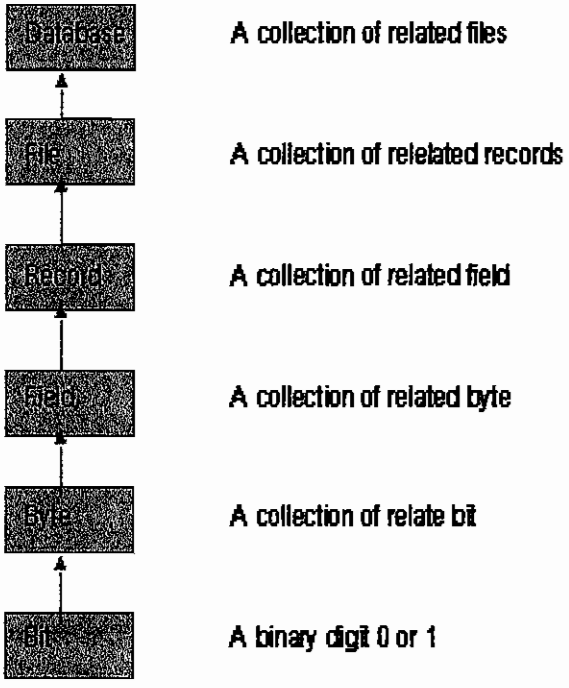
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1. ระบบฐานข้อมูล

ระบบฐานข้อมูล คือ ระบบที่รวบรวมข้อมูลไว้ในที่เดียวกันและถูกจัดการภายใต้ระบบเดียวกัน จากข้อมูลที่มีการเก็บรวบรวมไว้มีจำนวนมากขึ้น โดยข้อมูลทั้งหมดนั้นจะเก็บรวบรวมเป็นแฟ้มข้อมูล (File) ไว้ในระบบคอมพิวเตอร์ ทำให้โปรแกรมคอมพิวเตอร์สามารถเข้าไปดึงข้อมูลที่ต้องการได้อย่างรวดเร็ว ซึ่งอาจเปรียบฐานข้อมูลเสมือนเป็น electronic filing system



รูปที่ 2.1.1 electronic filing system

(บัณฑิต, 2547)

2.1.1. แฟ้มข้อมูล

การประมวลผลข้อมูลเพื่อให้ได้สารสนเทศจะมีองค์ประกอบที่สำคัญ คือ แฟ้มข้อมูล ความหมายของแฟ้มข้อมูลหนึ่งๆนั้น มักจะเป็นเอกสารที่เป็นเรื่องเดียวกันและจัดเก็บรวบรวมไว้เป็นแฟ้มข้อมูล เพื่อสะดวกในการค้นหาข้อมูล เช่น แฟ้มข้อมูลประวัติพนักงาน การเก็บรวบรวมข้อมูลในรูปของเอกสารเพื่อประโยชน์ในการใช้งาน

ข้อมูล หมายถึง กลุ่มของสารสนเทศที่สัมพันธ์กัน ความสัมพันธ์ของกลุ่มสารสนเทศหรือข้อมูลนั้น ถูกกำหนดโดยผู้ใช้แฟ้มข้อมูล ข้อมูลเป็นวัตถุดิบสำคัญในการประมวลผลข้อมูลทั้งหมดของคอมพิวเตอร์ (บัณฑิต, 2547) ซึ่งประกอบด้วย บิต (bit) ซึ่งเป็นโครงสร้างที่เล็กที่สุด ในแต่ละบิตจะเป็นตัวเลขในระบบเลขฐานสอง ประกอบด้วย 0 และ 1 ซึ่งนำมาใช้แทนระหว่าง 2 สถานะ เช่น จริง-เท็จ เปิด-ปิด เป็นต้น

การที่เก็บข้อมูลต่างๆ โดยให้คอมพิวเตอร์แสดงออกมาในรูปอักขระตัวอักษร จะต้องนำบิต หลายๆ บิต มาเรียงต่อกัน คือการนำ บิต จำนวน 8 บิต มาเรียงเป็น 1 ชุด เรียกว่า 1 ไบต์

เช่น 10100001 หมายถึง ก

10100010 หมายถึง ข

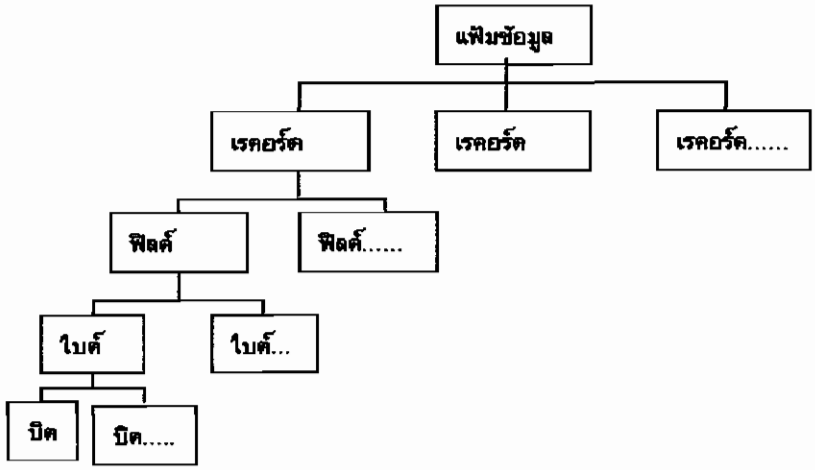
เมื่อเรานำ ไบต์ (byte) หลายๆ ไบต์ มาเรียงต่อกัน เรียกว่า เขตข้อมูล (field) เช่น Name ใช้เก็บชื่อ LastName ใช้เก็บนามสกุล เป็นต้น

เมื่อนำเขตข้อมูลหลายๆเขตข้อมูลมาเรียงต่อกัน เรียกว่า ระเบียน(record) เช่นระเบียนที่ 1 เก็บชื่อ นามสกุล วันเดือนปีเกิด ของ นักเรียนคนที่ 1 เป็นต้น

การเก็บระเบียนหลายๆระเบียนรวมกัน เรียกว่า แฟ้มข้อมูล (File) เช่น แฟ้มข้อมูลของนักเรียนจะเก็บ ชื่อ นามสกุล วันเดือนปีเกิดของนักเรียน จำนวน 500 คน เป็นต้น

การจัดเก็บแฟ้มข้อมูลหลายๆแฟ้มข้อมูลไว้ในระบบเดียวกัน เรียกว่า ฐานข้อมูล (database) เช่น เก็บแฟ้มข้อมูล นักเรียน อาจารย์ วิชาที่เปิดสอน เป็นต้น

การเข้าถึงข้อมูลในฐานข้อมูลจึงจำเป็นต้องมีระบบการจัดการฐานข้อมูลมาช่วย เรียกว่า database management system (DBMS) ซึ่งจะช่วยให้ผู้ใช้สามารถจัดการกับข้อมูลตามความต้องการได้ ในหน่วยงานใหญ่ๆอาจมีฐานข้อมูลมากกว่า 1 ฐานข้อมูล เช่น ฐานข้อมูลบุคลากร ฐานข้อมูลลูกค้า ฐานข้อมูลสินค้า เป็นต้น ซึ่งจะแสดงรูปแบบโครงสร้างของแฟ้มข้อมูล (บัณฑิต, 2547) ดังรูปที่ 2.1.2



รูปที่ 2.1.2 แสดงโครงสร้างของแฟ้มข้อมูล

(บัณฑิต, 2547)

2.1.1.1. ประเภทของแฟ้มข้อมูล

ประเภทของแฟ้มข้อมูลจำแนกตามลักษณะของการใช้งานได้ดังนี้

1. แฟ้มข้อมูลหลัก (master file) แฟ้มข้อมูลหลักเป็นแฟ้มข้อมูลที่บรรจุข้อมูลพื้นฐานที่จำเป็นสำหรับระบบงาน และเป็นข้อมูลหลักที่เก็บไว้ใช้ประโยชน์ข้อมูลเฉพาะเรื่อง ไม่มีรายการเปลี่ยนแปลงในช่วงปัจจุบัน มีสภาพค่อนข้างคงที่ไม่เปลี่ยนแปลงหรือเคลื่อนไหวบ่อยแต่จะถูกเปลี่ยนแปลงเมื่อมีการสิ้นสุดของข้อมูล เป็นข้อมูลที่สำคัญที่เก็บไว้ใช้ประโยชน์ ตัวอย่างเช่น แฟ้มข้อมูลหลักของนักศึกษาจะแสดงรายละเอียดของนักศึกษา ซึ่งมี ชื่อ นามสกุล ที่อยู่ ผลการศึกษา หรือแฟ้มข้อมูลหลักของลูกค้า ในแต่ละระเบียนของแฟ้มข้อมูลจะแสดงรายละเอียดของลูกค้า เช่น ชื่อ สกุล ที่อยู่ หรือประเภทของลูกค้า (บัณฑิต จามรภูมิ, 2547)

2. แฟ้มข้อมูลรายการเปลี่ยนแปลง (transaction file) แฟ้มข้อมูลรายการเปลี่ยนแปลงเป็นแฟ้มข้อมูลที่ประกอบด้วยระเบียนข้อมูลที่มีการเคลื่อนไหว ซึ่งจะถูกรวบรวมเป็นแฟ้มข้อมูลรายการเปลี่ยนแปลงที่เกิดขึ้นในแต่ละงวดในส่วนที่เกี่ยวข้องกับข้อมูลนั้น แฟ้มข้อมูลรายการเปลี่ยนแปลงนี้จะนำไปปรับรายการในแฟ้มข้อมูลหลัก ให้ได้ยอดปัจจุบัน ตัวอย่างเช่น แฟ้มข้อมูลลงทะเบียนเรียนของนักศึกษา (บัณฑิต, 2547)

3. แฟ้มข้อมูลตาราง (table file) แฟ้มข้อมูลตารางเป็นแฟ้มข้อมูลที่มีค่าคงที่ ซึ่งประกอบด้วยตารางที่เป็นข้อมูล หรือชุดของข้อมูลที่มีความเกี่ยวข้องกันและถูกจัดให้อยู่รวมกันอย่างมีระเบียบ โดยแฟ้มข้อมูลตารางนี้จะถูกใช้ในการประมวลผลกับแฟ้มข้อมูลอื่นเป็นประจำอยู่เสมอ เช่น ตารางอัตราภาษี ตารางราคาสินค้า (บัณฑิต, 2547)

4. แฟ้มข้อมูลเรียงลำดับ (sort file) แฟ้มข้อมูลเรียงลำดับเป็นการจัดเรียงระเบียนที่จะบรรจุในแฟ้มข้อมูลนั้นใหม่ โดยเรียงตามลำดับค่าของฟิลด์ข้อมูลหรือค่าของข้อมูลค่าใดค่าหนึ่งในระเบียนนั้นก็ได้ เช่น จัดเรียงลำดับตาม วันเดือนปี ตามลำดับตัวอักขระเรียงลำดับจากมากไปหาน้อย หรือจากน้อยไปหามาก เป็นต้น (บัณฑิต, 2547)

