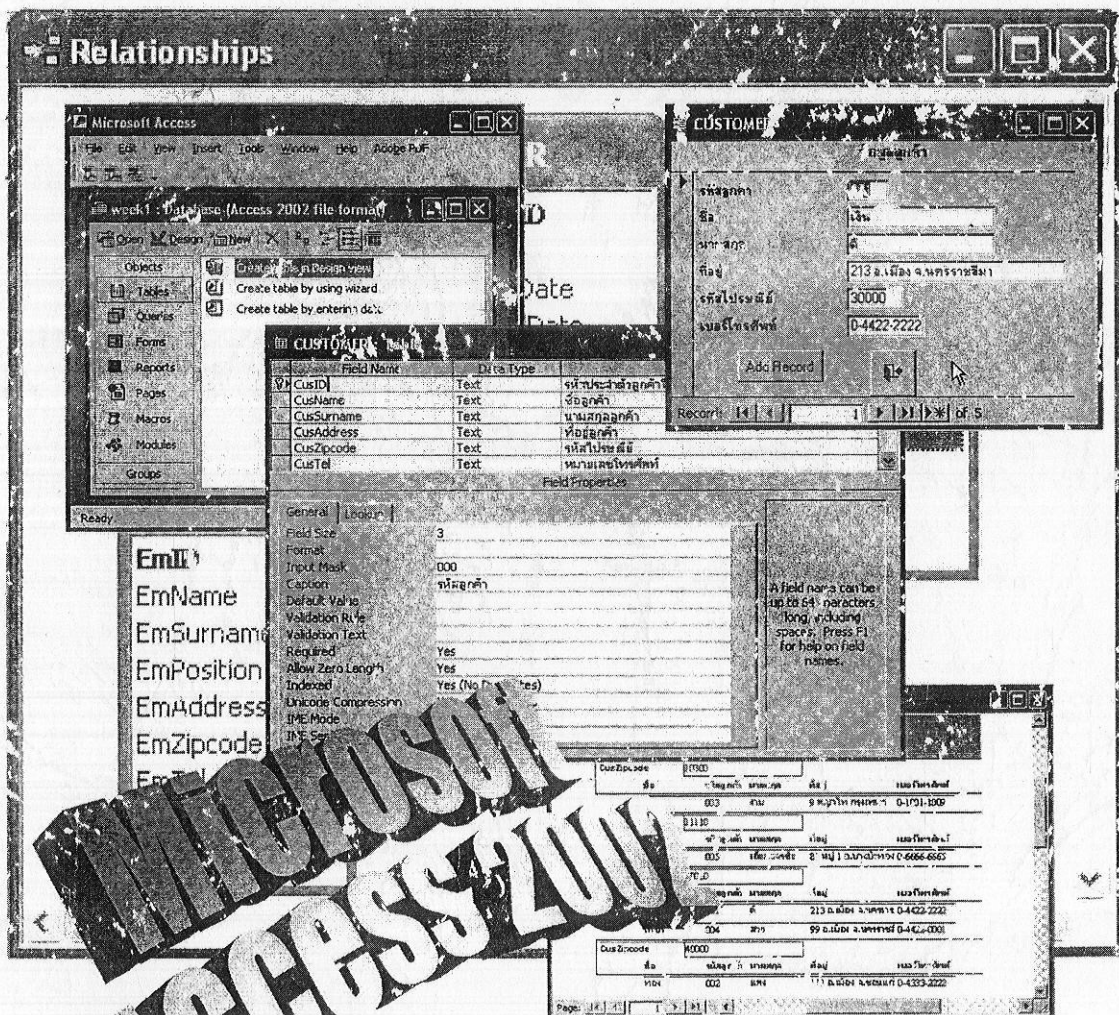


202103 เทคโนโลยีสารสนเทศ 2

Information Technology II



Microsoft ACCESS 2003

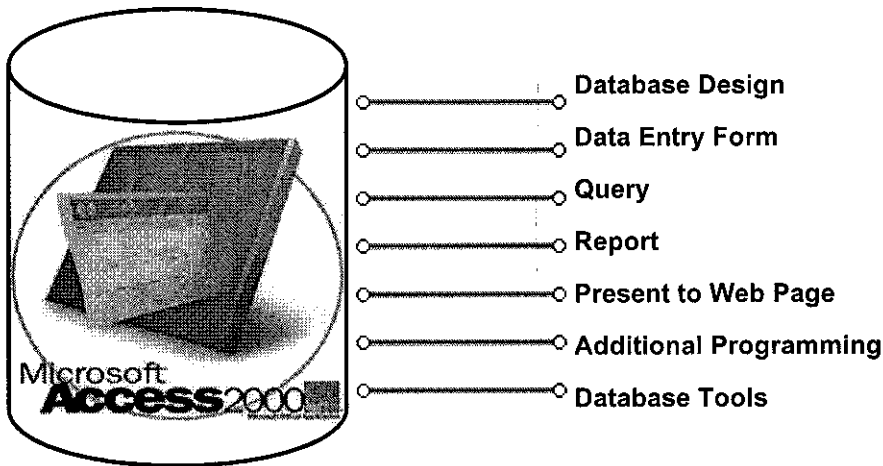
อ.พรพนันต์ เอี่ยมจรรย์ชัย

สารบัญ

	หน้า
บทนำ	
ความสามารถของ Microsoft Access 2002	1
ฐานข้อมูลเชิงสัมพันธ์สัมพันธ์ (Relational Database)	3
ขั้นตอนการออกแบบฐานข้อมูล Microsoft Access 2002	6
องค์ประกอบของ Microsoft Access 2002	11
ขั้นตอนการสร้างตาราง (Table)	19
ปฏิบัติการครั้งที่ 1	
การสร้างฐานข้อมูล	20
การกำหนดโครงสร้างตารางและองค์ประกอบของฟิลด์	25
ภาคผนวก : โครงสร้างตาราง	40
ปฏิบัติการครั้งที่ 2	
การเชื่อมความสัมพันธ์ระหว่างตาราง	44
การกรอกข้อมูลใน Datasheet	50
ปฏิบัติการครั้งที่ 3	
ประเภทและมุมมองในการสร้าง Query	53
การสร้าง Query จากตารางเดียว	54
การสร้าง Query จาก 2 ตารางขึ้นไป	65
การเพิ่มความสามารถให้กับ Query	67
ปฏิบัติการครั้งที่ 4	
ประเภทของ Form	77
คอนโทรล (Controls)	78
การสร้าง Form ด้วย Form Wizard	80
การสร้างคอนโทรล	86
ปฏิบัติการครั้งที่ 5	
องค์ประกอบ, มุมมอง และประเภท Report	101
การสร้าง Report ด้วย Report Wizard	103
บรรณานุกรม	115

Microsoft Access 2002 ภาคทฤษฎี

Microsoft Access เป็นฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ที่ได้รับความนิยมใช้งานในการจัดการฐานข้อมูลของหน่วยงานหรือองค์กรขนาดเล็กมากที่สุดอีกโปรแกรมหนึ่งที่ช่วยให้ผู้ใช้งานจัดการกับข้อมูลปริมาณที่มากมาย ได้อย่างมีประสิทธิภาพ ทั้งในด้านการจัดเก็บข้อมูล (เพิ่ม – ลบ – แก้ไข – ปรับปรุง) การสืบค้นข้อมูล การจัดทำรายงานข้อมูล การนำเสนอข้อมูลบนเครือข่าย และการสำรองข้อมูล



Microsoft Access เป็นโปรแกรมที่ใช้ในการจัดการฐานข้อมูลหรือเรียกว่า DBMS (Database Management System) เช่นเดียวกับโปรแกรมการจัดการฐานข้อมูลอื่นๆ เช่น dBase, FoxPro, Oracle, SQL Server เป็นต้น แต่โปรแกรม Microsoft Access เป็นระบบการจัดการฐานข้อมูลสำหรับคอมพิวเตอร์ส่วนบุคคล ซึ่งอยู่ในระดับเดียวกับ dBase, FoxPro ส่วนระบบจัดการฐานข้อมูลขนาดใหญ่ที่ใช้เครื่อง Server ในการรองรับผู้ใช้หลายคนพร้อมกัน นิยมใช้โปรแกรม Oracle, SQL Server

ความสามารถของ Microsoft Access 2002

การที่หน่วยงานและองค์กรต่างๆ นำระบบการจัดการฐานข้อมูลมาใช้งานนั้น ก็เพื่อลดความซ้ำซ้อนของข้อมูล และมีการจัดการกับข้อมูลอย่างมีประสิทธิภาพ ทำให้สืบค้นข้อมูลได้อย่างรวดเร็ว ถูกต้อง Microsoft Access 2002 ก็ถือเป็นอีกโปรแกรมจัดการฐานข้อมูลหนึ่งซึ่งมีความสามารถต่างๆ ดังนี้

- การสร้างฐานข้อมูล ประกอบด้วย การออกแบบฐานข้อมูล การสร้างตาราง และการสร้างความสัมพันธ์ระหว่างตาราง

- การจัดการข้อมูล ประกอบด้วย การนำข้อมูลเข้าสู่ฐานข้อมูลด้วยวิธีการต่างๆ การปรับปรุงแก้ไขข้อมูล การลบข้อมูลออกจากฐานข้อมูล
- การค้นหาข้อมูล เป็นการเรียกดูข้อมูลที่ผู้ใช้งานสนใจ โดยสามารถกำหนดเงื่อนไขหรือมุมมองของข้อมูลที่ต้องการได้ นอกจากนี้ยังสามารถถ่วงน้ำหนักหรือจัดสรรเฉพาะข้อมูลที่ต้องการได้อย่างมีประสิทธิภาพ รวมถึงการวิเคราะห์ข้อมูลต่างๆ
- การสร้างรายงาน เป็นการแสดงข้อมูลในรูปแบบของรายงานซึ่งสามารถแสดงในลักษณะข้อความ แผนภาพ แผนภูมิ และกราฟได้ ซึ่งช่วยให้ผู้ใช้งานมองเห็นภาพของข้อมูลในรูปแบบต่างๆ เพื่อใช้ในการบริหารหรือดำเนินการต่างๆ ต่อไปได้อย่างมีประสิทธิภาพ
- การจัดการความเรียบร้อยของข้อมูล เป็นการจัดการข้อมูลภายในฐานข้อมูลให้เกิดความถูกต้อง ครบถ้วน และเป็นระเบียบเรียบร้อย รวมถึงการจัดเก็บข้อมูลที่เหมาะสมโดยใช้พื้นที่ในการจัดเก็บที่มีประสิทธิภาพ
- การเขียนโปรแกรมเพื่อสร้าง **Application** เป็นการเพิ่มความสามารถให้กับผู้ใช้งานฐานข้อมูล ในการพัฒนาหรือเขียนโปรแกรมเพื่อช่วยในการทำงาน เพิ่มความสะดวกสบาย และเพิ่มประสิทธิภาพในการใช้งานของฐานข้อมูล

นอกจากนี้แล้วโปรแกรม Microsoft Access ได้มีการพัฒนามาอย่างต่อเนื่องตั้งแต่เวอร์ชัน 1.0 ซึ่งทำงานบนระบบปฏิบัติการ Windows 3.1X มาจนถึงเวอร์ชัน 2002 หรือเวอร์ชันที่ 10 ของ Access ซึ่งโปรแกรมนี้มีความสามารถเพิ่มขึ้นจากเวอร์ชัน 9 (Access 2000) อย่างมากมาย ได้แก่

- รองรับการใช้งานภาษา **XML** ปัจจุบันภาษา XML จะเป็นภาษาที่มีความสำคัญต่อการเขียนโปรแกรมที่มีการติดต่อกับฐานข้อมูล ซึ่ง Microsoft Access 2002 ได้เพิ่มความสามารถนี้ด้วยเช่นเดียวกัน จึงทำให้การติดต่อกับฐานข้อมูลและการส่งผ่านข้อมูลในรูปแบบภาษา XML ทำงานได้อย่างมีประสิทธิภาพ
- **Access Project** : ทำงานร่วมกับ **SQL Server 2000** ในปัจจุบันมักนิยมทดลองสร้างฐานข้อมูล และทดสอบฐานข้อมูลด้วย Microsoft Access ก่อนเพื่อความแน่ใจว่าฐานข้อมูลดังกล่าวสามารถใช้งานได้อย่างถูกต้อง จากนั้นจึงมีนำไปใช้งานกับระบบงานขนาดใหญ่ เช่น SQL Server เป็นต้น ซึ่งนิยมจัดการผ่าน Access Project ซึ่งจะมีความสะดวกในการจัดการ
- ความสามารถด้านความปลอดภัย แม้ว่า Microsoft Access เป็นโปรแกรมจัดการฐานข้อมูลที่เน้นการใช้งานกับเครื่องที่เป็น Stand Alone ก็ตาม แต่ผู้ใช้สามารถประยุกต์การใช้งานให้เข้ากับเครือข่ายคอมพิวเตอร์ที่มีผู้ใช้งานหลายๆ คนได้ โดยมีความสามารถในการรักษาความปลอดภัยของข้อมูล และสามารถกำหนดสิทธิผู้ใช้งานฐานข้อมูลในลักษณะ Workgroup ได้
- การวิเคราะห์ข้อมูลขั้นสูงด้วย **Pivot View** และ **Pivot Chart** ผู้ใช้งานสามารถนำข้อมูลที่จัดเก็บไว้ในฐานข้อมูลมาวิเคราะห์เพื่อใช้ในการทำงานอื่นๆ ต่อไปได้ โดย

Microsoft Access สามารถถลั่นกรองข้อมูล และวิเคราะห์ข้อมูลผ่าน Pivot View และ Pivot Chart ได้เช่นเดียวกับการใช้งาน Microsoft Excel

- การทำ Redo และ Undo ได้หลายระดับ Microsoft Access เพิ่มความสามารถในการทำ Redo เพื่อทำซ้ำ และการทำ Undo เพื่อย้อนกลับ ได้หลายระดับและมากกว่าเวอร์ชันเดิม แม้ว่าผู้ใช้งานจะบันทึกการเปลี่ยนแปลงไปแล้วก็ตาม โปรแกรมก็สามารถย้อนคืนการทำงานได้ ซึ่งช่วยให้ผู้ใช้ทำงานได้อย่างสะดวกมากยิ่งขึ้น
- การสร้าง SubForm และ SubReport ได้ Microsoft Access สามารถจัดการกับฟอร์มและรายงานต่างๆ ตามกลุ่มผู้ใช้ได้อย่างมีประสิทธิภาพ

ก่อนการเรียนรู้เรื่องการสร้างฐานข้อมูลด้วยโปรแกรม Microsoft Access 2002 ผู้เรียนควรมีความรู้พื้นฐานเกี่ยวกับฐานข้อมูลเชิงสัมพันธ์ องค์ประกอบของฐานข้อมูลสัมพันธ์ และขั้นตอนการออกแบบฐานข้อมูลก่อน เนื่องจากความรู้พื้นฐานเหล่านี้จะช่วยให้ผู้เรียนมีความรู้ความเข้าใจ และสามารถสร้างฐานข้อมูลด้วยโปรแกรม Microsoft Access 2002 ได้อย่างถูกต้องและมีประสิทธิภาพ

ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เป็นฐานข้อมูลที่พัฒนาขึ้นในปี ค.ศ.1970 โดย อี เอฟ คอดด์ (E.F Codd) ฐานข้อมูลเชิงสัมพันธ์เป็นรูปแบบฐานข้อมูลที่เข้าใจง่ายสำหรับผู้ใช้งาน ไม่ซับซ้อน และมีรูปแบบฐานข้อมูลที่มีระบบการจัดการฐานข้อมูล (Database Management System : DBMS) สนับสนุนในการจัดข้อมูล และตอบสนองความต้องการของผู้ใช้ได้เป็นอย่างดีและมีประสิทธิภาพ

องค์ประกอบของฐานข้อมูลเชิงสัมพันธ์

1. **Entity (เอนทิตี)** หมายถึง วัตถุหรือสิ่งที่เราสนใจและต้องการจัดเก็บข้อมูลไว้ในฐานข้อมูล ซึ่งวัตถุหรือสิ่งที่เราสนใจต้องมีเอกลักษณ์ในตัวเอง หรือสามารถบอกความแตกต่างของตนเองได้ (เอนทิตีแต่ละตัวมีความแตกต่างกัน) และเอนทิตีถูกสร้างขึ้นโดยชุดของแอตทริบิวต์

ตัวอย่างเช่น มหาวิทยาลัยเทคโนโลยีสุรนารี ประกอบด้วย พนักงาน-เจ้าหน้าที่ปฏิบัติงาน อาจารย์ นักวิจัย และนักศึกษา ซึ่งบุคลากรแต่ละกลุ่มก็มีความแตกต่างกันตามหน้าที่ความรับผิดชอบ จึงสามารถกำหนดเอนทิตีได้ดังนี้คือ

- เอนทิตีบุคลากร ประกอบด้วย พนักงาน-เจ้าหน้าที่ปฏิบัติงาน อาจารย์และนักวิจัย
- เอนทิตีนักศึกษา ประกอบด้วย นักศึกษาระดับปริญญาตรี ปริญญาโทและเอก

2. **Attribute (แอตทริบิวต์)** หมายถึง คุณลักษณะหรือรายละเอียดของเอนทิตีแต่ละตัว เช่น

- เอนทิตีบุคลากร มทส ประกอบด้วย รหัสพนักงาน ชื่อ-นามสกุล วันเดือนปีเกิด
ประเภทบุคลากร หน่วยงานสังกัด
- เอนทิตีนักศึกษา มทส ประกอบด้วย รหัสประจำตัวนักศึกษา ชื่อ-นามสกุล วันเดือนปีเกิด สำนักวิชา สาขาวิชา อาจารย์ที่ปรึกษา
เกรดเฉลี่ยสะสม

3. Relationship (ความสัมพันธ์) หมายถึง ความสัมพันธ์ระหว่างเอนทิตี ซึ่งถือเป็นส่วนสำคัญของฐานข้อมูลเชิงสัมพันธ์ โดยทำหน้าที่เชื่อมโยงข้อมูลระหว่างเอนทิตีต่างๆ ที่สัมพันธ์กัน โดยใช้แอตทริบิวต์ที่เหมือนกัน (มีค่าตรงกัน) เป็นตัวเชื่อมความสัมพันธ์ ซึ่งเราสามารถแบ่งความสัมพันธ์ได้เป็น 3 ประเภท คือ

➤ **ความสัมพันธ์แบบ 1:1 (One to One Relationship)**

ความสัมพันธ์แบบ 1:1 เป็นความสัมพันธ์ที่จัดการง่ายที่สุด เนื่องจากมีความซับซ้อนน้อยที่สุด แต่พบได้ยากที่สุดในฐานข้อมูลเชิงสัมพันธ์ ความสัมพันธ์แบบนี้มีลักษณะดังนี้คือ ข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับข้อมูลอีกเอนทิตีหนึ่งได้เพียงข้อมูลเดียวเท่านั้น ในลักษณะที่เป็นหนึ่งต่อหนึ่ง เช่น ธนาคารกำหนดให้ลูกค้าแต่ละคนสามารถเปิดบัญชีได้เพียงบัญชีเดียวเท่านั้น และแต่ละบัญชีมีเจ้าของบัญชีเพียงคนเดียวเท่านั้น เป็นต้น

➤ **ความสัมพันธ์แบบ 1:M (One to Many Relationships)**

ความสัมพันธ์แบบ 1:M เป็นความสัมพันธ์ที่พบบ่อยที่สุดในฐานข้อมูลเชิงสัมพันธ์ ความสัมพันธ์แบบนี้มีลักษณะดังนี้คือ ข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับข้อมูลอีกเอนทิตีหนึ่งได้ในหลายข้อมูล เช่น ธนาคารกำหนดให้ลูกค้าแต่ละคนสามารถเปิดบัญชีได้ตั้งแต่ 1 บัญชีขึ้นไป และแต่ละบัญชีมีเจ้าของบัญชีได้เพียงคนเดียวเท่านั้น เป็นต้น

➤ **ความสัมพันธ์แบบ M:N (Many to Many Relationships)**

ความสัมพันธ์แบบ M:N เป็นความสัมพันธ์ที่พบบ่อยไม่บ่อยนัก เนื่องจากมีความซับซ้อนมากกว่าความสัมพันธ์แบบ 1:M ความสัมพันธ์แบบนี้มีลักษณะดังนี้คือ ข้อมูลของเอนทิตีหนึ่ง (หลายข้อมูล) มีความสัมพันธ์กับข้อมูลอีกเอนทิตีหนึ่งได้ในหลายข้อมูล ซึ่งความสัมพันธ์ของข้อมูลทั้งสองเอนทิตีจะมีลักษณะเป็นแบบกลุ่มต่อกลุ่ม เช่น ธนาคารกำหนดให้ลูกค้าแต่ละคนสามารถเปิดบัญชีได้ตั้งแต่ 1 บัญชีขึ้นไป และแต่ละบัญชีมีเจ้าของบัญชีได้มากกว่า 1 คน เป็นต้น

4. Key (คีย์) หมายถึง แอตทริบิวต์ที่สามารถใช้ในการบ่งบอกความแตกต่างของแต่ละเรคคอร์ดได้ โดยคีย์แบ่งได้หลายชนิด แต่มีคีย์ที่ผู้เรียนควรทราบดังนี้

➤ **Primary Key (คีย์หลัก)**

คือ Attribute ที่ใช้บ่งบอกความแตกต่างของข้อมูลแต่ละเรคคอร์ดใน Entity โดยที่คีย์หลักอาจเกิดจาก Attribute เดียว หรือหลายแอตทริบิวต์รวมกันก็ได้ ในการเลือก Attribute ที่จะมาใช้เป็นคีย์หลักควรที่จะเลือกจาก Attribute ที่มีข้อมูลภายใน Attribute นั้นที่สุด ไม่มีข้อมูลซ้ำซ้อนกัน และต้องมีค่าข้อมูลในทุกเรคคอร์ด (ไม่เป็นค่าว่าง (Null))

เช่น Entity นักศึกษา มทส ประกอบด้วย Attribute รหัสประจำตัวนักศึกษา ชื่อ-นามสกุล วันเดือนปีเกิด สำนักวิชา สาขาวิชา อาจารย์ที่ปรึกษา เกรดเฉลี่ยสะสม จากการพิจารณาจะพบว่า Attribute รหัสประจำตัวนักศึกษาไม่ซ้ำกัน เนื่องจาก

นักศึกษาแต่ละคนมีรหัสประจำตัวที่แตกต่างกัน จึงเลือกใช้ Attribute รหัสประจำตัวนักศึกษาเป็นคีย์หลัก (Primary Key)

➤ **Candidate Key (คีย์คู่แข่ง)**

คือ Attribute ที่มีคุณสมบัติเป็นคีย์หลัก (Primary Key) ตั้งแต่ 1 Attribute ขึ้นไป โดยปกติแล้วนิยมเลือก Candidate Key ที่มีค่าข้อมูลสั้นที่สุดมาเป็น Primary Key

เช่น Entity นักศึกษา มทส ประกอบด้วย Attribute รหัสประจำตัวนักศึกษา รหัสประจำตัวบัตรประชาชน ชื่อ-นามสกุล วันเดือนปีเกิด สำนักวิชา สาขาวิชา อาจารย์ที่ปรึกษา เกรตเฉลี่ยสะสม จากการพิจารณาจะพบว่า Attribute รหัสประจำตัวนักศึกษา และ Attribute รหัสประจำตัวบัตรประชาชน สามารถใช้เป็นคีย์หลัก (Primary Key) ได้

ดังนั้น Entity นักศึกษา มทส มี Attribute ที่เป็น Candidate Key ซึ่งสามารถใช้เป็นคีย์หลัก (Primary Key) ได้ 2 Attribute ได้แก่ Attribute รหัสประจำตัวนักศึกษา และ Attribute รหัสประจำตัวบัตรประชาชน

➤ **Composite Key (คีย์ร่วม)**

คือ Attribute ตั้งแต่ 2 Attribute ขึ้นไปรวมกันเพื่อให้มีคุณสมบัติเป็นคีย์หลัก (Primary Key) เนื่องจากหากใช้เพียง Attribute ใด Attribute หนึ่ง อาจทำให้เกิดความซ้ำซ้อนของข้อมูลได้ หรือขาดคุณสมบัติการเป็นคีย์หลัก

➤ **Foreign Key (คีย์นอก)**

คือ Attribute ที่ใช้ในการเชื่อมต่อกับ Entity อื่นเพื่อแสดงความสัมพันธ์ระหว่าง Entity โดยคีย์นอกนี้สามารถมีค่าซ้ำกันหรือเป็นค่าว่าง (Not Null) ได้

เช่น ฐานข้อมูลนักศึกษา ประกอบด้วย

- Entity นักศึกษา ประกอบด้วย รหัสประจำตัวนักศึกษา ชื่อ-นามสกุล
วันเดือนปีเกิด สำนักวิชา สาขาวิชา
รหัสอาจารย์ที่ปรึกษา เกรตเฉลี่ยสะสม
- Entity อาจารย์ที่ปรึกษา ประกอบด้วย รหัสอาจารย์ที่ปรึกษา
ชื่อ-นามสกุล สำนักวิชา สาขาวิชา

จากการพิจารณาพบว่า Attribute รหัสอาจารย์ที่ปรึกษา ของ Entity อาจารย์ที่ปรึกษาเป็นคีย์หลัก และปรากฏใน Entity นักศึกษา ซึ่งสรุปได้ว่า Attribute รหัสอาจารย์ที่ปรึกษา ของ Entity นักศึกษา เป็นคีย์นอก (Foreign Key) เพื่อใช้ในการเชื่อมความสัมพันธ์ของข้อมูลระหว่าง Entity อาจารย์ที่ปรึกษาเข้ากับ Entity นักศึกษา

สำหรับฐานข้อมูลเชิงสัมพันธ์ที่สร้างขึ้นจากโปรแกรม Microsoft Access 2002 นั้นมีการใช้คำศัพท์ที่เรียก Entity และ Attribute ที่แตกต่างออกไป โดยที่เรียก Entity ว่า Table และเรียก Attribute ว่า Field นอกจากนี้ยังมีองค์ประกอบอื่นๆ ที่ควรทราบได้แก่

1. **Table (ตาราง)** หมายถึง แหล่งเก็บรวบรวมข้อมูลที่มีความสัมพันธ์กันอย่างเป็นระบบในฐานข้อมูล โดยแต่ละตารางจะประกอบด้วย คอลัมน์ (Column) ซึ่งเป็นแนวตั้งของตาราง เรียกว่า "ฟิลด์" (Field) และ แถว (Row) ซึ่งเป็นแนวนอนของตาราง เรียกว่า "เรคคอร์ด" (Record)

2. **Record (เรคคอร์ด)** หมายถึง กลุ่มหรือชุดของฟิลด์ที่อยู่แถวเดียวกันในตารางตามแนวนอน เช่น ตารางนักศึกษา มทส ประกอบด้วย ชุดข้อมูลนักศึกษา มทส ซึ่งแต่ละเรคคอร์ดก็มีความข้อมูลที่แตกต่างกันไป เป็นต้น

3. **Field (ฟิลด์)** หมายถึง กลุ่มของข้อมูลที่มีลักษณะหรือคุณสมบัติของเรคคอร์ด เช่น ตารางนักศึกษา มทส ประกอบด้วย รหัสประจำตัวนักศึกษา ชื่อ-นามสกุล วันเดือนปีเกิด สำนักวิชา สาขาวิชา อาจารย์ที่ปรึกษา เกรดเฉลี่ยสะสม เป็นต้น

Table นักศึกษา

รหัสประจำตัว	ชื่อ-นามสกุล	สาขาวิชา	สำนักวิชา	GPAX
B1111111	นายน้อย คงดี	เทคโนโลยีสารสนเทศ	เทคโนโลยีสังคม	3.23
B2222222	น.ส.ชยัน มากกลิ่น	อนามัยสิ่งแวดล้อม	แพทยศาสตร์	3.11
B3333333	น.ส.บุญพร้อม มีดี	เทคโนโลยีผลิตสัตว์	เทคโนโลยีการเกษตร	3.59
B4444444	นายรักเรียน รอบรู้	วิศวกรรมศาสตร์โยธา	วิศวกรรมศาสตร์	3.99

ฟิลด์ (Field)

ตัวอย่างที่ 1 ตารางนักศึกษา

ขั้นตอนการออกแบบฐานข้อมูล Microsoft Access 2002

การออกแบบฐานข้อมูลเป็นขั้นตอนที่ต้องทำก่อนการสร้างฐานข้อมูล และถือเป็นขั้นตอนที่มีความสำคัญเป็นอย่างยิ่ง เนื่องจากถ้าออกแบบฐานข้อมูลไม่ดีหรือผิดพลาดจะส่งผลกระทบต่อการจัดการข้อมูลในฐานข้อมูล รวมถึงการสืบค้นข้อมูลจากฐานข้อมูลอาจเกิดความผิดพลาดได้เช่นเดียวกัน

- ขั้นตอนการออกแบบฐานข้อมูลมีต่อไปนี
1. การกำหนดวัตถุประสงค์ในการออกแบบฐานข้อมูล
 2. การกำหนดองค์ประกอบของฐานข้อมูล
 3. การกำหนดองค์ประกอบของตาราง
 4. การกำหนดความสัมพันธ์ระหว่างตารางต่างๆ ในฐานข้อมูล

1. การกำหนดวัตถุประสงค์ในการออกแบบฐานข้อมูล

ก่อนที่จะสร้างฐานข้อมูลต้องมีการกำหนดก่อนว่าจะสร้างฐานข้อมูลขึ้นมาเพื่อใช้ในงานใด และมีวัตถุประสงค์ในการสร้างฐานข้อมูลอย่างไร เพื่อที่จะได้ทราบถึงข้อมูลที่จะ

รวบรวมและจัดเก็บในฐานข้อมูล รวมถึงเป็นการกำหนดโครงสร้างของฐานข้อมูลด้วย เช่นเดียวกัน ในการรวบรวมข้อมูลนั้นเราสามารถกระทำได้หลายวิธี เช่น การรวบรวมข้อมูลจากเอกสารหรือรายงาน การสอบถามผู้ใช้งาน หรือการสังเกตการทำงานของระบบงานเดิม (กรณีที่ต้องการพัฒนาฐานข้อมูลเพื่อใช้ในการทำงานและปรับปรุงประสิทธิภาพการทำงาน) ซึ่งจะทำให้ได้รับข้อมูลที่ถูกต้องและครบถ้วน

