

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ฐานข้อมูลเชิงวัตถุสัมพันธ์และการประยุกต์เชิงเวลา
Object-Relational Database And Temporal Application



ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2542

เลขหมึก.....
เลขทะเบียน..... 37059
วัน, เดือน, ปี 30.8.2543

สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2542

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง


เรื่อง ฐานข้อมูลเชิงวัตถุสัมพันธ์และการประยุกต์เชิงเวลา

Object-Relational Database And Temporal Application

ผู้จัดทำ

1. นายปิ่นมัทธ ธนชัยนนท์ รหัสประจำตัว 39014094
2. นางสาวสุมลรัตน์ สดางศ์วิวัฒน์ รหัสประจำตัว 39014603




(รศ.ดร.ศุภมิตร จิตตะยโสธร)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานข้อมูลเชิงวัตถุสัมพันธ์และการประยุกต์เชิงเวลา

นายปัทมทัต รัชญนนท์ 39014094

นางสาวศุภลรัตน์ สดาวงศ์วิวัฒน์ 39014603

รศ.ดร.ศุภมิตร จิตตะยโสธร อาจารย์ที่ปรึกษา

ปีการศึกษา 2542

บทคัดย่อ

ในระบบฐานข้อมูลเชิงสัมพันธ์ การเก็บข้อมูลจะเก็บเฉพาะความจริง (Fact) ก็คือข้อมูลที่เป็นจริงในเวลาปัจจุบันเท่านั้น โดยข้อมูลเดิมที่ถูกเปลี่ยนแปลงหรือถูกลบก็จะหายไป และจะไม่เก็บข้อมูลที่จะเป็นจริงในอนาคตแต่ยังไม่เป็นจริงในปัจจุบัน ฐานข้อมูลเชิงเวลาสามารถจัดการกับสิ่งเหล่านี้ได้โดยการเพิ่มส่วนของเวลาไปกับข้อมูล ทำให้ฐานข้อมูลเชิงเวลาสามารถเก็บข้อมูลทั้งในอดีต ปัจจุบัน และอนาคตได้ โดยฐานข้อมูลเชิงเวลาจะสนับสนุนทั้ง Temporal Data Definition Language, Temporal Data Manipulation Language, Temporal Query Language และ Temporal Constraint ซึ่งทำให้การจัดการข้อมูลเชิงเวลาทำได้อย่างสะดวก

ในปฏิญานีพนธ์ฉบับนี้ เป็นวิธีการใช้ภาษา SQL ในการจัดการกับข้อมูลเชิงเวลา โดยโปรแกรมที่พัฒนาขึ้นมาจะทำให้ผู้ใช้สามารถใช้ภาษา TSQL2 ได้ โดยโปรแกรมจะทำการแปลงภาษา TSQL2 ที่ผู้ใช้ส่งเข้ามาเป็นภาษา SQL โดยภาษา SQL ที่ได้จะสามารถนำไปใช้บนระบบจัดการฐานข้อมูลต่างๆไปได้

Object-Relational Database and Temporal Application

Panatat Tanatanyanon

Sumonrat Ladawongwiwat

Assoc. Prof. Dr.Suphamit Chittayasothon Advisor

ABSTRACT

Relational Database keeps the data that true in present time only. The modified or deleted data will go away from database and future data has never been kept in database. Temporal Database could manage this by attaching additional attribute to each data. So they can keep the past, present and future data. Temporal database has its own language : Temporal Data Definition Language, Temporal Data Manipulation Language, Temporal Query Language and Temporal Constraint then it is simple to manage temporal data.

In this thesis we are using SQL-92 language to manage temporal data. Developed application program allow users to use TSQL2 language. Our program will translate TSQL2 language to SQL-92 language. So users can execute SQL-92 code in conventional database.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้คงไม่อาจเสร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และร่วมมือจากหลายๆ ฝ่ายด้วยกัน บุคคลแรกที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้วิทยานิพนธ์นี้เสร็จลงได้ก็คือ อาจารย์ศุภมิตร จิตตะยโสธร อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา ซึ่งต้องขอขอบพระคุณเป็นอย่างมาก

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดามารดาอันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมา ในทุกๆ ด้านอันหาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณอันสุดประมาณ และขอกราบขอบพระคุณมา ณ ที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญภาพ	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของงานวิจัย	1
1.3 ขอบเขตของงานวิจัย	2
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ฐานข้อมูลเชิงเวลา	3
2.1 ฐานข้อมูลเชิงเวลา	3
2.2 ประเภทของฐานข้อมูลเชิงเวลา	4
2.3 ระบบจัดการฐานข้อมูลเชิงเวลา	6
2.4 ปัญหาในการใช้ระบบจัดการฐานข้อมูลทั่วไปจัดการข้อมูลเชิงเวลา	6
2.5 ตัวอย่างแอปพลิเคชันที่เกี่ยวข้องกับข้อมูลเชิงเวลา	7
บทที่ 3 ความซ้ำซ้อน	8
3.1 ประเภทของความซ้ำซ้อน	8
3.2 การป้องกันความซ้ำซ้อนแบบค่าเท่ากัน	9
3.3 การป้องกันความซ้ำซ้อนแบบนอนซีเควนซ์	10
3.4 การป้องกันความซ้ำซ้อนแบบปัจจุบัน	10
3.5 การป้องกันความซ้ำซ้อนแบบซีเควนซ์	11
บทที่ 4 การตอบคำถามตารางวลิตไทม์โดยใช้ภาษา SQL	12
4.1 การตอบคำถามเชิงเวลาโดยใช้ภาษา SQL	12
4.1.1 การตอบคำถามแบบทั่วๆไป	12
4.1.2 การตอบคำถามแบบปัจจุบัน	12
4.1.3 การตอบคำถามแบบซีเควนซ์	13
4.1.4 การตอบคำถามแบบนอนซีเควนซ์	13
4.2 การตอบคำถามเชิงเวลาโดยใช้การ JOIN ของภาษา SQL	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
4.2.1 การตอบคำถามแบบทั่วๆไปโดยใช้การ JOIN	14
4.2.2 การตอบคำถามแบบปัจจุบันโดยใช้การ JOIN	14
4.2.3 การตอบคำถามแบบนอนซีแควนซ์โดยใช้การ JOIN	14
4.2.4 การตอบคำถามแบบซีแควนซ์โดยใช้การ JOIN	15
บทที่ 5 การเปลี่ยนแปลงตารางวาลิดใหม่โดยใช้ภาษา SQL	19
5.1 การเปลี่ยนแปลงแบบปัจจุบัน	19
5.1.1 การลบแบบปัจจุบัน	19
5.1.2 การแก้ไขแบบปัจจุบัน	20
5.2 การเปลี่ยนแปลงแบบซีแควนซ์	22
5.2.1 การลบแบบซีแควนซ์	22
5.2.2 การแก้ไขแบบซีแควนซ์	23
5.3 การเปลี่ยนแปลงแบบนอนซีแควนซ์	25
บทที่ 6 การจัดการตารางทรานแซคชันใหม่โดยใช้ภาษา SQL	27
6.1 การรักษาตารางทรานแซคชันใหม่	28
6.2 การตอบคำถามตารางทรานแซคชันใหม่	29
6.2.1 การตอบคำถามแบบปัจจุบัน	29
6.2.2 การตอบคำถามแบบซีแควนซ์	30
6.2.3 การตอบคำถามแบบนอนซีแควนซ์	31
6.3 การเปลี่ยนแปลงตารางทรานแซคชันใหม่	31
บทที่ 7 ภาษา SQL เิงเวลา (ภาษา TSQL2)	33
7.1 ลักษณะเฉพาะที่ต้องการในภาษา SQL เิงเวลา	33
7.2 โครงสร้างประโยคของภาษา SQL เิงเวลา	33
บทที่ 8 การใช้ภาษา SQL เิงเวลา	51
8.1 Temporal Data Definition Language และ Temporal Constraint	52
8.2 Temporal Data Manipulation Language	53
8.3 Temporal Query Language	57
8.3.1 คำตอบเป็นค่าของแอททริบิวต์	58
8.3.2 คำตอบเป็นวาลิดใหม่	74
8.3.3 คำตอบเป็นค่าของแอททริบิวต์ที่มีวาลิดใหม่อยู่ด้วย	82
8.4 การแปลงภาษา SQL เิงเวลาเป็นภาษา SQL	98
บทที่ 9 การแปลงภาษา SQL เิงเวลา	100
9.1 ทฤษฎี Time Normalization	100
9.2 การนำทฤษฎี Time Normalization มาประยุกต์ใช้ในการแปลงภาษา SQL เิงเวลา	102

	หน้า
ภาคผนวก ก วิธีการใช้โปรแกรม QUERY.EXE	104
ภาคผนวก ข Source Code	110
ภาคผนวก ค การตอบคำถามเชิงเวลาด้วยภาษาที่ใช้ในการตอบคำถามเชิงเวลาแบบต่างๆ	117
บรรณานุกรม	138



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2-1 ตัวอย่างตารางในระบบจัดการฐานข้อมูลเชิงสัมพันธ์	3
ตารางที่ 2-2 ตัวอย่างตารางในระบบจัดการฐานข้อมูลเชิงเวลา	4
ตารางที่ 3-1 ตารางสถานะคนไข้	8
ตารางที่ 3-2 ตารางสแน็ปช็อตของตารางสถานะคนไข้	9
ตารางที่ 3-3 ความสัมพันธ์ของความซ้ำซ้อนแบบต่างๆ	9
ตารางที่ 4-1 ตารางวิว	12
ตารางที่ 4-2 คำตอบของตัวอย่างการตอบคำถามแบบปัจจุบัน	13
ตารางที่ 4-3 คำตอบของตัวอย่างการตอบคำถามแบบซีเควนซ์	13
ตารางที่ 4-4 คำตอบของตัวอย่างการตอบคำถามแบบนอนซีเควนซ์	14
ตารางที่ 4-5 คำตอบของตัวอย่างการตอบคำถามแบบนอนซีเควนซ์โดยใช้การ JOIN	15
ตารางที่ 4-6 คำตอบของตัวอย่างการตอบคำถามแบบซีเควนซ์โดยใช้การ JOIN	18
ตารางที่ 5-1 ตารางเพศวิว	19
ตารางที่ 5-2 ผลของตัวอย่างการลบแบบปัจจุบัน	20
ตารางที่ 6-1 ตารางดวงดาว1	27
ตารางที่ 6-2 ตารางดวงดาว2	27
ตารางที่ 6-3 ตารางดวงดาวในวันที่ 1 เดือนเมษายน ปี ค.ศ. 1994	30
ตารางที่ 6-4 คำตอบของตัวอย่างที่ 1 ของการตอบคำถามแบบซีเควนซ์	30
ตารางที่ 6-5 คำตอบของตัวอย่างที่ 2 ของการตอบคำถามแบบซีเควนซ์	31
ตารางที่ 6-6 คำตอบของตัวอย่างการตอบคำถามแบบนอนซีเควนซ์	31
ตารางที่ 8-1 ตารางพนักงาน	57
ตารางที่ 8-2 ตารางทักษะ	57
ตารางที่ 8-3 ตารางแผนก	57
ตารางที่ 9-1 ตารางพนักงาน_ชื่อ	100
ตารางที่ 9-2 ตารางพนักงาน_เงินเดือน	100
ตารางที่ 9-3 ตารางพนักงาน_แผนก	101
ตารางที่ 9-4 ตารางพนักงาน_อื่น	101
ตารางที่ 9-5 ตารางทักษะ_ทักษะ	101
ตารางที่ 9-6 ตารางแผนก_งบประมาณ	101
ตารางที่ 9-7 ตารางแผนก_ผู้จัดการ	102
ตารางที่ 9-8 ตารางเมต้า	103

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

	หน้า
รูปที่ 2-1 ฐานข้อมูลสแน็ปช็อต	5
รูปที่ 2-2 ฐานข้อมูลไบเทมโพรอล	6
รูปที่ 4-1 กรณีที่ 1	15
รูปที่ 4-2 กรณีที่ 2	16
รูปที่ 4-3 กรณีที่ 3	16
รูปที่ 4-4 กรณีที่ 4	16
รูปที่ 5-1 การเปลี่ยนแปลงเพศของวัว	19
รูปที่ 5-2 กรณีของการแก้ไขแบบปัจจุบัน	21
รูปที่ 5-3 กรณีของการลบแบบซีเควนซ์	22
รูปที่ 5-4 กรณีของการแก้ไขแบบซีเควนซ์	24
รูปที่ 8-1 แบบจำลองความสัมพันธ์ของเอนติตี	51



บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

ในปัจจุบัน การพัฒนาทางด้านการจัดการฐานข้อมูลนับว่าได้ก้าวหน้าไปมาก มีนักวิจัยในสาขานี้เป็นจำนวนมากได้ทำการวิจัยและพัฒนาทางด้านการออกแบบฐานข้อมูล รวมทั้งทางด้านการบริหารจัดการฐานข้อมูล (Database Management System : DBMS) ให้มีคุณภาพสูง และมีมาตรฐานเป็นที่ยอมรับ

อย่างไรก็ตามเป็นที่น่าสังเกตว่า การจัดการฐานข้อมูลที่ใช้กันอยู่ในปัจจุบันส่วนใหญ่นั้น ถูกออกแบบมาเพื่อเก็บข้อมูลสถานะสุดท้าย เมื่อมีการเปลี่ยนแปลงข้อมูล ข้อมูลที่มีอยู่เดิมจะถูกลบทิ้งไปจากฐานข้อมูล แม้ว่าฐานข้อมูลทั่วไปจะสามารถรองรับบางแอปพลิเคชัน (Application) ได้ดี แต่บางครั้งก็ไม่เพียงพอกับแอปพลิเคชันที่ต้องการข้อมูลอดีต (Past Data) และ/หรือข้อมูลอนาคต (Future Data) ด้วย สิ่งที่ต้องการก็คือฐานข้อมูลที่สนับสนุนการเก็บและการตอบคำถาม (Query) ข้อมูลที่เปลี่ยนแปลงตามเวลาอย่างเต็มที่ ฐานข้อมูลที่เก็บข้อมูลอดีต ข้อมูลปัจจุบัน และข้อมูลอนาคต เรียกว่า ฐานข้อมูลเชิงเวลา (Temporal Database)

เวลาที่ใช้ในฐานข้อมูลเชิงเวลาและแบบจำลองข้อมูลเชิงเวลา (Temporal Data Model) ได้แก่ ช่วงเวลาที่ข้อมูลเป็นจริง (Valid Time) ช่วงเวลาที่ข้อมูลถูกเก็บลงในฐานข้อมูล (Transaction Time) และเวลาที่กำหนดโดยผู้ใช้ (User-Defined Time) คือ เวลาที่เสมือนเป็นข้อมูลชนิดหนึ่ง เช่น วันเกิด

เวลาทั้งสามชนิดก่อให้เกิดฐานข้อมูลหลายประเภท ฐานข้อมูลดั้งเดิมซึ่งไม่สนับสนุนทั้งวาลิดไทม์และทรานแซกชันไทม์ เรียกว่า ฐานข้อมูลสแน็ปช็อต (Snapshot Database) เพราะเก็บเฉพาะข้อมูลปัจจุบันเท่านั้น ฐานข้อมูลวาลิดไทม์ (Valid-Time Database) จะเก็บวาลิดไทม์ของข้อมูลทั้งหมดตั้งแต่อดีต ปัจจุบัน จนถึงอนาคต ฐานข้อมูลทรานแซกชันไทม์ (Transaction-Time Database) เก็บเวลาในการปรับปรุงข้อมูล ดังนั้นจึงสามารถย้อนกลับ (Roll Back) ฐานข้อมูลไปยังสถานะก่อนได้ ฐานข้อมูลชนิดนี้จะเก็บบันทึกข้อผิดพลาดทั้งหมดเพื่อสามารถตรวจสอบในภายหลังได้อย่างครบถ้วน ฐานข้อมูลไบเทมโปรอล (Bitemporal Database) เก็บทั้งวาลิดไทม์และทรานแซกชันไทม์เพื่อรวมข้อดีของทั้งสองไว้ด้วยกัน

แม้ว่าระบบจัดการฐานข้อมูลบางระบบจะสนับสนุนชนิดของข้อมูลที่เป็วันและเวลา แต่ก็ไม่สามารถเรียกได้ว่าเป็นระบบจัดการฐานข้อมูลเชิงเวลา เนื่องจากถ้าต้องการข้อมูลอดีตและ/หรือข้อมูลอนาคต ผู้ใช้จะต้องจัดการเองทั้งหมด โดยปราศจากการสนับสนุนใดๆจากระบบจัดการฐานข้อมูล

และจากที่ได้กล่าวมาทั้งหมดนั้น ผู้อ่านคงจะเห็นถึงความไม่เหมาะสมของระบบจัดการฐานข้อมูลทั่วไป ความสามารถของระบบจัดการฐานข้อมูลเชิงเวลา (Temporal Database Management System : TDBMS) สามารถจะแก้ปัญหาดังกล่าวนี้ได้ ซึ่งจะได้อีกต่อไป

1.2 วัตถุประสงค์ของงานวิจัย

1.2.1 เพื่อศึกษาการนำฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database) มาประยุกต์ใช้กับงานที่เกี่ยวข้องกับข้อมูลเชิงเวลา (Temporal Data)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1.2.2 เพื่อศึกษาการนำเอาฐานข้อมูลเชิงเวลามาประยุกต์ใช้กับงานที่เกี่ยวข้องกับข้อมูลเชิงเวลา
- 1.2.3 เพื่อศึกษาการนำภาษา SQL เชิงเวลามาประยุกต์ใช้กับฐานข้อมูลเชิงวัตถุสัมพันธ์ว่าจะต้องมีการออกแบบโครงสร้างข้อมูล และตารางเมตต้า (Meta Table) อย่างไรบ้าง

1.3 ขอบเขตของงานวิจัย

โครงการนี้เป็นโครงการที่มีลักษณะเป็นงานวิจัย โดยหาวิธีต่างๆมาจัดการกับข้อมูลเชิงเวลาว่ามีวิธีการใดบ้าง โครงการนี้ไม่รวมถึงการพัฒนาภาษา SQL เชิงเวลา แต่จะอ้างอิงจากภาษาที่มีอยู่แล้วนำมาใช้งาน โครงการนี้ยังได้มีการพัฒนาโปรแกรมเพื่อแปลงภาษา SQL เชิงเวลาไปเป็นภาษา SQL เพื่อให้สามารถเขียนภาษา SQL เชิงเวลาบนระบบจัดการฐานข้อมูลต่างๆไปได้ แต่จะไม่ครอบคลุมภาษา SQL เชิงเวลาทุกกรณี เนื่องจากโปรแกรมที่พัฒนาขึ้นยังอยู่ในส่วนของการทดลอง

1.4 วิธีการดำเนินงาน

โครงการนี้จะเป็นการศึกษาทฤษฎีที่เกี่ยวข้องกับงานวิจัย โดยมีเรื่องต่างๆดังต่อไปนี้คือ

- ฐานข้อมูลเชิงเวลา ศึกษาลักษณะและคุณสมบัติต่างๆของฐานข้อมูลเชิงเวลา
- ความซ้ำซ้อนของข้อมูล ศึกษาความซ้ำซ้อนแบบต่างๆของข้อมูลเชิงเวลา
- ตารางวลิตใหม่และตารางทรานแซคชันใหม่ ศึกษาการใช้ฐานข้อมูลเชิงสัมพันธ์จัดการกับข้อมูลเชิงเวลา
- ภาษา SQL เชิงเวลา ศึกษาการใช้ฐานข้อมูลเชิงเวลาจัดการกับข้อมูลเชิงเวลา

บทที่ 2 ฐานข้อมูลเชิงเวลา

2.1 ฐานข้อมูลเชิงเวลา

ระบบจัดการฐานข้อมูลเชิงพาณิชย์ (Commercial Database Management System) เช่น ออราเคิล (Oracle) ไชเบส (Sybase) อินฟอร์มิกซ์ (Informix) มักจะเก็บข้อมูลที่เป็นจริงในเวลาปัจจุบัน จะไม่เก็บข้อมูลอดีตหรือข้อมูลอนาคต ข้อมูลอดีตคือข้อมูลซึ่งเคยถูกเก็บอยู่ในฐานข้อมูล แล้วถูกเปลี่ยนแปลงหรือลบออก ข้อมูลอดีตมักจะถูกเขียนทับด้วยข้อมูลใหม่ ส่วนข้อมูลอนาคตคือข้อมูลที่จะเป็นจริงในอนาคต แต่ไม่เป็นจริงในเวลาปัจจุบัน

ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) จะเก็บข้อมูลในตารางหรือเรียกว่ารีเลชัน (Relation) ดังนั้นฐานข้อมูลเชิงสัมพันธ์จึงประกอบด้วยตารางต่างๆ โดยแต่ละตารางประกอบด้วยแถว (Row) หรือทัพเพิล (Tuple) และคอลัมน์ (Column) หรือแอททริบิวต์ (Attribute) แต่ละแถวเก็บข้อมูลเกี่ยวกับเอนทิตี (Entity) เช่น พนักงาน แต่ละคอลัมน์เก็บคุณสมบัติของเอนทิตี เช่น ชื่อของพนักงาน เงินเดือน เป็นต้น ยกตัวอย่างตารางในระบบจัดการฐานข้อมูลเชิงสัมพันธ์ดังตารางที่ 2-1

รหัสพนักงาน	ชื่อ	แผนก	เงินเดือน
i0	สุรัชย์	การตลาด	12000
12	คารา	วิจัย	10000
13	วิวัฒน์	การตลาด	15000

ตารางที่ 2-1 ตัวอย่างตารางในระบบจัดการฐานข้อมูลเชิงสัมพันธ์

ระบบจัดการฐานข้อมูลเชิงวัตถุ (Object-Oriented Database Management System) เก็บข้อมูลเกี่ยวกับเอนทิตีเป็นวัตถุ (Object) ดังนั้นพนักงานแต่ละคนจึงเป็นวัตถุ โดยชนิด (Type) ของวัตถุระบุคุณสมบัติที่วัตถุมี ดังนั้นวัตถุพนักงานจึงมีคุณสมบัติ เช่น ชื่อ เงินเดือน เป็นต้น ชุด (Set) ของวัตถุชนิดเดียวกัน เรียกว่า คอลเลกชัน (Collection) ดังนั้นในระบบจัดการฐานข้อมูลเชิงวัตถุฐานข้อมูลจึงเก็บชุดของคอลเลกชัน

ส่วนข้อมูลเชิงเวลาที่ถูกรับในฐานข้อมูลเชิงเวลานั้น แตกต่างจากข้อมูลที่ถูกเก็บในฐานข้อมูลทั่วไปตรงที่เก็บช่วงเวลาไว้กับข้อมูล (Timestamp) เพื่อแสดงว่าข้อมูลถูกเก็บในฐานข้อมูลเมื่อใดหรือข้อมูลเป็นจริงในช่วงเวลาใด ดังที่ได้กล่าวไปแล้วข้างต้น ฐานข้อมูลทั่วไปจะเก็บเฉพาะข้อมูลที่เป็นจริงในเวลาปัจจุบันเท่านั้น โดยจะไม่เก็บข้อมูลอดีตและข้อมูลอนาคต ดังนั้นการที่เก็บช่วงเวลาไว้กับข้อมูลจึงสามารถที่จะเก็บข้อมูลอดีตและข้อมูลอนาคตได้

ช่วงเวลาที่เก็บในฐานข้อมูลเชิงเวลามี 3 แบบคือ วาลิดไทม์ ทรานแซกชันไทม์ และเวลาที่กำหนดโดยผู้ใช้ โดยวาลิดไทม์ คือ ช่วงเวลาที่ข้อมูลเป็นจริง ส่วนทรานแซกชันไทม์ คือ ช่วงเวลาที่ข้อมูลถูกเก็บในฐานข้อมูล สังเกตว่าช่วงเวลา 2 ช่วงนี้ไม่จำเป็นต้องเป็นช่วงเวลาเดียวกันสำหรับข้อมูลเดียวกัน ตัวอย่างไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น สมมติว่ามีฐานข้อมูลเชิงเวลาที่เก็บข้อมูลของศตวรรษที่ 19 วาลิดไทม์ของข้อมูลเหล่านี้คือช่วงเวลาใดๆระหว่างปี ค.ศ. 1900 และ 1999 ส่วนทรานแซคชันไทม์จะเป็นเวลาที่เรใส่ข้อมูลเข้าไปในฐานข้อมูล ตัวอย่างเช่น วันที่ 21 เดือนมกราคม ปี ค.ศ. 2000 เป็นต้น ส่วนเวลาที่กำหนดโดยผู้ใช้ คือ เวลาที่เสมือนเป็นข้อมูลชนิดหนึ่ง เช่น วันเกิด

สมมติว่าต้องการเก็บข้อมูลเกี่ยวกับพนักงาน ยกตัวอย่างตารางในระบบจัดการฐานข้อมูลเชิงเวลาดังตารางที่ 2-2

รหัสพนักงาน	ชื่อ	แผนก	เงินเดือน	เวลาเริ่มต้น	เวลาสิ้นสุด
10	สุรชัย	วิจัย	11000	03/01/1985	12/31/1990
10	สุรชัย	การตลาด	11000	01/01/1991	12/31/1993
10	สุรชัย	การตลาด	12000	01/01/1994	NOW
11	พิพัฒน์	วิจัย	10000	02/01/1988	10/31/1995
12	คารา	วิจัย	10000	04/01/1991	NOW
13	วิวัฒน์	การตลาด	15000	06/01/1988	NOW
14	สุรีย์	วิจัย	20000	01/01/2002	NULL

ตารางที่ 2-2 ตัวอย่างตารางในระบบฐานข้อมูลเชิงเวลา

ตารางวาลิดไทม์ (Valid-Time Table) ข้างต้นนี้เก็บประวัติของพนักงาน จะเห็นว่าระหว่างวันที่ 1 เดือนมีนาคม ปี ค.ศ. 1985 ถึงวันที่ 31 เดือนธันวาคม ปี ค.ศ. 1990 พนักงานชื่อสุรชัยทำงานอยู่ในแผนกวิจัยและมีเงินเดือน 11000 บาท จากนั้นได้ย้ายไปทำงานที่แผนกการตลาดโดยยังคงมีเงินเดือน 11000 บาท และในวันที่ 1 เดือนมกราคม ปี ค.ศ. 1994 สุรชัยได้รับเงินเดือนเพิ่มขึ้นเป็น 12000 บาท สังเกตว่าตารางนี้สามารถเก็บข้อมูลอดีตได้แล้ว จะเห็นว่าพิพัฒน์ทำงานตั้งแต่วันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1988 จนถึงวันที่ 31 เดือนตุลาคม ปี ค.ศ. 1995 ซึ่งในฐานข้อมูลทั่วไป ข้อมูลนี้จะถูกลบออกเมื่อพิพัฒน์ลาออกจากบริษัท และจะเห็นว่ามีกรวางแผนที่จะจ้างพนักงานชื่อสุรีย์ให้มาทำงานที่แผนกวิจัยตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 2002 ซึ่งในฐานข้อมูลทั่วไปจะไม่มีข้อมูลนี้อยู่ในฐานข้อมูล

ข้อมูลที่ใส่เข้าไปอาจจะยังไม่ทราบเวลาสิ้นสุดของวาลิดไทม์ (T_E) จึงกำหนดให้เวลาสิ้นสุดของวาลิดไทม์เป็นปัจจุบัน (Now) ถ้าเวลาเริ่มต้นของวาลิดไทม์ (T_S) เป็นอดีต และให้เวลาสิ้นสุดของวาลิดไทม์เป็นนัล (Null) ถ้าเวลาเริ่มต้นของวาลิดไทม์เป็นอนาคต

$$T_E := \text{NOW}, \text{ if } T_S \leq \text{NOW}$$

$$T_E := \text{NULL}, \text{ if } T_S > \text{NOW}$$

โดยเป็นที่ตกลงกันในหมู่นักวิจัยว่า จะกำหนดให้ค่า NOW เท่ากับวันที่ 31 เดือนธันวาคม ปี ค.ศ.

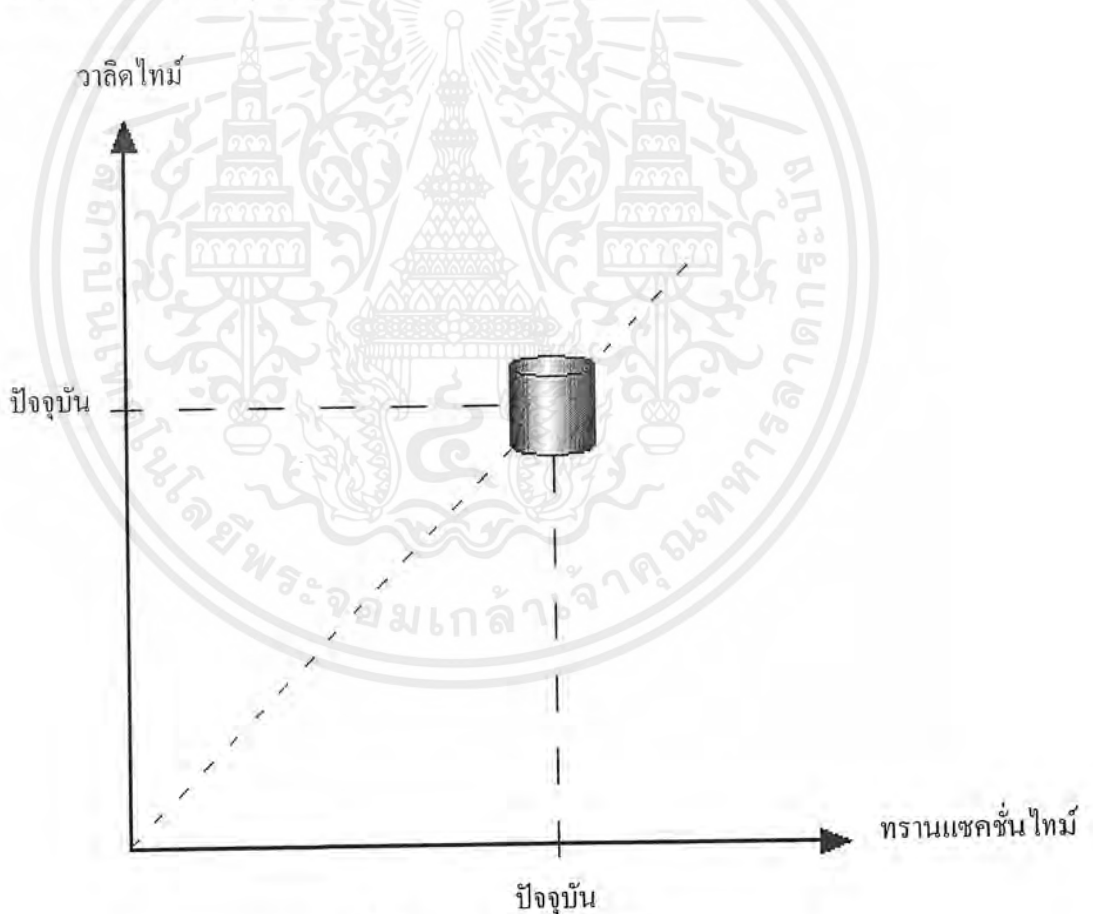
เอกสารที่ 9999 เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ประเภทของฐานข้อมูลเชิงเวลา

วาลิดไทม์และทรานแซคชันไทม์ก่อให้เกิดฐานข้อมูลเชิงเวลาหลายลักษณะดังต่อไปนี้

- 2.2.1 ฐานข้อมูลวาลิดไทม์ (Valid-Time Database) หรือฐานข้อมูลประวัติ (Historical Database) เก็บเฉพาะวาลิดไทม์
- 2.2.2 ฐานข้อมูลทรานแซคชันไทม์ (Transaction-Time Database) หรือฐานข้อมูลย้อนกลับ (Rollback Database) เก็บเฉพาะทรานแซคชันไทม์
- 2.2.3 ฐานข้อมูลไบเทมโปรอล (Bitemporal Database) เก็บทั้งวาลิดไทม์และทรานแซคชันไทม์

ดังที่ได้กล่าวไว้แล้วข้างต้น ระบบจัดการฐานข้อมูลเชิงพาณิชย์เก็บข้อมูลที่เป็นจริงในปัจจุบันเท่านั้น ฐานข้อมูลแบบนี้มักจะเรียกว่า ฐานข้อมูลสแน็ปช็อต โดยฐานข้อมูลสแน็ปช็อตในแบบของวาลิดไทม์และทรานแซคชันไทม์แสดงดังรูปที่ 2-1



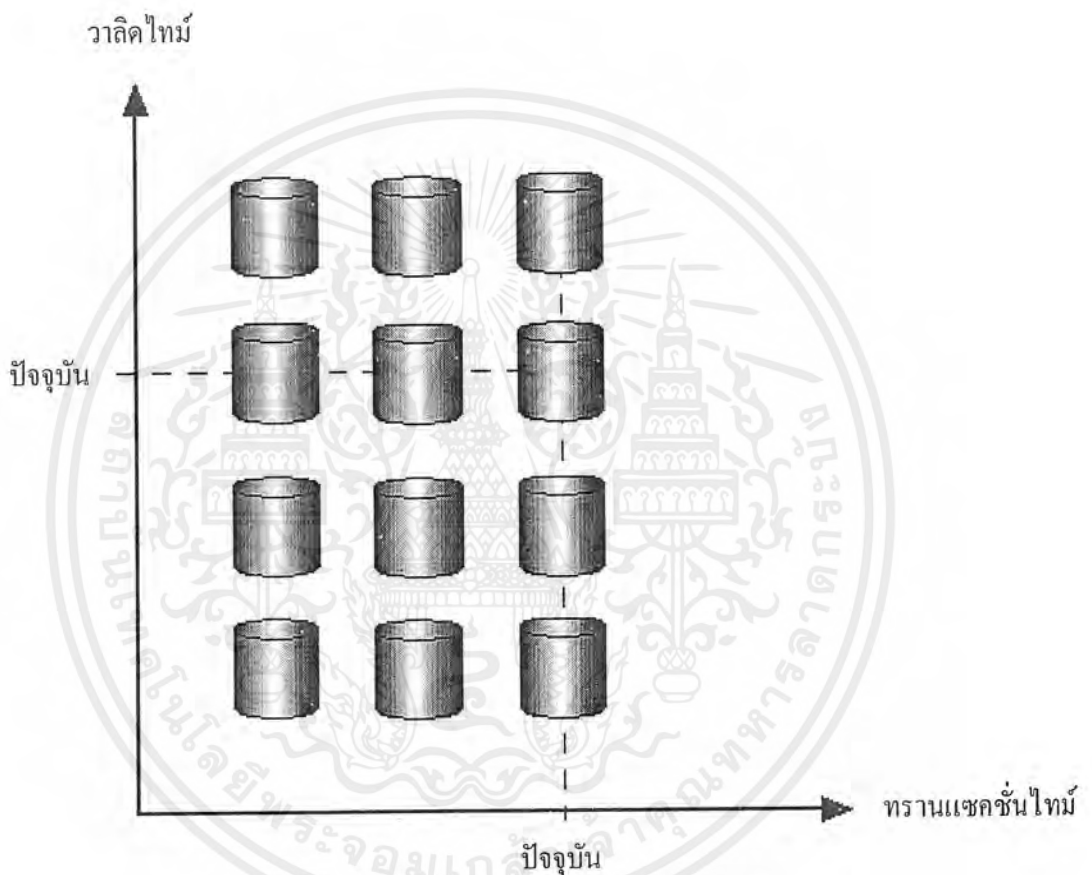
รูปที่ 2-1 ฐานข้อมูลสแน็ปช็อต

ในขณะที่ฐานข้อมูลไบเทมโปรอลจะเก็บประวัติของข้อมูลทั้งวาลิดไทม์ และทรานแซคชันไทม์ ตั้งแต่ประวัติที่ว่าข้อมูลถูกเก็บไว้ในฐานข้อมูลเมื่อใดหรือทรานแซคชันไทม์ ถูกจำกัดที่เวลาในอดีตจนถึงปัจจุบัน เพราะว่าทรานแซคชันไทม์ถูกจัดการโดยระบบโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางในระบบจัดการฐานข้อมูลเชิงสัมพันธ์แบบไบเทมโปรอล (Bitemporal Relational Database Management System) อาจจะเป็นได้ทั้งตารางสแน็ปช็อต (Snapshot Table) คือเก็บเฉพาะข้อมูลปัจจุบัน หรือตารางวาลิดไทม์ คือเก็บว่าเมื่อใดที่ข้อมูลเป็นจริง หรือตารางทรานแซกชันไทม์ (Transaction-Time Table) คือเก็บว่าข้อมูลถูกบันทึกในฐานข้อมูลเมื่อใด หรือตารางไบเทมโปรอล (Bitemporal Table) คือเก็บทั้งวาลิดไทม์และทรานแซกชันไทม์ โดยจะกำหนดชนิดของตารางที่ต้องการ ขณะที่สร้างตาราง (Create Table) หรือกำหนดชนิดของตารางจากตารางที่มีอยู่แล้วก็ได้ (Alter Table)

ฐานข้อมูลไบเทมโปรอลในแบบของวาลิดไทม์และทรานแซกชันไทม์แสดงดังรูปที่ 2-2



รูปที่ 2-2 ฐานข้อมูลไบเทมโปรอล

2.3 ระบบจัดการฐานข้อมูลเชิงเวลา

ระบบจัดการฐานข้อมูลเชิงเวลาควรจะสนับสนุน Temporal Data Definition Language, Temporal Data Manipulation Language, Temporal Query Language และ Temporal Constraint ซึ่งจะได้กล่าวต่อไป

แม้ว่าระบบจัดการฐานข้อมูลบางระบบจะสนับสนุนชนิดของข้อมูลที่เป็นวันและเวลา แต่ก็ไม่สามารถเรียกได้ว่าเป็นระบบจัดการฐานข้อมูลเชิงเวลา เนื่องจากถ้าต้องการข้อมูลอดีต ผู้ใช้จะต้องจัดการเองทั้งหมด โดยปราศจากการสนับสนุนใดๆจากระบบจัดการฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ปัญหาในการใช้ระบบจัดการฐานข้อมูลทั่วไปจัดการข้อมูลเชิงเวลา

สมมติให้ตารางพนักงานมี 3 คอลัมน์ คือ ชื่อ ผู้จัดการ และแผนก เราจะเก็บข้อมูลในอดีตโดยเพิ่มคอลัมน์ที่ 4 ซึ่งมีชนิดของข้อมูลเป็นช่วงเวลาเข้าไป