5. แฟ้มข้อมูลรายงาน (report file) เป็นแฟ้มข้อมูลที่ถูกจัดเรียงระเบียบตามรูปแบบของรายงาน ที่ต้องการแล้วจัดเก็บไว้ในรูปของแฟ้มข้อมูล ตัวอย่าง เช่น แฟ้มข้อมูลรายงานควบคุมการปรับเปลี่ยนข้อมูลที่เกิดขึ้นในขณะปฏิบัติงานแต่ละวัน (บัณฑิต, 2547)

2.1.1.2. การจัดโครงสร้างแฟ้มข้อมูล (file organization)

เป็นการจัดเก็บข้อมูลลงในแฟ้มข้อมูล โดยกำหนดวิธีการที่ระเบียนจะถูกจัดเก็บอยู่ในแฟ้มข้อมูลบนอุปกรณ์ที่ใช้เก็บข้อมูล ซึ่งลักษณะโครงสร้างของระเบียนจะถูกจัดเก็บไว้เป็นระบบ โดยมีวัตถุประสงค์เพื่อให้การจัดเก็บข้อมูล และการเข้าถึงข้อมูลมีความสะดวกรวดเร็ว การจัดโครงสร้างแฟ้มข้อมูลอาจแบ่งได้เป็น 3 ลักษณะ คือ

1. โครงสร้างของแฟ้มข้อมูลแบบลำดับ (sequential file) เป็นการจัดแฟ้มข้อมูลซึ่งระเบียนภายในแฟ้มข้อมูลจะถูกบันทึกโดยเรียงตามลำดับคีย์ฟิลด์ หรืออาจจะไม่เรียงลำดับตามคีย์ฟิลด์ก็ได้ ข้อมูลจะถูกบันทึกลงในสื่อบันทึกข้อมูลโดยจะถูกบันทึกไว้ในตำแหน่งที่อยู่ติดๆกัน การนำข้อมูลมาใช้ของโครงสร้างแฟ้มข้อมูลแบบลำดับจะต้องอ่านข้อมูลไปตามลำดับ จะเข้าถึงข้อมูลโดยตรงไม่ได้ (บัณฑิต, 2547)

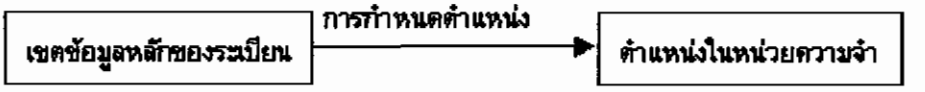
2. โครงสร้างของแฟ้มข้อมูลแบบลำดับตามดัชนี (index sequential file) เป็นวิธีการเก็บข้อมูลโดยแต่ละระเบียนในแฟ้มข้อมูลจะมีค่าของคีย์ฟิลด์ที่ใช้เป็นตัวระบุระเบียนนั้น ค่าคีย์ฟิลด์ของแต่ละระเบียนจะต้องไม่ซ้ำกับค่าคีย์ฟิลด์ในระบบอื่นๆในแฟ้มข้อมูลเดียวกัน เพราะการจัดโครงสร้างแฟ้มข้อมูลแบบนี้จะใช้คีย์ฟิลด์เป็นตัวเข้าถึงข้อมูล การเข้าถึงข้อมูลหรือการอ่านระเบียนใดๆจะเข้าถึงได้อย่างสุ่ม การจัดโครงสร้างแฟ้มข้อมูลต้องบันทึกลงสื่อข้อมูลที่เข้าถึงข้อมูลได้โดยตรง เช่น จานแม่เหล็ก การสร้างแฟ้มข้อมูลประเภทนี้ไม่ว่าจะสร้างครั้งแรกหรือสร้างใหม่ ข้อมูลแต่ละระเบียนต้องมีฟิลด์หนึ่งใช้เป็นคีย์ฟิลด์ของข้อมูล ระบบปฏิบัติการจะนำคีย์ฟิลด์ของข้อมูลไปสร้างเป็นตาราง

ดัชนีทำให้สามารถเข้าถึงระเบียนได้เร็ว นอกจากจะเข้าถึงระเบียนใดๆได้เร็วขึ้นแล้ว ยังมีประโยชน์สามารถเพิ่มระเบียนเข้าในส่วนใดๆของแฟ้มข้อมูลได้ โครงสร้างแฟ้มข้อมูลแบบลำดับตามดัชนีประกอบด้วย

ดัชนี (index) ของแฟ้มข้อมูล จะเก็บค่าคีย์ฟิลด์ของข้อมูล และที่อยู่ในหน่วยความจำ (address) ที่ระเบียนนั้นถูกนำไปบันทึกไว้ ซึ่งดัชนีนี้จะต้องเรียงลำดับจากน้อยไปมากหรือจากมากไปน้อย โดยที่ส่วนของดัชนีจะมีตัวบ่งชี้ไปยังที่อยู่ในหน่วยความจำ เพื่อจะได้นำไปถึงระเบียนข้อมูลในข้อมูลหลัก

ข้อมูลหลัก(data area) จะเก็บระเบียนข้อมูล ซึ่งระเบียนนั้นอาจจะเรียงตามลำดับจากน้อยไปมากหรือมากไปน้อย ในการจัดลำดับของข้อมูลหลักอาจจะจัดข้อมูลออกไปกลุ่มๆ โดยจะเว้นที่ไว้เพื่อให้มีการปรับปรุงแฟ้มข้อมูลได้

3. โครงสร้างของแฟ้มข้อมูลแบบสัมพัทธ์ (relative file) เป็นโครงสร้างที่สามารถเข้าถึงข้อมูลหรืออ่านระเบียนใดๆได้โดยตรง วิธีนี้เป็นการจัดเรียงข้อมูลเข้าไปในแฟ้มข้อมูลโดยอาศัยฟิลด์ข้อมูลเป็นตัวกำหนดตำแหน่งของระเบียนนั้นๆ โดยค่าของคีย์ฟิลด์ข้อมูลในแต่ละระเบียนของแฟ้มข้อมูลจะมีความสัมพัทธ์กับตำแหน่งที่ระเบียนนั้นถูกจัดเก็บไว้ในหน่วยความจำ ค่าความสัมพัทธ์นี้เป็นการกำหนดตำแหน่ง (mapping function) ดังรูที่ 2.1.3 ซึ่งเป็นฟังก์ชั่นที่ใช้ในการเปลี่ยนแปลงคีย์ฟิลด์ของระเบียนให้เป็นตำแหน่งในหน่วยความจำ โดยที่การจัดเรียงลำดับที่ของระเบียน ไม่จำเป็นต้องมีความสัมพันธ์กับการจัดลำดับที่ของระเบียนที่ถูกจัดเก็บไว้ในหน่วยความจำ ดังรูปที่ 2.1.3 (บัณฑิต, 2547)



รูปที่ 2.1.3 การกำหนดตำแหน่ง (mapping function)
(บัณฑิต, 2547)

การจัดเก็บข้อมูลลงแฟ้มข้อมูลแบบสัมพัทธ์ (relative file) จะถูกจัดเก็บอยู่บนสื่อที่สามารถเข้าถึงได้โดยตรง เช่น แผ่นจานแม่เหล็ก ลักษณะโครงสร้างแฟ้มข้อมูลแบบสัมพัทธ์จะประกอบด้วยตำแหน่งในหน่วยความจำ ซึ่งเกิดจากนำคีย์ฟิลด์ของระเบียนมาทำการกำหนดตำแหน่ง ซึ่งการกำหนดตำแหน่งนี้จะทำการปรับเปลี่ยนค่าคีย์ฟิลด์ของระเบียนใ ห้เป็นตำแหน่งในหน่วยความจำที่คำนวณได้ แฟ้มข้อมูลหลักแฟ้มข้อมูลนี้ประกอบด้วยระเบียนที่จัดเรียงตามตำแหน่งในหน่วยความจำ โดยจะเรียงจากระเบียนที่ 1 จนถึง N แต่จะไม่เรียงลำดับตามค่าของคีย์ฟิลด์

2.1.2. การจัดทำระบบฐานข้อมูล

การจัดข้อมูลให้เป็นระบบฐานข้อมูลทำให้ข้อมูลมีส่วนดีมากกว่าการเก็บข้อมูลในรูปของแฟ้มข้อมูล คือ ลดการเก็บข้อมูลที่ซ้ำซ้อน, ทำให้ข้อมูลบูรณะกลับสู่สภาพปกติได้เร็วและมีมาตรฐาน, มีการป้องกันและรักษาความปลอดภัยให้กับข้อมูล, สามารถใช้ข้อมูลร่วมกันได้, มีความเป็นอิสระของข้อมูล, สามารถขยายงานได้ง่าย และมีการรักษาความถูกต้องของข้อมูล เป็นต้น

2.1.2.1. การออกแบบฐานข้อมูล

ในการสร้างฐานข้อมูลขึ้นมาใช้งานในองค์การหรือหน่วยงานต่างๆ ต้องมีขั้นตอนดำเนินการและวิธีการจัดการเกี่ยวกับข้อมูลที่เหมาะสม โดยปกติการสร้างฐานข้อมูลจำเป็นจะต้องออกแบบฐานข้อมูลเป็นสองระยะหรือสองขั้นตอนด้วยกัน ขั้นแรกก็คือการออกแบบเชิงแนวคิด (conceptual design) หรือเชิงตรรกะ (logical design) ซึ่งเป็นมุมมองของผู้ใช้เกี่ยวกับฐานข้อมูล และขั้นที่สองก็คือการออกแบบเชิงกายภาพ (physical design) ซึ่งเป็นมุมมองของระบบปฏิบัติการเกี่ยวกับฐานข้อมูล (บัณฑิต, 2547)

1. การออกแบบเชิงตรรกะ เน้นในด้านการจัดกลุ่มข้อมูลในฐานข้อมูลให้เป็นหมวดหมู่ หรือเป็นตารางที่เหมาะสม การออกแบบเริ่มต้นด้วยการพิจารณาว่าหน่วยงานจะต้องใช้ข้อมูลอะไรบ้าง ข้อมูลเหล่านั้นมีความสัมพันธ์กันอย่างไรบ้าง จะจัดกลุ่มข้อมูลอย่างไรจึงจะเหมาะสมและไม่เกิดความซ้ำซ้อน การพิจารณาการจัดกลุ่มนี้จะต้องคำนึงถึงลักษณะของประเภทฐานข้อมูลที่จะจัดทำขึ้นด้วย (บัณฑิต, 2547)

2. การออกแบบเชิงกายภาพ เน้นในด้านการกำหนดว่าข้อมูลแต่ละรายการ หรือตารางข้อมูลต่างๆ จะจัดเก็บลงในสื่อข้อมูลเช่นจานแม่เหล็กได้อย่างไร มีการกำหนดว่าข้อมูลแต่ละรายการเป็นข้อมูลประเภทอักขระ จำนวน หรือประเภทอื่นๆ และต้องใช้เนื้อที่ในการเก็บมากน้อยเท่าใด การออกแบบฐานข้อมูลในส่วนนี้ จำเป็นที่จะต้องใช้ผู้เชี่ยวชาญที่ศึกษาด้านฐานข้อมูลมาโดยตรง (บัณฑิต, 2547)

