK (carpark_floor <= 7);
```

ทดลองใส่ค่า ชั้นที่จอดรถยนต์ เป็น 10 จะไม่สามารถทำได้เนื่องจากชั้นที่จอดรถยนต์จะต้องไม่เกิน 7

```
INSERT INTO car_parking
VALUES ('P00110', 'Disabled Parking', 4, 10, 'Ready to use');
```

Output

Action Output

#	Time	Action	Message
1	17:26:01	INSERT INTO car_parking VALUES ('P00110', 'Disabled Parking', 4, 10, 'Ready to use')	Error Code: 3819. Check constraint 'chk_carpark_floor' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางที่จอดรถยนต์
ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_carpark_status
CHECK สถานะที่จอดรถยนต์ (carpark_status) มี 2 สถานะ คือ Ready to use และ Out of Service

```
ALTER TABLE car_parking
ADD CONSTRAINT chk_carpark_status
CHECK (carpark_status IN ('Ready To Use', 'Out of Service'));
```

ทดลองใส่สถานะ Broken จะไม่สามารถทำได้เนื่องจากค่าสถานะเป็นได้แค่ 2 สถานะคือ Ready to use และ Out of Service เท่านั้น

```
1   INSERT INTO car_parking
2   VALUES ('P00110', 'Disabled Parking', 4, 2, 'Broken');
```

Output

Action Output

#	Time	Action	Message
1	17:27:10	INSERT INTO car_parking VALUES ('P00110', 'Disabled Parking', 4, 2, 'Broken')	Error Code: 3819. Check constraint 'chk_carpark_status' is violated.

การสร้างตาราง **การจองที่จอดรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการจองที่จอดรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ reservation_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ carpark_id เป็น FK และอ้างอิงจากตาราง car_parking

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ license_plate เป็น FK และอ้างอิงจากตาราง cars

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM ParkingReservation แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE ParkingReservation (
2      reservation_id CHAR(6) PRIMARY KEY,
3      carpark_id CHAR(6) NOT NULL,
4      license_plate VARCHAR(10) NOT NULL,
5      reservation_date TINYINT NOT NULL,
6      reservation_month TINYINT NOT NULL,
7      reservation_year SMALLINT NOT NULL,
8      reservation_time TIME NOT NULL,
9      park_date TINYINT NOT NULL,
10     park_month TINYINT NOT NULL,
11     park_year SMALLINT NOT NULL,
12     park_time TIME NOT NULL,
13     reservation_status VARCHAR(20) NOT NULL,
14     FOREIGN KEY (carpark_id) REFERENCES car_parking(carpark_id),
15     FOREIGN KEY (license_plate) REFERENCES cars(license_plate)
16 );
17  INSERT INTO ParkingReservation
18  VALUES  ('B00001', 'P00101', 'กข-1234', 1, 1, 2567, '10:30:00', 2, 1, 2567, '08:00:00', 'Success'),
19          ('B00002', 'P00102', 'ขค-5678', 2, 2, 2567, '14:00:00', 3, 2, 2567, '09:15:00', 'Success'),
20          ('B00003', 'P00102', 'งจ-9101', 2, 2, 2567, '09:45:00', 3, 2, 2567, '07:30:00', 'Success'),
21          ('B00004', 'P00104', 'ฉช-2345', 2, 2, 2567, '16:20:00', 3, 2, 2567, '10:00:00', 'Success'),
22          ('B00005', 'P00105', 'ฉฉ-6789', 2, 2, 2567, '11:10:00', 2, 2, 2567, '11:45:00', 'Cancelled'),
23          ('B00006', 'P00103', 'ฉฉ-6789', 3, 2, 2567, '13:10:00', 4, 2, 2567, '10:00:00', 'Success'),
24          ('B00007', 'P00103', 'ฉฉ-6789', 4, 2, 2567, '10:00:00', 5, 2, 2567, '08:30:00', 'Success'),
25          ('B00008', 'P00107', 'ขค-5678', 4, 3, 2567, '11:30:00', 5, 3, 2567, '09:00:00', 'Success'),
26          ('B00009', 'P00107', 'งจ-9101', 4, 4, 2567, '13:00:00', 5, 4, 2567, '10:15:00', 'Success'),
27          ('B00010', 'P00107', 'ฉช-2345', 4, 4, 2567, '15:00:00', 5, 4, 2567, '11:00:00', 'Success');
28  SELECT * FROM ParkingReservation;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

reservation_id	carpark_id	license_plate	reservation_date	reservation_month	reservation_year	reservation_time	park_date	park_month	park_year	park_time	reservation_statu
B00001	P00101	กข-1234	1	1	2567	10:30:00	2	1	2567	08:00:00	Success
B00002	P00102	ขค-5678	2	2	2567	14:00:00	3	2	2567	09:15:00	Success
B00003	P00102	งจ-9101	2	2	2567	09:45:00	3	2	2567	07:30:00	Success
B00004	P00104	ฉช-2345	2	2	2567	16:20:00	3	2	2567	10:00:00	Success
B00005	P00105	ฉฉ-6789	2	2	2567	11:10:00	2	2	2567	11:45:00	Cancelled
B00006	P00103	ฉฉ-6789	3	2	2567	13:10:00	4	2	2567	10:00:00	Success
B00007	P00103	ฉฉ-6789	4	2	2567	10:00:00	5	2	2567	08:30:00	Success
B00008	P00107	ขค-5678	4	3	2567	11:30:00	5	3	2567	09:00:00	Success
B00009	P00107	งจ-9101	4	4	2567	13:00:00	5	4	2567	10:15:00	Success
B00010	P00107	ฉช-2345	4	4	2567	15:00:00	5	4	2567	11:00:00	Success

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจองที่จอดรถยนต์

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_parkingreservation_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าปีที่ทำการจองที่จอดรถยนต์ (reservation_year) และ ปีที่จะทำการจอด (park_year) อยู่ในช่วง 2565 ถึง 2568

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ทำการจองที่จอดรถยนต์ (reservation_month) และ เดือนที่จะทำการจอด (park_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการจองที่จอดรถยนต์ (reservation_date) และ วันที่จะทำการจอด (park_date) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่ทำการตรวจสอบ ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่ทำการตรวจสอบ (inspection_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ทำการตรวจสอบไม่เกิน 29

- กำหนดเงื่อนไขที่มูลในตาราง ให้การทำการจองที่จอดรถยนต์ เกิดขึ้นก่อน การจอดรถยต์ แบ่งเป็น 4 แบบ

1. ตรวจสอบว่าปีที่ทำการจองที่จอดรถยนต์ น้อยกว่า ปีที่จะทำการจอด หรือไม่
2. ตรวจสอบว่าปีที่ทำการจองที่จอดรถยนต์ เท่ากับ ปีที่จะทำการจอด และ เดือนที่ทำการจองที่จอดรถยนต์ น้อยกว่า เดือนที่จะทำการจอด หรือไม่
3. ตรวจสอบว่าปีที่ทำการจองที่จอดรถยนต์ เท่ากับ ปีที่จะทำการจอดรถยนต์ และ เดือนที่ทำการจองที่จอดรถยนต์ เท่ากับ เดือนที่จะทำการจอดรถยนต์ และ วันที่ทำการจองที่จอดรถยนต์ น้อยกว่า วันที่จะทำการจอดรถยนต์ หรือไม่
4. ตรวจสอบว่าปีที่ทำการจองที่จอดรถยนต์ เท่ากับ ปีที่จะทำการจอดรถยนต์ และ เดือนที่ทำการจองที่จอดรถยนต์ เท่ากับ เดือนที่จะทำการจอดรถยนต์ และ วันที่ทำการจองที่จอดรถยนต์ เท่ากับ วันที่จะทำการจอดรถยนต์ และ เวลาที่ทำการจองที่จอดรถยนต์ น้อยกว่า เวลาที่จะทำการจอดรถยนต์ หรือไม่

```sql
ALTER TABLE parkingreservation
ADD CONSTRAINT chk_parkingreservation_date
CHECK (
    reservation_year BETWEEN 2565 AND 2568 AND
    reservation_month BETWEEN 1 AND 12 AND
    reservation_date BETWEEN 1 AND 31 AND
    (
        (reservation_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (reservation_month IN (4, 6, 9, 11) AND reservation_date <= 30) OR
        (reservation_month = 2 AND reservation_date <= 29)
    ) AND
    park_year BETWEEN 2565 AND 2568 AND
    park_month BETWEEN 1 AND 12 AND
    park_date BETWEEN 1 AND 31 AND
    (
        (park_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (park_month IN (4, 6, 9, 11) AND park_date <= 30) OR
        (park_month = 2 AND park_date <= 29)
    ) AND
    (
        (reservation_year < park_year) OR
        (reservation_year = park_year AND reservation_month < park_month) OR
        (reservation_year = park_year AND reservation_month = park_month AND reservation_date < park_date) OR
        (reservation_year = park_year AND reservation_month = park_month AND reservation_date = park_date AND reservation_time < park_time)
    )
);
```

ทดลองใส่ค่าวันเดือนปีที่ทำการจอง วันที่ 23 เดือน 2 ปี 2568 เวลา 12.00 น. และ วันที่จะทำการจอด วันที่ 23 เดือน 2 ปี 2568 เวลา 08.00 น. จะไม่สามารถทำได้เนื่องจาก เวลาที่ทำการจองจะต้องเกิดขึ้นก่อนเวลาที่จอด

```
1 •  INSERT INTO ParkingReservation
2    VALUES  ('B00011', 'P00107', 'ฮฮ-2345', 23, 2, 2568, '12:00:00', 23, 2, 2568, '08:00:00', 'Success');
```

Output

Action Output

#	Time	Action	Message
1	17:49:52	INSERT INTO ParkingReservation VALUES ('B00011', 'P00107', 'ฮฮ-2345', 23, 2, 2568, '12:00:00', 23, 2, 2568, '08:00:0...	Error Code: 3819. Check constraint 'chk_parkingreservation_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจองที่จอดรถยนต์

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_parkingreservation_status

CHECK สถานะการจองที่จอดรถยนต์ (reservation_status) มี 2 สถานะ คือ Success และ Cancelled

```
1 •  ALTER TABLE parkingreservation
2    ADD CONSTRAINT chk_parkingreservation_status
3    CHECK (reservation_status IN ('Success', 'Cancelled'));
```

ทดลองใส่สถานะ In progress ไม่สามารถทำได้เนื่องจาก ค่าสถานะเป็นได้แค่ 2 สถานะคือ Success และ Cancelled

```
1 •  INSERT INTO ParkingReservation
2    VALUES  ('B00011', 'P00107', 'ฮฮ-2345', 4, 4, 2567, '15:00:00', 5, 4, 2567, '11:00:00', 'In progress');
```

Output

Action Output

#	Time	Action	Message
1	17:48:26	INSERT INTO ParkingReservation VALUES ('B00011', 'P00107', 'ฮฮ-2345', 4, 4, 2567, '15:00:00',...	Error Code: 3819. Check constraint 'chk_parkingreservation_status' is violated.

การสร้างตาราง **การจอดรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการจอดรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ parking_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ reservation_id เป็น FK และอ้างอิงจากตาราง parkingreservation

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Parking แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Parking (
    parking_id CHAR(6) PRIMARY KEY,
    reservation_id CHAR(6) NOT NULL,
    park_date TINYINT NOT NULL,
    park_month TINYINT NOT NULL,
    park_year SMALLINT NOT NULL,
    come_time TIME NOT NULL,
    out_time TIME NOT NULL,
    service_fee DECIMAL(6,2),
    FOREIGN KEY (reservation_id) REFERENCES parkingreservation(reservation_id)
);
INSERT INTO Parking
VALUES  ('A00001', 'B00001', 2, 1, 2567, '08:00:00', '09:00:00', 30),
        ('A00002', 'B00002', 3, 2, 2567, '09:15:00', '11:15:00', 100),
        ('A00003', 'B00003', 3, 2, 2567, '07:30:00', '11:30:00', 200),
        ('A00004', 'B00004', 3, 2, 2567, '10:00:00', '11:00:00', 60),
        ('A00005', 'B00006', 4, 2, 2567, '10:00:00', '11:00:00', 50),
        ('A00006', 'B00007', 5, 2, 2567, '08:30:00', '11:30:00', 150),
        ('A00007', 'B00008', 5, 3, 2567, '09:00:00', '12:00:00', 150),
        ('A00008', 'B00009', 5, 4, 2567, '10:15:00', '13:15:00', 150),
        ('A00009', 'B00010', 5, 4, 2567, '11:00:00', '13:00:00', 100);
SELECT * FROM Parking;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Co

parking_id	reservation_id	park_date	park_month	park_year	come_time	out_time	service_fee
A00001	B00001	2	1	2567	08:00:00	09:00:00	30.00
A00002	B00002	3	2	2567	09:15:00	11:15:00	100.00
A00003	B00003	3	2	2567	07:30:00	11:30:00	200.00
A00004	B00004	3	2	2567	10:00:00	11:00:00	60.00
A00005	B00006	4	2	2567	10:00:00	11:00:00	50.00
A00006	B00007	5	2	2567	08:30:00	11:30:00	150.00
A00007	B00008	5	3	2567	09:00:00	12:00:00	150.00
A00008	B00009	5	4	2567	10:15:00	13:15:00	150.00
A00009	B00010	5	4	2567	11:00:00	13:00:00	100.00

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจอดรถยนต์

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_parking_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าปีที่ทำการจอดรถยนต์ (park_year) อยู่ในช่วง 2565 ถึง 2568
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ทำการจอดรถยนต์ (park_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการจอดรถยนต์ (park_date) อยู่ในช่วง 1 ถึง 31 หรือไม่ หากวันที่ทำการตรวจสอบ ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ
    1. ตรวจสอบว่าเป็นเดือนที่ทำการจอดรถยนต์ (park_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
    2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
    3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ทำการตรวจสอบไม่เกิน 29
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเวลาที่เข้าจอด (come_time) น้อยกว่า เวลาที่ออก (out_time)

```
ALTER TABLE parking
ADD CONSTRAINT chk_parking_date
CHECK (
      park_year BETWEEN 2565 AND 2568 AND
      park_month BETWEEN 1 AND 12 AND
      park_date BETWEEN 1 AND 31 AND
     (
         (park_month IN (1, 3, 5, 7, 8, 10, 12)) OR
         (park_month IN (4, 6, 9, 11) AND park_date <= 30) OR
         (park_month = 2 AND park_date <= 29)
     ) AND
     come_time < out_time
);
```

ทดลองใส่ค่าเวลาที่เข้าจอดมากกว่าเวลาที่ออก จะไม่ได้สามารถทำได้เพราะว่าที่ออกจะต้องเกิดขึ้นหลังเวลาที่เข้าจอด

```
INSERT INTO Parking
VALUES  ('A00010', 'B00011', 5, 3, 2567, '13:00:00', '11:00:00', 100);
```

Output

Action Output

#	Time	Action	Message
1	17:57:49	INSERT INTO Parking VALUES ('A00010', 'B00011', 5, 3, 2567, '13:00:00', '11:00:00', 100)	Error Code: 3819. Check constraint 'chk_parking_date' is violated.

การสร้างตาราง **การชำระเงิน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการชำระเงินขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ payment_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ parking _id เป็น FK และอ้างอิงจากตาราง Parking

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Payment แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Payment (
2      payment_id CHAR(6) PRIMARY KEY,
3      parking_id CHAR(6) NOT NULL,
4      amount DECIMAL(10,2) NOT NULL,
5      pay_date TINYINT NOT NULL,
6      pay_month TINYINT NOT NULL,
7      pay_year SMALLINT NOT NULL,
8      pay_time TIME NOT NULL,
9      FOREIGN KEY (parking_id) REFERENCES Parking(parking_id)
10 );
11 INSERT INTO Payment
12 VALUES  ('P00001', 'A00001', 30, 2, 1, 2567, '09:10:00'),
13         ('P00002', 'A00002', 100, 3, 2, 2567, '11:25:00'),
14         ('P00003', 'A00003', 200, 3, 2, 2567, '11:40:00'),
15         ('P00004', 'A00004', 60, 3, 2, 2567, '11:10:00'),
16         ('P00005', 'A00005', 50, 4, 2, 2567, '11:10:00'),
17         ('P00006', 'A00006', 150, 5, 2, 2567, '11:40:00'),
18         ('P00007', 'A00007', 150, 5, 3, 2567, '12:10:00'),
19         ('P00008', 'A00008', 150, 5, 4, 2567, '13:25:00'),
20         ('P00009', 'A00009', 100, 5, 4, 2567, '13:10:00');
21 SELECT * FROM Payment;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

payment_id	parking_id	amount	pay_date	pay_month	pay_year	pay_time
P00001	A00001	30.00	2	1	2567	09:10:00
P00002	A00002	100.00	3	2	2567	11:25:00
P00003	A00003	200.00	3	2	2567	11:40:00
P00004	A00004	60.00	3	2	2567	11:10:00
P00005	A00005	50.00	4	2	2567	11:10:00
P00006	A00006	150.00	5	2	2567	11:40:00
P00007	A00007	150.00	5	3	2567	12:10:00
P00008	A00008	150.00	5	4	2567	13:25:00
P00009	A00009	100.00	5	4	2567	13:10:00

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_payment_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าปีที่ทำการชำระเงิน (pay_year) อยู่ในช่วง 2565 ถึง 2568
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ทำการชำระเงิน (pay_month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการชำระเงิน (pay_date) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่ทำการชำระเงิน ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่ทำการชำระเงิน (pay_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ทำการชำระเงิน ไม่เกิน 29

```sql
ALTER TABLE payment
ADD CONSTRAINT chk_payment_date
CHECK (
    pay_year BETWEEN 2565 AND 2568 AND
    pay_month BETWEEN 1 AND 12 AND
    pay_date BETWEEN 1 AND 31 AND
    (
        (pay_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (pay_month IN (4, 6, 9, 11) AND pay_date <= 30) OR
        (pay_month = 2 AND pay_date <= 29)
    )
);
```

ทดลองใส่ค่าเดือนที่ไม่ตรงตามข้อกำหนด ทดลองใส่ วันที่ 20 เดือน 10 ปี 2569 ไม่สามารถทำได้เนื่องจากจะสามารถบันทึกข้อมูลในช่วงปี พ.ศ. 2565-2568

```sql
INSERT INTO Payment
VALUES  ('P00010', 'A00010', 100, 20, 10, 2569, '13:10:00');
SELECT * FROM Payment;
```

Output

Action Output

#	Time	Action	Message
1	18:16:08	INSERT INTO Payment VALUES ('P00010', 'A00010', 100, 20, 10, 2569, '13:10:00')	Error Code: 3819. Check constraint 'chk_payment_date' is violated.

การสร้างตาราง **กล้องวงจรปิด** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางกล้องวงจรปิดขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ CCTV_id เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ carpark_id เป็น FK และอ้างอิงจากตาราง car_parking

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM CCTV แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE CCTV(
2      CCTV_id CHAR(6) PRIMARY KEY,
3      carpark_id CHAR(6) NOT NULL,
4      CCTV_brand VARCHAR(50) NOT NULL,
5      CCTV_checkDay TINYINT NOT NULL,
6      CCTV_checkMonth TINYINT NOT NULL,
7      CCTV_checkYear SMALLINT NOT NULL,
8      CCTV_checkTime TIME NOT NULL,
9      CCTV_maxDay TINYINT NOT NULL,
10     CCTV_path BLOB NOT NULL,
11     CCTV_status VARCHAR(10) NOT NULL,
12     FOREIGN KEY (carpark_id) REFERENCES car_parking (carpark_id)
13 );
14 INSERT INTO CCTV
15 VALUES  ('C00001', 'P00101', 'Hikvision', 11, 1, 2567, '21:24:00', 90, 'D:/CCTV/Camera1/', 'Active'),
16         ('C00002', 'P00102', 'Dahua', 11, 1, 2567, '12:48:00', 90, 'D:/CCTV/Camera2/', 'Active'),
17         ('C00003', 'P00103', 'Bosch', 11, 1, 2567, '09:43:00', 90, 'D:/CCTV/Camera3/', 'Broken'),
18         ('C00004', 'P00104', 'Panasonic', 11, 2, 2567, '15:53:00', 90, 'D:/CCTV/Camera4/', 'Active'),
19         ('C00005', 'P00105', 'Sony', 11, 2, 2567, '16:02:00', 90, 'D:/CCTV/Camera5/', 'Active'),
20         ('C00006', 'P00106', 'Hikvision', 12, 3, 2567, '10:30:00', 90, 'D:/CCTV/Camera6/', 'Broken'),
21         ('C00007', 'P00107', 'Hikvision', 12, 3, 2567, '11:15:00', 90, 'D:/CCTV/Camera7/', 'Broken'),
22         ('C00008', 'P00108', 'Hikvision', 12, 3, 2567, '13:00:00', 120, 'D:/CCTV/Camera8/', 'Active'),
23         ('C00009', 'P00109', 'Panasonic', 12, 4, 2567, '14:45:00', 90, 'D:/CCTV/Camera9/', 'Active');
24 SELECT * FROM CCTV;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CCTV_id	carpark_id	CCTV_brand	CCTV_checkDay	CCTV_checkMonth	CCTV_checkYear	CCTV_checkTime	CCTV_maxDay	CCTV_path	CCTV_status
C00001	P00101	Hikvision	11	1	2567	21:24:00	90	BLOB	Active
C00002	P00102	Dahua	11	1	2567	12:48:00	90	BLOB	Active
C00003	P00103	Bosch	11	1	2567	09:43:00	90	BLOB	Broken
C00004	P00104	Panasonic	11	2	2567	15:53:00	90	BLOB	Active
C00005	P00105	Sony	11	2	2567	16:02:00	90	BLOB	Active
C00006	P00106	Hikvision	12	3	2567	10:30:00	90	BLOB	Broken
C00007	P00107	Hikvision	12	3	2567	11:15:00	90	BLOB	Broken
C00008	P00108	Hikvision	12	3	2567	13:00:00	120	BLOB	Active
C00009	P00109	Panasonic	12	4	2567	14:45:00	90	BLOB	Active

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางกล้องวงจรปิด

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_CCTV_maxDay

CHECK จำนวนวันสูงสุดที่กล้องวงรปิดบันทึกวิดิโอได้ (CCTV_maxDay) ต้องไม่น้อยกว่า 90 วัน

```
1  ALTER TABLE cctv
2  ADD CONSTRAINT chk_CCTV_maxDay
3  CHECK (CCTV_maxDay >= 90);
```

ทดลองใส่ค่า CCTV_maxDay เป็น 50 วัน จะไม่สามารถทำได้เนื่องจากกำหนดจำนวนวันที่บันทึกวิดิโอต้องไม่ต่ำกว่า 90 วัน

```
INSERT INTO CCTV
VALUES ('C00010', 'P00109', 'Panasonic', 12, 4, 2567, '14:45:00', 50, 'D:/CCTV/Camera9/', 'Active');
```

Output

Action Output

#	Time	Action	Message
1	18:02:37	INSERT INTO CCTV VALUES ('C00010', 'P00109', 'Panasonic', 12, 4, 2567, '14:45:00', 50, 'D:/CCTV/Camera9/', 'Active')	Error Code: 3819. Check constraint 'chk_CCTV_maxDay' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางกล้องวงจรปิด

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_cctv_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าปีที่ทำการตรวจสอบกล้องวงจรปิดล่าสุด (CCTV_checkYear) อยู่ในช่วง 2565 ถึง 2568
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ทำการตรวจสอบกล้องวงจรปิดล่าสุด (CCTV_checkMonth) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการตรวจสอบกล้องวงจรปิดล่าสุด (CCTV_checkDay) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่ทำการตรวจสอบกล้องวงจรปิดล่าสุด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่ทำการตรวจสอบกล้องวงจรปิดล่าสุด (CCTV_checkMonth) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ทำการตรวจสอบกล้องวงจรปิดล่าสุด ไม่เกิน 29

```
ALTER TABLE cctv
ADD CONSTRAINT chk_cctv_date
CHECK (
    CCTV_checkYear BETWEEN 2565 AND 2568 AND
    CCTV_checkMonth BETWEEN 1 AND 12 AND
    CCTV_checkDay BETWEEN 1 AND 31 AND
    (
        (CCTV_checkMonth IN (1, 3, 5, 7, 8, 10, 12)) OR
        (CCTV_checkMonth IN (4, 6, 9, 11) AND CCTV_checkDay <= 30) OR
        (CCTV_checkMonth = 2 AND CCTV_checkDay <= 29)
    )
);
```

ทดลองใส่ค่าเดือนที่ไม่ตรงตามข้อกำหนด ทดลองใส่ วันที่ 3 เดือน 15 ปี 2565 ไม่สามารถทำได้ เนื่องจากเดือนจะต้องเป็น 1-12 เท่านั้น

```
1 •  INSERT INTO CCTV
2    VALUES ('C00010', 'P00109', 'Panasonic', 13, 15, 2567, '14:45:00', 90, 'D:/CCTV/Camera9/', 'Active');
```

Output

Action Output

#	Time	Action	Message
1	18:03:29	INSERT INTO CCTV VALUES ('C00010', 'P00109', 'Panasonic', 13, 15, 2567, '14:45:00', 90, 'D:/CCTV/Camera9/', 'Acti...	Error Code: 3819. Check constraint 'chk_cctv_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางกล้องวงจรปิด

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_cctv_status

CHECK สถานะกล้องวงจรปิด (cctv_status) มี 2 สถานะ คือ Active และ Broken

```
1 •  ALTER TABLE  cctv
2    ADD CONSTRAINT chk_cctv_status
3    CHECK (cctv_status IN ('Active', 'Broken'));
```

ทดลองใส่สถานะ New จะไม่สามารถทำได้ เนื่องจากค่าสถานะเป็นได้แค่ 2 สถานะคือ Active และ Broken เท่านั้น

```
1 •  INSERT INTO CCTV
2    VALUES ('C00010', 'P00109', 'Panasonic', 13, 5, 2567, '14:45:00', 90, 'D:/CCTV/Camera10/', 'New');
3
```

Output

Action Output

#	Time	Action	Message
1	18:08:13	INSERT INTO CCTV VALUES ('C00010', 'P00109', 'Panasonic', 13, 5, 2567, '14:45:00', 90, 'D:/CCTV/Camera10/', 'New')	Error Code: 3819. Check constraint 'chk_cctv_status' is violated.

การสร้างตาราง **ช่วงเวลาการทำงาน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางช่วงเวลาการทำงานขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ officer_worktime เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM worktime; แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE worktime (
    worktime CHAR(10) PRIMARY KEY,
    start_worktime TIME NOT NULL,
    end_worktime TIME NOT NULL
);
INSERT INTO worktime
VALUES  ('Period 1', '8:00:00', '16:00:00'),
        ('Period 2', '16:00:00', '00:00:00'),
        ('Period 3', '00:00:00', '8:00:00'),
        ('Period 4', '7:00:00', '15:00:00');
SELECT * FROM worktime;
```

Result Grid | Filter Rows: | Edit:

worktime	start_worktime	end_worktime
Period 1	08:00:00	16:00:00
Period 2	16:00:00	00:00:00
Period 3	00:00:00	08:00:00
Period 4	07:00:00	15:00:00

การสร้างตาราง **เจ้าหน้าที่** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางเจ้าหน้าที่ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ officer_idเป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ carpark_id เป็น FK และอ้างอิงจากตาราง car_parking

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ officer_worktime เป็น FK และอ้างอิงจากตาราง worktime

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Officer แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Officer(
    officer_id CHAR(6) PRIMARY KEY,
    carpark_id CHAR(6) NOT NULL,
    officer_name VARCHAR(20) NOT NULL,
    officer_lastname VARCHAR(20) NOT NULL,
    officer_position VARCHAR(50) NOT NULL,
    officer_phonenumber CHAR(11) NOT NULL,
    officer_worktime CHAR(10) NOT NULL,
    FOREIGN KEY (carpark_id) REFERENCES car_parking (carpark_id),
    FOREIGN KEY (officer_worktime) REFERENCES worktime (worktime)
);
INSERT INTO Officer
VALUES  ('N00007', 'P00101', 'Somporn', 'Khajornsak', 'Guard', '667786251', 'Period 1'),
        ('N00008', 'P00102', 'Somjit', 'Mitphon', 'Guard', '991234545', 'Period 1'),
        ('N00001', 'P00103', 'Somrak', 'Maksuk', 'Head Guard', '885789012', 'Period 1'),
        ('N00003', 'P00104', 'Somthawin', 'Nimitmai', 'Guard', '633372775', 'Period 2'),
        ('N00004', 'P00105', 'Sompong', 'Kongkiat', 'Guard', '856547895', 'Period 3'),
        ('N00009', 'P00106', 'Somchai', 'Tangjaijang', 'Cleaner', '623456789', 'Period 4'),
        ('N00010', 'P00103', 'Somying', 'Saetang', 'Maintenance', '778902345', 'Period 1'),
        ('N00011', 'P00104', 'Sompong', 'Chancharoen', 'Cashier', '956783456', 'Period 2'),
        ('N00012', 'P00109', 'Somrdee', 'Pheande', 'Cleaner', '534567890', 'Period 4');
SELECT * FROM Officer;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

officer_id	carpark_id	officer_name	officer_lastname	officer_position	officer_phonenumber	officer_worktime
N00001	P00103	Somrak	Maksuk	Head Guard	885789012	Period 1
N00003	P00104	Somthawin	Nimitmai	Guard	633372775	Period 2
N00004	P00105	Sompong	Kongkiat	Guard	856547895	Period 3
N00007	P00101	Somporn	Khajornsak	Guard	667786251	Period 1
N00008	P00102	Somjit	Mitphon	Guard	991234545	Period 1
N00009	P00106	Somchai	Tangjaijang	Cleaner	623456789	Period 4
N00010	P00103	Somying	Saetang	Maintenance	778902345	Period 1
N00011	P00104	Sompong	Chancharoen	Cashier	956783456	Period 2
N00012	P00109	Somrdee	Pheande	Cleaner	534567890	Period 4

การสร้างตาราง **การตรวจสอบที่จอดรถยนต์** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการตรวจสอบที่จอดรถยนต์ขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Inspection_ID เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ carpark_id เป็น FK และอ้างอิงจากตาราง car_parking

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Inspection แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Inspection (
    inspection_id CHAR(6) PRIMARY KEY,
    carpark_id CHAR(6) NOT NULL,
    inspection_day TINYINT NOT NULL,
    inspection_month TINYINT NOT NULL,
    inspection_year SMALLINT NOT NULL,
    inspection_time TIME NOT NULL,
    inspection_issue VARCHAR(100),
    inspection_Record VARCHAR(100),
    FOREIGN KEY (carpark_id) REFERENCES car_parking(carpark_id)
);
INSERT INTO Inspection
VALUES  ('K00001', 'P00101', 20, 2, 2567, '08:00:00', 'ไฟไม่ติด', 'Waiting for repairs'),
        ('K00002', 'P00102', 1, 3, 2567, '09:15:00', 'สีเส้นไม่ชัด', 'Repair completed'),
        ('K00003', 'P00103', 22, 4, 2567, '07:30:00', NULL, NULL),
        ('K00004', 'P00104', 23, 5, 2567, '10:00:00', NULL, NULL),
        ('K00005', 'P00105', 24, 6, 2567, '08:45:00', 'กล้องมองไม่เห็น', 'Under repair'),
        ('K00006', 'P00101', 25, 7, 2567, '12:00:00', 'ไฟไม่ติด', 'Repair completed'),
        ('K00007', 'P00101', 26, 7, 2567, '15:00:00', NULL, NULL),
        ('K00008', 'P00101', 27, 7, 2567, '14:30:00', 'กล้องบันทึกภาพไม่ได้', 'Waiting for repairs'),
        ('K00009', 'P00102', 28, 8, 2567, '16:00:00', 'ไฟไม่ติด', 'Repair completed');
SELECT * FROM Inspection;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

inspection_id	carpark_id	inspection_day	inspection_month	inspection_year	inspection_time	inspection_issue	inspection_Record
K00001	P00101	20	2	2567	08:00:00	ไฟไม่ติด	Waiting for repairs
K00002	P00102	1	3	2567	09:15:00	สีเส้นไม่ชัด	Repair completed
K00003	P00103	22	4	2567	07:30:00	NULL	NULL
K00004	P00104	23	5	2567	10:00:00	NULL	NULL
K00005	P00105	24	6	2567	08:45:00	กล้องมองไม่เห็น	Under repair
K00006	P00101	25	7	2567	12:00:00	ไฟไม่ติด	Repair completed
K00007	P00101	26	7	2567	15:00:00	NULL	NULL
K00008	P00101	27	7	2567	14:30:00	กล้องบันทึกภาพไม่ได้	Waiting for repairs
K00009	P00102	28	8	2567	16:00:00	ไฟไม่ติด	Repair completed

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการตรวจสอบที่จอดรถยนต์

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Inspection_date

CHECK

- กำหนดเงื่อนไขที่มูลในตาราง ตรวจสอบว่าปีที่ทำการตรวจสอบ (inspection_year) อยู่ในช่วง 2565 ถึง 2568
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ทำการตรวจสอบ (inspection_month) อยู่ในช่วง 1 ถึง 12

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการตรวจสอบ (inspection_day) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่ทำการตรวจสอบ ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่ทำการตรวจสอบ (inspection_month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ทำการตรวจสอบไม่เกิน 29

```sql
ALTER TABLE inspection
ADD CONSTRAINT chk_Inspection_date
CHECK (
    inspection_year BETWEEN 2565 AND 2568 AND
    inspection_month BETWEEN 1 AND 12 AND
    inspection_day BETWEEN 1 AND 31 AND
    (
        (inspection_month IN (1, 3, 5, 7, 8, 10, 12)) OR
        (inspection_month IN (4, 6, 9, 11) AND inspection_day <= 30) OR
        (inspection_month = 2 AND inspection_day <= 29)
    )
);
```

ทดลองใส่ค่าวันที่ไม่ตรงตามข้อกำหนด ทดลองใส่ วันที่ 31 เดือน 11 ปี 2565 จะไม่สามารถทำได้ เนื่องจากเดือน 11 มีแค่ 30 วัน

```sql
INSERT INTO Inspection
VALUES  ('K00010', 'P00102', 31, 11, 2565, '16:00:00', 'ไฟไม่ติด', 'Repair completed');
SELECT * FROM Inspection;
```

Output

Action Output

#	Time	Action	Message
1	18:31:56	INSERT INTO Inspection VALUES('K00010', 'P00102', 31, 11, 2565, '16:00:00', 'ไฟไม่ติด', 'Repair completed')	Error Code: 3819. Check constraint 'chk_Inspection_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการตรวจสอบที่จอดรถยนต์

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Inspection_status

CHECK สถานะการซ่อมแซม (cctv_status) มี 4 สถานะ คือ Waiting for repairs, Under repair, Repair completed และ NULL

```sql
ALTER TABLE inspection
ADD CONSTRAINT chk_Inspection_status
CHECK (inspection_Record IN('Waiting for repairs', 'Under repair', 'Repair completed', 'NULL'));
```

ทดลองใส่สถานะ Wait จะไม่สามารถทำได้ เนื่องจากค่าสถานะเป็นได้แค่ 4 สถานะ คือ Waiting for repairs, Under repair, Repair completed และ NULL เท่านั้น

```
1 •  INSERT INTO Inspection
2    VALUES ('K00011', 'P00102', 28, 8, 2567, '16:00:00', 'ไฟไม่ติด', 'Wait');
```

Output

Action Output

#	Time	Action	Message
1	21:45:08	INSERT INTO Inspection VALUES ('K00011', 'P00102', 28, 8, 2567, '16:00:00', 'ไฟไม่ติด', 'Wait')	Error Code: 3819. Check constraint 'chk_Inspection_status' is violated.