2.4.1 ผู้จัดการเป็น Foreign Key ของคอลัมน์ชื่อในตารางพนักงาน ซึ่งหมายความว่าค่าในคอลัมน์ผู้จัดการจะมีในคอลัมน์ชื่อในขณะนั้น สิ่งนี้ไม่สามารถแสดงโดย Foreign Key Constraint ของ SQL ได้

2.4.2 พิจารณาคำถามที่ว่า “พนักงานคนใดไม่ได้เป็นผู้จัดการ” คำถามนี้สามารถตอบได้โดยใช้ EXCEPT หรือ NOT EXISTS บนตารางเดิมที่มี 3 คอลัมน์ได้ ลองพิจารณาคำถามที่ว่า “พนักงานคนใดไม่เคยเป็นผู้จัดการ” จะไม่สามารถใช้ EXCEPT และ NOT EXISTS ได้ เพราะว่า EXCEPT และ NOT EXISTS ไม่ได้คิดเวลาด้วย

2.4.3 พิจารณาคำถามที่ว่า “หาจำนวนของพนักงานในแต่ละแผนก” นี่ก็เป็นคำถามที่สามารถตอบได้ง่ายๆ แต่ถ้าเปลี่ยนให้คำถามนี้มีเวลาเข้ามาเกี่ยวข้องด้วย เป็น “หาจำนวนของพนักงานในแต่ละแผนกทั้งหมดรวมในอดีตด้วย” จะสามารถตอบได้ยากมากถ้าปราศจากการสนับสนุนเชิงเวลาในภาษา

2.4.4 พิจารณาการเปลี่ยนแปลงที่ว่า “เปลี่ยนชื่อผู้จัดการของแผนกเครื่องมือในปี ค.ศ. 1994 เป็นอำนาจ” การแก้ไขนี้ทำได้ยากในภาษา SQL เพราะเป็นส่วนหนึ่งของหลายช่วงเวลา ต้องการเปลี่ยนแปลงโดยที่ข้อมูลที่ไม่ใช่ปี ค.ศ. 1994 ต้องยังคงเดิม

SQL เป็นภาษาที่สามารถตอบคำถามที่ไม่เกี่ยวข้องกับเวลาได้ดี แต่ภาษา SQL นั้นไม่ได้ช่วยในการเขียน Temporal Query, Temporal Modification และ Temporal Constraint

2.5 ตัวอย่างแอปพลิเคชันที่เกี่ยวข้องกับข้อมูลเชิงเวลา

2.5.1 การเงิน เช่น การวิเคราะห์แนวโน้มของหุ้นในแต่ละวันทำการ

2.5.2 การประกันภัย เช่น กรรมธรรม์ประกันภัยจะคุ้มครองในช่วงเวลาใด

2.5.3 ระบบการจอง เช่น ห้องพักในโรงแรมห้องใดถูกจองในช่วงเวลาใด

2.5.4 ระบบการจัดการข้อมูลทางการแพทย์ เช่น ประวัติของคนไข้

2.5.5 ระบบการสนับสนุนการตัดสินใจ (Decision Support Systems) เช่น การวางแผนในอนาคต

2.5.6 ระบบการผลิต เช่น การวิเคราะห์คุณภาพและปริมาณของสินค้าในแต่ละชั่วโมง

2.5.7 การวิจัยทางวิทยาศาสตร์ เช่น การวิเคราะห์คุณภาพของอากาศในแต่ละวัน

ในความเป็นจริงเกือบทุกแอปพลิเคชันจะมีความเกี่ยวข้องกับข้อมูลเชิงเวลา

บทที่ 3 ความซ้ำซ้อน (Duplicate)

3.1 ประเภทของความซ้ำซ้อน

เพื่อแสดงให้เห็นถึงความซ้ำซ้อนแบบต่างๆ ขอยกตัวอย่างตารางสถานะคนไข้ของโรงพยาบาลเด็กดังตารางที่ 3-1

ชื่อ	สถานะ	เวลาเริ่มต้น	เวลาสิ้นสุด
กฤษณ์	สาหัส	11/19/1997	11/20/1997
อำนาจ	สาหัส	11/19/1997	11/26/1997
สมชาย	สาหัส	11/19/1997	11/24/1997
สุรีย์	สาหัส	11/19/1997	11/25/1997
วีรชัย	สาหัส	11/19/1997	11/25/1997
รศมี	สาหัส	11/19/1997	11/27/1997
สมหญิง	ถูกเฉิน	11/19/1997	11/19/1997
สมหญิง	สาหัส	11/20/1997	11/25/1997
กฤษณ์	ดี	11/21/1997	01/02/1998
อำนาจ	ดี	11/27/1997	01/10/1998
อำนาจ	ดี	12/02/1997	12/31/9999
อำนาจ	ดี	12/02/1997	12/31/9999

ตารางที่ 3-1 ตารางสถานะคนไข้

ในตารางเชิงเวลา (Temporal Table) นี้ แต่ละแถวบอกสถานะของเด็ก โดยมีคอลัมน์เวลาเริ่มต้น บอกวันที่เด็กเริ่มมีสถานะนั้นซึ่งจะมีสถานะนี้ไปจนถึงวันที่ในคอลัมน์เวลาสิ้นสุด ค่า “12/31/9999” ในคอลัมน์เวลาสิ้นสุด หมายถึง เด็กจะมีสถานะนี้จนถึงปัจจุบัน

ความซ้ำซ้อนแบบต่างๆมีดังต่อไปนี้

- 3.1 ความซ้ำซ้อนแบบอนซีควเอนซ์ (Nonsequenced Duplicate) คือ ความซ้ำซ้อนที่ค่าของทุกคอลัมน์เหมือนกัน ยกตัวอย่างเช่น 2 แถวสุดท้ายของตารางที่ 3-1
- 3.2 ความซ้ำซ้อนแบบค่าเท่ากัน (Value-Equivalent Duplicate) คือ ความซ้ำซ้อนที่ค่าของทุกคอลัมน์ยกเว้นคอลัมน์เวลาเหมือนกัน ยกตัวอย่างเช่น 3 แถวสุดท้ายของตารางที่ 3-1
- 3.3 ความซ้ำซ้อนแบบปัจจุบัน (Current Duplicate) คือ ความซ้ำซ้อนที่ค่าของทุกคอลัมน์ยกเว้นคอลัมน์เวลาเหมือนกันในเวลาปัจจุบัน ยกตัวอย่างเช่น สมมติว่าวันนี้คือวันที่ 6 เดือนมกราคม ปี ค.ศ. 1998 3 แถวสุดท้ายของตารางที่ 3-1 จะเป็นความซ้ำซ้อนแบบปัจจุบัน เนื่องจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากค่าของทุกคอลัมน์ซ้ำกันในเวลาปัจจุบัน
สถานะคนไข้ดังตารางที่ 3-2

จะเห็นได้จากตารางสเน็ปช็อตของตาราง

ชื่อ	สถานะ
อำนาจ	ดี
อำนาจ	ดี
อำนาจ	ดี

ตารางที่ 3-2 ตารางสเน็ปช็อตของตารางสถานะคนไข้

3.4 ความซ้ำซ้อนแบบซีควเอนซ์ (Sequenced Duplicate) คือ ความซ้ำซ้อนที่มีค่าของทุกคอลัมน์ ยกเว้นคอลัมน์เวลาเหมือนกันอยู่ในช่วงระยะเวลาหนึ่งที่เหลื่อมซ้อนทับกัน ยกตัวอย่างเช่น 3 แถวสุดท้ายของตารางที่ 3-1 เนื่องจากทั้งสามแถวมีค่าเหมือนกันในช่วงเวลาที่เหลื่อมซ้อนกันคือ ตั้งแต่วันที่ 2 เดือนธันวาคม ปี ค.ศ. 1997 ถึงวันที่ 10 เดือนมกราคม ปี ค.ศ. 1998

ตารางที่ 3-3 แสดงให้เห็นถึงความสัมพันธ์ของความซ้ำซ้อนแบบต่างๆ ยกตัวอย่างเช่น ถ้าเป็นความซ้ำซ้อนแบบนอนซีควเอนซ์ก็จะเป็นความซ้ำซ้อนแบบซีควเอนซ์ด้วย แต่ถ้าเป็นความซ้ำซ้อนแบบซีควเอนซ์ก็ไม่จำเป็นต้องเป็นความสัมพันธ์แบบนอนซีควเอนซ์ เป็นต้น

ความซ้ำซ้อน	ซีควเอนซ์	ปัจจุบัน	ค่าเท่ากัน	นอนซีควเอนซ์
ซีควเอนซ์	✓		✓	
ปัจจุบัน	✓	✓	✓	
ค่าเท่ากัน			✓	
นอนซีควเอนซ์	✓		✓	✓

ตารางที่ 3-3 ความสัมพันธ์ของความซ้ำซ้อนแบบต่างๆ

3.2 การป้องกันความซ้ำซ้อนแบบค่าเท่ากัน

CREATE TABLE สถานะคนไข้ (

ชื่อ CHAR (15),

สถานะ CHAR (8),

เวลาเริ่มต้น DATE,

เวลาสิ้นสุด DATE,

UNIQUE (ชื่อ, สถานะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งก็คือ คนไข้ไม่ได้สถานะเดียวเท่านั้น ไม่สามารถมีหลายสถานะได้ในภายหลัง

3.3 การป้องกันความซ้ำซ้อนแบบนอนซีเควนซ์

```
CREATE TABLE สถานะคนไข้ (
    ...
    UNIQUE (ชื่อ, สถานะ, เวลาเริ่มต้น, เวลาสิ้นสุด)
)
```

ซึ่งก็คือ คนไข้ไม่สามารถมีหลายสถานะได้ในช่วงเวลาเดียวกัน แต่อย่างไรก็ตามการป้องกันเช่นนี้อาจจะไม่ได้ผล เนื่องจากช่วงเวลาเดียวกันอาจไม่ได้มีวันเริ่มต้นและวันสิ้นสุดเหมือนกัน

3.4 การป้องกันความซ้ำซ้อนแบบปัจจุบัน

```
CREATE TABLE สถานะคนไข้ (
    ...
    CHECK (NOT EXISTS (SELECT N1.ชื่อ
        FROM สถานะคนไข้ AS N1
        WHERE 1 < (SELECT COUNT (ชื่อ)
            FROM สถานะคนไข้ AS N2
            WHERE N1.ชื่อ = N2.ชื่อ AND N1.สถานะ = N2.สถานะ
                AND N1.เวลาเริ่มต้น <= CURRENT_DATE
                AND CURRENT_DATE < N1.เวลาสิ้นสุด
                AND N2.เวลาเริ่มต้น <= CURRENT_DATE
                AND CURRENT_DATE < N2.เวลาสิ้นสุด)))
)
```

ซึ่งก็คือ ในเวลาปัจจุบันคนไข้มีได้สถานะเดียว แต่อย่างไรก็ตามการป้องกันเช่นนี้อาจจะไม่ได้ผล เนื่องจากอาจไม่มีความซ้ำซ้อนในเวลาปัจจุบัน แต่อาจมีความซ้ำซ้อนเกิดขึ้นในอนาคต ดังนั้นถ้าทราบว่าแอปพลิเคชันจะไม่เก็บข้อมูลอนาคต ก็จะสามารถป้องกันได้โดยการเพิ่มคอลัมน์เวลาสิ้นสุดลงไปในคำสั่ง UNIQUE เนื่องจากข้อมูลปัจจุบันทุกตัวจะมีค่าเวลาสิ้นสุดเหมือนกัน คือ "12/31/9999"

```
CREATE TABLE สถานะคนไข้ (
    ...
    UNIQUE (ชื่อ, สถานะ, เวลาสิ้นสุด)
)
```

3.5 การป้องกันความซ้ำซ้อนแบบซีเควนซ์

จะคล้ายกับการป้องกันความซ้ำซ้อนแบบปัจจุบัน แต่จะแตกต่างกันตรงที่เงื่อนไขในการตรวจสอบเวลาเริ่มต้นกับเวลาสิ้นสุด เนื่องจากไม่จำเป็นต้องตรวจสอบเวลาปัจจุบัน

```
CREATE TABLE สถานะคนไข้ (
    ...
    CHECK (NOT EXISTS (SELECT N1.ชื่อ
        FROM สถานะคนไข้ AS N1
        WHERE 1 < (SELECT COUNT (ชื่อ)
            FROM สถานะคนไข้ AS N2
            WHERE N1.ชื่อ = N2.ชื่อ AND N1.สถานะ = N2.สถานะ
            AND N1.เวลาเริ่มต้น < N2.เวลาสิ้นสุด
            AND N2.เวลาเริ่มต้น < N1.เวลาสิ้นสุด)))
)
```

ซึ่งก็คือ ไม่มีเวลาใดที่คนไข้จะมีหลายสถานะ

บทที่ 4

การตอบคำถามตารางวลิตใหม่โดยใช้ภาษา SQL

เพื่อแสดงให้เห็นถึงการตอบคำถามตารางวลิตใหม่โดยใช้ภาษา SQL ขอยกตัวอย่างตารางวลิตดังตารางที่ 4-1

หมายเลขฟาร์ม	หมายเลขลือท	หมายเลขคอก	จำนวนวัว	เวลาเริ่มต้น	เวลาสิ้นสุด
1	317	1	17	02/07/1998	02/17/1998
1	219	1	43	02/25/1998	02/28/1998
1	219	1	20	03/01/1998	03/13/1998
1	219	2	23	03/01/1998	03/13/1998
1	219	2	43	03/14/1998	12/31/9999
1	374	1	14	02/20/1998	12/31/9999

ตารางที่ 4-1 ตารางวลิต

4.1 การตอบคำถามเชิงเวลาโดยใช้ภาษา SQL

4.1.1 การตอบคำถามแบบทั่วไป (Conventional Query)

พิจารณาคำถามที่ว่า “ในแต่ละคอกมีวัวกี่ตัวที่มาจากลือทที่ 219 ในฟาร์มที่ 1” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT หมายเลขคอก, จำนวนวัว
FROM วัว
WHERE หมายเลขฟาร์ม = 1 AND หมายเลขลือท = 219
```

4.1.2 การตอบคำถามแบบปัจจุบัน (Current Query)

พิจารณาคำถามที่ว่า “ปัจจุบันในแต่ละคอกมีวัวกี่ตัวที่มาจากลือทที่ 219 ในฟาร์มที่ 1” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT หมายเลขคอก, จำนวนวัว
FROM วัว
WHERE หมายเลขฟาร์ม = 1 AND หมายเลขลือท = 219
AND เวลาสิ้นสุด = '12/31/9999'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะได้คำตอบดังตารางที่ 4-2

หมายเลขคอก	จำนวนวัว
2	43

ตารางที่ 4-2 คำตอบของตัวอย่างการตอบคำถามแบบปัจจุบัน

4.1.3 การตอบคำถามแบบซีเควนซ์ (Sequenced Query)

พิจารณาคำถามที่ว่า “ตั้งแต่ในอดีตจนถึงปัจจุบันในแต่ละคอกมีวัวกี่ตัวที่มาจากลีสทที่ 219 ในฟาร์มที่ 1” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT หมายเลขคอก, จำนวนวัว, เวลาเริ่มต้น, เวลาสิ้นสุด
FROM วัว
WHERE หมายเลขฟาร์ม = 1 AND หมายเลขลีสท = 219
```

จะได้คำตอบดังตารางที่ 4-3

หมายเลขคอก	จำนวนวัว	เวลาเริ่มต้น	เวลาสิ้นสุด
1	43	02/25/1998	02/28/1998
1	20	03/01/1998	03/13/1998
2	23	03/01/1998	03/13/1998
2	43	03/14/1998	12/31/9999

ตารางที่ 4-3 คำตอบของตัวอย่างการตอบคำถามแบบซีเควนซ์

4.1.4 การตอบคำถามแบบนอนซีเควนซ์ (Nonsequenced Query)

พิจารณาคำถามที่ว่า “ในเวลาใดๆในแต่ละคอกมีวัวกี่ตัวที่มาจากลีสทที่ 219 ในฟาร์มที่ 1” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT หมายเลขคอก, จำนวนวัว
FROM วัว
WHERE หมายเลขฟาร์ม = 1 AND หมายเลขลีสท = 219
```

จะได้คำตอบดังตารางที่ 4-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลขคอก	จำนวนวัว
1	43
1	20
2	23
2	43

ตารางที่ 4-4 คำตอบของตัวอย่างการตอบคำถามแบบนอนซีแควนซ์

4.2 การตอบคำถามเชิงเวลาโดยใช้การ JOIN ของภาษา SQL

4.2.1 การตอบคำถามแบบทั่วไปโดยใช้การ JOIN (Conventional Join Query)

พิจารณาคำถามที่ว่า “มีวัวสีทใดที่อยู่ด้วยกันในแต่ละคอก” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT L1.หมายเลขลือท, L2.หมายเลขลือท, L1.หมายเลขคอก
FROM วัว AS L1, วัว AS L2
WHERE L1.หมายเลขลือท < L2.หมายเลขลือท
AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม
AND L1.หมายเลขคอก = L2.หมายเลขคอก
```

4.2.2 การตอบคำถามแบบปัจจุบันโดยใช้การ JOIN (Current Join Query)

พิจารณาคำถามที่ว่า “ปัจจุบันมีวัว Lot ใดที่อยู่ด้วยกันในแต่ละคอก” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT L1.หมายเลขลือท, L2.หมายเลขลือท, L1.หมายเลขคอก
FROM วัว AS L1, วัว AS L2
WHERE L1.หมายเลขลือท < L2.หมายเลขลือท
AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม
AND L1.หมายเลขคอก = L2.หมายเลขคอก
AND L1.เวลาสิ้นสุด = '12/31/9999'
AND L2.เวลาสิ้นสุด = '12/31/9999'
```

4.2.3 การตอบคำถามแบบนอนซีแควนซ์โดยใช้การ JOIN (Nonsequenced Join Query)

พิจารณาคำถามที่ว่า “มีวัวสีทใดที่อยู่ด้วยกันในแต่ละคอก อาจจะเป็นคนละเวลา” จะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT L1.หมายเลขล็อต, L2.หมายเลขล็อต, L1.หมายเลขคอก
FROM วิว AS L1, วิว AS L2
WHERE L1.หมายเลขล็อต < L2.หมายเลขล็อต
      AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม
      AND L1.หมายเลขคอก = L2.หมายเลขคอก

```

จะได้คำตอบดังตารางที่ 4-5

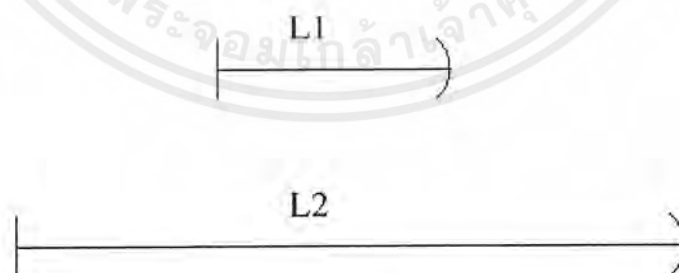
L1	L2	หมายเลขคอก
137	219	1
137	219	1
137	374	1
219	374	1
219	374	1

ตารางที่ 4-5 คำตอบของตัวอย่างการตอบคำถามแบบนอนซีเควนซ์โดยใช้การ JOIN

4.2.4 การตอบคำถามแบบซีเควนซ์โดยใช้การ JOIN (Sequenced Join Query)

พิจารณาคำถามที่ว่า “ตั้งแต่ในอดีตจนถึงปัจจุบันมีวัว Lot ใดบ้างที่อยู่ด้วยกันในแต่ละคอก” คำถามแบบนี้จะต้องแบ่งออกเป็น 4 กรณีดังนี้

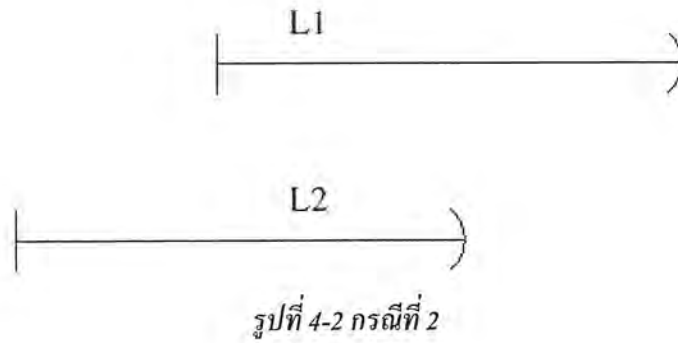
4.2.4.1 ช่วงเวลาของตาราง L1 อยู่ในช่วงเวลาของตาราง L2 แสดงดังรูปที่ 4-1



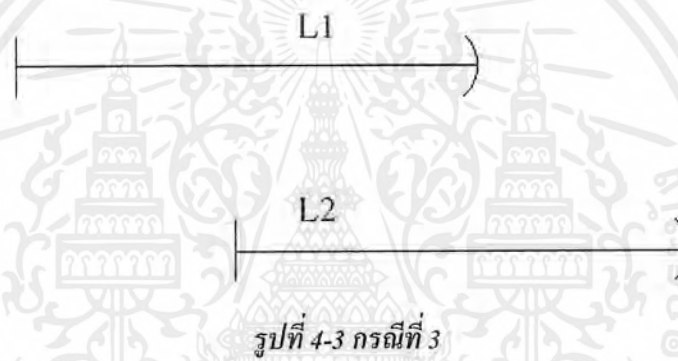
รูปที่ 4-1 กรณีที่ 1

4.2.4.2 เวลาเริ่มต้นของตาราง L1 มากกว่าเวลาเริ่มต้นของตาราง L2 แต่น้อยกว่าเวลาสิ้นสุดของตาราง L2 และเวลาสิ้นสุดของตาราง L1 มากกว่าเวลาสิ้นสุดของตาราง L2 แสดงดังรูปที่ 4-2

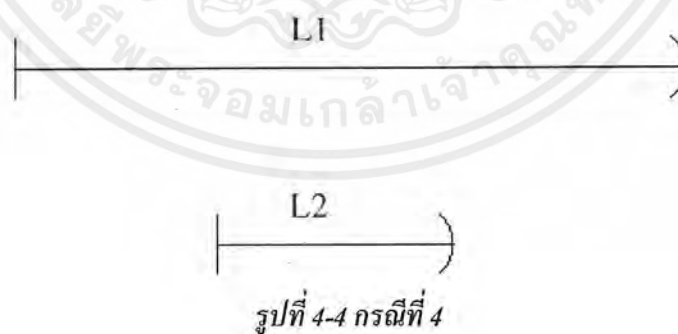
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- 4.2.4.3 เวลาเริ่มต้นของตาราง L2 มากกว่าเวลาเริ่มต้นของตาราง L1 แต่น้อยกว่าเวลาสิ้นสุดของตาราง L1 และเวลาสิ้นสุดของตาราง L2 มากกว่าเวลาสิ้นสุดของตาราง L1 แสดงดังรูปที่ 4-3



- 4.2.4.4 ช่วงเวลาของตาราง L2 อยู่ในช่วงเวลาของตาราง L1 แสดงดังรูปที่ 4-4



ดังนั้นจะสามารถตอบได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT L1.หมายเลขลือท, L2.หมายเลขลือท, L1.หมายเลขคอก, L1.เวลาเริ่มต้น, L1.เวลาสิ้นสุด
FROM วิว AS L1, วิว AS L2
WHERE L1.หมายเลขลือท < L2.หมายเลขลือท
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม

AND L1.หมายเลขคอก = L2.หมายเลขคอก

AND L2.เวลาเริ่มต้น <= L1.เวลาเริ่มต้น

AND L1.เวลาสิ้นสุด <= L2.เวลาสิ้นสุด

UNION

SELECT L1.หมายเลขลี้ท, L2.หมายเลขลี้ท, L1.หมายเลขคอก, L1.เวลาเริ่มต้น, L2.เวลาสิ้นสุด

FROM วิว AS L1, วิว AS L2

WHERE L1.หมายเลขลี้ท < L2.หมายเลขลี้ท

AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม

AND L1.หมายเลขคอก = L2.หมายเลขคอก

AND L1.เวลาเริ่มต้น > L2.เวลาเริ่มต้น

AND L2.เวลาสิ้นสุด < L1.เวลาสิ้นสุด

AND L1.เวลาเริ่มต้น < L2.เวลาสิ้นสุด

UNION

SELECT L1.หมายเลขลี้ท, L2.หมายเลขลี้ท, L1.หมายเลขคอก, L2.เวลาเริ่มต้น, L1.เวลาสิ้นสุด

FROM วิว AS L1, วิว AS L2

WHERE L1.หมายเลขลี้ท < L2.หมายเลขลี้ท

AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม

AND L1.หมายเลขคอก = L2.หมายเลขคอก

AND L2.เวลาเริ่มต้น > L1.เวลาเริ่มต้น

AND L1.เวลาสิ้นสุด < L2.เวลาสิ้นสุด

AND L2.เวลาเริ่มต้น < L1.เวลาสิ้นสุด

UNION

SELECT L1.หมายเลขลี้ท, L2.หมายเลขลี้ท, L1.หมายเลขคอก, L2.เวลาเริ่มต้น, L2.เวลาสิ้นสุด

FROM วิว AS L1, วิว AS L2

WHERE L1.หมายเลขลี้ท < L2.หมายเลขลี้ท

AND L1.หมายเลขฟาร์ม = L2.หมายเลขฟาร์ม

AND L1.หมายเลขคอก = L2.หมายเลขคอก

AND L2.เวลาเริ่มต้น >= L1.เวลาเริ่มต้น

AND L2.เวลาสิ้นสุด <= L1.เวลาสิ้นสุด

จะได้คำตอบดังตารางที่ 4-6

หมายเลขถือท	หมายเลขถือท	หมายเลขคอก	เวลาเริ่มต้น	เวลาสิ้นสุด
219	374	1	02/25/1998	02/28/1998
219	374	1	03/01/1998	03/13/1998

ตารางที่ 4-6 คำตอบของตัวอย่างการตอบคำถามแบบซีเควนซ์โดยใช้การ JOIN

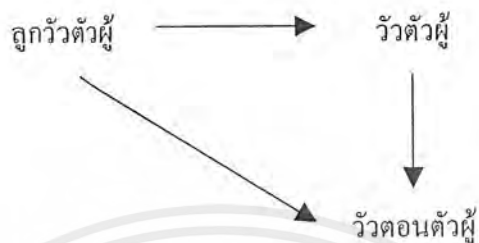


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การเปลี่ยนแปลงตารางวลิตใหม่โดยใช้ภาษา SQL

รูปที่ 5-1 ต่อไปนี้แสดงถึงการเปลี่ยนแปลงเพศของวัว



รูปที่ 5-1 การเปลี่ยนแปลงเพศของวัว

เพื่อแสดงให้เห็นถึงการเปลี่ยนแปลงตารางวลิตใหม่โดยใช้ภาษา SQL ขอยกตัวอย่างตารางเพศวัวดังตารางที่ 5-1

หมายเลขลือท	รหัสเพศ	เวลาเริ่มต้น	เวลาสิ้นสุด
101	C	01/01/1998	03/22/1998
101	S	03/23/1998	12/31/9999
234	C	02/17/1998	10/16/1998
234	S	10/17/1998	12/31/9999
799	C	03/12/1998	12/31/9999
800	B	01/01/1998	12/31/9999
801	H	01/01/1998	12/31/9999
802	W	01/01/1998	12/31/9999
803	N	01/01/1998	12/31/9999

ตารางที่ 5-1 ตารางเพศวัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยรหัสเพศ C หมายถึงลูกวัวตัวผู้ B หมายถึงวัวตัวผู้ S หมายถึงวัวตอนตัวผู้ H หมายถึงลูกวัวตัวเมีย W หมายถึงวัวตัวเมีย และ N หมายถึงวัวตอนตัวเมีย

5.1 การเปลี่ยนแปลงแบบปัจจุบัน (Current Modification)

5.1.1 การลบแบบปัจจุบัน (Current Deletion)

พิจารณาการเปลี่ยนแปลงที่ว่า “วัวลือท 234 ย้ายออกจากฟาร์ม” จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
UPDATE เพศวัว
SET เวลาสิ้นสุด = CURRENT_DATE
WHERE หมายเลขลือท = 234
AND เวลาสิ้นสุด >= CURRENT_DATE
AND เวลาเริ่มต้น < CURRENT_DATE
DELETE FROM เพศวัว
WHERE หมายเลขลือท = 234
AND เวลาเริ่มต้น > CURRENT_DATE
```

สมมติให้วันนี้เป็นวันที่ 29 เดือนกรกฎาคม ปี ค.ศ. 1998 ดังนั้นจะได้ผลดังตารางที่ 5-2

หมายเลขลือท	รหัสเพศ	เวลาเริ่มต้น	เวลาสิ้นสุด
101	C	01/01/1998	03/22/1998
101	S	03/23/1998	12/31/9999
234	C	02/17/1998	07/28/1998
799	C	03/12/1998	12/31/9999
800	B	01/01/1998	12/31/9999
801	H	01/01/1998	12/31/9999
802	W	01/01/1998	12/31/9999
803	N	01/01/1998	12/31/9999

ตารางที่ 5-2 ผลของตัวอย่างการลบแบบปัจจุบัน

5.1.2 การแก้ไขแบบปัจจุบัน (Current Update)

พิจารณาการเปลี่ยนแปลงที่ว่า “เปลี่ยนวัวลือท 799 เป็นวัวตอนตัวผู้” การเปลี่ยนแปลงแบบนี้จะต้องแบ่งออกเป็น 3 กรณี ดังรูปที่ 5-2 ดังต่อไปนี้

เอกสารอ้างอิงฉบับที่ 3 กรมปศุสัตว์ กระทรวงเกษตรและสหกรณ์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 1 :



ผล : ไม่เปลี่ยนแปลง

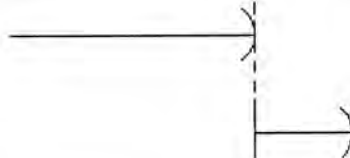
กรณีที่ 2 :



ผล : เปลี่ยนเวลาสิ้นสุด

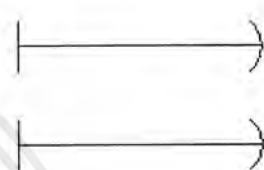
เป็นวันปัจจุบัน

และเพิ่มรหัสเพศใหม่เข้าไป



กรณีที่ 3 :

ผล : เปลี่ยนรหัสเพศใหม่



ปัจจุบัน

รูปที่ 5-2 กรณีของการแก้ไขแบบปัจจุบัน

ดังนั้นจะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```

INSERT INTO เพศวัว
SELECT หมายเลขลือท, 'S', CURRENT_DATE, เวลาสิ้นสุด
FROM เพศวัว
WHERE หมายเลขลือท = 799
      AND เวลาเริ่มต้น <= CURRENT_DATE
      AND เวลาสิ้นสุด > CURRENT_DATE

UPDATE เพศวัว
SET เวลาสิ้นสุด = CURRENT_DATE
WHERE หมายเลขลือท = 799
      AND รหัสเพศ <> 'S'
      AND เวลาเริ่มต้น < CURRENT_DATE
      AND เวลาสิ้นสุด > CURRENT_DATE

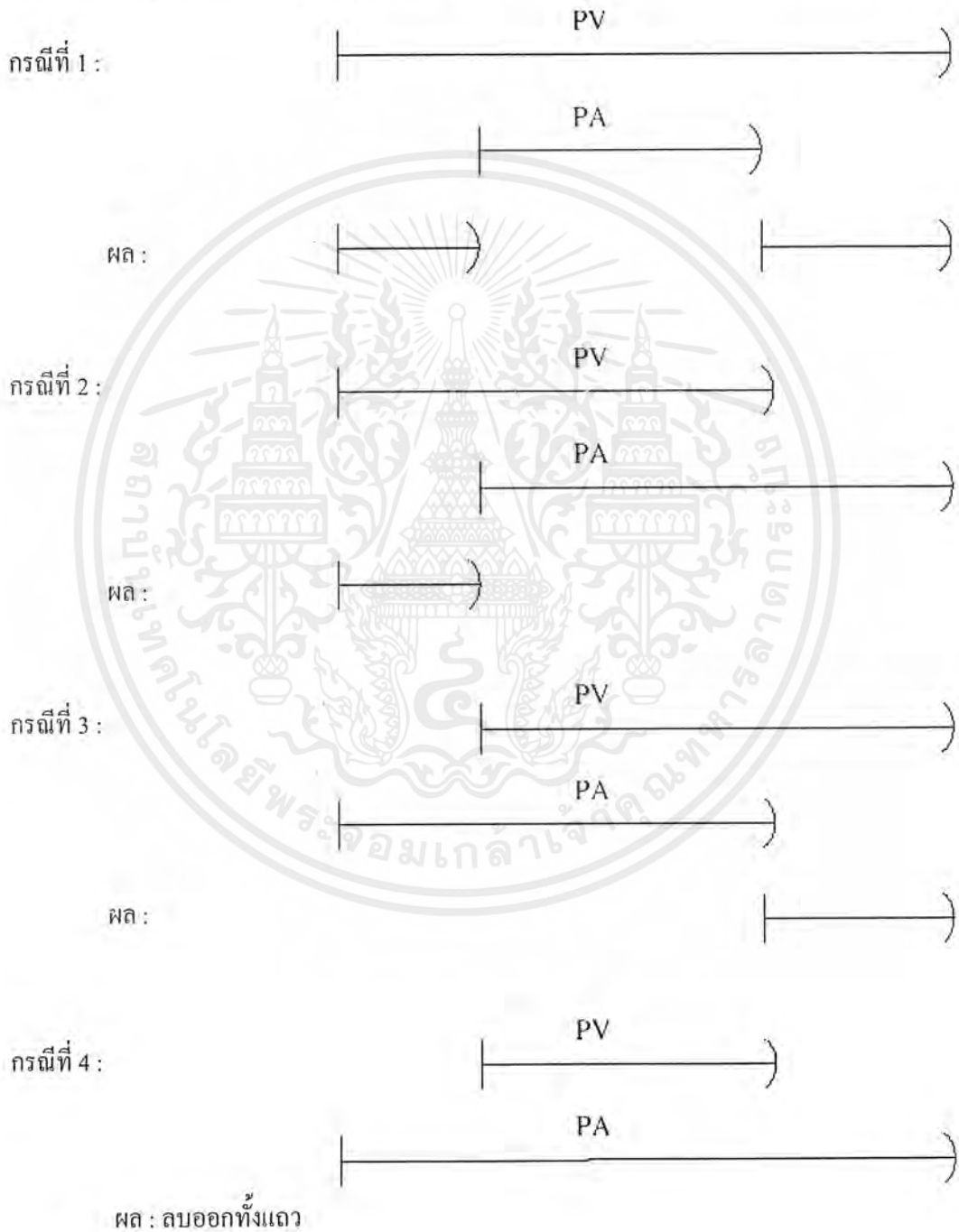
UPDATE เพศวัว
SET รหัสเพศ = 'S'
WHERE หมายเลขลือท = 799 AND เวลาเริ่มต้น >= CURRENT_DATE
  
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การเปลี่ยนแปลงแบบซีเควนซ์ (Sequenced Modification)

5.2.1 การลบแบบซีเควนซ์ (Sequenced Deletion)

พิจารณาการเปลี่ยนแปลงที่ว่า “วีล๊อต 234 ย้ายออกจากฟาร์มใน 3 สัปดาห์แรกของเดือนตุลาคม ซึ่งในช่วงเวลานั้นได้มีการวางแผนที่จะเปลี่ยนวีล๊อต 234 นี้เป็นวัวตอนตัวผู้” การเปลี่ยนแปลงแบบนี้จะต้องแบ่งออกเป็น 4 กรณี แสดงดังรูปที่ 5-3 ดังต่อไปนี้



รูปที่ 5-3 กรณีของการลบแบบซีเควนซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย PV คือ ช่วงเวลาของข้อมูลเดิม (Period Of Validity) และ PA คือ ช่วงเวลาที่ต้องการเปลี่ยนแปลง (Period Of Applicability) ดังนั้นจะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```

INSERT INTO เพศวิว
SELECT หมายเลขล๊อต, รหัสเพศ, '10/22/1998', เวลาสิ้นสุด
FROM เพศวิว
WHERE หมายเลขล๊อต = 234
      AND เวลาเริ่มต้น <= '10/01/1998'
      AND เวลาสิ้นสุด > '10/22/1998'

UPDATE เพศวิว
SET เวลาสิ้นสุด = '10/01/1998'
WHERE หมายเลขล๊อต = 234
      AND เวลาเริ่มต้น < '10/01/1998'
      AND เวลาสิ้นสุด >= '10/01/1998'

UPDATE เพศวิว
SET เวลาเริ่มต้น = '10/22/1998'
WHERE หมายเลขล๊อต = 234
      AND เวลาเริ่มต้น < '10/22/1998'
      AND เวลาสิ้นสุด >= '10/22/1998'

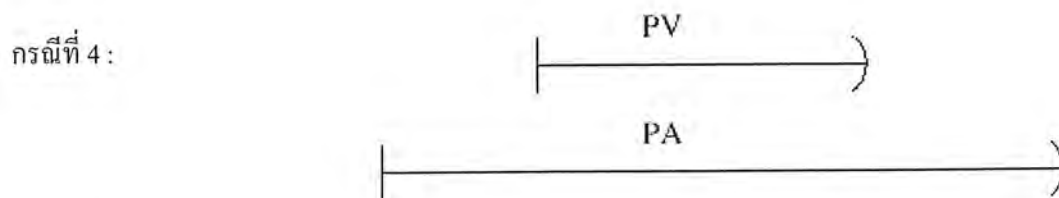
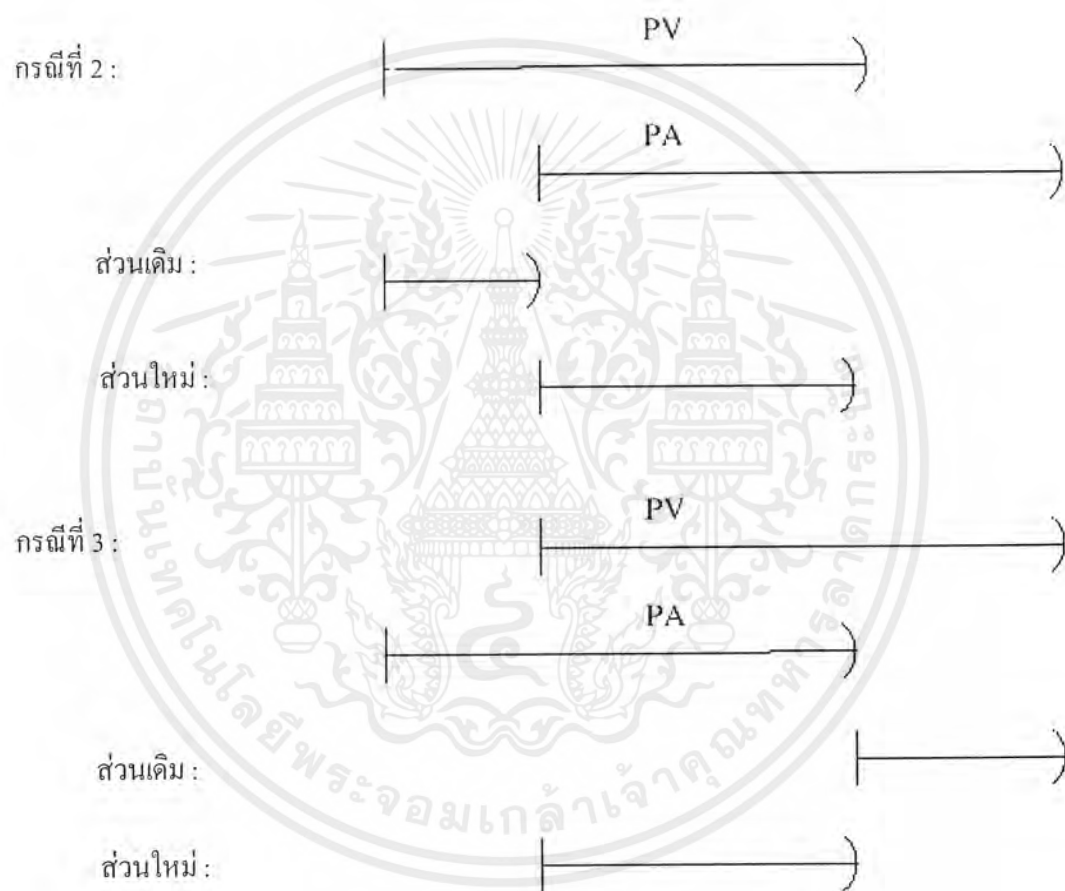
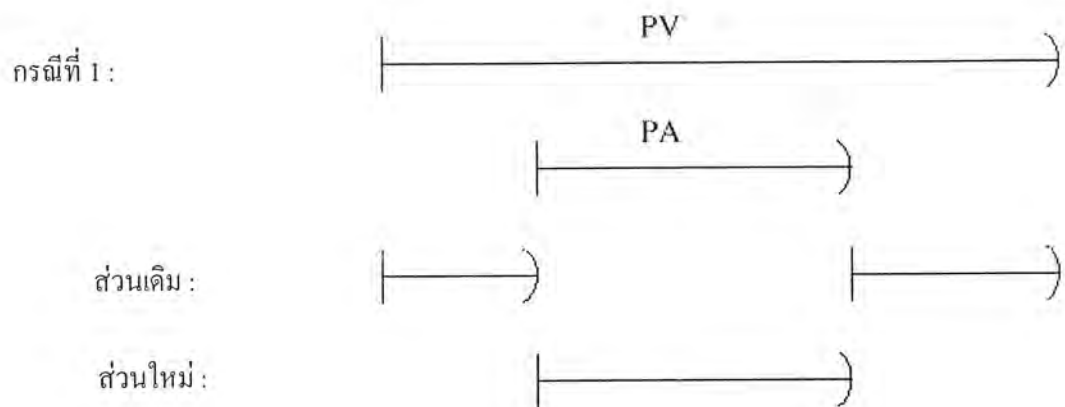
DELETE FROM เพศวิว
WHERE หมายเลขล๊อต = 234
      AND เวลาเริ่มต้น >= '10/01/1998'
      AND เวลาสิ้นสุด <= '10/22/1998'