2.1.2.2. คุณสมบัติหลักของฐานข้อมูล

คุณสมบัติหลักของฐานข้อมูลไม่ว่าเป็นฐานข้อมูลใดๆ จะมีคุณสมบัติหลักดังนี้ คือ

1. โครงสร้างข้อมูล (data structures) คือ โครงสร้างข้อมูลระดับตรรกะที่โปรแกรมประยุกต์เห็น เป็นการบอกว่าฐานข้อมูลนั้นๆมีโครงสร้างข้อมูลเป็นอย่างไร

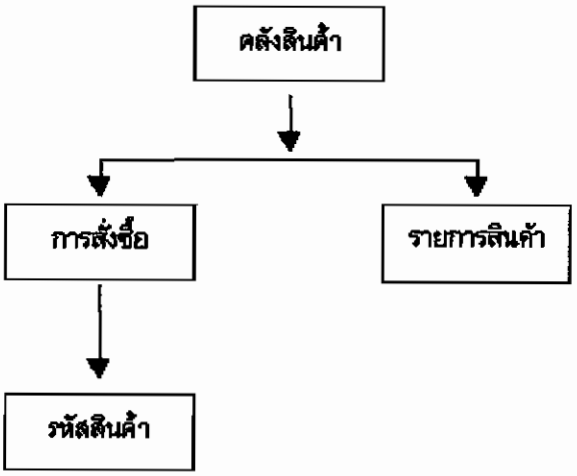
2. กฎควบคุมความถูกต้อง (integrity constraint) เป็นการบอกว่าโครงสร้างข้อมูลนั้นมีกฎบังคับความถูกต้องอย่างไร

3. ภาษาจัดการข้อมูล (data manipulation language) เป็นการบอกว่ามีภาษาจัดการข้อมูลบนโครงสร้างข้อมูลเป็นอย่างไร

2.1.2.3. ประเภทของฐานข้อมูล

ฐานข้อมูลที่รู้จักกันในปัจจุบันมีด้วยกัน 4 ประเภท คือ ฐานข้อมูลแบบลำดับชั้น (hierarchy model) , ฐานข้อมูลแบบเครือข่าย (hierarchical model) , ฐานข้อมูลแบบสัมพันธ์ (relational model) และฐานข้อมูลแบบจำลองเชิงวัตถุ (object oriented model)
(บัณฑิต, 2547)

1. ฐานข้อมูลแบบลำดับชั้น (hierarch model) ฐานข้อมูลแบบนี้จะมีโครงสร้างของข้อมูล เป็นลักษณะความสัมพันธ์แบบพ่อ-ลูก คือ พ่อ (parent) 1 คนมีลูก (child) 1 คนหรือหลายคนก็ได้ (นั่นคือเป็นความสัมพันธ์แบบ 1 ต่อ n) ดังรูปที่ 2.1.4



รูปที่ 2.1.4 แสดงโครงสร้างฐานข้อมูลแบบลำดับขั้น
(บัณฑิต, 2547)

ในฐานข้อมูลแบบลำดับชั้นจะมีปัญหา ถ้าความสัมพันธ์ของข้อมูลเป็นแบบลูกมีพ่อได้หลายคนจะใช้โครงสร้างฐานข้อมูลแบบลำดับชั้นไม่ได้ เพราะโครงสร้างฐานข้อมูลแบบลำดับชั้นไม่สามารถออกแบบลักษณะข้อมูลแบบนี้ได้ ปัญหาเช่นนี้ทำให้ไม่ค่อยมีผู้นิยมใช้ฐานข้อมูลแบบลำดับชั้น

ฐานข้อมูลแบบลำดับชั้นจะมีกฎควบคุมความถูกต้อง คือ เรคอร์ดพ่อสามารถมีเรคอร์ดลูกได้หลายเรคอร์ด แต่เรคอร์ดลูกแต่ละเรคอร์ดจะมีเรคอร์ดพ่อได้เพียงเรคอร์ดเดียวเท่านั้น
(บัณฑิต, 2547)

การจัดการรูปของโครงสร้างฐานข้อมูลแบบลำดับชั้น จะใช้ภาษาปฏิบัติการทีละเรคอร์ด ตัวอย่างของภาษาปฏิบัติการของฐานข้อมูลนี้ ได้แก่ IMS/VS

ฐานข้อมูลแบบนี้มีลักษณะเด่นและข้อจำกัดของการจัดการฐานข้อมูล ดังนี้

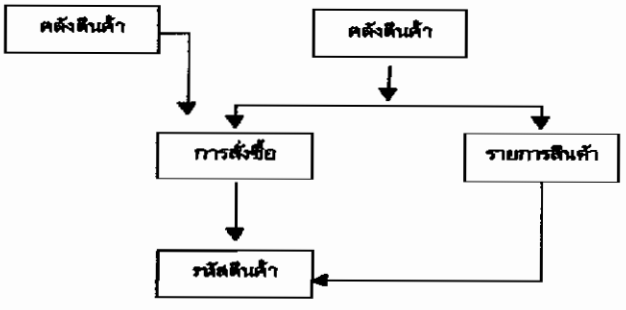
ลักษณะเด่น

- เป็นระบบฐานข้อมูลที่มีระบบโครงสร้างซับซ้อนน้อยที่สุด
- มีค่าใช้จ่ายในการจัดสร้างฐานข้อมูลน้อย
- ลักษณะโครงสร้างเข้าใจง่าย
- เหมาะสำหรับงานที่ต้องการค้นหาข้อมูลแบบมีเงื่อนไขเป็นระดับและออกงานแบบเรียงลำดับต่อเนื่อง
- ป้องกันระบบความลับของข้อมูลได้ดี เนื่องจากต้องอ่านแฟ้มข้อมูลที่เป็นต้นกำเนิดก่อน

ข้อจำกัด

- มีโอกาสเกิดความซ้ำซ้อนมากที่สุดเมื่อเทียบกับระบบฐานข้อมูลแบบโครงสร้างอื่น
- ขาดความสัมพันธ์ระหว่างแฟ้มข้อมูลในรูปเครือข่าย
- มีความคล่องตัวน้อยกว่าโครงสร้างแบบอื่นๆ เพราะการเรียกใช้ข้อมูลต้องผ่านทางต้นกำเนิด (root) เสมอ ถ้าต้องการค้นหาข้อมูลในระดับล่างๆ แล้วจะต้องค้นหาทั้งแฟ้ม

2. ฐานข้อมูลแบบแบบเครือข่าย (hierarchical model) ฐานข้อมูลแบบเครือข่ายจะมีโครงสร้างของข้อมูลแต่ละแฟ้มข้อมูลมีความสัมพันธ์คล้ายร่างแห เปรียบเสมือนมีความสัมพันธ์แบบลูกจ้างกับงานที่ทำ โดยงานชิ้นหนึ่งอาจทำโดยลูกจ้างหลายคน (m ต่อ n) ดังรูปที่ 2.1.5



รูปที่ 2.1.5 แสดงโครงสร้างของฐานข้อมูลแบบเครือข่าย

(บัณฑิต, 2547)

ฐานข้อมูลแบบเครือข่ายมีกฎควบคุมความถูกต้อง คือ สามารถยินยอมให้ระดับชั้นที่อยู่เหนือกว่าจะมีได้หลายแฟ้มข้อมูล ถึงแม้ว่าระดับชั้นถัดลงมาจะมีเพียงแฟ้มข้อมูลเดียว โดยเรคอร์ดที่อยู่เหนือกว่ามีความสัมพันธ์กับเรคอร์ดที่อยู่ระดับล่างได้มากกว่า 1 เรคคอร์ด โดยแต่ละเรคอร์ดสัมพันธ์กันด้วยลิงค์ (links) ฐานข้อมูลแบบเครือข่ายจะทำให้สะดวกในการค้นหามากกว่าฐานข้อมูลแบบลำดับชั้น เพราะไม่ต้องไปเริ่มค้นหาตั้งแต่ข้อมูลต้นกำเนิดโดยทางเดียว

ฐานข้อมูลแบบเครื่อข่ายนี้จะมีกฎควบคุมความถูกต้อง คือ เรคอร์ดพ่อสามารถมีเรคอร์ดลูกได้หลายเรคอร์ด แต่เรคอร์ดลูกแต่ละเรคอร์ดจะมีเรคอร์ดพ่อได้เพียงเรคอร์ดเดียวเท่านั้น (บัณฑิต, 2547)

การจัดการรูปของโครงสร้างฐานข้อมูลแบบลำดับชั้น จะใช้ภาษาปฏิบัติการทีละเรคอร์ดจัดการกับฐานข้อมูล ตัวอย่างของภาษาปฏิบัติการของฐานข้อมูลแบบเครือข่าย ได้แก่ IDMS

ฐานข้อมูลแบบนี้มีลักษณะเด่นและข้อจำกัดของการจัดการฐานข้อมูล ดังนี้

ลักษณะเด่น เหมาะสำหรับงานที่แฟ้มข้อมูลมีความสัมพันธ์แบบเครือข่าย มีโอกาสเกิดความซ้ำซ้อนของข้อมูลน้อยกว่าโครงสร้างแบบลำดับชั้น การค้นหาข้อมูลมีเงื่อนไขได้มากและกว้างกว่าโครงสร้างแบบลำดับชั้น

ข้อจำกัด โครงสร้างแบบเครือข่ายเป็นโครงสร้างที่ง่ายไม่ซับซ้อน จึงทำให้ป้องกันความลับของข้อมูลได้ยาก มีค่าใช้จ่ายและสิ้นเปลืองพื้นที่ในหน่วยความจำเพราะจะเสียพื้นที่ในอุปกรณ์เก็บข้อมูลสำหรับตัวบ่งชี้มาก ถ้าความสัมพันธ์ของระเบียนประเภทต่างๆเกิน 3 ประเภท จะทำให้การออกแบบโครงสร้างแบบเครือข่ายมียุ่งยากซับซ้อนมาก

3. ฐานข้อมูลแบบสัมพันธ์ (relational model) เป็นรูปแบบที่ถูกพัฒนาขึ้นมาภายหลังในปัจจุบันและเป็นฐานข้อมูลที่มีการนิยมใช้กันมากที่สุด โดยมีลักษณะโครงสร้างของฐานข้อมูลแบบสัมพันธ์เป็นการนำเสนอข้อมูลและความสัมพันธ์ระหว่างข้อมูลในรูป relation ซึ่ง relation นี้จะเรียกอีกอย่างหนึ่งว่า ตาราง (Table) โครงสร้างของ Relationประกอบด้วย