ตัวอย่างโจทย์ SQL

1.แสดงชื่อ นามสกุลของผู้จอด และค่าจอดรถรวมทั้งหมดที่มากที่สุดในเดือน 2 ปี 2567

คำสั่ง SQL

```
SELECT owner_name, owner_lastname, SUM(amount) AS total_Expense
FROM owners, cars, parkingreservation, parking, payment
WHERE owners.owner_id = cars.owner_id
AND cars.license_plate = parkingreservation.license_plate
AND parkingreservation.reservation_id = parking.reservation_id
AND parking.parking_id = payment.parking_id
AND parking.park_month = 2
AND parking.park_year = 2567
GROUP BY owners.owner_name, owners.owner_lastname
HAVING total_Expense = (
    SELECT MAX(total_Expense)
    FROM (
        SELECT owner_name, owner_lastname, SUM(amount) AS total_Expense
        FROM owners, cars, parkingreservation, parking, payment
        WHERE owners.owner_id = cars.owner_id
        AND cars.license_plate = parkingreservation.license_plate
        AND parkingreservation.reservation_id = parking.reservation_id
        AND parking.parking_id = payment.parking_id
        AND parking.park_month = 2
        AND parking.park_year = 2567
        GROUP BY owners.owner_name, owners.owner_lastname
    ) AS test_table
)
```

```sql
SELECT owner_name, owner_lastname, SUM(amount) AS total_Expense
FROM owners, cars, parkingreservation, parking, payment
WHERE owners.owner_id = cars.owner_id
AND cars.license_plate = parkingreservation.license_plate
AND parkingreservation.reservation_id = parking.reservation_id
AND parking.parking_id = payment.parking_id
AND parking.park_month = 2
AND parking.park_year = 2567
GROUP BY owners.owner_name, owners.owner_lastname
HAVING total_Expense = (
    SELECT MAX(total_Expense)
    FROM (
        SELECT owner_name, owner_lastname, SUM(amount) AS total_Expense
        FROM owners, cars, parkingreservation, parking, payment
        WHERE owners.owner_id = cars.owner_id
        AND cars.license_plate = parkingreservation.license_plate
        AND parkingreservation.reservation_id = parking.reservation_id
        AND parking.parking_id = payment.parking_id
        AND parking.park_month = 2
        AND parking.park_year = 2567
        GROUP BY owners.owner_name, owners.owner_lastname
    ) AS test_table
)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

owner_name	owner_lastname	total_Expense
Saifon	Wiriya	200.00
Wipa	Ratanawong	200.00

คำอธิบาย

ทำการค้นหา **ชื่อ** (owner_name) **นามสกุล** (owner_lastname) **ผลรวมค่าบริการทั้งหมด** (total_Expense)

จากตารางเจ้าของรถยนต์ (owners) ตารางรถยนต์ (cars) ตารางการจองที่จอดรถยนต์ (parkingreservation) ตารางการจอดรถยนต์ (parking) ตารางการชำระเงิน (payment)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**เจ้าของรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ reservation_id
- ตาราง**การจอดรถยนต์** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ parking_id

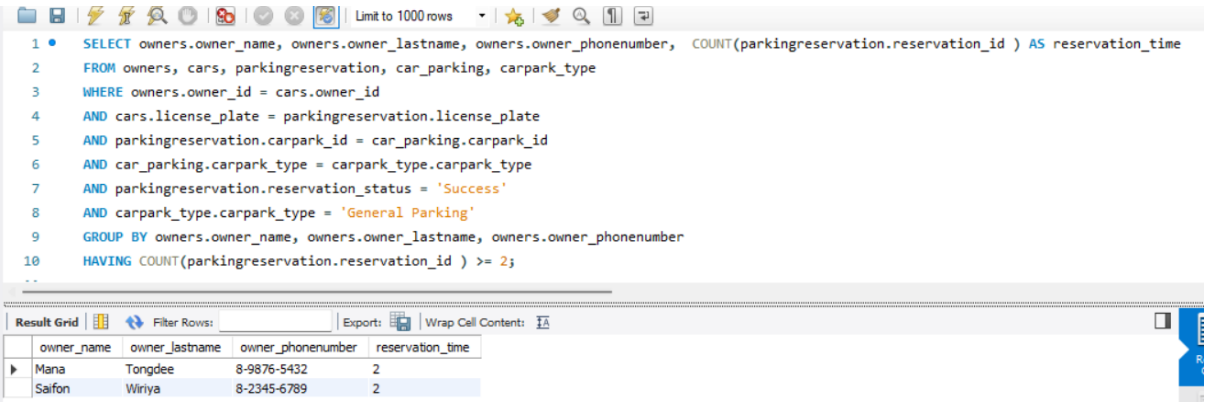
โดยเลือกเฉพาะ รถยนต์ที่มีการจอดในเดือน 2 ปี 2567 จัดกลุ่มด้วย ชื่อและนามสกุลเจ้าของรถยนต์ จากนั้นเลือกเฉพาะคนที่มีผลรวมค่าบริการทั้งหมด เท่ากับ ผลรวมค่าบริการทั้งหมดสูงสุด ด้วยการทำ Subqueryได้ผลลัพธ์เป็นตาราง test_table นำไปเปรียบเทียบด้วยคำสั่ง HAVING

เพื่อเก็บสถิติผู้ที่มีผลรวมค่าบริการทั้งหมดมากที่สุดในแต่ละเดือน อาจใช้ในการมอบส่วนลดให้ผู้ที่มีผลรวมค่าบริการทั้งหมดมากที่สุดในแต่ละเดือน

2. แสดงชื่อ นามสกุล เบอร์โทรศัพท์ ของเจ้าของรถยนต์ ที่ทำการจองที่จอดรถยนต์ประเภทที่จอดทั่วไป สำเร็จ อย่างน้อย 2 ครั้ง

คำสั่ง SQL

```
SELECT owners.owner_name, owners.owner_lastname, owners.owner_phonenumber,
COUNT(parkingreservation.reservation_id ) AS reservation_time
FROM owners, cars, parkingreservation, car_parking, carpark_type
WHERE owners.owner_id = cars.owner_id
AND cars.license_plate = parkingreservation.license_plate
AND parkingreservation.carpark_id = car_parking.carpark_id
AND car_parking.carpark_type = carpark_type.carpark_type
AND parkingreservation.reservation_status = 'Success'
AND carpark_type.carpark_type = 'General Parking'
GROUP BY owners.owner_name, owners.owner_lastname,owners.owner_phonenumber
HAVING COUNT(parkingreservation.reservation_id ) >= 2;
```



คำอธิบาย

ทำการค้นหา **ชื่อ** (owner_name) **นามสกุล** (owner_lastname) **เบอร์โทรศัพท์** (phonenumber) **จำนวนครั้งที่ทำการจองที่จอดรถยนต์** (reservation_time)

จากตารางเจ้าของรถยนต์ (owners) ตารางรถยนต์ (cars) ตารางการจองที่จอดรถยนต์ (parkingreservation) ตารางการจอดรถยนต์ (parking)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**เจ้าของรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate
- ตาราง**การจองที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**ที่จอดรถยต์** กับ **ประเภทที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_type

โดยเลือกเฉพาะการจองที่มีสถานะสำเร็จ (Success) และเป็นที่จอดทั่วไปเท่านั้น จัดกลุ่มด้วยชื่อนามสกุลและเบอร์โทรศัพท์ จากนั้นเลือกเฉพาะคนที่มีการจองที่ทำการจองมากกว่า 2 ครั้ง

เพื่อใช้สำหรับการเก็บสะสมสถิติการจอง ว่าทำการจองสำเร็จแล้วกี่ครั้ง สามารถนำไปใช้เป็นการสะสมแต้มเพื่อแลกส่วนลดได้

3. แสดงประเภทที่จอดรถที่ถูกจองมากที่สุด และจำนวนครั้งที่ถูกจอง

คำสั่ง SQL

```
SELECT carpark_type.carpark_type, COUNT(parkingreservation.carpark_id) AS
reservation_count
FROM parkingreservation, car_parking, carpark_type
WHERE parkingreservation.carpark_id = car_parking.carpark_id
AND car_parking.carpark_type = carpark_type.carpark_type
GROUP BY carpark_type.carpark_type
HAVING reservation_count = (
        SELECT MAX(reservation_count)
        FROM (
                SELECT carpark_type.carpark_type,
                COUNT(parkingreservation.carpark_id) AS reservation_count
                FROM parkingreservation, car_parking, carpark_type
                WHERE parkingreservation.carpark_id = car_parking.carpark_id
                AND car_parking.carpark_type = carpark_type.carpark_type
                GROUP BY carpark_type.carpark_type
        ) AS test_table
);
```

```
1   SELECT carpark_type.carpark_type, COUNT(parkingreservation.carpark_id) AS reservation_count
2   FROM parkingreservation, car_parking, carpark_type
3   WHERE parkingreservation.carpark_id = car_parking.carpark_id
4   AND car_parking.carpark_type = carpark_type.carpark_type
5   GROUP BY carpark_type.carpark_type
6   HAVING reservation_count = (
7       SELECT MAX(reservation_count)
8       FROM (
9           SELECT carpark_type.carpark_type, COUNT(parkingreservation.carpark_id) AS reservation_count
10          FROM parkingreservation, car_parking, carpark_type
11          WHERE parkingreservation.carpark_id = car_parking.carpark_id
12          AND car_parking.carpark_type = carpark_type.carpark_type
13          GROUP BY carpark_type.carpark_type
14      ) AS test_table
15  );
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

carpark_type	reservation_count
General Parking	7

คำอธิบาย

ทำการค้นหา **ประเภทที่จอดรถยนต์** (carpark_type) **จำนวนครั้งที่ที่จอดรถยนต์ถูกจอง** (reservation_count)

จากตารางการจองที่จอดรถยนต์ (parkingreservation) ตารางที่จอดรถยนต์ (car_parking) ตารางประเภทที่จอดรถยนต์ (carpark_type)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**การจองที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**ที่จอดรถยนต์** กับ **ประเภทที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_type

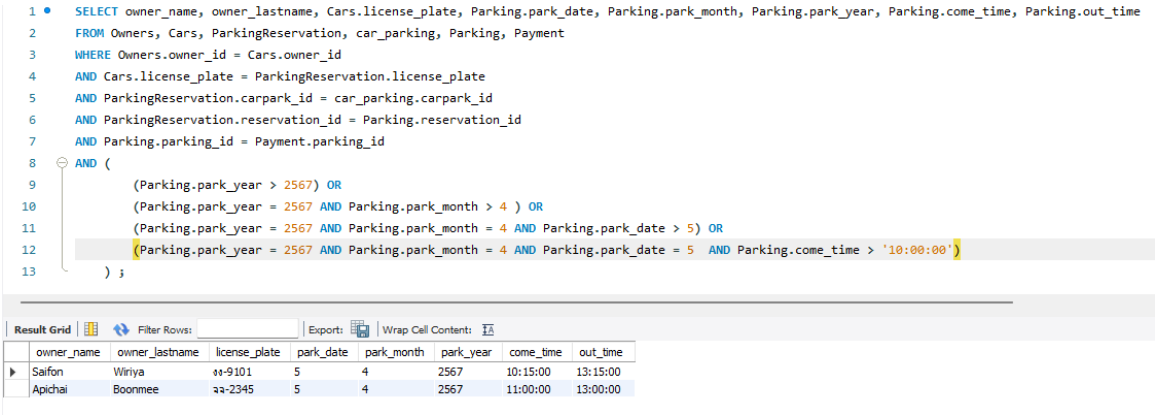
จัดกลุ่มด้วยประเภทที่จอดรถยนต์ การนับประเภทที่จอดรถยนต์ที่ถูกจองเป็นการนับแยกประเภทกันเกิดการจากการเชื่อมตาราง เมื่อจัดกลุ่มด้วยประเภทที่จอดรถยนต์ แล้วนับรหัสที่จอดรถยนต์ที่อยู่ในตารางการจองที่จอดรถยนต์ จะสามารถจำนวนครั้งที่ที่จอดรถยนต์แต่ละประเภทถูกจองได้ จากนั้นเลือกเฉพาะประเภทที่จอดรถยนต์ที่มีการถูกจอง เท่ากับ ประเภทที่จอดรถยนต์ที่ถูกจองบ่อยที่สุด ด้วยการทำ Subquery ได้ผลลัพธ์เป็นตาราง test_table นำไปเปรียบเทียบด้วยคำสั่ง HAVING

เพื่อเก็บข้อมูลว่าที่จอดรถยนต์ประเภทไหนถูกจองมากที่สุด เพื่อทำการเพิ่มจำนวนที่จอดประเภทนั้นเพื่อลดความแออัด

4. แสดงรายชื่อ นามสกุล ป้ายทะเบียน วันที่ เดือนและปี ของรถที่ใช้บริการตั้งแต่วันที่ 5 เมษายน 2567 เวลา 10:00:00 น.

คำสั่ง SQL

```
SELECT owner_name, owner_lastname, Cars.license_plate, Parking.park_date,
Parking.park_month, Parking.park_year, Parking.come_time, Parking.out_time
FROM Owners, Cars, ParkingReservation, car_parking, Parking, Payment
WHERE Owners.owner_id = Cars.owner_id
AND Cars.license_plate = ParkingReservation.license_plate
AND ParkingReservation.carpark_id = car_parking.carpark_id
AND ParkingReservation.reservation_id = Parking.reservation_id
AND Parking.parking_id = Payment.parking_id
AND (
    (Parking.park_year > 2567) OR
    (Parking.park_year = 2567 AND Parking.park_month > 4 ) OR
    (Parking.park_year = 2567 AND Parking.park_month = 4 AND
      Parking.park_date > 5) OR
    (Parking.park_year = 2567 AND Parking.park_month = 4 AND
      Parking.park_date = 5  AND Parking.come_time > '10:00:00')) ;
```



owner_name	owner_lastname	license_plate	park_date	park_month	park_year	come_time	out_time
Saifon	Wiriya	ฮฮ-9101	5	4	2567	10:15:00	13:15:00
Apichai	Boonmee	ลล-2345	5	4	2567	11:00:00	13:00:00

คำอธิบาย

ทำการค้นหา**ชื่อเจ้าของรถยนต์** (owner_name) และ **นามสกุลเจ้าของรถยนต์** (owner_lastname)

จากตารางเจ้าของรถยนต์ (Owners) ตารางรถยนต์ (Cars) ตารางการจองที่จอดรถยนต์ (ParkingReservation) ตารางที่จอดรถยนต์ (car_parking) ตารางการจอดรถยนต์ (Parking) ตารางการชำระเงิน (Payment)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**เจ้าของรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ owner_id

- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate
- ตาราง**การจองที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ reservation_id
- ตาราง**การจอดรถยนต์** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ parking_id

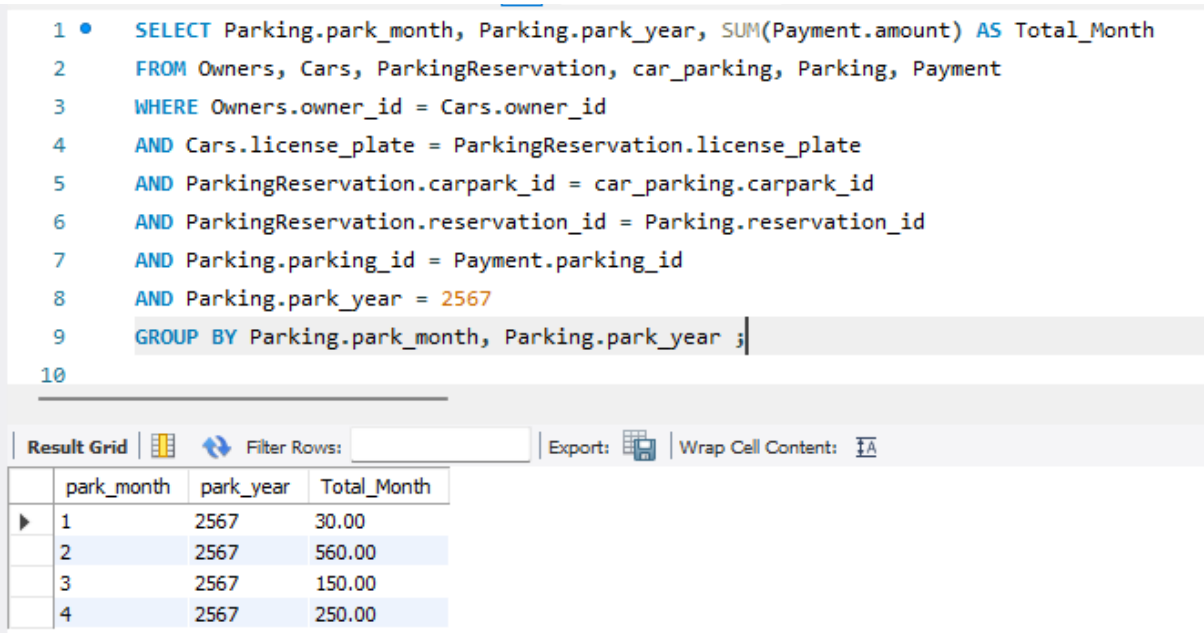
โดยเลือกเฉพาะ รถยนต์ที่ใช้บริการหลังวันที่ 5 เมษายน 2567 ช่วงเวลา 10:00:00 น. เป็นต้นไป

เพื่อใช้ในการตรวจสอบย้อนหลัง ในกรณีที่มีคนมาติดต่อขอดูประวัติการจอดรถ ในช่วงเวลานั้นว่ามีรถที่ต้องสงสัยมาจอดที่นี่ไหม

5.แสดงเดือน ปี และยอดรวมรายได้แต่ละเดือนว่า มีรายได้ในแต่ละเดือนเป็นเท่าใด

คำสั่ง SQL

```
SELECT Parking.park_month, Parking.park_year, SUM(Payment.amount) AS
Total_Month
FROM Owners, Cars, ParkingReservation, car_parking, Parking, Payment
WHERE Owners.owner_id = Cars.owner_id
AND Cars.license_plate = ParkingReservation.license_plate
AND ParkingReservation.carpark_id = car_parking.carpark_id
AND ParkingReservation.reservation_id = Parking.reservation_id
AND Parking.parking_id = Payment.parking_id
AND Parking.park_year = 2567
GROUP BY Parking.park_month, Parking.park_year ;
```



คำอธิบาย

ทำการค้นหา**เดือนที่จอด** (park_month) **ปีที่จอด** (park_year) และ **ยอดรวมในแต่ละเดือน** (Total_Month)

จากตารางเจ้าของรถยนต์ (Owners) ตารางรถยนต์ (Cars) ตารางการจองที่จอดรถยนต์ (ParkingReservation) ตารางที่จอดรถยนต์ (car_parking) ตารางการจอดรถยนต์ (Parking) และ ตารางการชำระเงิน (Payment)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**เจ้าของรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate

- ตาราง**การจองที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ reservation_id
- ตาราง**การจอดรถยนต์** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ parking_id

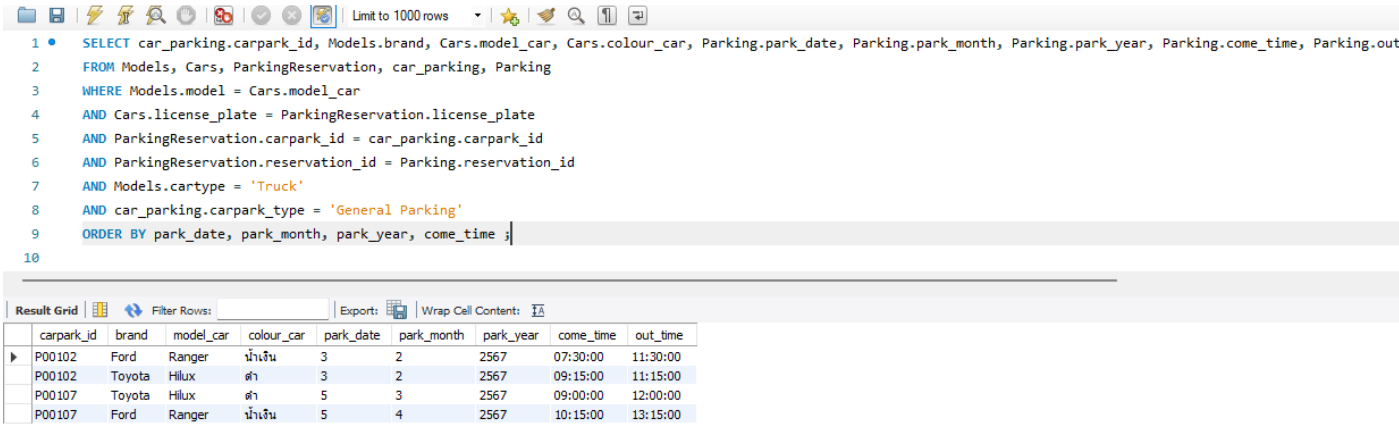
โดยเลือกเฉพาะเดือนในปี 2567 และทำการจัดกลุ่มด้วยเดือน

เพื่อเก็บข้อมูลจำนวนยอดเงินรวมแต่ละเดือนในปี 2567 ซึ่งจะถูกนำไปทำการสรุปยอดรายรับของการให้บริการที่จอดรถยนต์

6. แสดงรหัสที่จอดรถ ยี่ห้อรถยนต์ รุ่นรถยนต์ สีรถของรถที่เป็นประเภท "รถกระบะ" ที่ใช้บริการที่จอดรถยนต์ ประเภททั่วไป พร้อมทั้งแสดงวันที่ เดือน ปี เวลาเข้าใช้งาน และเวลาออก โดยเรียงจากวัน เดือน ปี และเวลาที่นำรถมาจอดจากน้อยไปมาก

คำสั่ง SQL

```
SELECT car_parking.carpark_id, Models.brand, Cars.model_car, Cars.colour_car,
Parking.park_date, Parking.park_month, Parking.park_year, Parking.come_time,
Parking.out_time
FROM Models, Cars, ParkingReservation, car_parking, Parking
WHERE Models.model = Cars.model_car
AND Cars.license_plate = ParkingReservation.license_plate
AND ParkingReservation.carpark_id = car_parking.carpark_id
AND ParkingReservation.reservation_id = Parking.reservation_id
AND Models.cartype = 'Truck'
AND car_parking.carpark_type = 'General Parking'
ORDER BY park_date, park_month, park_year, come_time ;
```



carpark_id	brand	model_car	colour_car	park_date	park_month	park_year	come_time	out_time
P00102	Ford	Ranger	น้ำเงิน	3	2	2567	07:30:00	11:30:00
P00102	Toyota	Hilux	ดำ	3	2	2567	09:15:00	11:15:00
P00107	Toyota	Hilux	ดำ	5	3	2567	09:00:00	12:00:00
P00107	Ford	Ranger	น้ำเงิน	5	4	2567	10:15:00	13:15:00

คำอธิบาย

ทำการค้นหา**รหัสที่จอดรถยนต์** (carpark_id) **ยี่ห้อรถยนต์** (brand) **รุ่นรถยนต์** (model_car) และ**สีของรถยนต์** (colour_car) **วันที่จอด** (park_date) **เดือนที่จอด** (park_month) **ปีที่จอด** (park_year) **เวลาที่เริ่มจอด** (come_time) และ**เวลาที่เลิกจอด**(out_time)

จากตารางรุ่นรถยนต์ (Models) ตารางรถยนต์ (Cars) ตารางการจองที่จอดรถยนต์ (ParkingReservation) ตารางที่จอดรถยนต์ (car_parking) และตารางการจอดรถยนต์ (Parking)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**รุ่นรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ model, model_car
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate

- ตาราง**การจองที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ reservation_id

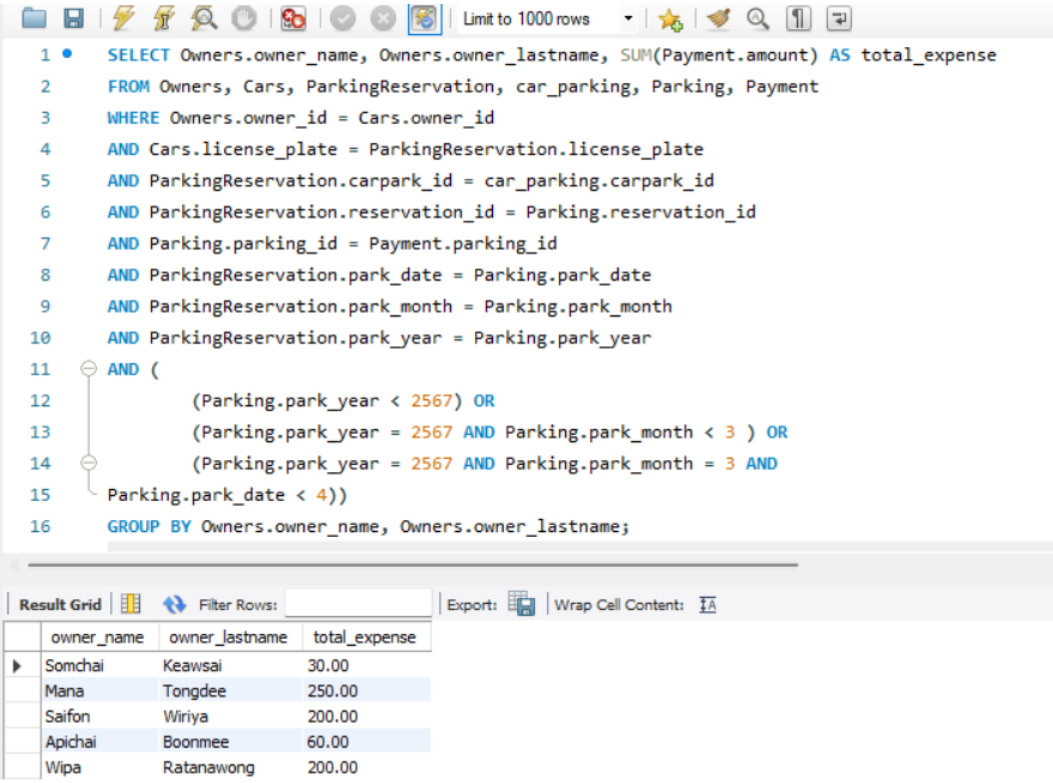
โดยเลือกเฉพาะ ประเภทที่เป็นรถกระบะ และทำการจอดในที่จอดประเภทที่จอดรถทั่วไป โดยเรียงจากวันที่ เดือน ปี และเวลาที่นำรถมาจอด จากน้อยไปมาก

เพื่อเก็บข้อมูลว่า รถกระบะที่มาใช้บริการที่จอดทั่วไป มีจำนวนมากน้อยเพียงใด เพื่อนำข้อมูลไปพิจารณาขยายพื้นที่จอดรถประเภททั่วไปให้เหมาะกับรถกระบะในอนาคต

7.แสดงรายชื่อ นามสกุล ของผู้ที่นำรถมาจอดก่อน วันที่ 4 มีนาคม 2567 และสรุปว่าแต่ละคนเสียค่าจอดทั้งหมดกี่บาท

คำสั่ง SQL

```
SELECT Owners.owner_name, Owners.owner_lastname, SUM(Payment.amount) AS
total_expense
FROM Owners, Cars, ParkingReservation, car_parking, Parking, Payment
WHERE Owners.owner_id = Cars.owner_id
AND Cars.license_plate = ParkingReservation.license_plate
AND ParkingReservation.carpark_id = car_parking.carpark_id
AND ParkingReservation.reservation_id = Parking.reservation_id
AND Parking.parking_id = Payment.parking_id
AND ParkingReservation.park_date = Parking.park_date
AND ParkingReservation.park_month = Parking.park_month
AND ParkingReservation.park_year = Parking.park_year
AND (
    (Parking.park_year < 2567) OR
    (Parking.park_year = 2567 AND Parking.park_month < 3 ) OR
    (Parking.park_year = 2567 AND Parking.park_month = 3 AND
      Parking.park_date < 4))
GROUP BY Owners.owner_name, Owners.owner_lastname;
```



owner_name	owner_lastname	total_expense
Somchai	Keawsai	30.00
Mana	Tongdee	250.00
Saifon	Wiriya	200.00
Apichai	Boonmee	60.00
Wipa	Ratanawong	200.00

คำอธิบาย

ทำการค้นหา**ชื่อเจ้าของรถยนต์**(owner_name) **นามสกุลเจ้าของรถยนต์**(owner_lastname) และ **ผลรวมค่าบริการทั้งหมด** (amount)

จากตารางเจ้าของรถยนต์ (Owners) ตารางรถยนต์ (Cars) ตารางการจองที่จอดรถยนต์ (ParkingReservation) ตารางที่จอดรถยนต์ (car_parking) ตารางการจอดรถยนต์ (Parking) และตารางการชำระเงิน (Payment)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**เจ้าของรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate
- ตาราง**การจองที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ reservation_id
- ตาราง**การจอดรถยนต์** กับ **การชำระเงิน** ด้วยแอตทริบิวต์ parking_id
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ park_date
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ park_month
- ตาราง**การจองที่จอดรถยนต์** กับ **การจอดรถยนต์** ด้วยแอตทริบิวต์ park_year

โดยเลือกเฉพาะวันที่น้อยกว่า วันที่ 4 เดือนที่ 3 และปีที่ 2567 โดยจัดกลุ่มด้ว ชื่อ-นามสกุลเจ้าของ

เพื่อตรวจสอบว่ามีใครมาใช้บริการที่จอดรถบ้าง และสรุปยอดเงินที่ได้จากผู้ใช้บริการแต่ละคน ทำสะดวกต่อการคำนวณยอดเงินที่ได้จากการเช่าที่จอดรถ

8.แสดงชื่อนามสกุลและเบอร์โทรศัพท์ของเจ้าหน้าที่ ที่ทำงานในที่จอดรถประเภท "ที่จอดรถสำหรับรถไฟฟ้า"

คำสั่ง SQL

```
SELECT officer_name,officer_lastname,officer_phonenumber
FROM officer , car_parking , carpark_type
WHERE officer.carpark_id = car_parking.carpark_id
AND  car_parking.carpark_type = carpark_type.carpark_type
AND carpark_type.carpark_type = 'EV Parking';
```

```
1 •  SELECT officer_name,officer_lastname,officer_phonenumber
2    FROM officer , car_parking , carpark_type
3    WHERE officer.carpark_id = car_parking.carpark_id
4    AND  car_parking.carpark_type = carpark_type.carpark_type
5    AND carpark_type.carpark_type = 'EV Parking';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

officer_name	officer_lastname	officer_phonenumber
Somthawin	Nimitmai	633372775
Sompong	Chancharoen	956783456
Sompong	Kongkiat	856547895

คำอธิบาย

ทำการค้นหา**ชื่อเจ้าหน้าที่** (owner_name) **นามสกุลเจ้าหน้าที่** (owner_lastname) **เบอร์โทรศัพท์เจ้าหน้าที่** (officer_phonenumber)

จากตารางเจ้าหน้าที่ (officer) ตารางที่จอดรถยนต์ (car_parking) ตารางประเภทที่จอดรถยนต์ (carpark_type)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**เจ้าหน้าที่** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id
- ตาราง**ที่จอดรถยนต์** กับ **ประเภทที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_type

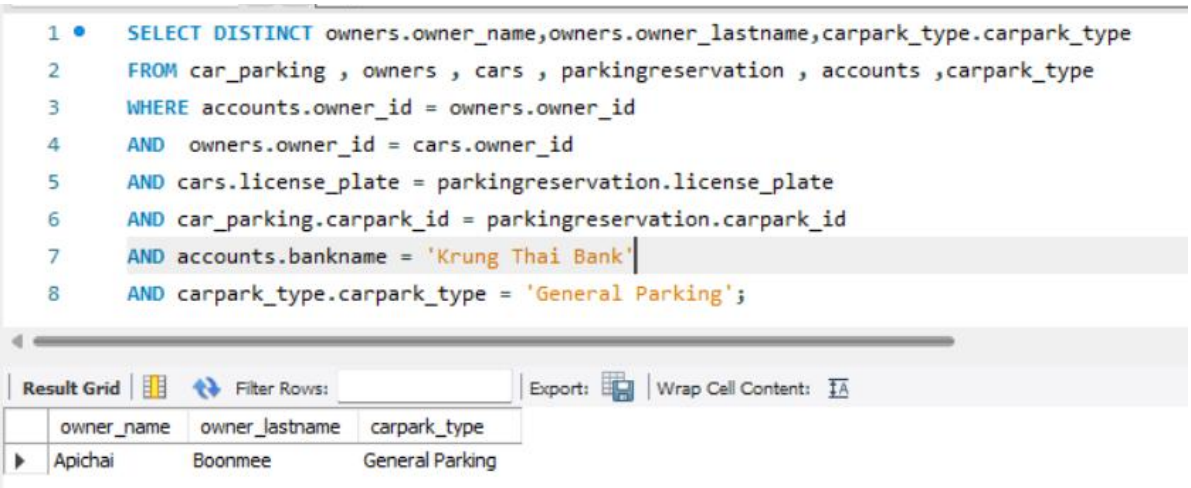
โดยเลือกเฉพาะที่จอดรถสำหรับรถไฟฟ้า (EV Parking)

เพื่อไว้ตรวจสอบชื่อ-นามสกุล และเบอร์โทรศัพท์ของเจ้าหน้าที่ที่เป็นคนดูแลที่จอดรถประเภทรถไฟฟ้า สำหรับใช้ติดต่อเมื่อเกิดปัญหา บริเวณที่จอดรถยนต์

9.แสดงรายชื่อเจ้าของรถยนต์ที่ใช้ธนาคารกรุงไทยในการชำระค่าจอดรถและมีการจองที่จอดรถ “ประเภททั่วไป”

คำสั่ง SQL