```

5.2.2 การแก้ไขแบบซีควเอนซ์ (Sequenced Update)

พิจารณาการเปลี่ยนแปลงที่ว่า “เปลี่ยนวีวล๊อต 799 เป็นวีวลอนตัวผู้เฉพาะในเดือนมีนาคม” การเปลี่ยนแปลงแบบนี้จะต้องแบ่งออกเป็น 4 กรณี แสดงดังรูปที่ 5-4 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ผล : แก้ไขทั้งแถว

รูปที่ 5-4 กรณีของการแก้ไขแบบซีเควนซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นจะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
INSERT INTO เพศวิว
SELECT หมายเลขลือท, รหัสเพศ, เวลาเริ่มต้น, '03/01/1998'
FROM เพศวิว
WHERE หมายเลขลือท = 799
      AND เวลาเริ่มต้น < '03/01/1998'
      AND เวลาสิ้นสุด > '03/01/1998'
```

```
INSERT INTO เพศวิว
SELECT หมายเลขลือท, รหัสเพศ, '04/01/1998', เวลาสิ้นสุด
FROM เพศวิว
WHERE หมายเลขลือท = 799
      AND เวลาเริ่มต้น < '04/01/1998'
      AND เวลาสิ้นสุด > '04/01/1998'
```

```
UPDATE เพศวิว
SET รหัสเพศ = 'S'
WHERE หมายเลขลือท = 799
      AND เวลาเริ่มต้น < '04/01/1998'
      AND เวลาสิ้นสุด > '03/01/1998'
```

```
UPDATE เพศวิว
SET เวลาเริ่มต้น = '03/01/1998'
WHERE หมายเลขลือท = 799
      AND เวลาเริ่มต้น < '03/01/1998'
      AND เวลาสิ้นสุด > '03/01/1998'
```

```
UPDATE เพศวิว
SET เวลาสิ้นสุด = '04/01/1998'
WHERE หมายเลขลือท = 799
      AND เวลาเริ่มต้น < '04/01/1998'
      AND เวลาสิ้นสุด > '04/01/1998'
```

5.3 การเปลี่ยนแปลงแบบนอนซีควเอนซ์ (Nonsequenced Modification)

พิจารณาการเปลี่ยนแปลงที่ว่า “ลบข้อมูลของวิวลือท 234 ที่มีช่วงเวลามากกว่า 3 เดือน” จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DELETE FROM เพศวิ

WHERE หมายเลขลือท = 234

AND (เวลาสิ้นสุด - เวลาเริ่มต้น MONTH) > INTERVAL '3' MONTH



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การจัดการตารางฐานข้อมูลแบบใหม่โดยใช้ภาษา SQL

ในบทที่ผ่านมาได้กล่าวถึงตารางวัตถุใหม่ซึ่งเก็บช่วงเวลาที่มีข้อมูลเป็นจริง ในบทนี้จะกล่าวถึงตารางฐานข้อมูลแบบใหม่ซึ่งเก็บช่วงเวลาที่มีข้อมูลถูกเก็บลงในฐานข้อมูล

เพื่อแสดงให้เห็นถึงการจัดการตารางฐานข้อมูลแบบใหม่โดยใช้ภาษา SQL ขอยกตัวอย่างตารางดวงดาว1 ดังตารางที่ 6-1 และตารางดวงดาว2 ดังตารางที่ 6-2

ชั่วโมง	นาที	วินาที	มุมเอียง	นาฬิกาเอียง	ผู้ค้นพบ	ความสว่าง
00	00	08	75	30	'A 1248'	10.5
05	57	40	00	02	'BU 1190'	6.5
04	13	20	50	32	'CHR 15'	15.5
01	23	70	-09	55	'HJ 3433'	10.5

ตารางที่ 6-1 ตารางดวงดาว1

ชั่วโมง	นาที	วินาที	มุมเอียง	นาฬิกาเอียง	ผู้ค้นพบ	ความสว่าง	เวลาเริ่มต้น	เวลาสิ้นสุด
00	00	00	75	30	'A 1248'	12.0	03/12/1989	11/14/1992
00	00	09	75	30	'A 1248'	12.0	11/15/1992	05/17/1994
00	00	09	75	30	'A 1248'	10.5	05/18/1994	07/22/1995
00	00	08	75	30	'A 1248'	10.5	07/23/1995	12/31/9999
05	57	40	00	02	'BU 1190'	6.5	11/08/1988	12/31/9999
04	13	20	50	32	'CHR 15'	15.5	02/09/1990	12/31/9999
01	23	70	-09	55	'HJ 3433'	10.5	03/25/1991	12/31/9999
02	33	10	-09	25	'LDS 3402'	10.6	12/19/1993	07/08/1996

ตารางที่ 6-2 ตารางดวงดาว2

ตารางดวงดาว2 เป็นตารางฐานข้อมูลแบบใหม่ของตารางดวงดาว1 โดย 5 คอลัมน์แรกของตารางเป็นตำแหน่งของดวงดาว คอลัมน์ผู้ค้นพบประกอบด้วยตัวอักษรตั้งแต่ 1 ถึง 3 ตัวอักษรตามด้วยหมายเลขผู้ค้นพบ คอลัมน์ผู้ค้นพบนี้เป็น Primary Key ของตารางนี้ด้วย ส่วนคอลัมน์เวลาเริ่มต้นจะเก็บเวลาที่ข้อมูลถูกใส่เข้าไปในตารางหรือเวลาที่ข้อมูลถูกแก้ไข (ข้อมูลใหม่จะถูกเก็บไว้ที่นี่) และคอลัมน์เวลาสิ้นสุดเก็บเวลาที่ข้อมูลถูกลบออกจากตารางหรือถูกแก้ไข (ข้อมูลเก่าจะถูกเก็บไว้ที่นี่)

จะเห็นว่าข้อมูลของดวงดาว LDS 3402 ถูกใส่เข้าไปในวันที่ 19 เดือนธันวาคม ปี ค.ศ. 1993 แล้วถูกลบออกในวันที่ 8 เดือนกรกฎาคม ปี ค.ศ. 1996 เนื่องจากพบข้อผิดพลาดของข้อมูล และข้อมูลของดวง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดาว A 1248 ถูกใส่เข้าไปครั้งแรกเมื่อวันที่ 12 เดือนมีนาคม ปี ค.ศ. 1989 และได้ทำการแก้ไขข้อมูลตำแหน่งวินาทีของดวงดาวนี้ในวันที่ 15 เดือนพฤศจิกายน ปี ค.ศ. 1992 แก้ไขข้อมูลความสว่างในวันที่ 18 เดือนพฤษภาคม ปี ค.ศ. 1994 และแก้ไขข้อมูลตำแหน่งวินาทีในวันที่ 23 เดือนกรกฎาคม ปี ค.ศ. 1995 สังเกตว่าการเปลี่ยนแปลงเหล่านี้ไม่ได้หมายความว่าดวงดาวเกิดการเปลี่ยนแปลง แต่หมายความว่าเกิดความผิดพลาดขึ้นในการวัดครั้งก่อน จึงทำการแก้ไขข้อมูลให้ถูกต้อง ดังนั้นแถวที่มีคอลัมน์เวลาสิ้นสุดเป็นอดีตจะเป็นข้อมูลที่ไมถูกต้อง

สังเกตว่าข้อมูลในตารางทรานแซกชันใหม่จะเป็นข้อมูลตั้งแต่อดีตจนถึงปัจจุบันเท่านั้น เนื่องจากไม่มีทางทราบได้เลยว่าในอนาคตข้อมูลใดจะเกิดข้อผิดพลาด

6.1 การรักษาตารางทรานแซกชันใหม่

การรักษาตารางทรานแซกชันใหม่สามารถทำได้โดยอัตโนมัติโดยใช้ Trigger ดังต่อไปนี้

```
CREATE TRIGGER INSERT_WDS
AFTER INSERT ON ดวงดาว1
REFERENCING NEW AS N
FOR EACH ROW
INSERT INTO ดวงดาว2 (ชั่วโมง, นาที, วินาที, มุมเอียง, นาฬิกาเอียง,
ผู้ค้นพบ, ความสว่าง, เวลาเริ่มต้น, เวลาสิ้นสุด)
VALUES (N.ชั่วโมง, N.นาที, N.วินาที, N.มุมเอียง, N.นาฬิกาเอียง,
N.ผู้ค้นพบ, N.ความสว่าง, CURRENT_DATE, '12/31/9999')
```

```
CREATE TRIGGER DELETE_WDS
AFTER DELETE ON ดวงดาว1
REFERENCING OLD AS O
FOR EACH ROW
UPDATE ดวงดาว2
SET STOP_TIME = CURRENT_DATE
WHERE ดวงดาว2.ผู้ค้นพบ = O.ผู้ค้นพบ
AND ดวงดาว2.เวลาสิ้นสุด = '12/31/9999'
```

```
CREATE TRIGGER UPDATE_P
AFTER UPDATE ON ดวงดาว1
REFERENCING OLD AS O NEW AS N
FOR EACH ROW
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BEGIN ATOMIC
```

```
    UPDATE ดวงดาว2
    SET เวลาสิ้นสุด = CURRENT_DATE
    WHERE ดวงดาว2.ผู้ค้นพบ = O.ผู้ค้นพบ
    AND ดวงดาว2.เวลาสิ้นสุด = '12/31/9999';
    INSERT INTO ดวงดาว2 (ชั่วโมง, นาที, วินาที, มุมเอียง, นาทีเอียง,
    ผู้ค้นพบ, ความสว่าง, เวลาเริ่มต้น, เวลาสิ้นสุด)
    VALUES (N.ชั่วโมง, N.นาที, N.วินาที, N.มุมเอียง, N.นาทีเอียง,
    N.ผู้ค้นพบ, N.ความสว่าง, CURRENT_DATE, '12/31/9999')
```

```
END
```

สังเกตว่าตารางจะมีขนาดเพิ่มขึ้นเพียงอย่างเดียวเท่านั้น เนื่องจากจะไม่มีข้อมูลใดถูกลบออกไปจริงๆ Trigger INSERT จะเพิ่มข้อมูลเข้าไปในตาราง Trigger DELETE จะเปลี่ยนค่าของคอลัมน์เวลาสิ้นสุดเป็นวันที่ปัจจุบัน ส่วน Trigger UPDATE จะทำทั้งสองอย่าง

6.2 การตอบคำถามตารางทรานแซกชันใหม่

6.2.1 การตอบคำถามแบบปัจจุบัน (Current Query)

ถ้าต้องการทราบข้อมูลปัจจุบันของตารางดวงดาว จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT ชั่วโมง, นาที, วินาที, มุมเอียง, นาทีเอียง, ผู้ค้นพบ, ความสว่าง
FROM ดวงดาว2
WHERE เวลาสิ้นสุด = '12/31/9999'
```

ถ้าต้องการทราบข้อมูลของตารางดวงดาวในวันที่ 1 เดือนเมษายน ปี ค.ศ. 1994 จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
CREATE VIEW WDS_April_1 AS
SELECT ชั่วโมง, นาที, วินาที, มุมเอียง, นาทีเอียง, ผู้ค้นพบ, ความสว่าง
FROM ดวงดาว2
WHERE เวลาเริ่มต้น <= '04/01/1994' AND '04/01/1994' < เวลาสิ้นสุด
```

จะได้คำตอบดังตารางที่ 6-3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั่วโมง	นาทื	วินาที	มุมเอียง	นาทืเอียง	ผู้ค้นพบ	ความสว่าง
00	00	09	75	30	'A 1248'	12.0
05	57	40	00	02	'BU 1190'	6.5
04	13	20	50	32	'CHR 15'	15.5
01	23	70	-09	55	'HJ 3433'	10.5
02	33	10	-09	25	'LDS 3402'	10.6

ตารางที่ 6-3 ตารางดวงดาวในวันที่ 1 เดือนเมษายน ปี ค.ศ. 1994

สังเกตว่ามีข้อมูลของดวงดาว LDS 3402 อยู่ด้วย เนื่องจากในวันที่ 1 เดือนเมษายน ปี ค.ศ. 1994 ยังไม่พบข้อผิดพลาดของข้อมูล และข้อมูลของดวงดาว A 1248 ก็มีตำแหน่งและความสว่างที่ไม่ถูกต้อง

6.2.2 การตอบคำถามแบบซีเควนซ์ (Sequenced Query)

พิจารณาคำถามที่ว่า “เมื่อใดที่มีการบันทึกไว้ว่าดวงดาว A 1248 มีความสว่างไม่เท่ากับ 10.5” จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT ความสว่าง, เวลาเริ่มต้น, เวลาสิ้นสุด
FROM ดวงดาว2
WHERE ผู้ค้นพบ = 'A 1248'
AND ความสว่าง <> 10.5
```

จะได้คำตอบดังตารางที่ 6-4

ความสว่าง	เวลาเริ่มต้น	เวลาสิ้นสุด
12.0	03/12/1989	11/14/1992
12.0	11/15/1992	05/17/1994

ตารางที่ 6-4 คำตอบของตัวอย่างการตอบคำถามแบบซีเควนซ์

พิจารณาคำถามที่ว่า “เมื่อใดที่มีการบันทึกไว้ว่ามีดวงดาวที่มีความสว่างเท่ากับดวงดาว A 1248” จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT W1.ผู้ค้นพบ, GREATEST (W1.เวลาเริ่มต้น, W2.เวลาเริ่มต้น),
LEAST (W1.เวลาสิ้นสุด, W2.เวลาสิ้นสุด)
FROM ดวงดาว2 W1, ดวงดาว2 W2
```

```
WHERE W1.ผู้ค้นพบ = 'A 1248'
```

เอกสารนี้จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND W2.ผู้ค้นพบ <> W1.ผู้ค้นพบ

AND W1.ความสว่าง = W2.ความสว่าง

AND GREATEST (W1.เวลาเริ่มต้น, W2.เวลาเริ่มต้น) < LEAST (W1.เวลาสิ้นสุด, W2.เวลาสิ้นสุด)

จะได้คำตอบดังตารางที่ 6-5

ผู้ค้นพบ	เวลาเริ่มต้น	เวลาสิ้นสุด
'HJ 3433'	05/18/1994	07/22/1995
'HJ 3433'	07/23/1995	12/31/9999

ตารางที่ 6-5 คำตอบของตัวอย่างการตอบคำถามแบบซีเควนซ์

6.2.3 การตอบคำถามแบบนอนซีเควนซ์ (Nonsequenced Query)

พิจารณาคำถามที่ว่า “เมื่อใดที่มีการบันทึกไว้ว่ามีการเปลี่ยนแปลงตำแหน่งวินาทีของดวงดาว” จะสามารถทำได้โดยใช้ SQL ดังต่อไปนี้

```
SELECT W1.ผู้ค้นพบ, W1.วินาที AS Old_Value, W2.วินาที AS New_Value,
W1.เวลาสิ้นสุด AS When_Changed
FROM ดวงดาว2 AS W1, ดวงดาว2 AS W2
WHERE W1.ผู้ค้นพบ = W2.ผู้ค้นพบ
AND W1.เวลาสิ้นสุด = W2.เวลาเริ่มต้น
AND W1.วินาที <> W2.วินาที
```

จะได้คำตอบดังตารางที่ 6-6

ผู้ค้นพบ	Old_Value	New_Value	When_Changed
'A 1248'	00	09	11/15/1992
'A 1248'	09	08	07/23/1995

ตารางที่ 6-6 คำตอบของตัวอย่างการตอบคำถามแบบนอนซีเควนซ์

6.3 การเปลี่ยนแปลงตารางทรานแซคชันใหม่

ในขณะที่การตอบคำถามตารางทรานแซคชันใหม่มีทั้งแบบปัจจุบัน แบบซีเควนซ์ และแบบนอนซีเควนซ์ แต่การเปลี่ยนแปลงตารางทรานแซคชันใหม่นั้นกลับไม่ได้มีทั้ง 3 แบบ การเปลี่ยนแปลงที่เกิดขึ้นกับตารางทรานแซคชันใหม่มีเพียงแบบเดียวคือการเปลี่ยนแปลงแบบปัจจุบัน ซึ่งก็คือการเปลี่ยนแปลงสถานะ

ที่เก็บอยู่ในปัจจุบัน เอกสารนี้เขียนขึ้นไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงแบบซีควนซ์และแบบนอนซีควนซ์สามารถเปลี่ยนแปลงสถานะอดีตของตาราง วาติดใหม่ได้ แต่การทำเช่นนี้ไม่สามารถทำได้ในตารางทรานแซกชันใหม่ เนื่องจากเป็นการละเมิดกฎ (Semantics) ของตาราง กล่าวคือเราไม่สามารถใส่ข้อมูลเข้าไปในตารางควงควา2 โดยใส่ค่าของคอดีมันน์ เวลาเริ่มต้นเป็นวันที่ในอดีตได้ หรือก็คือเราไม่สามารถเปลี่ยนแปลงอดีตได้ ด้วยเหตุนี้การเปลี่ยนแปลง ตารางทรานแซกชันใหม่จึงควรรใช้ Trigger เท่านั้นในการเปลี่ยนแปลง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

ภาษา SQL เชิงเวลา (ภาษา TSQL2)

บทนี้จะกล่าวถึงภาษา TSQL2 ซึ่งเป็นภาษา SQL ที่สนับสนุนข้อมูลเชิงเวลา ซึ่งภาษา SQL เชิงเวลาที่ใช้อ้างอิงนี้จะใช้ภาษา TSQL2 ซึ่งได้มีการตั้งคณะกรรมการขึ้นมาพัฒนาเพื่อให้เป็นมาตรฐานสำหรับภาษา SQL เชิงเวลา โดยส่วนแรกจะกล่าวถึงลักษณะเฉพาะ (Feature) ที่ต้องการในภาษา TSQL2 และส่วนต่อมาเป็นโครงสร้างประโยค (Syntax) ของภาษา TSQL2

7.1 ลักษณะเฉพาะที่ต้องการในภาษา SQL เชิงเวลา

ส่วนนี้จะกล่าวถึงลักษณะเฉพาะที่ต้องการที่ควรจะถูกสนับสนุนโดยภาษา TSQL2 ลักษณะเฉพาะเหล่านี้จะใช้เป็นเกณฑ์ในการออกแบบภาษา TSQL2

ขั้นแรกจะพิจารณาถึงลักษณะของแบบจำลองข้อมูล

- ภาษา TSQL2 ควรสนับสนุนวาลิดไทม์เพียงมิติเดียวเท่านั้น
- ควรใช้การเก็บช่วงเวลาไว้กับข้อมูล
- การสนับสนุนวาลิดไทม์ควรสนับสนุนทั้งอดีตและอนาคต
- ค่าของเวลาไม่ควรถูกจำกัดเป็นช่วงหรือเป็นค่าแน่นอน

ขั้นต่อมาจะพิจารณาถึงคุณสมบัติของภาษา

- ภาษา TSQL2 ควรจะเข้ากันได้กับภาษา SQL
- ภาษา TSQL2 ควรจะมีการจัดโครงสร้างใหม่ของตารางบนเซตใดๆของแอททริบิวต์
- การสนับสนุนเชิงเวลาควรจะสามารถเลือกได้
- การสนับสนุนเวลาที่กำหนดโดยผู้ใช้ควรจะรวมทั้งข้อมูล และช่วงเวลา
- Aggregate ที่มีอยู่ควรจะสามารถใช้ได้ทั้งในภาษา TSQL2
- ควรจะมีการสนับสนุนภาษาและปฏิทินหลายแบบในการเก็บช่วงเวลาไว้กับข้อมูล
- ควรจะสามารถสร้างตารางเชิงเวลาจากรายการเชิงเวลาและจากรายการทั่วไป

7.2 โครงสร้างประโยคของภาษา SQL เชิงเวลา

<delimiter token> ::=

| <period string>

<reserved word> ::=

CALENDRIC | CONTAINS | CREDIBILITY
 DISTRIBUTION
 EVENT
 GENERAL
 INAPPLICABLE | INDETERMINATE
 MEETS
 NEW | NOBIND | NONSTANDARD
 PERIOD | PLAUSIBILITY | PRECEDES | PREVIOUS | PROPERTIES
 RISING
 SCALE | SNAPSHOT | STATE | SURROGATE
 VACUUM
 VALID
 WEIGHTED

<general literal> ::=

| <period literal>

<date literal> ::=

DATE <date string> <calendric-property specification>

<time literal> ::=

TIME <time string> <calendric-property specification>

<timestamp literal> ::=

TIMESTAMP <timestamp string> [<timestamp precision>] <calendric-property specification>

<interval literal> ::=

INTERVAL [<sign>] <interval string> <interval qualifier> <calendric-property specification>

<date string> ::=

<datetime string>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<datetime string> ::=

<character string literal>
 <determinate datetime string>
 <now-relative datetime string>
 <indeterminate now-relative datetime string>
 <now-relative with indeterminate datetime string>

<time string> ::=

<datetime string>

<timestamp string> ::=

<datetime string>

<interval string> ::=

<character string literal>
 <determinate interval string>
 <indeterminate interval string>
 <now-relative interval string>

<period literal> ::=

PERIOD <period string> [<period precision>] <calendric-property specification>

<period string> ::=

<character string literal>

<format string> ::=

<quote> [<character representation> | field specification]... <quote>

<field specification> ::=

<less than operator> <field identifier>
 [<comma> <translation table name> [<comma> <field formatting specification>...]]
 <greater than operator>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<field identifier> ::=

<identifier>

<translation table name> ::=

<identifier>

<field formatting specification> ::=

W <unsigned integer>

L

R

Z

B

S

<calendric system name> ::=

<identifier>

<property table name> ::=

<table name>

<time granularity> ::=

<identifier>

<data type> ::=

| <period type>

| <surrogate type>

<period type> ::=

[<indeterminate data type>] PERIOD [<period precision>] [WITH TIME ZONE]

<period precision> ::=

<period qualifier>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<indeterminate data type> ::=

[NONSTANDARD] [GENERAL] INDETERMINATE

<surrogate type> ::=

SURROGATE

<datetime type> ::=

[<indeterminate data type>] DATE

[<indeterminate data type>] TIME [<time precision>] [WITH TIME ZONE]

[<indeterminate data type>] TIMESTAMP [<timestamp precision>] [WITH TIME ZONE]

<time precision> ::=

<left paren> <time fractional seconds precision> <right paren>

<timestamp precision> ::=

<timestamp qualifier>

<interval type> ::=

[<indeterminate data type>] INTERVAL [<interval qualifier>]

<parameter specification> ::=

<parameter name> [<indicator parameter>] [<calendric-property specification>]

<variable specification> ::=

<embedded variable name> [<indicator variable>] [<calendric-property specification>]

<table reference> ::=

<table source> [[AS] <corr> { <corr> } ...]

<derived table> [AS] <corr> { <corr> } ...

<joined table>

<corr> ::=

<correlation> [WITH CREDIBILITY <integer>]

| <joined table>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<table source> ::=

<table name> <correlation modifier>
| <correlation name> <correlation modifier>

<correlation> ::=

<correlation name> [<left paren> <derived column list> <right paren>]

<correlation modifier> ::=

[<left paren> <coalescing columns> <right paren>]
[<left paren> <partitioning unit> <right paren>]

<coalescing columns> ::=

<column name> [{<comma> <column name> }...]
| <asterisk>

<partitioning unit> ::=

INSTANT
| PERIOD

<general function type> ::=

<set function type> <left paren> [<set quantifier>]
[WEIGHTED]
<value expression> <right paren>

<set function type> ::=

RISING

<datetime value function> ::=

BEGIN <left paren> <period value expression> <right paren>
END <left paren> <period value expression> <right paren>
FIRST <left paren> <datetime value expression> <comma> <datetime value expression>
<right paren>

LAST <left paren> <datetime value expression> <comma> <datetime value expression>

<right paren>

FIRST <left paren> <instant set value expression> <right paren>

LAST <left paren> <instant set value expression> <right paren>

VALID <left paren> { <table name> | <correlation name> } <right paren>

SCALE <left paren> <datetime value expression> AS <time granularity> <right paren>

NOBIND <left paren> <datetime literal> <right paren>

NOBIND <left paren> <column reference> <right paren>

<cast target> ::=

<domain name>

<data type>

<time granularity>

<value expression> ::=

<period value expression>

<temporal element value expression>

<instant set value expression>

<time zone specifier> ::=

LOCAL

TIME ZONE <interval value expression>

TIME ZONE <character string literal>

<interval value expression> ::=

| <interval value function>

<interval value function> ::=

INTERVAL <left paren> <period value expression> <right paren>

ABSOLUTE <left paren> <interval value expression> <right paren>

SCALE <left paren> <interval value expression> AS <time granularity> <right paren>

NOBIND <left paren> <interval literal> <right paren>

NOBIND <left paren> <column reference> <right paren>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<period value expression> ::=

<period primary>

| <interval value expression> <plus sign> <period value expression>

| <period value expression> { <plus sign> | <minus sign> } <interval value expression>

<period primary> ::=

<period literal>

<column reference>

<scalar subquery>

<case expression>

<period value function>

<cast specification>

<period value function> ::=

VALID <left paren> { <table name> | <correlation name> } <right paren>

TRANSACTION <left paren> { <table name> | <correlation name> } <right paren>

PERIOD <left paren> <datetime value expression> <comma> <datetime value expression>
<right paren>

INTERSECT <left paren> <period value expression> <comma>

<period value expression> <right paren>

FIRST <left paren> <temporal element value expression> <right paren>

LAST <left paren> <temporal element value expression> <right paren>

SCALE <left paren> <period value expression>

NOBIND <left paren> <period literal> <right paren>

NOBIND <left paren> <column reference> <right paren>

<temporal element value expression> ::=

<temporal element value term>

| <temporal element value expression> { <plus sign> | <minus sign> }

<temporal element value term>

<temporal element value term> ::=

<temporal element value factor>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<temporal element value factor> ::=

<temporal element value primary>

<temporal element value primary> ::=

<temporal element value function>

<temporal element value function> ::=

VALID <left paren> { <table name> | <correlation name> } <right paren>

| INTERSECT <left paren> <temporal element value expression> <comma>

<temporal element value expression> <right paren>

| SCALE <left paren> <temporal element expression> AS <time granularity> <right paren>

<instant set value expression> ::=

<instant set value primary>

| <instant set value expression> { <minus> | <plus> } <instant set value primary>

<instant set value primary> ::=

<instant set value function>

<instant set value function> ::=

VALID <left paren> { <table name> | <correlation name> } <right paren>

| INTERSECT <left paren> <instant set value expression> <comma>

<instant set value expression> <right paren>

<row value constructor> ::=

<row value constructor element>

<left paren> <row value constructor list> <right paren> [<valid value>]

<row subquery>

<valid value> ::=

VALID { <element value expression> | <interval value expression>

| <event value expression> | <event set value expression> }

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<table expression> ::=

[<valid clause>]
 <from clause>
 [<where clause>]
 [<group by clause>]
 [<having clause>]

<valid clause> ::=

{ VALID | VALID INTERSECT } { <temporal element value expression>
 | <period value expression> | <datetime value expression>
 | <instant set value expression> }

<where clause> ::=

WHERE <search condition> [WITH PLAUSIBILITY <integer>]

<grouping column reference> ::=

<column reference> [<collate clause>]
 | <temporal partition>

<temporal partition> ::=

{ VALID <left paren> { <table name> | <correlation name> } <right paren>
 | <column reference> }
 [USING { <partition expression> | INSTANT }]
 [LEADING <partition expression>]
 [TRAILING <partition expression>]

<partition expression> ::=

<integer>
 | <time granularity granularity>
 | <integer> <time granularity>
 | PERIOD 'All of time' <time granularity>

<select statement: single row> ::=

SELECT [<set quantifier>] [SNAPSHOT] <select list> <table expression>

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับหน่วยงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<select list> ::=

<column list>

<asterisk>

<asterisk> <asterisk>

<select sublist> [{ <comma> <select sublist> }...]

<predicate> ::=

<comparison predicate>

<between predicate>

<in predicate>

<like predicate>

<null predicate>

<quantified comparison predicate>

<exists predicate>

<unique predicate>

<match predicate>

<precedes predicate>

<meets predicate>

<overlaps predicate>

<contains predicate>

<overlaps predicate> ::=

<row value constructor 1> OVERLAPS <row value constructor 2>

| <row value expression 1> OVERLAPS <row value expression 2>

<precedes predicate> ::=

<row value expression 1> PRECEDES <row value expression 2>

<meets predicate> ::=

<row value expression 1> MEETS <row value expression 2>

<contains predicate> ::=

<row value expression 1> CONTAINS <row value expression 2>

```

<start field> ::=
    <time granularity> [ <left paren> <interval leading field precision> <right paren> ]
    | <left paren> <interval string> <interval qualifier> <right paren>

<end field> ::=
    <time granularity> [ <left paren> <interval fractional seconds precision> <right paren> ]

<single datetime field> ::=
    <time granularity> [ <left paren> <interval leading fixed position>
    [ <comma> <interval trailing field position> ] <right paren> ]

<timestamp qualifier> ::=
    [ <start field> TO ] <end field>
    | <single datetime field>

<period qualifier> ::=
    <timestamp qualifier>

<schema element> ::=
    | <create distribution statement>

<table definition> ::=
    CREATE [ { GLOBAL | LOCAL } TEMPORARY ] TABLE <table name>
    <table elements>
    [ <temporal definition> ]
    [ <vacuuming definition> ]
    [ ON COMMIT { DELETE | PRESERVE } ] ROWS ]

<temporal definition> ::=
    AS { VALID [ STATE | EVENT ] [ <timestamp precision> ]
    [ AND TRANSACTION ]
    | AS TRANSACTION

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<vacuuming definition> ::=

VACUUM <datetime value expression>

<column definition> ::=

<column name> {<data type>
 [INAPPLICABLE <value expression>]
 | <domain name> }
 [<default clause>]
 [<column constraint definition>...]
 [<collate clause>]

<default clause> ::=

<literal>
 <datetime value function>
 <interval value function>
 <period value function>
 USER
 SYSTEM USER
 NULL

<alter table action> ::=

<add valid definition>
 <drop valid definition>
 <replace valid def>
 <add transaction definition>
 <delete transaction definition>
 <scale valid definition>
 <cast valid definition>
 <alter vacuuming definition>

<add valid definition> ::=

ADD [VALID] { STATE | EVENT } [<timestamp precision>]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<drop valid definition> ::=

DROP VALID

<replace valid definition> ::=

REPLACE [VALID] [{ STATE | EVENT }] [<timestamp precision>]

<add transaction definition> ::=

ADD TRANSACTION

<drop transaction definition> ::=

DROP TRANSACTION

<scale valid definition> ::=

SCALE VALID AS <timestamp precision>

<cast valid definition> ::=

CAST VALID AS <timestamp precision>

<alter vacuuming definition> ::=

VACUUM <datetime value expression>

<create distribution statement> ::=

CREATE [{ GLOBAL | LOCAL } TEMPORARY] DISTRIBUTION

<distribution name> USING <table name>

<alter distribution statement> ::=

ALTER DISTRIBUTION <distribution name> USING <table name>

<drop distribution statement> ::=

DROP DISTRIBUTION <distribution name>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<module contents> ::=

<declare cursor>

<declare calendric system>

<dynamic declare cursor>

<procedure>

<declare calendric system> ::=

DECLARE CALENDRIC SYSTEM WITH <calendric spec>

<calendric spec> ::=

DEFAULT

| <calendric system name>

<SQL session statement> ::=

<SQL set identifier statement>

<set constraints mode statement>

<set transaction statement>

<set properties statement>

<set scale statement>

<set cast statement>

<set credibility statement>

<set plausibility statement>

<create distribution statement>

<alter distribution statement>

<drop distribution statement>

<schema set statement>

<set properties statement> ::=

SET PROPERTIES

[FOR CHARACTER SET [DEFAULT | NATIONAL | <character set>]]

[FOR { <time granularity> | <calendar name> }]

WITH <property spec>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<calendric-property specification> ::=

[<calendric-spec clause>] [<property-spec clause>...]