แถว (row) ข้อมูล (body) แถวข้อมูล 1 แถว หมายถึง ข้อมูล 1 รายการ ซึ่งแต่ละแถวของ relation เรียกว่า tuple ซึ่ง tuple คือ แถวข้อมูลในตาราง โดยแต่ละแถวของข้อมูลจะประกอบไปด้วยหลาย attribute หรือคอลัมน์ของข้อมูล จำนวนแถวข้อมูลในตารางเราเรียกว่า cardinality และจำนวน attributes ทั้งหมดในตารางเราเรียกว่า degree

สดมภ์ (column) แต่ละสดมภ์ของ relation ได้แก่ คุณลักษณะของข้อมูลในแต่ละแถวซึ่งเราเรียกว่า attribute โดยภายในคอลัมน์จะประกอบด้วย โดเมน (domain) เป็นการกำหนดขอบเขตค่าข้อมูลและชนิดข้อมูลของแต่ละ attribute ที่เป็นไปได้ทั้งหมด

คีย์หลัก (primary key) เป็น attribute หรือกลุ่มของ attribute ที่บ่งบอกว่าข้อมูลจะต้องไม่ซ้ำกันในแต่ละแถวข้อมูลของตาราง

การควบคุมความถูกต้องให้กับข้อมูลในฐานข้อมูลแบบสัมพันธ์ มีดังนี้

- tuple ต้องมีข้อมูลไม่ซ้ำกัน เนื่องจาก relation ในโครงสร้างข้อมูลแบบสัมพันธ์อยู่ในรูปแบบของเซตทางคณิตศาสตร์ โดยภายในเซตจะต้องประกอบไปด้วยสมาชิกที่ไม่ซ้ำกัน ดังนั้น relation ใดๆ ต้องมี attribute หนึ่งที่ทำให้แต่ละ tuple ใน relation มีข้อมูลไม่ซ้ำกัน (บัณฑิต, 2547)

- attribute ไม่มีลำดับจากซ้ายไปขวา เนื่องจาก heading ของ relation ในโครงสร้างข้อมูลแบบสัมพันธ์อยู่ในรูปแบบของเซตทางคณิตศาสตร์ โดยภายในเซตจะต้องประกอบไปด้วยสมาชิกที่ไม่มีลำดับ (บัณฑิต จามรภูมิ, 2547)

- ค่าของ attribute ทุกค่าจะต้องเป็น atomicity เนื่องจาก domain มีเฉพาะค่าที่เป็น atomic เท่านั้น ดังนั้นทุก attribute ในแต่ละตำแหน่งของ tuple ใน relation จะมีค่าเพียงค่าเดียว จะไม่มีรายการของข้อมูล (repeating group หรือ list of value หรือ array) ใน relation นั่นคือ relation ต้องผ่านขั้นตอนการเพิ่มประสิทธิภาพของข้อมูล (normalization) ซึ่งหลังจากทำการเพิ่มประสิทธิภาพของข้อมูลในตารางแล้ว ค่าข้อมูลในแต่ละ tuple ไม่มีรายการข้อมูลอีก (บัณฑิต, 2547)

- กำหนด domain ให้กับข้อมูล จะมีข้อกำหนดว่าค่าที่กำหนดให้กับข้อมูลจะต้องเป็นค่า Scalar นั่นคือจะต้องเป็นค่าข้อมูลที่มีความหมายและเป็นหน่วยเล็กที่สุด ไม่สามารถแบ่งแยกออกไปได้อีก ค่าข้อมูลที่เก็บมีค่าเป็น Scalar นี้เราจะเรียกว่า มีคุณลักษณะของ atomicity ข้อมูลที่สามารถนำมากำหนด domain ได้ จะต้องเป็นข้อมูลที่เป็นอิสระจากข้อมูลอื่น

การจัดการข้อมูลนี้นิยมใช้ภาษาฐานข้อมูล (structured query language, SQL) เป็นภาษาที่ใช้กันอย่างแพร่หลายมีลักษณะคล้ายกับภาษาอังกฤษ ใช้ในการปฏิบัติงานและควบคุมฐานข้อมูล ในภาษาฐานข้อมูลจะมีคำสั่งดังนี้ การสร้างตารางสามารถทำด้วยคำสั่ง create table คำสั่งสอบถามข้อมูลพื้นฐานเป็นการสอบถามข้อมูล หรือ query คำสั่งการป้อนข้อมูลเพิ่มลงสู่ตาราง เราสามารถปรับปรุงเปลี่ยนแปลงแก้ไขข้อมูลที่มีอยู่แล้วในตารางได้ด้วยคำสั่ง update นอกจากการแก้ไขข้อมูลด้วย

คำสั่ง delete เราสามารถสอบถามข้อมูลครั้งหนึ่งจากตารางได้มากกว่าหนึ่งตาราง โดยใช้โครงสร้างของ SQL ที่เรียกว่า joining คำสั่งใน SQL สามารถกำหนดความปลอดภัยในการควบคุมการเข้าถึงข้อมูลในฐานข้อมูลได้ โดยคำสั่ง Ggrant เป็นการกำหนดสิทธิมอบอำนาจให้สามารถเข้าถึงข้อมูล revoke เป็นการเรียกสิทธิอำนาจคืนจากการกำหนดมอบสิทธิด้วยคำสั่ง gran
(บัณฑิต, 2547)

คีย์ คุณสมบัติหนึ่งที่สำคัญของความสัมพันธ์ ก็คือ ความเป็นเอกลักษณ์ (uniqueness property) สิ่งที่ใช้กำหนดความเป็นเอกลักษณ์ของแถวในความสัมพันธ์ เรียกว่า คีย์ (key) ฐานข้อมูลหนึ่งๆ จะมีข้อมูลอยู่มากมาย ยิ่งฐานข้อมูลมีขนาดใหญ่ขึ้นก็จะมีข้อมูลจำนวนมากขึ้นเป็นเงาตามตัวข้อมูลเหล่านี้อาจมีค่าแตกต่างกัน คล้ายกัน หรือแม้กระทั่งเหมือนกัน ทำให้การแยกแยะโดยอาศัยเพียงตัวข้อมูลอย่างเดียวทำได้อย่างยากลำบาก ดังนั้นจึงมีการกำหนดค่า keys ประจำข้อมูล เพื่อทำให้การแยกแยะข้อมูลในฐานข้อมูลเป็นไปอย่างถูกต้อง คีย์มีหลายประเภท ได้แก่ คีย์หลัก, secondary key, foreign key, candidate key, super key ดังรูปที่ 2.1.6

คีย์หลัก (primary key) คีย์หลัก คือ key หลักที่ใช้ในการอ้างถึง entity ในฐานข้อมูล การเลือกคีย์หลักสามารถเลือกได้จากRecord ใดๆก็ได้ ที่ไม่มีโอกาสซ้ำซ้อนกันบนฐานข้อมูลนั้น

เลขประจำตัวประชาชน	ชื่อ	นามสกุล	อายุ
3501552150054	สมชาย	แซ่ตั้ง	25
3210077565107	สมศรี	แซ่อึ้ง	42
4110597520235	สมชาย	แซ่ตั้ง	25
2156800512473	สมปอง	แซ่แต้	16
7812350453784	สมชัย	แซ่เล้ง	50

ตารางที่ 2.1.1 ข้อมูลทั่วไปของประชาชน

(บัณฑิต, 2547)

ตารางที่ 2.1.1 แสดง entity ของประชาชนซึ่งประกอบด้วยเลขประจำตัวประชาชน ชื่อ นามสกุล และอายุ โดยจะสามารถเห็นได้ว่า นอกจาก field เลขประจำตัวประชาชนแล้ว field อื่นๆ คือ

ชื่อ นามสกุล และอายุ อาจซ้ำกันได้ทั้งนั้น ในกรณีนี้ field ที่เหมาะสมที่จะเป็นคีย์หลักที่สุดก็คือ Field เลขประจำตัวประชาชนนั่นเอง

คีย์หลักเป็นข้อมูลสำคัญที่จะทำให้การเข้าถึงข้อมูลบนฐานข้อมูลเป็นไปได้อย่างรวดเร็ว ดังนั้นผู้ใช้จึงควรกำหนดคีย์หลักให้ชัดเจนตั้งแต่ขั้นตอนออกแบบฐานข้อมูล หากไม่มีข้อมูลใดเลยในฐานข้อมูลที่เหมาะที่จะเป็นคีย์หลักก็ควรที่จะกำหนด record ใหม่สำหรับให้เป็นคีย์หลักโดยเฉพาะ

ชื่อ	นามสกุล	อายุ	เพศ
สมชาย	แซ่ตั้ง	25	ชาย
สมศรี	แซ่อึ้ง	42	หญิง
สมชาย	แซ่ตั้ง	25	ชาย
สมปอง	แซ่แต้	16	ชาย
สมชัย	แซ่เล้ง	50	ชาย

ตารางที่ 2.1.2 ข้อมูลทั่วไปของพนักงานบริษัท

(บัณฑิต, 2547)

จากตารางที่ 2.1.2 จะเห็นได้ว่าฐานข้อมูลพนักงานบริษัทนี้ไม่มี field ใดเลยที่เหมาะสมที่จะใช้เป็นคีย์หลัก ดังนั้น ผู้ออกแบบฐานข้อมูลจึงควรเพิ่ม field เฉพาะสำหรับใช้เป็นคีย์หลักของฐานข้อมูล ดังตารางที่ 2.1.3

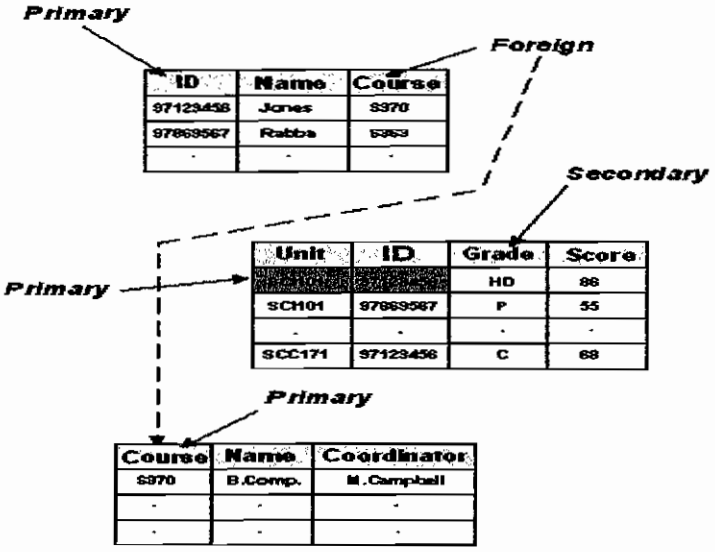
ID	ชื่อ	นามสกุล	อายุ	เพศ
1	สมชาย	แซ่ตั้ง	25	ชาย
2	สมศรี	แซ่อึ้ง	42	หญิง
3	สมชาย	แซ่ตั้ง	25	ชาย
4	สมปอง	แซ่แต้	16	ชาย
5	สมชัย	แซ่เล้ง	50	ชาย