```
SELECT DISTINCT
owners.owner_name,owners.owner_lastname,carpark_type.carpark_type
FROM car_parking , owners , cars , parkingreservation , accounts ,carpark_type
WHERE accounts.owner_id = owners.owner_id
AND  owners.owner_id = cars.owner_id
AND cars.license_plate = parkingreservation.license_plate
AND car_parking.carpark_id = parkingreservation.carpark_id
AND accounts.bankname = 'Krung Thai Bank'
AND carpark_type.carpark_type = 'General Parking';
```



คำอธิบาย

ทำการค้นหา**ชื่อเจ้าของรถยนต์** (owner_name) **นามสกุลเจ้าของรถยนต์** (owner_lastname) และ **ประเภทที่จอดรถยนต์**(carpark_type)

จากตารางเจ้าของรถยนต์ (owners) ตารางที่จอดรถยนต์ (car_parking) ตารางรถยนต์ (cars) ตารางการจองที่จอดรถยนต์ (parkingreservation) ตารางบัญชี (accounts) และ ตารางประเภทที่จอดรถยนต์ (carpark_type)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**บัญชี** กับ **เจ้าของรถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**เจ้าของรถยนต์** กับ **รถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate
- ตาราง**ที่จอดรถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_id

- ตาราง**ประเภทที่จอดรถยนต์** กับ **ที่จอดรถยนต์** ด้วยแอตทริบิวต์ carpark_type

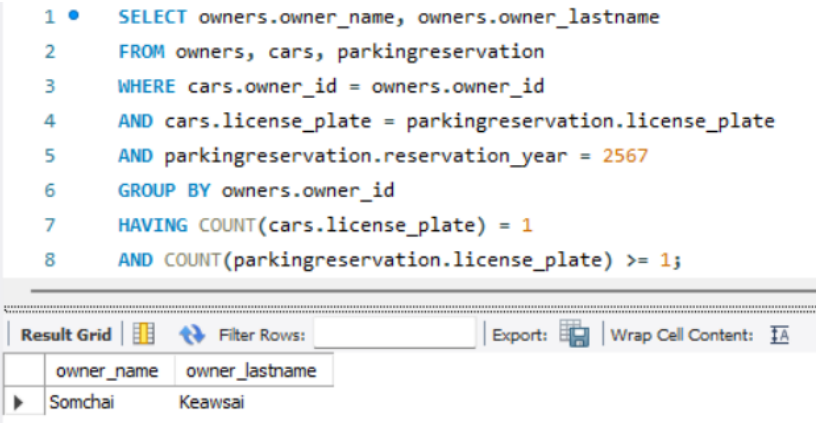
โดยเลือกเฉพาะ ชื่อธนาคาร เป็น 'Krung Thai Bank' และ ที่จอดรถยนต์ทั่วไป

เพื่อตรวจสอบการว่ามีการชำระเงินด้วยบัญชีกรุงไทยและใช้บริการที่จอดรถยนต์ประเภทที่จอดทั่วไป มากน้อยแค่ไหน สามารถนำข้อมูลไปพิจารณาให้สิทธิพิเศษแก่ลูกค้าได้ในอนาคต

10.แสดงชื่อ นามสกุล เจ้าของรถยนต์ที่มีรถยนต์ 1 คัน และมีการจองที่จอดรถในปี 2567 อย่างน้อย 2 ครั้ง

คำสั่ง SQL