<calendric-spec clause> ::=

WITH CALENDRIC <calendric spec>

<property-spec clause> ::=

WITH PROPERTIES <property spec>

<property spec> ::=

PREVIOUS

DEFAULT

<property table name>

<table value constructor>

<schema set statement> ::=

SET SCHEMA <datetime value expression>

<set credibility statement> ::=

SET CREDIBILITY { <integer> | AS DEFAULT }

<set plausibility statement> ::=

SET PLAUSIBILITY { <integer> | AS DEFAULT }

<set scale statement> ::=

SET SCALE { <time granularity> | AS DEFAULT }

<set cast statement> ::=

SET CAST { <time granularity> | AS DEFAULT }

<fetch statement> ::=

FETCH [[<fetch orientation>] FROM] <cursor name> [INTO <fetch target list>]

[INTO VALID [PERIOD] <fetch target list>]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<select statement: single row> ::=

```
SELECT [ <set quantifier> ] [ SNAPSHOT ] <select list>
INTO <select target list>
<table expression>
```

<delete statement: searched> ::=

```
DELETE FROM <table name>
[ WHERE <search condition> ]
[ <valid value> ]
```

<insert column list> ::=

```
<insert column> [ { <comma> <insert column> } ... ]
```

<insert column> ::=

```
<column name>
| NEW
```

<update statement: searched> ::=

```
UPDATE <table name>
SET <set clause list>
[ <valid value> ]
[ WHERE <search condition> ]
```

ALTER TABLE TABLES ADD COLUMN

```
VALID_TIME CHARACTER DATA
CONSTRAINT VALID_TIME_CHECK
CHECK (VALID_TIME IN ('STATE', 'EVENT', 'NONE'))
```

ALTER TABLE TABLES ADD COLUMN

```
TRANSACTION_TIME CHARACTER DATA
CONSTRAINT TRANSACTION_TIME_CHECK
CHECK (TRANSACTION_TIME IN ('STATE', 'NONE'))
```

ALTER TABLE TABLES ADD COLUMN

```
VACUUM_CUT_OFF TIMESTAMP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CREATE TABLE TEMPORAL_SPEC {
    TABLE_NAME CHARACTER_DATA,
    VALID_SCALE INTERVAL,
    SCALE_GRANULARITY CHARACTER_DATA,
    VALID_PRECISION INTERVAL,
    PRECISION_GRANULARITY CHARACTER_DATA,
    DISTRIBUTION CHARACTER_DATA,
    GENERAL CHARACTER_DATA,
    DEFAULT_EVENT NONSTANDARD GENERAL INDETERMINATE TIMESTAMP,
    DEFAULT_STATE NONSTANDARD GENERAL INDETERMINATE PERIOD,
    CONSTRAINT TEMPORAL_SPEC_PRIMARY_KEY
        PRIMARY_KEY (TABLE_NAME),
    CONSTRAINT DISTRIBUTION_CHECK
        CHECK (DISTRIBUTION IN ('STANDARD', 'NONSTANDARD'))
    CONSTRAINT GENERAL_CHECK
        CHECK (GENERAL IN ('NONGENERAL', 'GENERAL'))
}

CREATE TABLE SURROGATE {
    TABLE_NAME CHARACTER_DATA,
    COLUMN_NAME CHARACTER_DATA,
    CONSTRAINT SURROGATE_PRIMARY_KEY
        PRIMARY_KEY (TABLE_NAME, COLUMN_NAME)
}

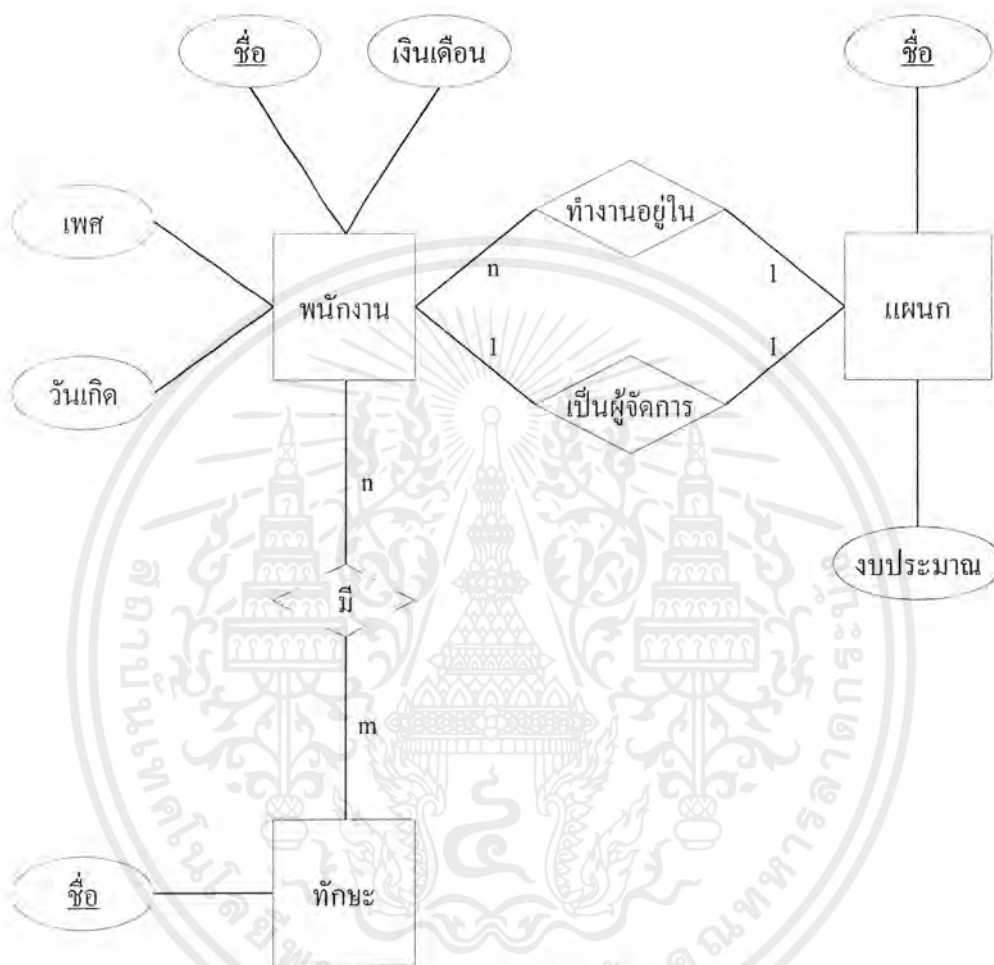
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

การใช้ภาษา SQL เชิงเวลา

สมมติว่ามีแบบจำลองความสัมพันธ์ของเอนทิตี (Entity Relationship Model) ดังรูปที่ 8-1



รูปที่ 8-1 แบบจำลองความสัมพันธ์ของเอนทิตี

เอนทิตีพนักงานมีแอททริบิวต์ดังนี้

- ชื่อ เก็บชื่อของพนักงาน
- เงินเดือน เก็บเงินเดือนของพนักงาน
- เพศ เก็บเพศของพนักงานเป็น “หญิง” และ “ชาย”
- วันเกิด เก็บวันเกิดของพนักงาน

เอนทิตีทักษะมีแอททริบิวต์เดียวคือ ชื่อ ซึ่งเก็บชื่อของทักษะ

ความสัมพันธ์ “มี” เป็นความสัมพันธ์ระหว่างทักษะและพนักงาน โดยเป็นแบบหลายต่อหลาย

(Many to Many Relationship) และข้อมูลสามารถเปลี่ยนแปลงได้ ยกตัวอย่างเช่น พนักงานจะสามารถขยับ
รถได้ในช่วงเวลาที่มิใช่ขยับที่เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอนิตีที่แผนกมีแอททริบิวต์ดังนี้

- ชื่อ เก็บชื่อของแผนก
- งบประมาณ เก็บงบประมาณของแผนกซึ่งข้อมูลสามารถเปลี่ยนแปลงได้

ความสัมพันธ์ “ทำงานอยู่ใน” เป็นความสัมพันธ์แบบหลายต่อหนึ่ง (Many to One Relationship)
บอกว่าพนักงานคนไหนทำงานอยู่ในแผนกอะไร

ความสัมพันธ์ “เป็นผู้จัดการ” เป็นความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to One Relationship)
บอกว่าพนักงานคนไหนเป็นผู้จัดการของแผนกอะไร

8.1 Temporal Data Definition Language และ Temporal Constraint สามารถสร้างตารางได้โดย

SET SCALE AS INTERVAL '1' DAY

CREATE TABLE พนักงาน (

รหัส	SURROGATE NOT NULL,
ชื่อ	CHARACTER (30) NOT NULL, PRIMARY KEY (ชื่อ),
เงินเดือน	DECIMAL (8, 2),
เพศ	CHARACTER (1),
วันเกิด	DATE,
ชื่อแผนก	CHARACTER (30), FOREIGN KEY (ชื่อแผนก) REFERENCES แผนก (ชื่อแผนก)

)

AS VALID STATE

CREATE TABLE ทักษะ (

รหัส	SURROGATE NOT NULL, FOREIGN KEY (รหัส) REFERENCES พนักงาน (รหัส),
ทักษะ	CHARACTER (30) NOT NULL

)

AS VALID STATE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CREATE TABLE แผนก (

ชื่อแผนก CHARACTER (30) NOT NULL,
 PRIMARY KEY (ชื่อแผนก),
 งบประมาณ DECIMAL (9, 0),
 รหัส SURROGATE NOT NULL,
 FOREIGN KEY (รหัส) REFERENCES พนักงาน (รหัส)

)

AS VALID STATE

8.2 Temporal Data Manipulation Language

คำสั่งต่อไปนี้จะเป็นการใส่และการแก้ไขข้อมูลของพนักงานชื่อสมชาย โดยกำหนดให้

1. สมชายทำงานในแผนกของเล่นตั้งแต่วันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนมกราคม ปีค.ศ. 1987
2. สมชายทำงานในแผนกหนังสือตั้งแต่วันที่ 1 เดือนเมษายน ปี ค.ศ. 1987 ถึงปัจจุบัน
3. สมชายเป็นผู้จัดการของแผนกหนังสือตั้งแต่วันที่ 1 เดือนเมษายน ปี ค.ศ. 1987 ถึงปัจจุบัน
4. งบประมาณของแผนกนี้เท่ากับ 50000 บาทตั้งแต่สมชายเข้ามาเป็นผู้จัดการ
5. สมชายมีชื่อว่าสมชายตั้งแต่วันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนธันวาคม ปี ค.ศ. 1987
6. สมชายมีชื่อว่าชาติชายตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1988 ถึงปัจจุบัน
7. สมชายมีเงินเดือน 20000 บาทตั้งแต่วันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนพฤษภาคม ปี ค.ศ. 1982
8. สมชายมีเงินเดือน 30000 บาทตั้งแต่วันที่ 1 เดือนมิถุนายน ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนมกราคม ปี ค.ศ. 1985
9. สมชายมีเงินเดือน 40000 บาทตั้งแต่วันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1985 ถึงวันที่ 31 เดือนมกราคม ปี ค.ศ. 1987 และตั้งแต่วันที่ 1 เดือนเมษายน ปี ค.ศ. 1987 ถึงปัจจุบัน
10. สมชายเป็นผู้ชายและเกิดวันที่ 1 เดือนกรกฎาคม ปี ค.ศ. 1955
11. สมชายสามารถพิมพ์ดีดได้ตั้งแต่วันที่ 1 เดือนเมษายน ปี ค.ศ. 1982
12. สมชายสามารถจัดเรียงเอกสารได้ตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1985
13. สมชายสามารถขับรถได้ตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 ถึงวันที่ 1 เดือนพฤษภาคม ปี ค.ศ. 1982 และตั้งแต่วันที่ 1 เดือนมิถุนายน ปี ค.ศ. 1984 ถึงวันที่ 31 เดือนพฤษภาคม ปี ค.ศ. 1988

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
INSERT INTO พนักงาน
VALUES (NEW, 'สมชาย', 20000, 'ชาย', DATE '07/01/1955', 'ของเล่น') VALID PERIOD
'[02/01/1982 - 01/31/1987]'
```

```
INSERT INTO พนักงาน
SELECT (รหัส, 'สมชาย', 40000, 'ชาย', DATE '07/01/1955', 'หนังสือ')
VALID PERIOD '[04/01/1987 - NOW]'
FROM พนักงาน
WHERE ชื่อ = 'สมชาย'
```

```
INSERT INTO แผนก
SELECT ('หนังสือ', 50000, พนักงาน.รหัส)
VALID PERIOD '[04/01/1987 - NOW]'
FROM พนักงาน
WHERE พนักงาน.ชื่อ = 'สมชาย'
```

```
UPDATE พนักงาน
SET ชื่อ TO 'ชาติชาย' VALID PERIOD '[01/01/1988 - NOW]'
WHERE รหัส = (SELECT DISTINCT รหัส FROM พนักงาน WHERE ชื่อ = 'สมชาย')
```

```
UPDATE พนักงาน
SET เงินเดือน TO 30000 VALID PERIOD '[06/01/1982 - 01/31/1985]'
WHERE รหัส = (SELECT DISTINCT รหัส FROM พนักงาน WHERE ชื่อ = 'สมชาย')
```

```
UPDATE พนักงาน
SET เงินเดือน TO 40000 VALID PERIOD '[02/01/1985 - 01/31/1987]'
WHERE รหัส = (SELECT DISTINCT รหัส FROM พนักงาน WHERE ชื่อ = 'สมชาย')
```

```
INSERT INTO ทักษะ
SELECT (รหัส, 'พิมพ์ดีด')
VALID PERIOD '[04/01/1982 - NOW]'
FROM พนักงาน
WHERE ชื่อ = 'สมชาย'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INSERT INTO ทักษะ

```
SELECT (รหัส, 'จัดเรียงเอกสาร')
VALID PERIOD '[01/01/1985 - NOW]'
FROM พนักงาน
WHERE ชื่อ = 'สมชาย'
```

INSERT INTO ทักษะ

```
SELECT (รหัส, 'ขับรถ')
VALID (PERIOD '[01/01/1982 - 05/01/1982]' + PERIOD '[06/01/1984 - 05/31/1988]')
FROM พนักงาน
WHERE ชื่อ = 'สมชาย'
```

คำสั่งต่อไปนี้จะเป็นการใส่และการแก้ไขข้อมูลของพนักงานชื่อสมหญิง โดยกำหนดให้

1. สมหญิงทำงานและเป็นผู้จัดการในแผนกของเล่นตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 ถึงปัจจุบัน
2. สมหญิงมีชื่อว่าสมหญิงตลอดเวลาในการทำงาน
3. งบประมาณของแผนกของเล่นเท่ากับ 150000 บาทตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนกรกฎาคม ปี ค.ศ. 1984
4. งบประมาณของแผนกของเล่นเท่ากับ 200000 บาทตั้งแต่วันที่ 1 เดือนสิงหาคม ปี ค.ศ. 1984 ถึงวันที่ 31 เดือนธันวาคม ปี ค.ศ. 1986
5. งบประมาณของแผนกของเล่นเท่ากับ 100000 บาทตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1987 ถึงปัจจุบัน
6. สมหญิงมีเงินเดือน 30000 บาทตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนกรกฎาคม ปี ค.ศ. 1984
7. สมหญิงมีเงินเดือน 40000 บาทตั้งแต่วันที่ 1 เดือนสิงหาคม ปี ค.ศ. 1984 ถึงวันที่ 31 เดือนสิงหาคม ปี ค.ศ. 1986
8. สมหญิงมีเงินเดือน 50000 บาทตั้งแต่วันที่ 1 เดือนกันยายน ปี ค.ศ. 1986 ถึงปัจจุบัน
9. สมหญิงเป็นผู้หญิง และเกิดวันที่ 1 เดือนตุลาคม ปี ค.ศ. 1960
10. สมหญิงสามารถบริหารงานได้ตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 จนถึงปัจจุบัน

INSERT INTO พนักงาน

```
VALUES (NEW, 'สมหญิง', 30000, 'หญิง', DATE '10/01/1960', 'ของเล่น') VALID PERIOD
'[01/01/1982 - NOW]'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INSERT INTO แผนก

```
SELECT ('ของเล่น', 150000, พนักงาน.รหัส)
VALID PERIOD '[01/01/1982 - NOW]'
FROM พนักงาน
WHERE พนักงาน.ชื่อ = 'สมหญิง'
```

UPDATE แผนก

```
SET งบประมาณ TO 200000 VALID PERIOD '[08/01/1984 - 12/31/1986]'
WHERE ชื่อ = 'ของเล่น'
```

UPDATE แผนก

```
SET งบประมาณ TO 100000 VALID PERIOD '[01/01/1987 - NOW]'
WHERE ชื่อ = 'ของเล่น'
```

UPDATE พนักงาน

```
SET เงินเดือน TO 40000 VALID PERIOD '[08/01/1984 - 08/31/1986]'
WHERE รหัส = (SELECT DISTINCT รหัส FROM พนักงาน WHERE ชื่อ = 'สมหญิง')
```

UPDATE พนักงาน

```
SET เงินเดือน TO 50000 VALID PERIOD '[09/01/1986 - NOW]'
WHERE รหัส = (SELECT DISTINCT รหัส FROM พนักงาน WHERE ชื่อ = 'สมหญิง')
```

INSERT INTO ทักษะ

```
SELECT (รหัส, 'บริหารงาน')
VALID PERIOD '[01/01/1982 - NOW]'
FROM พนักงาน
WHERE รหัส = (SELECT รหัส FROM พนักงาน WHERE ชื่อ = 'สมหญิง')
```

จะได้ตารางพนักงานมีลักษณะดังตารางที่ 8-1

รหัส	ชื่อ	เงินเดือน	เพศ	วันเกิด	ชื่อแผนก	เวลาเริ่มต้น	เวลาสิ้นสุด
1	สมชาย	20000	ชาย	07/01/1955	ของเล่น	02/01/1982	05/31/1982
1	สมชาย	30000	ชาย	07/01/1955	ของเล่น	06/01/1982	01/31/1985
1	สมชาย	40000	ชาย	07/01/1955	ของเล่น	02/01/1985	01/31/1987
1	สมชาย	40000	ชาย	07/01/1955	หนังสือ	04/01/1987	12/31/1987
1	ชาติชาย	40000	ชาย	07/01/1955	หนังสือ	01/01/1988	NOW
2	สมหญิง	30000	หญิง	10/01/1960	ของเล่น	01/01/1982	07/31/1984
2	สมหญิง	40000	หญิง	10/01/1960	ของเล่น	08/01/1984	08/31/1986
2	สมหญิง	50000	หญิง	10/01/1960	ของเล่น	09/01/1986	NOW

ตารางที่ 8-1 ตารางพนักงาน

ตารางทักษะมีลักษณะดังตารางที่ 8-2

รหัส	ทักษะ	เวลาเริ่มต้น	เวลาสิ้นสุด
1	พิมพ์ดีด	04/01/1982	NOW
1	จัดเรียงเอกสาร	01/01/1985	NOW
1	ขับรถ	01/01/1982	05/01/1982
1	ขับรถ	06/01/1984	05/31/1988
2	บริหารงาน	01/01/1982	NOW

ตารางที่ 8-2 ตารางทักษะ

และได้ตารางแผนกมีลักษณะดังตารางที่ 8-3

ชื่อแผนก	งบประมาณ	รหัส	เวลาเริ่มต้น	เวลาสิ้นสุด
ของเล่น	150000	2	01/01/1982	07/31/1984
ของเล่น	200000	2	08/01/1984	12/31/1986
ของเล่น	100000	2	01/01/1987	NOW
หนังสือ	50000	1	04/01/1987	NOW

ตารางที่ 8-3 ตารางแผนก

8.3 Temporal Query Language

กำหนดให้เวลาปัจจุบันซึ่งก็คือค่าของ CURRENT_DATE สำหรับการตอบคำถามต่อไปนี้เท่ากับ วันที่ 1 เดือนมกราคม ปี ค.ศ. 1990 การตอบคำถามเชิงเวลาแบ่งเป็น 3 ประเภท คือ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1 คำตอบเป็นค่าของแอททริบิวต์

8.3.1.1 แผนกใดมีผู้จัดการที่ทำงานเป็นช่วงเวลาติดต่อกันที่สั้นที่สุด (คำตอบ : 'หนังสือ')

```
SELECT SNAPSHOT ชื่อ
FROM แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D
WHERE CAST (VALID (D) AS INTERVAL DAY) <= ALL (SELECT CAST (VALID (D2) AS
INTERVAL DAY)
FROM แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D2
WHERE D2.ชื่อแผนก = D.ชื่อแผนก)
```

8.3.1.2 ใครทำงานติดต่อกันในแผนกหนังสือเป็นเวลาอย่างน้อยเท่ากับที่สมหญิงทำงานที่นั่น
(คำตอบ : ไม่มี)

```
SELECT SNAPSHOT E4.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1 E2, E1 (ชื่อ) AS E3, E2 (ชื่อ) AS E4
WHERE E3.ชื่อ = 'สมหญิง' AND E1.ชื่อแผนก = 'หนังสือ' AND E2.ชื่อแผนก = 'หนังสือ'
AND CAST (VALID (E2) AS INTERVAL DAY) > CAST (VALID (E1) AS INTERVAL DAY)
```

8.3.1.3 ใครทำงานติดต่อกันในแผนกของเล่นเป็นเวลาอย่างน้อยเท่ากับที่สมหญิงทำงานที่นั่น
(คำตอบ : 'สมชาย' และ 'ชาติชาย')

```
SELECT SNAPSHOT E4.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1 E2, E1 (ชื่อ) AS E3, E2 (ชื่อ) AS E4
WHERE E3.ชื่อ = 'สมหญิง' AND E1.ชื่อแผนก = 'ของเล่น' AND E2.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (E2) AS INTERVAL DAY) > CAST (VALID (E1) AS INTERVAL DAY)
```

8.3.1.4 ใครทำงานติดต่อกันในแผนกยาวนานกว่าผู้จัดการของแผนกนั้น (คำตอบ : ไม่มี)

```
SELECT SNAPSHOT E3.ชื่อ
FROM แผนก (ชื่อแผนก, รหัส) AS D, พนักงาน (รหัส) AS EM,
พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E2, E2 (ชื่อ) AS E3
WHERE D.รหัส = EM.รหัส
AND CAST (VALID (E2) AS INTERVAL DAY) > CAST (VALID (EM) AS INTERVAL DAY)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1.5 ใครมีเงินเดือนเท่าเดิมเป็นระยะเวลาติดต่อกันยาวนานที่สุด (คำตอบ : 'สมหญิง')

```
SELECT SNAPSHOT ชื่อ
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E, E (ชื่อ) AS E2
WHERE CAST (VALID (E) AS INTERVAL DAY) > ALL (SELECT CAST (VALID (E3) AS
INTERVAL DAY)
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E3
WHERE E3.รหัส <> E.รหัส)
```

8.3.1.6 ใครทำงานเป็นผู้จัดการในแผนกเป็นเวลาเท่ากับผู้จัดการ
(คำตอบ : 'สมชาย' 'ชาติชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E2.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E, แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D, E (ชื่อ) AS E2
WHERE E.ชื่อแผนก = D.ชื่อแผนก AND VALID (E) CONTAINS VALID (D)
```

8.3.1.7 ผู้จัดการคนใดทำงานติดต่อกันยาวนานกว่าผู้จัดการคนอื่นๆ (คำตอบ : 'สมหญิง')

```
SELECT SNAPSHOT E.ชื่อ
FROM แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D1 D2, พนักงาน (รหัส, ชื่อ) AS E
WHERE CAST (VALID (D1) AS INTERVAL DAY) > CAST (VALID (D2) AS INTERVAL DAY)
AND D1.รหัส = E.รหัส
```

8.3.1.8 พนักงานคนใดที่เงินเดือนไม่เปลี่ยนแปลงเป็นเวลายาวอย่างน้อย 3 ปี (คำตอบ : 'สมหญิง')

```
SELECT SNAPSHOT E2.ชื่อ
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E1, E1 (ชื่อ) AS E2
WHERE CAST (VALID (E1) AS INTERVAL YEAR) >= INTERVAL '3' YEAR
```

8.3.1.9 ใครทำงานกับผู้จัดการคนเดิมติดต่อกันเป็นระยะเวลาอย่างน้อย 5 ปี
(คำตอบ : 'สมชาย' 'ชาติชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E2.ชื่อ
FROM (SELECT E3.รหัส FROM พนักงาน (รหัส, ชื่อแผนก) AS E3, แผนก (ชื่อแผนก, รหัส) AS D
```

เอกสารนี้เป็นเอกสารที่สงวนเวลาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WHERE ชื่อแผนก = ชื่อแผนก) (PERIOD) AS E, พนักงาน (รหัส, ชื่อ) AS E2
WHERE E.รหัส = E2.รหัส
AND CAST (VALID (E) AS INTERVAL YEAR) >= INTERVAL '5' YEAR
```

8.3.1.10 พนักงานคนใดอยู่ในแผนกเดิมในระยะเวลา 5 ปีที่แล้ว (คำตอบ : 'สมหญิง')

```
SELECT SNAPSHOT E2.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, E1 (ชื่อ) AS E2
WHERE CAST (VALID (E1) AS INTERVAL DAY) >= INTERVAL '5' YEAR
AND VALID (E1) MEETS CURRENT_DATE
```

8.3.1.11 บอกชื่อแผนก ชื่อผู้จัดการ และงบประมาณของแผนกที่มีผู้จัดการคนเดิม และมีงบประมาณเท่าเดิมในระยะเวลา 18 เดือนที่แล้ว (คำตอบ : ('ของเล่น', 'สมหญิง', 100000) และ ('หนังสือ', 'สมชาย', 50000))

```
SELECT D2.ชื่อแผนก, E.ชื่อ, D2.งบประมาณ
FROM แผนก (ชื่อแผนก, รหัส, งบประมาณ) (PERIOD) AS D1 D2, พนักงาน (รหัส, ชื่อ) AS E
WHERE CAST (VALID (D1) AS INTERVAL MONTH) >= INTERVAL '18' MONTH
AND VALID (D1) OVERLAPS CURRENT_DATE
AND VALID (D2) OVERLAPS CURRENT_DATE
AND D1.ชื่อแผนก = D2.ชื่อแผนก AND D2.รหัส = E.รหัส
AND VALID (E) OVERLAPS CURRENT_DATE
```

8.3.1.12 ใครทำงานอยู่ในแผนกของเล่นและได้เงินเดือนอย่างน้อย 40000 บาทในระยะเวลา 2 ปีที่ผ่านมา (คำตอบ : 'สมชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E2.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1, E1 (ชื่อ) AS E2,
(SELECT รหัส FROM พนักงาน (รหัส,เงินเดือน) AS E
WHERE รหัส = E1.รหัส AND เงินเดือน >= 40000) (PERIOD) AS S
WHERE E1.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (S) AS INTERVAL DAY) >= INTERVAL '2' YEAR
AND VALID (S) MEETS CURRENT_DATE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1.13 ใครได้ขึ้นเงินเดือนอย่างน้อย 3 ครั้งตลอดระยะเวลา 5 ปี

(คำตอบ : 'สมหญิง' และ 'ชาติชาย')

```
SELECT SNAPSHOT E1.ชื่อ
FROM พนักงาน (รหัส, ชื่อ) AS E1,
(SELECT รหัส, CNT = COUNT (รหัส) FROM พนักงาน (รหัส, เงินเดือน) AS E
GROUP BY รหัส, VALID (E) USING INSTANT LEADING INTERVAL '5' YEAR) AS Temp
WHERE Temp.CNT >= 3 AND E1.รหัส = Temp.รหัส
```

8.3.1.14 ใครได้ขึ้นเงินเดือนมากที่สุดตลอดระยะเวลา 5 ปี (คำตอบ : 'สมหญิง' และ 'ชาติชาย')

```
SELECT SNAPSHOT E1.ชื่อ
FROM พนักงาน (รหัส, ชื่อ) AS E1,
(SELECT รหัส, COUNT (รหัส) FROM พนักงาน (รหัส, เงินเดือน) AS E
GROUP BY รหัส,
VALID (E) USING INSTANT LEADING INTERVAL '5' YEAR) AS Temp1 (รหัส, CNT)
(SELECT MAX (Temp2.CNT)
FROM (SELECT COUNT (รหัส) FROM พนักงาน (รหัส, เงินเดือน) AS E
GROUP รหัส, VALID (E) USING INSTANT LEADING INTERVAL '5' YEAR)
AS Temp2 (CNT)) AS Temp3 (SalMax)
WHERE Temp1.CNT = Temp3.SalMax AND E1.รหัส = Temp1.รหัส
```

8.3.1.15 ใครทำงานในแผนกของเล่นเป็นเวลานานอย่างน้อยเท่ากับที่สมหญิงทำงานที่นั่น

(คำตอบ : 'สมหญิง')

```
SELECT SNAPSHOT E4.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E2, พนักงาน (รหัส, ชื่อ) AS E3 E4
WHERE E.ชื่อแผนก = 'ของเล่น' AND E2.ชื่อแผนก = 'ของเล่น'
AND E3.ชื่อ = 'สมหญิง' AND E.รหัส = E3.รหัส AND E2.รหัส = E4.รหัส
AND CAST (VALID (E2) AS INTERVAL DAY) >= CAST (VALID (E) AS INTERVAL DAY)
```

8.3.1.16 ใครทำงานเป็นระยะเวลายาวนานกว่าที่ผู้จัดการของแผนกนั้นทำ (คำตอบ : ไม่มี)

```
SELECT SNAPSHOT E3.ชื่อ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FROM พนักงาน (รหัส, ชื่อแผนก) AS E E2, พนักงาน (รหัส, ชื่อ) AS E3, แผนก (ชื่อแผนก, รหัส) AS D
 WHERE E.ชื่อแผนก = D.ชื่อแผนก AND D.รหัส = E2.รหัส AND E.รหัส = E3.รหัส
 AND CAST (VALID (E) AS INTERVAL DAY) > CAST (VALID (E2) AS INTERVAL DAY)

8.3.1.17 ผู้จัดการคนใดที่ทำงานยาวนานกว่าที่สมหญิงเป็นผู้จัดการของแผนกของเล่น และใน
 แผนกใด (คำตอบ : ไม่มี)

SELECT SNAPSHOT E2.ชื่อ, D2.ชื่อแผนก
 FROM พนักงาน (รหัส, ชื่อ) AS E E2, แผนก (ชื่อแผนก, รหัส) AS D D2
 WHERE D.รหัส = E.รหัส AND E.ชื่อ = 'สมหญิง'
 AND D.ชื่อแผนก = 'ของเล่น' AND D2.รหัส = E2.รหัส
 AND CAST (VALID (D2) AS INTERVAL DAY) > CAST (VALID (D) AS INTERVAL DAY)

8.3.1.18 ใครมีเงินเดือนต่ำเดิมเป็นเวลานานที่สุด (คำตอบ : 'สมชาย')

SELECT SNAPSHOT E2.ชื่อ
 FROM พนักงาน (รหัส, เงินเดือน) AS E1, พนักงาน (รหัส, ชื่อ) AS E2
 (SELECT MAX (CAST (VALID (E) AS INTERVAL DAY)
 FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E) AS E3
 WHERE E2.รหัส = E1.รหัส

8.3.1.19 แผนกใดที่มีผู้จัดการทำงานเป็นระยะเวลาสั้นที่สุด (คำตอบ : 'หนังสือ')

SELECT SNAPSHOT D1.ชื่อแผนก
 FROM แผนก (ชื่อแผนก, รหัส) AS D1,
 (SELECT รหัส, Duration = CAST (VALID (พนักงาน) AS INTERVAL DAY)
 FROM แผนก (ชื่อแผนก, รหัส)) AS D
 WHERE CAST (VALID (D1) AS INTERVAL DAY) = MIN (D.Duration)

8.3.1.20 บอกรหัสพนักงานทุกคนที่ปัจจุบันทำงานอยู่ในแผนกหนังสือมานานกว่าที่สมชายทำ และมี
 เงินเดือนมากกว่า 40000 บาท (คำตอบ : ไม่มี)

SELECT SNAPSHOT E3.ชื่อ
 FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) E, พนักงาน (รหัส, เงินเดือน) AS E1 E2,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พนักงาน (รหัส, ชื่อ) AS E3 E4
 WHERE E.ชื่อแผนก = 'หนังสือ' AND VALID (E) OVERLAPS CURRENT_DATE
 AND E.รหัส = E3.รหัส AND E1.เงินเดือน > 40000 AND E2.เงินเดือน > 40000
 AND E3.รหัส = E1.รหัส AND E4.รหัส = E2.รหัส AND E4.ชื่อ = 'สมชาย'
 AND CAST (VALID (E1) AS INTERVAL DAY) > CAST (VALID (E2) AS INTERVAL DAY)

8.3.1.21 ใครทำงานอยู่ในแผนกของเล่นเป็นเวลานานอย่างน้อยเท่ากับเวลาที่สมชายไม่ได้เป็นผู้จัดการของแผนกของเล่น (คำตอบ : 'สมหญิง')

SELECT SNAPSHOT E.ชื่อ
 FROM พนักงาน (รหัส, ชื่อแผนก) AS E, พนักงาน (รหัส, ชื่อ) AS E2 E3, แผนก (ชื่อแผนก) AS D,
 แผนก (ชื่อแผนก, รหัส) AS D2
 WHERE E.ชื่อแผนก = 'ของเล่น' AND E2.รหัส = E.รหัส
 AND D.ชื่อแผนก = 'ของเล่น' AND D2.ชื่อแผนก = 'ของเล่น'
 AND D2.รหัส = E3.รหัส AND E3.ชื่อ = 'สมชาย'
 AND CAST (VALID (E) AS INTERVAL DAY) >= (VALID (D) - VALID (D2)) DAY

8.3.1.22 บอกชื่อพนักงานที่ทำงานอยู่ในแผนกของเล่นเป็นเวลาน้อยกว่าสมหญิง
 (คำตอบ : 'สมชาย' และ 'ชาติชาย')

SELECT SNAPSHOT E.ชื่อ
 FROM พนักงาน (รหัส, ชื่อ) AS E E2, พนักงาน (รหัส, ชื่อแผนก) AS E3 E4
 WHERE E.รหัส = E3.รหัส AND E2.รหัส = E4.รหัส AND E2.ชื่อ = 'สมหญิง'
 AND E3.ชื่อแผนก = 'ของเล่น' AND E4.ชื่อแผนก = 'ของเล่น'
 AND CAST (VALID (E3) AS INTERVAL DAY) < CAST (VALID (E4) AS INTERVAL DAY)

8.3.1.23 บอกชื่อปัจจุบันและแผนกปัจจุบันของพนักงานที่มีเงินเดือน 40000 บาทเป็นเวลานานกว่าสมหญิง (คำตอบ : ('ชาติชาย', 'หนังสือ'))

SELECT SNAPSHOT E.ชื่อ, E.ชื่อแผนก
 FROM พนักงาน (รหัส, ชื่อ, ชื่อแผนก) AS E, พนักงาน (รหัส, เงินเดือน) AS E2 E3,
 พนักงาน (รหัส, ชื่อ) AS E4
 WHERE E.รหัส = E2.รหัส AND VALID (E) OVERLAPS CURRENT_DATE
 AND E3.รหัส = E4.รหัส AND E4.ชื่อ = 'สมหญิง'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND E2.เงินเดือน = 40000 AND E3.เงินเดือน = 40000

AND CAST (VALID (E2) AS INTERVAL DAY) > CAST (VALID (E3) AS INTERVAL DAY)

8.3.1.24 ใครเป็นผู้จัดการแผนกหนังสือเป็นเวลาอย่างน้อย 2 ปี

(คำตอบ : 'สมชาย' และ 'ชาติชาย')

```
SELECT SNAPSHOT ชื่อ
FROM พนักงาน (รหัส, ชื่อ) AS E, แผนก (ชื่อแผนก, รหัส) AS D
WHERE E.รหัส = รหัส AND แผนก.ชื่อแผนก = 'หนังสือ'
AND CAST (VALID (D) AS INTERVAL YEAR) >= INTERVAL '2' YEAR
```

8.3.1.25 พนักงานคนใดมีเงินเดือนเท่าเดิมเป็นเวลาอย่างน้อย 3 ปี

(คำตอบ : 'สมชาย' 'ชาติชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E.ชื่อ
FROM พนักงาน (รหัส, ชื่อ) AS E, พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E1
WHERE E.รหัส = E1.รหัส
AND CAST (VALID (E1) AS INTERVAL YEAR) >= INTERVAL '3' YEAR
```

8.3.1.26 ใครทำงานกับผู้จัดการคนเดิมเป็นเวลาอย่างน้อย 5 ปี

(คำตอบ : 'สมชาย' 'ชาติชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E.ชื่อ
FROM พนักงาน (รหัส, ชื่อ) AS E,
(SELECT E1.รหัส, D.รหัส
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1, แผนก (ชื่อแผนก, รหัส) AS D
WHERE E1.ชื่อแผนก = D.ชื่อแผนก) (PERIOD) AS DT
WHERE E.รหัส = DT.รหัส
AND CAST (VALID (DT) AS INTERVAL YEAR) >= INTERVAL '5' YEAR
```

8.3.1.27 ใครทำงานในแผนกเป็นเวลาน้อยกว่า 6 เดือน (คำตอบ : ไม่มี)

```
SELECT SNAPSHOT E1.ชื่อ
FROM พนักงาน (รหัส, ชื่อ) AS E1, พนักงาน (รหัส, ชื่อแผนก) AS E2
```

WHERE E1.รหัส = E2.รหัส
AND CAST (VALID (E2) AS INTERVAL MONTH) < INTERVAL '6' MONTH

8.3.1.28 บอกทักษะของสมชายเมื่อสมชายเข้าทำงานในแผนกหนังสือ
(คำตอบ : 'พิมพ์ดีด' 'จัดเรียงเอกสาร' และ 'ขับรถ')

SELECT SNAPSHOT S.ทักษะ
FROM พนักงาน (รหัส, ชื่อ, ชื่อแผนก) AS E, ทักษะ AS S
WHERE E.ชื่อ = 'สมชาย' AND E.ชื่อแผนก = 'หนังสือ' AND E.รหัส = S.รหัส
AND VALID (S) CONTAINS FIRST (VALID (E))

8.3.1.29 บอกชื่อแผนกและงบประมาณของแผนกที่สมชายทำงาน
(คำตอบ : ('ของเล่น', 150000) และ ('หนังสือ', 50000))

SELECT SNAPSHOT E.ชื่อแผนก, D.งบประมาณ
FROM พนักงาน (ชื่อ, ชื่อแผนก) AS E, แผนก (ชื่อแผนก, งบประมาณ) AS D
WHERE E.ชื่อ = 'สมชาย' AND E.ชื่อแผนก = D.ชื่อแผนก
AND VALID (D) CONTAINS FIRST (VALID (E))

8.3.1.30 บอกข้อมูลทั้งหมดรวมทั้งทักษะของพนักงานที่เคยทำงานอยู่ในแผนกของเล่น
(คำตอบ : ('สมหญิง', 30000, 'ของเล่น', 'หญิง', '10/01/1960', 'บริหารงาน'))

SELECT SNAPSHOT E.*, S.*
FROM พนักงาน AS E, ทักษะ (รหัส, ทักษะ) AS S, แผนก(ชื่อแผนก) AS D
WHERE S.รหัส = E.รหัส AND VALID (E) CONTAINS FIRST (VALID (D))
AND E.ชื่อแผนก = 'ของเล่น' AND D.ชื่อแผนก = 'ของเล่น'

8.3.1.31 บอกชื่อของพนักงานที่ทำงานอยู่ในแผนกของเล่นก่อนที่จะถูกลดงบประมาณ
(คำตอบ : 'สมชาย' และ 'สมหญิง')

SELECT SNAPSHOT E3.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E2, พนักงาน (รหัส, ชื่อ) AS E3,
แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D1 D2
WHERE VALID (D1) MEETS VALID (D2) AND VALID (D1) PRECEDES VALID (D2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND D1.งบประมาณ > D2.งบประมาณ AND E.ชื่อแผนก = 'ของเล่น'
 AND E2.ชื่อ = 'ของเล่น' AND E1.รหัส = E2.รหัส AND E2.รหัส = E3.รหัส
 AND VALID (E) OVERLAPS VALID (D1) AND VALID (E) OVERLAPS VALID (D2)

8.3.1.32 บอกทักษะของสมชายเมื่อสมชายได้ขึ้นเงินเดือนจาก 30000 บาทเป็น 40000 บาท
 (คำตอบ : 'พิมพ์ดีด' 'จัดเรียงเอกสาร' และ 'ขับรถ')

```
SELECT SNAPSHOT ทักษะ
FROM ทักษะ, พนักงาน (เงินเดือน, รหัส) (PERIOD) AS E1 E2
WHERE VALID (E1) MEETS VALID (E2) AND VALID (E1) PRECEDES VALID (E2)
AND E1.รหัส = E2.รหัส AND E1.รหัส = S.รหัส
AND E1.เงินเดือน = 30000 AND E2.เงินเดือน = 40000
AND VALID (S) CONTAINS BEGIN (VALID (E2))
```

8.3.1.33 บอกชื่อ งบประมาณ และชื่อผู้จัดการปัจจุบันของแผนกของเล่น
 (คำตอบ : ('ของเล่น', 100000, 'สมหญิง'))