ตารางที่ 2.1.3 ข้อมูลทั่วไปของพนักงานบริษัทหลังจากเพิ่ม คีย์หลัก แล้ว

(บัณฑิต, 2547)

หลังจากเพิ่ม field พิเศษคือ ID เข้าไปในตารางที่ 2.1.3 เพื่อใช้เป็นคีย์หลักโดยเฉพาะ จะสามารถทำให้การอ้างถึงข้อมูลในฐานข้อมูลเป็นไปได้สะดวกและมีประสิทธิภาพมากขึ้น

โครงสร้างของคีย์



รูปที่ 2.1.6 โครงสร้างของคีย์

(บัณฑิต, 2547)

คีย์รอง (secondary key) คีย์สำรอง คือ คีย์เดี่ยวหรือคีย์ผสม (Single or Composite key) ซึ่งเมื่อใช้ในการค้นหาข้อมูลจากความสัมพันธ์จะได้มากกว่าหนึ่งเรคคอร์ด ต่างจากคีย์หลักที่ทำให้ข้อมูลในตารางไม่ซ้ำกัน ดังนั้นคีย์รองจึงไม่จำเป็นจะต้องเป็นเอกลักษณ์

คีย์นอก (foreign key) คีย์นอก คือ คีย์เดี่ยวหรือคีย์ผสม ซึ่งปรากฏเป็นคีย์ทั่วไปของความสัมพันธ์หนึ่ง แต่ไปปรากฏเป็นอีกคีย์หลักในอีกความสัมพันธ์หนึ่ง คีย์นอกเป็นอีกคีย์หนึ่งที่มีความสำคัญมากในฐานข้อมูลเชิงสัมพันธ์ เนื่องจากเป็นตัวที่ใช้สร้างการเชื่อมต่อระหว่างความสัมพันธ์การเปลี่ยนแปลงค่าของคีย์ นอกจะต้องอาศัยความระมัดระวังเป็นอย่างมากเนื่องจากจะมีผลกระทบโดยตรงต่อข้อมูลในความสัมพันธ์อื่นที่มีการอ้างอิงถึงคีย์นอกตัวนี้ จึงมีกฎและเงื่อนไขที่บังคับใช้เพื่อทำให้ข้อมูลมีความถูกต้องอยู่เสมอ

การกำหนดค่าให้กับคีย์นอกของความสัมพันธ์ที่อ้างอิงถึงจะต้องกำหนดค่าของคีย์ให้อยู่ในโดเมนเดียวกันกับความสัมพันธ์ที่คีย์นอกนั้นเป็นคีย์หลัก แต่คีย์นอกนั้นไม่จำเป็นจะต้องเป็นส่วนหนึ่งในคีย์หลักของความสัมพันธ์อื่น

ซุปเปอร์คีย์ (superkey) คือกลุ่มของแอททริบิวที่สามารถนำไปใช้ในการค้นหาข้อมูลที่เป็นเอกลักษณ์ได้

คีย์แข่งขัน (candidate key) คีย์แข่งขันก็คือ ซุปเปอร์คีย์ และไม่มีกลุ่มย่อยของคีย์ใดในคีย์แข่งขันที่จะสามารถเป็นซุปเปอร์คีย์ได้อีก สนับสนุนการสร้างมาตรฐานในการพัฒนาระบบงาน โดยการนำสารสนเทศที่ได้จากพจนานุกรมข้อมูล ตัวอย่างเช่น ชื่อเขตข้อมูลในแต่ละตาราง คุณสมบัติของแต่ละเขตข้อมูลในตาราง ความสัมพันธ์ระหว่างตาราง เป็นต้น ไปใช้ประกอบการแก้ไขโครงสร้างข้อมูลหรือปรับปรุงโปรแกรมให้มีศักยภาพเพิ่มขึ้นและมีมาตรฐานเดียวกัน

2.1.2.4. การนำฐานข้อมูลที่ออกแบบในระดับแนวคิดเข้าสู่ระบบจัดการฐานข้อมูล

ขั้นตอนนี้เป็นการแปลงโครงร่างของฐานข้อมูลที่ได้ทำการออกแบบไว้ ในระดับแนวคิดเข้าสู่รูปแบบของข้อมูลของระบบจัดการฐานข้อมูลที่เลือกใช้ โดยทำการกำหนดภาษาสำหรับนิยามข้อมูลตามระบบจัดการฐานข้อมูลที่เลือกใช้ ให้เป็นไปตามโครงร่างของฐานข้อมูลในระดับแนวคิดที่ออกแบบไว้แล้ว ซึ่งประกอบด้วย 2 ขั้นตอน คือ การกำหนดโครงสร้างของฐานข้อมูล และการกำหนดการอ้างอิงระหว่างตารางข้อมูล
(บัณฑิต, 2547)

1. การกำหนดโครงสร้างของฐานข้อมูล เป็นการกำหนดโครงสร้างของข้อมูลในลักษณะของตารางสองมิติ (two dimension) ซึ่งประกอบด้วย สดมภ์ (column) ซึ่งใช้แทน แอททริบิวต์ และแถว ซึ่งใช้แทนความสัมพันธ์ระหว่างแอททริบิวต์ โดยเรียกตารางสองมิตินี้ว่า ตารางข้อมูล ทั้งนี้ตารางข้อมูลจะประกอบด้วยคุณสมบัติต่างๆ ได้แก่ ชื่อตารางข้อมูล ชื่อแอททริบิวต์ การกำหนดคุณสมบัติของข้อมูล ได้แก่ ค่าที่เป็นไปได้ ประเภทและขนาดของข้อมูลที่จัดเก็บ ฯลฯ

2. การกำหนดการอ้างอิงระหว่างตารางข้อมูล เป็นการใช้คำสั่งในระบบจัดการฐานข้อมูลที่เลือกทำการกำหนดคีย์ระหว่างตารางข้อมูลที่เกี่ยวข้องกัน โดยต้องพิจารณาถึงคุณสมบัติของคีย์ ค่าของคีย์ ข้อจำกัดและกฎเกณฑ์ของแอททริบิวต์ต่างๆ เช่น คีย์หลัก และคีย์นอก โดเมนของแอททริบิวต์ ตลอดจนข้อจำกัดเฉพาะของกฎเกณฑ์ในการปฏิบัติงาน

2.1.2.5. การนำฐานข้อมูลไปใช้และการประเมินผล

หลังจากการออกแบบฐานข้อมูลในระดับกายภาพเสร็จสิ้นลง ซอฟต์แวร์ระบบจัดการฐานข้อมูลจะนำภาษาสำหรับนิยามข้อมูลที่ครบถ้วนสมบูรณ์ไปสร้างเป็นฐานข้อมูลและตารางข้อมูล เพื่อบรรจุข้อมูลลงในระบบฐานข้อมูลสำหรับการใช้งานจริง

เมื่อระบบจัดการฐานข้อมูลทำการสร้างรายละเอียดต่างๆ ของฐานข้อมูลที่ถูกออกแบบ เช่น ตารางข้อมูลทั้งหมดในฐานข้อมูล แอททริบิวต์ในแต่ละตารางข้อมูล ระดับสิทธิของผู้ใช้ข้อมูลแต่ละคน เป็นต้น เรียบร้อยแล้ว ก็จะทำการจัดเก็บรายละเอียดต่างๆ เหล่านี้ไว้ในพจนานุกรมข้อมูล (data dictionary) ซึ่งผู้บริหารฐานข้อมูลสามารถเรียกดูได้ ทั้งนี้ การทำงานในขั้นตอนนี้จะครอบคลุมถึงการทดสอบประสิทธิภาพในการทำงานของฐานข้อมูล (performance) และการทดสอบการทำงานของฐานข้อมูลที่จัดทำขึ้นก่อนนำไปใช้งานจริงด้วย

จากนั้นจึงเป็นการนำฐานข้อมูลมาใช้งานจริง ซึ่งได้แก่ การเรียกใช้ข้อมูลจากระบบฐานข้อมูลผ่านภาษาสอบภาม (query language) และการจัดทำรายงานต่างๆ ซึ่งผลจากการทำงานที่เกิดขึ้นจะมีการประเมินและตรวจสอบเพื่อปรับปรุงแก้ไข และบำรุงรักษาฐานข้อมูลในเรื่องต่างๆ ได้แก่ การสำรองข้อมูล การกู้ข้อมูลหากระบบฐานข้อมูลมีปัญหา การปรับปรุงประสิทธิภาพการทำงานของฐานข้อมูลให้รวดเร็วขึ้น เป็นต้น (บัณฑิต, 2547)

2.2 การศึกษาเวลาโดยการจับเวลาโดยตรง

การจับเวลาโดยตรงเป็นส่วนหนึ่งของการศึกษาการเคลื่อนไหวและเวลา (motiom and time study) ซึ่งวิธีการจับเวลาโดยตรงนี้เป็นที่นิยมในปัจจุบัน โดยผู้จับเวลาจะเข้าไปจับเวลาบริเวณที่คนงานทำงาน ทำให้สามารถมองเห็นลักษณะการทำงานอย่างละเอียดและเวลาที่ได้เป็นเวลาทำงานจริง แต่มีข้อเสียตรงที่ว่า การจับเวลาโดยตรงนี้มักมีผลโดยตรงต่อคนงานทางด้านจิตใจ ทำให้เวลาที่ได้เร็วหรือช้าไปเสมอจากการทำงานในลักษณะไม่ปกติของเขาเอง เขาอาจทำงานเร่งขึ้นหรืออาจทำงานช้าลง ดังนั้นต้องมีขั้นตอนในการจับเวลาวิธีนี้ ดังนี้
(วัชรินทร์, 2547)

2.2.1. การทำความเข้าใจเกี่ยวกับคนงานและหัวหน้าคนงาน

จากปัญหาดังกล่าวก่อนหน้านี้ จึงต้องมีการอธิบายให้คนงานทราบและเข้าใจถึงวัตถุประสงค์ของการศึกษาก่อน ว่าเราต้องการศึกษาดูเวลาเฉลี่ยของการทำงาน ไม่ใช่จับเวลาของการทำงานของเรา หัวหน้าคนงานจะช่วยได้ในการอธิบายให้คนงานเข้าใจและดูว่าที่ทำอยู่ถูกต้องตามวิธีและความรวดเร็วที่ต้องการ

ก่อนทำการศึกษาเวลาต้องมั่นใจว่างานนั้นพร้อมที่จะถูกศึกษา นั่นคือ

1. วิธีใช้อยู่เป็นวิธีที่ดีที่สุด
2. การวางเครื่องมือเครื่องจักรอยู่ในลักษณะที่เหมาะสม