```
SELECT owner_name, owner_lastname
FROM owners, cars, parkingreservation
WHERE cars.owner_id = owners.owner_id
AND cars.license_plate = parkingreservation.license_plate
AND parkingreservation.reservation_year = 2567
GROUP BY owners.owner_id
HAVING COUNT(cars.license_plate) = 1
AND COUNT(parkingreservation.license_plate) >= 1;
```



คำอธิบาย

ทำการค้นหา **ชื่อเจ้าของรถยนต์** (owner_name) และ **นามสกุลเจ้าของรถยนต์** (owner_lastname)

จากตารางเจ้าของรถยนต์(owners) ตารางรถยนต์ (cars) และตารางการจองที่จอดรถยนต์ (parkingreservation)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**รถยนต์** กับ **เจ้าของรถยนต์** ด้วยแอตทริบิวต์ owner_id
- ตาราง**รถยนต์** กับ **การจองที่จอดรถยนต์** ด้วยแอตทริบิวต์ license_plate

โดยเลือกปี 2567 จัดกลุ่มด้วยเจ้าของรถยนต์ เลือกเฉพาะคนที่มีรถยนต์ 1 คัน และมีการจองที่จอดรถยนต์มากกว่า 1 ครั้ง

เพื่อตรวจสอบชื่อ นามสกุลของเจ้าของรถที่มีรถยนต์ 1 คัน และทำการจอดรถยนต์มากกว่า 1 ครั้ง และใช้ในการตรวจสอบพฤติกรรมได้ว่า คนที่มีรถ 1 คัน และคนที่มีรถมากกว่า 1 คัน มีการใช้งานที่จอดรถยนต์แตกต่างกันอย่างไรบ้าง

ระบบการลงทะเบียนเรียน

1. เรื่องราว(Story)

1. ก่อนที่นิสิตจะทำการลงทะเบียนเรียนได้ นิสิตจะต้องกรอกข้อมูลส่วนตัวของตนเองเพื่อนำเก็บไว้ในระบบ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสนิสิต ชื่อสาขาวิชา ชื่อ นามสกุล ชั้นปี วัน/เดือน/ปีที่เกิด เพศ อีเมล เบอร์โทรติดต่อ สถานะนิสิต และ GPA
2. ภายในหลักสูตรจะมีรายละเอียดของหลักสูตร โดยขั้นตอนนี้จะมีการเก็บข้อมูล ชื่อหลักสูตร คณะ ชื่อปริญญา ระดับการศึกษา
3. สาขาวิชาจะอยู่ภายใต้หลักสูตร ซึ่งจะมีรายละเอียดข้อมูลของสาขาวิชานั้นๆ โดยจะมีการเก็บข้อมูล ชื่อสาขาวิชา รหัสหลักสูตร จำนวนหน่วยกิตรวมตลอดหลักสูตร ปีที่ปรับปรุงหลักสูตร หมวดวิชาศึกษาทั่วไป หมวดวิชาเฉพาะเลือก ค่าเทอมภาคปกติ และค่าเทอมภาคพิเศษ
4. อาจารย์ที่ปรึกษาต้องมีการบันทึกข้อมูลส่วนตัวที่ถูกต้องและครบถ้วนในระบบมหาวิทยาลัย โดยขั้นตอนนี้จะมีการเก็บข้อมูล ชื่อสาขาวิชา ชื่อ นามสกุล วัน/เดือน/ปีที่เกิด เพศ ตำแหน่งวิชาการ ห้องทำงาน อีเมล เบอร์โทรติดต่อ
5. อาจารย์ผู้สอนต้องมีการบันทึกข้อมูลส่วนตัวที่ถูกต้องและครบถ้วนในระบบมหาวิทยาลัย โดยขั้นตอนนี้จะมีการเก็บข้อมูล ชื่อสาขาวิชา ชื่อ นามสกุล วัน/เดือน/ปีที่เกิด เพศ ตำแหน่งวิชาการ ห้องทำงาน อีเมล เบอร์โทรติดต่อ
6. รายวิชาที่เปิดสอนจะมีรายละเอียดของรายวิชาและอาจารย์ที่สอนวิชานี้ โดยจะมีการเก็บข้อมูล รหัสรายวิชาที่เปิดสอน รหัสอาจารย์ผู้สอน ชื่อรายวิชาที่เปิดสอน หน่วยกิต คำอธิบายรายวิชา ปีการศึกษา ภาคการศึกษา ประเภทรายวิชาที่เปิดสอน สถานะรายวิชาที่เปิดสอน และหมายเหตุของรายวิชา
7. การจองห้องเรียนจะมีรายละเอียดข้อมูลในการจองห้องเรียน โดยอาจารย์ผู้สอนจะต้องทำการจองห้องเรียนเอง โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสการจองห้องเรียน รหัสวิชาที่เปิดสอน รหัสห้องเรียน วันที่เรียน เวลาเริ่ม - เวลาสิ้นสุด หมู่เรียน จำนวนนิสิต สถานะการจองห้องเรียน
8. การชำระเงินค่าเทอม นิสิตจะต้องทำการชำระเงินค่าเทอมก่อนจึงจะสามารถลงทะเบียนเรียนได้ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสนิสิต จำนวนเงิน วัน/เดือน/ปีที่ชำระเงิน เวลาที่ชำระเงิน ส่วนลด5A จำนวนเงินสุทธิ ช่องทางการชำระเงิน การขอผ่อนผัน กองทุนที่ได้รับ สถานะการชำระเงินค่าเทอม
9. หลังจากการชำระเงินค่าเทอมเสร็จสิ้นแล้ว นิสิตจะสามารถทำการลงทะเบียนเรียนได้ โดยขั้นตอนนี้จะมีการเก็บข้อมูล รหัสรายวิชาที่เปิดสอน หมายเลขการชำระเงินค่าเทอม วัน/เดือน/ปีที่ลงทะเบียน เวลาที่ลงทะเบียนเรียน สถานะการลงทะเบียน ผลการเรียน

ข้อกำหนด

- นิสิตจะต้องชำระเงินค่าเทอมก่อนจึงจะสามารถลงทะเบียนเรียนได้
- รายวิชาที่เปิดสอนจะมีอาจารย์ผู้สอนได้เพียง 1 คน และอาจารย์ผู้สอนจะทำการจองห้องเรียนเอง
- ส่วนลด5A นิสิตจะได้ก็ต่อเมื่อนิสิตสอบได้ระดับคะแนน A อย่างน้อย 5 รายวิชา และมีจำนวนหน่วยกิตไม่น้อยกว่า 12 หน่วยกิต
- การลงทะเบียนเรียนในแต่ละภาคการศึกษานิสิตจะไม่สามารถลงรายวิชาที่เปิดสอนซ้ำกันได้
- สาขาวิชาจะทำการแก้ไขหรือพัฒนาเนื้อหาหลักสูตรใหม่ ทุกๆ 5 ปีการศึกษา

2. สร้าง Entity Relational Diagram

ตารางนิสิต

รหัสนิสิต	ชื่อสาขาวิชา	รหัสอาจารย์ที่ปรึกษา	ชื่อ	นามสกุล	ภาคปกติ/ภาคพิเศษ	ชั้นปี	วันที่เกิด
6621600221	สาขาวิชาวิทยาการคอมพิวเตอร์	TC0314	ศิริพร	อำไพพงศ์	RP	2	4
6621600873	สาขาวิชาการบัญชี	TC0314	ใจดี	บ่อ	RP	2	6
6721600896	สาขาวิชาการตลาด	TC0411	กะตอย	โชว์หวย	SP	1	8
6721604593	สาขาวิชาคณิตศาสตร์ประยุกต์	TC0512	ตะนอย	แซ่เลี้ยว	SP	1	4
6712594569	สาขาวิชาเทคโนโลยีสารสนเทศ	TC0519	จักกี้	โกโก้ครั้น	RP	3	13
6760078701	สาขาสัตวบาล	TC0318	แสตนด์	จัพพ์	SP	1	31
6721790065	สาขาวิชาวิทยาการคอมพิวเตอร์	TC0315	สิวลักษณ์	ชินวัตร	SP	1	24
6521730068	สาขาวิศวกรรมโยธา	TC0167	ตั๊ก	เชิญธง	RP	3	23
6421876003	สาขาวิชาวิทยาการคอมพิวเตอร์	TC1124	ทองแดง	จิตต์ดี	SP	4	10
6621900345	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	TC2101	แฟ่ง	ไร่แตง	RP	2	15

เดือนที่เกิด	ปีที่เกิด	เพศ	อีเมล	เบอร์โทรติดต่อ	สถานะนิสิต	GPA
12	2549	F	Siripon.a@ku.th	9-8293-6201	ST	3.22
2	2548	F	Jaidee.b@ku.th	9-2567-1388	ST	2.54
10	2548	M	Katoy.ch@ku.th	8-7453-2169	LO	1.85
2	2547	M	Tanoy.s@ku.th	9-1257-8910	ST	2.89
8	2546	M	Jakkee.k@ku.th	8-6253-4915	ST	3.48
10	2548	M	Stan.j@ku.th	9-8768-5438	ST	3.72
3	2549	F	Siwaluk.ch@ku.th	8-7965-4321	ST	3.11
3	2546	M	Tak.ch@ku.th	6-1891-6905	ST	2.59
6	2543	M	Tongdang.j@ku.th	8-6453-7881	LO	2.98
4	2547	F	Fang.R@ku.th	6-5359-8723	ST	3.8

ตารางนิสิต ประกอบด้วย Attributes ดังนี้

รหัสนิสิต : สำหรับการค้นหาข้อมูลของนิสิต 1 คน

ชื่อสาขาวิชา : เพื่อบอกนิสิตคนนี้สังกัดอยู่สาขาวิชา

รหัสอาจารย์ที่ปรึกษา : หมายเลขประจำตัวของอาจารย์ผู้รับผิดชอบดูแลนิสิตคนนั้น

ชื่อ-นามสกุล : เพื่อให้ทราบ ชื่อ-นามสกุลนิสิต สามารถนำข้อมูลส่วนนี้ใช้ระบุตัวตนได้

ภาคปกติ/ภาคพิเศษ : ระบุโครงการการศึกษา นำไปใช้ในกรณีระบุตัวตนนิสิตให้เจาะจงมากขึ้น

ชั้นปี : ระดับชั้นปีของนิสิตเริ่มนับจากการเข้าเรียนตั้งแต่ ปี 1

วัน/เดือน/ปีเกิด : สำหรับการระบุอายุของนิสิต และใช้ในการวิเคราะห์เพื่อพัฒนาการศึกษาและสวัสดิการนิสิต

เพศ : สำหรับระบุเพศสภาพ เพื่อกรณีที่ต้องการทราบว่านิสิตเป็นเพศหญิงหรือชาย

อีเมล : ใช้ในการติดต่อสื่อสารกับนิสิต เช่น การส่งจดหมายผ่านอีเมลเพื่อติดต่ออาจารย์ และ ประชาสัมพันธ์ข่าวสารในมหาวิทยาลัย

เบอร์โทรติดต่อ : ใช้ในการติดต่อสื่อสารกับนิสิต เช่น เมื่อเกิดปัญหาเร่งด่วนหรือเหตุการณ์ฉุกเฉินสามารถติดต่อนิสิตได้สะดวกรวดเร็ว

ตารางหลักสูตร

รหัสหลักสูตร	ชื่อหลักสูตร	คณะ	ชื่อปริญญา	ระดับการศึกษา
CO4003	หลักสูตรวิทยาศาสตรบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	วิทยาศาสตรบัณฑิต	B
CO6505	หลักสูตรบริหารธุรกิจบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	บริหารธุรกิจบัณฑิต	B
CO6506	หลักสูตรคณิตศาสตรบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	คณิตศาสตรบัณฑิต	B
CO4506	หลักสูตรวิทยาศาสตการกีฬารบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	วิทยาศาสตรการกีฬาบัณฑิต	B
CO4808	หลักสูตรวิทยาศาสตรมหาบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	วิทยาศาสตรมหาบัณฑิต	M
CO0165	หลักสูตรวิศวกรรมศาสตรบัณฑิต	วิศวกรรมศาสตร์	วิศวกรรมศาสตรบัณฑิต	B
CO2156	หลักสูตรศิลปศาสตรบัณฑิต	อุตสาหกรรมบริการ	ศิลปศาสตรบัณฑิต	B
CO0867	หลักสูตรศึกษาศาสตรบัณฑิต	ศึกษาศาสตร์	ศึกษาศาสตรบัณฑิต	B

ตารางหลักสูตร ประกอบด้วย Attributes ดังนี้

รหัสหลักสูตร : สำหรับการค้นหาข้อมูลของหลักสูตร 1 หลักสูตร

ชื่อหลักสูตร : ระบุชื่อเต็มของหลักสูตรที่เปิดสอนในมหาวิทยาลัย

คณะ : ระบุชื่อคณะที่หลักสูตรนั้นสังกัดอยู่ เพื่อให้ทราบว่าหลักสูตรอยู่ภายใต้การดูแลของคณะใด

ชื่อปริญญา : ชื่อปริญญาที่นิสิตจะได้รับเมื่อสำเร็จการศึกษาในหลักสูตรนั้น

ระดับการศึกษา : ระบุระดับการศึกษาในมหาวิทยาลัย เช่น ปริญญาตรี, ปริญญาโท และปริญญาเอก

ตารางการจองห้องเรียน

รหัสการจองห้องเรียน	รหัสรายวิชาที่เปิดสอน	รหัสห้องเรียน	วันที่เรียน	เวลาเริ่ม	เวลาสิ้นสุด	หมู่เรียน	จำนวนนิสิต	สถานะการจองห้องเรียน
JH1489	1418234	LH4-301	Mon	9:00:00	12:00:00	800	80	SC
JH1490	1418282	LH2-303	Tue	14:00:00	17:00:00	702	50	UF
JH1491	1418299	SC9-303	Fri	8:30:00	10:30:00	703	100	SC
JH1492	CHEM101	SC4-502	Wed	13:00:00	16:00:00	700	70	UF
JH1493	1418284	LH2-204	Sat	17:00:00	20:00:00	701	98	SC
JH1494	1418244	LH4-201	Mon	13:00:00	4:00:00	700	50	SC
JH1495	1418245	LH2-202	Tue	14:00:00	3:00:00	700	60	SC
JH1496	1418246	LH3-307	Fri	9:00:00	13:00:00	800	70	SC
JH1497	1418247	SC4-302	Sun	8:00:00	9:30:00	802	70	SC

ตารางการจองห้องเรียน ประกอบด้วย Attributes ดังนี้

รหัสการจองห้องเรียน : สำหรับทำการค้นหาข้อมูลของการจองห้องเรียน 1 ครั้ง

รหัสรายวิชาที่เปิดสอน : หมายเลขประจำรายวิชาที่ใช้ห้องเรียน

รหัสห้องเรียน : หมายเลขห้องเรียนที่ทำการจองห้องเรียน

วันที่เรียน : วันที่ใช้ห้องเรียนตามตารางเรียน

เวลาเริ่ม : เวลาเริ่มใช้ห้องรียน

เวลาสิ้นสุด : เวลาสิ้นสุดการใช้ห้องเรียน

หมู่เรียน : กลุ่มหรือหมู่เรียนของนิสิตที่อาจารย์ผู้สอนทำการสอนในรายวิชานี้

จำนวนนิสิต : จำนวนนิสิตที่เรียน

สถานะจองห้องเรียน : สถานะของการจอง เช่น ยืนยันแล้ว รออนุมัติ หรือถูกยกเลิก

ตารางสาขาวิชา

ชื่อสาขาวิชา	รหัสหลักสูตร	จำนวนหน่วยกิตรวมตลอดหลักสูตร(ไม่น้อยกว่า)	ปีที่ปรับปรุงหลักสูตร(พ.ศ.)	หมวดวิชาศึกษาทั่วไป(นก.)	หมวดวิชาเฉพาะเลือก(นก.)	ค่าเทอมภาคปกติ (บาท)	ค่าเทอมภาคพิเศษ (บาท)
สาขาวิชาวิทยาการคอมพิวเตอร์	CO4003	124	2565	30	23	16,900.00	40,000.00
สาขาวิชาการบัญชี	CO6505	130	2565	30	24	15,400.00	28,200.00
สาขาวิชาการตลาด	CO6505	124	2565	30	35	15,400.00	26,700.00
สาขาวิชาคณิตศาสตร์ประยุกต์	CO4506	124	2565	30	28	21,000.00	36,000.00
สาขาวิชาเทคโนโลยีสารสนเทศ	CO4808	126	2564	30	24	21,000.00	34,700.00
สาขาวิศวกรรมโยธา	CO0165	126	2565	30	28	17,300.00	32,400.00
สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	CO2156	124	2565	30	26	18,300.00	33,400.00
สาขาสัตวบาล	CO4003	128	2564	30	30	17,300.00	32,400.00
สาขาพลศึกษา	CO8067	130	2564	30	94	13,300.00	26,000.00

ตารางสาขาวิชา ประกอบด้วย Attributes ดังนี้

ชื่อสาขาวิชา : สำหรับทำการค้นหาข้อมูลของสาขาวิชา 1 สาขา

รหัสหลักสูตร : หมายเลขประจำหลักสูตรที่สาขาวิชาสังกัดอยู่

จำนวนหน่วยกิตรวมตลอดหลักสูตร(ไม่น้อยกว่า) : จำนวนหน่วยกิตที่นิสิตต้องเรียนให้ครบตามที่หลักสูตรกำหนด เพื่อสำเร็จการศึกษา โดยรวมทั้งวิชาบังคับ วิชาเลือก และกิจกรรมเสริมอื่นๆ

ปีที่ปรับปรุงหลักสูตร(พ.ศ.) : ปีที่มหาวิทยาลัยได้ทำการแก้ไขหรือพัฒนาเนื้อหาหลักสูตรใหม่

หมวดวิชาศึกษาทั่วไป(นก.) : จำนวนหน่วยกิตของวิชาพื้นฐานที่นิสิตทุกคนต้องเรียน

หมวดวิชาเฉพาะเลือก(นก.) : จำนวนหน่วยกิตของวิชาเลือกในสาขาวิชาที่นิสิตสามารถเลือกเรียนตามความสนใจ

ค่าเทอมภาคปกติ (บาท) : ค่าธรรมเนียมการศึกษาในหลักสูตรปกติ

ค่าเทอมภาคพิเศษ (บาท) : ค่าธรรมเนียมการศึกษาในหลักสูตรพิเศษ ซึ่งมีค่าใช้จ่ายสูงกว่าหลักสูตรภาคปกติ

ตารางอาจารย์ที่ปรึกษา

รหัสอาจารย์ที่ปรึกษา	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
TC0314	สาขาวิชาวิทยาการคอมพิวเตอร์	ใจดี	มีตัง	1	5
TC0315	สาขาวิชาการบัญชีบริหาร	สงยศ	สุขสม	2	8
TC0411	สาขาวิชาการตลาด	มั่นคง	มั่งคั่ง	3	10
TC0512	สาขาวิชาคณิตศาสตร์ประยุกต์	ใฝ่สอน	ชดเชย	4	12
TC0519	สาขาวิชาเทคโนโลยีสารสนเทศ	สมสาย	ผนงเดช	5	2
TC0234	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	จิตลดี	บานชื่น	6	5
TC0923	สาขาวิศวกรรมโยธา	ภพ	อดีตชาติ	7	11
TC0318	สาขาสัตวบาล	วัวน้อย	คอยรัก	10	7

ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ
2531	F	PA	SC9-330	Jaidee.me@ku.th	6-5552-2134
2539	M	LE	SC1-321	Songyod.s@ku.th	9-8645-1235
2537	M	AP	SC9-440	Mangkong.m@ku.th	6-5128-7895
2529	F	LE	SC5-123	Faison.ch@ku.th	9-8645-1278
2528	M	LE	SC9-232	Somsai.p@ku.th	9-0456-2879
2539	F	AP	SL2-201	Jitladee.b@ku.th	9-8234-6890
2527	M	PA	SE2-101	Phop.a@ku.th	6-8974-3241
2525	M	PA	ASS-101	Wuanoi.k@ku.th	8-9564-7821

ตารางอาจารย์ที่ปรึกษา ประกอบด้วย Attributes ดังนี้

รหัสอาจารย์ที่ปรึกษา : สำหรับทำการค้นหาข้อมูลของอาจารย์ที่ปรึกษา 1 คน

ชื่อสาขาวิชา : เพื่อระบุข้อมูลสาขาวิชาที่อาจารย์ที่ปรึกษาสังกัด

ชื่อ-นามสกุล : เพื่อให้ทราบชื่อ-นามสกุลของอาจารย์ที่ปรึกษา และสามารถนำข้อมูลส่วนนี้ใช้ระบุตัวตนได้

วัน/เดือน/ปีเกิด : สำหรับระบุอายุอาจารย์ที่ปรึกษา ช่วยให้สถาบันสามารถวิเคราะห์อายุของอาจารย์ที่ปรึกษา เพื่อวางแผนด้านทรัพยากรบุคคล เช่น การสรรหาบุคลากรใหม่ หรือการเตรียมแผนเกษียณอายุ

เพศ : สำหรับระบุเพศสภาพ เพื่อกรณีที่ต้องการทราบว่าอาจารย์ที่ปรึกษาเป็นเพศหญิงหรือชาย

ตำแหน่งวิชาการ : ระบุตำแหน่งทางวิชาการของอาจารย์ เช่น อาจารย์ ผู้ช่วยศาสตราจารย์ รองศาสตราจารย์ หรือศาสตราจารย์

ห้องทำงาน : หมายเลขห้องทำงานของอาจารย์ที่ปรึกษา ระบุเพื่อใช้เป็นข้อมูลติดต่อของอาจารย์ที่ปรึกษา

อีเมล : ใช้ในการติดต่อสื่อสารอาจารย์ที่ปรึกษาทางราชการ

เบอร์โทรติดต่อ : ใช้ในการติดต่อสื่อสารอาจารย์ที่ปรึกษาทางราชการในเวลาเร่งด่วน

ตารางอาจารย์ผู้สอน

รหัสอาจารย์ผู้สอน	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
TC1236	สาขาวิชาวิทยาการคอมพิวเตอร์	วินัย	วิเชียร	2	1
TC2345	สาขาวิชาการบัญชี	ปลาดุก	ชนเชื่อน	10	2
TC3456	สาขาวิชาการตลาด	มินิมง	สีเขียว	19	3
TC4567	สาขาวิชาคณิตศาสตร์ประยุกต์	จันทร์เพ็ญ	สมฤดี	24	4
TC5678	สาขาวิชาเทคโนโลยีสารสนเทศ	สมศรี	คำคง	28	5
TC1879	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	จิตลดี	บานชื่น	6	5
TC6945	สาขาวิศวกรรมโยธา	ภพ	อดีตชาติ	7	11

ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ
2529	M	LE	SC9-333	Winai.w@ku.th	9-7123-5789
2527	M	PA	SIL-102	Padong.c@ku.th	9-0123-3256
2528	F	LE	SC14-508	Minimong.s@ku.th	9-8006-1238
2526	F	AP	SIL-305	Chanpang.s@ku.th	8-9125-4863
2530	F	LE	SC9-402	Somsei.k@ku.th	8-9775-5423
2539	F	AP	HI-201	Jitladee.b@ku.th	9-8234-6890
2527	M	PA	AE-101	Phop.a@ku.th	6-8974-3241

ตารางอาจารย์ผู้สอน ประกอบด้วย Attributes ดังนี้

รหัสอาจารย์ผู้สอน : สำหรับทำการค้นหาข้อมูลของอาจารย์ผู้สอน 1 คน

ชื่อสาขาวิชา : เพื่อระบุข้อมูลสาขาวิชาที่อาจารย์ผู้สอนสังกัดอยู่

ชื่อ-นามสกุล : เพื่อให้ทราบว่าอาจารย์ผู้สอน มี ชื่อ-นามสกุล อะไรและ สามารถนำข้อมูลส่วนนี้ใช้ระบุตัวตนได้

วัน/เดือน/ปีเกิด : สำหรับระบุอายุอาจารย์ผู้สอน ช่วยให้สถาบันสามารถวิเคราะห์อายุของอาจารย์ผู้สอน เพื่อวางแผนด้านทรัพยากรบุคคล เช่น การสรรหาบุคลากรใหม่ หรือการเตรียมแผนเกษียณอายุ

เพศ : สำหรับระบุเพศสภาพ เพื่อกรณีที่ต้องการทราบว่าอาจารย์ผู้สอน เป็นเพศหญิงหรือชาย

ตำแหน่งวิชาการ : ระบุตำแหน่งทางวิชาการของอาจารย์ผู้สอน เช่น อาจารย์ ผู้ช่วยศาสตราจารย์ รองศาสตราจารย์ หรือศาสตราจารย์

ห้องทำงาน : หมายเลขห้องทำงานของอาจารย์ผู้สอน ระบุเพื่อใช้เป็นข้อมูลติดต่อของอาจารย์ที่ผู้สอน

อีเมล : ใช้ในการติดต่อสื่อสารอาจารย์ผู้สอนทางราชการ

เบอร์โทรติดต่อ : ใช้ในการติดต่อสื่อสารอาจารย์ผู้สอนทางราชการในเวลาเร่งด่วน

ตารางรายวิชาที่เปิดสอน

รหัสรายวิชาที่เปิดสอน	รหัสอาจารย์ผู้สอน	ชื่อรายวิชาที่เปิดสอน	หน่วยกิต	คำอธิบายรายวิชา
1418234	TC1236	การเขียนโปรแกรมเบื้องต้น	3	รายวิชานี้แนะนำแนวคิดพื้นฐานของวิทยาการคอมพิวเตอร์ รวมถึงโครงสร้างข้อมูลอัลกอริทึม และกระบวนทัศน์การเขียนโปรแกรม นักศึกษาจะได้เรียนรู้ทักษะการเขียนโปรแกรมพื้นฐานและเทคนิคการแก้ปัญหา
1418288	TC2345	คณิตศาสตร์สำหรับวิทยาการคอมพิวเตอร์	3	รายวิชานี้ครอบคลุมหลักการและแนวปฏิบัติของระบบจัดการฐานข้อมูล นักศึกษาจะได้เรียนรู้เกี่ยวกับการออกแบบฐานข้อมูล SQL และการดูแลระบบฐานข้อมูล
1417111	TC3456	แคลคูลัส1	3	รายวิชานี้สอนนักศึกษาถึงวิธีการพัฒนาเว็บแอปพลิเคชันแบบไดนามิกโดยใช้ HTML, CSS, JavaScript และเทคโนโลยีฝั่งเซิร์ฟเวอร์
1418284	TC4567	ฟิสิกส์เบื้องต้น	3	รายวิชานี้สำรวจโครงสร้างข้อมูลและอัลกอริทึมทั่วไป และการประยุกต์ใช้ในการพัฒนาซอฟต์แวร์ นักศึกษาจะได้เรียนรู้วิธีการวิเคราะห์และนำอัลกอริทึมที่มีประสิทธิภาพไปใช้
CHEM101	TC5678	เคมีเบื้องต้น	3	รายวิชานี้เป็นไปตามข้อกำหนดการศึกษาทั่วไป และให้นักศึกษาได้เข้าใจอย่างกว้างขวางเกี่ยวกับสาขาวิชาการต่างๆ
1418244	TC5678	การพัฒนาเว็บแอปพลิเคชัน	3	รายวิชานี้เน้นการพัฒนาเว็บแอปพลิเคชันด้วยเทคโนโลยีล่าสุด เช่น Node.js, React, และฐานข้อมูล NoSQL นักศึกษาจะได้เรียนรู้การออกแบบและพัฒนาเว็บที่มีประสิทธิภาพ
1418245	TC6945	การวิเคราะห์และออกแบบระบบ	3	รายวิชานี้ครอบคลุมแนวคิดในการวิเคราะห์และออกแบบระบบซอฟต์แวร์อย่างเป็นระบบ โดยใช้ UML และเทคนิคการออกแบบเชิงวัตถุ
1418246	TC1236	การเขียนโปรแกรมเชิงวัตถุ	3	รายวิชานี้เน้นการเขียนโปรแกรมด้วยแนวคิดเชิงวัตถุ เช่น การใช้คลาสและอ็อบเจ็กต์ การสืบทอด และการห่อหุ้มข้อมูล เพื่อการพัฒนาซอฟต์แวร์ที่มีความยืดหยุ่น
1418247	TC1236	ปัญญาประดิษฐ์เบื้องต้น	3	รายวิชานี้แนะนำแนวคิดพื้นฐานของปัญญาประดิษฐ์ เช่น การเรียนรู้ของเครื่อง การประมวลผลภาษาธรรมชาติ และการหาทางออกในปัญหาซับซ้อน

ปีการศึกษา	ภาคการศึกษา	ประเภทรายวิชาที่เปิดสอน	สถานะรายวิชาที่เปิดสอน	หมายเหตุ
2567	ภาคต้น	RC	WT	CP
2567	ภาคต้น	EL	CC	CP
2567	ภาคปลาย	RC	PC	NA
2567	ภาคปลาย	RC	OP	IP
2567	ภาคต้น	RC	CL	IP
2567	ภาคต้น	RC	CC	CP
2567	ภาคต้น	EL	CC	CP
2567	ภาคต้น	RC	CC	CP
2567	ภาคปลาย	RC	CC	CP

ตารางรายวิชาที่เปิดสอน ประกอบด้วย Attributes ดังนี้

รหัสรายวิชาที่เปิดสอน : สำหรับการค้นหาข้อมูลของรายวิชาที่เปิดสอน 1 รายวิชา

รหัสอาจารย์ผู้สอน : หมายเลขประจำตัวของอาจารย์ผู้สอน

ชื่อรายวิชาที่เปิดสอน : ชื่อทางการของรายวิชาตามที่กำหนดในหลักสูตร

หน่วยกิต : จำนวนหน่วยกิตที่ผู้เรียนจะได้รับเมื่อผ่านการศึกษาตามเกณฑ์ที่กำหนด

คำอธิบายรายวิชา : เพื่อเก็บคำอธิบายรายละเอียดเกี่ยวกับเนื้อหาของรายวิชา

ปีการศึกษา : ปีการศึกษาที่รายวิชานี้เปิดสอน

ภาคการศึกษา : ภาคการศึกษาที่รายวิชานี้เปิดสอน เช่น ภาคต้น หรือภาคปลาย

ประเภทรายวิชาที่เปิดสอน : หมวดหมู่ของรายวิชาตามลักษณะของหลักสูตร ซึ่งอาจแบ่งออกเป็นประเภทต่าง ๆ

สถานะรายวิชาที่เปิดสอน : สถานะของรายวิชาว่าอยู่ในหมวดหมู่เปิดสอน หรือไม่เปิดสอนในปีการศึกษานั้น

หมายเหตุ : ข้อมูลเพิ่มเติมเกี่ยวกับสถานะรายวิชาที่เปิดสอน

ตารางห้องเรียน

รหัสห้องเรียน	อาคาร	ชั้น	เลขห้อง	จำนวนที่นั่ง	ประเภทห้องเรียน
LH4-301	LH4	3	301	50	LR
LH2-303	LH2	3	303	40	LR
SC9-303	SC9	3	303	30	LB
SC4-502	SC4	5	502	30	LR
LH2-204	LH2	2	204	70	LR
LH4-201	LH4	2	201	60	LB
LH2-202	LH2	2	202	60	LR
LH3-307	LH3	3	307	100	LR
SC4-302	SC4	3	302	100	LB

ตารางห้องเรียน ประกอบด้วย Attributes ดังนี้

รหัสห้องเรียน : สำหรับทำการค้นหาข้อมูลของห้องเรียน 1 ห้อง

อาคาร : ชื่อหรือหมายเลขของอาคารที่ตั้งของห้องเรียน

ชั้น : ระดับชั้นที่ห้องเรียนตั้งอยู่ภายในอาคาร

เลขห้อง : หมายเลขประจำห้องเรียน

จำนวนที่นั่ง : จำนวนที่นั่งสูงสุดที่รองรับนิสิตได้ในห้องเรียน

ประเภทห้องเรียน : ลักษณะของห้องเรียน เช่น ห้องบรรยาย ห้องปฏิบัติการ

ตารางการชำระเงินค่าเทอม

หมายเลขการชำระเงินค่าเทอม	รหัสนิสิต	จำนวนเงิน	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน (พ.ศ.)	เวลาการชำระเงิน
B65234	6621600873	15,400.00	17	10	2567	9:23:00
B66982	6721600896	26,700.00	20	11	2567	7:55:00
B67894	6721604593	36,000.00	23	10	2567	11:25:00
B78965	6712594569	21,000.00	25	11	2567	12:43:00
B65214	6621600221	16,900.00	12	10	2567	15:20:00
B65215	6760078701	32,400.00	13	11	2567	11:20:00
B65216	6721790065	40,000.00	17	10	2567	10:25:00
B65217	6521730068	17,300.00	17	10	2567	11:12:00
B65218	6621900345	18,300.00	16	11	2567	11:30:00
B75941	6421876003	40,000.00	12	10	2567	10:36:00

ส่วนลด5A	จำนวนเงินสุทธิ	ช่องทางการชำระเงิน	การขอผ่อนผัน	กองทุนที่ได้รับ	สถานะการชำระเงินค่าเทอม
5,000.00	10,400.00	CA	NULL	NULL	SC
NULL	26,700.00	MB	NULL	NULL	SC
NULL	36,000.00	NL	NULL	SL	SC
NULL	21,000.00	NL	NULL	IC	SC
NULL	16,900.00	NULL	ได้รับ	NULL	DP
NULL	32,400.00	CA	NULL	NULL	SC
NULL	40,000.00	MB	NULL	NULL	SC
NULL	17,300.00	MB	NULL	NULL	SC
NULL	18,300.00	NULL	ได้รับ	NULL	PD
NULL	40,000.00	MB	NULL	NULL	SC

ตารางการชำระเงินค่าเทอม ประกอบด้วย Attributes ดังนี้

หมายเลขการชำระเงินค่าเทอม : สำหรับทำการค้นหาข้อมูลของการชำระเงินค่าเทอม 1 ครั้ง

รหัสนิสิต : หมายเลขประจำตัวของนิสิตที่ทำการชำระเงินค่าเทอม

จำนวนเงิน : ยอดเงินที่ต้องชำระก่อนหักส่วนลด

วัน/เดือน/ปีที่ชำระเงิน : วันที่ดำเนินการชำระเงินค่าเทอม เพื่อเป็นหลักฐานการทำธุรกรรม

เวลาที่ชำระเงิน : เวลาที่ทำธุรกรรมเสร็จสมบูรณ์ เพื่อเป็นหลักฐานการทำธุรกรรม

ส่วนลด5A : ส่วนลดที่ได้รับตามเงื่อนไขยกเว้นค่าหน่วยกิตของมหาวิทยาลัย

จำนวนสุทธิ : ยอดเงินสุทธิที่ต้องชำระหลังหักส่วนลด 5A

ช่องทางการชำระเงิน : วิธีการชำระเงิน เช่น ธนาคารระบบออนไลน์ เคาน์เตอร์บริการ หรือบัตรเครดิต/เดบิต

การขอผ่อนผัน : การขอเลื่อนหรือแบ่งชำระค่าเทอมตามเงื่อนไขของสถาบัน

กองทุนที่ได้รับ : เงินสนับสนุนหรือเงินกู้ยืม เช่น กยศ., กรอ., หรือทุนการศึกษาต่าง ๆ

สถานะการชำระเงินค่าเทอม : สถานะการดำเนินการ เช่น ชำระแล้ว ค้างชำระ หรือขอผ่อนผัน เป็นต้น เพื่อใช้ติดตามและยืนยันสถานะการชำระเงินค่าเทอมของนิสิต

ตารางการลงทะเบียนเรียน

รหัสการลงทะเบียนเรียน	รหัสรายวิชาที่เปิดสอน	หมายเลขการชำระเงินค่าเทอม	วันที่ลงทะเบียน	เดือนที่ลงทะเบียน
J36521	1418234	B65234	10	11
J37875	1418288	B66982	12	11
J62331	1264889	B67894	14	11
J78994	1481339	B78965	13	11
J88462	1418284	B65214	13	11
J88463	1418244	B65215	9	11
J88464	1418245	B65216	8	11
J88465	1418246	B65217	9	11
J88466	1418244	B65214	9	11
J88467	1418245	B65214	9	11

ปีที่ลงทะเบียน (พ.ศ.)	เวลาที่ลงทะเบียนเรียน	สถานะการลงทะเบียน	ผลการเรียน
2567	9:02:00	Add	N
2567	9:12:00	Add	N
2567	9:20:00	Drop	W
2567	9:05:00	Add	N
2567	9:03:00	Add	N
2567	11:02:00	Add	N
2567	11:13:00	Add	N
2567	11:24:00	Add	N
2567	13:22:00	Add	N
2567	13:30:00	Add	N

ตารางการลงทะเบียนเรียน ประกอบด้วย Attributes ดังนี้

รหัสการลงทะเบียนเรียน : สำหรับทำการค้นหาข้อมูลของการลงทะเบียนเรียน 1 ครั้ง

รหัสรายวิชาที่เปิดสอน : หมายเลขประจำรายวิชาที่นิสิตทำการลงทะเบียนเรียน

หมายเลขการชำระเงินค่าเทอม : หมายเลขอ้างอิงสำหรับการชำระค่าเทอมเพื่อตรวจสอบว่านิสิตคนไหนทำการลงทะเบียนเรียนและมีการชำระเงินค่าเทอมหรือไม่

วัน/เดือน/ปีที่ลงทะเบียน : วันที่ดำเนินการลงทะเบียนเรียน เพื่อใช้เป็นข้อมูลการลงทะเบียนในระบบ

เวลาที่ลงทะเบียนเรียน : เวลาที่ทำการลงทะเบียนเรียนเสร็จสิ้น เพื่อบันทึกเวลาที่ดำเนินการลงทะเบียน

สถานะการลงทะเบียน : สถานะของการลงทะเบียน เช่น เพิ่ม หรือ ถอน รายวิชาที่เปิดสอน เพื่อใช้ติดตามสถานะการลงทะเบียน

ผลการเรียน : คะแนนหรือเกรดที่ได้รับหลังสิ้นสุดภาคการศึกษา เพื่อใช้ประเมินผลการเรียนในแต่ละภาคการศึกษา

การวิเคราะห์หา Entity **และ** Attributes

- นิสิต
- หลักสูตร
- สาขาวิชา
- อาจารย์ที่ปรึกษา
- อาจารย์ผู้สอน
- รายวิชาที่เปิดสอน
- การจองห้องเรียน
- ห้องเรียน
- การชำระเงินค่าเทอม
- การลงทะเบียนเรียน

Key Attributes

- รหัสนิสิต
- รหัสหลักสูตร
- ชื่อสาขาวิชา
- รหัสอาจารย์ที่ปรึกษา
- รหัสอาจารย์ผู้สอน
- รหัสรายวิชาที่เปิดสอน
- รหัสการจองห้องเรียน
- รหัสห้องเรียน
- หมายเลขการชำระเงินค่าเทอม
- รหัสการลงทะเบียนเรียน

วิเคราะห์หาความสัมพันธ์ระหว่าง Entity

- **นิสิต กับ อาจารย์ที่ปรึกษา**
  - นิสิต 1 คน มีอาจารย์ที่ปรึกษาได้ 1 คน
  - อาจารย์ที่ปรึกษา 1 คน มีนิสิตในที่ปรึกษาได้หลายคน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M

- **นิสิต กับ สาขาวิชา**
  - นิสิต 1 คน มีสาขาวิชาได้ 1 สาขา
  - สาขาวิชา 1 สาขา มีนิสิตได้หลายคน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M

- **นิสิต กับ การชำระเงินค่าเทอม**
  - นิสิต 1 คน มีการชำระเงินค่าเทอม 1 ครั้งต่อเทอม
  - การชำระเงินค่าเทอม 1 ครั้ง ถูกชำระโดยนิสิต 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : 1

- **การชำระเงินค่าเทอม กับ การลงทะเบียนเรียน**
  - การชำระเงินค่าเทอม 1 ครั้ง มีการลงทะเบียนเรียนได้ 1 ครั้ง
  - การลงทะเบียนเรียน 1 ครั้ง มีการชำระเงินค่าเทอม 1 ครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ 1 : 1

- **การลงทะเบียนเรียน กับ รายวิชาที่เปิดสอน**
  - การลงทะเบียนเรียน 1 ครั้ง มีรายวิชาที่เปิดสอนได้หลายวิชา
  - รายวิชาที่เปิดสอน 1 วิชา มีการลงทะเบียนเรียนได้หลายครั้ง
  - ดังนั้น มีความสัมพันธ์แบบ M : M

- **อาจารย์ผู้สอน กับ รายวิชาที่เปิดสอน**
  - อาจารย์ผู้สอน 1 คน มีรายวิชาที่เปิดสอนได้หลายรายวิชา
  - รายวิชาที่เปิดสอน 1 รายวิชา มีอาจารย์ผู้สอนได้ 1 คน
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M

- **รายวิชาที่เปิดสอน กับ การจองห้องเรียน**
  - รายวิชาที่เปิดสอน 1 รายวิชา มีการจองห้องเรียนได้หลายครั้ง
  - การจองห้องเรียน 1 ครั้ง มีรายวิชาที่เปิดสอนจองได้ 1 รายวิชา
  - ดังนั้น เป็นความสัมพันธ์แบบ 1 : M

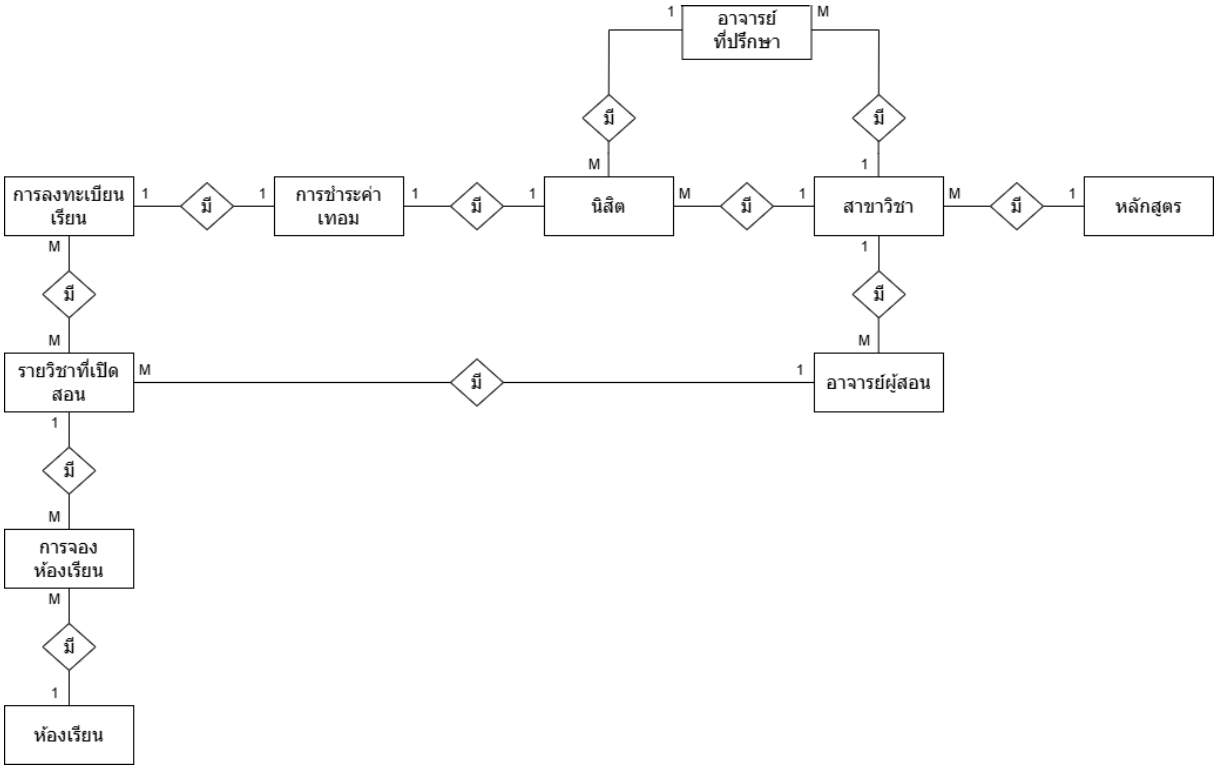
- **การจองห้องเรียน กับ ห้องเรียน**
  - การจองห้องเรียน 1 ครั้ง มีห้องเรียนได้ 1 ห้อง
  - ห้องเรียน 1 ห้อง มีการจองห้องเรียนได้หลายครั้ง
  - ดังนั้น เป็นความสัมพันธ์แบบ 1: M

- **หลักสูตร กับ สาขาวิชา**
  - หลักสูตร 1 หลักสูตร มีสาขาวิชาได้หลายสาขาวิชา
  - สาขาวิชา 1 สาขาวิชา มีหลักสูตรได้ 1 หลักสูตร
  - ดังนั้น เป็นความสัมพันธ์แบบ 1 : M

- **สาขาวิชา กับ อาจารย์ผู้สอน**
  - สาขาวิชา 1 สาขามีอาจารย์ผู้สอนได้หลายคน
  - อาจารย์ผู้สอน 1 คน มีสาขาวิชาได้ 1 สาขา
  - ดังนั้น มีความสัมพันธ์แบบ 1: M

- **สาขาวิชา กับ อาจารย์ที่ปรึกษา**
  - สาขาวิชา 1 สาขามีอาจารย์ที่ปรึกษาได้หลายคน
  - อาจารย์ที่ปรึกษา 1 คนมีสาขาวิชาที่ประจำได้ 1 สาขาวิชา
  - ดังนั้น มีความสัมพันธ์แบบ 1 : M

Entity Relationship Diagram



3. การแปลงโมเดลเชิงแนวคิด ER-Diagram เป็น Relational Database Model

นิสิต

รหัสนิสิต	ชื่อสาขาวิชา		รหัสอาจารย์ที่ปรึกษา	ชื่อ	นามสกุล	ภาคปกติ/ภาคพิเศษ	ชั้นปี	วันที่เกิด
เดือนที่เกิด	ปีที่เกิด	เพศ	อีเมล	เบอร์โทรติดต่อ	สถานะนิสิต	GPA	

หลักสูตร

รหัสหลักสูตร	ชื่อหลักสูตร	คณะ	ชื่อปริญญา	ระดับการศึกษา

สาขาวิชา

ชื่อสาขาวิชา	รหัสหลักสูตร	จำนวนหน่วยกิตรวมตลอดหลักสูตร(ไม่น้อยกว่า)	ปีที่ปรับปรุงหลักสูตร(พ.ศ
หมวดวิชาศึกษาทั่วไป(นก.)	หมวดวิชาเฉพาะเลือก(นก.)	ค่าเทอมภาคปกติ (บาท)	ค่าเทอมภาคพิเศษ (บาท)

อาจารย์ที่ปรึกษา

รหัสอาจารย์ที่ปรึกษา	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ

อาจารย์ผู้สอน

รหัสอาจารย์ผู้สอน	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ

รายวิชาที่เปิดสอน

รหัสรายวิชาที่เปิดสอน	รหัสอาจารย์ผู้สอน	ชื่อรายวิชาที่เปิดสอน	หน่วยกิต	คำอธิบายรายวิชา
ปีการศึกษา	ภาคการศึกษา	ประเภทรายวิชาที่เปิดสอน	สถานะรายวิชาที่เปิดสอน	หมายเหตุ

การจองห้องเรียน

รหัสการจองห้องเรียน	รหัสรายวิชาที่เปิดสอน	รหัสห้องเรียน	วันที่เรียน	เวลาเริ่ม
เวลาสิ้นสุด	หมู่เรียน	จำนวนนิสิต	สถานะการจองห้องเรียน	

ห้องเรียน

รหัสห้องเรียน	อาคาร	ชั้น	เลขห้อง	จำนวนที่นั่ง	ประเภทห้องเรียน

การชำระเงินค่าเทอม

หมายเลขการชำระเงินค่าเทอม	รหัสนิสิต	จำนวนเงิน	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน (พ.ศ.)	เวลาการชำระเงิน
ส่วนลด5A	จำนวนเงินสุทธิ	ช่องทางการชำระเงิน	การขอผ่อนผัน	กองทุนที่ได้รับ	สถานะการชำระเงินค่าเทอม	

การลงทะเบียนเรียน

รหัสการลงทะเบียนเรียน	รหัสรายวิชาที่เปิดสอน	หมายเลขการชำระเงินค่าเทอม	วันที่ลงทะเบียน	เดือนที่ลงทะเบีย
ปีที่ลงทะเบียน (พ.ศ.	เวลาที่ลงทะเบียนเรียน	สถานะการลงทะเบียน	ผลการเรียน	

4. การทำบรรทัดฐาน(Normalization)

ตารางนิสิต

รหัสนิสิต	ชื่อสาขาวิชา	รหัสอาจารย์ที่ปรึกษา	ชื่อ	นามสกุล	ภาคปกติ/ภาคพิเศษ	ชั้นปี
6621600221	สาขาวิชาวิทยาการคอมพิวเตอร์	TC0314	ศิริพร	อำไพพงศ์	ภาคปกติ	1
6621600873	สาขาวิชาการบัญชี	TC0314	ใจดี	บ่อ	ภาคปกติ	2
6721600896	สาขาวิชาการตลาด	TC0411	กะตอย	โชว์หวย	ภาคพิเศษ	1
6721604593	สาขาวิชาคณิตศาสตร์ประยุกต์	TC0512	ตะนอย	แซ่เลี้ยว	ภาคพิเศษ	1
6712594569	สาขาวิชาเทคโนโลยีสารสนเทศ	TC0519	จักกี้	โกโก้ครั้น	ภาคปกติ	3
6760078701	สาขาสัตวบาล	TC0314	แสตนด์	จ้พพ์	ภาคพิเศษ	1
6721790065	สาขาวิทยาการคอมพิวเตอร์	TC0315	สิวลักษณ์	ชินวัตร	ภาคพิเศษ	1
6521730068	สาขาวิศวกรรมโยธา	TC0167	ตั๊ก	เชิญธง	ภาคปกติ	3
6421876003	สาขาวิทยาการคอมพิวเตอร์	TC1124	ทองแดง	จิตต์ดี	ภาคพิเศษ	4
6621900345	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	TC2101	แฟ่ง	ไร่แตง	ภาคปกติ	2

วันที่เกิด	เดือนที่เกิด	ปีที่เกิด	เพศ	อีเมล	เบอร์โทรติดต่อ	สถานะนิสิต	GPA
4	12	2549	F	Siripon.a@ku.th	9-8293-6201	ST	3.22
6	2	2548	F	Jaidee.b@ku.th	9-2567-1388	ST	2.54
8	10	2548	M	Katoy.ch@ku.th	8-7453-2169	LO	1.85
4	2	2547	M	Tanoy.s@ku.th	9-1257-8910	ST	2.89
13	8	2546	M	Jakkee.k@ku.th	8-6253-4915	ST	3.48
31	10	2548	M	Stan.j@ku.th	987685438	ST	3.72
24	3	2549	F	Siwaluk.ch@ku.th	879654321	ST	3.11
23	3	2546	M	Tak.ch@ku.th	618916905	ST	2.59
10	6	2543	M	Tongdang.j@ku.th	864537881	LO	2.98
15	4	2547	F	Fang.R@ku.th	653598723	ST	3.8

1NF : ตารางนี้มีรหัสนิสิตเป็นคีย์หลัก (PK) และเป็น Attributes เป็น Atomic Value และ ทุก Attributes ขึ้นตรงกับคีย์หลักอย่างสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางหลักสูตร

รหัสหลักสูตร	ชื่อหลักสูตร	คณะ	ชื่อปริญญา	ระดับการศึกษา
CO4003	หลักสูตรวิทยาศาสตรบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	วิทยาศาสตรบัณฑิต	B
CO6505	หลักสูตรบริหารธุรกิจบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	บริหารธุรกิจบัณฑิต	B
CO6505	หลักสูตรคณิตศาสตรบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	คณิตศาสตรบัณฑิต	B
CO4506	หลักสูตรวิทยาศาสตรบัณฑิต	วิทยาศาสตร์การกีฬา	วิทยาศาสตรบัณฑิต	B
CO4808	หลักสูตรวิทยาศาสตรมหาบัณฑิต	ศิลปศาสตร์และวิทยาศาสตร์	วิทยาศาสตรมหาบัณฑิต	M
CO0165	หลักสูตรวิศวกรรมศาสตรบัณฑิต	วิศวกรรมศาสตร์	วิศวกรรมศาสตรบัณฑิต	B
CO2156	หลักสูตรศิลปศาสตรบัณฑิต	อุตสาหกรรมบริการ	ศิลปศาสตรบัณฑิต	B
CO0867	หลักสูตรศึกษาศาสตรบัณฑิต	ศึกษาศาสตร์	ศึกษาศาสตรบัณฑิต	B

1NF : ตารางนี้มีรหัสหลักสูตรเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางสาขาวิชา

ชื่อสาขาวิชา	รหัสหลักสูตร	จำนวนหน่วยกิตรวมตลอดหลักสูตร(ไม่น้อยกว่า)	ปีที่ปรับปรุงหลักสูตร(พ.ศ.)	หมวดวิชาศึกษาทั่วไป(นก.)	หมวดวิชาเฉพาะเลือก(นก.)	ค่าเทอมภาคปกติ (บาท)	ค่าเทอมภาคพิเศษ (บาท)
สาขาวิชาวิทยาการคอมพิวเตอร์	CO4003	124	2565	30	23	16,900.00	40,000.00
สาขาวิชาการบัญชี	CO6505	130	2565	30	24	15,400.00	28,200.00
สาขาวิชาการตลาด	CO6505	124	2565	30	35	15,400.00	26,700.00
สาขาวิชาคณิตศาสตร์ประยุกต์	CO4506	124	2565	30	28	21,000.00	36,000.00
สาขาวิชาเทคโนโลยีสารสนเทศ	CO4808	126	2564	30	24	21,000.00	34,700.00
สาขาวิศวกรรมโยธา	CO0165	126	2565	30	28	17,300.00	32,400.00
สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	CO2156	124	2565	30	26	18,300.00	33,400.00
สาขาสัตวบาล	CO4003	128	2564	30	30	17,300.00	32,400.00
สาขาพลศึกษา	CO8067	130	2564	30	94	13,300.00	26,000.00

1NF : ตารางนี้มีสาขาวิชาเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางอาจารย์ที่ปรึกษา

รหัสอาจารย์ที่ปรึกษา	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด	ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ
TC0314	สาขาวิชาวิทยาการคอมพิวเตอร์	ใจดี	มีดัง	1	5	2531	F	PA	SC9-330	Jaidee.me@ku.th	6-5552-2134
TC0315	สาขาวิชาการบัญชีบริหาร	สงยศ	สุขสม	2	8	2539	M	LE	SC1-321	Songyod.s@ku.th	9-8645-1235
TC0411	สาขาวิชาการตลาด	มั่นคง	มั่งคั่ง	3	10	2537	M	AP	SC9-440	Mangkong.m@ku.th	6-5128-7895
TC0512	สาขาวิชาคณิตศาสตร์ประยุกต์	ใฝ่สอน	ขดเขย	4	12	2529	F	LE	SC5-123	Faison.ch@ku.th	9-8645-1278
TC0519	สาขาวิชาเทคโนโลยีสารสนเทศ	สมสาย	ผนงเดช	5	2	2528	M	LE	SC9-232	Somsai.p@ku.th	9-0456-2879
TC0234	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	จิตลดี	บานชื่น	6	5	2539	F	AP	SL2-201	Jitladee.b@ku.th	9-8234-6890
TC0923	สาขาวิศวกรรมโยธา	ภพ	อดีตชาติ	7	11	2527	M	PA	SE2-101	Phop.a@ku.th	6-8974-3241
TC0318	สาขาสัตวบาล	วัวน้อย	คอยรัก	10	7	2525	M	PA	ASS-101	Wuanoi.k@ku.th	8-9564-7821

1NF : ตารางนี้มีรหัสอาจารย์ที่ปรึกษาเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางอาจารย์ผู้สอน

รหัสอาจารย์ผู้สอน	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
TC1236	สาขาวิชาวิทยาการคอมพิวเตอร์	วินัย	วิเชียร	2	1
TC2345	สาขาวิชาการบัญชี	ปลาดุก	ชนเชื่อน	10	2
TC3456	สาขาวิชาการตลาด	มินิมง	สีเขียว	19	3
TC4567	สาขาวิชาคณิตศาสตร์ประยุกต์	จันทร์เพ็ญ	สมฤดี	24	4
TC5678	สาขาวิชาเทคโนโลยีสารสนเทศ	สมศรี	คำคง	28	5
TC1879	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	จิตลดี	บานชื่น	6	5
TC6945	สาขาวิศวกรรมโยธา	ภพ	อดีตชาติ	7	11

ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ
2529	M	LE	SC9-333	Winai.w@ku.th	9-7123-5789
2527	M	PA	SIL-102	Padong.c@ku.th	9-0123-3256
2528	F	LE	SC14-508	Minimong.s@ku.th	9-8006-1238
2526	F	AP	SIL-305	Chanpang.s@ku.th	8-9125-4863
2530	F	LE	SC9-402	Somsei.k@ku.th	8-9775-5423
2539	F	AP	HI-201	Jitladee.b@ku.th	9-8234-6890
2527	M	PA	AE-101	Phop.a@ku.th	6-8974-3241

1NF : ตารางนี้มีรหัสอาจารย์ผู้สอนเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการจองห้องเรียน

รหัสการจองห้องเรียน	รหัสรายวิชาที่เปิดสอน	รหัสห้องเรียน	วันที่เรียน	เวลาเริ่ม	เวลาสิ้นสุด	หมู่เรียน	จำนวนนิสิต	สถานะการจองห้องเรียน
JH1489	1418234	LH4-301	Mon	9:00:00	12:00:00	800	80	SC
JH1490	1418282	LH2-303	Tue	14:00:00	17:00:00	702	50	UF
JH1491	1418299	SC9-303	Fri	8:30:00	10:30:00	703	100	SC
JH1492	CHEM101	SC4-502	Wed	13:00:00	16:00:00	700	70	UF
JH1493	1418284	LH2-204	Sat	17:00:00	20:00:00	701	98	SC
JH1494	1418244	LH4-201	Mon	13:00:00	4:00:00	700	50	SC
JH1495	1418245	LH2-202	Tue	14:00:00	3:00:00	700	60	SC
JH1496	1418246	LH3-307	Fri	9:00:00	13:00:00	800	70	SC
JH1497	1418247	SC4-302	Sun	8:00:00	9:30:00	802	70	SC

1NF : ตารางนี้มีรหัสการจองห้องเรียนเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางรายวิชาที่เปิดสอน

รหัสรายวิชาที่เปิดสอน	รหัสอาจารย์ผู้สอน	ชื่อรายวิชาที่เปิดสอน	หน่วยกิต	คำอธิบายรายวิชา
1418234	TC1236	การเขียนโปรแกรมเบื้องต้น	3	รายวิชานี้แนะนำแนวคิดพื้นฐานของวิทยาการคอมพิวเตอร์ รวมถึงโครงสร้างข้อมูลอัลกอริทึม และกระบวนทัศน์การเขียนโปรแกรม นักศึกษาจะได้เรียนรู้ทักษะการเขียนโปรแกรมพื้นฐานและเทคนิคการแก้ปัญหา
1418288	TC2345	คณิตศาสตร์สำหรับวิทยาการคอมพิวเตอร์	3	รายวิชานี้ครอบคลุมหลักการและแนวปฏิบัติของระบบจัดการฐานข้อมูล นักศึกษาจะได้เรียนรู้เกี่ยวกับการออกแบบฐานข้อมูล SQL และการดูแลระบบฐานข้อมูล
1417111	TC3456	แคลคูลัส1	3	รายวิชานี้สอนนักศึกษาถึงวิธีการพัฒนาเว็บแอปพลิเคชันแบบไดนามิกโดยใช้ HTML, CSS, JavaScript และเทคโนโลยีฝั่งเซิร์ฟเวอร์
1418284	TC4567	ฟิสิกส์เบื้องต้น	3	รายวิชานี้สำรวจโครงสร้างข้อมูลและอัลกอริทึมทั่วไป และการประยุกต์ใช้ในการพัฒนาซอฟต์แวร์ นักศึกษาจะได้เรียนรู้วิธีการวิเคราะห์และนำอัลกอริทึมที่มีประสิทธิภาพไปใช้
CHEM101	TC5678	เคมีเบื้องต้น	3	รายวิชานี้เป็นไปตามข้อกำหนดการศึกษาทั่วไป และให้นักศึกษาได้เข้าใจอย่างกว้างขวางเกี่ยวกับสาขาวิชาการต่างๆ
1418244	TC5678	การพัฒนาเว็บแอปพลิเคชัน	3	รายวิชานี้เน้นการพัฒนาเว็บแอปพลิเคชันด้วยเทคโนโลยีล่าสุด เช่น Node.js, React, และฐานข้อมูล NoSQL นักศึกษาจะได้เรียนรู้การออกแบบและพัฒนาเว็บที่มีประสิทธิภาพ
1418245	TC6945	การวิเคราะห์และออกแบบระบบ	3	รายวิชานี้ครอบคลุมแนวคิดในการวิเคราะห์และออกแบบระบบซอฟต์แวร์อย่างเป็นระบบ โดยใช้ UML และเทคนิคการออกแบบเชิงวัตถุ
1418246	TC1236	การเขียนโปรแกรมเชิงวัตถุ	3	รายวิชานี้เน้นการเขียนโปรแกรมด้วยแนวคิดเชิงวัตถุ เช่น การใช้คลาสและอ็อบเจ็กต์ การสืบทอด และการห่อหุ้มข้อมูล เพื่อการพัฒนาซอฟต์แวร์ที่มีความยืดหยุ่น
1418247	TC1236	ปัญญาประดิษฐ์เบื้องต้น	3	รายวิชานี้แนะนำแนวคิดพื้นฐานของปัญญาประดิษฐ์ เช่น การเรียนรู้ของเครื่อง การประมวลผลภาษาธรรมชาติ และการหาทางออกในปัญหาซับซ้อน

ปีการศึกษา	ภาคการศึกษา	ประเภทรายวิชาที่เปิดสอน	สถานะรายวิชาที่เปิดสอน	หมายเหตุ
2567	ภาคต้น	RC	WT	CP
2567	ภาคต้น	EL	CC	CP
2567	ภาคปลาย	RC	PC	NA
2567	ภาคปลาย	RC	OP	IP
2567	ภาคต้น	RC	CL	IP
2567	ภาคต้น	RC	CC	CP
2567	ภาคต้น	EL	CC	CP
2567	ภาคต้น	RC	CC	CP
2567	ภาคปลาย	RC	CC	CP

1NF : ตารางนี้มีรหัสรายวิชาที่เปิดสอนเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางห้องเรียน

รหัสห้องเรียน	อาคาร	ชั้น	เลขห้อง	จำนวนที่นั่ง	ประเภทห้องเรียน
LH4-301	LH4	3	301	50	LR
LH2-303	LH2	3	303	40	LR
SC9-303	SC9	3	303	30	LB
SC4-502	SC4	5	502	30	LR
LH2-204	LH2	2	204	70	LR
LH4-201	LH4	2	201	60	LB
LH2-202	LH2	2	202	60	LR
LH3-307	LH3	3	307	100	LR
SC4-302	SC4	3	302	100	LB

1NF : ตารางนี้มีรหัสห้องเรียนเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการชำระเงินค่าเทอม

หมายเลขการชำระเงินค่าเทอม	รหัสนิสิต	จำนวนเงิน	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน (พ.ศ.)	เวลาการชำระเงิน
B65234	6621600873	15,400.00	17	10	2567	9:23:00
B66982	6721600896	26,700.00	20	11	2567	7:55:00
B67894	6721604593	36,000.00	23	10	2567	11:25:00
B78965	6712594569	21,000.00	25	11	2567	12:43:00
B65214	6621600221	16,900.00	12	10	2567	15:20:00
B65215	6760078701	32,400.00	13	11	2567	11:20:00
B65216	6721790065	40,000.00	17	10	2567	10:25:00
B65217	6521730068	17,300.00	17	10	2567	11:12:00
B65218	6621900345	18,300.00	16	11	2567	11:30:00
B75941	6421876003	40,000.00	12	10	2567	10:36:00

ส่วนลด5A	จำนวนเงินสุทธิ	ช่องทางการชำระเงิน	การขอผ่อนผัน	กองทุนที่ได้รับ	สถานะการชำระเงินค่าเทอม
5,000.00	10,400.00	CA	NULL	NULL	SC
NULL	26,700.00	MB	NULL	NULL	SC
NULL	36,000.00	NL	NULL	SL	SC
NULL	21,000.00	NL	NULL	IC	SC
NULL	16,900.00	NULL	ได้รับ	NULL	DP
NULL	32,400.00	CA	NULL	NULL	SC
NULL	40,000.00	MB	NULL	NULL	SC
NULL	17,300.00	MB	NULL	NULL	SC
NULL	18,300.00	NULL	ได้รับ	NULL	PD
NULL	40,000.00	MB	NULL	NULL	SC

1NF : ตารางนี้มีหมายเลขการชำระเงินค่าเทอมเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

ตารางการลงทะเบียนเรียน

รหัสการลงทะเบียนเรียน	รหัสรายวิชาที่เปิดสอน	หมายเลขการชำระเงินค่าเทอม	วันที่ลงทะเบียน	เดือนที่ลงทะเบียน
J36521	1418234	B65234	10	11
J37875	1418288	B66982	12	11
J62331	1264889	B67894	14	11
J78994	1481339	B78965	13	11
J88462	1418284	B65214	13	11
J88463	1418244	B65215	9	11
J88464	1418245	B65216	8	11
J88465	1418246	B65217	9	11
J88466	1418244	B65214	9	11
J88467	1418245	B65214	9	11

ปีที่ลงทะเบียน (พ.ศ.)	เวลาที่ลงทะเบียนเรียน	สถานะการลงทะเบียน	ผลการเรียน
2567	9:02:00	Add	N
2567	9:12:00	Add	N
2567	9:20:00	Drop	W
2567	9:05:00	Add	N
2567	9:03:00	Add	N
2567	11:02:00	Add	N
2567	11:13:00	Add	N
2567	11:24:00	Add	N
2567	13:22:00	Add	N
2567	13:30:00	Add	N

1NF : ตารางนี้มีรหัสการลงทะเบียนเรียนเป็นคีย์หลัก (PK) และเป็น Atomic Value และทุก Attributes อื่นในตารางที่ไม่ใช่คีย์หลัก ขึ้นกับคีย์หลักโดยสมบูรณ์

2NF : ตารางนี้อยู่ในรูปแบบ 1NF แล้ว และไม่มีคีย์ร่วมจึงไม่มีขึ้นต่อกันเพียงบางส่วน

3NF : ตารางนี้อยู่ในรูปแบบ 2NF แล้ว และไม่มีการขึ้นต่อกันแบบ Transitive

New Relational Database Model

นิสิต

รหัสนิสิต	ชื่อสาขาวิชา	รหัสอาจารย์ที่ปรึกษา	ชื่อ	นามสกุล	ภาคปกติ/ภาคพิเศษ	ชั้นปี	วันที่เกิด
เดือนที่เกิด	ปีที่เกิด	เพศ	อีเมล	เบอร์โทรติดต่อ	สถานะนิสิต	GPA	

หลักสูตร

รหัสหลักสูตร	ชื่อหลักสูตร	คณะ	ชื่อปริญญา	ระดับการศึกษา

สาขาวิชา

ชื่อสาขาวิชา	รหัสหลักสูตร	จำนวนหน่วยกิตรวมตลอดหลักสูตร(ไม่น้อยกว่า)	ปีที่ปรับปรุงหลักสูตร(พ.ศ
หมวดวิชาศึกษาทั่วไป(นก.)	หมวดวิชาเฉพาะเลือก(นก.)	ค่าเทอมภาคปกติ (บาท)	ค่าเทอมภาคพิเศษ (บาท)

อาจารย์ที่ปรึกษา

รหัสอาจารย์ที่ปรึกษา	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ

อาจารย์ผู้สอน

รหัสอาจารย์ผู้สอน	ชื่อสาขาวิชา	ชื่อ	นามสกุล	วันที่เกิด	เดือนที่เกิด
ปีที่เกิด	เพศ	ตำแหน่งวิชาการ	ห้องทำงาน	อีเมล	เบอร์โทรติดต่อ

รายวิชาที่เปิดสอน

รหัสรายวิชาที่เปิดสอน	รหัสอาจารย์ผู้สอน	ชื่อรายวิชาที่เปิดสอน	หน่วยกิต	คำอธิบายรายวิชา
ปีการศึกษา	ภาคการศึกษา	ประเภทรายวิชาที่เปิดสอน	สถานะรายวิชาที่เปิดสอน	หมายเหตุ

การจองห้องเรียน

รหัสการจองห้องเรียน	รหัสรายวิชาที่เปิดสอน	รหัสห้องเรียน	วันที่เรียน	เวลาเริ่ม
เวลาสิ้นสุด	หมู่เรียน	จำนวนนิสิต	สถานะการจองห้องเรียน	

ห้องเรียน

รหัสห้องเรียน	อาคาร	ชั้น	เลขห้อง	จำนวนที่นั่ง	ประเภทห้องเรียน

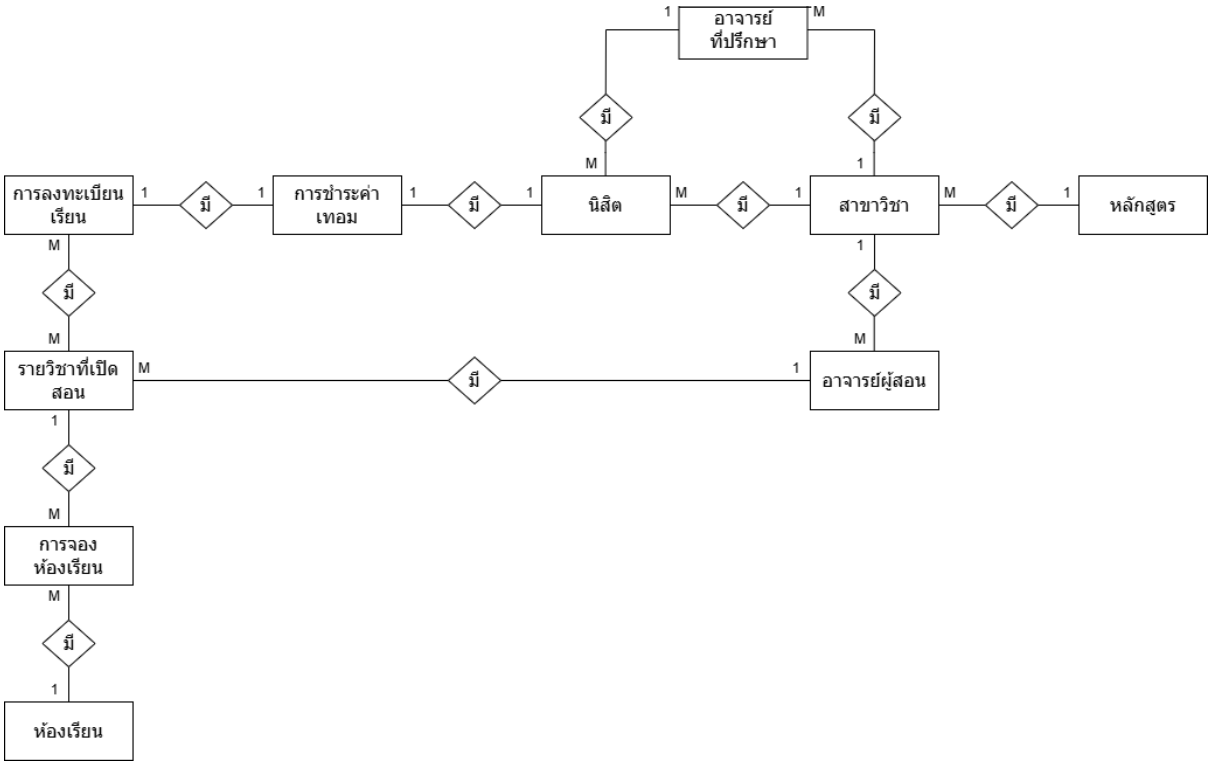
การชำระเงินค่าเทอม

หมายเลขการชำระเงินค่าเทอม	รหัสนิสิต	จำนวนเงิน	วันที่ชำระเงิน	เดือนที่ชำระเงิน	ปีที่ชำระเงิน (พ.ศ.)	เวลาการชำระเงิน
ส่วนลด5A	จำนวนเงินสุทธิ	ช่องทางการชำระเงิน	การขอผ่อนผัน	กองทุนที่ได้รับ	สถานะการชำระเงินค่าเทอม	

การลงทะเบียนเรียน

รหัสการลงทะเบียนเรียน	รหัสรายวิชาที่เปิดสอน	หมายเลขการชำระเงินค่าเทอม	วันที่ลงทะเบียน	เดือนที่ลงทะเบีย
ปีที่ลงทะเบียน (พ.ศ.	เวลาที่ลงทะเบียนเรียน	สถานะการลงทะเบียน	ผลการเรียน	

New Entity Relationship Diagram



5.พจนานุกรมข้อมูล(Data Dictionary)

ตารางนิสิต

Attributes	Data Type	Description
ID_Nisit (รหัสนิสิต)	CHAR(10)	รหัสประจำตัวนิสิต 10 หลัก
Major_name (ชื่อสาขาวิชา)	VARCHAR(50)	ชื่อสาขาวิชาที่นิสิตกำลังศึกษาอยู่
ID_Advisor (รหัสอาจารย์ที่ปรึกษา)	CHAR(7)	รหัสอาจารย์ที่ปรึกษาของนิสิต 6 หลัก
First_name (ชื่อ)	VARCHAR(20)	ชื่อของนิสิต
Last_name (นามสกุล)	VARCHAR(20)	นามสกุลของนิสิต
Program_Type (ภาคปกติ/ภาคพิเศษ)	VARCHAR(2)	ภาคปกติ/ภาคพิเศษ - RP = ภาคปกติ - SP = ภาคพิเศษ
Year_Level (ชั้นปี)	TINYINT	ชั้นปีที่กำลังศึกษา(1-8)
Birth_Day (วันที่เกิด)	TINYINT	วันที่เกิดของนิสิต (1-31)
Birth_Month (เดือนที่เกิด)	TINYINT	เดือนเกิดของนิสิต (1-12)
Birth_Year (ปีที่เกิด)	SMALLINT	ปีพุทธศักราชที่นิสิตเกิด
Gender (เพศ)	CHAR(1)	ระบุเพศ - M-ชาย - F-หญิง
Email (อีเมล)	VARCHAR(40)	อีเมลติดต่อของนิสิต
Phone_Number (เบอร์โทรติดต่อ)	VARCHAR(10)	เบอร์โทรศัพท์ของนิสิต 10 หลัก
Nisit_Status (สถานะนิสิต)	VARCHAR(2)	สถานะของนิสิต - ST = Studying (กำลังศึกษา) - LO = Leave of Absence (พักการศึกษา) - DS = Dismissed (พ้นสภาพนิสิต)
GPA (เกรดเฉลี่ย)	DECIMAL(3,2)	เกรดเฉลี่ยสะสมของนิสิต ที่มีจำนวนทั้งหมด 3 หลัก เป็นทศนิยม 2 หลัก และจำนวนเต็ม 1 หลัก เช่น 3.22

ตารางหลักสูตร

Attributes	Data Type	Description
ID_Curriculum (รหัสหลักสูตร)	CHAR(6)	รหัสหลักสูตร 6 หลัก
Curriculum_Name (ชื่อหลักสูตร)	VARCHAR(50)	ชื่อเต็มของหลักสูตร
Faculty_Name (คณะ)	VARCHAR(50)	ชื่อคณะในหลักสูตร
Degree_Name (ชื่อปริญญา)	VARCHAR(50)	ชื่อปริญญาที่ได้รับเมื่อจบการศึกษา
Education_Level (ระดับการศึกษา)	CHAR(1)	ระดับการศึกษา - B = ปริญญาตรี - M = ปริญญาโท

		- D = ปริญญาเอก

ตารางสาขาวิชา

Attributes	Data Type	Description
Major_Name (ชื่อสาขาวิชา)	VARCHAR(50)	ชื่อเต็มสาขาวิชา
ID_Curriculum (รหัสหลักสูตร)	CHAR(6)	รหัสหลักสูตรที่ดูแล 6 หลัก
Total_Credits (จำนวนหน่วยกิตรวมตลอดหลักสูตร)	CHAR(3)	จำนวนหน่วยกิตรวมตลอดหลักสูตร
Last_Updated_Year (ปีที่ปรับปรุงหลักสูตร)	CHAR(4)	ปีพุทธศักราชที่ปรับปรุงหลักสูตร
General_Education_Credits (หมวดวิชาศึกษาทั่วไป)	CHAR(2)	จำนวนหน่วยกิตรวมที่ต้องเรียนของวิชาในหมวดวิชาศึกษาทั่วไปในสาขานั่นๆ
Elective_Credits (หมวดวิชาเฉพาะเลือก)	CHAR(2)	จำนวนหน่วยกิตรวมที่ต้องเรียนของวิชาในหมวดวิชาเฉพาะเลือกในสาขานั่นๆ
Regular_Tuition_Fee (ค่าเทอมภาคปกติ)	CHAR(6)	จำนวนเงินค่าเทอมภาคปกติ (บาท)
Special_Tuition_Fee (ค่าเทอมภาคพิเศษ)	CHAR(6)	จำนวนเงินค่าเทอมภาคพิเศษ (บาท)

ตารางอาจารย์ที่ปรึกษา

Attributes	Data Type	Description
ID_Advisor (รหัสอาจารย์ที่ปรึกษา)	CHAR(6)	รหัสของอาจารย์ที่ปรึกษา 6 หลัก
Major_Name (ชื่อสาขาวิชา)	VARCHAR(50)	ชื่อสาขาวิชาที่อาจารย์ที่ปรึกษาสังกัดอยู่
First_Name (ชื่อ)	VARCHAR(20)	ชื่ออาจารย์ที่ปรึกษา
Last_Name (นามสกุล)	VARCHAR(20)	นามสกุลอาจารย์ที่ปรึกษา
Birth_Day (วันที่เกิด)	TINYINT	วันที่เกิดของอาจารย์ (1-31)
Birth_Month (เดือนที่เกิด)	TINYINT	เดือนเกิดของอาจารย์ (1-12)
Birth_Year (ปีที่เกิด)	SMALLINT	ปีพุทธศักราชที่อาจารย์ที่ปรึกษาเกิด
Gender (เพศ)	CHAR(1)	ระบุเพศ - M-ชาย - F-หญิง
Academic_Title (ตำแหน่งวิชาการ)	VARCHAR(2)	ตำแหน่งวิชาการของอาจารย์ผู้สอน - PF = ศาสตราจารย์ - AP = รองศาสตราจารย์ - PA = ผู้ช่วยศาสตราจารย์ - LE = อาจารย์
Office_Room (ห้องทำงาน)	CHAR(8)	หมายเลขห้องทำงานของอาจารย์

Emai (อีเมล)	VARCHAR(40)	อีเมลติดต่ออาจารย์ที่ปรึกษา
Phone_Number (เบอร์โทรติดต่อ)	CHAR(10)	เบอร์โทรศัพท์ของอาจารย์ 10 หลัก

ตารางอาจารย์ผู้สอน

Attributes	Data Type	Description
ID_Instructor (รหัสอาจารย์ผู้สอน)	CHAR(6)	รหัสประจำตัวของอาจารย์ผู้สอน 6 หลัก
Major_Name (ชื่อสาวิชา)	VARCHAR(50)	ชื่อสาขาวิชาที่อาจารย์ที่ผู้สอนสังกัดอยู่
First_Name (ชื่อ)	VARCHAR(20)	ชื่ออาจารย์ผู้สอน
Last_Name (นามสกุล)	VARCHAR(20)	นามสกุลอาจารย์ผู้สอน
Birth_Day (วันที่เกิด)	TINYINT	วันที่เกิดของอาจารย์ผู้สอน (1-31)
Birth_Month (เดือนที่เกิด)	TINYINT	เดือนเกิดของอาจารย์ผู้สอน (1-12)
Birth_Year (ปีที่เกิด)	SMALLINT	ปีพุทธศักราชที่อาจารย์ผู้สอนเกิด
Gender (เพศ)	CHAR(1)	ระบุเพศ - M-ชาย - F-หญิง
Academic_Title (ตำแหน่งวิชาการ)	VARCHAR(2)	ตำแหน่งวิชาการของอาจารย์ผู้สอน - PF =ศาสตราจารย์ - AP = รองศาสตราจารย์ - PA = ผู้ช่วยศาสตราจารย์ - LE = อาจารย์
Office_Room (ห้องทำงาน)	VARCHAR(8)	ห้องทำงานของอาจารย์ผู้สอน
Email (อีเมล)	CHAR(13)	อีเมลของอาจารย์ผู้สอน
Phone_Number (เบอร์ติดต่อ)	VARCHAR(10)	เบอร์โทรศัพท์ของอาจารย์ผู้สอน 10 หลัก

ตารางรายวิชาที่เปิดสอน

Attributes	Data Type	Description
ID_Course (รหัสรายวิชาที่เปิดสอน)	CHAR(10)	รหัสเฉพาะของรายวิชาที่เปิดสอน 12 หลัก
ID_Instructor (รหัสอาจารย์ผู้สอน)	CHAR(6)	รหัสอาจารย์ที่สอนรายวิชาที่เปิดสอน 6 หลัก
Course_Name (ชื่อรายวิชาที่เปิดสอน)	VARCHAR(100)	ชื่อของรายวิชาที่เปิดสอน
Credits (หน่วยกิต)	INT	จำนวนหน่วยกิตของรายวิชาที่เปิดสอน
Course_Description (คำอธิบายรายวิชา)	TEXT	คำอธิบายเนื้อหาของรายวิชาที่เปิดสอน
Academic_Year (ปีการศึกษา)	INT	ปีพุทธศักราชที่รายวิชาเปิดสอน
Semester (ภาคการศึกษา)	VARCHAR(20)	ภาคการศึกษาที่เปิดสอน - ภาคต้น - ภาคปลาย - ภาคฤดูร้อน
Course_Type (ประเภทรายวิชาที่เปิดสอน)	CHAR(2)	ประเภทของรายวิชา - RC = Required Course (วิชาบังคับ) - EL = Elective Course (วิชาเลือก) - FE = Free Elective (วิชาเสรี) - FC = Foundation Course (วิชาพื้นฐาน) - LC = Language Course (วิชาภาษา) - LS = Life Skills Course (วิชาทักษะชีวิต) - LB = Laboratory/Practical Course (วิชาปฏิบัติการ)
Course_Status (สถานะรายวิชาที่เปิดสอน)	CHAR(2)	สถานะของรายวิชา - OP = เปิดสอน - CL = ปิดสอน - CC = ยกเลิก - PC = กำลังดำเนินการ - WT = รออนุมัติ
Notes (หมายเหตุ)	TEXT	คำอธิบายสถานะของรายวิชาที่เปิดสอน - CP = ดำเนินการเสร็จสิ้น (รายวิชาที่เปิดสอนสามารถเปิดสอนได้ตามเงื่อนไข) - IP = ยังไม่เสร็จสิ้น (รายวิชาที่เปิดสอนยังอยู่ในการดำเนินการเปิดสอน)

		- NA = ไม่ผ่านเงื่อนไข (รายวิชาที่เปิดสอนไม่ผ่านเงื่อนไข ไม่สามารถเปิดสอนได้)

ตารางการจองห้องเรียน

Attributes	Data Type	Description
ID_Booking (รหัสการจองห้องเรียน)	CHAR(6)	รหัสเฉพาะของการจองห้องเรียน 6 หลัก
ID_Course (รหัสรายวิชาที่เปิดสอน)	CHAR(10)	รหัสของรายวิชาที่ต้องการจองห้อง 12 หลัก
Classroom_ID (รหัสห้องเรียน)	CHAR(7)	รหัสเฉพาะของห้องเรียน 7 หลัก
Class_Day (วันที่เรียน)	CHAR(3)	วันที่เรียนของวิชา - Sun = วันอาทิตย์ - Mon = วันจันทร์ - Tue = วันอังคาร - Wed = วันพุธ - Thu = วันพฤหัสบดี - Fri = วันศุกร์ - Sat = วันเสาร์
Start_Time (เวลาเริ่ม)	TIME	เวลาที่เริ่มเรียนของรายวิชา
End_Time (เวลาสิ้นสุด)	TIME	เวลาที่สิ้นสุดการเรียน
Section (หมู่เรียน)	CHAR(3)	หมู่เรียนที่ใช้ห้องเรียน
Nisit_Count (จำนวนนิสิต)	CHAR(3)	จำนวนนิสิตที่เข้าเรียน
Booking_Status (สถานะการจองห้องเรียน)	VARCHAR(2)	สถานะการจองห้อง - SC = Successful (จองห้องเรียนสำเร็จ) - UF = Unsuccessful/Failed (จองไม่สำเร็จกรณีที่ห้องเรียนไม่ว่างหรือติดปัญหาทางเทคนิคอื่นๆ)

ตารางห้องเรียน

Attributes	Data Type	Description
Classroom_ID (รหัสห้องเรียน)	CHAR(7)	รหัสเฉพาะของห้องเรียน 7 หลัก
Building (อาคาร)	VARCHAR(10)	ชื่ออาคารที่ตั้งของห้องเรียน
Floor (ชั้น)	CHAR(1)	ชั้นของอาคารที่ตั้งห้องเรียน
Room_Number (เลขห้อง)	CHAR(3)	เลขที่ห้องเรียน
Seat_Capacity (จำนวนที่นั่ง)	CHAR(3)	จำนวนที่นั่งในห้องเรียน
Classroom_Type (ประเภทห้องเรียน)	VARCHAR(2)	ประเภทของห้องเรียน - LR = Lecture Room (ห้องเรียนบรรยาย) - LB = Laboratory/Practical Room (ห้องเรียนปฏิบัติการ)

ตารางการชำระเงินค่าเทอม

Attributes	Data Type	Description
ID_Payment (หมายเลขการชำระเงินค่าเทอม)	CHAR(6)	รหัสเฉพาะของการชำระเงินค่าเทอม 6 หลัก
ID_Nisit (รหัสนิสิต)	CHAR(10)	รหัสประจำตัวของนิสิต 10 หลัก
Amount (จำนวนเงิน)	INT	จำนวนเงินที่ต้องชำระ (บาท)
Payment_Day (วันที่ชำระเงิน)	TINYINT	วันที่ทำการชำระเงินค่าเทอม(1-31)
Payment_Month (เดือนที่ชำระเงิน)	TINYINT	เดือนที่ทำการชำระเงินค่าเทอม (1-12)
Payment_Year (ปีที่ชำระเงิน)	SMALLINT	ปีพุทธศักราชที่ทำการชำระเงินค่าเทอม
Payment_Time (เวลาการชำระเงิน)	TIME	เวลาที่ทำการชำระเงินค่าเทอม
Discount_5A (ส่วนลด5A)	INT	จำนวนเงินส่วนลดที่ได้จากโครงการที่มอบส่วนลดค่าหน่วยกิตให้กับนิสิตที่ ได้เกรด A ในรายวิชาไม่น้อยกว่า 5 รายวิชา รวมกันไม่น้อยกว่า 15 หน่วยกิต ในภาคการศึกษาที่กำหนด จำนวนเงิน 5,000 บาท
Net_Amount (จำนวนเงินสุทธิ)	INT	ยอดชำระเต็มหลังจากหักส่วนลด 5A ที่ได้รับ
Payment_Method (ช่องทางการชำระเงิน)	VARCHAR(2)	ช่องทางการชำระเงินค่าเทอม - CA = Cash (เงินสด) - MB = Mobile Banking (ธนาคารระบบออนไลน์) - CC = Credit Card (บัตรเครดิต)

		- CH = (Cashier's Check) (เคาน์เตอร์บริการ) - BK = (Bank Transfer/Payment at Bank) (ธนาคาร) - NL = Nist Loan (เงินกู้)
Deferment_Status (การขอผ่อนผัน)	VARCHAR(8)	สถานะการขอผ่อนผันค่าเทอมของนิสิต - ได้รับ = ได้รับอนุมัติการผ่อนผัน
Fund_Received (กองทุนที่ได้รับ)	VARCHAR(2)	กองทุนที่นิสิตได้รับ - SL = ได้รับกองทุนกยศ. - IC = ได้รับกองทุนกรอ.
Payment_Status (สถานะการชำระเงินเงินค่าเทอม)	VARCHAR(2)	สถานะการชำระเงินค่าเทอม - SC = Successful (ชำระเงินสำเร็จ) - U/F = Unsuccessful/Failed (ยังไม่ชำระเงิน/ชำระเงินไม่สำเร็จ) - PD = Pending (รอดำเนินการ / อยู่ระหว่างตรวจสอบ) - RP = Rejected/Refunded (ถูกปฏิเสธ / มีการคืนเงิน) - DP = Deferred Payment (ผ่อนผัน)

ตารางการลงทะเบียนเรียน

Attributes	Data Type	Description
ID_Registration (รหัสการลงทะเบียนเรียน)	CHAR(6)	รหัสเฉพาะของการลงทะเบียน 6 หลัก
ID_Course (รหัสรายวิชาที่เปิดสอน)	CHAR(10)	รหัสของรายวิชาที่ลงทะเบียน 12 หลัก
ID_Payment (หมายเลขการชำระเงินค่าเทอม)	CHAR(6)	รหัสอ้างอิงการชำระเงินค่าเทอม 6 หลัก
Registration_Day (วันที่ลงทะเบียน)	TINYINT	วันที่ทำการลงทะเบียน(1-31)
Registration_Month (เดือนที่ลงทะเบียนเรียน)	TINYINT	เดือนที่ทำการลงทะเบียน (1-12)
Registration_Year (ปีที่ลงทะเบียน)	SMALLINT	ปีพุทธศักราชที่ลงทะเบียน
Registration_Time (เวลาที่ลงทะเบียนเรียน)	TIME	เวลาที่ทำการลงทะเบียน
Registration_Status (ผลการเรียน)	VARCHAR(4)	สถานะการลงทะเบียน - Add เพิ่ม (เพิ่มการลงทะเบียนรายวิชาที่เปิดสอน) - Drop ถอน (ถอนการลงทะเบียนรายวิชาที่เปิดสอน)

Grade (เกรดเฉลี่ย)	CHAR(2)	ผลการเรียนของรายวิชาที่เปิดสอน - A = 4.00 - B+ = 3.50 - B = 3.00 - C+ = 2.50 - C = 2.00 - D+ = 1.50 - D = 1.00 - F = 0.00 - W = Withdrawal (ถอนรายวิชา) - N = Grade not reported (ยังไม่ทราบระดับคะแนน) - I = Incomplete (ยังไม่สมบูรณ์)

6.การสร้างฐานข้อมูลด้วย MySQL Workbench

การสร้างตาราง **นิสิต** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางนิสิตขึ้นมา

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Nisit เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Major_Name เป็น FK

และอ้างอิงจากตาราง Major

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Advisor เป็น FK

และอ้างอิงจากตาราง Advisor

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Nisit แสดงผลของตารางที่ได้ผ่านการ Query