ตัวอย่าง สาขาวิชาเทคโนโลยีสารสนเทศ สำนักวิชาเทคโนโลยีสังคม มหาวิทยาลัยเทคโนโลยีสุรนารี ต้องการสร้างฐานข้อมูลเพื่อจัดเก็บข้อมูลเกี่ยวกับบัณฑิตที่สำเร็จการศึกษาจากสาขาวิชาเทคโนโลยีสารสนเทศ เพื่อจัดเก็บข้อมูลเกี่ยวกับบัณฑิตที่เป็นปัจจุบันมากที่สุด และใช้ในการติดต่อกับบัณฑิตของสาขาวิชา ซึ่งสามารถรวบรวมข้อมูลได้จากเอกสารและติดต่อขอข้อมูลจากบัณฑิตแต่ละคนได้ทางจดหมาย

2. การกำหนดองค์ประกอบของฐานข้อมูล

เมื่อรวบรวมข้อมูลที่ต้องการจัดเก็บในฐานข้อมูลได้แล้ว ขั้นตอนต่อไปคือการวิเคราะห์และกำหนดองค์ประกอบของฐานข้อมูลว่า ภายในฐานข้อมูลที่พัฒนาขึ้นประกอบด้วยตารางอะไรบ้าง และแต่ละตารางมีความสัมพันธ์เกี่ยวข้องกันอย่างไร ซึ่งในขั้นตอนนี้อาจใช้การร่างลงในกระดาษคร่าวๆ ก่อน ซึ่งมีขั้นตอนดังต่อไปนี้

1. การวิเคราะห์ข้อมูลที่เก็บรวบรวมได้ พบว่ามีข้อมูลดังต่อไปนี้ที่ควรจัดเก็บไว้ในฐานข้อมูลบัณฑิต ได้แก่ รหัสบัณฑิต ชื่อ-นามสกุล ที่อยู่ หมายเลขโทรศัพท์ E-mail ภาพถ่ายของบัณฑิต หลักสูตรที่สำเร็จการศึกษา ปีที่สำเร็จการศึกษา ตำแหน่งงาน ชื่อที่ทำงาน ที่อยู่ที่ทำงาน หมายเลขโทรศัพท์ที่ทำงาน หมายเลขโทรสารที่ทำงาน e-mail ที่ทำงาน

2. การจำแนกข้อมูลได้เป็นกลุ่มดังนี้

- ข้อมูลบัณฑิต ประกอบด้วย รหัสบัณฑิต ชื่อ-นามสกุล ที่อยู่ หมายเลขโทรศัพท์ E-mail ภาพถ่ายของบัณฑิต หลักสูตรที่สำเร็จการศึกษา ปีที่สำเร็จการศึกษา
- ข้อมูลที่ทำงาน ประกอบด้วย ตำแหน่งงาน ชื่อที่ทำงาน ที่อยู่ที่ทำงาน หมายเลขโทรศัพท์ หมายเลขโทรสารที่ทำงาน e-mail ที่ทำงาน

3. กำหนดองค์ประกอบของฐานข้อมูล ก่อนกำหนดองค์ประกอบของฐานข้อมูล จะต้องมีการทำนอร์มัลไลซ์ (Normalization) คือการจัดกลุ่มข้อมูลต่างๆ ให้มีความสมบูรณ์ และลดความซ้ำซ้อนของข้อมูล รวมถึงการเชื่อมกลุ่มข้อมูลต่างๆ ให้เป็นระบบ สัมพันธ์และสอดคล้องกัน ซึ่งจากการทำนอร์มัลไลซ์แล้วจะได้ตารางดังต่อไปนี้

- ตารางบัณฑิต จัดเก็บข้อมูลเกี่ยวกับบัณฑิตสาขา ซึ่งจัดเก็บข้อมูลดังต่อไปนี้
 - รหัสบัณฑิต
 - ชื่อ-นามสกุล
 - ที่อยู่
 - หมายเลขโทรศัพท์
 - E-mail
 - ภาพถ่ายบัณฑิต
 - รหัสหลักสูตร
 - ปีที่สำเร็จการศึกษา
 - ตำแหน่งงาน
 - รหัสที่ทำงาน

- ตารางบริษัท จัดเก็บข้อมูลเกี่ยวกับบริษัทที่บัณฑิตทำงาน ซึ่งจัดเก็บข้อมูลดังต่อไปนี้
 - รหัสที่ทำงาน
 - ชื่อที่ทำงาน
 - ที่อยู่
 - หมายเลขโทรศัพท์
 - หมายเลขโทรสาร
 - E-mail
- ตารางหลักสูตร จัดเก็บข้อมูลเกี่ยวกับหลักสูตรสาขาเปิดสอน และใช้ในการเชื่อมโยงกับข้อมูลบัณฑิต (หลักสูตรที่สำเร็จการศึกษา) ซึ่งจัดเก็บข้อมูล
 - รหัสหลักสูตร
 - ชื่อหลักสูตร

3. การกำหนดองค์ประกอบของตาราง

การกำหนดองค์ประกอบของตาราง เป็นการกำหนดว่าตารางประกอบด้วยฟิลด์ใดบ้าง แต่ละฟิลด์จัดเก็บข้อมูลประเภทใด และมีคุณสมบัติอย่างไร รวมถึงการกำหนดคีย์หลัก ซึ่งจากตัวอย่างพบว่ามี 3 ตารางได้แก่ ตารางบัณฑิต ตารางบริษัท และตารางหลักสูตร ซึ่งแต่ละตารางมีองค์ประกอบภายในที่แตกต่างกัน ดังนี้

ตารางบัณฑิต

ชื่อฟิลด์	ประเภทข้อมูล	คำอธิบาย
GradID	Text (ขนาด 8 ตัวอักษร)	รหัสบัณฑิต
GradName	Text (ขนาด 100 ตัวอักษร)	ชื่อ-นามสกุล
GradAddress	Text (ขนาด 255 ตัวอักษร)	ที่อยู่
GradTel	Text (ขนาด 9 ตัวอักษร)	หมายเลขโทรศัพท์
GradEmail	Text (ขนาด 50 ตัวอักษร)	E-mail
GradPic	OLE Object	ภาพถ่ายบัณฑิต
MajorID	Text (ขนาด 2 ตัวอักษร)	รหัสหลักสูตร
GradDate	Text (ขนาด 4 ตัวอักษร)	ปีที่สำเร็จการศึกษา
GradPosition	Text (ขนาด 100 ตัวอักษร)	ตำแหน่งงาน
CompID	Text (ขนาด 2 ตัวอักษร)	รหัสที่ทำงาน

ตารางบริษัท

ชื่อฟิลด์	ประเภทข้อมูล	คำอธิบาย
CompID	Text (ขนาด 2 ตัวอักษร)	รหัสที่ทำงาน
CompName	Text (ขนาด 100 ตัวอักษร)	ชื่อที่ทำงาน
CompAddress	Text (ขนาด 255 ตัวอักษร)	ที่อยู่
CompTel	Text (ขนาด 9 ตัวอักษร)	หมายเลขโทรศัพท์
CompFax	Text (ขนาด 9 ตัวอักษร)	หมายเลขโทรสาร
CompEmail	Text (ขนาด 50 ตัวอักษร)	E-mail

ตารางหลักสูตร

ชื่อฟิลด์	ประเภทข้อมูล	คำอธิบาย
MajorID	Text (ขนาด 2 ตัวอักษร)	รหัสหลักสูตร
MajorName	Text (ขนาด 100 ตัวอักษร)	ชื่อหลักสูตร

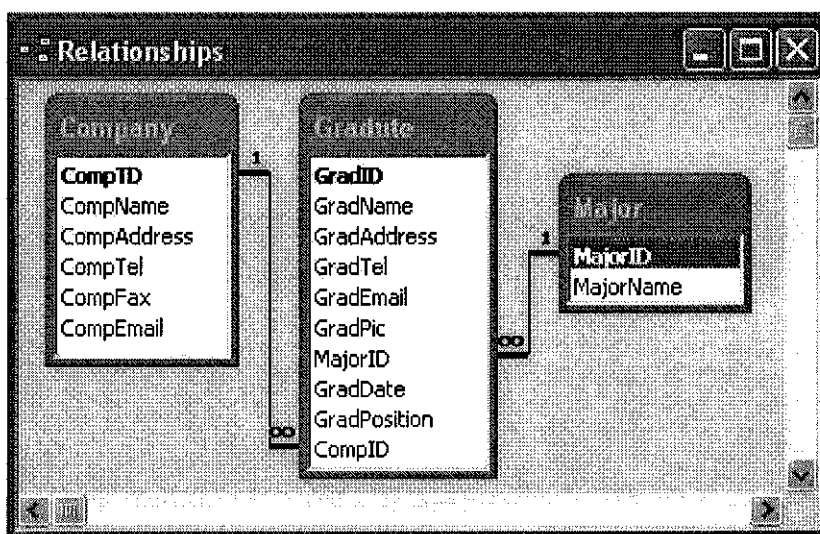
เมื่อกำหนดองค์ประกอบของตารางเสร็จแล้วจึงกำหนดคีย์หลัก (Primary Key)

- ตารางบัณฑิต กำหนดให้ฟิลด์ GradID เป็นคีย์หลัก
- ตารางหลักสูตร กำหนดให้ฟิลด์ MajorID เป็นคีย์หลัก
- ตารางบริษัท กำหนดให้ฟิลด์ CompID เป็นคีย์หลัก

4. การกำหนดความสัมพันธ์ระหว่างตารางต่าง ๆ ในฐานข้อมูล

เมื่อกำหนดองค์ประกอบของตารางเสร็จแล้ว ขั้นตอนต่อไปก็คือการกำหนดความสัมพันธ์ระหว่างตารางในฐานข้อมูล จากการออกแบบจะพบว่ามีทั้งหมด 3 ตาราง คือ ตารางบัณฑิต ตารางหลักสูตร และตารางบริษัท ซึ่งทั้งสามตารางเกี่ยวข้องและมีความสัมพันธ์กันกล่าวคือ

- ตารางหลักสูตรมีความสัมพันธ์กับตารางบัณฑิต
 - ฟิลด์รหัสหลักสูตร (MajorID) ของตารางหลักสูตรมีความสัมพันธ์กับฟิลด์รหัสหลักสูตร (MajorID) ของตารางบัณฑิต และมีรูปแบบความสัมพันธ์เป็นแบบ 1: M เนื่องจากแต่ละหลักสูตรผลิตบัณฑิตออกมามากกว่า 1 คน
- ตารางบริษัทมีความสัมพันธ์กับตารางบัณฑิต
 - ฟิลด์รหัสบริษัท (CompID) ของตารางบริษัทมีความสัมพันธ์กับฟิลด์รหัสบริษัท (CompID) ของตารางบัณฑิต และมีรูปแบบความสัมพันธ์เป็นแบบ 1: M เนื่องจากบัณฑิตทำงานในบริษัท 1 บริษัท บางบริษัทอาจมีบัณฑิตของสาขาทำงานมากกว่า 1 คน

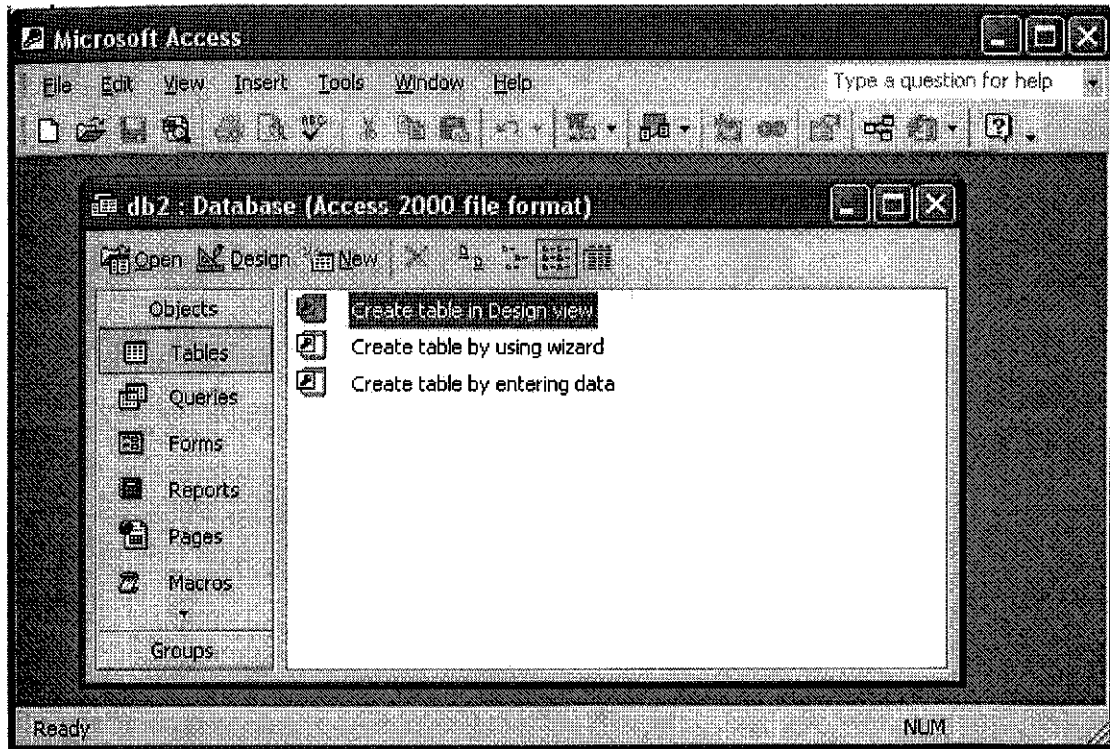


ภาพแสดงความสัมพันธ์ระหว่างตาราง ฐานข้อมูลบัณฑิตสาขาวิชาเทคโนโลยีสารสนเทศ

องค์ประกอบของ Microsoft Access 2002

ส่วนประกอบของจอภาพ

เมื่อเรียกใช้โปรแกรม Access โดยการ Double Click ที่ไอคอน Access จะปรากฏหน้าต่าง Microsoft Access และเมื่อเลือกที่จะสร้างหรือเปิดฐานข้อมูล จะปรากฏหน้าต่างฐานข้อมูล (Database Window) ดังภาพ



ภาพหน้าจอและองค์ประกอบของ Microsoft Access 2002

1. แถบชื่อเรื่อง (Title Bar) เป็นแถบที่อยู่บนสุด จะแสดงชื่อของโปรแกรม Microsoft Access
2. แถบเมนู (Menu Bar) แสดงรายการคำสั่งที่ใช้ในการจัดการไฟล์ แก้ไข เป็นต้น
3. แถบเครื่องมือ (Tool Bar) แสดงปุ่มคำสั่งที่ใช้ภาพแทนตัวอักษร จะแสดงเฉพาะปุ่มคำสั่งที่ใช้งานประจำ เมื่อต้องการใช้งานเพียงแค่คลิกเมาส์ที่ปุ่มนั้น เมื่อมีการใช้งานต่างออกแบบเจ็ดกันปุ่มอาจจะแตกต่างกันบ้าง ดังนั้น อย่ยากงวลหากเห็นความแตกต่างที่เกิดขึ้น
4. หน้าต่างฐานข้อมูล (Database Window) แสดงออกแบบเจ็ดต่างๆ ในฐานข้อมูล แถบด้านบนของ หน้าต่างจะเป็นชื่อของ Database ที่เปิดใช้งาน
5. ปุ่มออกแบบเจ็ด (Object Button) แสดงออกแบบเจ็ดต่างๆ ในฐานข้อมูล (ดูรายละเอียดเรื่องส่วนประกอบของฐานข้อมูล)
6. ปุ่มคำสั่ง (Command Button) เป็นปุ่มที่ใช้ในการสั่งให้ทำงานต่างๆ ของออกแบบเจ็ดที่ใช้งานอยู่ เช่น ใช้งานออกแบบเจ็ด Table