3 วัตถุที่ต้องการเป็นไปตามคุณลักษณะที่ต้องการ

4 สภาพการทำงานดีและไม่มีปัญหาของความปลอดภัย

5. คุณภาพของชิ้นงานที่ผลิตได้เป็นไปตามความต้องการ

6. ความเร็วของเครื่องจักรเป็นไปตามที่ตั้งไว้

7. คนงานมีความชำนาญหรือประสบการณ์พอสมควร

2.2.2. การบันทึกข้อมูล

ก่อนที่จะทำการจับเวลาเก็บข้อมูล ควรที่จะจัดทำรูปแบบตารางในการเก็บข้อมูล และควรที่จะมีการบันทึกรายละเอียดต่างของสถานที่ทำงาน เครื่องมือ แต่ไม่ใช่เครื่องจักร ซึ่งสามารถแบ่งออกเป็นกลุ่มได้ ดังนี้

1. ข้อมูลเกี่ยวข้องกับการอ้างอิงในวันหลัง (หาได้ง่ายเมื่อต้องการใช้อีก) ได้แก่ เลขที่, แผ่นที่และจำนวนแผ่น, ชื่อหรือชื่อย่อของผู้ศึกษา, วันที่ศึกษา, ชื่อผู้ตรวจสอบ

2. รายละเอียดผลิตภัณฑ์ ได้แก่ ชื่อผลิตภัณฑ์, วัสดุ ,คุณภาพที่ต้องการ

3. วิธีการผลิต วิธีการทำ เครื่องมือที่ใช้ ได้แก่ แผนกหรือตำแหน่งของการทำงานนั้น, คำอธิบายว่าทำงานอย่างไร, วิธีทำงานมาตรฐาน, เครื่องจักร (ผู้สร้าง แบบ ขนาดและความจุ), เครื่องมือเครื่องจับ, เครื่องวัดที่ใช้วัดคร่าวๆ, สถานที่ทำงาน, อัตราการทำงานของเครื่องจักร, การป้อนงานและอื่นๆ ที่มีผลต่ออัตราการผลิต

4. ผู้ปฏิบัติงาน ได้แก่ ชื่อผู้ปฏิบัติงาน

5. ระยะเวลาของการศึกษา ได้แก่ เวลาเริ่ม, เวลาสิ้นสุด, เวลาทั้งหมด

6. สภาพการทำงาน ได้แก่ อุณหภูมิ, ความชื้น, แสงสว่าง

2.3. การกำหนดงานและขั้นตอนของงาน (scheduling and sequencing)

การจัดตั้งวิธีการกำหนดงาน (Scheduling) เกิดขึ้นเนื่องจากความต้องการที่จะกำหนดงานหนึ่งงานใดให้กับหน่วยงานต่างๆภายในระยะเวลาที่ต้องเริ่มต้นและสิ้นสุดตามที่กำหนดไว้เพื่อความเข้าใจถึงความสำคัญของขบวนการกำหนดงาน จึงจำเป็นที่จะกล่าวถึงปัญหาที่จำเป็นต้องใช้การกำหนดงานไปช่วยแก้ไข ซึ่งมักจะเป็นปัญหาของการผลิตในงานอุตสาหกรรมขนาดใหญ่ ปัญหาที่จะเกิดขึ้นประกอบด้วย

1. การสั่งผลิตหรือสั่งซื้อของให้กับแผนกผลิตของโรงงานเป็นวาระๆ

2. การกำหนดชนิดของงานให้กับหน่วยงานต่างๆ

3. การสำรวจขอบข่ายต่างๆที่เกี่ยวข้องกับการผลิต

4. ในขณะที่งานอยู่ในระหว่างการดำเนินงาน ก็จะมีการแก่งแย่งกับงานอื่นๆ ซึ่งใช้ทรัพยากรอย่างเดียวกัน เช่น ต้องใช้เครื่องจักรเครื่องเดียวกัน ซึ่งทำให้เกิดการขัดแย้งขึ้นในหน่วยงาน

5. ความขัดข้องของเครื่องจักร การหยุดงานของคนงาน ความสามารถในการทำงานที่ต่ำกว่ามาตรฐานของคนงาน เครื่องมือเสียหรือสึกหรอ วัสดุที่ใช้ในการผลิตมีจุดบกพร่อง เครื่องจักรต้องว่างเนื่องจากจะต้องรองานที่จะออกจากหน่วยงานอื่น

6. คำสั่งผลิตถูกระงับ ลดขนาด หรือเพิ่มขนาดการผลิต

7. ไม่มีวัตถุดิบเข้ามาในโรงงานตามที่คาดหมายไว้

8. การขายสินค้าตกลงหรือเพิ่มขึ้นอย่างกะทันหัน

9. มีการปรับปรุงเปลี่ยนแปลงสินค้าซึ่งจะทำให้มีการเปลี่ยนแปลงของเวลามาตรฐานในการผลิต เวลาจัดตั้งขบวนการผลิตใหม่ ลำดับขั้นในการผลิต การเสนอแนะวิธีการผลิตต่อผู้คุมหน่วยงานผลิตต่างๆ ฯลฯ (รศ.พิภพ, 2546)

จากปัญหาข้างต้นนั้นจะเห็นได้ว่าเป็นงานที่ยากมาก ที่จะควบคุมให้มีการทำงานอย่างมีประสิทธิภาพภายใต้สภาวะแวดล้อมที่มีความเปลี่ยนแปลงได้ตลอดเวลา จึงมีการนำเอาวิธีการกำหนดงานเข้าไปช่วยแก้ปัญหา ถึงแม้ว่าวิธีการกำหนดงานจะไม่อาจช่วยแก้ไขปัญหาทั้งหมดได้ในปัจจุบัน เนื่องจากเทคนิคต่างๆทางคณิตศาสตร์ และการบริหารยังมีประสิทธิภาพไม่เพียงพอ แต่ก็อาจสามารถช่วยให้การแก้ไขปัญหามีประสิทธิภาพดีขึ้น

การกำหนดงานมีความหมายทั่วไปคือ การจัดเตรียมตารางเวลา (time table) ของขั้นตอนของงาน (activities) ที่เกี่ยวข้องในการทำงานอย่างหนึ่งอย่างใดให้สำเร็จลุล่วงไป งานที่กล่าวถึงในที่นี้มีความหมายถึงงานทุกชนิดที่ต้องการวางแผนหรือกำหนดขั้นตอนในงาน (sequencing) นั้นๆ ให้ดำเนินไปอย่างมีประสิทธิภาพ

สำหรับประสิทธิภาพในแต่ละงานมีความหมายแตกต่างกันออกไปแล้วแต่จุดประสงค์หรือนโยบายการดำเนินงาน กล่าวโดยทั่วไปประสิทธิภาพของแต่ละงานที่อ้างถึงนี้ก็คือ การกำหนดงานของแต่ละงานให้บรรลุเป้าหมายใดเป้าหมายหนึ่งที่ตั้งไว้ เช่น

1. ใช้ค่าใช้จ่ายในการดำเนินงานน้อยที่สุด
2. ทำงานเสร็จเร็วที่สุด
3. ใช้เงินทุนสำหรับวัสดุคงคลังน้อยที่สุด
4. ใช้เนื้อที่ในการเก็บวัสดุน้อยที่สุด

5. ใช้เวลารอคอยระหว่างขบวนการผลิตน้อยที่สุด
6. ใช้เงินทุนในการหมุนเวียนน้อยที่สุด
7. ใช้แรงงานน้อยที่สุด
8. ใช้เครื่องมือหรือสิ่งต่างๆ ที่มีอยู่แล้วให้เป็นประโยชน์ที่สุด
9. ทำให้ลูกค้าพอใจมากที่สุด
10. ทำให้ผู้ดำเนินงานมีความสบายใจที่จะทำงานมากที่สุด

(รศ.พิภพ, 2546)

จากความหมายโดยทั่วไปของการกำหนดงานจะเห็นได้ว่าไม่ได้หมายความเฉพาะถึงวิธีการนำไปใช้ในงานอุตสาหกรรมเพียงอย่างเดียว แต่สามารถประยุกต์ใช้กับงานในแขนงอื่นๆ ได้ด้วย เนื่องจากการกำหนดงานได้ถูกนำไปใช้ในงานอุตสาหกรรม โดยเฉพาะงานด้านการวางแผนการผลิตอย่างแพร่หลาย ดังนั้นการกำหนดงานกับการกำหนดขั้นตอนของงานมักถูกเรียกและใช้ควบคู่กันไป

2.4. การพัฒนา Web Page ด้วยภาษา ASP

ASP (Active Server Page) เป็นภาษาสคริปต์ชนิดหนึ่งเพื่อใช้สร้างเอกสารบนเว็บถูกออกแบบมาให้สามารถประมวลผลได้ทั้งฝั่งไคลเอ็นต์และฝั่งเซิร์ฟเวอร์ โดยที่ทั้งไคลเอ็นต์และเซิร์ฟ เวอร์ต้องสนับสนุนภาษา ASP ด้วย จุดประสงค์ที่ใช้ ASP สร้างเว็บเพจก็เพื่อขยายขีดความสามารถของเว็บเพจให้สามารถทำงานในลักษณะโต้ตอบกับผู้ใช้ (Interactive) ในลักษณะ Real-Time รวมถึง ใช้ ASP ติดต่อกับฐานข้อมูลผ่านทางบราวเซอร์ได้ เช่นเดียวกับภาษาสคริปท์ CGI ASP ประกอบด้วย 2 ส่วนใหญ่ๆ ได้แก่ ส่วนแรกเป็น Tag <%.....%> เมื่อเซิร์ฟเวอร์พบ Tag ดังกล่าวจะมีขั้นตอนในการตีความชุดคำสั่งของ ASP นั้นๆก่อน แล้วจึงส่งผลลัพธ์กลับมาในรูปแบบของ Tag HTML โดยเราสามารถใช้โครงสร้างของภาษา JScript หรือ VBScript ในการเขียน ASP ก็ได้ และส่วนที่สองเป็น Text คือข้อความต่างๆ ที่ต้องการแสดงบนเว็บ (สัจจะ และ สมพร, 2543)

ลักษณะของ ASP ที่สำคัญคือ ต้องทำงานทางฝั่งเซิร์ฟเวอร์ (Server Side) หรือฝั่งไคลเอ็นต์ (Client Side) ก็ได้ เว็บเพจที่มาจากการใช้สคริปท์ ASP จะเป็นไปในลักษณะผู้ใช้ร้องขอ (Request) เอกสารไปยังฝั่งเซิร์ฟเวอร์ แล้วคำร้องขอดังกล่าวจะต้องถูกตีความที่ฝั่งเซิร์ฟเวอร์ก่อน ด้วยเว็บเซิร์ฟเวอร์ ผลจากการตีความ เว็บเซิร์ฟเวอร์จะส่งเอกสารในรูปแบบ HTML กลับไปยังฝั่งไคลเอ็นต์ที่ร้องขอเอกสารดังกล่าวเข้ามา กระบวนการนี้จะกระผ่านโปรโตคอล HTTP ถ้าใช้งาน ASP ในลักษณะ Client Side จะทำให้ผู้ใช้ที่ใช้บราวเซอร์ของ Netscape ไม่สามารถแสดงเอกสารที่มาจาก ASP ได้