```
1  • CREATE TABLE Nisit (
2        ID_Nisit CHAR(10) PRIMARY KEY NOT NULL,
3        Major_Name VARCHAR(50) NOT NULL,
4        ID_Advisor CHAR(7) NOT NULL,
5        First_Name VARCHAR(20) NOT NULL,
6        Last_Name VARCHAR(20) NOT NULL,
7        Program_type VARCHAR(2) NOT NULL,
8        Year_Level TINYINT NOT NULL,
9        Birth_Day TINYINT NOT NULL,
10       Birth_Month TINYINT NOT NULL,
11       Birth_Year SMALLINT NOT NULL,
12       Gender CHAR(1) NOT NULL,
13       Email VARCHAR(40) NOT NULL,
14       Phone_Number VARCHAR(10) NOT NULL,
15       Nisit_Status VARCHAR(2) NOT NULL,
16       GPA DECIMAL(3,2) NOT NULL,
17       FOREIGN KEY (Major_Name) REFERENCES Major(Major_Name),
18       FOREIGN KEY (ID_Advisor) REFERENCES Advisor(ID_Advisor)
19   );
20 • INSERT INTO Nisit VALUES
21       ('6621608221', วิทยาการคอมพิวเตอร์, 'TC3814', ศิริพร, อำไพทองค์, 'RP', 2, 4, 12, 2549, 'F', 'Siriporn.@eku.th', '9-8293-6221', 'ST', 3.22),
22       ('6621608873', วิทยาศาสตร์ชีวภัณฑ์, 'TC3814', ใจดี, ดีมาก, 'RP', 2, 6, 2, 2548, 'F', 'Jaidee.@eku.th', '9-2567-1388', 'ST', 2.54),
23       ('6721680895', วิทยาศาสตร์คลัง, 'TC8411', คลินิ, เจกิต, 'SP', 1, 8, 10, 2548, 'M', 'Katoy.@chku.th', '8-7453-2169', 'LO', 1.85),
24       ('6721645937', วิทยาศาสตร์สิ่งแวดล้อม, 'TC5214', ทอง, แผ่นทอง, 'SP', 1, 4, 2, 2547, 'M', 'Tong.@eku.th', '9-1257-8910', 'ST', 2.89),
25       ('6712549569', วิทยาการสิ่งแวดล้อม, 'TC3818', แจ็คกี้, กิกพ, 'RP', 3, 18, 3, 2546, 'M', 'Jakkee.@eku.th', '8-6253-4915', 'ST', 3.48),
26       ('6712970865', วิทยาศาสตร์การอาหาร, 'TC3818', สตแน, เจ็ด, 'SP', 1, 31, 10, 2548, 'M', 'Stan.@eku.th', '9-8768-5438', 'ST', 3.72),
27       ('672179085', วิทยาศาสตร์ชีวภาพ, 'TC3817', สิวลักษณ์, ชินสีห์, 'SP', 1, 24, 3, 2549, 'F', 'Siwaluk.@chku.th', '8-7965-4321', 'ST', 3.11),
28       ('6721528958', วิทยาศาสตร์ชีวเวชภาพ, 'TC6617', เต้ก, ชัย, 'RP', 3, 23, 3, 2546, 'M', 'Takk.@chku.th', '6-1891-6965', 'ST', 2.59),
29       ('6421870632', วิทยาศาสตร์โภชนาการ, 'TC1124', ทองแดง, เล็ก, 'SP', 4, 16, 6, 2543, 'M', 'Tongdang.@eku.th', '8-6453-7881', 'LO', 2.98),
30       ('6621826931', วิทยาศาสตร์ภาษาภาพ, 'TC1201', ฟาง, ริน, 'RP', 2, 15, 4, 2547, 'F', 'Fang.@eku.th', '6-5359-8723', 'ST', 3.8);
31 • SELECT * FROM Nisit;
```

ID_Nisit	Major_Name	ID_Advisor	First_Name	Last_Name	Program_Type	Year_Level	Birth_Day	Birth_Month	Birth_Year	Gender	Email	Phone_Number	Nisit_Status	GPA
6621600221	สาขาวิชาวิทยาการคอมพิวเตอร์	TC0314	ศิริพร	อำไพทองค์	RP	2	4	2	2549	F	Siripon.a@ku.th	864537881	ST	3.22
6621600873	สาขาวิชาการบัญชี	TC0314	ใจดี	บ่อ	RP	2	6	12	2548	F	Jaidee.b@ku.th	618916905	ST	2.54
6721600896	สาขาวิชาการตลาด	TC0411	กะตอย	โชว์หวย	SP	1	8	10	2548	M	Katoy.ch@ku.th	982936201	LO	1.85
6721604593	สาขาวิชาคณิตศาสตร์ประยุกต์	TC0512	ตะนอย	แว่นเลื่อว	SP	1	4	2	2547	M	Tanoy.s@ku.th	925671388	ST	2.89
6712594569	สาขาวิชาเทคโนโลยีสารสนเทศ	TC0519	จักกี้	โกโก้ครั้น	RP	3	13	8	2546	M	Jakkee.k@ku.th	653598723	ST	3.48
6760078701	สาขาสัตวบาล	TC0318	แสตนด์	จิพพ์	SP	1	31	10	2548	M	Stan.j@ku.th	862534915	ST	3.72
6721790065	สาขาวิชาการคอมพิวเตอร์	TC0315	สิวลักษณ์	ชินวัตร	SP	1	24	3	2549	F	Siwaluk.ch@ku.th	874532169	ST	3.11
6521730068	สาขาวิศวกรรมโยธา	TC0167	เต้ก	เชี่ยยง	RP	3	23	3	2546	M	Tak.ch@ku.th	912578910	ST	2.59
6421876003	สาขาวิชาการคอมพิวเตอร์	TC1124	ทองแดง	จิตต์ดี	SP	4	10	6	2543	M	Tongdang.j@ku.th	879654321	LO	2.98
6621900345	สาขาภาษาอังกฤษเพื่อธุรกิจสากกรรมบริการ	TC2101	แฟง	ไจ้นดง	RP	2	15	4	2547	F	Fang.R@ku.th	987685438	ST	3.80

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางนิสิต

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_birth_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนเกิด (Birth_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่เกิด (Birth_Day) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่เกิด (Birth _Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)
2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)
3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เกิด ไม่เกิน 29

```
ALTER TABLE nisit
ADD CONSTRAINT chk_birth_date
CHECK (
    Birth_Month BETWEEN 1 AND 12
    AND (
        (Birth_Month IN (1, 3, 5, 7, 8, 10, 12) AND Birth_Day BETWEEN 1 AND 31)
        OR (Birth_Month IN (4, 6, 9, 11) AND Birth_Day BETWEEN 1 AND 30)
        OR (Birth_Month = 2 AND Birth_Day BETWEEN 1 AND 29)
    )
);
```

ทดสอบการใส่ค่า โดยใส่วันที่ 30 เดือน 2 ซึ่งโดยที่กำหนดไว้ เดือนที่ 2 สามารถเพิ่มค่าได้ไม่เกิน 29 วัน ทำให้ไม่สามารถเพิ่มค่าได้

```
1 • INSERT INTO Nisit VALUES
2   ('6621900345', 'สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'TC2101', 'แฟง', 'ไร่แตง', 'RP', 2, 31, 2, 2547, 'F', 'Fang.R@ku.th', '653598723', 'ST', 3.80);
```

Output

Action Output

#	Time	Action	Message
1	01:52:57	INSERT INTO Nisit VALUES ('6621900345', 'สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'TC2101', 'แฟง', 'ไร่แตง', 'RP', ...	Error Code: 3819. Check constraint 'chk_birth_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางนิสิต

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_nisit_gender

CHECK เพศ(Gender) กำหนดค่า 2 ค่า คือ M - Male ผู้ชาย และ F - Female ผู้หญิง

```
1 •  ALTER TABLE Nisit
2    ADD CONSTRAINT chk_nisit_gender
3    CHECK (Gender IN('M','F'));
```

ทดลองใส่ค่าเป็น D ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร M,F เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO Nisit (ID_Nisit, Major_Name, ID_Advisor, First_Name, Last_Name, Year_Level, Birth_Day, Birth_Month, Birth_Year, Gender, Email, Phone_Number, Nisit_Status, GPA)
2    VALUES ('66210845', สาขาวิชาการปัญญาประดิษฐ์, 'TC0411', แฟง, ร้อนผ่า, 2, 15, 4, 2547, 'D', 'Fang.R@ku.th', '65598723', 'ST', 3.88);
3
```

Output

Action Output

#	Time	Action	Message
1	18:48:48	INSERT INTO Nisit (ID_Nisit, Major_Name, ID_Advisor, First_Name, Last_Name, Year_Level, Birth_Day, Birth_Month, ...	Error Code: 3819. Check constraint 'chk_nisit_gender' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางนิสิต

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Nisit_Status

CHECK สถานภาพนิสิต(Nisit_Stasus) กำหนดค่า 3 ค่า คือ ST = Studying (กำลังศึกษา), LO = Leave of Absence (พักการศึกษา), DS = Dismissed (พ้นสภาพนิสิต)

```
1 •  ALTER TABLE Nisit
2    ADD CONSTRAINT chk_Nisit_Status
3    CHECK (Nisit_Status IN ('ST', 'LO', 'DS'));
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร ST,LO,DS เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO Nisit VALUES ('6421876103', สาขาวิชาการคอมพิวเตอร์, 'TC1124', ทองดี, จิตดี, 'SP', 4, 10, 6, 2543, 'M', 'Tongdee.j@ku.th', '804537881', 'XX', 2.98);
2
```

Output

Action Output

#	Time	Action	Message
1	02:20:56	INSERT INTO Nisit VALUES ('6421876103', 'สาขาวิชาการคอมพิวเตอร์', 'TC1124', 'ทองดี', 'จิตดี', 'SP', 4, 10, 6, 2543,...	Error Code: 3819. Check constraint 'chk_Nisit_Status' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางนิสิต

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Program_Type

CHECK ภาคปกติ/ภาคพิเศษ(Program_Type) กำหนดค่า 2 ค่า คือ RP = ภาคปกติ และ SP = ภาคพิเศษ

```
1 •  ALTER TABLE Nisit
2    ADD CONSTRAINT chk_Program_Type
3    CHECK (Program_Type IN ('RP', 'SP'));
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร RP,SP เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO Nisit VALUES
2    ('6621900745', 'สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'TC2101', 'แฟง', 'ไร่แดง', 'XX', 2, 15, 4, 2547, 'F', 'Fang.R@ku.th', '653598723', 'ST', 3.8);
```

Output

Action Output

#	Time	Action	Message
1	23:15:25	INSERT INTO Nisit VALUES ('6621900745', 'สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'TC2101', 'แฟง', 'ไร่แดง', 'XX', ...	Error Code: 3819. Check constraint 'chk_Program_Type' is violated.