```
SELECT SNAPSHOT *
FROM แผนก AS D
WHERE ชื่อแผนก = 'ของเล่น' AND VALID (D) CONTAINS CURRENT_DATE
```

8.3.1.34 บอกทักษะที่สมชายมีหลังวันที่ 1 เดือนมกราคม ปี ค.ศ. 1983
 (คำตอบ : 'จัดเรียงเอกสาร' และ 'ขับรถ')

```
SELECT SNAPSHOT ทักษะ
FROM พนักงาน (รหัส, ชื่อ) AS E, ทักษะ (PERIOD) AS S
WHERE E.ชื่อ = 'สมชาย' AND E.รหัส = S.รหัส AND DATE '01/01/1983' PRECEDES VALID (S)
```

8.3.1.35 บอกเงินเดือนของสมหญิงในปีที่สมหญิงอายุ 25 ปี (คำตอบ : 40000)

```
SELECT SNAPSHOT E2.เงินเดือน
FROM พนักงาน AS E1 E2
WHERE E1.ชื่อ = 'สมหญิง' AND E1.รหัส = E2.รหัส
AND VALID (E1) CONTAINS (E2.วันเกิด + INTERVAL '25' YEAR)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1.36 บอกรหัสชื่อแผนกที่สมชายเคยทำงานก่อนวันที่ 1 เดือนมกราคม ปี ค.ศ. 1988
(คำตอบ : 'ของเล่น')

```
SELECT SNAPSHOT ชื่อแผนก
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1, พนักงาน (รหัส,ชื่อ) AS E2
WHERE E1.รหัส = E2.รหัส AND E2.ชื่อ = 'สมชาย'
AND VALID (E1) PRECEDES DATE '01/01/1988'
```

8.3.1.37 บอกวันเกิดและชื่อของผู้หญิงที่เคยทำงานในแผนกของเล่นในวันที่ 1 เดือนมกราคม ปี
ค.ศ. 1983 (คำตอบ : ('สมหญิง', '10/01/1960'))

```
SELECT SNAPSHOT E2.ชื่อ, E1.วันเกิด
FROM พนักงาน (รหัส, เพศ, วันเกิด, ชื่อแผนก) AS E1, E (ชื่อ) (PERIOD) AS E2
WHERE E1.เพศ = 'หญิง' AND E1.ชื่อแผนก = 'ของเล่น'
AND VALID (E1) CONTAINS DATE '01/01/1983' AND E1.รหัส = E2.รหัส
AND VALID (E2) CONTAINS CURRENT_DATE
```

8.3.1.38 ใครทำงานอยู่ในแผนกเป็นเวลานานกว่าที่ผู้จัดการปัจจุบันของแผนกนั้นทำ
(คำตอบ : ไม่มี)

```
SELECT SNAPSHOT E.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E M, แผนก (ชื่อแผนก, รหัส) AS D
WHERE VALID (E) CONTAINS CURRENT_DATE
AND VALID (D) CONTAINS CURRENT_DATE
AND E.ชื่อแผนก = D.ชื่อแผนก AND M.รหัส = D.รหัส
AND CAST (VALID (E) AS INTERVAL DAY) > CAST (VALID (M) AS INTERVAL DAY)
```

8.3.1.39 บอกรหัสชื่อของพนักงานทุกคนที่เปลี่ยนแผนกขณะที่สมหญิงทำงานอยู่ในแผนกของเล่น
(คำตอบ : 'สมชาย' และ 'ชาติชาย')

```
SELECT SNAPSHOT E3.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1 E2, E2 (ชื่อ) AS E3
WHERE E1.รหัส = E2.รหัส AND E2.ชื่อ = 'สมหญิง'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND E1.ชื่อแผนก = 'ของเล่น' AND E2.รหัส = E3.รหัส

AND VALID (E2) OVERLAPS VALID (E1) AND VALID (E2) MEETS VALID (E3)

8.3.1.40 บอกทักษะที่สมชายไม่มีขณะทำงานอยู่ในแผนกหนังสือ

(คำตอบ : 'ขับรถ' 'บริหารงาน' 'จัดเรียงเอกสาร' และ 'พิมพ์ดีด')

SELECT SNAPSHOT ทักษะ

FROM พนักงาน (รหัส, ชื่อ) AS E, ทักษะ AS S

WHERE E.รหัส = S.รหัส AND E.ชื่อ = 'สมชาย'

MINUS

SELECT SNAPSHOT ทักษะ

FROM ทักษะ (รหัส, ทักษะ) AS S1, พนักงาน (รหัส, ชื่อแผนก) AS E1 E2

WHERE E1.รหัส = E2.รหัส AND E2.ชื่อ = 'สมชาย'

AND E1.ชื่อแผนก = 'หนังสือ' AND E1.รหัส = S1.รหัส

AND VALID (E1) OVERLAPS BEGIN (VALID (S1))

8.3.1.41 บอกทักษะที่สมชายเคยมี แต่ไม่มีอยู่ขณะทำงานอยู่ในแผนกหนังสือ

(คำตอบ : 'ขับรถ' 'จัดเรียงเอกสาร' และ 'พิมพ์ดีด')

SELECT SNAPSHOT ทักษะ

FROM พนักงาน (รหัส, ชื่อ) AS E, ทักษะ AS S

WHERE E.รหัส = S.รหัส AND E.ชื่อ = 'สมชาย'

MINUS

SELECT SNAPSHOT S1.ทักษะ

FROM ทักษะ (รหัส, ทักษะ) AS S1, พนักงาน (รหัส, ชื่อแผนก) AS E1

WHERE E1.รหัส = E2.รหัส AND E2.ชื่อ = 'สมชาย'

AND E1.ชื่อแผนก = 'หนังสือ' AND E1.รหัส = S1.รหัส

8.3.1.42 บอกชื่อของพนักงานที่มีทักษะอันเดิมหลายครั้ง (คำตอบ : 'สมชาย')

SELECT SNAPSHOT ชื่อ

FROM พนักงาน (รหัส, ชื่อ), ทักษะ (รหัส, ทักษะ) (PERIOD) AS S1 S2

WHERE S1.รหัส = S2.รหัส AND รหัส = S1.รหัส AND S1.ทักษะ = S2.ทักษะ

AND VALID (S1) PRECEDES VALID (S2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1.43 บอกเพศและชื่อของพนักงานทุกคน ที่เริ่มทำงานก่อนที่สมชายจะมีทักษะขับรถเป็นครั้งที่สอง (คำตอบ : ('สมชาย', 'ชาย') และ ('สมหญิง', 'หญิง'))

```
SELECT SNAPSHOT E3.ชื่อ, E3.เพศ
FROM พนักงาน (รหัส, ชื่อ) AS E1, พนักงาน (ชื่อแผนก) (PERIOD) AS E2, E2 (ชื่อ, เพศ) AS E3,
ทักษะ (รหัส, ทักษะ) (PERIOD) AS S1 S2
WHERE E1.ชื่อ = 'สมชาย' AND E1.รหัส = S1.รหัสพนักงาน
AND S1.รหัส = S2.รหัส AND S1.ทักษะ = S2.ทักษะ
AND VALID (S1) PRECEDES VALID (S2)
AND NOT EXISTS (SELECT SNAPSHOT S3.รหัส
FROM ทักษะ(รหัส, ทักษะ) (PERIOD) AS S3
WHERE S3.รหัส = S1.รหัส AND S3.ทักษะ = S1.ทักษะ
AND VALID (S3) PRECEDES VALID (S1))
AND BEGIN (VALID (E2)) PRECEDES VALID (S2)
```

8.3.1.44 ใครทำงานอยู่ในแผนกเป็นเวลาอย่างน้อยที่ผู้จัดการของแผนกนั้นทำ (คำตอบ : 'สมชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT M1.รหัส
FROM (SELECT E.รหัส, E.ชื่อแผนก, D.รหัส FROM พนักงาน AS E, แผนก AS D
WHERE E.ชื่อแผนก = D.ชื่อแผนก) (PERIOD) AS M, M (E.รหัส, E.ชื่อแผนก) (PERIOD) AS M1,
M (E.ชื่อแผนก, D.รหัส) (PERIOD) AS M2
WHERE M1.รหัส = M2.รหัส AND VALID (M1) CONTAINS VALID (M2)
```

8.3.1.45 บอกเงินเดือนที่สูงที่สุดที่สมชายได้รับก่อนที่จะเปลี่ยนชื่อเป็นชาติชาย (คำตอบ : 40000)

```
SELECT SNAPSHOT MAX (E3.เงินเดือน)
FROM พนักงาน (รหัส, ชื่อ) (PERIOD) AS E1 E2, E1 (เงินเดือน) AS E3
WHERE E1.ชื่อ = 'สมชาย' AND E2.ชื่อ = 'ชาติชาย' AND E1.รหัส = E2.รหัส
AND VALID (E1) MEETS VALID (E2)
```

- 8.3.1.46 บอกชื่อและทักษะของพนักงานที่ทำงานอยู่ในแผนกหนังสือหรือแผนกของเล่นเมื่อปีที่แล้ว (คำตอบ : ('ชาติชาย', 'พิมพ์ดีด') ('ชาติชาย', 'จัดเรียงเอกสาร') ('ชาติชาย', 'ขับรถ') ('สมชาย', 'พิมพ์ดีด') ('สมชาย', 'จัดเรียงเอกสาร') ('สมชาย', 'ขับรถ') และ ('สมหญิง', 'บริหารงาน'))

```
SELECT SNAPSHOT E1.ชื่อ, S.ทักษะ
FROM พนักงาน (ชื่อ, ชื่อแผนก) (PERIOD) AS E1, ทักษะ (PERIOD) AS S
WHERE E1.รหัส = S.รหัส
AND E1.รหัส = ANY (SELECT SNAPSHOT E2.รหัส
FROM พนักงาน (รหัส, ชื่อแผนก) AS E2
WHERE (E2.ชื่อแผนก = 'ของเล่น' OR E2.ชื่อแผนก = 'หนังสือ')
AND VALID (E2) OVERLAPS PERIOD '[NOW-1 - NOW]' YEAR
```

- 8.3.1.47 บอกชื่อปัจจุบันและทักษะของพนักงานที่ทำงานอยู่ในแผนกหนังสือหรือแผนกของเล่นเมื่อปีที่แล้ว (คำตอบ : ('ชาติชาย', 'พิมพ์ดีด') ('ชาติชาย', 'จัดเรียงเอกสาร') และ ('สมหญิง', 'บริหารงาน'))

```
SELECT SNAPSHOT E1.ชื่อ, S.ทักษะ
FROM พนักงาน (ชื่อ, ชื่อแผนก) (PERIOD) AS E1, ทักษะ (PERIOD) AS S
WHERE E1.รหัส = S1.รหัส
AND VALID (E1) OVERLAPS CURRENT_DATE AND VALID (S) OVERLAPS CURRENT_DATE
AND E1.รหัส = ANY (SELECT SNAPSHOT E2.รหัส
FROM พนักงาน (รหัส, ชื่อแผนก) AS E2,
WHERE (E2.ชื่อแผนก = 'ของเล่น' OR E2.ชื่อแผนก = 'หนังสือ')
AND VALID (E2) OVERLAPS PERIOD '[NOW-1 - NOW]' YEAR
```

- 8.3.1.48 บอกชื่อของพนักงานทุกคนที่เคยเป็นลูกน้องของสมหญิงก่อนปีที่แล้ว (คำตอบ : 'สมชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E.ชื่อ
FROM พนักงาน (ชื่อ, ชื่อแผนก) (PERIOD) AS E, แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D,
WHERE E.ชื่อแผนก = D.ชื่อแผนก AND D.รหัส = E.รหัส AND E.ชื่อ = 'สมหญิง'
AND VALID (E) OVERLAPS VALID (D)
AND BEGIN (INTERSECT (VALID(E), VALID(D))) PRECEDES DATE 'NOW-1' YEAR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1.49 บอกชื่อผู้จัดการของพนักงานที่มีทักษะระหว่างปี ค.ศ. 1983 ถึงปี ค.ศ. 1987
(คำตอบ : 'สมชาย')

```
SELECT SNAPSHOT E2.ชื่อ
FROM พนักงาน (ชื่อ, ชื่อแผนก) (PERIOD) AS E1, แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D
WHERE E1.ชื่อแผนก = D.ชื่อแผนก AND VALID (E1) OVERLAPS CURRENT_DATE
AND VALID (D) OVERLAPS CURRENT_DATE
AND E1.รหัส = ANY (SELECT SNAPSHOT S.รหัส
FROM ทักษะ (PERIOD) AS S,
WHERE BEGIN (VALID (S)) OVERLAPS PERIOD '[NOW-7 - NOW-2]' YEAR)
```

8.3.1.50 บอกชื่อปัจจุบันของพนักงานที่สูญเสียทักษะใน 4 ปีที่แล้ว (คำตอบ : 'ชาติชาย')

```
SELECT SNAPSHOT E.ชื่อ
FROM พนักงาน (ชื่อ) (PERIOD) AS E
WHERE VALID (E) OVERLAPS CURRENT_DATE
AND E.รหัส = ANY (SELECT SNAPSHOT S.รหัส
FROM ทักษะ (PERIOD) AS S,
WHERE END (VALID (S)) OVERLAPS PERIOD '[NOW-4 - NOW]' YEAR)
```

8.3.1.51 บอกชื่อปัจจุบันและแผนกของพนักงาน ที่เปลี่ยนชื่อหรือเงินเดือนระหว่างเดือน
กรกฎาคม ปี ค.ศ. 1987 ถึงเดือนมิถุนายน ปี ค.ศ. 1988 (คำตอบ : ('ชาติชาย',
'หนังสือ'))

```
SELECT SNAPSHOT E1.ชื่อ, E1.ชื่อแผนก
FROM พนักงาน (ชื่อ, ชื่อแผนก) (PERIOD) AS E1
WHERE VALID (E1) OVERLAPS CURRENT_DATE
AND E1.รหัส = ANY (SELECT SNAPSHOT E2.รหัส
FROM พนักงาน (รหัส, ชื่อ) AS E3 E4, พนักงาน (รหัส, เงินเดือน) AS E5 E6
WHERE (E3.รหัส = E4.รหัส AND E3.ชื่อ <> E4.ชื่อ
AND VALID (E3) OVERLAPS PERIOD '[07/01/1987 - 06/01/1988]'
AND VALID (E4) OVERLAPS PERIOD '[07/01/1987 - 06/01/1988]')
OR (E5.รหัส = E6.รหัส AND E5.ชื่อ <> E6.ชื่อ)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND VALID (E5) OVERLAPS PERIOD '[07/01/1987 - 06/01/1988]'

AND VALID (E6) OVERLAPS PERIOD '[07/01/1987 - 06/1/1988]')

8.3.1.52 บอกรหัส ชื่อผู้จัดการปัจจุบัน และงบประมาณของทุกแผนกที่มีงบประมาณน้อยกว่า 200000 บาทระหว่างวันที่ 1 เดือนมกราคม ปี ค.ศ. 1985 ถึงวันที่ 31 เดือนธันวาคม ปี ค.ศ. 1989 (คำตอบ : ('ของเล่น', 'สมหญิง', 100000) และ ('หนังสือ', 'สมชาย', 50000))

```
SELECT SNAPSHOT D1.ชื่อแผนก, D1.งบประมาณ, E.ชื่อ
FROM แผนก (ชื่อแผนก, รหัส) (PERIOD) AS D1, พนักงาน (ชื่อ) (PERIOD) AS E
WHERE E.รหัส = D1.รหัส AND VALID (D) OVERLAPS CURRENT_DATE
AND VALID (E) OVERLAPS CURRENT_DATE
AND D1.ชื่อแผนก = ANY (SELECT SNAPSHOT D2.ชื่อแผนก
FROM แผนก (ชื่อแผนก, งบประมาณ) AS D2
WHERE D2.งบประมาณ < 200000
AND VALID (D2) OVERLAPS PERIOD '[01/01/1985 - 01/31/1989]')
```

8.3.1.53 ใครเคยทำงานในแผนกของเล่น และได้เงินเดือนอย่างน้อย 40000 บาทใน 2 ปีที่ผ่านมา (คำตอบ : 'สมชาย' และ 'สมหญิง')

```
SELECT SNAPSHOT E1.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, พนักงาน (รหัส, เงินเดือน) AS E2
WHERE E1.ชื่อแผนก = 'ของเล่น' AND E2.เงินเดือน >= 40000 AND E1.รหัส = E2.รหัส
AND SELECT SUM (VALID (E2)) CONTAINS PERIOD '[NOW-2 - NOW]' YEAR
```

8.3.1.54 บอกรหัสแผนกที่มีงบประมาณมากกว่า 90000 บาทตลอดเวลาที่สมหญิงเป็นผู้จัดการ (คำตอบ : 'ของเล่น')

```
SELECT SNAPSHOT D2.ชื่อแผนก
FROM แผนก (รหัส, ชื่อแผนก) AS D1, แผนก (รหัส, งบประมาณ) (PERIOD) AS D2,
พนักงาน (รหัส, ชื่อ) AS M
WHERE D1.รหัส = M.รหัส AND M.ชื่อ = 'สมหญิง' AND D2.งบประมาณ > 90000
AND D1.รหัส = D2.รหัส AND VALID (D2) CONTAINS VALID (D1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.1.55 บอกรายชื่อของสมชายเมื่อสมชายทำงานแผนกเดียวกับสมหญิง
(คำตอบ : 20000 30000 และ 40000)

```
SELECT SNAPSHOT E3.เงินเดือน
FROM พนักงาน (ชื่อ, ชื่อแผนก) AS E1, E1 (เงินเดือน) AS E3
WHERE E1.ชื่อ = 'สมชาย' AND E2.ชื่อ = 'สมหญิง' AND E1.ชื่อแผนก = E2.ชื่อแผนก
```

8.3.1.56 บอกรายชื่อแผนกที่สมชายทำงานขณะได้รับเงินเดือน 40000 บาท
(คำตอบ : 'ของเล่น' และ 'หนังสือ')

```
SELECT SNAPSHOT ชื่อแผนก
FROM พนักงาน (ชื่อ, เงินเดือน, ชื่อแผนก) AS E
WHERE ชื่อ = 'สมชาย' AND เงินเดือน = 40000
```

8.3.1.57 บอกรายชื่อของสมชายหลังจากที่ย้ายออกจากแผนกของเล่น
(คำตอบ : 'สมชาย' และ 'ชาติชาย')

```
SELECT SNAPSHOT E3.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, E2 (ชื่อ) AS E3
WHERE E1.ชื่อแผนก = 'ของเล่น' AND E2.ชื่อแผนก <> 'ของเล่น' AND E1.รหัส = E2.รหัส
AND VALID (E1) MEETS VALID (E2)
```

8.3.1.58 บอกรายชื่อของสมชายที่มีชื่อทำงานอยู่แผนกของเล่น
(คำตอบ : 'ขั้บรถ' 'จัดเรียงเอกสาร' และ 'พิมพ์ดีด')

```
SELECT SNAPSHOT S1.ทักษะ
FROM พนักงาน (รหัส, ชื่อ, ชื่อแผนก) AS E1, ทักษะ (รหัส, ทักษะ) AS S1
WHERE E1.ชื่อ = 'สมชาย' AND E1.ชื่อแผนก = 'ของเล่น' AND E1.รหัส = S1.รหัส
```

8.3.1.59 บอกรายชื่อของสมชายทั้งหมดหลังจากเปลี่ยนชื่อเป็นชาติชาย (คำตอบ : ไม่มี)

```
SELECT SNAPSHOT S1.ทักษะ
FROM พนักงาน (รหัส, ชื่อ) (PERIOD) AS E1, E2, ทักษะ (รหัส, ทักษะ) AS S1
WHERE E1.ชื่อ = 'สมชาย' AND E2.ชื่อ = 'ชาติชาย' AND E1.รหัส = E2.รหัส
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

AND VALID (E1) MEETS VALID (E2) AND E2.รหัส = S1.รหัส
MINUS
SELECT SNAPSHOT S1.ทักษะ
FROM พนักงาน (รหัส, ชื่อ) (PERIOD) AS E1 E2, ทักษะ (รหัส, ทักษะ) AS S1
WHERE E1.รหัส = 'สมชาย' AND E1.รหัส = S1.รหัส AND E1.ชื่อ <> 'ชาติชาย'
AND E2.ชื่อ = 'ชาติชาย' AND E1.รหัส = E2.รหัส AND VALID (E1) MEETS VALID (E2)

```

8.3.1.60 บอกงบประมาณของแผนกของเล่นเมื่อมีสมหญิงเป็นผู้จัดการ
(คำตอบ : 150000 200000 และ 100000)

```

SELECT SNAPSHOT D1.งบประมาณ
FROM แผนก (ชื่อแผนก, รหัส, งบประมาณ) AS D1, พนักงาน (รหัส, ชื่อ) AS M
WHERE D1.รหัส = M.รหัส AND M.ชื่อ = 'สมหญิง' AND D1.ชื่อแผนก = 'ของเล่น'

```

8.3.1.61 บอกชื่อผู้จัดการและชื่อแผนกระหว่างวันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 ถึงวันที่ 31
เดือนธันวาคม ปี ค.ศ. 1989
(คำตอบ : ('สมหญิง', 'ของเล่น') ('สมชาย', 'หนังสือ') และ ('ชาติชาย', 'หนังสือ'))

```

SELECT SNAPSHOT M.ชื่อ, D.ชื่อแผนก
FROM แผนก (ชื่อแผนก, รหัส) AS D, พนักงาน (รหัส, ชื่อ) AS M
WHERE D.รหัส = M.รหัส AND VALID (D) CONTAINS PERIOD '[01/01/1982 - 12/31/1989]'

```

8.3.2 คำตอบเป็นวลีใดใหม่

8.3.2.1 บอกช่วงเวลาพนักงานทำงานในแผนกของเล่นน้อยกว่าที่สมหญิงทำงานในแผนก
หนังสือ (คำตอบ : ([04/01/1987 – NOW])

```

SELECT VALID (E1)
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1 E2 E3, E3 (ชื่อ) AS E4
WHERE E1.ชื่อแผนก = 'หนังสือ' AND E1.รหัส = E2.รหัส
AND E2.ชื่อแผนก = 'ของเล่น' AND E3.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (E2) AS INTERVAL DAY) < CAST (VALID (E3) AS INTERVAL DAY)
AND E4.ชื่อ = 'สมหญิง'

```

8.3.2.2 บอกช่วงเวลาทำงานของพนักงานที่ได้เงินเดือน 40000 บาทมากกว่าสมหญิง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(คำตอบ : ([02/01/1982 – 01/31/1987]) และ ([04/01/1987 – NOW]))

```
SELECT VALID (E1)
FROM พนักงาน (รหัส, เงินเดือน) AS E1 E2, E2 (ชื่อ) AS E3
WHERE E1.เงินเดือน = 40000 AND E2.เงินเดือน = 40000 AND E3.ชื่อ = 'สมหญิง'
AND CAST (VALID (E1) AS INTERVAL DAY) > CAST (VALID (E2) AS INTERVAL DAY)
```

8.3.2.3 บอกวันเริ่มทำงานในแผนกหนังสือของพนักงาน ที่มีทักษะจัดเรียงเอกสารมากกว่าสมหญิง (คำตอบ : 04/01/1987)

```
SELECT BEGIN (E1)
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1, ทักษะ (รหัส, ทักษะ) AS E2 E3, E3 (ชื่อ) AS E4
WHERE E1.ชื่อแผนก = 'หนังสือ' AND E1.รหัส = E2.รหัส
AND E2.ทักษะ = 'จัดเรียงเอกสาร' AND E4.ชื่อ = 'สมหญิง' AND E3.ทักษะ = 'จัดเรียงเอกสาร'
AND CAST (VALID (E2) AS INTERVAL DAY) > CAST (VALID (E3) AS INTERVAL DAY)
```

8.3.2.4 บอกเวลาที่พนักงานทำงานสั้นกว่าที่สมหญิงมีทักษะ (คำตอบ : 01/01/1982, 04/01/1982, 06/01/1984 และ 01/01/1985)

```
SELECT BEGIN (S4)
FROM พนักงาน (รหัส) AS E1 E2, E2 (ชื่อ) AS E3, ทักษะ (รหัส, ทักษะ) AS S4
WHERE E1.รหัส = S4.รหัส AND E3.ชื่อ = 'สมหญิง'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E2) AS INTERVAL DAY)
```

8.3.2.5 บอกช่วงเวลาทำงานที่พนักงานทำน้อยกว่าสมหญิง (คำตอบ : ([02/01/1982 – 01/31/1987]) และ ([04/01/1987 – NOW]))

```
SELECT VALID (E1)
FROM พนักงาน (รหัส) AS E1 E2, E2 (ชื่อ) AS E3
WHERE E3.ชื่อ = 'สมหญิง'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E2) AS INTERVAL DAY)
```

8.3.2.6 เมื่อใดที่มีพนักงานได้ขึ้นเงินเดือนก่อนที่สมหญิงจะได้ขึ้นเงินเดือนครั้งแรก

(คำตอบ : 06/01/1982 และ 09/01/1986)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT END (E1)
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E1 E3, E3 (ชื่อ) AS E4
WHERE EXISTS (SELECT E2.รหัส FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E2
WHERE E1.รหัส = E2.รหัส AND E1.เงินเดือน < E2.เงินเดือน
AND END (VALID (E1)) = BEGIN (VALID (E2)))
AND E4.ชื่อ = 'สมหญิง'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E3) AS INTERVAL DAY)
AND E3.เงินเดือน <= ALL (SELECT E5.รหัส FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E5
WHERE E3.รหัส = E5.รหัส AND E3.เงินเดือน < E5.เงินเดือน)

```

8.3.2.7 ช่วงเวลาที่นานที่สุดที่ไม่มีพนักงานออกจากบริษัท

(คำตอบ : ([02/01/1982 – 01/31/1987]))

```

CREATE VIEW VT (C1) AS
SELECT BEGIN (VALID (E1)) FROM พนักงาน (รหัส) (PERIOD) AS E1
UNION
SELECT END (VALID (E2)) FROM พนักงาน (รหัส) (PERIOD) AS E2;
SELECT MAX (V2.C1 - V1.C1)
FROM VT V1 V2
WHERE V2.C1 IN (SELECT MIN (C1) FROM VT WHERE VT.C1 > V1.C1);

```

8.3.2.8 ช่วงเวลาที่นานที่สุดที่ไม่มีใครได้ขึ้นเงินเดือน (คำตอบ : ([09/01/1986 – NOW]))

```

CREATE VIEW VT (C1) AS
SELECT END (E1) FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E1
WHERE EXISTS (SELECT E2.รหัส FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E2
WHERE E1.รหัส = E2.รหัส AND E1.เงินเดือน < E2.เงินเดือน
AND END (VALID (E1)) = BEGIN (VALID (E2)));
SELECT MAX (V2.C1 - V1.C1) FROM VT AS V1 V2
WHERE V2.C1 IN (SELECT MAX (CAST (VALID(C1) AS INTERVAL DAY) FROM VT
WHERE VT.C1 > V1.C1);

```

8.3.2.9 ช่วงเวลาที่นานที่สุดที่แต่ละแผนกไม่มีผู้จัดการ (คำตอบ : ไม่มี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SELECT MAX (CAST (VALID (NOMGR) AS INTERVAL DAY)
FROM (SELECT * FROM แผนก MINUS
SELECT D.* FROM แผนก (ชื่อแผนก, รหัส) AS D) AS NOMGR
```

8.3.2.10 ช่วงเวลาทำงานของพนักงานที่ทำงานอยู่ในแผนกของเล่นเป็นเวลาอย่างน้อย 8 ปี
(คำตอบ : ([01/01/1982 – NOW]))

```
SELECT VALID (E)
FROM พนักงาน (รหัส) (PERIOD) AS E
WHERE ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (พนักงาน) AS INTERVAL YEAR) > INTERVAL '8' YEAR
```

8.3.2.11 บอกเวลาเริ่มทำงานของผู้จัดการที่เป็นผู้จัดการอย่างน้อย 5 ปี
(คำตอบ : 01/01/1982)

```
SELECT BEGIN (VALID (D))
FROM แผนก (รหัส, ชื่อแผนก) (PERIOD) AS D
WHERE CAST (VALID (D) AS INTERVAL YEAR) > INTERVAL '5' YEAR
```

8.3.2.12 บอกวันที่ที่พนักงานกลับมาทำงานหลังจากลาออกไปแล้วเกิน 1 เดือน
(คำตอบ : 04/01/1987)

```
SELECT BEGIN (VALID (E2))
FROM พนักงาน (รหัส) (PERIOD) AS E1 E2
WHERE (E1.รหัส = E2.รหัส) AND VALID (E1) PRECEDES VALID (E2)
AND ((END (VALID (E2)) – BEGIN (VALID (E1))) MONTH) > INTERVAL '1' MONTH
```

8.3.2.13 ช่วงเวลาที่มีพนักงานสูญเสียทักษะและได้รับกลับมามากกว่า 1 ปี
(คำตอบ : ([01/01/1982 – 05/01/1982], [06/01/1984 – 05/31/1988]))

```
SELECT VALID (S1), VALID (S2)
FROM ทักษะ (รหัส, ทักษะ) (INTERVAL) AS S1 S2
WHERE S1.รหัส = S2.รหัส AND S1.ทักษะ = S2.ทักษะ AND VALID (S1) PRECEDES VALID (S2)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือนำไปใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND ((END (VALID (S2)) – BEGIN (VALID (S1))) YEAR) > INTERVAL '1' YEAR

8.3.2.14 หาช่วงเวลาของงบประมาณที่มากกว่า 2 ปี

(คำตอบ : [01/01/1987 – NOW] และ [04/01/1984 – NOW])

SELECT VALID (D)

FROM แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D

WHERE CAST (VALID (D) AS INTERVAL YEAR) > INTERVAL '2' YEAR

8.3.2.15 ช่วงเวลาที่ไม่มีใครเปลี่ยนเงินเดือนเป็นเวลาอย่างน้อย 6 เดือน

(คำตอบ : ((06/01/1982 – 07/31/1984)) ((08/01/1984 – 01/31/1985)) ((02/01/1985 – 08/31/1986)) และ ((09/01/1986 – NOW)))

SELECT VALID (E)

FROM พนักงาน (เงินเดือน) (PERIOD) AS E

WHERE CAST (VALID (E) AS INTERVAL MONTH) => INTERVAL '6' MONTH

8.3.2.16 เมื่อใดที่ไม่มีใครมีทักษะอย่างน้อยเท่ากับทักษะที่สมหญิงมีอยู่ในปัจจุบัน

(คำตอบ : ([01/01/1982 – NOW]))

SELECT CAST (VALID (S5) AS INTERVAL DAY)

FROM ทักษะ (รหัส) AS S5,

(SELECT SNAPSHOT S4.รหัส FROM ทักษะ AS S4

MINUS (SELECT รหัส FROM (SELECT DT1.รหัส, DT2.ทักษะ

FROM (SELECT SNAPSHOT S1.รหัส FROM ทักษะ AS S1) AS DT1,

(SELECT SNAPSHOT S2.ทักษะ FROM ทักษะ AS S2, พนักงาน (รหัส, ชื่อ) AS E

WHERE E.รหัส = S2.รหัส AND E.ชื่อ = 'สมหญิง'

AND VALID (S2) CONTAINS CURRENT_DATE)) AS DT2)

MINUS (SELECT SNAPSHOT S3.รหัส, S3.ทักษะ FROM ทักษะ AS S3))) (รหัส) AS DT

WHERE S5.รหัส = DT.รหัส

8.3.2.17 เมื่อใดที่งบประมาณของแผนกของเล่นลดลง (คำตอบ : 01/01/1987)

SELECT SNAPSHOT BEGIN (VALID (D2))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FROM แผนก (ชื่อแผนก) AS D, D (งบประมาณ) (PERIOD) AS D1 D2
 WHERE D1 MEETS D2 AND D1.งบประมาณ > D2.งบประมาณ

8.3.2.18 เมื่อใดที่มีพนักงานเปลี่ยนชื่อ (คำตอบ : 01/01/1988)

SELECT SNAPSHOT BEGIN (VALID (N2))
 FROM พนักงาน (รหัส) AS E, E (ชื่อ) (PERIOD) AS N1 N2
 WHERE VALID (N1) MEETS VALID (N2) AND N1.ชื่อ <> N2.ชื่อ

8.3.2.19 เมื่อใดที่พนักงาน ได้ขึ้นเงินเดือนขณะที่เป็นผู้จัดการ
 (คำตอบ : 08/01/1984 และ 09/01/1986)

SELECT SNAPSHOT BEGIN (VALID (Sal2))
 FROM พนักงาน (รหัส) AS Mgr, Mgr (เงินเดือน) (PERIOD) AS Sal1 Sal2
 WHERE VALID (Sal1) MEETS VALID (Sal2) AND Sal1.เงินเดือน < Sal2.เงินเดือน
 AND Mgr.รหัส IN (SELECT รหัส FROM แผนก
 WHERE VALID (แผนก) CONTAINS BEGIN (VALID (Sal2))))

8.3.2.20 ช่วงเวลาที่สมหญิงได้เงินเดือน 40000 บาทและเป็นผู้จัดการของแผนกของเล่น
 (คำตอบ : ([08/01/1984 – 08/31/1984]))

SELECT SNAPSHOT INTERSECT (VALID (พนักงาน), VALID (D))
 FROM พนักงาน, แผนก (รหัส) AS D
 WHERE พนักงาน.ชื่อ = 'สมหญิง' AND พนักงาน.เงินเดือน = 40000 AND D.รหัส = พนักงาน.รหัส

8.3.2.21 บอกวันเริ่มต้นที่สมชายมีทักษะใดๆก่อนปีที่สมชายจะเข้าไปอยู่แผนกของเล่น
 (คำตอบ : 01/01/1982 และ 04/01/1982)

SELECT SNAPSHOT BEGIN (VALID (S))
 FROM ทักษะ (PERIOD) AS S, พนักงาน (รหัส, ชื่อแผนก) AS E
 WHERE E.ชื่อ = 'สมชาย' AND E.ชื่อแผนก = 'ของเล่น' AND S.รหัส = E.รหัส
 AND BEGIN (VALID (S)) <= CAST (CAST (BEGIN (E) AS INTERVAL YEAR)
 + INTERVAL '1' YEAR) AS INTERVAL DAY) - INTERVAL '1' DAY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.2.22 วันเริ่มต้นที่สมชายมีชื่อว่าชาติชายและมีเงินเดือนคงที่จนถึงปี ค.ศ. 1989
(คำตอบ : 01/01/1988)

```
SELECT SNAPSHOT BEGIN (INTERSECT VALID (E1), VALID (E2))
FROM พนักงาน (รหัส, ชื่อ) (PERIOD) AS E1, พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E2
WHERE E1.ชื่อ = 'ชาติชาย' AND E1.รหัส = E2.รหัส
AND INTERSECT (VALID (E1), VALID (E2)) OVERLAPS PERIOD '[1989]'
```

8.3.2.23 บอกวันเริ่มต้นที่สมชายมีทักษะใดๆก่อนหรือหลังปี ค.ศ. 1984 และ 1985
(คำตอบ : 01/01/1982 และ 04/01/1982)

```
SELECT SNAPSHOT BEGIN (VALID (S))
FROM ทักษะ (PERIOD) AS S, พนักงาน (รหัส, ชื่อ) AS E
WHERE S.รหัส = E.รหัส AND E.ชื่อ = 'สมชาย'
AND (BEGIN (VALID (S)) PRECEDES DATE '01/01/1984'
OR DATE '12/31/1985' PRECEDES BEGIN (VALID (S)))
```

8.3.2.24 บอกช่วงเวลาที่สมชายไม่ได้ทำงานในขณะที่สมชายอายุยังไม่ถึง 30 ปี
(คำตอบ : ไม่มี)

```
SELECT SNAPSHOT PERIOD (END (VALID (E1)) + INTERVAL '1' DAY,
BEGIN (VALID (E2)) - INTERVAL '1' DAY) - PERIOD (E1.วันเกิด + INTERVAL '30' YEAR,
E1.วันเกิด + INTERVAL '31' YEAR - INTERVAL '1' DAY))
FROM พนักงาน (รหัส) (STATE) AS E1 E2, พนักงาน (รหัส) AS E3
WHERE E1.ชื่อ = 'สมชาย' AND E2.รหัส = E1.รหัส AND E3.รหัส = E1.รหัส
AND NOT (VALID (E3) CONTAINS PERIOD (END (VALID (E1)) + INTERVAL '1' DAY,
BEGIN (VALID (E2)) - INTERVAL '1' DAY))
```

8.3.2.25 บอกช่วงเวลาทำงานที่สมหญิงทำงานในแผนกที่สมหญิงทำอยู่ตลอดปี ค.ศ. 1987
(คำตอบ : ([01/01/1982 - 07/31/1984]) ([08/01/1984 - 08/31/1986]) และ ([09/01/1986
- NOW]))

```
SELECT SNAPSHOT PERIOD (BEGIN (VALID (E1)), END (VALID (E1)))
FROM พนักงาน (ชื่อ, ชื่อแผนก) (PERIOD) AS E1 E2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WHERE E1.ชื่อ = 'สมหญิง' AND E2.ชื่อ = 'สมหญิง' AND E1.ชื่อแผนก = E2.ชื่อแผนก
AND VALID (E2) CONTAINS PERIOD '[1987]'

8.3.2.26 บอกวันที่ที่แผนกของเล่นเปลี่ยนแปลงงบประมาณในระหว่างวันที่ 1 เดือนมกราคม ปี
ค.ศ. 1983 ถึงวันที่ 31 เดือนธันวาคม ปี ค.ศ. 1985 (คำตอบ : 08/01/1984)

SELECT SNAPSHOT BEGIN (VALID (D2))
FROM แผนก (ชื่อแผนก, งบประมาณ) AS D1 D2
WHERE D1.ชื่อแผนก = 'ของเล่น' AND D2.ชื่อแผนก = 'ของเล่น'
AND END (VALID (D1)) PRECEDES BEGIN (VALID (D2))
AND PERIOD '[01/01/1983, 12/31/1985]' CONTAINS
PERIOD (END (VALID (D1)), BEGIN (VALID (D2)))
AND D1.งบประมาณ <> D2.งบประมาณ

8.3.2.27 บอกช่วงเวลาที่พนักงานมีทักษะเดียวกับสมหญิงในช่วงเวลาเดียวกัน (คำตอบ : ไม่มี)