ก่อนที่จะแสดงเป็นหน้าโฮมเพจ การใช้งาน ASP เป็นการใช้ชุดคำสั่ง ASP หรือ VBScript เพื่อสั่งให้ Tag HTML ทำงานอีกทอดหนึ่ง (สัจจะ และ สมพร, 2543)

2.4.1. วิธีการเตรียมโฟลเดอร์ก่อนใช้งาน ASP

การกำหนดสิทธิ คือ การอนุญาตให้โฟลเดอร์นั้นๆ สามารถรันสคริปท์ ASP ได้โดยต้องกำหนดสิทธิทุกครั้งให้กับทุกโฟลเดอร์ที่มีการจัดเก็บเอกสาร ASP ถ้าไม่กำหนดสิทธิให้กับโฟลเดอร์นั้นๆ PWS หรือ IIS จะไม่สามารถสั่งให้ไฟล์ .ASP ที่เก็บอยู่ในโฟลเดอร์ดังกล่าวรันได้

การกำหนด Virturl Directory คือ การตั้งชื่อ Path สมมติขึ้นมา ชื่อ Alias เมื่อมีการแสดงเอกสาร .ASP ขึ้นในบราวเซอร์แล้ว บราวเซอร์จะแสดง Path ที่ตั้งชื่อสมมติขึ้นมาในช่องกรอก URL ซึ่งควรกำหนดชื่อ Path สมมติให้มีชื่อเหมือนกับ Path จริงที่เก็บไฟล์ .ASP โดยวิธีการ คือสร้างโฟลเดอร์ใหม่ และกำหนดให้ Share this folder คลิก Advanced ในหน้าต่าง PWS ใช้งานในแบบ Server Side เว็บเซิร์ฟเวอร์จะต้องสนับสนุน ASP ด้วย ซึ่งเว็บเซิร์ฟเวอร์จากไมโครซอฟท์ เช่น PWS, IIS จะสนับสนุน ASP อยู่แล้ว (ธวัชชัย และ ธาริน, 2546)

2.4.2. ข้อแตกต่างระหว่างโฮมเพจแบบ Static Homepage กับ Dynamic Homepage

การสร้างเว็บเพจด้วย HTML เป็นการเขียน Tag ควบคู่กับข้อความ Text เพื่อให้บราวเซอร์แปลความหมายตามหน้าที่ของแต่ละ Tag แล้วแสดงออกมาเป็นเว็บเพจในรูปแบบต่างๆ ไฟล์ HTML เป็นเพียงเอกสารที่ผู้จัดทำโฮมเพจเก็บไว้ที่โฮสต์ ถ้าไคลเอ็นต์ต้องการไฟล์ใดก็คลิกเลือกไฟล์นั้น ไม่สามารถโต้ตอบกับไคลเอ็นต์ได้ด้วยข้อจำกัดของ HTML ทำให้เกิดภาษาที่เรียกว่า สคริปท์ ที่ใช้สำหรับสร้างโฮมเพจเช่นเดียวกับ HTML แต่ได้เพิ่มความสามารถต่างๆ ให้สามารถสร้างเว็บเพจได้ดีขึ้น เช่น DHTML, JavaScript, VBScript ภาษาสคริปท์เหล่านี้เรียกว่า Client-Side Script แต่สำหรับภาษา ASP, Java, Java Server Pages (JSP), Common Gateway Interface (CGI), Professional Home Page (PHP) เป็นภาษาสคริปท์ที่เรียกว่า Server-Side Script เพื่อขยายขีดความสามารถในการจัดทำโฮมเพจโดยเฉพาะการทำโฮมเพจที่ต้องการใช้งานร่วมกับฐานข้อมูล (ธวัชชัย และ ธาริน, 2546)

2.4.3. ขั้นตอนการทำงานของโฮมเพจที่สร้างด้วย ASP

เมื่อไฟล์ Default.asp ซึ่งถูกเก็บบนเว็บเซิร์ฟเวอร์ถูกเรียกใช้งาน ซอฟท์แวร์ที่ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์จะทำหน้าที่ตีความหมายของชุดคำสั่งที่อยู่ในไฟล์ Default.asp ก่อน ด้วย ASP Script Engine (asp.dll) ผลจากการตีความ เว็บเซิร์ฟเวอร์จะตอบสนองกลับไปยังฝั่งไคลเอ็นต์ในรูปแบบของเอกสาร HTML เพื่อให้บราวเซอร์แปลความหมายของ Tag HTML คลิกปุ่ม Edit Properties คลิกปุ่ม

Browse ที่ช่อง Directory คลิกเลือกโฟลเดอร์ที่ต้องการ (ที่กำหนด Share ไว้) กำหนดชื่อ Alias และคลิกเช็คบ็อกซ์ Execute แล้วคลิกไอคอน Main จะพบว่า Path ในการอ้างอิงจะเปลี่ยนไปตามที่กำหนด (ธวัชชัย และ ธาริน, 2546)

2.4.4. การทำงานของสคริปท์

1. ทำการขอร้องไฟล์ .ASP (Request) ไปยัง PWS หรือ IIS

2. PWS หรือ IIS ได้รับคำร้องขอไฟล์ .ASP จากผู้ใช้ จะไปค้นหาไฟล์จาก Path ที่กำหนด ถ้าไม่มีจะแสดงข้อความ error ถ้ามีจะนำไฟล์ดังกล่าวไปให้ ASP.DLL ซึ่งทำหน้าที่เป็น ASP Script Engine แปลชุดคำสั่ง

3. ผลที่ได้จากการแปลชุดคำสั่งหรือตีความจะกลับมาในรูปของ Tag HTML ที่บราวเซอร์รู้จัก แล้ว PWS หรือ IIS จึงส่งเอกสาร (Response) ในรูปแบบ HTML กลับไปยังบราวเซอร์ของผู้ใช้ที่ร้องขอเข้ามา ทำให้ผู้ใช้ที่เข้าโฮมเพจไม่มีทางที่จะรู้ชุดคำสั่งของโฮมเพจ จะเห็นแต่เพียงชุดคำสั่ง HTML เพราะกระบวนการในการตีความจะอยู่ทางฝั่งเซิร์ฟเวอร์ทั้งหมด (Run at Server Side)

2.4.5. การระบุชนิดของภาษาว่าเป็น VBScript

สามารถระบุได้ 3 รูปแบบ

แบบที่ 1

<SCRIPT LANGUAGE = "VBSCRIPT">

คำสั่งต่างๆของ VBSCRIPT

</SCRIPT>

แบบที่ 2

<SCRIPT TYPE = "TEXT/VBSCRIPT">

คำสั่งต่างๆของ VBSCRIPT

</SCRIPT>

แบบที่ 3

<SCRIPT LANGUAGE = "VBS">

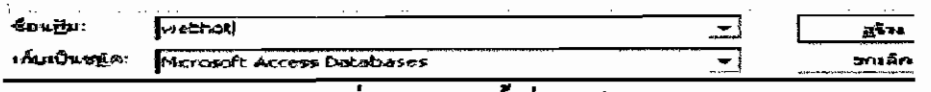
(ธวัชชัย และ ธาริน, 2546)

2.5. โปรแกรม Access

การใช้งานฐานข้อมูลมีให้เลือกใช้หลายหลาย 1 ในนั้นคือ Access นิยมใช้ร่วมกับ ASP เป็นที่นิยมอย่างมาก และภาษาในการทำเว็บอื่นๆ Access ไม่ได้เป็นแต่ฐานข้อมูล สามารถทำได้หลายอย่างมาก แต่จะนำเสนอในส่วนในการสร้างฐานข้อมูล

2.5.1. การสร้างฐานข้อมูลใหม่

ให้ไปที่ New หรือกดที่รูปกระดาษขาว แล้วดับเบิลคลิกที่ฐานข้อมูลเปล่า แล้วตั้งชื่อตัวอย่าง webhot นามสกุลของไฟล์จะเป็น .mdb แล้วคลิกสร้าง ดังรูปที่ 2.5.1



รูปที่ 2.5.1 การตั้งชื่อไฟล์

(กิตติ และ กุลชน, 2547)

จะได้ไฟล์ ดังรูปที่ 2.5.2

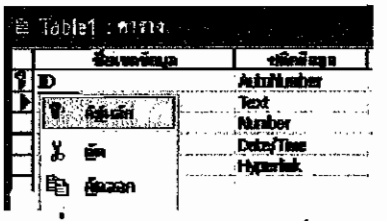
รูปที่ 2.5.2 ไฟล์ที่ได้จากการบันทึก

(กิตติ และ กุลชน, 2547)

2.5.2. การสร้างตารางในฐานข้อมูล

ให้คลิกที่สร้างตารางในมุมมองออกแบบ จากนั้นจะมีให้เราใส่ข้อมูล 3 ช่อง คือ

1. ชื่อเขตข้อมูล เป็นการกรองชื่อเขตข้อมูล
2. ชนิดข้อมูล คลิกที่ ปุ่มลูกศร เลือกแบบฐานชนิด
3. คำอธิบาย จะใส่หรือไมก็ได้

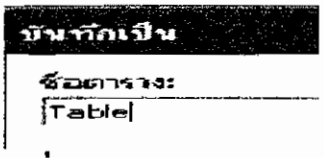


รูปที่ 2.5.3 การกำหนดคีย์หลัก

(กิตติ และ กุลชน, 2547)

เมื่อได้แล้ว ต้องกำหนดคีย์หลัก โดยการคลิกขวา ดังรูปที่ 2.5.3 การกำหนดคีย์หลักควรเลือกชื่อเขตข้อมูลที่ไม่มีโอกาสชื่อซ้ำกัน

เซฟทับอีกครั้ง แล้วตั้งชื่อ ดังตัวอย่าง ตั้งชื่อว่า table ดังรูปที่ 2.5.4

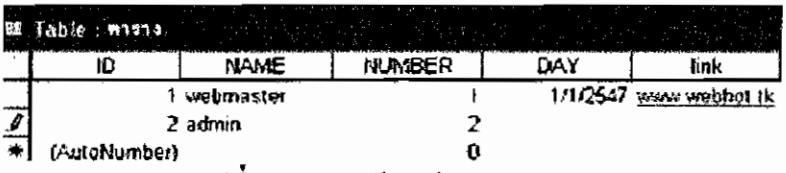


รูปที่ 2.5.4 การบันทึกตาราง

(กิตติ และ กุลชน, 2547)

2.5.3. การป้อนข้อมูลลงตาราง

ให้ดับเบิลคลิกที่ตารางที่ตั้งชื่อไว้ คลิกฟิลด์ที่ต้องการ แล้วป้อนข้อมูลที่ต้องการ ดังรูปที่ 2.5.5



รูปที่ 2.5.5 การป้อนข้อมูลลงตาราง

(กิตติ และ กุลชน, 2547)