การสร้างตาราง **หลักสูตร** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางหลักสูตรขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Curriculum เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Curriculum แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Curriculum (
2      ID_Curriculum CHAR(6) PRIMARY KEY NOT NULL,
3      Curriculum_Name VARCHAR(50) NOT NULL,
4      Faculty_Name VARCHAR(50) NOT NULL,
5      Degree_Name VARCHAR(50) NOT NULL,
6      Education_Level CHAR(1) NOT NULL
7  );
8  INSERT INTO Curriculum VALUES
9      ('CO4003', 'หลักสูตรวิทยาศาสตรบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'วิทยาศาสตรบัณฑิต', 'B'),
10     ('CO6505', 'หลักสูตรบริหารธุรกิจบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'บริหารธุรกิจบัณฑิต', 'B'),
11     ('CO6506', 'หลักสูตรคณิตศาสตรบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'คณิตศาสตรบัณฑิต', 'B'),
12     ('CO4506', 'หลักสูตรวิทยาศาสตรการกีฬาบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'วิทยาศาสตรการกีฬาบัณฑิต', 'B'),
13     ('CO4808', 'หลักสูตรวิทยาศาสตรมหาบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'วิทยาศาสตรมหาบัณฑิต', 'M'),
14     ('CO0165', 'หลักสูตรวิศวกรรมศาสตรบัณฑิต', 'วิศวกรรมศาสตร์', 'วิศวกรรมศาสตรบัณฑิต', 'B'),
15     ('CO2156', 'หลักสูตรศิลปศาสตรบัณฑิต', 'อุตสาหกรรมบริการ', 'ศิลปศาสตรบัณฑิต', 'B'),
16     ('CO0867', 'หลักสูตรศึกษาศาสตรบัณฑิต', 'ศึกษาศาสตร์', 'ศึกษาศาสตรบัณฑิต', 'B');
17 SELECT * FROM Curriculum;
```

ID_Curriculum	Curriculum_Name	Faculty	Degree_Name	Education_Level
CO0165	หลักสูตรวิศวกรรมศาสตรบัณฑิต	วิศวกรรมศาสตร์	วิศวกรรมศาสตรบัณฑิต	B
CO0867	หลักสูตรศึกษาศาสตรบัณฑิต	ศึกษาศาสตร์	ศึกษาศาสตรบัณฑิต	B
CO2156	หลักสูตรศิลปศาสตรบัณฑิต	อุตสาหกรรมบริการ	ศิลปศาสตรบัณฑิต	B
CO2307	หลักสูตรวิทยาศาสตรบัณฑิต	ศิลปศาสตร์และวิทยาศาตร์	วิทยาศาสตรบัณฑิต	B
CO4003	หลักสูตรวิทยาศาสตรบัณฑิต	ศิลปศาสตร์และวิทยาศาตร์	วิทยาศาสตรบัณฑิต	B
CO4506	หลักสูตรคณิตศาสตร์บัณฑิต	ศิลปศาสตร์และวิทยาศาตร์	คณิตศาสตร์บัณฑิต	B
CO4808	หลักสูตรวิทยาศาสตรมหาบัณฑิต	ศิลปศาสตร์และวิทยาศาตร์	วิทยาศาสตรมหาบัณฑิต	M
CO6505	หลักสูตรบริหารธุรกิจบัณฑิต	ศิลปศาสตร์และวิทยาศาตร์	บริหารธุรกิจบัณฑิต	B

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางหลักสูตร

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Education_Level

CHECK ระดับการศึกษา(Education_Level) กำหนดค่า 3 ค่า คือ B = ปริญญาตรี , M = ปริญญาโท , D = ปริญญาเอก

```
1 •  ALTER TABLE Curriculum
2    ADD CONSTRAINT chk_education_level
3    CHECK (Education_Level IN ('B', 'M', 'D'));
4
```

ทดลองใส่ค่าเป็น Z ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร B,M,D เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1  INSERT INTO Curriculum VALUES
2      ('CO4009', 'หลักสูตรวิทยาศาสตการกลรบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'วิทยาศาสตรบัณฑิต', 'Z');
```

Output

Action Output

#	Time	Action	Message
1	02:39:27	INSERT INTO Curriculum VALUES ('CO4009', 'หลักสูตรวิทยาศาสตการกลรบัณฑิต', 'ศิลปศาสตร์และวิทยาศาสตร์', 'วิท...	Error Code: 3819. Check constraint 'chk_education_level' is violated.

การสร้างตาราง **สาขาวิชา** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางสาขาวิชาขึ้นมา

PRIMARY KEY เพิ่มตัว PK โดยในตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Major_Name เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Curriculum เป็น FK และอ้างอิงจากตาราง Curriculum

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Major แสดงผลของตารางที่ได้ผ่านการ Query

```sql
CREATE TABLE Major (
    Major_Name VARCHAR(50) PRIMARY KEY NOT NULL,
    ID_Curriculum CHAR(6) NOT NULL,
    Total_Credits CHAR(3) NOT NULL,
    Last_Updated_Year CHAR(4) NOT NULL,
    General_Education_Credits CHAR(2) NOT NULL,
    Elective_Credits CHAR(2) NOT NULL,
    Regular_Tuition_Fee CHAR(6) NOT NULL,
    Special_Tuition_Fee CHAR(6) NOT NULL,
    FOREIGN KEY (ID_Curriculum) REFERENCES Curriculum (ID_Curriculum)
);
INSERT INTO Major VALUES
    ('สาขาวิชาการตลาด', 'CO6505', '124', '2565', '30', '35', '15,400', '26,700'),
    ('สาขาวิชาการบัญชีบริหาร', 'CO6505', '130', '2565', '30', '24', '15,400', '28,200'),
    ('สาขาวิชาคณิตศาสตร์ประยุกต์', 'CO4506', '124', '2565', '30', '28', '21,000', '36,000'),
    ('สาขาวิชาวิทยาการคอมพิวเตอร์', 'CO4003', '124', '2565', '30', '23', '16,900', '40,000'),
    ('สาขาวิชาเทคโนโลยีสารสนเทศ', 'CO4808', '126', '2564', '30', '24', '21,000', '34,700'),
    ('สาขาวิศวกรรมโยธา', 'CO0165', '126', '2565', '30', '28', '17,300', '32,400.00'),
    ('สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'CO2156', '124', '2565', '30', '26', '18,300', '33,400.00'),
    ('สาขาสัตวบาล', 'CO4003', '128', '2564', '30', '30', '17,300', '32,400.00'),
    ('สาขาพลศึกษา', 'CO0867', '130', '2564', '30', '94', '13,300', '26,000.00');
SELECT * FROM Major;
```

Major_Name	ID_Curriculum	Total_Credits	Last_Updated_Year	General_Education_Credits	Elective_Credits	Regular_Tuition_Fee	Special_Tuition_Fee
สาขาพลศึกษา	CO0867	130	2564	30	94	13300	26000
สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	CO2156	124	2565	30	26	18300	33400
สาขาวิชาการตลาด	CO6505	124	2565	30	35	15400	26700
สาขาวิชาการบัญชีบริหาร	CO6505	130	2565	30	24	15400	28200
สาขาวิชาคณิตศาสตร์ประยุกต์	CO4506	124	2565	30	28	21000	36000
สาขาวิชาวิทยาการคอมพิวเตอร์	CO4003	124	2565	30	23	16900	40000
สาขาวิชาเทคโนโลยีสารสนเทศ	CO4808	126	2564	30	24	21000	34700
สาขาวิศวกรรมโยธา	CO0165	126	2565	30	28	17300	32400
สาขาสัตวบาล	CO4003	128	2564	30	30	17300	32400

การสร้างตาราง **อาจารย์ที่ปรึกษา** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางอาจารย์ที่ปรึกษาขึ้นมา

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Advisor เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Major_Name เป็น FK และอ้างอิงจากตาราง Major

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Advisor แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Advisor (
2      ID_Advisor CHAR(6) PRIMARY KEY NOT NULL,
3      Major_Name VARCHAR(50) NOT NULL,
4      First_Name VARCHAR(20) NOT NULL,
5      Last_Name VARCHAR(20) NOT NULL,
6      Birth_Day TINYINT NOT NULL,
7      Birth_Month TINYINT NOT NULL,
8      Birth_Year SMALLINT NOT NULL,
9      Gender CHAR(1) NOT NULL,
10     Academic_Title VARCHAR(2) NOT NULL,
11     Office_Room VARCHAR(8) NOT NULL,
12     Email VARCHAR(40) NOT NULL,
13     Phone_Number CHAR(10) NOT NULL,
14   FOREIGN KEY (Major_Name) REFERENCES Major (Major_Name) ); INSERT INTO Advisor VALUES
15   ('TC0314', 'สาขาวิชาวิทยาการคอมพิวเตอร์', 'ใจดี', 'มีดัง', 1, 5, 2531, 'F', 'PA', 'SC9-330', 'Jaidee.me@ku.th', '655522134'),
16   ('TC0315', 'สาขาวิชาการบัญชีบริหาร', 'สงยศ', 'สุขสม', 2, 8, 2539, 'M', 'LE', 'SC1-321', 'Songyod.s@ku.th', '986451235'),
17   ('TC0411', 'สาขาวิชาการตลาด', 'มั่นคง', 'มังคัง', 3, 10, 2537, 'M', 'AP', 'SC9-440', 'Mangkong.m@ku.th', '651287895'),
18   ('TC0512', 'สาขาวิชาคณิตศาสตร์ประยุกต์', 'ไม่สอน', 'ชัดเขย', 4, 12, 2529, 'F', 'LE', 'SC5-123', 'Faison.ch@ku.th', '986451278'),
19   ('TC0519', 'สาขาวิชาเทคโนโลยีสารสนเทศ', 'สมสาย', 'ผนงเลข', 5, 2, 2528, 'M', 'LE', 'SC9-232', 'Somsai.p@ku.th', '904562879'),
20   ('TC0234', 'สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'จิตลดี', 'บานชื่น', 6, 5, 2539, 'F', 'AP', 'SL2-201', 'Jitladee.b@ku.th', '982346890'),
21   ('TC0923', 'สาขาวิศวกรรมโยธา', 'ภพ', 'อดิศราดี', 7, 11, 2527, 'M', 'PA', 'SE2-101', 'Phop.a@ku.th', '689743241'),
22   ('TC0318', 'สาขาสัตวบาล', 'วีรน้อย', 'คอยรัก', 10, 7, 2525, 'M', 'PA', 'ASS-101', 'Wuanoi.k@ku.th', '895647821');
23   SELECT * FROM Advisor;
```

ID_Advisor	Major_Name	First_Name	Last_Name	Birth_Day	Birth_Month	Birth_Year	Gender	Academic_Title	Office_Room	Email	Phone_Number
TC0314	สาขาวิชาวิทยาการคอมพิวเตอร์	ใจดี	มีดัง	1	5	2531	F	PA	SC9-330	Jaidee.me@ku.th	655522134
TC0315	สาขาวิชาการบัญชีบริหาร	สงยศ	สุขสม	2	8	2539	M	LE	SC1-321	Songyod.s@ku.th	986451235
TC0411	สาขาวิชาการตลาด	มั่นคง	มังคัง	3	10	2537	M	AP	SC9-440	Mangkong.m@ku.th	651287895
TC0512	สาขาวิชาคณิตศาสตร์ประยุกต์	ไม่สอน	ชัดเขย	4	12	2529	F	LE	SC5-123	Faison.ch@ku.th	986451278
TC0519	สาขาวิชาเทคโนโลยีสารสนเทศ	สมสาย	ผนงเลข	5	2	2528	M	LE	SC9-232	Somsai.p@ku.th	904562879
TC0234	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	จิตลดี	บานชื่น	6	5	2539	F	AP	SL2-201	Jitladee.b@ku.th	982346890
TC0923	สาขาวิศวกรรมโยธา	ภพ	อดิศราดี	7	11	2527	M	PA	SE2-101	Phop.a@ku.th	689743241
TC0318	สาขาสัตวบาล	วีรน้อย	คอยรัก	10	7	2525	M	PA	ASS-101	Wuanoi.k@ku.th	895647821

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างตารางอาจารย์ที่ปรึกษา

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_advisorbirth_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนเกิด (Birth_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่เกิด (Birth_Day) อยู่ในช่วง 1 ถึง 31 หรือไม่ หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่เกิด (Birth _Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม ฆมีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เกิด ไม่เกิน 29

```
ALTER TABLE  advisor
ADD CONSTRAINT chk_advisorbirth_date
CHECK (
    Birth_Month BETWEEN 1 AND 12
    AND (
        (Birth_Month IN (1, 3, 5, 7, 8, 10, 12) AND Birth_Day BETWEEN 1 AND 31)
        OR (Birth_Month IN (4, 6, 9, 11) AND Birth_Day BETWEEN 1 AND 30)
        OR (Birth_Month = 2 AND Birth_Day BETWEEN 1 AND 29)
    )
);
```

ทดสอบการใส่ค่า โดยใส่วันที่ 30 เดือน 2 ซึ่งโดยที่กำหนดไว้ เดือนที่ 2 สามารถเพิ่มค่าได้ไม่เกิน 29 วันทำให้ไม่สามารถเพิ่มค่าได้

```
INSERT INTO advisor VALUES
    ('TC0718', 'สายสัตวบาล', 'วัวน้อย', 'คอยรัก', 30, 2, 2525, 'M', 'PA', 'ASS-101', 'Wuanoi.k@ku.th', '895647821'); ;
```

Output

Action Output

#	Time	Action	Message
1	03:38:33	INSERT INTO advisor VALUES ('TC0718', 'สายสัตวบาล', 'วัวน้อย', 'คอยรัก', 30, 2, 2525, 'M', 'PA', 'ASS-101', 'Wua...	Error Code: 3819. Check constraint 'chk_advisorbirth_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางอาจารย์ที่ปรึกษา

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_advisor_gender

CHECK เพศ(Gender) กำหนดค่า 2 ค่า คือ M - Male ผู้ชาย และ F - Female ผู้หญิง

```
ALTER TABLE advisor
ADD CONSTRAINT chk_advisor_gender
CHECK (Gender IN('M','F'));
```

ทดลองใส่ค่าเป็น X ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร M,F เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO advisor VALUES
2     ('TC0712', 'QC0024', 'ไสมาศ', 'แน่นหนา', 4, 12, 2529, 'X', 'LE', 'SCS-123', 'Faison.ch@ku.th', '986278')   ;
```

Output

Action Output

#	Time	Action	Message
1	03:51:33	INSERT INTO advisor VALUES ('TC0712', 'QC0024', 'ไสมาศ', 'แน่นหนา', 4, 12, 2529, 'X', 'LE', 'SCS-123', 'Faison.c...	Error Code: 3819. Check constraint 'chk_advisor_gender' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางอาจารย์ที่ปรึกษา

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_advisorAcademic_Title

CHECK ตำแหน่งวิชาการของอาจารย์ที่ปรึกษา(Academic_Title) กำหนดให้มีค่า 4 ค่า คือ

PF =ศาสตราจารย์ , AP = รองศาสตราจารย์ ,PA = ผู้ช่วยศาสตราจารย์ , LE = อาจารย์

```
1 •   ALTER TABLE advisor
2     ADD CONSTRAINT chk_advisorAcademic_Title
3     CHECK (Academic_Title IN ('PF', 'AP', 'PA', 'LE'));
```

ทดลองใส่ค่าเป็น EA ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร PF,AP,PA,LE เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO advisor VALUES
2     ('TC0712', 'QC0024', 'ไสมาศ', 'แน่นหนา', 4, 12, 2529, 'F', 'EA', 'SCS-123', 'Faison.ch@ku.th', '986278')   ;
```

Output

Action Output

#	Time	Action	Message
1	04:01:54	INSERT INTO advisor VALUES ('TC0712', 'QC0024', 'ไสมาศ', 'แน่นหนา', 4, 12, 2529, 'F', 'EA', 'SCS-123', 'Faison.c...	Error Code: 3819. Check constraint 'chk_advisorAcademic_Title' is violated.

การสร้างตาราง **อาจารย์ผู้สอน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางอาจารย์ผู้สอนขึ้นมา

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Instructor เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ Major_Name เป็น FK และอ้างอิงจากตาราง Major

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Instructor แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Instructor (
2  ID_Instructor CHAR(6) PRIMARY KEY NOT NULL,
3   Major_Name VARCHAR(50) NOT NULL,
4   First_Name VARCHAR(20) NOT NULL,
5   Last_Name VARCHAR(20) NOT NULL,
6   Birth_Day TINYINT NOT NULL,
7   Birth_Month TINYINT NOT NULL,
8   Birth_Year SMALLINT NOT NULL,
9   Gender CHAR(1) NOT NULL,
10  Academic_Title VARCHAR(2) NOT NULL,
11  Email VARCHAR(40) NOT NULL,
12  Phone_Number CHAR(10) NOT NULL,
13  Office_Room VARCHAR(8) NOT NULL,
14  FOREIGN KEY (Major_Name) REFERENCES Major (Major_Name)
15  );
16 INSERT INTO Instructor VALUES
17   ('TC1236', 'สาขาวิชาวิทยาการคอมพิวเตอร์', 'วินัย', 'ลิ', 2, 1, 2529, 'M', 'LE', 'Winai.w@ku.th', '971235789', 'SC9-333'),
18   ('TC2345', 'สาขาวิชาการบัญชี', 'ปลาดุก', 'อนเอ็อน', 10, 2, 2527, 'M', 'PA', 'Padong.c@ku.th', '901233256', 'SIL-102'),
19   ('TC3456', 'สาขาวิชาการตลาด', 'มินิมง', 'สิงห์บา', 19, 3, 2528, 'F', 'LE', 'Minimong.s@ku.th', '980061238', 'SC14-508'),
20   ('TC4567', 'สาขาวิชาคณิตศาสตร์ประยุกต์', 'จันทร์พิญ', 'สมชุติ', 24, 4, 2526, 'F', 'AP', 'Chanpang.s@ku.th', '891254863', 'SIL-305'),
21   ('TC5678', 'สาขาวิชาเทคโนโลยีสารสนเทศ', 'สมศรี', 'ค่าคง', 28, 5, 2530, 'F', 'LE', 'Somsei.k@ku.th', '897755423', 'SC9-402'),
22   ('TC1879', 'สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ', 'จิตลดี', 'บานชื่น', 6, 5, 2539, 'F', 'AP', 'Jitladee.b@ku.th', '982346890', 'HI-201'),
23   ('TC6945', 'สาขาวิศวกรรมโยธา', 'ภพ', 'อดิศชาติ', 7, 11, 2527, 'M', 'PA', 'Phop.a@ku.th', '689743241', 'AE-101');
24   SELECT * FROM Instructor;
```

ID_Instructor	Major_Name	First_Name	Last_Name	Birth_Day	Birth_Month	Birth_Year	Gender	Academic_Title	Email	Phone_Number	Office_Room
TC1236	สาขาวิชาวิทยาการคอมพิวเตอร์	วินัย	ลิ	2	1	2529	M	LE	Winai.w@ku.th	0971235789	SC9-333
TC2345	สาขาวิชาการบัญชี	จิตลดี	บานชื่น	6	5	2539	F	AP	Jitladee.b@ku.th	0982346890	HI-201
TC3456	สาขาวิชาการตลาด	ปกรณ์	ปาน	10	2	2527	M	PA	Padong.c@ku.th	0901233256	SIL-182
TC4567	สาขาวิชาคณิตศาสตร์ประยุกต์	มินิมง	หวัง	19	3	2528	F	LE	Minimang.s@ku.th	0980061238	SC14-508
TC5678	สาขาวิชาเทคโนโลยีสารสนเทศ	ชาญพงศ์	กิตดี	24	4	2526	F	AP	Chanpang.g@ku.th	0819254863	SIL-385
TC1879	สาขาภาษาอังกฤษเพื่ออุตสาหกรรมบริการ	สมศรี	ก้อง	28	5	2530	F	LE	Somsei.k@ku.th	0897755423	SC9-402
TC6945	สาขาวิศวกรรมโยธา	ภพ	อดิศชาติ	7	11	2527	M	PA	Phop.a@ku.th	0689743241	AE-101

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างตารางอาจารย์ผู้สอน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_birthday_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนเกิด (Birth_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่เกิด (Birth_Day) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1. ตรวจสอบว่าเป็นเดือนที่เกิด (Birth _month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่เกิด ไม่เกิน 29

```
1   ALTER TABLE instructor
2   ADD CONSTRAINT chk_birthday_date
3   CHECK (
4       Birth_Month BETWEEN 1 AND 12
5       AND (
6           (Birth_Month IN (1, 3, 5, 7, 8, 10, 12) AND Birth_Day BETWEEN 1 AND 31)
7           OR (Birth_Month IN (4, 6, 9, 11) AND Birth_Day BETWEEN 1 AND 30)
8           OR (Birth_Month = 2 AND Birth_Day BETWEEN 1 AND 29)
9       )
10  );
11
```

ทดสอบการใส่ค่า โดยใส่วันที่ 30 เดือน 2 ซึ่งโดยที่กำหนดไว้ เดือนที่ 2 สามารถเพิ่มค่าได้ไม่เกิน 29 วันทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO instructor VALUES
2     ('TC1296', 'สาขาวิชาวิทยาการคอมพิวเตอร์', 'วิหก', 'ลื่ม', 30, 2, 2529, 'M', 'LE', 'Wihog.w@gku.th', '971235789', 'SCG-333')  ;
```

Output

Action Output

#	Time	Action	Message
1	03:41:20	INSERT INTO instructor VALUES ('TC1296', 'สาขาวิชาวิทยาการคอมพิวเตอร์', 'วิหก', 'ลื่ม', 30, 2, 2529, 'M', 'LE', 'Wiho...	Error Code: 3819. Check constraint 'chk_birthday_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางอาจารย์ผู้สอน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_instructor_gender

CHECK เพศ(Gender) กำหนดค่า 2 ค่า คือ M - Male ผู้ชาย และ F - Female ผู้หญิง

```
1 ●   ALTER TABLE instructor
2     ADD CONSTRAINT chk_instructor_gender
3     CHECK (Gender IN('M','F'));
```

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางอาจารย์ผู้สอน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_instructorAcademic_Title

CHECK ตำแหน่งวิชาการของอาจารย์ผู้สอน(Academic_Title) กำหนดให้มีค่า 4 ค่า คือ

PF = ศาสตราจารย์, AP = รองศาสตราจารย์, PA = ผู้ช่วยศาสตราจารย์, LE = อาจารย์

```
1 ●   ALTER TABLE instructor
2     ADD CONSTRAINT chk_instructorAcademic_Title
3     CHECK (Academic_Title IN ('PF', 'AP', 'PA', 'LE'));
```

ทดลองใส่ค่าเป็น EA ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร PF, AP, PA, LE เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO instructor VALUES
2   ('TC2375', 'สาขาวิชาการบัญชีบริหาร', 'พะอง', 'พงษ์', 10, 2, 2527, 'M', 'EA', 'Padong.g@gku.th', '901123566', 'SIL-182')   ;
```

Output

Action Output

#	Time	Action	Message
1	04:05:11	INSERT INTO instructor VALUES ('TC2375', 'สาขาวิชาการบัญชีบริหาร', 'พะอง', 'พงษ์', 10, 2, 2527, 'M', 'EA', 'Padong.g...	Error Code: 3819. Check constraint 'chk_instructorAcademic_Title' is violated.

การสร้างตาราง **รายวิชาที่เปิดสอน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางรายวิชาที่เปิดสอน

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Course เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Instuctor เป็น FK

และอ้างอิงจากตาราง Instuctor

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Course_offering แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Course_Offering (
    ID_Course CHAR(10) PRIMARY KEY NOT NULL,
    ID_Instructor CHAR(6) NOT NULL,
    Course_Name VARCHAR(100) NOT NULL,
    Credits INT NOT NULL,
    Course_Description TEXT NOT NULL,
    Academic_Year INT NOT NULL,
    Semester VARCHAR(20) NOT NULL,
    Course_Type CHAR(2) NOT NULL,
    Course_Status CHAR(2) NOT NULL,
    Notes TEXT NOT NULL,
    FOREIGN KEY (ID_Instructor) REFERENCES Instructor(ID_Instructor)
);
INSERT INTO Course_Offering (ID_Course, ID_Instructor, Course_Name, Credits, Course_Description, Academic_Year, Semester, Course_Type, Course_Status, Notes) VALUES
    ('1418234', 'TC1236', การพัฒนาซอฟต์แวร์, 3, หลักการพัฒนาซอฟต์แวร์กระบวนการพัฒนาซอฟต์แวร์วงจรชีวิตของซอฟต์แวร์ และเทคนิคการออกแบบซอฟต์แวร์, 2567, ภาคต้น, 'RC', 'WT', 'CP'),
    ('1418288', 'TC2545', การจัดการฐานข้อมูล, 3, แนวคิดพื้นฐานของฐานข้อมูล การออกแบบฐานข้อมูล ระบบจัดการฐานข้อมูล SQL และการประมวลผลให้ฐานข้อมูล, 2567, ภาคต้น, 'EL', 'CC', 'CP'),
    ('1411711', 'TC4567', การพัฒนาเว็บแอปพลิเคชัน, 3, การออกแบบและพัฒนาเว็บแอปพลิเคชันด้วย HTML, CSS, JavaScript รวมถึงการใช้เฟรมเวิร์กต่างๆ, 2567, ภาคต้น, 'RC', 'PC', 'HA'),
    ('1418241', 'TC5678', การวิเคราะห์และออกแบบระบบ, 3, หลักการวิเคราะห์ระบบ เทคนิคการออกแบบระบบ และการพัฒนาแอปพลิเคชันอีกวัตถุ, 2567, ภาคต้น, 'RC', 'OP', 'IP'),
    ('CH2118', 'TC3817', วิศวกรรมข้อมูล, 3, แนวคิดเกี่ยวกับโครงสร้างข้อมูลและอัลกอริทึม ที่ฐานของเอาต์พุต อีกทั้งวิศวกรรมและ, 2567, ภาคต้น, 'RC', 'LL', 'IP'),
    ('1418244', 'TC5678', เทคโนโลยีมือถือ, 3, การพัฒนาแอปพลิเคชันบนแพลตฟอร์มมือถือ เช่น iOS และ Android', 2567, ภาคต้น, 'EL', 'CC', 'CP'),
    ('1418245', 'TC5678', การพัฒนาเว็บขั้นสูง, 3, เทคนิคขั้นสูงในการพัฒนาเว็บ รวมถึง Node.js, React, และฐานข้อมูล NoSQL', 2567, ภาคต้น, 'EL', 'CC', 'CP'),
    ('1418246', 'TC1236', วิศวกรรมซอฟต์แวร์, 3, แนวคิดของวิศวกรรมซอฟต์แวร์ การจัดการวิศวกรรมซอฟต์แวร์ และเทคนิคการพัฒนาแบบไดนามิค, 2567, ภาคต้น, 'RC', 'CC', 'CP'),
    ('1418247', 'TC1236', ปัญญาประดิษฐ์, 3, แนวคิดของปัญญาประดิษฐ์ การค้นหาสู่ของเครื่อง และการพัฒนาโมเดลสำหรับ AI', 2567, ภาคต้น, 'RC', 'CC', 'CP');
SELECT * FROM Course_Offering;
```

ID_Course	ID_Instructor	Course_Name	Credits	Course_Description	Academic_Year	Semester	Course_Type	Course_Status	Notes
01417111	TC1236	การเขียนโปรแกรมเบื้องต้น	3	รายวิชานี้แนะนำแนวคิดพื้นฐานของวิทยาการคอมพิว...	2567	ภาคต้น	RC	WT	CP
01418234	TC2345	คณิตศาสตร์สำหรับวิทยาการคอมพิวเตอร์	3	รายวิชานี้ครอบคลุมหลักการและแนวปฏิบัติของระบ...	2567	ภาคปลาย	EL	CC	CP
01418284	TC3456	แคลคูลัสI	3	รายวิชานี้สอนนักศึกษาถึงวิธีการพัฒนาเว็บแอปพลิเ...	2567	ภาคปลาย	RC	PC	NA
01418288	TC4567	ฟิสิกส์เบื้องต้น	3	รายวิชานี้สำรวจโครงสร้างข้อมูลและอัลกอริทึมทั่วไ...	2567	ภาคต้น	RC	OP	IP
01418244	TC5678	การพัฒนาเว็บแอปพลิเคชัน	3	รายวิชานี้เน้นการพัฒนาเว็บแอปพลิเคชันด้วยเทคโน...	2567	ภาคต้น	RC	CC	CP
01418245	TC6945	การวิเคราะห์และออกแบบระบบ	3	รายวิชานี้ครอบคลุมแนวคิดในการวิเคราะห์และออก...	2567	ภาคต้น	EL	CC	CP
01418246	TC1236	การเขียนโปรแกรมเชิงวัตถุ	3	รายวิชานี้เน้นการเขียนโปรแกรมด้วยแนวคิดเชิงวัตถุ ...	2567	ภาคต้น	RC	CC	CP
01418247	TC1236	ปัญญาประดิษฐ์เบื้องต้น	3	รายวิชานี้แนะนำแนวคิดพื้นฐานของปัญญาประดิษฐ์ เ...	2567	ภาคปลาย	RC	CC	CP
CHEM1021	TC5678	เคมีเบื้องต้น	3	รายวิชานี้เป็นไปตามข้อกำหนดการศึกษาทั่วไป และ...	2567	ภาคต้น	RC	CL	IP

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางรายวิชาที่เปิดสอน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Course_Type

CHECK ประเภทของรายวิชาที่เปิดสอน(Course_Type) กำหนดให้มีค่า 7 ค่า คือ

RC = Required Course (วิชาบังคับ), EL = Elective Course (วิชาเลือก), FE = Free Elective (วิชาเสรี), FC = Foundation Course (วิชาพื้นฐาน), LC = Language Course (วิชาภาษา), LS = Life Skills Course (วิชาทักษะชีวิต), LB = Laboratory/Practical Course (วิชาปฏิบัติการ)

```
1 •  ALTER TABLE course_offering
2    ADD CONSTRAINT chk_Course_Type CHECK (
3        Course_Type IN ('RC', 'EL', 'FE', 'FC', 'LC', 'LS', 'LB')
4    );
```

ทดลองใส่ค่าเป็น EA ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร RC, EL, FE, FC, LC, LS, LB เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO course_offering VALUES
2   ('01418274', 'H20458', 'พีชคณิตเชิงเส้น', 3, 'รายวิชานี้สอนนักศึกษาถึงวิธีการพัฒนาเว็บแอปพลิเคชันแบบไดนามิกโดยใช้HTML, CSS, JavaScript และเทคนิคอื่นๆของซอฟต์แวร์', 2567, 'ภาคปลาย', 'EA', 'PC', ' NA ')
```

```
Output
Action Output
#  Time      Action                                                                                    Message
1  04:13:24  INSERT INTO course_offering VALUES ('01418274', 'H20458', 'พีชคณิตเชิงเส้น', 3, 'รายวิชานี้สอนนักศึกษาถึงวิธีการพั...  Error Code: 3819. Check constraint 'chk_Course_Type' is violated.
```

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางรายวิชาที่เปิดสอน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Course_Status

CHECK สถานะของรายวิชาที่เปิดสอน(Course_Status) กำหนดให้มีค่า 5 ค่า คือ OP = เปิดสอน (Open), CL = ปิดสอน (Closed), CC = ยกเลิก (Cancelled), PC = กำลังดำเนินการ (Processing), WT = รออนุมัติ (Waiting for Approval)

```
1 •  ALTER TABLE course_offering
2    ADD CONSTRAINT chk_Course_Status CHECK (
3        Course_Status IN ('OP', 'CL', 'CC', 'PC', 'WT')
4    );
```

ทดลองใส่ค่าเป็น EA ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร OP, CL, CC, PC, WT เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO course_offering VALUES
2   ('01418274', 'H20458', 'พีชคณิตเชิงเส้น', 3, 'รายวิชานี้สอนนักศึกษาถึงวิธีการพัฒนาเว็บแอปพลิเคชันแบบไดนามิกโดยใช้HTML, CSS, JavaScript และเทคนิคอื่นๆของซอฟต์แวร์', 2567, 'ภาคปลาย', 'PA', 'EA', ' NA ')    ;
```

```
Output
Action Output
#  Time      Action                                                                                    Message
1  04:24:13  INSERT INTO course_offering VALUES ('01418274', 'H20458', 'พีชคณิตเชิงเส้น', 3, 'รายวิชานี้สอนนักศึกษาถึงวิธีการพัฒ...  Error Code: 3819. Check constraint 'chk_Course_Status' is violated.
```

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางรายวิชาที่เปิดสอน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_ Notes

CHECK คำอธิบายสถานะของรายวิชาที่เปิดสอน(Notes) กำหนดให้มีค่า 3 ค่า คือ CP ดำเนินการเสร็จสิ้น (รายวิชาที่เปิดสอนสามารถเปิดสอนได้ตามเงื่อนไข), IP ยังไม่เสร็จสิ้น (รายวิชาที่เปิดสอนยังอยู่ในการดำเนินการเปิดสอน), NA ไม่ผ่านเงื่อนไข (รายวิชาที่เปิดสอนไม่ผ่านเงื่อนไข ไม่สามารถเปิดสอนได้)

```
1 •   ALTER TABLE course_offering
2     ADD CONSTRAINT chk_Notes CHECK (
3         Notes IN ('CP', 'IP', 'NA')
4     );
```

ทดลองใส่ค่าเป็น EA ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร CP, IP, NA เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO course_offering
2   VALUES ('7418474', 'TC5678', 'การวิเคราะห์ข้อมูล', 4, 'รายวิชานี้ครอบคลุมหลักการและแนวปฏิบัติของระบบจัดการฐานข้อมูล นักศึกษาจะได้เรียนรู้ทั้งทางทฤษฎีการออกแบบฐานข้อมูล SQL และการจัดการระบบฐานข้อมูล', 2567, 'ภาคปลาย', 'RC', 'CC', 'EA');
```

Output

Action Output

#	Time	Action	Message
1	04:41:36	INSERT INTO course_offering VALUES ('7418474', 'TC5678', 'การวิเคราะห์ข้อมูล', 4, 'รายวิชานี้ครอบคลุมหลักการและ...	Error Code: 3819. Check constraint 'chk_Notes' is violated.