ปุ่มคำสั่ง New หมายถึง สร้างตารางใหม่

ปุ่มคำสั่ง Open หมายถึง เปิด Datasheet ขึ้นมากรอกข้อมูล

ปุ่มคำสั่ง Design หมายถึง กำหนดโครงสร้างตาราง

7. แถบสถานะ (Status Bar) แสดงข้อความต่างๆ เกี่ยวกับสถานะการทำงาน เช่น Caps Lock, Num Lock เป็นต้น

องค์ประกอบของ Microsoft Access 2002

Microsoft Access 2002 ประกอบด้วยส่วนต่างๆ 6 ส่วน หรืออาจเรียกว่า "ออบเจ็กต์ฐานข้อมูล" (Database Object) ตามแต่ละแท็บที่ปรากฏบนหน้าต่าง Database (ดูจากภาพแสดงส่วนประกอบของจอภาพ)

1. **Table (ตาราง)** เป็นส่วนที่ทำหน้าที่ในการกำหนดโครงสร้างของ Database และจัดเก็บข้อมูลที่มีความสัมพันธ์กันไว้ด้วยกัน โดยจะเก็บข้อมูลในรูปของแถวและคอลัมน์ ข้อมูลในแต่ละแถวจะเรียกว่า เรคอร์ด (Record) แต่ละคอลัมน์เรียกว่า ฟیلด์ (field)

2. **Query (คิวรี)** ใช้ในการสอบถาม แก้ไข เพิ่มหรือลบข้อมูลในตาราง คิวรีอาจเรียกว่า "ข้อคำถาม" เนื่องจากเราสามารถใช้อิวรีในการสอบถามข้อมูลจากตารางเดียวหรือหลายตารางได้

3. **Form (แบบฟอร์ม)** ใช้ในการทำงานกับข้อมูลให้ง่ายและสะดวก เช่น ใช้แสดงเพิ่ม หรือ แก้ไขข้อมูลจากตารางต่างๆ บนแบบฟอร์ม สามารถเพิ่มข้อความ หรือปุ่มคำสั่งต่างๆ เพื่ออธิบายให้ผู้ใช้งานแบบฟอร์มใช้งานได้ง่าย แม้ว่าจะไม่เข้าใจภายในระบบก็ตาม

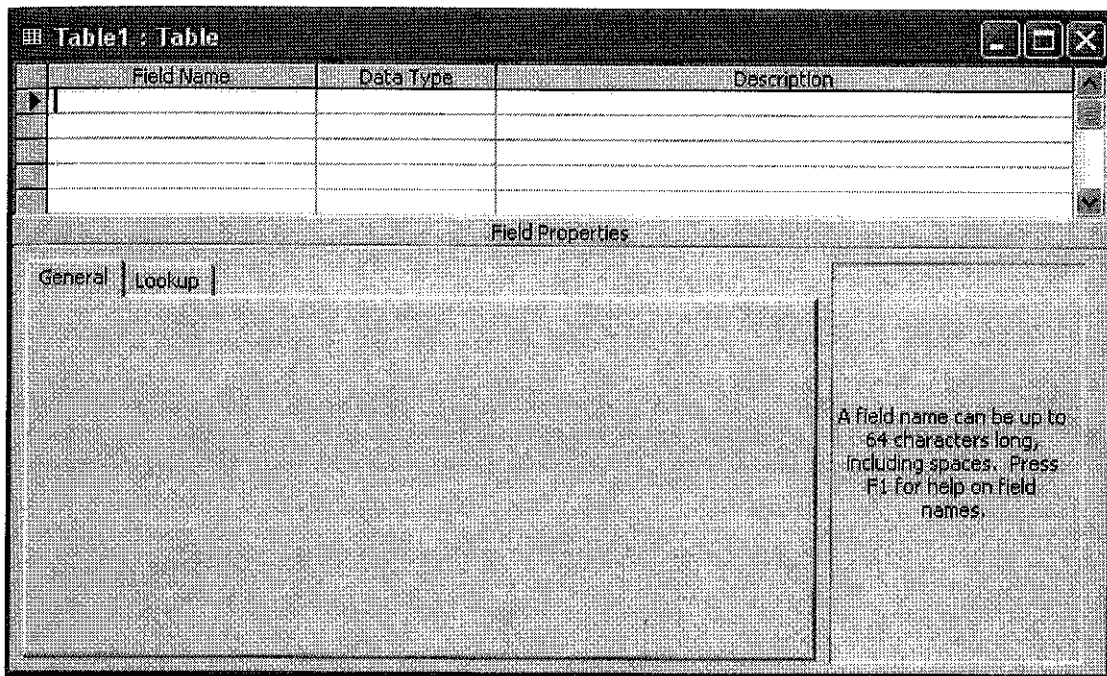
4. **Report (รายงาน)** ใช้ในการแสดงผลข้อมูลทั้งจากบนจอภาพและการพิมพ์ทางเครื่องพิมพ์ การแสดงผลด้วย Report จะไม่เหมือนกับการแสดงผลที่ Form คือ ไม่สามารถเพิ่ม ลบ หรือแก้ไขข้อมูลได้ แต่สามารถรวบรวมผลและนำเสนอข้อมูลสรุปแยกเป็นกลุ่มได้ดีกว่า Form

5. **Macro (มาโคร)** อาจเรียกว่า "ชุดคำสั่ง" ใช้เพื่อเสริมการทำงาน โดยการนำคำสั่งต่างๆ ที่ต้องการให้ทำงานเรียงลำดับกันไปเรื่อยๆ เช่น สร้างชุดคำสั่งให้เปิดฟอร์มโดยอัตโนมัติ หลังจากนั้นให้ Run เมนูเฉพาะงานขึ้นมาและสั่งพิมพ์รายงานทันที เป็นต้น เมื่อต้องการให้มาโครทำงานเพียงแคกดปุ่มควบคุมเท่านั้นเอง

6. **Module (โมดูล)** เป็นการเขียนคำสั่งโปรแกรม แต่ก่อนเขียนด้วย Access Basic Code ปัจจุบันเป็น Visual Basic For Application (VBA) การใช้งานเหมือน Macro แต่ซับซ้อนกว่ามาก หากเพิ่งเริ่มเรียนรู้การใช้งานข้อมูล Access ไม่ควรใช้ Module

คุณสมบัติต่างๆ ของฟیلด์

ก่อนสร้างตารางต้องทำความเข้าใจเกี่ยวกับคุณสมบัติต่างๆ ของฟیلด์ก่อน เพื่อการกำหนด โครงสร้างของตารางจะได้ไม่ต้องแก้ไขไปแก้ไขมา ซึ่งจะมีผลกระทบต่อข้อมูลที่กรอกไปแล้ว เพราะหากคุณสมบัติของฟیلด์เปลี่ยนไป ข้อมูลก็จะเปลี่ยนไปหรืออาจสูญหายได้



ภาพแสดงหน้าต่างการกำหนดโครงสร้างของตาราง (คุณสมบัติของฟิลด์)

1. ชื่อฟิลด์ (Field Name)

กฎในการตั้งชื่อฟิลด์มีดังนี้

1. ยาวได้ไม่เกิน 64 ตัวอักษร
2. ใช้ได้ทั้งตัวอักษรและตัวเลข และมีช่องว่างได้

กรณีที่มีช่องว่างในชื่อฟิลด์ใด เวลาอ้างอิงถึงชื่อนั้นใน Query หรือ Form ต้องใส่ชื่อฟิลด์ไว้ในเครื่องหมาย [] (bracket เช่น [Order ID])

3. ควรตั้งชื่อเป็นภาษาอังกฤษมากกว่าภาษาไทย
4. ควรตั้งให้สื่อความหมายกับคุณสมบัติของ Field นั้น

2. ประเภทของข้อมูลในฟิลด์ (Data Type)

การกำหนดข้อมูลในฟิลด์ต้องระบุที่ช่อง Data Type ในขั้นตอนการกำหนดโครงสร้างตาราง (Table Design) การเก็บข้อมูลต้องพิจารณาให้ดีว่าข้อมูลใดจะเป็นประเภทใด มีความยาวเท่าไร เพื่อให้เหมาะสมกับการทำงาน และเพื่อประหยัดพื้นที่ในการจัดเก็บด้วย เช่น

- กรณีที่พนักงานในบริษัทมีไม่เกิน 1,000 คน การเลือก Number ควรเรียกเป็น Integer ก็พอ ไม่ควรเลือกเป็น Long Integer เพราะกินพื้นที่มากกว่า และไม่มี ความจำเป็นเลยที่ต้องกำหนดให้เป็น Single หรือ Double เพราะไม่ต้องการทศนิยม
- กรณีหมายเลขโทรศัพท์ หมายเลขโทรสารกำหนดเป็น Text เนื่องจากไม่จำเป็นต้องใช้ในการคำนวณ

ประเภทข้อมูล	ขนาด	ชนิดของข้อมูล / คำอธิบาย
Text	255 ตัวอักษร	ตัวอักษร ตัวเลข และเครื่องหมายต่างๆ ที่ไม่ได้ใช้ในการคำนวณ
Memo	65,635 ตัวอักษร	ตัวอักษร ตัวเลข และเครื่องหมายต่างๆ ที่ใช้ในการอธิบาย บันทึก หรือหมายเหตุ
Number	8 ไบต์	ตัวเลขที่ใช้ในการคำนวณ ทั้งจำนวนเต็มและทศนิยม
Date/Time	8 ไบต์	วัน เดือน ปี ระหว่างปีที่ 100 ถึงปีที่ 9999 และเวลา
Currency	8 ไบต์	ตัวเลขที่ใช้ในการคำนวณ และตัวเลขที่เกี่ยวกับสกุลเงิน โดยเก็บตัวเลขจำนวน 15 หลัก (ตัวเลขหน้าทศนิยม) และเก็บตัวเลขทศนิยมไม่เกิน 4 ตำแหน่ง
Auto Number	4 ไบต์, 16 ไบต์	ค่าลำดับที่เพิ่มในตารางโดยอัตโนมัติและไม่ซ้ำกัน เมื่อมีการเพิ่มเรคคอร์ดใหม่ในตาราง
Yes/No	1 บิต	ข้อมูลเชิงตรรกะที่เป็นไปได้เพียง 2 ค่า เช่น True/False, On/Off หรือ Yes/No
OLE Object	1 กิกะไบต์	วัตถุ (Object) หรือสิ่งอื่นๆ ที่สร้างขึ้นจากโปรแกรมอื่นๆ เช่น รูปภาพ เสียง หรือแฟ้มเอกสาร เป็นต้น
Hyperlink	2,048 ตัวอักษร	ข้อความ หรือรูปแบบของข้อความที่ใช้สำหรับเชื่อมโยงไปสู่ข้อมูลอื่นๆ หรือการเชื่อมโยงในรูปแบบเว็บเพจ
Lookup Wizard	ขนาดเท่ากับ คีย์หลักของตารางที่ตั้งข้อมูลมาใช้งาน	ข้อมูลที่ดึงมาจากตารางอื่นที่เชื่อมความสัมพันธ์กัน

3. คำอธิบายเพิ่มเติม (Description)

ส่วนของการอธิบายรายละเอียดของฟิลด์นั้นเพื่อเพิ่มเติมให้ชัดเจนขึ้น เช่น Field ชื่อ EmpName ใน Description จะมีคำอธิบายเพิ่มเติมว่า "ชื่อของพนักงานในบริษัท ความยาวไม่เกิน 25 ตัวอักษร" เป็นต้น ข้อความนี้จะปรากฏที่แถบสถานะด้านล่าง (Status bar) เมื่อคลิกเลือกฟิลด์นี้ใน Form

4. คุณสมบัติของฟิลด์ (Field Properties)

ส่วนที่ใช้กำหนดคุณสมบัติต่างๆ ของ Field ประกอบด้วย

คุณสมบัติฟิลด์	ความหมาย	คำอธิบาย
Field Size	ขนาดของข้อมูล	ความยาวสูงสุดของข้อมูลที่บันทึกลงในฟิลด์นั้น
Format	รูปแบบของข้อมูล	รูปแบบของข้อมูลที่ต้องการแสดง เช่น ตัวเลข ค่าเงิน วันเดือนปี เป็นต้น
Decimal Place	จำนวนตำแหน่งหลังจุดทศนิยม	จำนวนตำแหน่งหลังจุดทศนิยม (สำหรับข้อมูลประเภท Number และ Currency)

คุณสมบัติฟิลด์	ความหมาย	คำอธิบาย
Input Mark	รูปแบบการรับค่าข้อมูล	รูปแบบในการรับค่า หรือบันทึกค่าข้อมูลลงในฟิลด์
Caption	ชื่อฟิลด์เพิ่มเติม	ชื่อฟิลด์ที่ใช้แสดงแทนชื่อฟิลด์ที่กำหนดไว้ในตาราง
Validation Text	ข้อความเตือนในการรับค่าข้อมูล	ข้อความเตือน กรณีที่ผู้ใช้บันทึกค่าข้อมูลในฟิลด์นั้น ไม่ถูกต้อง หรือไม่ตรงตามกฎเกณฑ์ที่กำหนดไว้
Required	ความต้องการข้อมูลของฟิลด์	การบังคับให้ผู้ใช้บันทึกค่าข้อมูลลงในฟิลด์ตามที่กำหนดไว้
Allow Zero Length	การยกเว้นการบันทึกค่าข้อมูลในฟิลด์	การกำหนดให้ฟิลด์นั้นๆ ไม่ต้องบันทึกค่าข้อมูลได้ (การทำให้เป็นค่าว่าง (ค่า Null))
Indexed	การกำหนดดัชนีให้กับฟิลด์	การกำหนดดัชนีให้กับฟิลด์นั้นๆ เพื่อใช้ในการค้นหาข้อมูล หรือการเรียงลำดับข้อมูล

Field Size

การกำหนดขนาดของฟิลด์ ใช้กับฟิลด์ที่มีประเภทข้อมูล (Data Type) เป็นแบบ Text, Number และ Auto Number เท่านั้น ซึ่งประเภทข้อมูลแต่ละชนิดมีรายละเอียดในการกำหนดขนาดของฟิลด์ดังนี้

1. Data Type ประเภท Text

โปรแกรมกำหนดขนาดความยาวตั้งต้นไว้ที่ 50 ตัวอักษร แต่สามารถกำหนดใหม่ได้ตั้งแต่ 0 – 255 ตัวอักษร