2.5.4. การลบฟิลด์ ให้คลิกขวาที่ฟิลด์ แล้วลบระเบียน

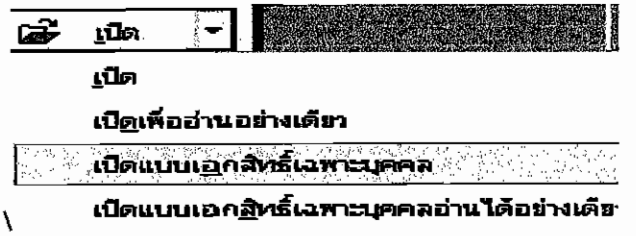
2.5.5. การเปลี่ยนชื่อและเพิ่มฟิลด์

การเปลี่ยนชื่อ ให้คลิกขวาที่ฟิลด์แล้วเปลี่ยนชื่อการแทรกคอลัมน์ ให้คลิกขวาที่ฟิลด์แล้วแทรกคอลัมน์ การลบคอลัมน์ ให้คลิกขวาที่ฟิลด์แล้วลบคอลัมน์ (การแทรกคอลัมน์แบบนี้อาจจะได้ชนิดข้อมูลไม่ถูกต้อง ควรจะไปเพิ่มที่มุมมองนักออกแบบ)

2.5.6. การใส่รหัสผ่าน

ให้ปิดก่อนแล้วเปิดใหม่ ให้เปิดแบบเอกสิทธิ์เฉพาะบุคคล ดังรูปที่ 2.5.6

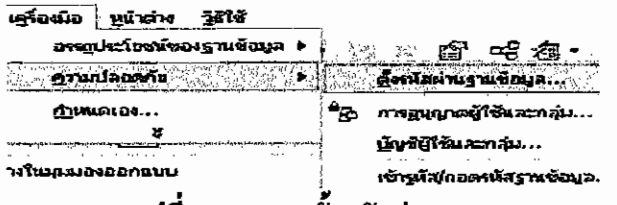
4940009
13 ส.ค. 2549
สำนักหอสมุด



รูปที่ 2.5.6 การเปิดเอกสารเฉพาะบุคคล

(กิตติ และ กุลชน, 2547)

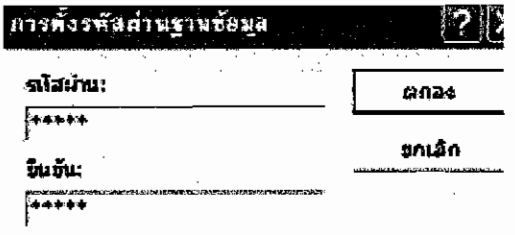
แล้วให้ไปที่เครื่องมือ > ความปลอดภัย > ตั้งรหัสผ่าน ดังรูปที่ 2.5.7



รูปที่ 2.5.7 การตั้งรหัสผ่าน(1)

(กิตติ และ กุลชน, 2547)

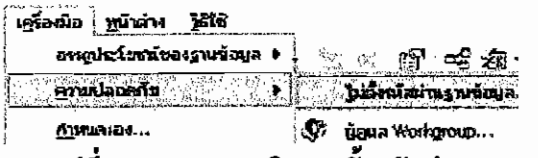
แล้วตั้งรหัสผ่านให้เหมือนกัน 2 ช่อง ดังรูปที่ 2.5.8



รูปที่ 2.5.8 การตั้งรหัสผ่าน(2)

(กิตติ และ กุลชน, 2547)

2.5.7. การยกเลิกการตั้งรหัสผ่าน ให้ปิดไฟล์ที่เปิดอยู่ก่อนแล้ว ให้เปิดแบบเอกสิทธ์เฉพาะบุคคล แล้วไปที่เครื่องมือ > ความปลอดภัย >ไม่ตั้งรหัสผ่าน แล้วใส่รหัสเดิมที่จะยกเลิก ดังรูปที่ 2.5.9



รูปที่ 2.5.9 การยกเลิกการตั้งรหัสผ่าน

(กิตติ และ กุลชน, 2547)

2.5.8. คำสั่งที่ใช้ในการควบคุมการทำงาน

จากการที่เราได้สร้างโปรแกรมกันมาบ้างแล้ว จะเห็นว่าโปรแกรมทำงานจากบนลงล่าง และจากซ้ายไปขวาตามลำดับ แต่ในบางครั้ง เราต้องการให้โปรแกรมสามารถเลือกทำงานตามเงื่อนไขของตัวแปรได้ด้วย ซึ่งใน VBScript ได้เตรียมคำสั่งที่ใช้ตรวจสอบการทำงานตามเงื่อนไขที่ต้องการไว้แล้ว คำสั่งในการควบคุมสามารถแบ่งได้ 2 ประเภทหลัก ดังนี้

2.5.8.1. คำสั่งที่ใช้ในการตัดสินใจ (Decision Structures) ในกรณีที่เราจำเป็นต้องตัดสินใจเลือกสิ่งใดสิ่งหนึ่ง เราจะใช้คำสั่งเงื่อนไข (Condition) ในการตัดสินใจ โดยคำสั่งประเภทนี้จะแบ่งออกได้เป็น 2 คำสั่ง คือ คำสั่ง If-then-Else และคำสั่ง Select-Case

1. คำสั่ง If-Theu-Else เป็นคำสั่งที่มีรูปแบบดังนี้

```
If<นิพจน์ที่เป็นBoolean>then
'คำสั่งสำหรับนิพจน์ที่มีค่าเป็นTrue
else
'คำสั่งสำหรับนิพจน์ที่มีค่าเป็นFalse
endIf
```

(กิตติ และ กุลชน, 2547)

ตัวอย่างที่ 1

```
if price>20 then
response.write("Too expensive")
else
response.write("Price is OK")
end If
```

2. คำสั่ง Select-Case เป็นคำสั่งใช้ในการเลือกเส้นทางการทำงานของโปรแกรมจากค่าของนิพจน์ที่กำหนดเป็นคำสั่งที่มีรูปแบบต่อไปนี้

```
Select Case <นิพจน์>
    Case<ค่าของนิพจน์กลุ่มแรก>
'คำสั่งที่ทำงานเมื่อค่านิพจน์ตรงกับค่าในกลุ่มแรก
    Case<ค่าของนิพจน์กลุ่มที่ 2>

'คำสั่งที่ทำงานเมื่อค่านิพจน์ตรงกับค่าในกลุ่ม 2
    Case Else
'คำสั่งเมื่อไม่ตรงกับกรณีไหนเลย
    End Select
```

ตัวอย่างที่ 2

```
Name=-1
Select Case Number
Case 1,2,3
Response.Write("Between 1 And 3")
Case 4,5,6,7,8
Response.Write("Between 4 And 8")
Case Number>8
Response.Write("Greather than 8")
Case Else
```

```
Response.Write("Not positive number")
End Select
```

2.5.8.2. คำสั่งในการทำซ้ำ (Loop Structures) เราสามารถใช้คำสั่งที่จะกล่าวถึงในหัวข้อนี้ ในการทำงานซ้ำเป็นจำนวนครั้งที่แน่นอน หรือจนกว่าจะทำให้เงื่อนไขเป็นดังที่เราต้องการได้ คำสั่งที่จะกล่าวถึงมี 2 คำสั่ง คือ (กิตติ และ กุลชน, 2547)

1. คำสั่ง For-Next คำสั่งนี้จะใช้ในการทำงานซ้ำเป็นจำนวนครั้งที่แน่นอน โดยมีรูปแบบดังนี้

For<ชื่อตัวแปร>=<ค่าเริ่มต้น>To<ค่าสุดท้าย> [step]<ค่าที่เพิ่มขึ้นของตัวแปร>1

ตัวอย่างที่ 3

```
Dim I
For I=0 to 10 step 2
Response.Write(I)
Next
```

2. คำสั่ง Do-Loop เป็นคำสั่งที่ใช้ในการทำซ้ำ โดยจำนวนครั้งในการซ้ำขึ้นอยู่กับเงื่อนไขหลังคำว่า while หรือ until รูปแบบของคำสั่งจะเป็นดังนี้

```
Do While<condition>
'ชุดคำสั่ง
Loop
Do
'ชุดคำสั่ง
Loop While<condition>
Do Until<condition>
'ชุดคำสั่ง
Loop
```

ตัวอย่างที่ 4

```
Dim I
Do While I<10
Response.write(I)
I=I+1
Loop
Dim I
Do
Response.write(I)
I=I+1
Loop While I<10
```

(กิตติ และ กุลชน, 2547)

2.5.9. สรุปโปรแกรม Access

1. Field Name ใส่ชื่อฟิลด์ Data Type กำหนดชนิดของข้อมูล โดยเมื่อคลิกในช่องนี้ จะปรากฏปุ่มลูกศร (Drop Down Menu) ให้คลิกที่ปุ่มลูกศรแล้วคลิกเลือก Data Type ที่ต้องการ หรือจะใช้วิธีพิมพ์ตัวอักษรตัวแรกของชนิดข้อมูลที่ต้องการ

2. Text ข้อมูลตัวอักษร ตัวเลข หรืออื่นๆผสมกัน ที่ยาวไม่เกิน 255 ตัวอักษร เช่น คำนำหน้า, ชื่อ, นามสกุล, ที่อยู่เป็นต้น

3. Memo ข้อมูลตัวอักษร ตัวเลข หรืออื่นๆผสมกัน ยาวเกิน 255 ตัวอักษร เช่น บันทึก หรือหมายเหตุ

4. Number ข้อมูลตัวเลขที่นำไปคำนวณได้ เช่น อายุ, เงินเดือน

5. Date/Time ข้อมูลวันที่ สามารถนำไปคำนวณได้ เช่น วันเกิด, วันเข้าทำงาน

6. Currency ข้อมูลตัวเลขรูปแบบสกุลเงิน

7. AutoNumber ข้อมูลตัวเลขที่ให้รันอัตโนมัติ เช่น ลำดับที่

8. Yes/No ข้อมูลตรรกะ ให้เลือก "ใช่" หรือ "ไม่ใช่"

9. OLE Object ข้อมูลเสียง, ภาพ, วิดีโอ

10. Hyperlink ข้อมูลที่สามารถคลิกลิงก์ได้

11. Lookup Wizard ข้อมูลค้นหา และแสดงเป็นรายการ เช่น Drop Down List

12. Description เป็นรายการอธิบายเกี่ยวกับการป้อนข้อมูล โดยจะปรากฏในโหมดป้อนข้อมูล ตรงบรรทัดสถานะ (Status Bar) เช่น ฟิลด์ชื่อ อาจจะใส่คำอธิบายว่า "พิมพ์ชื่อ ไม่ต้องใส่คำนำหน้า" หรือฟิลด์ลำดับที่ข้อมูล อาจจะใส่คำอธิบายเป็น "ให้กด <Enter> ผ่าน" เป็นต้น

13. Field Properties ฟิลด์แต่ละรายการ จะมีคุณสมบัติ (Field Properties) กำกับด้วยเสมอ ดังนั้นจะต้องกำหนดคุณสมบัติของฟิลด์ให้เหมาะสม เช่น ขนาดของฟิลด์ (Field Siz)
(กิตติ และ กุลชน, 2547)