การสร้างตาราง **การจองห้องเรียน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการจองห้องเรียน

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Booking เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Course เป็น FK และอ้างอิงจากตาราง Course_offering

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Classroom เป็น FK และอ้างอิงจากตาราง Classroom

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Classroom_Booking แสดงผลของตารางที่ได้ผ่านการ Query

```
CREATE TABLE Classroom_Booking (
  ID_Booking CHAR(6) PRIMARY KEY,
  ID_Course CHAR(10) NOT NULL,
  Classroom_ID CHAR(7) NOT NULL,
  Class_Day CHAR(3) NOT NULL,
  Start_Time TIME NOT NULL,
  End_Time TIME NOT NULL,
  Section CHAR(3) NOT NULL,
  Nisit_Count CHAR(3) NOT NULL,
  Booking_Status VARCHAR(2) NOT NULL,
  FOREIGN KEY (ID_Course) REFERENCES Course_Offering(ID_Course)

); INSERT INTO Classroom_Booking VALUES
    ('JH1489', '1418234', 'LH4-301', 'Mon', '09:00:00', '12:00:00', '800', '80', 'SC'),
    ('JH1490', '1418282', 'LH2-303', 'Tue', '14:00:00', '17:00:00', '702', '50', 'UF'),
    ('JH1491', '1418299', 'SC9-303', 'Fri', '08:30:00', '10:30:00', '703', '100', 'SC'),
    ('JH1492', 'CHEM101', 'SC4-502', 'Wed', '13:00:00', '16:00:00', '700', '70', 'UF'),
    ('JH1493', '1418284', 'LH2-204', 'Sat', '17:00:00', '20:00:00', '701', '98', 'SC'),
    ('JH1494', '1418244', 'LH4-201', 'Mon', '13:00:00', '04:00:00', '700', '50', 'SC'),
    ('JH1495', '1418245', 'LH2-202', 'Tue', '14:00:00', '03:00:00', '700', '60', 'SC'),
    ('JH1496', '1418246', 'LH3-307', 'Fri', '09:00:00', '13:00:00', '800', '70', 'SC'),
    ('JH1497', '1418247', 'SC4-302', 'Sun', '08:00:00', '09:30:00', '802', '70', 'SC');
SELECT * FROM Classroom_Booking;
```

ID_Booking	ID_Course	ID_Classroom	Class_Day	Start_Time	End_Time	Section	Nisit_Count	Booking_Status
JH1489	01418234	LH4-301	Mon	09:00:00	12:00:00	800	50	SC
JH1490	01418282	LH2-303	Tue	14:00:00	17:00:00	702	40	UF
JH1491	01418299	SC9-303	Fri	08:30:00	10:00:00	703	30	SC
JH1492	CHEM101	SC4-502	Wed	13:00:00	16:00:00	700	30	UF
JH1493	01418284	LH2-204	Sat	17:00:00	20:00:00	701	70	SC
JH1494	01418244	LH4-201	Mon	13:00:00	16:00:00	700	50	SC
JH1495	01418245	LH2-202	Tue	14:00:00	15:00:00	700	60	SC
JH1496	01418246	LH3-307	Fri	09:00:00	13:00:00	800	70	SC
JH1497	01418247	SC4-302	Sun	08:00:00	09:30:00	802	70	SC

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจองห้องเรียน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_classroom_booking _date

CHECK ตรวจสอบวันที่เรียนของรายวิชา(Class_Day) โดยกำหนดให้มี 7 ตัวเลือกดังนี้ Sun =วันอาทิตย์, Mon = วันจันทร์, Tue = วันอังคาร, Wed = วันพุธ, Thu = วันพฤหัสบดี, Fri =วันศุกร์, Sat=วันเสาร์

```
1 •  ALTER TABLE classroom_booking
2    ADD CONSTRAINT chk_classroom_booking_date CHECK (
3        Class_Day IN ('Sun', 'Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat')
4    );
```

ทดลองใส่ค่าเป็น ABC ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร Mon, Tue, Wed, Thu, Fri, Sat, Sun เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO classroom_booking
2   VALUES ('JH1442', '01418247', 'SC4-501', 'ABC', '13:00:00', '16:00:00', '700', '30', 'UF')  ;
```

Output

Action Output

#	Time	Action	Message
1	04:51:55	INSERT INTO classroom_booking VALUES ('JH1442', '01418247', 'SC4-501', 'ABC', '13:00:00', '16:00:00', '700', '30', '...	Error Code: 3819. Check constraint 'chk_classroom_booking_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการจองห้องเรียน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Booking_Status

CHECK ตรวจสอบสถานะการจองห้อง (Booking_Status) โดยกำหนดให้มี 2 ค่า คือ SC = Successful (จองห้องเรียนสำเร็จ), UF = Unsuccessful/Failed (จองไม่สำเร็จกรณีที่ห้องเรียนไม่ว่างหรือติดปัญหาทางเทคนิคอื่นๆ)

```
1 •  ALTER TABLE classroom_booking
2    ADD CONSTRAINT chk_Booking_Status CHECK (
3        Booking_Status IN ('SC', 'UF')
4    );
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร SC, UF เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```
1   INSERT INTO classroom_booking
2   VALUES ('JH1442', '01418247', 'SC4-501', 'Mon', '13:00:00', '16:00:00', '700', '30', 'XX')  ;
```

Output

Action Output

#	Time	Action	Message
1	04:56:25	INSERT INTO classroom_booking VALUES ('JH1442', '01418247', 'SC4-501', 'Mon', '13:00:00', '16:00:00', '700', '30', '...	Error Code: 3819. Check constraint 'chk_Booking_Status' is violated.

การสร้างตาราง **ห้องเรียน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางห้องเรียนขึ้นมา

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ Classroom_ID เป็น PK

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM ClassRoom แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Classroom (
2      Classroom_ID CHAR(7) PRIMARY KEY NOT NULL,
3      Building VARCHAR(10) NOT NULL,
4      Floor CHAR(1) NOT NULL,
5      Room_Number CHAR(3) NOT NULL,
6      Seat_Capacity CHAR(3) NOT NULL,
7      Classroom_Type VARCHAR(2) NOT NULL
8  );
9  INSERT INTO Classroom VALUES
10 ('LH4-301', 'LH4', '3', '301', '50', 'LR'),
11 ('LH2-303', 'LH2', '3', '303', '40', 'LR'),
12 ('SC9-303', 'SC9', '3', '303', '30', 'LB'),
13 ('SC4-502', 'SC4', '5', '502', '30', 'LR'),
14 ('LH2-204', 'LH2', '2', '204', '70', 'LR'),
15 ('LH4-201', 'LH4', '2', '201', '60', 'LB'),
16 ('LH2-202', 'LH2', '2', '202', '60', 'LR'),
17 ('LH3-307', 'LH3', '3', '307', '100', 'LR'),
18 ('SC4-302', 'SC4', '3', '302', '100', 'LB');
19 SELECT * FROM Classroom;
```

ID_Classroom	Building	Floor	Room_Number	Seat_Capacity	Classroom_Type
LH2-202	LH2	2	202	60	LR
LH2-204	LH2	2	204	70	LR
LH2-303	LH2	3	303	40	LR
LH3-307	LH3	3	307	100	LR
LH4-201	LH4	2	201	60	LB
LH4-301	LH4	3	301	50	LR
SC4-302	SC4	3	302	100	LB
SC4-502	SC4	5	502	30	LR
SC9-303	SC9	3	303	30	LB

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางห้องเรียน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_classroom_type

CHECK ตรวจสอบประเภทห้องเรียน(Classroom_Type) โดยกำหนดให้มี 2 ค่า คือ

LR = Lecture Room (ห้องเรียนบรรยาย),LB = Laboratory/Practical Room (ห้องเรียนปฏิบัติการ)

```sql
ALTER TABLE classroom
ADD CONSTRAINT chk_classroom_type CHECK (
    Classroom_Type IN ('LR', 'LB')
);
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มค่าแค่ตัวอักษร SC , UF เท่านั้น จึงทำให้ไม่สามารถเพิ่มค่าได้

```sql
INSERT INTO classroom
VALUES    ('LH2-373', 'LH2', '3', '303', '40', 'XX')  ;
```

Output

Action Output

#	Time	Action	Message
1	05:02:32	INSERT INTO classroom VALUES ('LH2-373', 'LH2', '3', '303', '40', 'XX')	Error Code: 3819. Check constraint 'chk_classroom_type' is violated.

การสร้างตาราง **การชำระเงินค่าเทอม** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางการชำระเงินค่าเทอมขึ้นมา

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Payment เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Nisit เป็น FK

และอ้างอิงจากตาราง Nisit

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Tuition_Payment แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Tuition_Payment (
2  ID_Payment CHAR(6) PRIMARY KEY NOT NULL,
3  ID_Nisit CHAR(10) NOT NULL,
4  Amount INT NOT NULL,
5  Payment_Day TINYINT NOT NULL,
6  Payment_Month TINYINT NOT NULL,
7  Payment_Year SMALLINT NOT NULL,
8  Payment_Time TIME NOT NULL,
9  Discount_5A INT,
10 Net_Amount INT NOT NULL,
11 Payment_Method VARCHAR(2) ,
12 Deferment_Status VARCHAR(2),
13 Fund_Received VARCHAR(2),
14 Payment_Status VARCHAR(2) NOT NULL,
15 FOREIGN KEY (ID_Nisit) REFERENCES Nisit (ID_Nisit)
16 );
17 INSERT INTO Tuition_Payment VALUES
18 ('B65234', '6621600873', 15400.00, 17, 10, 2567, '09:23:00', 5000.00, 10400.00, 'CA', NULL, NULL, 'SC'),
19 ('B66982', '6721600896', 26700.00, 20, 11, 2567, '07:55:00', NULL, 26700.00, 'MB', NULL, NULL, 'SC'),
20 ('B67894', '6721604593', 36000.00, 23, 10, 2567, '11:25:00', NULL, 36000.00, 'NL', NULL, 'SL', 'SC'),
21 ('B78965', '6712594569', 21000.00, 25, 11, 2567, '12:43:00', NULL, 21000.00, 'NL', NULL, 'IC', 'SC'),
22 ('B65214', '6621600221', 16900.00, 12, 10, 2567, '15:20:00', NULL, 16900.00, NULL, 'ได้รับ', NULL, 'DP'),
23 ('B65215', '6760078701', 32400.00, 13, 11, 2567, '11:20:00', NULL, 32400.00, 'CA', NULL, NULL, 'SC'),
24 ('B65216', '6721790065', 40000.00, 17, 10, 2567, '10:25:00', NULL, 40000.00, 'MB', NULL, NULL, 'SC'),
25 ('B65217', '6521730068', 17300.00, 17, 10, 2567, '11:12:00', NULL, 17300.00, 'MB', NULL, NULL, 'SC'),
26 ('B65218', '6621900345', 18300.00, 16, 11, 2567, '11:30:00', NULL, 18300.00, NULL, 'ได้รับ', NULL, 'PD'),
27 ('B75941', '6421876003', 40000.00, 12, 10, 2567, '10:36:00', NULL, 40000.00, 'MB', NULL, NULL, 'SC');
28 SELECT * FROM Tuition_Payment;
```

ID_Payment	ID_Nisit	Term_Fee_Amount	Payment_Day	Payment_Month	Payment_Year	Payment_Time	Discount_5A	Net_Amount	Payment_Method	Deferment_Status	Fund_Received	Payment_Status
B65234	6621600873	15400	17	10	2567	09:23:00	5000	10400	CA		NULL	SC
B66982	6721600896	26700	20	11	2567	07:55:00	NULL	26700	MB	NULL	NULL	SC
B67894	6721604593	36000	23	10	2567	11:25:00	NULL	36000	NL	NULL	NULL	SC
B78965	6712594569	21000	25	11	2567	12:43:00	NULL	21000	NL	NULL	NULL	SC
B65214	6621600221	16900	12	10	2567	15:20:00		16900		ได้รับ	NULL	DP
B65215	6760078701	32400	13	11	2567	11:20:00	NULL	32400	CA	NULL	NULL	SC
B65216	6721790065	40000	17	10	2567	10:25:00	NULL	40000	MB	NULL	SL	SC
B65217	6521730068	17300	17	10	2567	11:12:00	NULL	17300	MB	NULL	IC	SC
B65218	6621900345	18300	16	11	2567	11:30:00'	NULL	18300	NULL	ได้รับ	NULL	PD
B75941	6421876003	40000	12	10	2567	10:36:00	NULL	40000	MB	NULL	NULL	SC

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการชำระเงินค่าเทอม

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_payment_day

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าปีที่ทำการชำระเงิน (Payment_Year) อยู่ในช่วง 2567
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ทำการชำระเงิน (Payment_Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ทำการชำระเงิน (Payment_Day) อยู่ในช่วง 1 ถึง 31 หรือไม่

  หากวันที่ทำการชำระเงิน ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

  1.ตรวจสอบว่าเป็นเดือนที่ชำระเงิน (Payment_Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

  2.ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

  3.ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ทำการชำระเงินไม่เกิน 29

```
ALTER TABLE tuition_payment
ADD CONSTRAINT chk_payment_day
CHECK ( Payment_Year = 2567 AND Payment_Month
BETWEEN 1 AND 12 AND ( (Payment_Month IN (1, 3, 5, 7, 8, 10, 12)
AND Payment_Day BETWEEN 1 AND 31) OR (Payment_Month IN (4, 6, 9, 11)
AND Payment_Day BETWEEN 1 AND 30) OR (Payment_Month = 2 AND Payment_Day BETWEEN 1 AND 29) ) );
```

ทดสอบการใส่ค่า โดยใส่วันที่ 30 เดือน 2 ซึ่งโดยที่กำหนดไว้ เดือนที่ 2 สามารถเพิ่มค่าได้ไม่เกิน 29 วัน ทำให้ไม่สามารถเพิ่มค่าได้

```
INSERT INTO tuition_payment
VALUES     ('B78965', '6712594569', 34000, 30, 2, 2567, '12:43:00', NULL, 34000, ' NL ', Null, ' SC ')  ;
```

Output

Action Output

#	Time	Action	Message
1	05:14:12	INSERT INTO tuition_payment VALUES ('B78965', '6712594569', 34000, 30, 2, 2567, '12:43:00', NULL, 34000, ' N...	Error Code: 3819. Check constraint 'chk_payment_day' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_discount5A

CHECK ตรวจสอบสถานะส่วนลด5A(discount_5A) โดยกำหนดให้มี 2 ค่า คือ 5000 = ได้รับทุน5A และ NULL = ไม่ได้รับทุน5A

```
1 •  ALTER TABLE tuition_payment
2    ADD CONSTRAINT chk_discount5A CHECK (discount5A IN(5000,NULL)) ;
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า 5000 , NULL เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO tuition_payment
2    VALUES ('B6692', '6721600896', 28000, 20, 11, 2567, '07:55:00', '9000', 28000, 'MB', NULL, NULL, ' SC ') ;
```

Output

Action Output

#	Time	Action	Message
1	18:57:16	INSERT INTO tuition_payment VALUES ('B6692', '6721600896', 28000, 20, 11, 2567, '07:55:00', '9000', 28000, 'MB',...	Error Code: 3819. Check constraint 'chk_discount5A' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_payment_method

CHECK ตรวจสอบช่องทางการชำระเงิน(payment_method) โดยกำหนดให้มี 6 ค่า คือ ช่องทางการชำระเงินค่าเทอม CA = Cash เงินสด, MB = Mobile Banking โมบายแบงค์กิ้ง, CC = Credit Card บัตรเครดิต, CH = (Cashier's Check) แคชเชียร์เช็ค, BK = (Bank Transfer/Payment at Bank) ธนาคาร, NL = Nist Loan (เงินกู้)

```
1 •  ALTER TABLE tuition_payment
2    ADD CONSTRAINT chk_payment_method CHECK (payment_method IN ('CA', 'MB', 'CC', 'CH', 'BK', 'NL'));
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า CA, MB, CC, CH, BK, NL เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1
2 •  INSERT INTO Tuition_Payment VALUES
3    ('B65534', '6621600873', 15400, 17, 10, 2567, '09:23:00', NULL , 10400, 'XX', NULL, NULL, 'SC');
```

Output

Action Output

#	Time	Action	Message
1	13:11:30	INSERT INTO Tuition_Payment VALUES ('B65534', '6621600873', 15400, 17, 10, 2567, '09:23:00', NULL , 10400, 'X...	Error Code: 3819. Check constraint 'chk_payment_method' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_ Deferred_Status

CHECK ตรวจสอบสถานะการขอผ่อนผัน(Deferred_Status) โดยกำหนดให้มี 2 ค่า คือ ได้รับ = ได้รับอนุมัติการผ่อนผัน และ NULL คือไม่ได้รับอนุมัติการผ่อนผัน

```
1 •  ALTER TABLE tuition_payment
2    ADD CONSTRAINT chk_Deferred_Status CHECK (Deferred_Status IN('ได้รับ',NULL)) ;
```

ทดลองใส่ค่าเป็น รับ ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า ได้รับ, NULL เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO tuition_payment
2    VALUES ('B6692', '6721600896', 28000, 20, 11, 2567, '07:55:00', '5000', 28000, 'MB', รับ, NULL, ' SC ') ;
```

Output

Action Output

#	Time	Action	Message
1	19:01:32	INSERT INTO tuition_payment VALUES ('B6692', '6721600896', 28000, 20, 11, 2567, '07:55:00', '5000', 28000, 'MB',...	Error Code: 3819. Check constraint 'chk_Deferment_Status' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_ Fund_Received

CHECK ตรวจสอบกองทุนที่นิสิตได้รับ(Fund_Received) โดยกำหนดให้มี 3 ค่า คือ SL = ได้รับกองทุนกยศ. และ IC = ได้รับกองทุนกรอ. และค่า NULL หากไม่ได้รับ

```
1 •  ALTER TABLE tuition_payment
2    ADD CONSTRAINT chk_Fund_Received CHECK (Fund_Received IN ('กยศ.', 'กรอ.', NULL));
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า SL , IC ,NULL เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO tuition_payment
2    VALUES ('B68012', '6721600896', 28000, 20, 11, 2567, '07:55:00', '5000', 28000, 'MB', NULL, 'XX' , 'SC') ;
```

Output

Action Output

#	Time	Action	Message
1	22:31:24	INSERT INTO tuition_payment VALUES ('B68012', '6721600896', 28000, 20, 11, 2567, '07:55:00', '5000', 28000, 'MB...	Error Code: 3819. Check constraint 'chk_FundReceived' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการการชำระเงิน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_ Payment_Status

CHECK ตรวจสอบเสถานะการชำระเงิน(Payment_Status) โดยกำหนดให้มี 3 ค่า คือ SC = Successful (ชำระเงินสำเร็จ)UF = Unsuccessful/Failed (ยังไม่ชำระเงิน/ชำระเงินไม่สำเร็จ) , PD = Pending (รอดำเนินการ / อยู่ระหว่างตรวจสอบ) , RP = Rejected/Refunded (ถูกปฏิเสธ / มีการคืนเงิน) , DP = Deferred Payment (ผ่อนผัน)

```
1 •  ALTER TABLE tuition_payment
2    ADD CONSTRAINT chk_Payment_Status
3    CHECK (Payment_Status IN ('SC', 'UF', 'PD', 'RP', 'DP'));
4
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า SC,UF,PD,RP,DP เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO tuition_payment
2    VALUES ('B680c2', '6721600896', 28000, 20, 11, 2567, '07:55:00', '5000', 28000, 'MB', NULL, NULL , ' XX ') ;
3
```

Output

Action Output

#	Time	Action	Message
1	22:22:12	INSERT INTO tuition_payment VALUES ('B680c2', '6721600896', 28000, 20, 11, 2567, '07:55:00', '5000', 28000, 'MB...	Error Code: 3819. Check constraint 'chk_Payment_Status' is violated.

การสร้างตาราง **การลงทะเบียนเรียน** โดยมีคำสั่งดังนี้

CREATE TABLE สร้างตารางลงทะเบียนเรียนขึ้นมา

PRIMARY KEY เพิ่ม PK โดยตารางนี้จะให้แอตทริบิวต์ที่ชื่อ ID_Registration เป็น PK

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Course เป็น FK

และอ้างอิงจากตาราง Course_Offering

FOREIGN KEY เพิ่มตัว FK โดยในตารางนี้มีแอตทริบิวต์ชื่อ ID_Payment เป็น FK

และอ้างอิงจากตาราง Tuition_Payment

INSERT INTO เพิ่มข้อมูลเข้าไปในตารางด้วยคำสั่งนี้

SELECT * FROM Course_Registration แสดงผลของตารางที่ได้ผ่านการ Query

```
1  CREATE TABLE Course_Registration (
2  ID_Registration CHAR(6) PRIMARY KEY NOT NULL,
3  ID_Course CHAR(10) NOT NULL,
4  ID_Payment CHAR(6) NOT NULL,
5  Registration_Day TINYINT NOT NULL,
6  Registration_Month TINYINT NOT NULL,
7  Registration_Year SMALLINT NOT NULL,
8  Registration_Time TIME NOT NULL,
9  Registration_Status VARCHAR(4) NOT NULL,
10 Grade CHAR(2),
11 FOREIGN KEY (ID_Course) REFERENCES Course_offering (ID_Course),
12 FOREIGN KEY (ID_Payment) REFERENCES Tuition_payment (ID_Payment)
13 );
14 INSERT INTO Course_Registration VALUES
15 ('J36521', '1418234', 'B65234', 10, 11, 2567, '09:02:00', 'Add', 'N'),
16 ('J37875', '1418288', 'B66982', 12, 11, 2567, '09:12:00', 'Add', 'N'),
17 ('J62331', '1264889', 'B67894', 14, 11, 2567, '09:20:00', 'Drop', 'W'),
18 ('J78994', '1481339', 'B78965', 13, 11, 2567, '09:05:00', 'Add', 'N'),
19 ('J88462', '1418284', 'B65214', 13, 11, 2567, '09:03:00', 'Add', 'N'),
20 ('J88463', '1418244', 'B65215', 9, 11, 2567, '11:02:00', 'Add', 'N'),
21 ('J88464', '1418245', 'B65216', 8, 11, 2567, '11:13:00', 'Add', 'N'),
22 ('J88465', '1418246', 'B65217', 9, 11, 2567, '11:24:00', 'Add', 'N'),
23 ('J88466', '1418244', 'B65214', 9, 11, 2567, '13:22:00', 'Add', 'N'),
24 ('J88467', '1418245', 'B65214', 9, 11, 2567, '13:30:00', 'Add', 'N');
25 SELECT * FROM Course_Registration;
```

ID_Registration	ID_Course	ID_Payment	Registration_Day	Registration_Month	Registration_Year	Registration_Time	Registration_Status	Grade
J36521	01418234	B65234	10	11	2567	09:02:00	Add	N
J37875	01418288	B66982	12	11	2567	09:12:00	Add	N
J62331	01264889	B67894	14	11	2567	09:03:00	Add	N
J78994	01481339	B78965	13	11	2567	09:05:00	Add	N
J88462	01418284	B65214	12	11	2567	09:20:00	Drop	W
J88463	01418244	B65215	9	11	2567	11:02:00	Add	N
J88464	01418245	B65216	8	11	2567	11:13:00	Add	N
J88465	01418246	B65217	9	11	2567	11:24:00	Add	N

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างตารางการลงทะเบียนเรียน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_Registration_date

CHECK

- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าเดือนที่ลงทะเบียนเรียน(Registration _Month) อยู่ในช่วง 1 ถึง 12
- กำหนดเงื่อนไขที่ข้อมูลในตาราง ตรวจสอบว่าวันที่ลงทะเบียนเรียน (Registration _Day) อยู่ในช่วง 1 ถึง 31 หรือไม่

หากวันที่เกิด ผ่านเงื่อนไขแรกแล้ว แบ่ง 3 แบบ

1.ตรวจสอบว่าเป็นเดือนที่ลงทะเบียนเรียน (Registration _Month) เป็นเดือนที่มี 31 วันหรือไม่ (มกราคม มีนาคม พฤษภาคม กรกฎาคม สิงหาคม ตุลาคม ธันวาคม)

2. ตรวจสอบว่าเป็นเดือนที่มี 30 วันหรือไม่ (เมษายน มิถุนายน กันยายน พฤศจิกายน)

3. ตรวจสอบว่าเป็นเดือน 2 หรือไม่ หากเป็นเดือน 2 จะตรวจสอบว่าวันที่ลงทะเบียนเรียนไม่เกิน 29

```
1  ALTER TABLE  course_registration
2  ADD CONSTRAINT chk_Registration_date
3  CHECK (
4      Registration_Month BETWEEN 1 AND 12
5      AND (
6          (Registration_Month IN (1, 3, 5, 7, 8, 10, 12) AND Registration_Day BETWEEN 1 AND 31)
7          OR (Registration_Month IN (4, 6, 9, 11) AND Registration_Day BETWEEN 1 AND 30)
8          OR (Registration_Month = 2 AND Registration_Day BETWEEN 1 AND 29)
9      )
10 );
```

ทดสอบการใส่ค่า โดยใส่วันที่ 31 เดือน 2 ซึ่งโดยที่กำหนดไว้ เดือนที่ 2 สามารถเพิ่มค่าได้ไม่เกิน 29 วันทำให้ไม่สามารถเพิ่มค่าได้

```
1  INSERT INTO course_registration VALUES
2      ('J62381', '01264879', 'B67824', 31, 2, 2567, '09:20:00', 'Add', 'W')  ;
```

Output

Action Output

#	Time	Action	Message
1	03:24:45	INSERT INTO course_registration VALUES ('J62381', '01264879', 'B67824', 31, 2, 2567, '09:20:00', 'Add', 'W')	Error Code: 3819. Check constraint 'chk_Registration_date' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการลงทะเบียนเรียน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_registration_status

CHECKตรวจสอบสถานะการลงทะเบียน(registration_status) โดยกำหนดให้มี 2 ค่า คือ Add เพิ่ม(เพิ่มการลงทะเบียนรายวิชาที่เปิดสอน) Drop ถอน(ถอนการลงทะเบียนรายวิชาที่เปิดสอน)

```
1    ALTER TABLE course_registration
2    ADD CONSTRAINT chk_registration_status
3    CHECK (registration_status IN ('Add', 'Drop'));
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า Add, Drop เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO course_registration VALUES
2    ('J78994', '01481339', 'B78965', 13, 11, 2567, '09:05:00', 'XX', 'N')  ;
```

Output

Action Output

#	Time	Action	Message
1	15:04:05	INSERT INTO course_registration VALUES (J78994', '01481339', 'B78965', 13, 11, 2567, '09:05:00', 'XX', 'N')	Error Code: 3819. Check constraint 'chk_registration_status' is violated.

ALTER TABLE คำสั่งนี้ใช้แก้ไขโครงสร้างของตารางการลงทะเบียนเรียน

ADD CONSTRAINT คำสั่งนี้เพิ่มข้อจำกัดใหม่ชื่อ chk_grade_status

CHECK ตรวจสอบผลการเรียนของรายวิชา(grade) โดยกำหนดให้มี 11 ค่า คือ A = 4.00, B+ = 3.50, B = 3.00, C+ = 2.50, C = 2.00, D+ = 1.50, D = 1.00, F = 0.00, W = Withdrawal (ถอนรายวิชา), N = Grade not reported (ยังไม่ทราบระดับคะแนน), I = Incomplete (ยังไม่สมบูรณ์)

```
1 •  ALTER TABLE course_registration
2    ADD CONSTRAINT chk_grade_status
3    CHECK (grade IN ('A', 'B+', 'B', 'C+', 'C', 'D+', 'D', 'F', 'W', 'N', 'I'));
```

ทดลองใส่ค่าเป็น XX ซึ่งตามที่กำหนดไว้คือต้องเพิ่มได้แค่ค่า A, B+, B, C+, C, D+, D, F, W, N, I เท่านั้นจึงทำให้ไม่สามารถเพิ่มค่าได้

```
1 •  INSERT INTO course_registration VALUES
2    ('J78994', '01481339', 'B78965', 13, 11, 2567, '09:05:00', 'Add', 'X')  ;
```

Output

Action Output

#	Time	Action	Message
1	15:08:24	INSERT INTO course_registration VALUES (J78994', '01481339', 'B78965', 13, 11, 2567, '09:05:00', 'Add', 'X')	Error Code: 3819. Check constraint 'chk_grade_status' is violated.

ตัวอย่างโจทย์ SQL

1. จงแสดงข้อมูลรหัสนิสิต ชั้นปี ชื่อสาขาวิชา ประเภทภาคและค่าเทอมสุทธิ สถานะการชำระเงิน โดยแสดงเฉพาะนิสิตที่ศึกษาอยู่ชั้นปีที่ 3 เป็นนิสิตที่มีสถานะกำลังศึกษาอยู่เท่านั้น

คำสั่ง SQL

```
SELECT N.ID_Nisit, N.Year_Level, M.Major_Name, N.Program_Type, T.Net_Amount AS
Regular_Tuition_Fee, T.Payment_Status
FROM Nisit N, Major M, Tuition_Payment T
WHERE N.Major_Name = M.Major_Name
AND N.ID_Nisit = T.ID_Nisit
AND N.Year_Level = 3
AND N.Nisit_Status = 'ST'
ORDER BY N.ID_Nisit;
```

```
1 •  SELECT N.ID_Nisit, N.Year_Level, M.Major_Name, N.Program_Type, T.Net_Amount AS
2    Regular_Tuition_Fee, T.Payment_Status
3    FROM Nisit N, Major M, Tuition_Payment T
4    WHERE N.Major_Name = M.Major_Name
5    AND N.ID_Nisit = T.ID_Nisit
6    AND N.Year_Level = 3
7    AND N.Nisit_Status = 'ST'
8    ORDER BY N.ID_Nisit;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ID_Nisit	Year_Level	Major_Name	Program_Type	Regular_Tuition_Fee	Payment_Status
6521730068	3	สาขาวิศวกรรมโยธา	RP	17300	SC

คำอธิบาย

ทำการค้นหาข้อมูล **รหัสนิสิต**(ID_Nisit) **ชั้นปี**(Year_Level) **ชื่อสาขาวิชา**(Major_Name) **ประเภทภาค**(Program_Type) **ค่าเทอมสุทธิ**(Regular_Tuition_Fee) **สถานะการชำระเงิน** (Payment_Status)

จากตารางนิสิต (Nisit) ตารางสาขาวิชา (Major) ตารางการชำระค่าเทอม (Tuition_Payment)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**นิสิต** กับ **สาขาวิชา** ผ่านแอตทริบิวต์ Major_Name
- ตาราง**นิสิต** กับ **การชำระค่าเทอม** ผ่านแอตทริบิวต์ ID_Nisit

โดยเลือกเฉพาะนิสิตที่กำลังศึกษาอยู่ชั้นปีที่ 3 และมีสถานะกำลังศึกษาอยู่เท่านั้น

เพื่อให้ผู้ดูแลระบบสามารถตรวจสอบข้อมูลของนิสิตในสถานะต่างๆ ได้อย่างครบถ้วน เช่น การติดตามสถานะการชำระเงิน ค่าเทอม และ สาขาวิชา

2.ต้องการทราบรหัสนิสิตและหมายเลขการชำระเงิน ที่ถูกทำรายการจ่ายค่าเทอมใน วันที่ 17 ตุลาคม พ.ศ. 2567 ในช่วงเวลา 09:00 น. – 12:00 น. แสดงรายละเอียดการชำระเงินของแต่ละรายการ และแสดงสาขาวิชาของนิสิตด้วย