2. Data Type ประเภท Number

โปรแกรมกำหนดแบบ Number ไว้ที่ Long Integer (ค่าตัวเลขจำนวนปกติ) แต่สามารถกำหนดใหม่ให้เป็นแบบต่างๆ ได้ ดังนี้

รูปแบบ	ตำแหน่งทศนิยม	คำอธิบาย
Byte	ไม่มี	ตัวเลขจำนวนเต็มตั้งแต่ 0 – 255
Integer	ไม่มี	ตัวเลขจำนวนเต็มตั้งแต่ -32,768 ถึง 32,767
Long Integer	ไม่มี	ตัวเลขจำนวนเต็มตั้งแต่ -2,147,483,648 ถึง 2,147,483,647
Single	7 ตำแหน่ง	ตัวเลขตั้งแต่ -3.4×10^{38} ถึง -1.4×10^{45} สำหรับค่าลบ และ 1.4×10^{45} ถึง 3.4×10^{38} สำหรับค่าบวก
Double	15 ตำแหน่ง	ตัวเลขตั้งแต่ -1.8×10^{308} ถึง -4.5×10^{324} สำหรับค่าลบ และ 4.5×10^{324} ถึง 1.8×10^{308} สำหรับค่าบวก
Replication ID	ไม่มี	Globally unique Identifier (GRID) (ตัวเลขผสมตัวอักษร)
Decimal	28 หลัก	ตัวเลขตั้งแต่ -10^{28-1} ถึง 10^{28-1}

3. Data Type ประเภท Auto Number

โปรแกรมกำหนดแบบ Auto Number ไว้ที่ Long Integer (ค่าตัวเลขจำนวนปกติ) แต่สามารถกำหนดใหม่ให้เป็นแบบต่างๆ ได้ ดังนี้

รูปแบบ	ขนาด	คำอธิบาย
Long Integer	4 ไบต์	จำนวนเต็มที่เพิ่มขึ้นทีละ 1 หรือจากการสุ่ม
Replication ID	16 ไบต์	ตัวเลขผสมกับตัวอักษร (Globally unique Identifier (GRID))

Format

การกำหนดรูปแบบของฟิลด์ ใช้กับฟิลด์ที่มีประเภทข้อมูล (Data Type) ทุกประเภท แต่ที่นิยมกำหนดรูปแบบของฟิลด์ได้แก่ข้อมูลประเภท Text, Number, Currency, Date/Time และ Yes/No ซึ่งประเภทข้อมูลแต่ละชนิดมีรายละเอียดในการกำหนดขนาดของฟิลด์ดังนี้

1. Data Type ประเภท Text

สัญลักษณ์	คำอธิบาย	ตัวอย่าง
@	แทนอักษร ตัวเลข หรือเครื่องหมาย 1 ตัว กรณีที่ไม่ป้อนค่าข้อมูล โปรแกรมจะกำหนดให้ตำแหน่งนั้นเป็นค่าว่าง (Null)	รูปแบบ @@-@@@ ผลลัพธ์ 12-345 1-234
&	แทนอักษร ตัวเลข หรือเครื่องหมาย 1 ตัว กรณีที่ไม่ป้อนค่าข้อมูล โปรแกรมจะไม่กำหนดให้ตำแหน่งนั้นเป็นค่าว่าง (Null)	รูปแบบ &&-&&& ผลลัพธ์ 1-234 -123
<	ตัวอักษรทุกตัวแสดงเป็นตัวพิมพ์เล็ก	DATABASE
>	ตัวอักษรทุกตัวแสดงเป็นตัวพิมพ์ใหญ่	database

2. Data Type ประเภท Number และ Currency

รูปแบบ	คำอธิบาย	ตัวอย่าง
General Number	รูปแบบปกติ (ค่า Default) ตามค่าที่ป้อนข้อมูล	9999.989
Currency	รูปแบบสกุลเงินบาท โดยมีเครื่องหมายจุลภาค (,) คั่นระหว่างตัวเลขตำแหน่งที่ 3 กับ 4 และตำแหน่งที่ 6 กับ 7	฿9,999.99
Euro	รูปแบบสกุลเงินยูโร โดยมีเครื่องหมายจุลภาค (,) คั่นระหว่างตัวเลขตำแหน่งที่ 3 กับ 4 และตำแหน่งที่ 6 กับ 7	€9,999.99
Fixed	รูปแบบปกติตามค่าที่ป้อนข้อมูล แต่จะแสดงตัวเลขอย่างน้อย 1 ตัว และตัวเลขหลังจุดทศนิยม 2 ตำแหน่ง	9999.99
Standard	รูปแบบมาตรฐาน โดยใช้เครื่องหมายจุลภาค (,) ในการแยกตัวเลข 3 หลัก และมีตัวเลขหลังจุดทศนิยม 2 ตำแหน่ง	9,999.99

รูปแบบ	คำอธิบาย	ตัวอย่าง
Percent	รูปแบบค่าเปอร์เซ็นต์ โดยหารค่าที่ป้อนข้อมูลด้วย 100 แล้ว เติมเครื่องหมาย % ไว้ด้านหลังตัวเลข และมีตัวเลขหลังจุด ทศนิยม 2 ตำแหน่ง	99.00%
Scientific	รูปแบบตัวเลขมาตรฐานทางวิทยาศาสตร์	9.99E+03

3. Data Type ประเภท Date/Time

รูปแบบ	คำอธิบาย	ตัวอย่าง
General Date	รูปแบบวันที่และเวลาปกติ (ค่า Default) เป็นการแสดงผล ที่รวมระหว่างรูปแบบ Short Date และ Long Time	28/4/2549 10:06:30
Long Date	รูปแบบวันที่ปกติตามที่กำหนดค่าวันที่ของ Date and Time ใน Control Panel ของระบบปฏิบัติการ Windows	28 เมษายน 2549
Medium Date	รูปแบบวันที่แบบสั้น (วันที่ เดือน(อักษรย่อ) ปี(ย่อ))	28-มิ.ย.-49
Short Date	รูปแบบวันที่แบบสั้น (วันที่ เดือน(เลขเดือน) ปี(ย่อ))	28/4/49
Long Time	รูปแบบเวลาปกติตามที่กำหนดค่าเวลาของ Date and Time ใน Control Panel ของระบบปฏิบัติการ Windows	10:06:30
Medium Time	รูปแบบเวลาแบบสั้น (ชั่วโมง นาที หน่วยเวลา AM/PM)	10:06 AM
Short Time	รูปแบบเวลาแบบสั้น (ชั่วโมง นาที)	10:06

4. Data Type ประเภท Yes/No

รูปแบบ	คำอธิบาย	ตัวอย่าง
True / False	รูปแบบค่าจริง (True) และค่าเท็จ (False)	True / False
Yes / No	รูปแบบค่าใช่ (Yes) และไม่ใช่ (No)	Yes / No
On / Off	รูปแบบค่าเปิด (On) และค่าปิด (Off)	On / Off

Decimal Place

การกำหนดตำแหน่งตัวเลขหลังจุดทศนิยม ใช้กับฟิลด์ที่มีประเภทข้อมูล (Data Type) ประเภท Number, Currency ซึ่งโปรแกรมกำหนดมาให้แบบ Auto (ทศนิยม 2 ตำแหน่ง) แต่ผู้ใช้สามารถแก้ไขได้ โดยโปรแกรมกำหนดให้มีทศนิยมได้ตั้งแต่ 0 – 15 ตำแหน่ง

Input Mask

การกำหนดรูปแบบเพื่อใช้ในการรับค่าข้อมูลของฟิลด์ให้ถูกต้องตามหลักเกณฑ์ที่กำหนดไว้ เช่น หมายเลขโทรศัพท์ มีการใช้หลายรูปแบบ เช่น 044-224499, (044) 224499, 0-4422-4499 หรือ 6644224499 เป็นต้น หากไม่มีการ

กำหนดรูปแบบการรับค่าข้อมูล จะทำให้ผู้ใช้แต่ละคนป้อนค่าข้อมูลหลากหลาย ไม่เป็นรูปแบบเดียวกัน จึงควรมีการกำหนดรูปแบบเพื่อความสะดวกในการกรอกข้อมูล และเพื่อป้องกันความผิดพลาดในการกรอกข้อมูลของผู้ใช้งาน

สัญลักษณ์	คำอธิบาย
0	แทนตัวเลข 0 – 9 เท่านั้น และต้องใส่ค่าทุกครั้ง
9	แทนตัวเลข 0 – 9 หรือช่องว่างก็ได้ (ค่า Null) จะใส่หรือไม่ใส่ค่าก็ได้
#	แทนตัวเลข 0 – 9 หรือช่องว่างก็ได้ (ค่า Null) ที่เป็นค่าบวกหรือค่าลบ ก็ได้
L	แทนตัวอักษร และต้องใส่ค่าทุกครั้ง
?	แทนตัวอักษร จะใส่หรือไม่ใส่ค่าก็ได้
A	แทนตัวเลขหรือตัวอักษรเท่านั้น และต้องใส่ค่าทุกครั้ง
a	แทนตัวเลขหรือตัวอักษรเท่านั้น จะใส่หรือไม่ใส่ค่าก็ได้
&	แทนตัวเลข ตัวอักษร และช่องว่าง (ค่าว่าง) และต้องใส่ค่าทุกครั้ง
C	แทนตัวเลข ตัวอักษร และช่องว่าง (ค่าว่าง) จะใส่หรือไม่ใส่ค่าก็ได้
(จุด)	แทนเลขทศนิยม
,	แทนเครื่องหมายคั่นระหว่างตัวเลข ตามหลักสากล
;- /	แทนเครื่องหมายคั่นระหว่างวันที่ และเวลา
<	แทนเครื่องหมายเปลี่ยนตัวอักษรทุกตัวที่บันทึกค่าให้แสดงผลเป็นตัวพิมพ์เล็ก
>	แทนเครื่องหมายเปลี่ยนตัวอักษรทุกตัวที่บันทึกค่าให้แสดงผลเป็นตัวพิมพ์ใหญ่
!	แทนเครื่องหมายที่ให้ใส่ค่าที่บันทึกจากขวาไปซ้าย
\	แทนเครื่องหมายแสดงค่าตามอักษรที่ตามหลังเครื่องหมายนี้

Caption

การกำหนดชื่อใหม่, ชื่ออื่นๆ หรือชื่อที่เป็นภาษาอื่น ที่ใช้แทนชื่อฟิลด์ที่กำหนดไว้ใน Field Name เพื่อให้ผู้ใช้งานเข้าใจ เนื่องจากมีข้อจำกัดในการกำหนดชื่อฟิลด์ในช่อง Field Name เช่น Field Name ตั้งชื่อว่า "StuID" ซึ่งอาจไม่สื่อความหมาย จึงกำหนดชื่อฟิลด์ใหม่ในช่อง Caption ว่า "รหัสประจำตัวนักศึกษา" ซึ่งข้อความดังกล่าวจะปรากฏในส่วนของ Header ของ Field บนหน้า Datasheet ที่ใช้บันทึกข้อมูลลงตาราง

Default Value

การกำหนดค่าเริ่มต้นให้กับฟิลด์นั้นๆ ซึ่งค่าที่กำหนดไว้จะปรากฏในหน้า Datasheet ที่ใช้ในการบันทึกข้อมูล

Validation Rule

การกำหนดเงื่อนไขในการบันทึกค่าข้อมูล โดยกำหนดในรูปของประโยคเงื่อนไข เช่น เงินเดือนพนักงานอยู่ระหว่าง 10,000 บาทถึง 50,000 บาท ให้ใส่เงื่อนไขว่า Between 10000 and 30000 เป็นต้น

Validation Text

การกำหนดข้อความเตือนกรณีที่ป้อนข้อมูลลงตารางผิดพลาด หรือไม่ตรงกับ Validation Rule ที่กำหนดไว้ เช่น พิมพ์คำว่า "กรุณากรอกข้อมูลใหม่อีกครั้ง เงินเดือนพนักงานอยู่ระหว่าง 10,000 – 50,000 บาทเท่านั้น" เป็นต้น

Required

การกำหนดให้มีการป้อนข้อมูลลงในฟิลด์ทุกครั้งหรือไม่ กรณีที่เลือกตัวเลือก "Yes" หมายถึง ต้องการให้มีการป้อนข้อมูลทุกครั้ง และกรณีที่เลือกตัวเลือก "No" หมายถึง ไม่จำเป็นต้องป้อนข้อมูลลงในฟิลด์ทุกครั้ง หรือไม่ต้องป้อนข้อมูลลงในฟิลด์นั้นได้

Allow Zero Length

การกำหนดให้ข้อมูลในฟิลด์นั้นมี Data Type ประเภท Text และ Memo มีค่าเป็น "Null" ได้ กรณีที่เลือกตัวเลือก "Yes" หมายถึง ฟิลด์ดังกล่าวสามารถเป็นค่าว่างได้ และกรณีที่เลือกตัวเลือก "No" หมายถึง ฟิลด์ดังกล่าวไม่สามารถเป็นค่าว่างได้

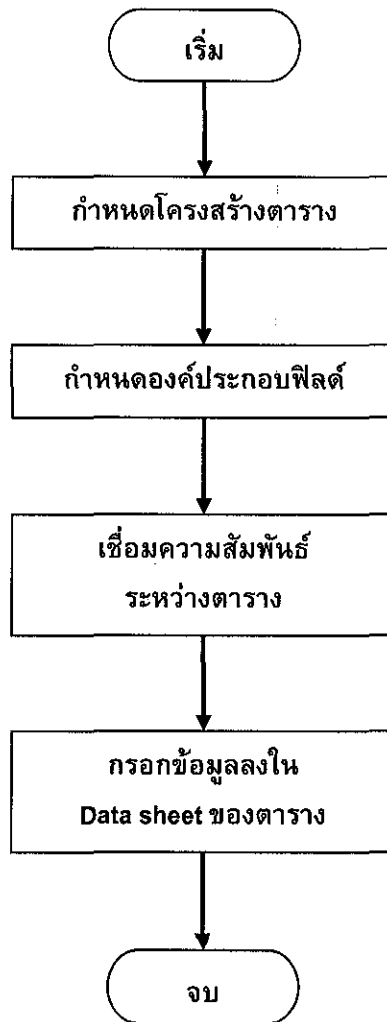
Indexed

การกำหนดดัชนีให้กับฟิลด์ ซึ่งมี 3 ประเภท คือ

รูปแบบ	ความหมาย
No	ไม่มีการกำหนดดัชนีให้กับฟิลด์นั้น
Yes (Duplicates OK)	กำหนดดัชนีให้กับฟิลด์นั้น (แบบซ้ำกันได้)
Yes (No Duplicates)	กำหนดดัชนีให้กับฟิลด์นั้น (แบบซ้ำกันไม่ได้)

ขั้นตอนการสร้างตาราง (Table)

การสร้างตารางเพื่อบันทึกข้อมูลลงฐานข้อมูลด้วยโปรแกรม Microsoft Access 2002 นั้นมีการแบ่งขั้นตอนออกเป็น 2 ขั้นตอน ดังนี้คือ 1) การกำหนดโครงสร้างตาราง, องค์กรประกอบของฟิลด์ และการเชื่อมความสัมพันธ์ระหว่างตาราง 2) การกรอกข้อมูลลงในตาราง