SELECT SNAPSHOT INTERSECT (VALID (E), VALID (S))
FROM พนักงาน (รหัส) AS E, ทักษะ (รหัส, ทักษะ) AS S
WHERE E.รหัส = ทักษะ.รหัส AND ทักษะ.ทักษะ CONTAINS
SELECT SDi.ทักษะ
FROM พนักงาน (รหัส, ชื่อ) AS EDi, ทักษะ (รหัส, ทักษะ) AS SDi
WHERE EDi.ชื่อ = 'สมหญิง' AND EDi.รหัส = SDi.รหัส

8.3.2.28 บอกช่วงเวลาที่แผนกของเล่นมีงบประมาณมากกว่า 100000 บาท
(คำตอบ : [01/01/1982 – 12/31/1986])

SELECT VALID (D)
FROM แผนก (ชื่อแผนก, งบประมาณ) AS D
WHERE D.งบประมาณ > 100000 AND D.ชื่อแผนก = 'ของเล่น'

8.3.2.29 บอกช่วงเวลาที่สมชายมีชื่อว่าชาติชายและมีทักษะขับรถ จัดเรียงเอกสาร และพิมพ์ดีด
(คำตอบ : [01/01/1988 – 05/31/1988])

SELECT SNAPSHOT MAX (INTERSECT (VALID (E), INTERSECT (VALID (SDrive),

```

INTERSECT (VALID (SFile), VALID (SType))))
FROM พนักงาน (รหัส, ชื่อ) (PERIOD) AS E, ทักษะ (รหัส, ทักษะ) (PERIOD) AS SDrive SFile SType
WHERE E.ชื่อ = 'ชาติชาย' AND E.รหัส = SDrive.รหัส
AND E.รหัส = SFile.รหัส AND E.รหัส = SType.รหัส
AND VALID (E) OVERLAPS INTERSECT (VALID (SFile),
INTERSECT (VALID (SDrive), VALID (SType)))

```

8.3.2.30 บอกช่วงเวลาที่มีสมหญิงมีเงินเดือนมากกว่าสมชาย (คำตอบ : ไม่มี)

```

SELECT SNAPSHOT VALID (E1)
FROM พนักงาน (ชื่อ, เงินเดือน) (PERIOD) AS E1 E2
WHERE E1.ชื่อ = 'สมหญิง' AND E2.ชื่อ = 'สมชาย' AND E1.เงินเดือน < E2.เงินเดือน

```

8.3.2.31 บอกช่วงเวลาที่มีสมชายทำงานอยู่ในแผนกของเล่น โดยมีสมหญิงเป็นผู้จัดการ
(คำตอบ : [02/01/1982 – 01/31/1987])

```

SELECT SNAPSHOT INTERSECT (VALID (VALID (E1)), VALID (VALID (E2)))
FROM พนักงาน (รหัส, ชื่อ, เงินเดือน, ชื่อแผนก) AS E1 E2, แผนก (รหัส, ชื่อแผนก) AS D1
WHERE E1.ชื่อ = 'สมหญิง' AND E2.ชื่อ = 'สมชาย'
AND E1.รหัส = D1.รหัส AND D1.ชื่อแผนก = 'ของเล่น' AND E2.ชื่อแผนก = 'ของเล่น'
AND VALID (D1) OVERLAPS VALID (E2)

```

8.3.2.32 บอกช่วงเวลาที่มีพนักงานที่ปัจจุบันชื่อชาติชายมีทักษะขับรถ
(คำตอบ : [01/01/1982 – 05/01/1982], [06/01/1984 – 05/31/1988])

```

SELECT VALID (S)
FROM พนักงาน (รหัส, ชื่อ) AS E, ทักษะ (รหัส, ทักษะ) (PERIOD) AS S
WHERE E.ชื่อ = 'ชาติชาย' AND E.รหัส = S.รหัส AND S.ทักษะ = 'ขับรถ'

```

8.3.3 คำตอบเป็นค่าของแอททริบิวต์ที่มีวาติคใหม่อยู่ด้วย

8.3.3.1 พนักงานคนใดทำงานติดต่อกันอยู่ในแผนกของเล่น เป็นเวลานานน้อยกว่าที่สมหญิง
ทำงานติดต่อกันในแผนกของเล่น และในช่วงเวลาใด
(คำตอบ : ('สมชาย' | [02/01/1982 – 01/31/1987]))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT E2.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, E1 (ชื่อ) AS E2,
พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E3
WHERE E3.รหัส = (SELECT รหัส FROM พนักงาน
WHERE ชื่อ = 'สมหญิง' AND VALID (พนักงาน) CONTAINS CURRENT_DATE)
AND E1.ชื่อแผนก = 'ของเล่น' AND E3.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E3) AS INTERVAL DAY)

```

- 8.3.3.2 พนักงานคนใดทำงานติดต่อกันอยู่ในแผนกของเล่น เป็นเวลานานน้อยกว่าที่สมหญิงทำงานติดต่อกันในแผนกของเล่น และในช่วงเวลาใด บอกละแวกและวันเกิดด้วย (คำตอบ : ('สมชาย', 'ชาย', 07/01/1955 | [02/01/1982 – 01/31/1987]))

```

SELECT E2.ชื่อ, E2.เพศ, E2.วันเกิด
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, E1 (ชื่อ, เพศ, วันเกิด) AS E2,
พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E3
WHERE E3.รหัส = (SELECT รหัส FROM พนักงาน
WHERE ชื่อ = 'สมหญิง' AND VALID (พนักงาน) CONTAINS CURRENT_DATE)
AND E1.ชื่อแผนก = 'ของเล่น' AND E3.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E3) AS INTERVAL DAY)

```

- 8.3.3.3 บอกละแวกเริ่มต้นที่มีพนักงานคนใดทำงานติดต่อกันอยู่ในแผนกของเล่น เป็นเวลานานน้อยกว่าที่สมหญิงทำงานติดต่อกันในแผนกของเล่น (คำตอบ : ('สมชาย', 02/01/1982))

```

SELECT SNAPSHOT E2.ชื่อ, BEGIN (E2)
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, E1 (ชื่อ) AS E2,
พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E3
WHERE E3.รหัส = (SELECT รหัส FROM พนักงาน
WHERE ชื่อ = 'สมหญิง' AND VALID (พนักงาน) CONTAINS CURRENT_DATE)
AND E1.ชื่อแผนก = 'ของเล่น' AND E3.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E3) AS INTERVAL DAY)

```

- 8.3.3.4 บอกละแวกพนักงานในวันที่ 1 เดือนมกราคม ปี ค.ศ. 1984 ที่ทำงานติดต่อกันอยู่ในแผนกของเล่น เป็นเวลานานน้อยกว่าที่สมหญิงทำงานติดต่อกันในแผนกของเล่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(คำตอบ : ('สมชาย', 01/01/1984))

```
SELECT SNAPSHOT E2.ชื่อ, DATE '01/01/1984'
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, E1 (ชื่อ) AS E2,
พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E3
WHERE E3.รหัส = (SELECT รหัส FROM พนักงาน
WHERE ชื่อ = 'สมหญิง' AND VALID (พนักงาน) CONTAINS CURRENT_DATE)
AND E1.ชื่อแผนก = 'ของเล่น' AND E3.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E3) AS INTERVAL DAY)
AND VALID (E2) OVERLAPS DATE '01/01/1984'
```

8.3.3.5 ช่วงเวลาใดที่งบประมาณของแผนกของเล่นมากกว่า 175000 บาทเป็นเวลายาวนานมากกว่า 1 ปี และงบประมาณนั้นเท่ากับเท่าไร

(คำตอบ : (200000 | [08/01/1984 – 12/31/1986]))

```
SELECT D1.งบประมาณ
FROM แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D1
WHERE D1.ชื่อแผนก = 'ของเล่น' AND D1.งบประมาณ > 175000
AND CAST (VALID (D1) AS INTERVAL YEAR) > INTERVAL '1' YEAR
```

8.3.3.6 ช่วงเวลาใดที่งบประมาณของแผนกของเล่นมากกว่า 175000 บาทเป็นเวลายาวนานมากกว่า 1 ปี และใครเป็นผู้จัดการอยู่ตอนนั้น

(คำตอบ : ('สมหญิง' | [08/01/1984 – 12/31/1986]))

```
SELECT E1.ชื่อ, VALID (D1)
FROM แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D1, D1(รหัส) AS D2, พนักงาน AS E1
WHERE D1.ชื่อแผนก = 'ของเล่น'
AND CAST (VALID (D1) AS INTERVAL DAY) > INTERVAL '1' YEAR
AND D1.งบประมาณ > 175000 AND D2.รหัส = E1.รหัส
```

8.3.3.7 ใครเป็นผู้จัดการของแผนกที่มีงบประมาณมากกว่า 175000 บาทเป็นเวลายาวนานมากกว่า 1 ปี และในช่วงเวลาใด

(คำตอบ : ('สมหญิง' | [08/01/1984 – 12/31/1986]))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT E1.ชื่อ, VALID (D1)
FROM แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D1, D1(รหัส) AS D2, พนักงาน AS E1
WHERE CAST (VALID (D1) AS INTERVAL DAY) > INTERVAL '1' YEAR
AND D1.งบประมาณ > 175000 AND D2.รหัส = E1.รหัส

```

8.3.3.8 แผนกใดมีช่วงเวลาทำงานนานกว่าช่วงเวลาระหว่างวันเกิดของสมชายและสมหญิง และในช่วงเวลาใด (คำตอบ : ('ของเล่น' | [01/01/1982 – 01/01/1990]))

```

SELECT D1.ชื่อแผนก
FROM แผนก (ชื่อแผนก) (PERIOD) AS D1
WHERE CAST (VALID (D1) AS INTERVAL DAY) > (LAST (SELECT วันเกิด FROM พนักงาน
WHERE ชื่อ = 'ชาติชาย', SELECT วันเกิด FROM พนักงาน WHERE ชื่อ = 'สมหญิง') -
FIRST (SELECT วันเกิด FROM พนักงาน WHERE ชื่อ = 'ชาติชาย',
SELECT วันเกิด FROM พนักงาน WHERE ชื่อ = 'สมหญิง')) DAY)

```

8.3.3.9 ใครทำงานในแผนกเดิมเป็นเวลายาวน้อย 2 ปีติดต่อกัน และในช่วงเวลาใด (คำตอบ : ('สมชาย' | [01/01/1982 – 01/31/1987]) ('สมชาย' | [04/01/1987 – 12/31/1987]) ('ชาติชาย' | [01/01/1988 – 01/01/1990]) และ ('สมหญิง' | [01/01/1982 – 01/01/1990]))

```

SELECT E2.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) (PERIOD) AS E1, พนักงาน AS E2
WHERE CAST (VALID (E1) AS INTERVAL YEAR) > INTERVAL '2' YEAR
AND E1.รหัส = E2.รหัส AND E1.ชื่อแผนก = E2.ชื่อแผนก

```

8.3.3.10 ใครและเมื่อใดที่มีแผนกที่พนักงานทำงานเท่ากับช่วงเวลาที่แผนกนั้นมีงบประมาณน้อยกว่า 200000 บาท (คำตอบ : ('สมหญิง', 'ของเล่น' | [01/01/1982 – 01/01/1990]) และ ('สมชาย', 'หนังสือ' | [04/01/1987 – 01/01/1990]))

```

SELECT E1.ชื่อ, E1.ชื่อแผนก
FROM พนักงาน (ชื่อ, ชื่อแผนก) AS E1, แผนก (ชื่อแผนก, งบประมาณ) AS D1
WHERE D1.ชื่อแผนก = E1.ชื่อแผนก AND D1.งบประมาณ < 200000
AND CAST (VALID (E1) AS INTERVAL DAY) >= CAST (VALID (D1) AS INTERVAL DAY)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.3.11 ใครและเมื่อใดที่มีพนักงานทำงานอยู่ในแผนกนานกว่าที่ผู้จัดการทำในแผนกนั้น
(คำตอบ : ไม่มี)

```
SELECT E4.ชื่อ
FROM พนักงาน AS E1, แผนก AS D1, พนักงาน (รหัส, ชื่อแผนก) AS E2,
พนักงาน (รหัส, ชื่อแผนก) AS E3, E3 (ชื่อ) AS E4
WHERE VALID (E1) OVERLAPS CURRENT_DATE
AND VALID (D1) OVERLAPS CURRENT_DATE
AND E1.ชื่อแผนก = D1.ชื่อแผนก AND D1.รหัส = E2.รหัส AND E3.รหัส = E1.รหัส
AND E2.ชื่อแผนก = E3.ชื่อแผนก
AND CAST (VALID (E3) AS INTERVAL DAY) > CAST (VALID (E2) AS INTERVAL DAY)
```

8.3.3.12 บอกชื่อพนักงาน เพศ แผนก และเงินเดือนทั้งหมดของพนักงานที่เป็นผู้จัดการของ
แผนกใดๆเป็นเวลาอย่างน้อยเท่ากับที่สมหญิงเป็นผู้จัดการของแผนกของเล่น
(คำตอบ : ('สมหญิง', 'หญิง', 'ของเล่น', 30000 | [01/01/1982 – 07/01/1984]) ('สม
หญิง', 'หญิง', 'ของเล่น', 40000 | [08/01/1984 – 08/31/1986]) และ ('สมหญิง', 'หญิง',
'ของเล่น', 50000 | [09/01/1986 – 01/01/1990]))

```
SELECT E1.ชื่อ, E1.เพศ, E1.ชื่อแผนก, E1.เงินเดือน
FROM พนักงาน AS E1, แผนก (ชื่อแผนก, รหัส) AS D1
WHERE E1.รหัส = D1.รหัส
AND CAST (VALID (D1) AS INTERVAL DAY) >=
(SELECT SNAPSHOT CAST (VALID (D2) AS INTERVAL DAY)
FROM แผนก(ชื่อแผนก, รหัส) AS D2
WHERE D2.ชื่อแผนก = 'ของเล่น' AND D2.รหัส = (SELECT รหัส FROM พนักงาน
WHERE พนักงาน.ชื่อ = 'สมหญิง' AND VALID (พนักงาน) OVERLAPS CURRENT_DATE))
```

8.3.3.13 บอกชื่อแผนก ชื่อผู้จัดการ และช่วงเวลาทำงานของแผนกที่มีผู้จัดการที่ทำงานเป็นเวลา
สั้นที่สุด (คำตอบ : ('หนังสือ', 'สมชาย', [04/01/1987 – 12/31/1987]) และ ('หนังสือ',
'ชาติชาย', [01/01/1988 – 01/01/1990]))

```
SELECT D1.ชื่อแผนก, E1.ชื่อ
FROM แผนก (ชื่อแผนก, รหัส) AS D1, พนักงาน AS E1
WHERE E1.รหัส = D1.รหัส
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND CAST (VALID (D1) AS INTERVAL DAY) =
 (SELECT MIN (CAST (VALID (D3) AS INTERVAL DAY)) FROM แผนก (รหัส) AS D3)

8.3.3.14 บอกรหัสแผนก งบประมาณ และช่วงเวลาที่แผนกมีงบประมาณอย่างน้อย 200000 บาท
 เป็นเวลานานกว่าที่มีงบประมาณน้อยกว่า 200000 บาท (คำตอบ : ไม่มี)

SELECT D1.ชื่อแผนก, D1.งบประมาณ
 FROM แผนก (ชื่อแผนก, งบประมาณ) AS D1
 WHERE D1.งบประมาณ >= 200000
 AND (SELECT SUM (CAST (VALID (D2) AS INTERVAL DAY))
 FROM แผนก (ชื่อแผนก, งบประมาณ) AS D2
 WHERE D2.งบประมาณ >= 200000 AND D2.ชื่อแผนก = D1.ชื่อแผนก) >
 (SELECT SUM (CAST (VALID (D3) AS INTERVAL DAY))
 FROM แผนก (ชื่อแผนก, งบประมาณ) AS D3
 WHERE D3.งบประมาณ < 200000 AND D3.ชื่อแผนก = D1.ชื่อแผนก)

8.3.3.15 ทักษะใดที่สมชายมีเป็นเวลอย่างน้อยเท่ากับเวลาทำงานของสมชาย ในขณะที่ไม่มี
 ทักษะขับรถ และในช่วงเวลาใด (คำตอบ : ('พิมพ์ดีด' | [04/01/1982 – 01/01/1990])
 ('จัดเรียงเอกสาร' | [01/01/1985 – 01/01/1990]) และ ('ขับรถ' | [01/01/1982 –
 05/01/1982] ∪ [06/01/1984 – 05/31/1988]))

SELECT S1.ทักษะ
 FROM ทักษะ AS S1, พนักงาน AS E1
 WHERE S1.รหัส = E1.รหัส AND E1.ชื่อ = 'ชาติชาย'
 AND CAST (VALID (S1) AS INTERVAL DAY) >=
 (SELECT SNAPSHOT (VALID (E2) – VALID (S2)) DAY)
 FROM พนักงาน (รหัส) AS E2, ทักษะ AS S2
 WHERE E2.รหัส = E1.รหัส AND S2.ทักษะ = 'ขับรถ' AND S2.รหัส = E1.รหัส)

8.3.3.16 บอกรหัสของพนักงานที่อายุมากที่สุดที่มีทักษะพิมพ์ดีดในวันที่ 31 เดือนธันวาคม ปี ค.ศ.
 1985 และช่วงเวลาด้วย (คำตอบ : ('สมชาย' | [04/01/1982 – 12/31/1985]) และ ('ชาติ
 ชาย' | [04/01/1982 – 12/31/1985]))

SELECT E1.ชื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

VALID (S1) INTERSECT PERIOD '[BEGINNING - 12/31/1985]'
FROM พนักงาน AS E1
WHERE E1.วันเกิด = (SELECT MIN (E2.วันเกิด) FROM ทักษะ AS S1, พนักงาน AS E2
WHERE VALID (S1) OVERLAPS DATE '12/31/1985'
AND S1.ทักษะ = 'พิมพ์ดีด' AND S1.รหัส = E2.รหัส) AND VALID(E1) INTERSECT VALID (S1)

```

8.3.3.17 บอกชื่อพนักงานในวันที่ 1 เดือนมกราคม ปี ค.ศ. 1985 ของพนักงานที่ทำงานในแผนกของเล่นน้อยกว่าเวลาที่สมหญิงทำงานในแผนกนั้นจนถึงวันที่ 1 เดือนมกราคม ปี ค.ศ. 1985 และผลต่างของเวลาดังกล่าว (คำตอบ : ('สมชาย', 1 เดือน))

```

SELECT E1.ชื่อ, E3.ชื่อ
FROM (SELECT E1.ชื่อ VALID INTERSECT PERIOD '[BEGINNING - 01/01/1985]'
FROM พนักงาน AS E1 WHERE E1.แผนก = 'ของเล่น') AS E1 E2, พนักงาน AS E3
WHERE E2.ชื่อ = 'สมหญิง' AND E1.รหัส = E3.รหัส
AND CAST (VALID (E1) AS INTERVAL DAY) < CAST (VALID (E2) AS INTERVAL DAY)
AND VALID (E3) CONTAINS DATE '01/01/1985'

```

8.3.3.18 บอกชื่อพนักงานที่ทำงานเป็นเวลานานที่สุดในปี ค.ศ. 1987 และช่วงเวลาดังกล่าว (คำตอบ : ('สมชาย' | [01/01/1987 – 01/31/1987]) และ ('สมชาย' | [04/01/1987 – 12/31/1987]))

```

SELECT E.ชื่อ
FROM พนักงาน AS E, (SELECT ชื่อ VALID INTERSECT PERIOD '[1987]' FROM พนักงาน) AS E2
WHERE VALID (E) CONTAINS CURRENT_DATE AND E.รหัส = E2.รหัส
AND CAST (VALID (E2) AS INTERVAL DAY) =
(SELECT MIN (CAST (VALID (E3) AS INTERVAL DAY))
FROM (SELECT ชื่อ VALID INTERSECT PERIOD '[1987]' FROM พนักงาน) AS E3)

```

8.3.3.19 บอกชื่อและอายุของพนักงานที่ได้ขึ้นเงินเดือนครั้งแรก (คำตอบ : ('สมชาย', 26 ปี 11 เดือน) และ ('สมหญิง', 23 ปี 10 เดือน))

```

SELECT A.ชื่อ, MIN(A.Age)
FROM (SELECT E1.ชื่อ, BEGIN (E2) - E1.วันเกิด AS Age
VALID BEGIN (E2)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FROM พนักงาน AS E1, พนักงาน AS E2
WHERE E1.รหัส = E2.รหัส AND E1.เงินเดือน <> E2.เงินเดือน
AND VALID (E1) MEETS VALID (E2)) AS A
GROUP BY A.ชื่อ
```

8.3.3.20 บอกชื่อและเงินเดือนจนถึงอายุ 25 ปีของพนักงานที่เป็นผู้หญิง

(คำตอบ : ('สมหญิง', 30000 | [01/01/1982 – 07/31/1984]) และ ('สมหญิง', 40000 | [08/01/1984 – 10/01/1985]))

```
SELECT พนักงาน.ชื่อ, พนักงาน.เงินเดือน
VALID INTERSECT PERIOD (DATE 'BEGINNING', พนักงาน.วันเกิด + INTERVAL '25' YEAR)
FROM พนักงาน
WHERE พนักงาน.เพศ = 'หญิง'
AND (พนักงาน PRECEDES (พนักงาน.วันเกิด + INTERVAL '25' YEAR)
OR VALID (พนักงาน) CONTAINS (พนักงาน.วันเกิด + INTERVAL '25' YEAR))
```

8.3.3.21 โครและเมื่อใดที่มีพนักงานเปลี่ยนชื่อ (คำตอบ : ('สมชาย' | 01/01/1988))

```
SELECT E1.ชื่อ
VALID BEGIN (E2)
FROM พนักงาน AS E1, พนักงาน AS E2
WHERE E1.รหัส = E2.รหัส AND E1.ชื่อ <> E2.ชื่อ AND VALID (E1) MEETS VALID (E2)
```

8.3.3.22 บอกอายุของสมชายและบอกทักษะที่สมชายมีในขณะที่เปลี่ยนชื่อเป็นชาติชาย

(คำตอบ : (32.5 ปี, 'ขับรถ') (32.5 ปี, 'จัดเรียงเอกสาร') และ (32.5 ปี, 'พิมพ์ดีด'))

```
SELECT SNAPSHOT BEGIN (Emp2) - Emp1.วันเกิด AS Age, ทักษะ.ทักษะ
FROM พนักงาน AS Emp1, พนักงาน AS Emp2, ทักษะ
WHERE Emp1.ชื่อ = 'สมชาย' AND Emp2.ชื่อ = 'ชาติชาย'
AND Emp1.รหัส = Emp2.รหัส AND VALID (Emp1) MEETS VALID (Emp2)
AND Emp2.รหัส = ทักษะ.รหัส AND VALID (ทักษะ) CONTAINS BEGIN (VALID (Emp2))
```

8.3.3.23 เมื่อใดที่สมชายมีทักษะขับรถ และสมชายมีทักษะอื่นใดในเวลานั้น

(คำตอบ : ('พิมพ์ดีด' | 06/01/1984))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT S1.ทักษะ
VALID BEGIN (S2)
FROM พนักงาน, ทักษะ AS S1, ทักษะ AS S2
WHERE พนักงาน.ชื่อ = 'สมชาย' AND พนักงาน.รหัส = S1.รหัส
AND S1.รหัส = S2.รหัส AND S2.ทักษะ = 'ขับรถ' AND VALID (พนักงาน) OVERLAPS VALID (S1)
AND VALID (S1) CONTAINS BEGIN (VALID (S2))

```

8.3.3.24 ใครเป็นผู้จัดการที่เป็นผู้หญิงคนแรกของแผนกของเล่น และเมื่อใด
(คำตอบ : ('สมหญิง' | 01/01/1982))

```

SELECT BEGIN (F), F.ชื่อ
FROM (SELECT * FROM พนักงาน AS E1
WHERE E1.เพศ = 'หญิง' AND E1.ชื่อแผนก = 'ของเล่น') AS F
WHERE BEGIN (VALID (F)) = (SELECT MIN (BEGIN (T)) FROM F AS T)

```

8.3.3.25 บอกรหัสและเงินเดือนทั้งหมดของพนักงานที่มีวันเกิดหลังวันที่ 1 เดือนมกราคม ปี ค.ศ.
1956 (คำตอบ : ('สมหญิง', 30000 | [01/01/1982 - 07/31/1984]) ('สมหญิง', 40000 |
[08/01/1984 - 08/31/1986]) และ ('สมหญิง', 50000 | [08/01/1986 - 01/01/1990]))

```

SELECT E1.ชื่อ, E1.เงินเดือน
FROM พนักงาน AS E1
WHERE E1.วันเกิด > DATE '01/01/1956'

```

8.3.3.26 บอกรหัสและเงินเดือนทั้งหมดของพนักงานชื่อสมชายหลังวันที่ 1 เดือนมกราคม ปี ค.ศ.
1988 (คำตอบ : ไม่มี)

```

SELECT E1.ชื่อ, E1.เงินเดือน
FROM พนักงาน AS E1
WHERE DATE '01/01/1988' PRECEDES VALID (E1) AND E1.ชื่อ = 'สมชาย'

```

8.3.3.27 บอกช่วงเวลาและเงินเดือนของพนักงานที่ทำงานอยู่ในแผนกของเล่นในช่วงเวลาก่อนที่
สมชายจะทำงานที่นั่น (คำตอบ : (30000 | [01/01/1982 - 01/31/1985]))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT Emp1.เงินเดือน
VALID E1 – PERIOD (BEGIN (E2), DATE 'FOREVER')
FROM พนักงาน AS Emp1, พนักงาน (ชื่อ, ชื่อแผนก) AS Emp2
WHERE Emp2.ชื่อ = 'สมชาย' AND Emp2.ชื่อแผนก = 'ของเล่น' AND Emp1.ชื่อแผนก = 'ของเล่น'
AND BEGIN (VALID (Emp1)) PRECEDES BEGIN (VALID (Emp2))

```

8.3.3.28 บอกรายชื่อพนักงานที่มากที่สุดที่น้อยกว่า 50000 บาทที่สมชายได้และช่วงเวลาด้วย

(คำตอบ : (40000 | [02/01/1985 – 01/31/1987]) และ (40000 | [04/01/1987 – 01/01/1990]))

```

SELECT MAX (Emp1.เงินเดือน)
FROM พนักงาน AS Emp1, พนักงาน AS Emp2
WHERE Emp2.ชื่อ = 'สมชาย' AND Emp1.รหัส = Emp2.รหัส
GROUP BY Emp1.รหัส
HAVING MAX (Emp1.เงินเดือน) < 50000

```

8.3.3.29 บอกรายชื่อพนักงานทั้งหมดของสมชาย (คำตอบ : (20000 | [02/01/1982 – 05/31/1982]) (30000 | 06/01/01982 – 01/31/1985] และ (40000 | [02/01/1985 – 01/31/1987] ∪ [04/01/1987 – 01/01/1990]))

```

SELECT E1.เงินเดือน
FROM พนักงาน AS E1, พนักงาน AS E2
WHERE E2.ชื่อ = 'สมชาย' AND E1.รหัส = E2.รหัส AND VALID (E1) OVERLAPS VALID (E2)

```

8.3.3.30 บอกรายชื่อ พนักงาน เงินเดือน และช่วงเวลาที่มีพนักงานมีทักษะขับรถและมีเงินเดือนน้อยกว่า 40000 บาท (คำตอบ : ('สมชาย', 20000 | [02/01/1982 – 05/01/1982]) และ ('สมชาย', 30000 | [06/01/1984 – 01/31/1985]))

```

SELECT พนักงาน.ชื่อ, พนักงาน.เงินเดือน
FROM พนักงาน, ทักษะ
WHERE พนักงาน.รหัส = ทักษะ.รหัส AND ทักษะ.ทักษะ = 'ขับรถ' AND พนักงาน.เงินเดือน < 40000
AND VALID (พนักงาน) OVERLAPS VALID (ทักษะ)

```

8.3.3.31 บอกรายชื่อพนักงานและงบประมาณของแผนกของเล่นในขณะที่สมชายทำงานที่นั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ค่าตอบ : (150000 | [02/01/1982 – 07/31/1984]) (200000 | [08/01/1984 – 12/31/1986])
และ (100000 | [01/01/1987 – 01/31/1987]))

```
SELECT แผนก.งบประมาณ
FROM พนักงาน, แผนก
WHERE แผนก.ชื่อแผนก = 'ของเล่น' AND แผนก.ชื่อแผนก = พนักงาน.ชื่อแผนก
AND พนักงาน.ชื่อ = 'สมชาย' AND VALID (พนักงาน) OVERLAPS VALID (แผนก)
```

8.3.3.32 บอกชื่อปัจจุบัน ทักษะ และช่วงเวลาของพนักงานที่เคยทำงานในแผนกหนังสือหรือแผนกของเล่นใน 2 ปีที่ผ่านมา (ค่าตอบ : ('ชาติชาย', 'พิมพ์ดีด' | [04/01/1982 – 01/01/1990]) ('ชาติชาย', 'จัดเรียงเอกสาร' | [01/01/1985 – 01/01/1990]) ('ชาติชาย', 'ขับรถ' | [01/01/1982 – 05/01/1982]) ('ชาติชาย', 'ขับรถ' | [06/01/1984 – 05/31/1988]) และ ('สมหญิง', 'บริหารงาน' | [01/01/1982 – 01/01/1990]))

```
SELECT E2.ชื่อ, S.ทักษะ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1, พนักงาน (รหัส, ชื่อ) AS E2, ทักษะ (รหัส, ทักษะ) AS S
WHERE E1.รหัส = S.รหัส AND E1.รหัส = E2.รหัส
AND VALID (E1) OVERLAPS PERIOD '[NOW-2 – NOW]' YEAR
AND VALID (E2) CONTAINS CURRENT_DATE
```

8.3.3.33 บอกชื่อปัจจุบันของพนักงานที่เคยเป็นลูกน้องของสมหญิงในปีที่ผ่านมา และช่วงเวลาด้วย (ค่าตอบ : ('ชาติชาย' | [02/01/1982 – 01/31/1987]))

```
SELECT E2.ชื่อ
FROM พนักงาน (รหัส, ชื่อแผนก) AS E1, พนักงาน (รหัส, ชื่อ) AS E2, พนักงาน (รหัส, ชื่อ) AS E3,
แผนก (ชื่อแผนก, รหัส) AS D
WHERE E1.รหัส = E2.รหัส AND VALID (E2) CONTAINS CURRENT_DATE
AND E1.ชื่อแผนก = D.ชื่อแผนก
AND E1.รหัส <> D.รหัส AND E3.รหัส = D.รหัส AND E3.ชื่อ = 'สมหญิง'
AND VALID (E1) PRECEDES PERIOD '[NOW-1 – NOW]' YEAR
```

8.3.3.34 สำหรับพนักงานที่มีทักษะใดๆ ในช่วงปี ค.ศ. 1983 ถึงปี ค.ศ. 1987 บอกชื่อผู้จัดการ และวันที่มีทักษะนั้น (ค่าตอบ : ('สมหญิง', 06/01/1984) และ ('สมหญิง', 01/01/1985))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT SNAPSHOT E1.ชื่อ, BEGIN (S)
FROM พนักงาน (รหัส, ชื่อ) AS E1, พนักงาน (รหัส, ชื่อแผนก) AS E2,
ทักษะ (รหัส, ทักษะ) (PERIOD) AS S, แผนก (ชื่อแผนก, รหัส) AS D
WHERE DATE '01/01/1983' PRECEDES BEGIN (VALID (S))
AND BEGIN (VALID (S)) PRECEDES DATE '01/01/1987' AND S.รหัส = E2.รหัส
AND E2.ชื่อแผนก = D.ชื่อแผนก AND D.รหัส = E1.รหัส

```

8.3.3.35 บอกชื่อปัจจุบัน และวันที่ได้ขึ้นเงินเดือนของพนักงานที่ได้ขึ้นเงินเดือนในช่วงเดือน
มีนาคม ปี ค.ศ. 1982 ถึงเดือนมีนาคม ปี ค.ศ. 1985 (คำตอบ : ('ชาติชาย', 06/01/1982)
และ ('ชาติชาย', 02/01/1985))

```

SELECT SNAPSHOT E2.ชื่อ, BEGIN (E1)
FROM พนักงาน (รหัส, เงินเดือน) PERIOD AS E1, พนักงาน (รหัส, ชื่อ) AS E2
WHERE E1.รหัส = E2.รหัส AND VALID (E2) CONTAINS CURRENT_DATE
AND DATE '03/01/1982' PRECEDES BEGIN (VALID (E1))
AND BEGIN (VALID (E1)) PRECEDES DATE '03/01/1985'
AND 2 <= (SELECT SNAPSHOT COUNT (*))
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E3
WHERE E3.รหัส = E1.รหัส AND E3.เงินเดือน > E1.เงินเดือน
AND VALID (E1) PRECEDES VALID (E3)
AND BEGIN (VALID (E3)) PRECEDES DATE '03/01/1985'

```

8.3.3.36 บอกเงินเดือนทั้งหมดของสมชายในขณะที่สมชายมีทักษะขับรถ (คำตอบ : (20000 |
[02/01/1982 – 05/01/1982]) (30000 | [06/01/1984 – 01/31/1985]) และ (40000 |
[02/01/1985 – 01/31/1987] \cup [04/01/1987 – 12/31/1987]))

```

SELECT เงินเดือน
VALID INTERSECT (พนักงาน, ทักษะ)
FROM พนักงาน, ทักษะ
WHERE พนักงาน.ชื่อ = 'สมชาย' AND ทักษะ.รหัส = พนักงาน.รหัส AND ทักษะ.ทักษะ = 'ขับรถ'
AND VALID (พนักงาน) OVERLAPS VALID (ทักษะ)

```

8.3.3.37 บอกเงินเดือนทั้งหมดในช่วงเวลาที่สมชายมีทักษะขับรถ ของพนักงานที่ได้เงินเดือน

น้อยกว่า 50000 บาทตลอดปี ค.ศ. 1989 (คำตอบ : (20000 | [02/01/1982 – 05/01/1982]))
เอกสารนี้เป็นเอกสารที่สงวนเวลาสำหรับการแข่งขันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(30000 | [06/01/1984 – 01/31/1985]) (40000 | [02/01/1985 – 01/31/1987]) และ (40000 | [04/01/1987 – 05/31/1988]))

```
SELECT E1.เงินเดือน
VALID INTERSECT (E1, ทักษะ)
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E1 E2, ทักษะ
WHERE E1.รหัส = E2.รหัส AND E2.เงินเดือน < 50000
AND VALID (E2) CONTAINS PERIOD '[01/01/1989 - 12/31/1989]'
AND ทักษะ.ทักษะ = 'ขับรถ' AND ทักษะ.รหัส = E1.รหัส AND E1.ชื่อ = 'สมชาย'
AND VALID (E1) OVERLAPS VALID (ทักษะ)
```

8.3.3.38 บอกชื่อและเงินเดือนทั้งหมดของพนักงานผู้ชายที่เป็นลูกน้องของพนักงานผู้หญิง (คำตอบ : ('สมชาย', 20000 | [02/01/1982 – 05/31/1982]) ('สมชาย', 30000 | [06/01/1982 – 01/31/1985]) และ ('สมชาย', 40000 | [02/01/1985 – 01/31/1987]))

```
SELECT E.ชื่อ, E.เงินเดือน
VALID INTERSECT (E, แผนก)
FROM พนักงาน (รหัส, ชื่อ, เงินเดือน) AS E, พนักงาน AS Mgr, แผนก (ชื่อแผนก, รหัส) AS D
WHERE E.เพศ = 'ชาย' AND E.ชื่อแผนก = แผนก.ชื่อแผนก
AND D.รหัส = Mgr.รหัส AND Mgr.เพศ = 'หญิง' AND VALID (E) OVERLAPS VALID (D)
```

8.3.3.39 บอกชื่อแผนก ชื่อผู้จัดการปัจจุบัน และช่วงเวลาที่ผู้จัดการแผนกได้เงินเดือนมากกว่า 1 ใน 3 ของงบประมาณแผนก (คำตอบ : 'ของเล่น', 'สมหญิง' | [01/01/1987 – 01/01/1990]) และ ('หนังสือ', 'ชาติชาย' | [04/01/1987 – 01/01/1990]))

```
SELECT แผนก.ชื่อ, CMgr.ชื่อ
VALID INTERSECT (แผนก, Mgr)
FROM แผนก, พนักงาน (รหัส, เงินเดือน, ชื่อแผนก) AS Mgr, Mgr(ชื่อ) AS CMgr
WHERE แผนก.รหัส = Mgr.รหัส AND (แผนก.งบประมาณ / 3) < Mgr.เงินเดือน
AND VALID (แผนก) OVERLAPS VALID (Mgr)
AND VALID (CMgr) CONTAINS CURRENT_DATE
```

8.3.3.40 บอกชื่อ เงินเดือน และช่วงเวลาที่พนักงานได้เงินเดือนมากกว่าผู้จัดการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(คำตอบ : ('สมชาย', 30000 | [06/01/1982 – 07/31/1984]) และ ('สมชาย', 40000 | 02/01/1985 – 08/31/1986)))

```
SELECT E.ชื่อ, E.เงินเดือน
FROM พนักงาน (รหัส, ชื่อ, เงินเดือน) AS E, พนักงาน (รหัส, เงินเดือน) AS Mgr, แผนก
WHERE E.ชื่อแผนก = แผนก.ชื่อแผนก AND VALID (E) OVERLAPS VALID (แผนก)
AND แผนก.รหัส = Mgr.รหัส AND VALID (แผนก) OVERLAPS VALID (Mgr)
AND E.เงินเดือน = Mgr.เงินเดือน AND E.รหัส <> Mgr.รหัส
AND VALID (E) OVERLAPS VALID (Mgr)
```

8.3.3.41 บอกชื่อพนักงานที่ได้เงินเดือนน้อยกว่าพนักงานที่อายุน้อยกว่า (คำตอบ : 'สมชาย', 'สมหญิง' | [02/01/1982 – 07/31/1982]) ('สมชาย', 'สมหญิง' | [08/01/1984 – 01/31/1985]) และ ('สมชาย', 'สมหญิง' | [09/01/1986 – 01/01/1990]))

```
SELECT Emp1.รหัส, Emp2.รหัส
FROM พนักงาน (รหัส, เงินเดือน) AS Emp1 Emp2
WHERE Emp1.เงินเดือน < Emp2.เงินเดือน AND Emp1.วันเกิด < Emp2.วันเกิด
AND VALID (Emp1) OVERLAPS VALID (Emp2)
```