คำสั่ง SQL

```
SELECT N.ID_Nisit, TP.ID_Payment, M.Major_Name, TP.Payment_Day,
TP.Payment_Month,TP.Payment_Year,TP.Payment_Time,TP.Payment_Method,
TP.Payment_Status
FROM Tuition_Payment TP, Nisit N, Major M
WHERE TP.ID_Nisit = N.ID_Nisit
AND N.Major_Name = M.Major_Name
AND TP.Payment_Day = 17
AND TP.Payment_Month = 10
AND TP.Payment_Year = 2567
AND TP.Payment_Time BETWEEN '09:00:00' AND '12:00:00';
```

```
1 • SELECT N.ID_Nisit, TP.ID_Payment, M.Major_Name, TP.Payment_Day, TP.Payment_Month,TP.Payment_Year,TP.Payment_Time,TP.Payment_Method,
2   TP.Payment_Status
3   FROM Tuition_Payment TP, Nisit N, Major M
4   WHERE TP.ID_Nisit = N.ID_Nisit
5   AND N.Major_Name = M.Major_Name
6   AND TP.Payment_Day = 17
7   AND TP.Payment_Month = 10
8   AND TP.Payment_Year = 2567
9   AND TP.Payment_Time BETWEEN '09:00:00' AND '12:00:00';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ID_Nisit	ID_Payment	Major_Name	Payment_Day	Payment_Month	Payment_Year	Payment_Time	Payment_Method	Payment_Status
6721790065	B65216	สาขาวิชาวิทยาการคอมพิวเตอร์	17	10	2567	10:25:00	MB	SC
6521730068	B65217	สาขาวิศวกรรมโยธา	17	10	2567	11:12:00	MB	SC
6621600873	B65234	สาขาวิชาการบัญชีบริหาร	17	10	2567	09:23:00	CA	SC

คำอธิบาย

ทำการค้นหา **รหัสนิสิต**(ID_Nisit) **หมายเลขการชำระเงิน**(ID_Payment) **ชื่อสาขาวิชา** (Major_Name) **วันชำระเงิน**(Payment_Day) **เดือนชำระเงิน**(Payment_Month) **ปีชำระเงิน** (Payment_Year) **เวลาชำระเงิน**(Payment_Time**) ช่องทางการชำระเงิน(**Payment_Method) **สถานะการชำระเงิน**(Payment_Status)

จากตารางนิสิต (Nisit) การชำระเงินค่าเทอม (Tuition_Payment) สาขาวิชา (Major)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**การชำระเงินค่าเทอม** กับ **นิสิต** ด้วยแอตทริบิวต์ ID_Nisit
- ตาราง**นิสิต** กับ **สาขาวิชา** ด้วยแอตทริบิวต์ Major_Name

โดยเลือกเฉพาะรายการที่มี วันที่ชำระเงินเป็น 17 ตุลาคม พ.ศ. 2567 และ ช่วงเวลาการชำระเงิน อยู่ระหว่าง 09:00 น. – 12:00 น.

เพื่อเป็นข้อมูลกรณีที่ระบบเกิดปัญหาขัดข้องในวันและช่วงเวลาดังกล่าว เพราะจะช่วยในการตรวจสอบการชำระเงินจริง และใช้ในการตรวจสอบข้อผิดพลาดเกี่ยวกับการชำระเงิน เพื่อให้สามารถจัดการข้อมูลได้ถูกต้องในภายหลัง

3. ในวันจันทร์และวันศุกร์ตั้งแต่เวลา 08:00 - 17:00 น. มีห้องเรียนใดบ้างที่ถูกจอง พร้อมชื่อ-นามสกุลอาจารย์ผู้สอน ชื่อวิชาที่สอน หมู่เรียน และจำนวนที่นั่งในห้องนั้นๆ โดยเรียงลำดับตามวันและเวลา

คำสั่ง SQL

```
SELECT cb.ID_Classroom, cb.Class_Day, cb.Start_Time, cb.End_Time, i.First_Name,
i.Last_Name, co.Course_Name, cb.Section, r.Seat_Capacity
FROM Classroom_Booking cb, Classroom r, Course_Offering co, Instructor i
WHERE cb.ID_Classroom = r.ID_Classroom
AND cb.ID_Course = co.ID_Course AND co.ID_Instructor = i.ID_Instructor
AND cb.Class_Day IN ('Mon', 'Fri') AND cb.Start_Time >= '08:00:00'
AND cb.End_Time <= '17:00:00'
ORDER BY cb.Class_Day, cb.Start_Time;
```

```
1  SELECT cb.ID_Classroom, cb.Class_Day, cb.Start_Time, cb.End_Time,
2  i.First_Name, i.Last_Name, co.Course_Name, cb.Section, r.Seat_Capacity
3  FROM Classroom_Booking cb, Classroom r, Course_Offering co, Instructor i
4  WHERE cb.ID_Classroom = r.ID_Classroom
5      AND cb.ID_Course = co.ID_Course
6      AND co.ID_Instructor = i.ID_Instructor
7      AND cb.Class_Day IN ('Mon', 'Fri')
8      AND cb.Start_Time >= '08:00:00'
9      AND cb.End_Time <= '17:00:00'
10 ORDER BY cb.Class_Day, cb.Start_Time;
11
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ID_Classroom	Class_Day	Start_Time	End_Time	First_Name	Last_Name	Course_Name	Section	Seat_Capacity
LH3-307	Fri	09:00:00	13:00:00	วินัย	ลี	การเขียนโปรแกรมเชิงวัตถุ	800	100
LH4-301	Mon	09:00:00	12:00:00	ปกรณ์	ปาน	คณิตศาสตร์สำหรับวิทยาการคอมพิวเตอร์	800	50
LH4-201	Mon	13:00:00	16:00:00	สมศรี	ก้อง	การพัฒนาเว็บแอปพลิเคชัน	700	60

คำอธิบาย

ทำการค้นหา **รหัสห้องเรียน**(ID_Classroom) **วันเรียน**(Class_Day) **เวลาเริ่ม**(Start_Time) **เวลาสิ้นสุด**(End_Time) **ชื่ออาจารย์ผู้สอน**(First_Name) **นามสกุลอาจารย์ผู้สอน**(Last_Name) **ชื่อรายวิชา**(Course_Name) **หมู่เรียน**(Section) **จำนวนที่นั่ง**(Seat_Capacity)

จากตารางการจองห้องเรียน (Classroom_Booking) ตารางห้องเรียน (Classroom) ตารางรายวิชาที่เปิดสอน (Course_Offering) ตารางอาจารย์ผู้สอน (Instructor)

โดยการเชื่อมตารางมีดั้งนี้

- ตาราง**การจองห้องเรียน** กับ **ห้องเรียน** ด้วยแอตทริบิวต์ ID_Classroom
- ตาราง**การจองห้องเรียน** กับ **รายวิชาที่เปิดสอน** ด้วยแอตทริบิวต์ ID_Course
- ตาราง**รายวิชาที่เปิดสอน** กับ **อาจารย์ผู้สอน** ด้วยแอตทริบิวต์ ID_Instructor

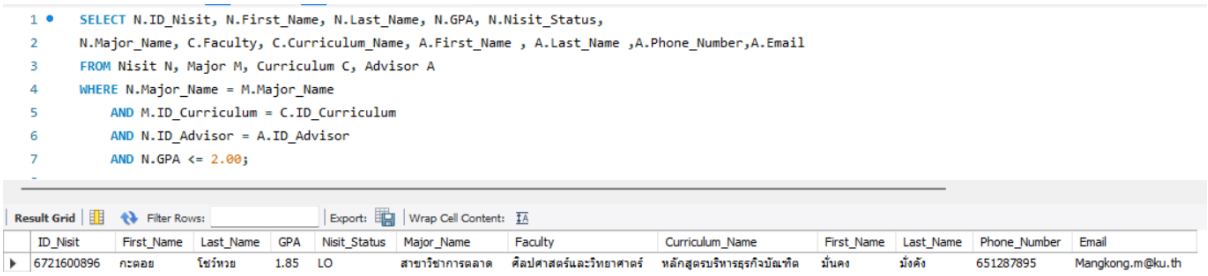
โดยเลือกเฉพาะการจองห้องเรียนเฉพาะวันจันทร์และวันศุกร์ ที่มีเวลาเริ่มเรียนตั้งแต่เวลา 08:00:00 น. ถึง 17:00:00 น. โดยเรียงลำดับตามวันและเวลาเริ่มต้น

เพื่อตรวจสอบและจัดตารางการใช้ห้องเรียนในวันและเวลาต่าง ๆ ให้มีประสิทธิภาพ และป้องกันการจองห้องเรียนซ้ำซ้อน

4. แสดงข้อมูลนิสิตโดยมี รหัสนิสิต รายชื่อนิสิต ที่มีเกรดเฉลี่ย (GPA) ต่ำกว่า 2.00 หรือเท่ากับ แต่ละคนอยู่ในสถานะใดพร้อมทั้งแสดงสาขา คณะ ที่สังกัดอยู่รวมถึงชื่อนามสกุลและข้อมูลติดต่อของอาจารย์ที่ปรึกษาที่ดูแลนิสิต

คำสั่ง SQL

```
SELECT N.ID_Nisit, N.First_Name, N.Last_Name, N.GPA, N.Nisit_Status,
N.Major_Name, C.Faculty, C.Curriculum_Name, A.First_Name ,
A.Last_Name ,A.Phone_Number,A.Email
FROM Nisit N, Major M, Curriculum C, Advisor A
WHERE N.Major_Name = M.Major_Name
AND M.ID_Curriculum = C.ID_Curriculum
AND N.ID_Advisor = A.ID_Advisor
AND N.GPA <= 2.00;
```



ID_Nisit	First_Name	Last_Name	GPA	Nisit_Status	Major_Name	Faculty	Curriculum_Name	First_Name	Last_Name	Phone_Number	Email
6721600896	กะดอย	โชว์ทวย	1.85	LO	สาขาวิชาการตลาด	ศิลปศาสตร์และวิทยาศาสตร์	หลักสูตรบริหารธุรกิจบัณฑิต	มั่นคง	มั่งคั่ง	651287895	Mangkong.m@ku.th

คำอธิบาย

ทำการค้นหา **รหัสนิสิต**(ID_Nitsit) **ชื่อนิสิต**(First_Name) **นามสกุลนิสิต**(Last_Name) GPA(GPA) **สถานะนิสิต**(Nisit_Status) **ชื่อสาขาวิชา**(Major_Name) **คณะ**(Faculty) **ชื่อหลักสูตร**(Curriculum_Name) **ชื่ออาจารย์ที่ปรึกษา**(First_Name) **นามสกุลอาจารย์ที่ปรึกษา**(Last_Name) **เบอร์โทรอาจารย์ที่ปรึกษา**(Phone_Number) **อีเมลอาจารย์ที่ปรึกษา**(Email)

จากตารางนิสิต (Nisit) ตารางหลักสูตร (Curriculum) ตารางสาขาวิชา (Major) ตารางอาจารย์ที่ปรึกษา(Advisor)

โดยการเชื่อมตารางมีดังนี้

- ตาราง**นิสิต** กับ **สาขาวิชา** ด้วยแอตทริบิวต์ Major_Name
- ตาราง**สาขาวิชา** กับ **หลักสูตร** ด้วยแอตทริบิวต์ ID_Curriculum
- ตาราง**นิสิต** กับ **อาจารย์ที่ปรึกษา** ด้วยแอตทริบิวต์ ID_Advisor

โดยจะเลือกเฉพาะนิสิตที่มี GPA น้อยกว่าหรือเท่ากับ 2.00 และแสดงข้อมูลนิสิตพร้อมข้อมูลอาจารย์ที่ปรึกษาที่ดูแลนิสิตอยู่

เพื่อติดตามและช่วยเสริมสร้างผลการเรียนให้กับนิสิตที่มีผลการเรียนต่ำให้บรรลุเกณฑ์มาตรฐานการศึกษาของมหาวิทยาลัย

5. แสดงข้อมูลทั้งหมดของนิสิตที่ทำการลงทะเบียนเรียนในวันที่12-14 พฤศจิกายน พ.ศ. 2567 และรายละเอียดการลงทะเบียนเรียน

คำสั่ง SQL

```
SELECT N.ID_Nisit,N.First_Name,N.Last_Name,CR.ID_Course,CR.Registration_Day,
CR.Registration_Month,CR.Registration_Year,CR.Registration_Time,TP.Payment_Day,
TP.Payment_Month,TP.Payment_Year,TP.Payment_Time,TP.Payment_Method,TP.Payme
nt_Status
FROM Nisit N,Tuition_Payment TP,Course_Registration CR
WHERE N.ID_Nisit = TP.ID_Nisit
AND TP.ID_Payment = CR.ID_Payment
AND CR.Registration_Day BETWEEN 12
AND 14 AND CR.Registration_Month = 11
AND CR.Registration_Year = 2567
ORDER BY CR.Registration_Year,CR.Registration_Month,CR.Registration_Day,
CR.Registration_Time;
```

```
1 • SELECT N.ID_Nisit,N.First_Name,N.Last_Name,CR.ID_Course,CR.Registration_Day,CR.Registration_Month,CR.Registration_Year,
2   CR.Registration_Time,TP.Payment_Day,TP.Payment_Month,TP.Payment_Year,TP.Payment_Time,TP.Payment_Method,TP.Payment_Status
3   FROM Nisit N,Tuition_Payment TP,Course_Registration CR
4   WHERE N.ID_Nisit = TP.ID_Nisit
5   AND TP.ID_Payment = CR.ID_Payment
6   AND CR.Registration_Day BETWEEN 12 AND 14
7   AND CR.Registration_Month = 11
8   AND CR.Registration_Year = 2567
9   ORDER BY CR.Registration_Year,CR.Registration_Month,CR.Registration_Day,CR.Registration_Time;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ID_Nisit	First_Name	Last_Name	ID_Course	Registration_Day	Registration_Month	Registration_Year	Registration_Time	Payment_Day	Payment_Month	Payment_Year	Payment_Time	Payment_Method	Payment_Status
6721600896	กะลอย	โชว์หวย	01418288	12	11	2567	09:12:00	20	11	2567	07:55:00	MB	SC
6621600221	ศิริพร	ฮ่าไพพงศ์	01418284	12	11	2567	09:20:00	12	10	2567	03:20:00	PAYS	DP
6712594569	จักกี้	โกโก้ครั้น	01481339	13	11	2567	09:05:00	25	11	2567	12:43:00	NL	SC
6721604593	ตะนอย	แซ่เฮ้ยว	01264889	14	11	2567	09:03:00	23	10	2567	11:25:00	NL	SC

คำอธิบาย

ทำการค้นหา **รหัสนิสิต**(ID_Nisit) **ชื่อนิสิต**(First_Name) **นามสกุลนิสิต**(Last_Name) **รหัสรายวิชาที่เปิดสอน**(ID_Course) **วันที่ลงทะเบียน**(Registration_Day) **เดือนที่ลงทะเบียน**(Registration_Month) **ปีที่ลงทะเบียน**(Registration_Year) **เวลาที่ลงทะเบียน**(Registration_Time) **วันที่ชำระเงิน**(Payment_Day) **เดือนที่ชำระเงิน**(Payment_Month) **ปีที่ชำระเงิน**(Payment_Year) **เวลาที่ชำระเงิน**(Payment_Time) **ช่องทางการชำระเงิน**(Payment_Method) **สถานะการชำระเงินค่าเทอม**(Payment_Status)

จากตารางนิสิต (Nisit) ตารางการลงทะเบียนเรียน (Course_Registration) ตารางการชำระเงินค่าเทอม (Tuition_Payment)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**นิสิต** กับ **การชำระเงินค่าเทอม** ด้วยแอตทริบิวต์ ID_Nisit
- ตาราง**การลงทะเบียนเรียน** กับ **การชำระเงินค่าเทอม** ด้วยแอตทริบิวต์ ID_Payment

โดยเลือกเฉพาะนิสิตที่ทำการลงทะเบียนเรียนในวันที่ 12 – 14 พฤษศจิกายน 2567 และ แสดงสถานะการชำระเงินของนิสิตแต่ละคน

เพื่อให้ทราบว่าในช่วงเวลาดังกล่าวมีนิสิตคนใดบ้างที่ทำการลงทะเบียนเรียนและสามารถตรวจสอบจำนวนการลงทะเบียนของรายวิชาต่างๆ ได้ เพื่อตรวจสอบความถูกต้องให้เป็นไปตามเกณฑ์และข้อกำหนดของรายวิชานั้นๆ

6. แสดงรายละเอียดห้องเรียนที่ถูกจองและมีสถานะจองสำเร็จโดยห้องเรียนนั้นจะรองรับนิสิตได้มากกว่าหรือเท่ากับ 60 คนแต่ไม่เกิน 90 คนและแสดงรายชื่ออาจารย์ผู้สอนของรายวิชาที่ทำการจองไป

คำสั่ง SQL

```
SELECT CB.ID_Booking, CB.ID_Classroom, CL.Building, CL.Room_Number,
CL.Seat_Capacity, CB.Class_Day, CB.Start_Time, CB.End_Time, CO.Course_Name,
I.First_Name, I.Last_Name
FROM Classroom_Booking CB,Classroom CL,Course_Offering CO,Instructor I
WHERE CB.ID_Classroom = CL.ID_Classroom
AND CB.ID_Course = CO.ID_Course
AND CO.ID_Instructor = I.ID_Instructor
AND CB.Booking_Status = 'SC'
AND CL.Seat_Capacity BETWEEN 60 AND 90;
```

```
1 ●  SELECT CB.ID_Booking, CB.ID_Classroom, CL.Building, CL.Room_Number, CL.Seat_Capacity,
2    CB.Class_Day, CB.Start_Time, CB.End_Time, CO.Course_Name, I.First_Name, I.Last_Name
3    FROM Classroom_Booking CB,Classroom CL,Course_Offering CO,Instructor I
4    WHERE CB.ID_Classroom = CL.ID_Classroom
5    AND CB.ID_Course = CO.ID_Course
6    AND CO.ID_Instructor = I.ID_Instructor
7    AND CB.Booking_Status = 'SC'
8    AND CL.Seat_Capacity BETWEEN 60 AND 90;
9
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ID_Booking	ID_Classroom	Building	Room_Number	Seat_Capacity	Class_Day	Start_Time	End_Time	Course_Name	First_Name	Last_Name
JH1493	LH2-204	LH2	204	70	Sat	17:00:00	20:00:00	แคลคูลัสI	มั่นคง	หวัง
JH1494	LH4-201	LH4	201	60	Mon	13:00:00	16:00:00	การพัฒนาเว็บแอปพลิเคชัน	สมศรี	ก้อง
JH1495	LH2-202	LH2	202	60	Tue	14:00:00	15:00:00	การวิเคราะห์และออกแบบระบบ	ภพ	อดิตชาติ

คำอธิบาย

ทำการค้นหา **รหัสการจองห้อง**(ID_Booking) **รหัสห้องเรียน**(ID_Classroom) **อาคาร**(Building) **เลขห้อง**(Room_Number) **จำนวนที่นั่ง**(Seat_Capacity) **วันที่เรียน**(Class_Day) **เวลาเริ่ม**(Start_Time) **เวลาสิ้นสุด**(End_Time) **ชื่อรายวิชาที่เปิดสอน**(Course_Name) **ชื่ออาจารย์ผู้สอน**(Instructor_First_Name) **นามสกุลอาจารย์ผู้สอน**(Instructor_Last_Name)

จากตารางการจองห้องเรียน (Classroom_Booking) ตารางห้องเรียน (Classroom) ตารางรายวิชาที่เปิดสอน (Course_Offering) ตารางอาจารย์ผู้สอน (Instructor)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**การจองห้องเรียน** กับ **ห้องเรียน** ด้วยแอตทริบิวต์ ID_Classroom
- ตาราง**การจองห้องเรียน** กับ **รายวิชาที่เปิดสอน** ด้วยแอตทริบิวต์ ID_Course
- ตาราง **รายวิชาที่เปิดสอน** กับ **อาจารย์ผู้สอน** ด้วยแอตทริบิวต์ ID_ Instructor

โดยเลือกเฉพาะห้องเรียนที่รองรับนิสิตได้มากกว่าหรือเท่ากับ 60 คนแต่ไม่เกิน 90 คน

เพื่อใช้ในการบริหารจัดการห้องเรียนให้มีประสิทธิภาพมากขึ้นและตรวจสอบและยืนยันว่าอาจารย์ผู้สอนที่เกี่ยวข้องกับการจองห้องเรียนเป็นใคร

และใช้ข้อมูลทางสถิติเกี่ยวกับข้อมูลของห้องเรียนที่มีการจองมากที่สุดในการวางแผนเพิ่มห้องเรียนในอนาคต เพื่อให้สามารถจัดการทรัพยากรการเรียนการสอนได้อย่างมีประสิทธิภาพ

7. ต้องการทราบรายละเอียดของรหัสวิชา 1418247 หมู่802 โดยจะแสดงชื่อ-นามสกุลอาจารย์ผู้สอน รหัสห้องเรียน ประเภทห้องเรียน วัน-เวลาเรียน พร้อมเบอร์โทรและอีเมลติดต่อด้วย

คำสั่ง SQL

```
SELECT i.First_Name, i.Last_Name, cb.ID_Classroom, r.Classroom_Type,
cb.Class_Day,   cb.Start_Time, cb.End_Time, i.Phone_Number, i.Email
FROM Classroom_Booking cb, Classroom r, Course_Offering co, Instructor i
WHERE cb.ID_Classroom = r.ID_Classroom
AND cb.ID_Course = co.ID_Course
AND co.ID_Instructor = i.ID_Instructor
AND cb.ID_Course = '01418247'
AND cb.Section = '802'
ORDER BY cb.Class_Day, cb.Start_Time;
```

```
1 •  SELECT i.First_Name, i.Last_Name, cb.ID_Classroom, r.Classroom_Type,
2    cb.Class_Day, cb.Start_Time, cb.End_Time, i.Phone_Number, i.Email
3    FROM Classroom_Booking cb, Classroom r, Course_Offering co, Instructor i
4    WHERE cb.ID_Classroom = r.ID_Classroom
5        AND cb.ID_Course = co.ID_Course
6        AND co.ID_Instructor = i.ID_Instructor
7        AND cb.ID_Course = '01418247'
8        AND cb.Section = '802'
9    ORDER BY cb.Class_Day, cb.Start_Time;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

First_Name	Last_Name	ID_Classroom	Classroom_Type	Class_Day	Start_Time	End_Time	Phone_Number	Email
วินัย	ลี	SC4-302	LB	Sun	08:00:00	09:30:00	0971235789	Winai.w@ku.th

คำอธิบาย

ทำการค้นหา **ชื่ออาจารย์ผู้สอน**(First_Name) **นามสกุลอาจารย์ผู้สอน**(Last_Name) **รหัสห้องเรียน**(ID_Classroom) **ประเภทห้องเรียน**(Classroom_Type) **วันที่เรียน**(Class_Day) **เวลาเริ่ม**(Start_Time) **เวลาสิ้นสุด**(End_Time) **เบอร์โทรติดต่อ**(Phone_Number) **อีเมลติดต่อของอาจารย์ผู้สอน**(Email)

จากตารางการจองห้องเรียน (Classroom_Booking) ตารางห้องเรียน (Classroom) ตารางรายวิชาที่เปิดสอน (Course_Offering) ตารางอาจารย์ผู้สอน (Instructor)

โดย การเชื่อมตารางมีดั้งนี้

- ตาราง**การจองห้องเรียน** กับ **ห้องเรียน** โดยใช้แอตทริบิวต์ ID_Classroom
- ตาราง**การจองห้องเรียน** กับ **รายวิชาที่เปิดสอน** โดยใช้แอตทริบิวต์ ID_Course
- ตาราง**รายวิชาที่เปิดสอน** กับ **อาจารย์ผู้สอน** โดยใช้แอตทริบิวต์ ID_Instructor

โดยเลือกเฉพาะรหัสวิชา 01418247 และหมู่เรียน 802 ซึ่งจะแสดงรายละเอียดการเรียนของวิชาดังกล่าว โดยจัดเรียงข้อมูลตามวันที่เรียน เวลาเริ่มเรียน พร้อมกับข้อมูลช่องทางการติดต่อของอาจารย์ผู้สอนและห้องเรียน

เพื่อเป็นข้อมูลสำหรับนิสิตต้องการทราบข้อมูลเพิ่มเติมเกี่ยวกับเนื้อหาวิชาหรือรายละเอียดเกี่ยวกับการเรียนการสอน ซึ่งอาจจะช่วยในการเตรียมตัวสำหรับการเรียนหรือสอบ

8. แสดงข้อมูลการลงทะเบียนของนิสิตชื่อ ศิริพร อำไพพงศ์ โดยต้องการทราบว่า ปัจจุบัน ลงทะเบียนรายวิชาอะไรไปแล้วบ้าง และแสดงชื่ออาจารย์ผู้สอนของแต่ละรายวิชาที่นิสิตลงทะเบียน

คำสั่ง SQL

```
SELECT CR.ID_Course,CO.Course_Name,I.First_Name ,I.Last_Name
FROM Nisit N, Tuition_Payment T, Course_Registration CR, Course_Offering CO,
Instructor I
WHERE T.ID_Nisit = N.ID_Nisit
AND T.ID_Payment = CR.ID_Payment
AND CR.ID_Course = CO.ID_Course
AND CO.ID_Instructor = I.ID_Instructor
AND N.First_Name = 'ศิริพร'
AND N.Last_Name = 'อำไพพงศ์';
```

```
1 •  SELECT CR.ID_Course, CO.Course_Name, I.First_Name ,I.Last_Name
2    FROM Nisit N, Tuition_Payment T, Course_Registration CR, Course_Offering CO, Instructor I
3    WHERE T.ID_Nisit = N.ID_Nisit
4    AND T.ID_Payment = CR.ID_Payment
5    AND CR.ID_Course = CO.ID_Course
6    AND CO.ID_Instructor = I.ID_Instructor
7    AND N.First_Name = 'ศิริพร'
8    AND N.Last_Name = 'อำไพพงศ์';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ID_Course	Course_Name	First_Name	Last_Name
01418284	แคลคูลัสI	มั่นคง	หวัง
01418244	การพัฒนาเว็บแอปพลิเคชัน	สมศรี	ก้อง
01418245	การวิเคราะห์และออกแบบระบบ	ภพ	อดิตชาติ

คำอธิบาย

ทำการค้นหา **รหัสรายวิชา**(ID_Course) **ชื่อรายวิชา**(Course_Name) **ชื่ออาจารย์ผู้สอน**(First_Name) **นามสกุลอาจารย์ผู้สอน**(Last_Name)
จากตารางนิสิต (Nisit) ตารางการชำระเงินค่าเทอม (Tuition_Payment) ตารางการลงทะเบียนเรียน (Course_Registration) ตารางรายวิชาที่เปิดสอน (Course_Offering) ตารางอาจารย์ (Instructor)
โดย การเชื่อมตารางมีดังนี้

- ตาราง**นิสิต** กับ **ชำระเงินค่าเทอม** โดยใช้แอตทริบิวต์ ID_Nisit
- ตาราง**การชำระเงินค่าเทอม** กับ **การลงทะเบียนเรียน** โดยใช้แอตทริบิวต์ ID_Payment

- ตาราง**การลงทะเบียนเรียน** กับ **รายวิชาที่เปิดสอน** โดยใช้แอตทริบิวต์ ID_Course
- ตาราง**รายวิชาที่เปิดสอน** กับ **อาจารย์ผู้สอน** โดยใช้แอตทริบิวต์ ID_Instructor

โดยเลือกเฉพาะรายการการลงทะเบียนเรียนของนิสิต ชื่อ ศิริพร นามสกุล อำไพพงศ์ เท่านั้น

เพื่อช่วยให้อาจารย์ที่ปรึกษาสามารถติดตามการศึกษาของนิสิตที่อยู่ในการดูแล และให้คำแนะนำที่เหมาะสม วางแผนการศึกษาที่เหมาะสม เพื่อช่วยเสริมการเรียนของนิสิต

9. ขอทราบข้อมูลของนิสิตในสาขาวิทยาการคอมพิวเตอร์ โดยต้องการทราบ ชื่อ-นามสกุล ชั้นปี ประเภทภาค อีเมลติดต่อ ค่าเทอมนิสิต รวมถึงชื่อ-นามสกุล และอีเมลติดต่อของอาจารย์ที่ปรึกษา

คำสั่ง SQL

```
SELECT N.First_Name, N.Last_Name, N.Year_Level, N.Program_Type, N.Email,
T.Net_Amount AS Tuition_Fee, A.First_Name AS Advisor_First_Name, A.Last_Name AS
Advisor_Last_Name, A.Email AS Advisor_Email
FROM Nisit N, Tuition_Payment T, Advisor A
WHERE N.ID_Nisit = T.ID_Nisit
AND N.ID_Advisor = A.ID_Advisor
AND N.Major_Name = 'สาขาวิชาวิทยาการคอมพิวเตอร์'
AND N.Nisit_Status = 'ST';
```

```
1 •  SELECT N.First_Name, N.Last_Name, N.Year_Level, N.Program_Type, N.Email, T.Net_Amount AS Tuition_Fee,
2    A.First_Name AS Advisor_First_Name, A.Last_Name AS Advisor_Last_Name, A.Email AS Advisor_Email
3    FROM Nisit N, Tuition_Payment T, Advisor A
4    WHERE N.ID_Nisit = T.ID_Nisit
5        AND N.ID_Advisor = A.ID_Advisor
6        AND N.Major_Name = 'สาขาวิชาวิทยาการคอมพิวเตอร์'
7        AND N.Nisit_Status = 'ST';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

First_Name	Last_Name	Year_Level	Program_Type	Email	Tuition_Fee	Advisor_First_Name	Advisor_Last_Name	Advisor_Email
สิริพร	ฮ่าไพพงศ์	2	RP	Siripon.a@ku.th	16900	ใจดี	มีดัง	Jaidee.me@ku.th
สิวลักษณ์	ชื่นวัดร	1	SP	Siwaluk.ch@ku.th	40000	สงยศ	สุขสม	Songyod.s@ku.th

คำอธิบาย

ทำการค้นหา **ชื่อนิสิต**(First_Name) **นามสกุลนิสิต**(Last_Name) **ชั้นปีของนิสิต**(Year_Level) **ประเภทภาคการศึกษา**(Program_Type) **อีเมลนิสิต**(Email) **ค่าเทอมที่นิสิตต้องชำระ**(Tuition_Fee) **ชื่ออาจารย์ที่ปรึกษา**(Advisor_First_Name) **นามสกุลอาจารย์ที่ปรึกษา**(Advisor_Last_Name) **อีเมลของอาจารย์ที่ปรึกษา**(Advisor_Email)

จากตารางนิสิต (Nisit) ตารางการชำระเงินค่าเทอม (Tuition_Payment) ตารางอาจารย์ที่ปรึกษา (Advisor)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**นิสิต** กับ **การชำระเงินค่าเทอม** ด้วยแอตทริบิวต์ ID_Nisit
- ตาราง**นิสิต** กับ **อาจารย์ที่ปรึกษา** ด้วยแอตทริบิวต์ ID_Advisor

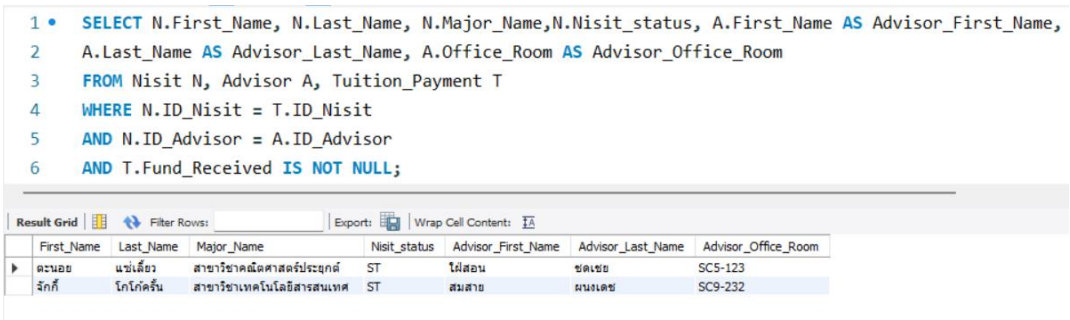
โดยเลือกเฉพาะนิสิตที่สังกัดอยู่ใน สาขาวิชาวิทยาการคอมพิวเตอร์ที่มีสถานะกำลังศึกษาอยู่ และแสดงข้อมูลในการติดต่อและติดตามนิสิต

เพื่อช่วยบริหารจัดการด้านการศึกษาและการจัดการอุปกรณ์และทรัพยากรที่เกี่ยวข้องกับการใช้ห้องเรียนอย่างมีประสิทธิภาพเหมาะสม สำหรับนิสิตสาขาวิชาวิทยาการคอมพิวเตอร์ในแต่ละชั้นปี

10.แสดงรายชื่อนิสิตทั้งหมดที่ได้รับกองทุนการศึกษาจากสถานศึกษา โดยต้องมีรายละเอียดสาขาวิชาที่นิสิตสังกัดอยู่ สถานะการศึกษาของนิสิต และต้องการทราบชื่อ-นามสกุล ห้องทำงานของอาจารย์ที่ปรึกษาที่ดูแลนิสิต

คำสั่ง SQL

```
SELECT N.First_Name, N.Last_Name, N.Major_Name, N.Nisit_status, A.First_Name AS
Advisor_First_Name, A.Last_Name AS Advisor_Last_Name, A.Office_Room AS
Advisor_Office_Room
FROM Nisit N, Advisor A, Tuition_Payment T
WHERE N.ID_Nisit = T.ID_Nisit
AND N.ID_Advisor = A.ID_Advisor
AND T.Fund_Received IS NOT NULL;
```



คำอธิบาย

ทำการค้นหา **ชื่อนิสิต**(First_Name) **นามสกุลนิสิต**(Last_Name) **สถานะนิสิต**(Nisit_status) **ชื่อสาขาวิชาที่นิสิตสังกัดอยู่**(Major_Name) **ชื่ออาจารย์ที่ปรึกษา**(Advisor_First_Name) **นามสกุลอาจารย์ที่ปรึกษา**(Advisor_Last_Name) **ห้องทำงานของอาจารย์ที่ปรึกษา** (Advisor_Office_Room)

จากตารางนิสิต (Nisit) ตารางอาจารย์ที่ปรึกษา (Advisor) ตารางการชำระเงินค่าเทอม (Tuition_Payment)

โดย การเชื่อมตารางมีดังนี้

- ตาราง**นิสิต** กับ **การชำระเงินค่าเทอม** ด้วยแอตทริบิวต์ ID_Nisit
- ตาราง**นิสิต** กับ **อาจารย์ที่ปรึกษา** ด้วยแอตทริบิวต์ ID_Advisor

โดยเลือกเฉพาะนิสิตที่ได้รับกองทุนการศึกษาจากสถานศึกษา และใช้ข้อมูลอาจารย์ที่ปรึกษาเพื่อติดตามให้คำแนะนำนิสิต

เพื่อช่วยให้สถานศึกษาและอาจารย์ที่ปรึกษาสามารถติดตามการแจกจ่ายกองทุนและตรวจสอบการใช้จ่ายในทางที่ถูกต้องได้