แผนผังแสดงขั้นตอนการสร้างตาราง (Table) ด้วยโปรแกรม Microsoft Access 2002

ปฏิบัติการที่ 1

การสร้างฐานข้อมูลและการกำหนดโครงสร้างตาราง

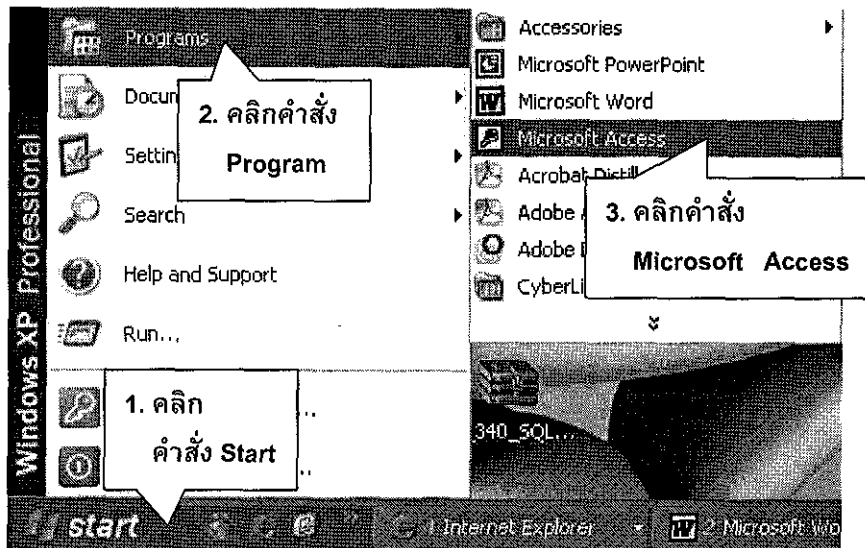
วัตถุประสงค์ : นักศึกษาสามารถ

1. สร้างตารางเพื่อจัดเก็บข้อมูลโดยใช้โปรแกรม Access ได้
2. กำหนดโครงสร้างของตารางได้
3. อธิบายคุณสมบัติของแต่ละฟิลด์ที่กำหนดในตารางได้

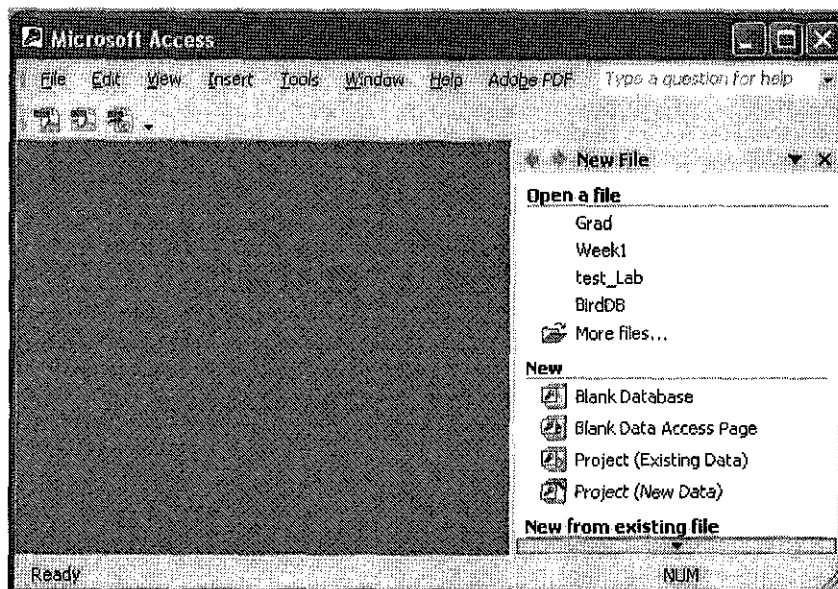
ขั้นตอนปฏิบัติที่ 1.1

การสร้างฐานข้อมูล Week1.mdb


1. เรียกใช้โปรแกรม Microsoft Access



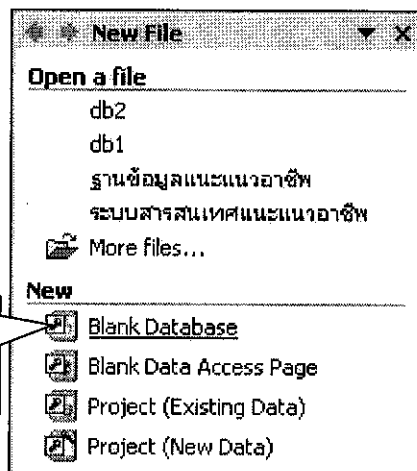
2. ปรากฏหน้าต่างการโปรแกรม Microsoft Access 2002



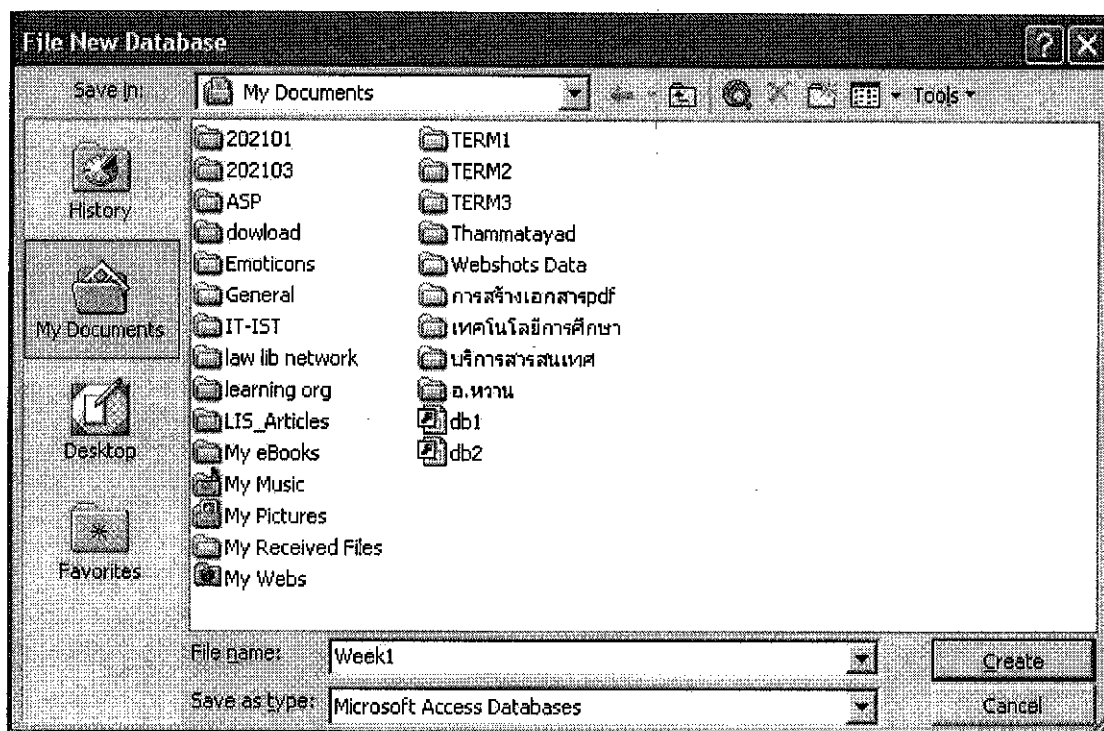
3. คลิกคำสั่ง Blank Database ในหัวข้อ New ซึ่งอยู่ในหน้าต่างงาน (Task pane) เพื่อสร้างฐานข้อมูลใหม่

หรือ คลิกที่ไอคอน  บนแถบ Standard Toolbar

คลิกคำสั่ง
Blank Database

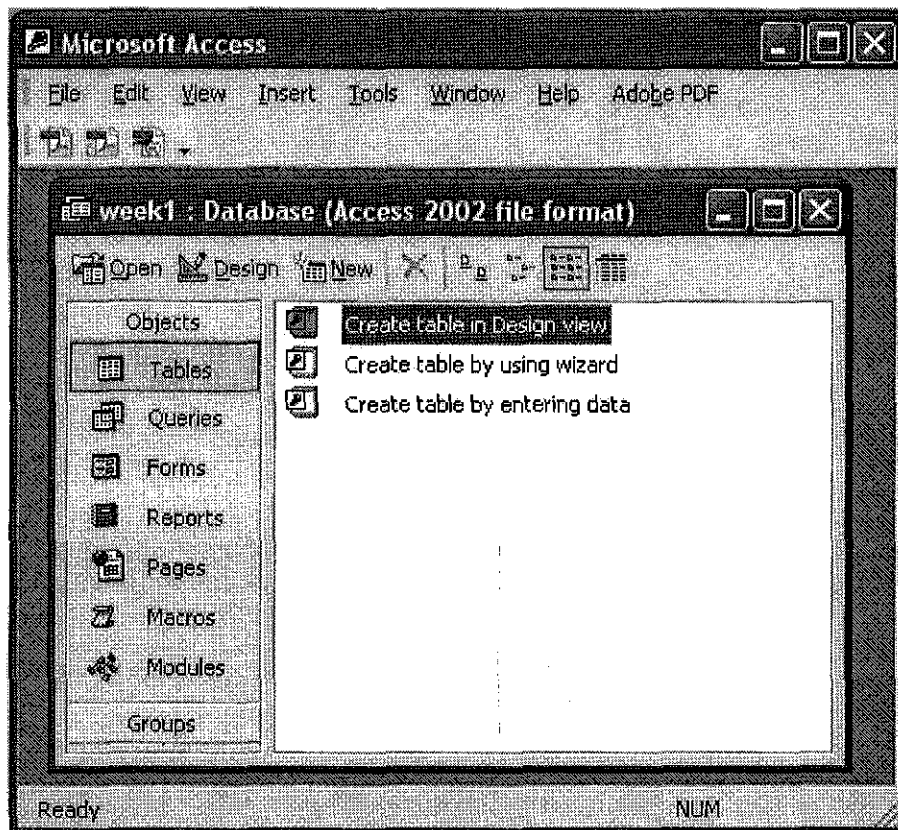


4. ตั้งชื่อฐานข้อมูลใหม่ที่สร้างว่า Week1.mdb ในช่อง File Name : ดังรูป



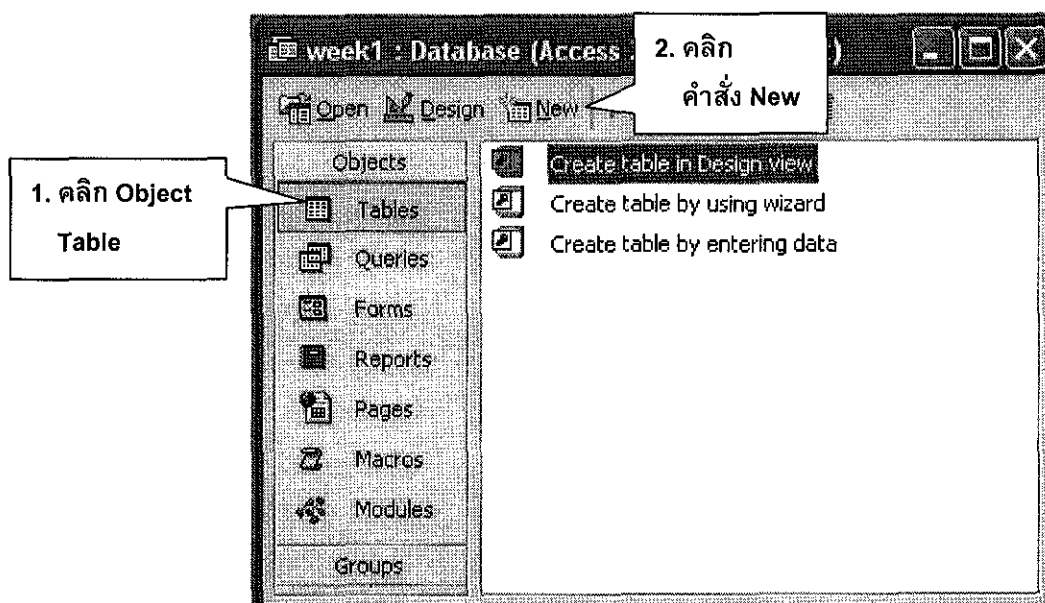
5. เมื่อตั้งชื่อฐานข้อมูลเสร็จเรียบร้อยแล้ว ให้คลิกที่ปุ่ม Create

6. ปรากฏหน้าต่าง Database ชื่อ Week1

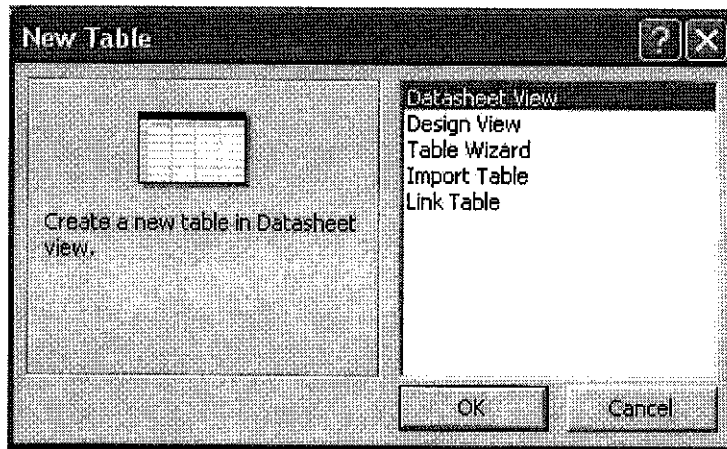


ที่หน้าต่าง Database จะปรากฏ Object ต่างๆ 6 Object ได้แก่ Table, Query, Form, Report, Macro และ Module