8.3.3.42 ใครมีเงินเดือนเท่าเดิมเป็นเวลายาวนานที่สุด (คำตอบ : ('สมหญิง' | [09/01/1986 – 01/01/1990]))

```
SELECT รหัส
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E
WHERE CAST (VALID (E) AS INTERVAL DAY) >=
ALL (SELECT SNAPSHOT CAST (VALID (E) AS INTERVAL DAY)
FROM พนักงาน (รหัส, เงินเดือน) (PERIOD) AS E)
```

8.3.3.43 บอกทักษะและเงินเดือนของสมหญิงในขณะที่สมหญิงเป็นผู้จัดการ (คำตอบ : ('บริหารงาน', 30000 | [01/01/1982 – 07/31/1984]) ('บริหารงาน', 40000 | [08/01/1984 – 08/31/1986]) และ ('บริหารงาน', 50000 | [09/01/1986 – 01/01/1990]))

```
SELECT ทักษะ.ทักษะ, พนักงาน.เงินเดือน
```

```
FROM พนักงาน, แผนก, ทักษะ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WHERE พนักงาน.ชื่อ = 'สมหญิง' AND แผนก.รหัส = พนักงาน.รหัส
 AND VALID (แผนก) OVERLAPS VALID (พนักงาน)
 AND ทักษะ.รหัส = พนักงาน.รหัส AND VALID (ทักษะ) OVERLAPS VALID (พนักงาน)
 AND VALID (แผนก) OVERLAPS VALID (ทักษะ)

8.3.3.44 บอกชื่อ และเงินเดือนของพนักงานที่เคยเป็นผู้จัดการและได้เงินเดือนอย่างน้อย 36000 บาท (คำตอบ : ('สมหญิง', 40000 | [08/01/1984 – 08/31/1986]) ('สมหญิง', 50000 | [09/01/1986 – 01/01/1990]) ('สมชาย', 40000 | [04/01/1987 – 12/31/1987]) และ ('ชาติชาย', 40000 | [01/01/1988 – 01/01/1990]))

SELECT พนักงาน.ชื่อ, พนักงาน.เงินเดือน
 FROM พนักงาน, แผนก
 WHERE พนักงาน.เงินเดือน > 36000 AND พนักงาน.รหัส = แผนก.รหัส
 AND VALID (พนักงาน) OVERLAPS VALID (แผนก)

8.3.3.45 บอกงบประมาณตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1982 ถึงวันที่ 31 เดือนมกราคม ปี ค.ศ. 1984 และตั้งแต่วันที่ 1 เดือนมกราคม ปี ค.ศ. 1987 ถึงปัจจุบันของทุกแผนกที่สมชายเคยทำงาน (คำตอบ : ('ของเล่น', 150000 | [02/01/1982 – 07/31/1984]) ('ของเล่น', 200000 | [08/01/1984 – 12/31/1984]) ('ของเล่น', 100000 | [01/01/1987 – 01/01/1990]) และ ('หนังสือ', 50000 | [04/01/1987 – 01/01/1990]))

SELECT D.งบประมาณ
 VALID PERIOD '01/01/1982 - 12/31/1984' + PERIOD '01/01/1987 - NOW'
 FROM พนักงาน (รหัส, ชื่อ, ชื่อแผนก) AS E1, แผนก (ชื่อแผนก, งบประมาณ) AS D
 WHERE E1.ชื่อ = 'สมชาย' AND E1.ชื่อแผนก = D.ชื่อแผนก
 AND VALID (D) CONTAINS PERIOD '01/01/1982 - 12/31/1984'
 OR VALID (D) CONTAINS PERIOD '01/01/1987 - NOW'

8.3.3.46 บอกชื่อแผนก และงบประมาณในปี ค.ศ. 1984 และปี ค.ศ. 1987 ของแผนกที่สมหญิงเป็นผู้จัดการ (คำตอบ : ('ของเล่น', 150000 | [01/01/1984 – 07/31/1984]) ('ของเล่น', 200000 | [08/01/1984 – 12/31/1984]) และ ('ของเล่น', 100000 | [01/01/1987 – 12/31/1987]))

SELECT D.ชื่อแผนก, D.งบประมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VALID PERIOD '[01/01/1994 - 12/31/1984]' + PERIOD '[01/01/1987 - 12/31/1987]'
 FROM พนักงาน (รหัส, ชื่อ) AS E1, แผนก (ชื่อแผนก, งบประมาณ, รหัส) AS D
 WHERE E1.รหัส = D.รหัส AND E1.ชื่อ = 'สมหญิง'

8.3.3.47 บอกชื่อแผนก และช่วงเวลาที่มีสมาชิกทำงานตั้งแต่ต้นปี ค.ศ. 1986 และปี ค.ศ. 1987 (คำตอบ : ('ของเล่น' | [02/01/1982 - 01/31/1987]))

SELECT E1.ชื่อแผนก
 FROM พนักงาน (ชื่อ, ชื่อแผนก) AS E1
 WHERE E1.ชื่อ = 'สมชาย' AND VALID (E1) CONTAINS DATE '01/01/1986'
 AND EXISTS (SELECT * FROM พนักงาน (ชื่อ, ชื่อแผนก) AS E2
 WHERE E1.ชื่อ = E2.ชื่อ AND E1.ชื่อแผนก = E2.ชื่อแผนก
 AND VALID (E2) CONTAINS DATE '01/01/1987')

8.3.3.48 บอกชื่อปัจจุบันของผู้จัดการที่เป็นผู้จัดการของสมาชิก ในวันคริสต์มาสปี ค.ศ. 1984 และวันเกิดปีที่ 27 ของสมาชิก และวันที่เริ่มเป็นผู้จัดการ (คำตอบ : ('สมหญิง', 01/01/1982))

SELECT SNAPSHOT E1.ชื่อ, BEGIN (D1)
 FROM พนักงาน (รหัส, ชื่อ) AS E1, แผนก (ชื่อแผนก, รหัส) AS D1,
 พนักงาน (รหัส, ชื่อ, ชื่อแผนก) AS E2
 WHERE E1.รหัส = D1.รหัส AND E2.ชื่อ = 'สมชาย' AND E2.ชื่อแผนก = D1.ชื่อแผนก
 AND VALID (E2) CONTAINS DATE '12/25/1984'
 AND D1.รหัส = (SELECT D2.รหัส
 FROM แผนก (ชื่อแผนก, รหัส) AS D2, พนักงาน (รหัส, วันเกิด, ชื่อแผนก) AS E3
 WHERE E3.รหัส = E2.รหัส AND D2.ชื่อแผนก = E3.ชื่อแผนก
 AND VALID (E3) CONTAINS E3.วันเกิด + INTERVAL '27' YEAR)

8.3.3.49 บอกชื่อแผนก ชื่อผู้จัดการ วันที่เปลี่ยนแปลง และค่าใหม่ของงบประมาณทุกๆงบประมาณที่เปลี่ยนแปลงในปี ค.ศ. 1984, 1986 และ 1988 (คำตอบ : ('ของเล่น', 'สมหญิง', 200000, 08/01/1984))

SELECT SNAPSHOT D1.ชื่อแผนก, E.ชื่อ, D1.งบประมาณ, BEGIN (D1)

VALID PERIOD '[01/01/1984 - 12/31/1984]' + PERIOD '[01/01/1986 - 12/31/1986]'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

+ PERIOD '[01/01/1988 - 12/31/1988]'
FROM พนักงาน (รหัส, ชื่อ) AS E, แผนก (ชื่อแผนก, งบประมาณ, รหัส) (PERIOD) AS D1
WHERE E.รหัส = D1.รหัส
AND D1.งบประมาณ <> (SELECT SNAPSHOT D2.งบประมาณ
FROM แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D2
WHERE D2.ชื่อแผนก = D1.ชื่อแผนก AND VALID (D1) MEETS VALID (D2))

```

8.4 การแปลงจากภาษา SQL เชิงเวลาเป็นภาษา SQL

จะขอยกตัวอย่างคำถามดังนี้ “บอกชื่อแผนก ชื่อผู้จัดการ วันที่เปลี่ยนแปลง และค่าใหม่ของงบประมาณทุกๆงบประมาณที่เปลี่ยนแปลงในปี ค.ศ. 1984, 1986 และ 1988” จะใช้ภาษา SQL เชิงเวลาดังนี้

```

SELECT SNAPSHOT D1.ชื่อแผนก, E.ชื่อ, D1.งบประมาณ, BEGIN (D1)
VALID PERIOD '[01/01/1984 - 12/31/1984]' + PERIOD '[01/01/1986 - 12/31/1986]'
+ PERIOD '[01/01/1988 - 12/31/1988]'
FROM พนักงาน (รหัส, ชื่อ) AS E, แผนก (ชื่อแผนก, งบประมาณ, รหัส) (PERIOD) AS D1
WHERE E.รหัส = D1.รหัส
AND D1.งบประมาณ <> (SELECT SNAPSHOT D2.งบประมาณ
FROM แผนก (ชื่อแผนก, งบประมาณ) (PERIOD) AS D2
WHERE D2.ชื่อแผนก = D1.ชื่อแผนก
AND VALID (D1) MEETS VALID (D2))

```

สามารถแปลงเป็นภาษา SQL ได้ดังนี้

```

SELECT D1.ชื่อแผนก, E.ชื่อ, D1.งบประมาณ, D1.เวลาเริ่มต้น
FROM แผนก D1, พนักงาน E
WHERE E.รหัส = D1.รหัส
AND D1.งบประมาณ <> (SELECT D2.งบประมาณ
FROM แผนก D2
WHERE D2.ชื่อแผนก = D1.ชื่อแผนก
AND D1.เวลาเริ่มต้น = D2.เวลาสิ้นสุด)
AND E.เวลาเริ่มต้น <= D1.เวลาสิ้นสุด
AND E.เวลาสิ้นสุด >= D1.เวลาเริ่มต้น
AND D1.เวลาเริ่มต้น <= '12/31/1984'
AND D1.เวลาสิ้นสุด >= '01/01/1984'

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

AND E.เวลาเริ่มต้น <= '12/31/1984'
AND E.เวลาสิ้นสุด >= '01/01/1984'
UNION
SELECT D1.ชื่อแผนก, E.ชื่อ, D1.งบประมาณ, D1.เวลาเริ่มต้น
FROM แผนก D1, พนักงาน E
WHERE E.รหัส = D1.รหัส
AND D1.งบประมาณ <> (SELECT D2.งบประมาณ
                        FROM แผนก D2
                        WHERE D2.ชื่อแผนก = D1.ชื่อแผนก
                        AND D1.เวลาเริ่มต้น = D2.เวลาสิ้นสุด)
AND E.เวลาเริ่มต้น <= D1.เวลาสิ้นสุด
AND E.เวลาสิ้นสุด >= D1.เวลาเริ่มต้น
AND D1.เวลาเริ่มต้น <= '12/31/1986'
AND D1.เวลาสิ้นสุด >= '01/01/1986'
AND E.เวลาเริ่มต้น <= '12/31/1986'
AND E.เวลาสิ้นสุด >= '01/01/1986'
UNION
SELECT D1.ชื่อแผนก, E.ชื่อ, D1.งบประมาณ, D1.เวลาเริ่มต้น
FROM แผนก D1, พนักงาน E
WHERE E.รหัส = D1.รหัส
AND D1.งบประมาณ <> (SELECT D2.งบประมาณ
                        FROM แผนก D2
                        WHERE D2.ชื่อแผนก = D1.ชื่อแผนก
                        AND D1.เวลาเริ่มต้น = D2.เวลาสิ้นสุด)
AND E.เวลาเริ่มต้น <= D1.เวลาสิ้นสุด
AND E.เวลาสิ้นสุด >= D1.เวลาเริ่มต้น
AND D1.เวลาเริ่มต้น <= '12/31/1988'
AND D1.เวลาสิ้นสุด >= '01/01/1988'
AND E.เวลาเริ่มต้น <= '12/31/1988'
AND E.เวลาสิ้นสุด >= '01/01/1988'

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9

การแปลงภาษา SQL เชิงเวลา

จากบทที่ผ่านมาจะเห็นว่า การแปลงภาษา SQL เชิงเวลาเป็นภาษา SQL นั้นสามารถทำได้ยากและไม่สะดวก ดังนั้นเพื่อความง่ายในการแปลงภาษา SQL เชิงเวลาเป็นภาษา SQL เพื่อใช้ในฐานข้อมูลทั่วไป จึงได้คิดวิธีใหม่ขึ้นมาใช้เพื่อความง่ายและความสะดวกในการแปลง ซึ่งวิธีนี้จะใช้ทฤษฎี Time Normalization ในการจัดโครงสร้างตารางใหม่

9.1 ทฤษฎี Time Normalization

ตารางจะเป็น Time Normal Form (TNF) ก็ต่อเมื่อเป็น BCNF (Boyce-Codd Normal Form) และไม่มีมีความเกี่ยวข้องเชิงเวลา (Temporal Dependency) โดยความเกี่ยวข้องเชิงเวลาจะเกิดขึ้นเมื่อข้อมูลที่มีการเปลี่ยนแปลงทางเวลาโดยไม่เกี่ยวข้องกัน 2 คอลัมน์ขึ้นไปอยู่รวมกันในตารางเดียวกัน ซึ่งจากตารางพนักงานในตารางที่ 8-1 จะสามารถจัดโครงสร้างตารางใหม่โดยใช้ทฤษฎี Time Normalization เพื่อให้ตารางเป็น Time Normal Form ได้ 4 ตารางคือ ตารางพนักงาน_ชื่อ ตารางพนักงาน_เงินเดือน ตารางพนักงาน_แผนก และตารางพนักงาน_อื่น ดังแสดงในตารางที่ 9-1 ตารางที่ 9-2 ตารางที่ 9-3 และตารางที่ 9-4 ตามลำดับ

รหัส	ชื่อ	เวลาเริ่มต้น	เวลาสิ้นสุด
1	สมชาย	02/01/1982	12/31/1987
1	ชาติชาย	01/01/1988	12/31/9999
2	สมหญิง	01/01/1982	12/31/9999

ตารางที่ 9-1 ตารางพนักงาน_ชื่อ

รหัส	เงินเดือน	เวลาเริ่มต้น	เวลาสิ้นสุด
1	20000	02/01/1982	05/31/1982
1	30000	06/01/1982	01/31/1985
1	40000	02/01/1985	01/31/1987
1	40000	04/01/1987	12/31/9999
2	30000	01/01/1982	07/31/1984
2	40000	08/01/1984	08/31/1986
2	50000	09/01/1986	12/31/9999

ตารางที่ 9-2 ตารางพนักงาน_เงินเดือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัส	ชื่อแผนก	เวลาเริ่มต้น	เวลาสิ้นสุด
1	ของเล่น	02/01/1982	01/31/1987
1	หนังสือ	04/01/1987	12/31/9999
2	ของเล่น	01/01/1982	12/31/9999

ตารางที่ 9-3 ตารางพนักงาน_แผนก

รหัส	เพศ	วันเกิด
1	ชาย	07/01/1955
2	หญิง	10/01/1960

ตารางที่ 9-4 ตารางพนักงาน_อื่น

จากตารางทักษะในตารางที่ 8-2 จะเห็นว่าตรงกับทฤษฎี Time Normalization แล้ว ซึ่งก็คือตารางเป็น Time Normal Form แล้ว แต่เนื่องจากเราไม่ควรเปลี่ยนแปลงตารางเดิม ดังนั้นจึงควรมีอีกตารางหนึ่งเพิ่มขึ้นมาคือตารางทักษะ_ทักษะ ดังแสดงในตารางที่ 9-5

รหัส	ทักษะ	เวลาเริ่มต้น	เวลาสิ้นสุด
1	พิมพ์ดีด	04/01/1982	12/31/9999
1	จัดเรียงเอกสาร	01/01/1985	12/31/9999
1	ขับรถ	01/01/1982	05/01/1982
1	ขับรถ	06/01/1984	05/31/1988
2	บริหารงาน	01/01/1982	12/31/9999

ตารางที่ 9-5 ตารางทักษะ_ทักษะ

และจากตารางแผนกในตารางที่ 8-3 จะสามารถจัดโครงสร้างตารางใหม่โดยใช้ทฤษฎี Time Normalization เพื่อให้ตารางเป็น Time Normal Form ได้ 2 ตารางคือ ตารางแผนก_งบประมาณ และตารางแผนก_ผู้จัดการ ดังแสดงในตารางที่ 9-6 และตารางที่ 9-7 ตามลำดับ

ชื่อแผนก	งบประมาณ	เวลาเริ่มต้น	เวลาสิ้นสุด
ของเล่น	150000	01/01/1982	07/31/1984
ของเล่น	200000	08/01/1984	12/31/1986
ของเล่น	100000	01/01/1987	12/31/9999
หนังสือ	50000	04/01/1987	12/31/9999

ตารางที่ 9-6 ตารางแผนก_งบประมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อแผนก	รหัส	เวลาเริ่มต้น	เวลาสิ้นสุด
ของเล่น	2	01/01/1982	12/31/9999
หนังสือ	1	04/01/1987	12/31/9999

ตารางที่ 9-7 ตารางแผนก_ผู้จัดการ

9.2 การนำทฤษฎี Time Normalization มาประยุกต์ใช้ในการแปลงภาษา SQL เชิงเวลา

การจัดโครงสร้างตารางใหม่นี้มีประโยชน์มากในการตอบคำถามเชิงเวลา เนื่องจากการเก็บข้อมูลของตารางที่ไม่ได้เป็น Time Normal Form นั้น เมื่อมีการเปลี่ยนแปลงข้อมูลเพียงคอลัมน์เดียวก็ต้องทำการเก็บข้อมูลใหม่หมดทั้งทัพบิลโดยเปลี่ยนแปลงเพียงค่าของคอลัมน์นั้นและคอลัมน์เวลา ทำให้การตอบคำถามเชิงเวลาบางประเภททำได้ไม่สะดวก เช่น ให้ออกเวลาที่สมชายเริ่มทำงานในแผนกของเล่น ถ้าใช้ภาษา SQL จะได้คำตอบเป็นวันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1982 วันที่ 1 เดือนมิถุนายน ปี ค.ศ. 1982 และวันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1985 ทั้งๆที่ความจริงแล้วสมชายเริ่มทำงานในแผนกของเล่นในวันที่ 1 เดือนกุมภาพันธ์ ปี ค.ศ. 1982 และทำอยู่ในแผนกนี้ตลอดจนกระทั่งลาออกไปในวันที่ 31 เดือนมกราคม ปี ค.ศ. 1987 แต่ที่ได้คำตอบเช่นนี้เป็นเพราะว่าสมชายได้ขึ้นเงินเดือนในวันที่นั้น

แต่ถ้าใช้ภาษา SQL บนตารางใหม่ที่จัดโครงสร้างตารางใหม่แล้วนั้น จะสามารถทำได้สะดวกมาก โดยใช้คำสั่งดังนี้

```
SELECT เวลาเริ่มต้น
FROM พนักงาน_แผนก
WHERE รหัส = (SELECT รหัส FROM พนักงาน_ชื่อ WHERE ชื่อ = 'สมชาย')
AND ชื่อแผนก = 'ของเล่น'
```

ในโปรแกรมแปลงภาษา SQL เชิงเวลาเป็นภาษา SQL นี้ ผู้ใช้จะไม่ทราบว่ามีการสร้างตารางที่จัดโครงสร้างใหม่นี้ แต่การใช้วิธีนี้จำเป็นต้องมีอีกตารางหนึ่งไว้เก็บว่าคอลัมน์ของตารางเดิมเมื่อจัดโครงสร้างตารางใหม่จะเป็นตารางใด ตารางนี้เรียกว่า ตารางเมตต้า (Meta Table) ดังแสดงในตารางที่ 9-8

ชื่อตาราง	ชื่อคอลัมน์	ชื่อตารางใหม่
พนักงาน	รหัส	NULL
พนักงาน	ชื่อ	พนักงาน_ชื่อ
พนักงาน	เงินเดือน	พนักงาน_เงินเดือน
พนักงาน	แผนก	พนักงาน_แผนก
พนักงาน	เพศ	พนักงาน_อื่น
พนักงาน	วันเกิด	พนักงาน_อื่น
ทักษะ	รหัส	NULL
ทักษะ	ทักษะ	ทักษะ_ทักษะ
แผนก	ชื่อแผนก	NULL
แผนก	งบประมาณ	แผนก_งบประมาณ
แผนก	รหัส	แผนก_ผู้จัดการ

ตารางที่ 9-8 ตารางเมต้า

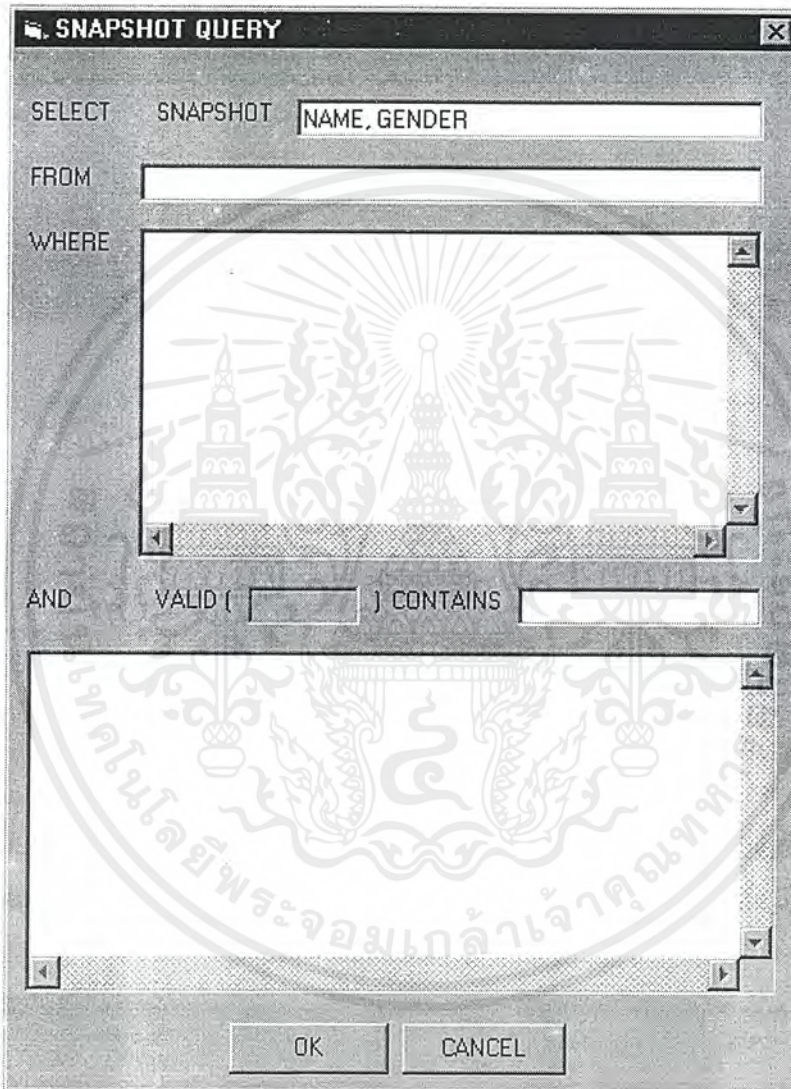
โดยค่า NULL ในคอลัมน์ชื่อตารางใหม่หมายถึงคอลัมน์นี้เป็น Primary Key ของตารางทั้งหมดที่จัดโครงสร้างใหม่ หรือก็คือมีคอลัมน์นี้อยู่ในทุกตารางที่จัดโครงสร้างใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วิธีการใช้โปรแกรม QUERY.EXE

ในส่วนนี้จะเป็นการอธิบายการใช้งานโปรแกรม QUERY.EXE ว่าสามารถทำงานได้ครอบคลุมภาษา TSQL2 เท่าไร และมีข้อจำกัดในการใช้งานโปรแกรมอย่างไรบ้าง



ใน Textbox ที่ต่อจากคำว่า SELECT SNAPSHOT ให้ใส่คอดัมน์ที่ต้องการ SELECT ในโปรแกรมนี้จะครอบคลุมเฉพาะส่วนของการ SELECT แบบ SNAPSHOT เท่านั้น และไม่สามารถใช้ Built-in Function ได้ ในที่นี้จะยกตัวอย่าง SELECT ชื่อของพนักงาน (NAME) และเพศ (GENDER)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SNAPSHOT QUERY

SELECT SNAPSHOT

FROM

WHERE

AND VALID () CONTAINS

ใน Textbox ที่ต่อจากคำว่า FROM จะใส่ตารางที่ต้องการ SELECT โดยในโปรแกรมนี้จะสามารถ SELECT ได้จากตารางเพียง 1 ตารางเท่านั้น ในที่นี้จะยกตัวอย่างโดย SELECT จากตารางพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SNAPSHOT QUERY

SELECT SNAPSHOT NAME, GENDER

FROM EMP

WHERE SALARY > 30000
AND DEPTNAME = 'ของเล่น'

AND VALID (EMP) CONTAINS

OK CANCEL

ใน Textbox ที่ต่อจากคำว่า WHERE ให้ใส่เงื่อนไขที่กำหนดขึ้นไป โดยสามารถใส่ได้อย่างอิสระ ในที่นี้จะยกตัวอย่างโดยใส่ SALARY > 30000 AND DEPTNAME = 'ของเล่น'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SNAPSHOT QUERY

SELECT SNAPSHOT

FROM

WHERE

AND VALID () CONTAINS

ใน Textbox ที่ต่อจากคำว่า CONTAIN จะเป็นการบอกว่าเงื่อนไขทั้งหมดที่ใส่มาจะต้องเป็นจริงในช่วงเวลาใด ในที่นี้จะยกตัวอย่างโดยใส่ '01/01/1987'

เมื่อใส่เงื่อนไขต่างๆครบถ้วนแล้วให้กดปุ่ม OK โปรแกรมจะทำการเชื่อมต่อกับ Database Server เพื่อดึงข้อมูลจากตารางเมต้าที่ชื่อ METATAB โดยจะขึ้น Dialog Box ให้ทำการ Login

Logon to INFORMIX

Database Name:

Host Name:

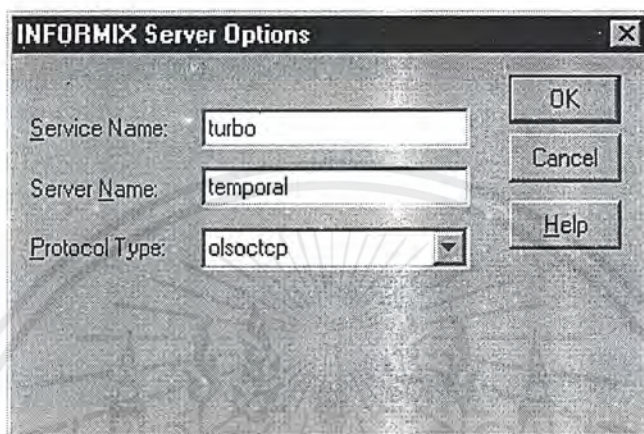
User Name:

Password:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในช่อง DATABASE NAME ให้เลือก `db1@temporal` หรือฐานข้อมูลที่ต้องการติดต่อ ในช่อง HOST NAME ให้ใส่ชื่อ HOST หรือ IP Address ของ Database Server เช่น 192.168.0.5 ในช่อง USERNAME ให้ใส่ชื่อของ User ที่มีสิทธิ์อ่านตาราง METATAB บนฐานข้อมูล ในช่อง PASSWORD ให้ใส่รหัสผ่านของ User นั้น

ปุ่ม OPTION ใช้เพื่อตั้งค่าเริ่มต้นต่างๆในการเชื่อมต่อ เมื่อกดแล้วจะมี Dialog Box ปรากฏขึ้นมา



SERVICE NAME ให้ใส่ชื่อ SERVICE ที่ใช้ติดต่อ ในที่นี้ใช้ turbo (DEFAULT)

SERVER NAME ให้ใส่ชื่อของ Database Server ที่ใช้ติดต่อ ในที่นี้ใช้ temporal

PROTOCOL TYPE ให้เลือกชนิดของโปรโตคอล ในที่นี้เลือก olsoc tcp (DEFAULT)

เมื่อเลือกเสร็จแล้วให้กด OK เพื่อ SAVE ค่าที่กำหนดไว้ ถ้าไม่ต้องการ SAVE ให้กด CANCEL

เพื่อยกเลิก

เมื่อกำหนดค่าเสร็จเรียบร้อยแล้วให้กด OK เพื่อติดต่อกับ Database Server ถ้าติดต่อสำเร็จจะได้ผลลัพธ์ใน Textbox ด้านล่างสุดดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SNAPSHOT QUERY

SELECT SNAPSHOT

FROM

WHERE

AND VALID () CONTAINS

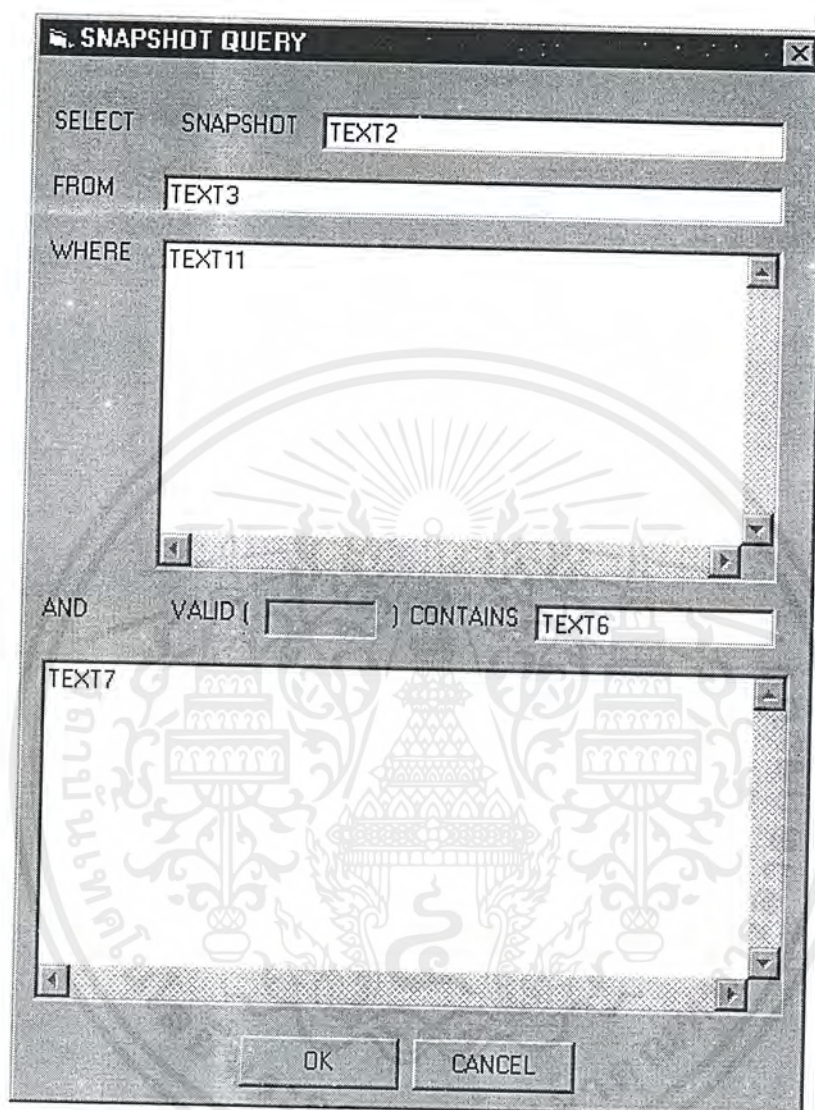
OK CANCEL

โดยผลลัพธ์ที่ได้จะสามารถนำไปใช้บนฐานข้อมูลต่างๆไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

Source Code



Private SQL92 As String

Private FromTable(1 To 20) As String

Private Splited(1 To 20) As String

Private Function IsCol(ColName As String) As Boolean

IsCol = False

If ColName = Text9.Text Then

 IsCol = True

End If

Do

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

NavControl1.ddoTable.NextRecord
If IsCol = False And ColName = Text9.Text Then
    IsCol = True
End If
Loop Until NavControl1.ddoTable.CurrentRecord.LastRecord Or IsCol = True
NavControl1.ddoTable.FirstRecord
End Function

```

```

Private Sub splitstr(String1 As String)

```

```

Dim i, j As Integer

```

```

String1 = Trim(String1)

```

```

j = 1

```

```

i = InStr(String1, " ")

```

```

While i <> 0

```

```

    Splited(j) = Left(String1, i - 1)

```

```

    j = j + 1

```

```

    String1 = Trim(Right(String1, Len(String1) - i))

```

```

    i = InStr(String1, " ")

```

```

Wend

```

```

Splited(j) = String1

```

```

End Sub

```

```

Private Function TNFTab(TabName As String, ColName As String) As String

```

```

TNFTab = ""

```

```

If TabName = Text8.Text And ColName = Text9.Text Then

```

```

    TNFTab = Text10.Text

```

```

End If

```

```

Do

```

```

NavControl1.ddoTable.NextRecord

```

```

If TNFTab = "" And TabName = Text8.Text And ColName = Text9.Text Then

```

```

    TNFTab = Text10.Text

```

```

End If

```

```

Loop Until NavControl1.ddoTable.CurrentRecord.LastRecord Or TNFTab <> ""

```

เอกสารนี้เป็นทรัพย์สินทางปัญญาหรือสิขสิทธิ์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ หากมีข้อผิดพลาดประการใด ผู้จัดทำขออภัยไว้ ณ ที่นี้ และขอสงวนสิทธิ์ในการนำเอกสารฉบับนี้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NavControl1.ddoTable.FirstRecord
```

```
End Function
```

```
Private Function ColName(TabName As String) As String
```

```
ColName = ""
```

```
If TabName = Text8.Text And Text10.Text = "" Then
```

```
ColName = Text9.Text
```

```
End If
```

```
Do
```

```
NavControl1.ddoTable.NextRecord
```

```
If ColName = "" And TabName = Text8.Text And Text10.Text = "" Then
```

```
ColName = Text9.Text
```

```
End If
```

```
Loop Until NavControl1.ddoTable.CurrentRecord.LastRecord Or ColName <> ""
```

```
NavControl1.ddoTable.FirstRecord
```

```
End Function
```

```
Private Sub Main()
```

```
Dim i, j As Integer
```

```
Dim TMPStr As String
```

```
On Error GoTo EndMain
```

```
NavControl1.ddoDataGroup.QueryData
```

```
SQL92 = ""
```

```
Erase FromTable
```

```
SQL92 = "SELECT " & Text2.Text & vbCrLf & "FROM " & Text3.Text
```

```
j = 1
```

```
FromTable(1) = Trim(Text3.Text)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

splitstr (Text11.Text)
' find from clause
While Splited(j) <> ""
    TMPStr = TNFTab(Text3.Text, Splited(j))
    If IsCol(Splited(j)) And TMPStr <> "" Then
        SQL92 = SQL92 & "," & TMPStr
        FromTable(i) = TMPStr
    End If
    j = j + 1
Wend
SQL92 = SQL92 & vbCrLf & "WHERE "
j = 1
While Splited(j) <> ""
    TMPStr = TNFTab(Text3.Text, Splited(j))
    If IsCol(Splited(j)) And TMPStr <> "" Then
        SQL92 = SQL92 & Text3.Text & "." & ColName(Text3.Text) & " = " & TMPStr & "." &
        ColName(Text3.Text) & vbCrLf & "AND "
        Splited(j) = TMPStr & "." & Splited(j)
    ElseIf IsCol(Splited(j)) And TMPStr = "" Then
        Splited(j) = Trim(Text3.Text) & "." & Splited(j)
    End If
    j = j + 1
Wend
j = 1
While Splited(j) <> ""
    If UCase(Splited(j)) = "AND" Then
        SQL92 = SQL92 & vbCrLf
    End If
    SQL92 = SQL92 & Splited(j) & " "
    j = j + 1
Wend

```

SQL92 = SQL92 & vbCrLf

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

j = 1
While FromTable(j) <> ""
    SQL92 = SQL92 & "AND " & FromTable(j) & ".VALIDTIMESTART <= " & Text6.Text & vbCrLf
    -
    & "AND " & FromTable(j) & ".VALIDTIMEEND >= " & Text6.Text & vbCrLf
    j = j + 1
Wend

```

```

SQL92 = SQL92 & "AND " & Text3.Text & ".VALIDTIMESTART <= " & Text6.Text & vbCrLf _
    & "AND " & Text3.Text & ".VALIDTIMEEND >= " & Text6.Text & ";"

```

```
Text7.Text = SQL92
```

```
EndMain:
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
Main
```

```
End Sub
```

```
Private Sub Command2_Click()
```

```
Unload Form1
```

```
End Sub
```

```
Private Sub Text3_Change()
```

```
Text5.Text = Text3.Text
```

```
End Sub
```

```
'Private Sub DataDirector_Info()
```

```
'[DataDirectorInfoStart]
```

```
'NumDataLinks=3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'[DataLink]
'GroupName=metatab
'ControlName=Text10
'ClassId=9
'Synchronize=0
'GetAllData=1
'StoreSelect=1
'SortOrder=0
'SortFlag=0
'DefaultType=0
'TrimTrailingBlanks=0
'[DataLinkButtonInfo]
'[EndDataLinkButtonInfo]
'[DataLinkPath]
'NumElements=1
'[DataLinkPathInfo]
'TableName=metatab
'NumColumns=1
'ColumnName=tnftab
'[EndDataLinkPathInfo]
'[EndDataLinkPath]
'[EndDataLink]
'[DataLink]
'GroupName=metatab
'ControlName=Text8
'ClassId=9
'Synchronize=0
'GetAllData=1
'StoreSelect=1
'SortOrder=0
'SortFlag=0
'DefaultType=1
'TrimTrailingBlanks=0
'[DataLinkButtonInfo]

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'[EndDataLinkButtonInfo]
'[DataLinkPath]
'NumElements=1
'[DataLinkPathInfo]
'TableName=metatab
'NumColumns=1
'ColumnName=tablename
'[EndDataLinkPathInfo]
'[EndDataLinkPath]
'[EndDataLink]
'[DataLink]
'GroupName=metatab
'ControlName=Text9
'ClassId=9
'Synchronize=0
'GetAllData=1
'StoreSelect=1
'SortOrder=0
'SortFlag=0
'DefaultType=1
'TrimTrailingBlanks=0
'[DataLinkButtonInfo]
'[EndDataLinkButtonInfo]
'[DataLinkPath]
'NumElements=1
'[DataLinkPathInfo]
'TableName=metatab
'NumColumns=1
'ColumnName=colname
'[EndDataLinkPathInfo]
'[EndDataLinkPath]
'[EndDataLink]
'[DataDirectorInfoEnd]
'End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การตอบคำถามเชิงเวลาด้วยภาษาที่ใช้ในการตอบคำถามเชิงเวลาแบบต่างๆ

ตารางต่อไปนี้เป็นตารางที่ใช้ในการตอบคำถามเชิงเวลา

ตาราง EMP

ID	NAME	SALARY	GENDER	D-BIRTH	DEPTNAME	VALIDTimestart	VALIDTIMEEND
1	สมชาย	20000	ชาย	07/01/1955	ของเล่น	02/01/1982	05/31/1982
1	สมชาย	30000	ชาย	07/01/1955	ของเล่น	06/01/1982	01/31/1985
1	สมชาย	40000	ชาย	07/01/1955	ของเล่น	02/01/1985	01/31/1987
1	สมชาย	40000	ชาย	07/01/1955	หนังสือ	04/01/1987	12/31/1987
1	ชาติชาย	40000	ชาย	07/01/1955	หนังสือ	01/01/1988	12/31/9999
2	สมหญิง	30000	หญิง	10/01/1960	ของเล่น	01/01/1982	07/31/1984
2	สมหญิง	40000	หญิง	10/01/1960	ของเล่น	08/01/1984	08/31/1986
2	สมหญิง	50000	หญิง	10/01/1960	ของเล่น	09/01/1986	12/31/9999

ตาราง SKILL

ID	SKILL	VALIDTimestart	VALIDTIMEEND
1	พิมพ์ดีด	04/01/1982	12/31/9999
1	จัดเรียงเอกสาร	01/01/1985	12/31/9999
1	ขับรถ	01/01/1982	05/01/1982
1	ขับรถ	06/01/1984	05/31/1988
2	บริหารงาน	01/01/1982	12/31/9999

ตาราง DEPT

DEPTNAME	BUDGET	ID	VALIDTimestart	VALIDTIMEEND
ของเล่น	150000	2	01/01/1982	07/31/1984
ของเล่น	200000	2	08/01/1984	12/31/1986
ของเล่น	100000	2	01/01/1987	12/31/9999
หนังสือ	50000	1	04/01/1987	12/31/9999

ภาษา TSQL2 ซึ่งเป็นภาษา SQL เชิงเวลาที่มีคุณสมบัติในการจัดโครงสร้างตารางใหม่ (Restructuring) ซึ่งเป็นกรรวมค่าที่เหมือนกันของคอลัมน์ที่เปลี่ยนแปลงตามเวลาเข้าไว้ด้วยกัน สามารถเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ได้โดยเขียนชื่อตารางตามด้วยชื่อคอลัมน์ที่ต้องการจัดโครงสร้างตารางใหม่อยู่ในวงเล็บ ยกตัวอย่างเช่น การจัดโครงสร้างตารางใหม่ของตาราง EMP บนคอลัมน์ ID และคอลัมน์ NAME หรือ EMP (ID, NAME) จะมีลักษณะดังต่อไปนี้

ID	NAME	SALARY	GENDER	D-BIRTH	DEPTNAME	VALIDTimestart	VALIDTIMEEND
1	สมชาย	20000	ชาย	07/01/1955	ของเล่น	02/01/1982	05/31/1982
		30000	ชาย	07/01/1955	ของเล่น	06/01/1982	01/31/1985
		40000	ชาย	07/01/1955	ของเล่น	02/01/1985	01/31/1987
		40000	ชาย	07/01/1955	หนังสือ	04/01/1987	12/31/1987
1	ชาติชาย	40000	ชาย	07/01/1955	หนังสือ	01/01/1988	12/31/9999
2	สมหญิง	30000	หญิง	10/01/1960	ของเล่น	01/01/1982	07/31/1984
		40000	หญิง	10/01/1960	ของเล่น	08/01/1984	08/31/1986
		50000	หญิง	10/01/1960	ของเล่น	09/01/1986	12/31/9999

และการจัดโครงสร้างตารางใหม่ของตาราง SKILL บนคอลัมน์ ID และคอลัมน์ SKILL หรือ SKILL (ID, SKILL) จะมีลักษณะดังต่อไปนี้

ID	SKILL	VALIDTimestart	VALIDTIMEEND
1	พิมพ์ดีด	04/01/1982	12/31/9999
1	จัดเรียงเอกสาร	01/01/1985	12/31/9999
1	ขับรถ	01/01/1982	05/01/1982
		06/01/1984	05/31/1988
2	บริหารงาน	01/01/1982	12/31/9999

ภาษา TSQL2 ยังมีคุณสมบัติในการจัดโครงสร้างตารางใหม่แบบช่วงเวลา (Period) ซึ่งเป็นการรวมค่าที่เหมือนกันของคอลัมน์ที่เปลี่ยนแปลงตามเวลาที่ต่อเนื่องกันเข้าไว้ด้วยกัน ยกตัวอย่างเช่น การจัดโครงสร้างตารางใหม่แบบช่วงเวลาของตาราง SKILL บนคอลัมน์ ID และคอลัมน์ SKILL หรือ SKILL (ID, SKILL) (PERIOD) จะมีลักษณะดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ID	SKILL	VALIDTimestart	VALIDTIMEEND
1	พิมพ์ดีด	04/01/1982	12/31/9999
1	จัดเรียงเอกสาร	01/01/1985	12/31/9999
1	ขับรถ	01/01/1982	05/01/1982
1	ขับรถ	06/01/1984	05/31/1988
2	บริหารงาน	01/01/1982	12/31/9999

จะเห็นว่าทักษะขับรถไม่ได้อยู่รวมกันเนื่องจากช่วงเวลาไม่ได้ต่อเนื่องกัน

แต่เนื่องจากภาษา SQL-92 ไม่มีการจัดโครงสร้างตารางใหม่ จึงต้องทำการจัดโครงสร้างข้อมูลเองโดยการทำให้ Time Normalized ซึ่งจะคล้ายกับการจัดโครงสร้างตารางใหม่ โดยตารางจะเป็น Time Normal Form (TNF) ก็ต่อเมื่อเป็น BCNF (Boyce-Codd Normal Form) และไม่มี Temporal Dependency โดย Temporal Dependency จะเกิดขึ้นเมื่อข้อมูลที่มีการเปลี่ยนแปลงทางเวลาโดยไม่เกี่ยวข้องกัน 2 คอลัมน์ขึ้นไปอยู่รวมกันในตารางเดียวกัน ซึ่งจากตาราง EMP จะแปลงเป็น Time Normal Form ได้ 4 ตารางคือ ตาราง EMP_NAME, ตาราง EMP_SALARY, ตาราง EMP_DEPT และตาราง EMP_NOTIME ดังนี้

ตาราง EMP_NAME

ID	NAME	VALIDTimestart	VALIDTIMEEND
1	สมชาย	02/01/1982	12/31/1987
1	ชาติชาย	01/01/1988	12/31/9999
2	สมหญิง	01/01/1982	12/31/9999

ตาราง EMP_SALARY

ID	SALARY	VALIDTimestart	VALIDTIMEEND
1	20000	02/01/1982	05/31/1982
1	30000	06/01/1982	01/31/1985
1	40000	02/01/1985	01/31/1987
1	40000	04/01/1987	12/31/9999
2	30000	01/01/1982	07/31/1984
2	40000	08/01/1984	08/31/1986
2	50000	09/01/1986	12/31/9999

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง EMP_DEPT

ID	DEPTNAME	VALIDTimestart	VALIDTIMEEND
1	ของเล่น	02/01/1982	01/31/1987
1	หนังสือ	04/01/1987	12/31/9999
2	ของเล่น	01/01/1982	12/31/9999

ตาราง EMP_NOTIME

ID	GENDER	D-BIRTH
1	ชาย	07/01/1955
2	หญิง	10/01/1960

จากตาราง SKILL จะเห็นว่าตารางเป็น Time Normal Form แล้ว แต่เนื่องจากเราไม่ควรเปลี่ยนแปลงตารางเดิม ดังนั้นจึงควรมีอีกตารางหนึ่งเพิ่มขึ้นมาคือตาราง SKILL_SKILL ดังนี้

ตาราง SKILL_SKILL

ID	SKILL	VALIDTimestart	VALIDTIMEEND
1	พิมพ์ดีด	04/01/1982	12/31/9999
1	จัดเรียงเอกสาร	01/01/1985	12/31/9999
1	ขับรถ	01/01/1982	05/01/1982
1	ขับรถ	06/01/1984	05/31/1988
2	บริหารงาน	01/01/1982	12/31/9999

และจากตาราง DEPT จะแปลงเป็น Time Normal Form ได้ 2 ตารางคือ ตาราง DEPT_BUDGET และตาราง DEPT_MGR ดังนี้

ตาราง DEPT_BUDGET

DEPTNAME	BUDGET	VALIDTimestart	VALIDTIMEEND
ของเล่น	150000	01/01/1982	07/31/1984
ของเล่น	200000	08/01/1984	12/31/1986
ของเล่น	100000	01/01/1987	12/31/9999
หนังสือ	50000	04/01/1987	12/31/9999

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง DEPT_MGR

DEPTNAME	ID	VALIDTimestart	VALIDTIMEEND
ของเล่น	2	01/01/1982	12/31/9999
หนังสือ	1	04/01/1987	12/31/9999

โดยฟังก์ชันที่ใช้สำหรับช่วยในการตอบคำถามมีดังนี้

1. ฟังก์ชันสำหรับแปลงวันที่ 12 เดือนธันวาคม ปี ค.ศ. 9999 ให้เป็นวันที่ปัจจุบัน

```
CREATE FUNCTION NOW (VE DATE) RETURNING DATE;
DEFINE NEWVE DATE;
IF VE = '12/31/9999' THEN LET NEWVE = DATE (CURRENT);
ELSE LET NEWVE = VE;
END IF;
RETURN NEWVE;
END FUNCTION;
```

2. ฟังก์ชันสำหรับหาวันที่ที่มากกว่า

```
CREATE FUNCTION MAXDATE (DATE1 DATE, DATE2 DATE) RETURNING DATE;
DEFINE DATE3 DATE;
IF DATE1 > DATE2 THEN LET DATE3 = DATE1;
ELSE LET DATE3 = DATE2;
END IF;
RETURN DATE3;
END FUNCTION;
```

3. ฟังก์ชันสำหรับหาวันที่ที่น้อยกว่า

```
CREATE FUNCTION MINDATE (DATE1 DATE, DATE2 DATE) RETURNING DATE;
DEFINE DATE3 DATE;
IF DATE1 < DATE2 THEN LET DATE3 = DATE1;
ELSE LET DATE3 = DATE2;
END IF;
RETURN DATE3;
END FUNCTION;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการตอบคำถามเชิงเวลา

ตัวอย่างการตอบคำถามเชิงเวลาโดยใช้ภาษา TSQL2 ภาษา SQL-92 บนตารางที่เป็น TNF และ ภาษา SQL-92 มีดังนี้ (สมมติให้วันที่ปัจจุบันคือวันที่ 1 เดือนมกราคม ปี ค.ศ. 1990)

1. บอกชื่อแผนกและงบประมาณของแผนกในวันที่สมชายเข้าทำงาน

คำตอบ : {'ของเล่น', 150000}, {'หนังสือ', 50000}

ภาษา TSQL2 :

```
SELECT SNAPSHOT E.DEPTNAME, D.BUDGET
FROM EMP (NAME, DEPTNAME) AS E, DEPT (DEPTNAME, BUDGET) AS D
WHERE E.NAME = 'สมชาย'
AND E.DEPTNAME = D.DEPTNAME
AND VALID(D) CONTAINS FIRST (VALID (E))
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT E.DEPTNAME, D.BUDGET
FROM EMP_DEPT E, DEPT_BUDGET D
WHERE E.ID = (SELECT ID FROM EMP_NAME WHERE NAME = 'สมชาย')
AND E.DEPTNAME = D.DEPTNAME
AND D.VALIDTimestart <= E.VALIDTimestart
AND D.VALIDTIMEEND >= E.VALIDTimestart
```

ภาษา SQL-92 :

```
SELECT E.DEPTNAME, D.BUDGET
FROM EMP E, DEPT D
WHERE E.NAME = 'สมชาย'
AND E.DEPTNAME = D.DEPTNAME
AND D.VALIDTimestart <= E.VALIDTimestart
AND D.VALIDTIMEEND >= E.VALIDTimestart
AND E.VALIDTimestart-1 NOT IN (SELECT E2.VALIDTIMEEND FROM EMP E2
WHERE E.ID = E2.ID AND E.DEPTNAME = E2.DEPTNAME)
```

2. แผนกใดมีผู้จัดการคนเดิมบริหารงานติดต่อกันสั้นที่สุด

คำตอบ : {'หนังสือ'}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา TSQL2 :

```
SELECT SNAPSHOT NAME
FROM DEPT (DEPTNAME, ID) (PERIOD) AS D
WHERE CAST (VALID (D) AS INTERVAL DAY) <=
ALL (SELECT CAST (VALID (D2) AS INTERVAL DAY)
FROM DEPT (DEPTNAME, ID) (PERIOD) AS D2)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT D1.DEPTNAME
FROM DEPT_MGR D1
WHERE D1.VALIDTIMEEND - D1.VALIDTimestart <=
ALL (SELECT D2.VALIDTIMEEND - D2.VALIDTimestart FROM DEPT_MGR D2)
```

ภาษา SQL-92 :

???

3. ใครมีเงินเดือนเท่าเดิมติดต่อกันนานที่สุด

คำตอบ : {'สมหญิง'}

ภาษา TSQL2 :

```
SELECT SNAPSHOT E2.NAME
FROM EMP (ID, SALARY) (PERIOD) AS E, E (NAME) AS E2
WHERE CAST (VALID (E) AS INTERVAL DAY) >
ALL (SELECT CAST (VALID (E3) AS INTERVAL DAY)
FROM EMP (ID, SALARY) (PERIOD) AS E3
WHERE E3.ID <> E.ID)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT NAME
FROM EMP_SALARY E1 , EMP_NAME
WHERE NOW (E1.VALIDTIMEEND) - E1.VALIDTimestart >
ALL (SELECT NOW (E2.VALIDTIMEEND) - E2.VALIDTimestart
FROM EMP_SALARY E2
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WHERE E1.ID <> E2.ID)
AND EMP_NAME.ID = E1.ID
```

ภาษา SQL-92 :

???

4. ใครมีเงินเดือนเท่าเดิมนานที่สุด

คำตอบ : {'สมชาย'}

ภาษา TSQL2 :

```
SELECT SNAPSHOT E2.NAME
FROM EMP (ID, SALARY) AS E1, E1 (NAME) AS E2
WHERE CAST (VALID (E1) AS INTERVAL DAY) >
ALL (SELECT CAST (VALID (E3) AS INTERVAL DAY)
FROM EMP (ID, SALARY) AS E3
WHERE E3.ID <> E1.ID)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
CREATE VIEW V1 (ID, SALARY, INTERVAL) AS
SELECT ID, SALARY, (SUM (NOW (VALIDTIMEEND) - VALIDTimestart))
FROM EMP_SALARY
GROUP BY ID, SALARY;
```

```
SELECT E2.NAME
FROM V1, EMP_NAME E2
WHERE E2.ID = V1.ID
AND V1.INTERVAL >= ALL (SELECT V2.INTERVAL FROM V1 V2);
```

```
DROP VIEW V1;
```

ภาษา SQL-92 :

```
CREATE VIEW V1 (ID, SALARY, INTERVAL) AS
SELECT ID, SALARY, (SUM (NOW (VALIDTIMEEND) - VALIDTimestart))
FROM EMP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
GROUP BY ID, SALARY;
```

```
SELECT DISTINCT (E2.NAME)
FROM V1, EMP E2
WHERE E2.ID = V1.ID
AND V1.INTERVAL >= ALL (SELECT V2.INTERVAL FROM V1 V2);

DROP VIEW V1;
```

5. ใครเป็นผู้จัดการแผนกหนังสือเป็นเวลาอย่างน้อย 2 ปี

คำตอบ : {(‘สมชาย’), (‘ชาติชาย')}

ภาษา TSQL2 :

```
SELECT SNAPSHOT NAME
FROM EMP (ID, NAME) AS E, DEPT (DEPTNAME, ID) AS D
WHERE E.ID = D.ID AND DEPT.DEPTNAME = ‘หนังสือ’
AND CAST (VALID (D) AS INTERVAL YEAR) >= INTERVAL ‘2’ YEAR
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT DISTINCT NAME
FROM EMP_NAME E1, DEPT_MGR D1
WHERE E1.ID = D1.ID
AND D1.DEPTNAME = ‘หนังสือ’
AND 2 <= (SELECT SUM (YEAR (NOW (VALIDTIMEEND))) – YEAR (VALIDTimestart))
FROM DEPT_MGR D2
WHERE D1.ID = D2.ID AND D1.DEPTNAME = D2.DEPTNAME
GROUP BY DEPTNAME, ID)
```

ภาษา SQL-92 :

```
SELECT DISTINCT NAME
FROM EMP E1, DEPT D1
WHERE E1.ID = D1.ID
AND D1.DEPTNAME = ‘หนังสือ’
AND 2 <= (SELECT SUM (YEAR (NOW (VALIDTIMEEND))) – YEAR (VALIDTimestart))
```

```
FROM DEPT D2
WHERE D1.ID = D2.ID AND D1.DEPTNAME = D2.DEPTNAME
GROUP BY DEPTNAME, ID)
```

6. บอกวันที่ที่สมชายมีทักษะต่างๆ

คำตอบ : {(01/01/1982), (04/01/1982), (06/01/1984), (01/01/1985)}

ภาษา TSQL2 :

```
SELECT BEGIN (S)
FROM EMP (ID, NAME) AS E, SKILL (ID, SKILL) AS S
WHERE E.ID = S.ID
AND E.NAME = 'สมชาย'
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT S1.VALIDTimestart
FROM EMP_NAME E, SKILL_SKILL S1
WHERE E.NAME = 'สมชาย'
AND E.ID = S1.ID
AND S1.VALIDTimestart-1 NOT IN (SELECT S2.VALIDTIMEEND FROM SKILL_SKILL S2
WHERE S1.ID = S2.ID AND S1.SKILL = S2.SKILL)
```

ภาษา SQL-92 :

```
SELECT DISTINCT S1.VALIDTimestart
FROM EMP E, SKILL S1
WHERE E.NAME = 'สมชาย'
AND E.ID = S1.ID
AND S1.VALIDTimestart-1 NOT IN (SELECT S2.VALIDTIMEEND FROM SKILL_SKILL S2
WHERE S1.ID = S2.ID AND S1.SKILL = S2.SKILL)
```

7. บอกช่วงเวลาในการทำงานของพนักงานที่เคยได้รับเงินเดือน 40000 บาทนานกว่าเวลาที่สมหญิงได้รับเงินเดือน 40000 บาท

คำตอบ : {(02/01/1982 – 01/31/1987)}, {(04/01/1987 – 01/01/1990)}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา TSQL2 :

```
SELECT VALID (E1)
FROM EMP (ID, SALARY) AS E1 E2, E2 (NAME) AS E3
WHERE E1.SALARY = 40000 AND E2.SALARY = 40000 AND E3.NAME = 'สมหญิง'
AND CAST (VALID (E1) AS INTERVAL DAY) > CAST (VALID (E2) AS INTERVAL DAY)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
CREATE VIEW V1 (ID, SALARY, INTERVAL) AS
SELECT ID, SALARY, (SUM (NOW (VALIDTIMEEND) - VALIDTimestart))
FROM EMP_SALARY
GROUP BY ID, SALARY;
```

```
SELECT VALIDTimestart, NOW (VALIDTIMEEND)
FROM V1 E1 , EMP_DEPT E4
WHERE E1.SALARY = 40000
AND E1.INTERVAL > (SELECT E2.INTERVAL FROM V1 E2, EMP_NAME E3
WHERE E2.ID = E3.ID AND E2.SALARY = 40000 AND E3.NAME = 'สมหญิง')
AND E1.ID = E4.ID;
```

```
DROP VIEW V1;
```

ภาษา SQL-92 :

```
CREATE VIEW V1 (ID, SALARY, INTERVAL) AS
SELECT ID, SALARY, (SUM (NOW (VALIDTIMEEND) - VALIDTimestart))
FROM EMP
GROUP BY ID, SALARY;
```

```
SELECT VALIDTimestart, NOW (VALIDTIMEEND)
FROM V1 E1, EMP_DEPT E4
WHERE E1.SALARY = 40000
AND E1.INTERVAL > (SELECT E2.INTERVAL FROM V1 E2, EMP_NAME E3
WHERE E2.ID = E3.ID AND E2.SALARY = 40000 AND E3.NAME = 'สมหญิง')
AND E1.ID = E4.ID;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DROP VIEW V1;

8. บอกช่วงเวลาที่ยับประมาณของแผนกคงที่นานกว่า 2 ปี

คำตอบ : {[01/01/1987 – 01/01/1990], [04/01/1987 – 01/01/1990]}

ภาษา TSQL2 :

SELECT VALID (D)

FROM DEPT (DEPTNAME, BUDGET) (PERIOD) AS D

WHERE CAST (VALID (D) AS INTERVAL YEAR) > INTERVAL '2' YEAR

ภาษา SQL-92 บนตารางที่เป็น TNF :

SELECT VALIDTimestart, NOW (VALIDTIMEEND)

FROM DEPT_BUDGET D

WHERE YEAR (NOW (VALIDTIMEEND)) – YEAR (VALIDTimestart) > 2

ภาษา SQL-92 :

???

9. บอกช่วงเวลาทำงานของพนักงานที่ทำงานในแผนกของเล่นนั้นติดต่อกันอย่างน้อย 8 ปี

คำตอบ : {[01/01/1982 – 01/01/1990]}

ภาษา TSQL2 :

SELECT VALID (E)

FROM EMP (ID) AS E

WHERE DEPTNAME = 'ของเล่น'

AND CAST (VALID (EMP) AS INTERVAL YEAR) >= INTERVAL '8' YEAR

ภาษา SQL-92 บนตารางที่เป็น TNF :

SELECT VALIDTimestart, NOW (VALIDTIMEEND)

FROM EMP_DEPT E1

WHERE E1.DEPTNAME = 'ของเล่น'

AND 8 <= (SELECT SUM (YEAR (NOW (VALIDTIMEEND)) – YEAR (VALIDTimestart))

FROM EMP_DEPT E2

WHERE E1.ID = E2.ID AND E2.DEPTNAME = 'ของเล่น'

GROUP BY ID, DEPTNAME)

ภาษา SQL-92 :

```
SELECT VALIDTimestart, NOW (VALIDTIMEEND)
FROM EMP E1
WHERE E1.DEPTNAME = 'ของเล่น'
AND 8 <= (SELECT SUM (YEAR (NOW (VALIDTIMEEND))) - YEAR (VALIDTimestart))
FROM EMP E2 WHERE E1.ID = E2.ID AND E2.DEPTNAME = 'ของเล่น'
GROUP BY ID, DEPTNAME)
```

10. เมื่อใดที่มีแผนกได้รับงบประมาณลดลง

คำตอบ : {(01/01/1987)}

ภาษา TSQL2 :

```
SELECT SNAPSHOT BEGIN (VALID (D2))
FROM DEPT (DEPTNAME) AS D, D (BUDGET) (PERIOD) AS D1 D2
WHERE VALID (D1) MEETS VALID (D2) AND D1.BUDGET > D2.BUDGET
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT D1.VALIDTimestart
FROM DEPT_BUDGET D1, DEPT_BUDGET D2
WHERE D1.DEPTNAME = D2.DEPTNAME
AND D1.VALIDTimestart-1 = NOW (D2.VALIDTIMEEND)
AND D1.BUDGET < D2.BUDGET
```

ภาษา SQL-92 :

```
SELECT D1.VALIDTimestart
FROM DEPT D1, DEPT D2
WHERE D1.DEPTNAME = D2.DEPTNAME
AND D1.VALIDTimestart-1 = NOW (D2.VALIDTIMEEND)
AND D1.BUDGET < D2.BUDGET
```

11. ช่วงเวลาใดที่แผนกของเล่นมีงบประมาณมากกว่า 100000 บาท

คำตอบ : {(01/01/1982 - 12/31/1986)}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา TSQL2 :

```
SELECT VALID (D)
FROM DEPT (DEPTNAME, BUDGET) AS D
WHERE D.BUDGET > 100000 AND D.DEPTNAME = 'ของเล่น'
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT VALIDTimestart, NOW (VALIDTIMEEND)
FROM DEPT_BUDGET D
WHERE D.BUDGET > 100000 AND D.DEPTNAME = 'ของเล่น'
```

ภาษา SQL-92 :

```
SELECT VALIDTimestart, NOW (VALIDTIMEEND)
FROM DEPT D
WHERE D.BUDGET > 100000 AND D.DEPTNAME = 'ของเล่น'
```

12. บอกชื่อแผนก ชื่อผู้จัดการ และช่วงเวลาทำงานของแผนกที่มีผู้จัดการที่ทำงานเป็นเวลาด้านที่สุด
คำตอบ : {(‘หนังสือ’, ‘สมชาย’, [04/01/1987 – 12/31/1987]) และ (‘หนังสือ’, ‘ชาติชาย’, [01/01/1988 – 01/01/1990])}

ภาษา TSQL2 :

```
SELECT D1.DEPTNAME, E1.NAME
FROM DEPT (DEPTNAME, ID) AS D1, EMP AS E1
WHERE E1.ID = D1.ID
AND CAST (VALID (D1) AS INTERVAL DAY) =
(SELECT MIN (CAST (VALID (D3) AS INTERVAL DAY)) FROM DEPT (ID) AS D3)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
CREATE VIEW V1 (DEPTNAME, ID, INTERVAL) AS
SELECT DEPTNAME, ID, (SUM (NOW (VALIDTIMEEND) - VALIDTimestart))
FROM DEPT_MGR
GROUP BY DEPTNAME, ID;
```

SELECT D1.DEPTNAME, E1.NAME, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

FROM V1 D1, EMP E1
WHERE E1.ID = D1.ID
AND D1.INTERVAL = (SELECT MIN (D3.INTERVAL) FROM V1 D3)
AND E1.DEPTNAME = D1.DEPTNAME;

```

```
DROP VIEW V1;
```

ภาษา SQL-92 :

```

CREATE VIEW V1 (DEPTNAME, ID, INTERVAL) AS
SELECT DEPTNAME, ID, (SUM (NOW (VALIDTIMEEND) - VALIDTimestart))
FROM DEPT
GROUP BY DEPTNAME, ID;

```

```

SELECT D1.DEPTNAME, E1.NAME, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)
FROM V1 D1, EMP E1
WHERE E1.ID = D1.ID
AND D1.INTERVAL = (SELECT MIN (D3.INTERVAL) FROM V1 D3)
AND E1.DEPTNAME = D1.DEPTNAME;

```

```
DROP VIEW V1;
```

13. บอกเงินเดือนที่มากที่สุดที่น้อยกว่า 50000 บาทที่สมชายได้และช่วงเวลาด้วย
คำตอบ : {(40000 | [02/01/1985 – 01/31/1987]) และ (40000 | [04/01/1987 – 01/01/1990])}

ภาษา TSQL2 :

```

SELECT MAX (E1.SALARY)
FROM EMP AS E1, EMP AS E2
WHERE E2.NAME = 'สมชาย' AND E1.ID = E2.ID
GROUP BY E1.ID
HAVING MAX (E1.SALARY) < 50000

```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```

SELECT DISTINCT E1.SALARY, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)
FROM EMP SALARY E1, EMP_NAME E2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WHERE E2.NAME = 'สมชาย' AND E1.ID = E2.ID
AND E1.SALARY = (SELECT MAX (SALARY) FROM EMP_SALARY E3 WHERE E3.ID = E2.ID
AND E3.SALARY < 50000)
```

ภาษา SQL-92 :

```
SELECT DISTINCT E1.SALARY, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)
FROM EMP E1, EMP E2
WHERE E2.NAME = 'สมชาย' AND E1.ID = E2.ID
AND E1.SALARY = (SELECT MAX (SALARY) FROM EMP_SALARY E3 WHERE E3.ID = E2.ID
AND E3.SALARY < 50000)
```

14. บอกชื่อ เงินเดือน และช่วงเวลาที่ มีพนักงานมีทักษะขับรถและมีเงินเดือนน้อยกว่า 40000 บาท

คำตอบ : {('สมชาย', 20000 | [02/01/1982 – 05/01/1982] } และ { 'สมชาย', 30000 | [06/01/1984 – 01/31/1985] }

ภาษา TSQL2 :

```
SELECT EMP.NAME, EMP.SALARY
FROM EMP, SKILL
WHERE EMP.ID = SKILL.ID AND SKILL.SKILL = 'ขับรถ' AND EMP.SALARY < 40000
AND VALID (EMP) OVERLAPS VALID (SKILL)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT E1.NAME, E2.SALARY, MAXDATE (E2.VALIDTimestart, S1.VALIDTimestart),
MINDATE (NOW (E2.VALIDTIMEEND), NOW (S1.VALIDTIMEEND))
FROM EMP_NAME E1, EMP_SALARY E2, SKILL_SKILL S1
WHERE E1.ID = S1.ID AND E2.ID = S1.ID
AND S1.SKILL = 'ขับรถ' AND E2.SALARY < 40000
AND E1.VALIDTimestart <= S1.VALIDTIMEEND
AND S1.VALIDTimestart <= E1.VALIDTIMEEND
AND E2.VALIDTimestart <= S1.VALIDTIMEEND
AND S1.VALIDTimestart <= E2.VALIDTIMEEND
AND E2.VALIDTimestart <= E1.VALIDTIMEEND
AND E1.VALIDTimestart <= E2.VALIDTIMEEND
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา SQL-92 :

```
SELECT E1.NAME, E1.SALARY, MAXDATE (E1.VALIDTimestart, S1.VALIDTimestart),
MINDATE (NOW (E1.VALIDTIMEEND), NOW (S1.VALIDTIMEEND))
FROM EMP E1, SKILL_SKILL S1
WHERE E1.ID = S1.ID
AND S1.SKILL = 'ขับรถ' AND E1.SALARY < 40000
AND E1.VALIDTimestart <= S1.VALIDTIMEEND
AND S1.VALIDTimestart <= E1.VALIDTIMEEND
```

15. บอกชื่อ เงินเดือน และช่วงเวลาที่ มีพนักงาน ได้เงินเดือนมากกว่าผู้จัดการ

คำตอบ : {'สมชาย', 30000 | [06/01/1982 – 07/31/1984]} และ {'สมชาย', 40000 | 02/01/1985 – 08/31/1986}]}

ภาษา TSQL2 :

```
SELECT E.NAME, E.SALARY
FROM EMP (ID, NAME, SALARY) AS E, EMP (ID, SALARY) AS Mgr, DEPT AS D1
WHERE E.DEPTNAME = D1.DEPTNAME AND VALID (E) OVERLAPS VALID (D1)
AND D1.ID = Mgr.ID AND VALID (D1) OVERLAPS VALID (Mgr)
AND E.SALARY = Mgr.SALARY AND E.ID <> Mgr.ID
AND VALID (E) OVERLAPS VALID (Mgr)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

???

ภาษา SQL-92 :

???

16. ช่วงเวลาใดที่แผนกของเล่นมีงบประมาณมากกว่า 175000 บาทติดต่อกันเป็นเวลายาวนานมากกว่า 1 ปี และงบประมาณนั้นเท่ากับเท่าไร

คำตอบ : {(200000 | [08/01/1984 – 12/31/1986])}

ภาษา TSQL2 :

```
SELECT D1.BUDGET
```

```
FROM DEPT (DEPTNAME, BUDGET) (PERIOD) AS D1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WHERE D1.DEPTNAME = 'ของเล่น' AND D1.BUDGET > 175000
AND CAST (VALID (D1) AS INTERVAL YEAR) > INTERVAL '1' YEAR
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT D1.BUDGET, D1.VALIDTimestart, D1.VALIDTIMEEND
FROM DEPT_BUDGET D1
WHERE D1.DEPTNAME = 'ของเล่น' AND D1.BUDGET > 175000
AND YEAR (NOW (D1.VALIDTIMEEND)) - YEAR (D1.VALIDTimestart) > 1
```

ภาษา SQL-92 :

???

17. บอกชื่อและเงินเดือนทั้งหมดของพนักงานที่เกิดหลังวันที่ 1 เดือนมกราคม ปี ค.ศ. 1956

คำตอบ : {(‘สมหญิง’, 30000 | [01/01/1982 – 07/31/1984]), (‘สมหญิง’, 40000 | [08/01/1984 – 08/31/1986]), (‘สมหญิง’, 50000 | [09/01/1986 – 01/01/1990])}

ภาษา TSQL2 :

```
SELECT E1.NAME, E1.SALARY
FROM EMP AS E1
WHERE E1.D-BIRTH > DATE '01/01/1956'
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT E1.NAME, E2.SALARY, E2.VALIDTimestart, NOW (E2.VALIDTIMEEND)
FROM EMP_NAME E1, EMP_SALARY E2, EMP_NOTIME E3
WHERE E1.ID = E2.ID AND E1.ID = E3.ID AND E3.D-BIRTH > '01/01/1956'
```

ภาษา SQL-92 :

```
SELECT E1.NAME, E1.SALARY, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)
FROM EMP E1
WHERE E1.D-BIRTH > '01/01/1956'
```

18. บอกเงินเดือนทั้งหมดของสมชาย

คำตอบ : {(20000 | [02/01/1982 – 05/31/1982]), (30000 | 06/01/01982 – 01/31/1985), (40000 | [02/01/1985 – 01/31/1987] \cup [04/01/1987 – 01/01/1990])}

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา TSQL2 :

```
SELECT E1.SALARY
FROM EMP AS E1, EMP AS E2
WHERE E2.NAME = 'สมชาย' AND E1.ID = E2.ID AND VALID (E1) OVERLAPS VALID (E2)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT E1.SALARY, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)
FROM EMP_SALARY E1, EMP_NAME E2
WHERE E2.NAME = 'สมชาย' AND E1.ID = E2.ID
AND E1.VALIDTimestart <= E2.VALIDTIMEEND
AND E2.VALIDTimestart <= E1.VALIDTIMEEND
```

ภาษา SQL-92 :

```
SELECT E1.SALARY, E1.VALIDTimestart, NOW (E1.VALIDTIMEEND)
FROM EMP E1
WHERE E1.ID IN (SELECT ID FROM EMP WHERE NAME = 'สมชาย')
```

19. บอกชื่อ และเงินเดือนของพนักงานที่เคยเป็นผู้จัดการและได้เงินเดือนอย่างน้อย 36000 บาท

คำตอบ : {(‘สมหญิง’, 40000 | [08/01/1984 – 08/31/1986]), (‘สมหญิง’, 50000 | [09/01/1986 – 01/01/1990]), (‘สมชาย’, 40000 | [04/01/1987 – 12/31/1987]), (‘ชาติชาย’, 40000 | [01/01/1988 – 01/01/1990])}

ภาษา TSQL2 :

```
SELECT EMP.NAME, EMP.SALARY
FROM EMP, DEPT
WHERE EMP.SALARY > 36000 AND EMP.ID = DEPT.ID
AND VALID (EMP) OVERLAPS VALID (DEPT)
```

ภาษา SQL-92 บนตารางที่เป็น TNF :

```
SELECT E1.NAME, E2.SALARY, MAXDATE (E1.VALIDTimestart, E2.VALIDTimestart),
MINDATE (E1.VALIDTIMEEND, E2.VALIDTIMEEND)
FROM EMP_NAME E1, EMP_SALARY E2
WHERE E2.SALARY > 36000
```

ในภาษา SQL-92 ในการตอบคำถามเชิงเวลาบางครั้งจะได้ช่วงเวลาที่ไม่วางกัน เช่น คำตอบควรจะเป็น (20000 | [01/01/1982 – 12/31/1986]) แต่จะได้เป็น (20000 | [01/01/1982 – 12/31/1984]), (20000 | [01/01/1985 – 12/31/1986]) ซึ่งอาจทำให้การสื่อความหมายผิดไปได้ และภาษา SQL-92 ไม่สามารถหาช่วงเวลาที่ต่อเนื่องกันได้ ทำให้ตอบคำถามไม่ครอบคลุมทั้งหมด เช่น ตอบคำถามที่ว่า “ใครมีเงินเดือนเท่าเดิมนานที่สุด” ได้ แต่ไม่สามารถตอบคำถามที่ว่า “ใครมีเงินเดือนเท่าเดิมนานที่สุด” ได้ แต่ถ้าเป็นภาษา SQL-92 บนตารางที่เป็น TNF จะสามารถตอบได้ครอบคลุมมากกว่า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] Abdullah Uz Tansel, James Clifford, Shashi Gadia, Sushil Jajodia, Arie Segev, Richard Snodgrass (1993) : “*Temporal Databases : Theory, Design, and Implementation*”, “The Benjamin/Cummings Publishing Company, Inc.”, 1993.
- [2] Richard T. Snodgrass (1998) : “*Managing Temporal Data : A Five-Part Series*”, “A TimeCenter Technical Report”, 1998.
- [3] Richard T. Snodgrass, Ilsoo Ahn, Gad Ariav, Petra Bayer, James Clifford, Curtis Dyreson, Fabio Grandi, Luis Hermosilla, Christian S. Jensen, Wolfgang Kafer, Nick Kline, T.Y. Cliff Leung, Nikos Lorentzos, Y. Mitsopoulos, John F. Roddick, Michael Soo, Suryanarayana M. Sripada (1994) : “*An Evaluation of TSQL2*”, “The TSQL2 Commentary”, 1994.
- [4] Richard T. Snodgrass (1996) : “*A Road Map of Additions to SQL/Temporal*”, 1996.
- [5] Richard T. Snodgrass, Michael H. Bohlen, Christian S. Jensen, Andreas Steiner (1996) : “*Adding Valid Time to SQL/Temporal*”, “ANSI Expert's Contribution”, 1996.
- [6] Richard T. Snodgrass, Michael H. Bohlen, Christian S. Jensen, Andreas Steiner (1996) : “*Adding Transaction Time to SQL/Temporal*”, “ANSI Expert's Contribution”, 1996.
- [7] Richard T. Snodgrass, Ilsoo Ahn, Gad Ariav, Petra Bayer, James Clifford, Curtis Dyreson, Fabio Grandi, Luis Hermosilla, Christian S. Jensen, Wolfgang Kafer, Nick Kline, T.Y. Cliff Leung, Nikos Lorentzos, Y. Mitsopoulos, John F. Roddick, Michael Soo, Suryanarayana M. Sripada (1994) : “*TSQL2 Language Specification*”, “The TSQL2 Commentary”, 1994.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้