7. ให้คลิกเลือก object : Table แล้วคลิกที่ปุ่มคำสั่ง New

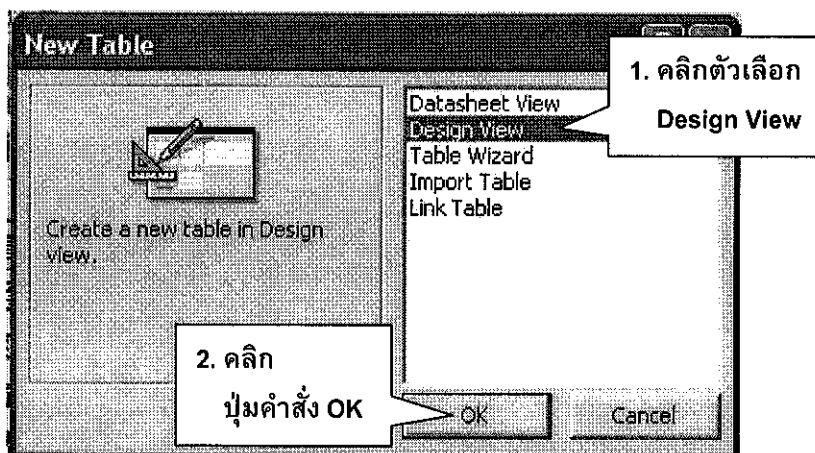


8. ปรากฏหน้าจอ New Table

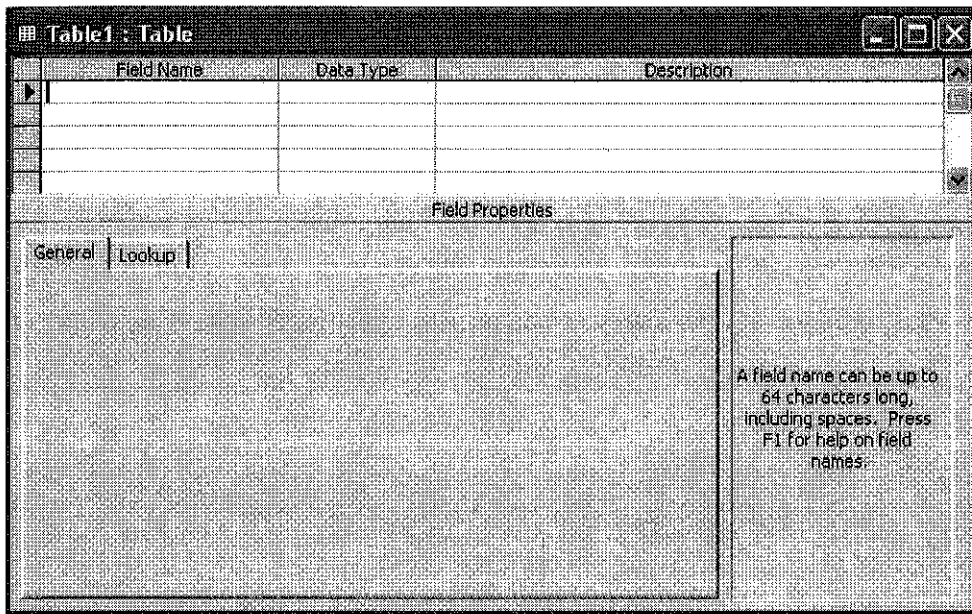


ประเภทตาราง	คำอธิบาย
Datasheet View	ตารางข้อมูลแบบ Datasheet ขนาด 20 คอลัมน์ X 30 แถว
Design View	ตารางข้อมูลที่ใช้กำหนดโครงสร้างตาราง และองค์ประกอบของฟิลด์ด้วยตนเอง
Table Wizard	ตารางข้อมูลที่ Microsoft Access เตรียมโครงสร้างตารางไว้ให้แล้ว
Import Table	ตารางข้อมูลที่สร้างโดยการทำสำเนาจากตารางข้อมูลที่มีอยู่แล้วมาเก็บไว้ในฐานข้อมูลที่ใช้งานอยู่ในปัจจุบัน
Link Table	ตารางข้อมูลที่ Microsoft Access ดึงข้อมูลด้วยวิธีการเชื่อมโยงจากฐานข้อมูลอื่นๆ นำมาใช้งานในฐานข้อมูลปัจจุบัน

9. คลิกที่คำสั่ง Design View จากนั้นคลิกที่ปุ่ม OK



10. ปรากฏหน้าต่างกำหนดโครงสร้างตาราง ดังรูป



หน้าต่างกำหนดโครงสร้างตารางประกอบไปด้วย Field Name (ชื่อฟิลด์), Data Type (ชนิดของฟิลด์), Description (รายละเอียดของฟิลด์), Field Properties (คุณสมบัติของฟิลด์)
(อ่านรายละเอียดจากเรื่องคุณสมบัติของฟิลด์ ในส่วนบทนำ หน้า 11 – 18)

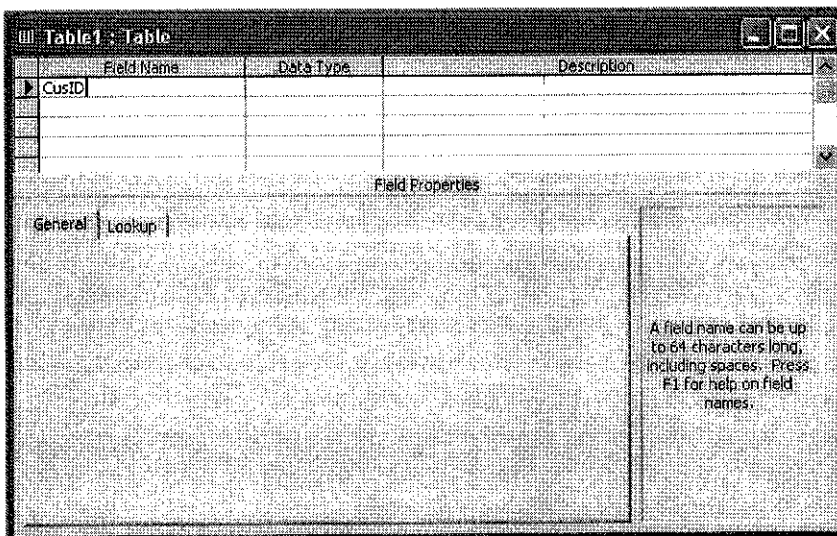
ขั้นตอนปฏิบัติที่ 1.2

การกำหนดโครงสร้างตารางและองค์ประกอบของฟิลด์

คำสั่ง:

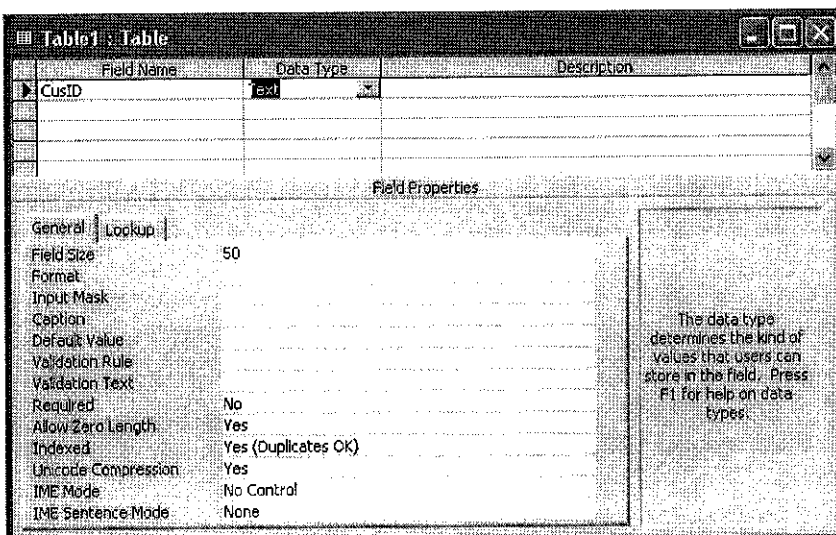
ให้นักศึกษาสร้างตาราง **CUSTOMER, PRODUCT, EMPLOYEE, ORDER** และตาราง **ORDER_Detail** โดยดูโครงสร้างตารางและองค์ประกอบของฟิลด์ จากภาคผนวก หน้า 41 – 43

1. จากหน้าต่าง Table1: Table ให้พิมพ์ชื่อฟิลด์ CusID ลงในช่อง Field Name ดังรูป

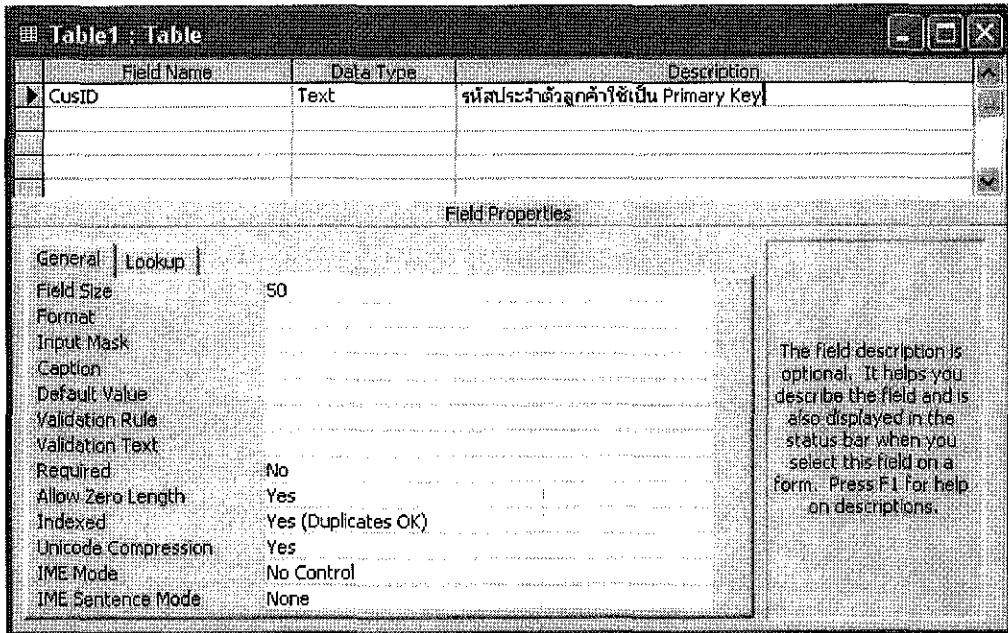


2. เมื่อพิมพ์ชื่อฟิลด์เสร็จแล้วให้กดปุ่ม Enter เคอร์เซอร์ (Curser) จะเลื่อนไปช่อง Data Type โดยอัตโนมัติ

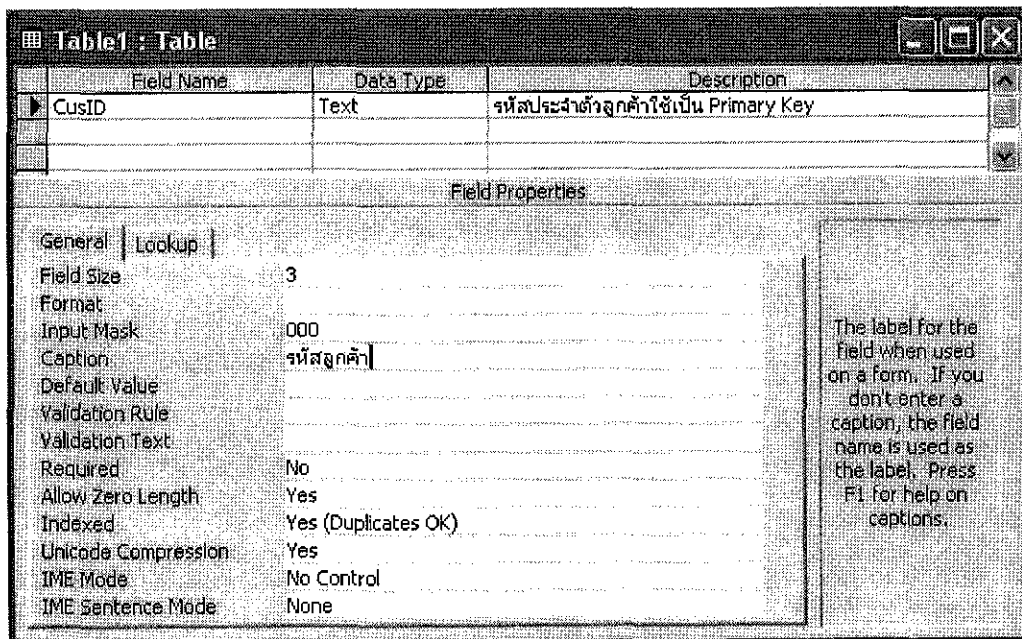
- โดยปกติโปรแกรม Microsoft Access 2002 จะกำหนดค่าเริ่มต้น (Default) ของ Data Type ไว้ที่ Text หากต้องการกำหนดเป็นชนิดอื่น ให้คลิกที่ช่องนั้นจะมีปุ่มลูกศรให้เลือกชนิดของฟิลด์แบบต่างๆ ได้ สำหรับฟิลด์นี้ให้กำหนดเป็น Text (ไม่ต้องแก้ไขค่า Data Type)



3. เมื่อเลือกชนิดของข้อมูลแล้วให้กด Enter เคอร์เซอร์จะปรากฏที่ช่อง Description ให้นักศึกษาพิมพ์รายละเอียดของฟิลด์ ดังนี้ “รหัสประจำตัวลูกค้า ใช้เป็น Primary key”




4. ที่ส่วน Field Properties ให้นักศึกษากำหนดคุณสมบัติต่างๆ ของฟิลด์ (ดูจากภาคผนวก หน้า13-18)

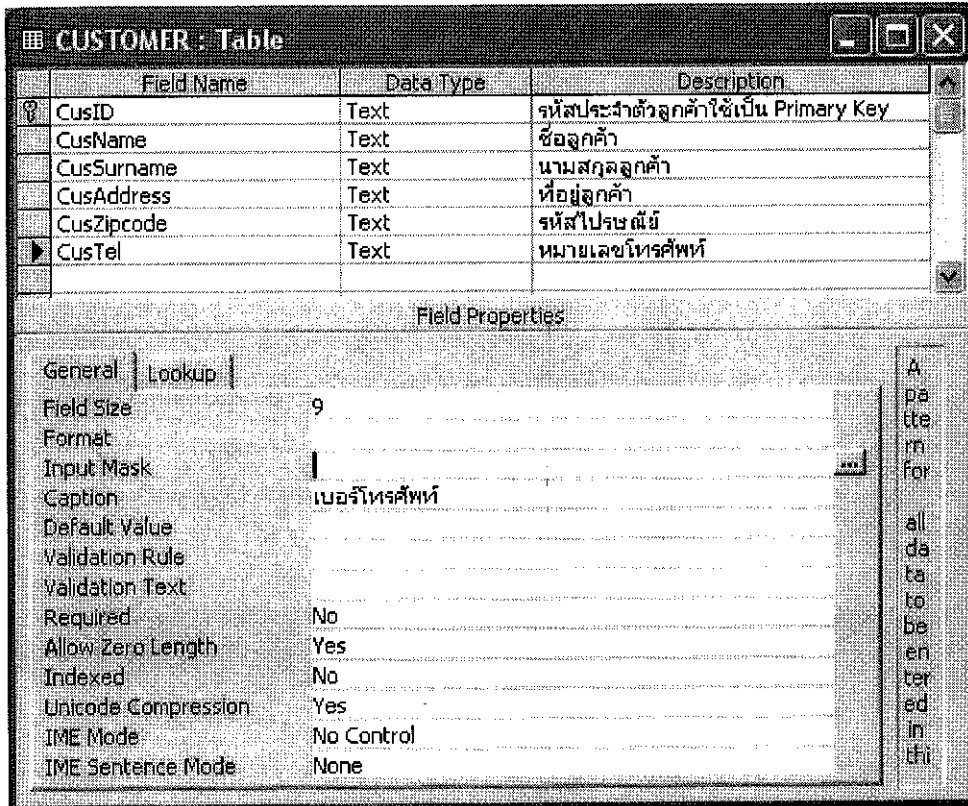



TIP

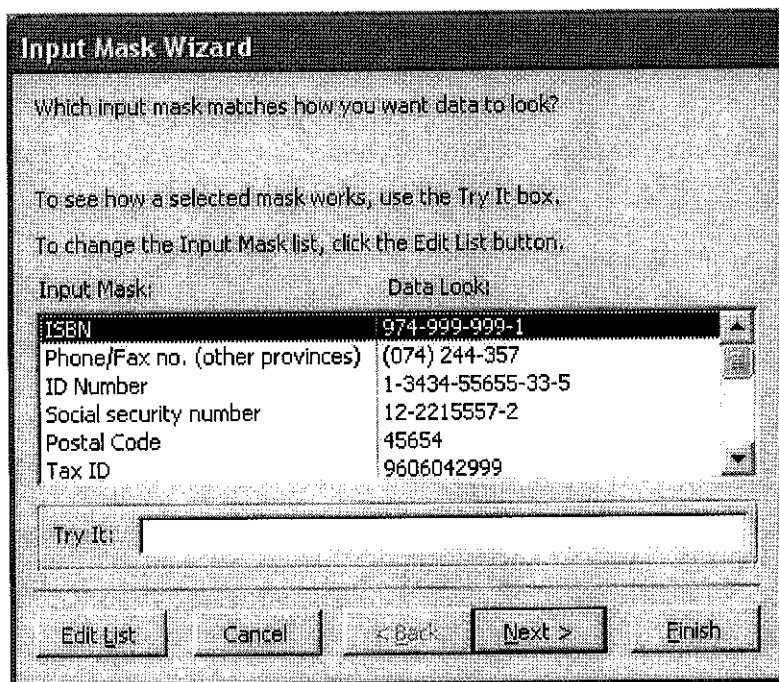
การเลือกฟิลด์ (เลื่อนเมาส์ไปที่ด้านหน้าของฟิลด์ที่ต้องการเลือก เช่น ฟิลด์ CusID จะปรากฏลูกศรสีดำ (ให้คลิกเมาส์จะปรากฏแถบสีดำที่ฟิลด์นั้น)

การกำหนด Field Properties: Input Mask

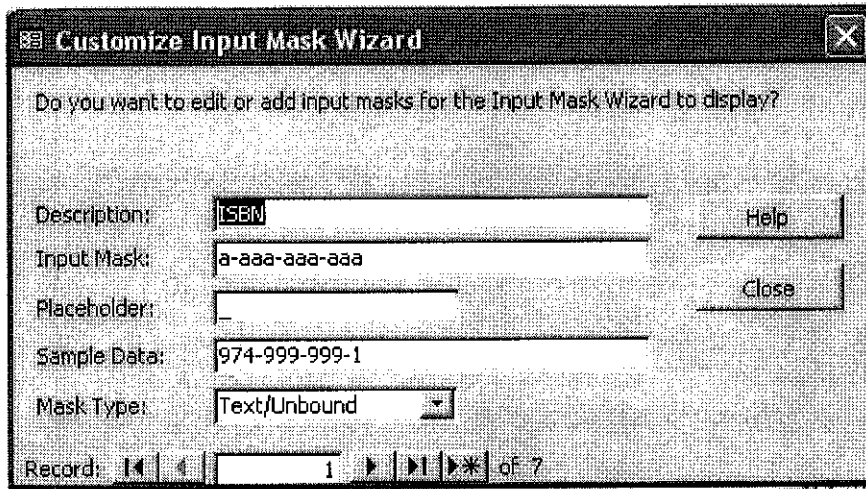
1. คลิกฟิลด์ที่ต้องการกำหนด Input Mask ในตัวอย่างเป็นการกำหนด Input Mask ให้กับฟิลด์หมายเลขโทรศัพท์ (CusTel)
2. คลิกที่ช่อง Input Mask จะปรากฏปุ่ม 



3. คลิกที่ปุ่ม  จะปรากฏหน้าต่าง Input Mask Wizard ดังรูป

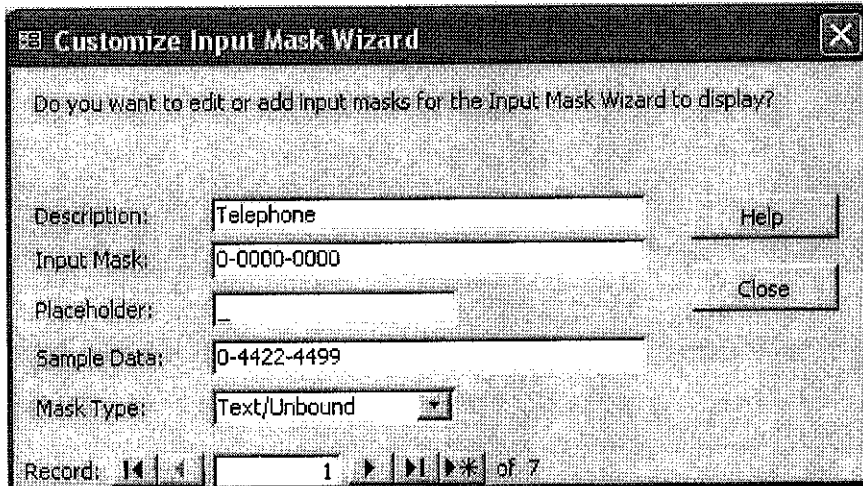


4. คลิกเลือกปุ่ม Edit List จะปรากฏหน้าต่าง Customize Input Mask Wizard ดังรูป



5. แก้ไขข้อมูลใน Customize Input Mask Wizard ดังนี้

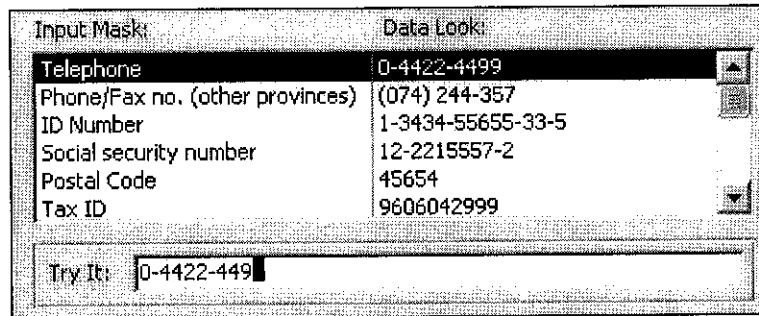
- 1) ช่อง Description: ให้แก้ไขเป็น Telephone
- 2) ช่อง Input Mask: ให้แก้ไขเป็น 0-0000-0000
- 3) ช่อง Sample Data: ให้แก้ไขเป็น 0-4422-4499



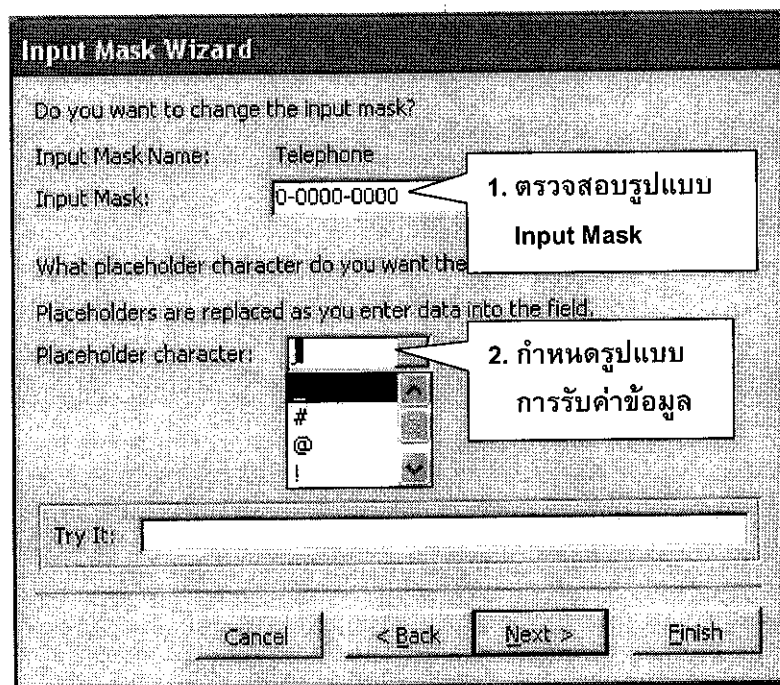
6. เมื่อแก้ไขข้อมูลเสร็จแล้วให้คลิกที่ปุ่มคำสั่ง Close จะปรากฏหน้าต่าง Input Mask Wizard พร้อมรายการ Input Mask ที่สร้างขึ้น ดังรูป

Input Mask:	Data Look:
Telephone	0-4422-4499
Phone/Fax no. (other provinces)	(074) 244-357
ID Number	1-3434-55655-33-5
Social security number	12-2215557-2
Postal Code	45654
Tax ID	9606042999

7. ที่ช่อง Try It: ให้ทดสอบการพิมพ์ค่าข้อมูล เพื่อตรวจสอบว่า Input Mask ที่กำหนดขึ้นตรงความต้องการหรือไม่ ดังรูป

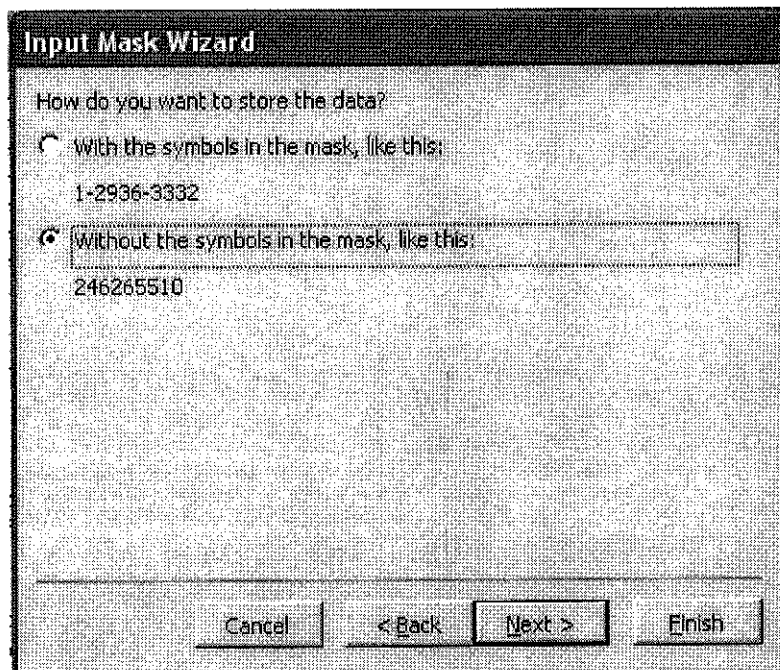


8. คลิกปุ่มคำสั่ง Next จะปรากฏหน้าต่างถัดไป เพื่อกำหนดรูปแบบของสัญลักษณ์ที่ใช้ในการรับค่าข้อมูล ดังรูป

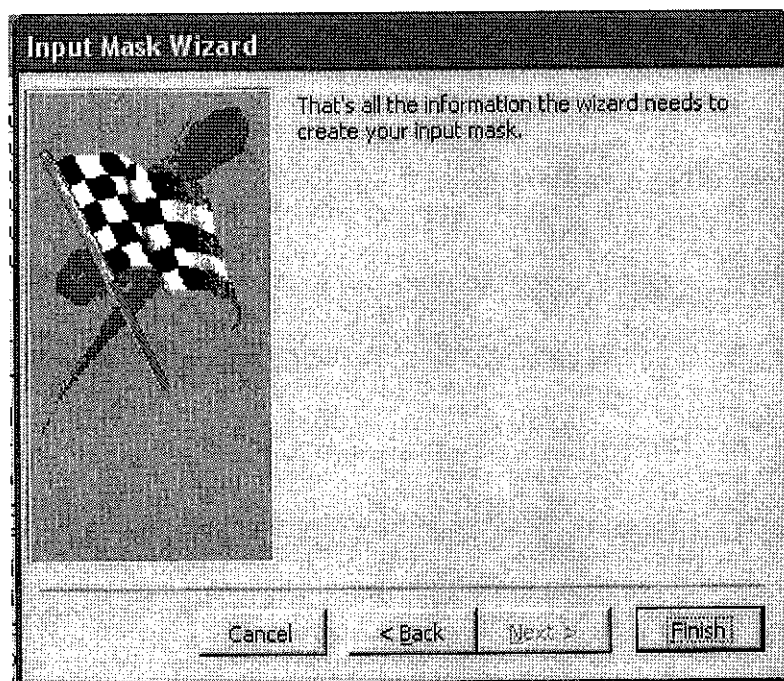


9. คลิกปุ่มคำสั่ง Next จะปรากฏหน้าต่างถัดไป เพื่อกำหนดรูปแบบการจัดเก็บข้อมูลลงตาราง ซึ่งมีด้วยกัน 2 รูปแบบคือ

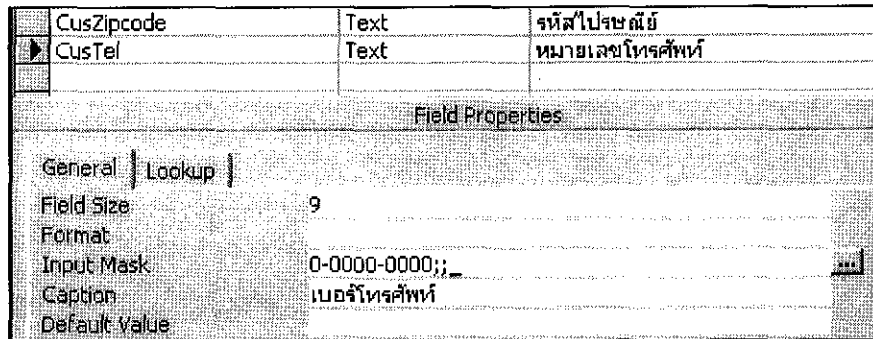
- 1) รูปแบบการจัดเก็บข้อมูลพร้อมสัญลักษณ์ Input Mask
- 2) รูปแบบการจัดเก็บข้อมูลเฉพาะข้อมูลเท่านั้น ไม่รวมสัญลักษณ์ Input Mask



10. คลิกปุ่มคำสั่ง Next ปรากฏข้อความยืนยันการกำหนด Input Mask



11. คลิกปุ่มคำสั่ง Finish จะกลับมายังหน้าต่าง Table1:Table พบว่า Field Properties: Input Mask ปรากฏสัญลักษณ์รูปแบบ Input Mask ตามที่กำหนดไว้ ดังรูป



การกำหนดคีย์หลัก (Primary Key : PK)

เมื่อกำหนดโครงสร้างตารางและองค์ประกอบของฟิลด์ในตารางเสร็จแล้วให้นักศึกษาทำการกำหนดคีย์หลัก (Primary Key : PK) ให้กับตารางที่สร้างขึ้น โดยมีขั้นตอนดังนี้

1. คลิกเลือกฟิลด์ที่ต้องการกำหนดให้เป็นคีย์หลัก ในที่นี้คือฟิลด์ CusID ดังรูป

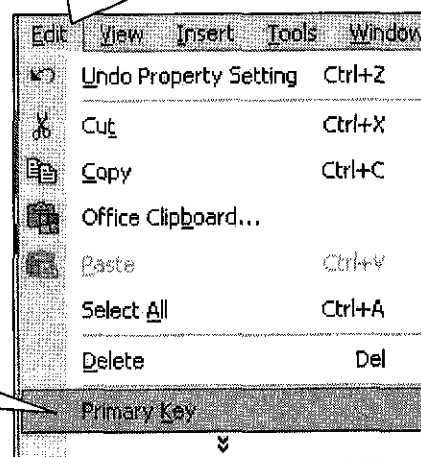
Field Name	Data Type	Description
CusID	Text	รหัสประจำตัวลูกค้าใช้เป็น Primary Key

2. คลิกเลือกคำสั่ง Edit จากแถบเมนูคำสั่ง จากนั้นเลือกคำสั่ง Primary Key

หรือ กดปุ่ม Alt + E จากนั้นกดปุ่ม K

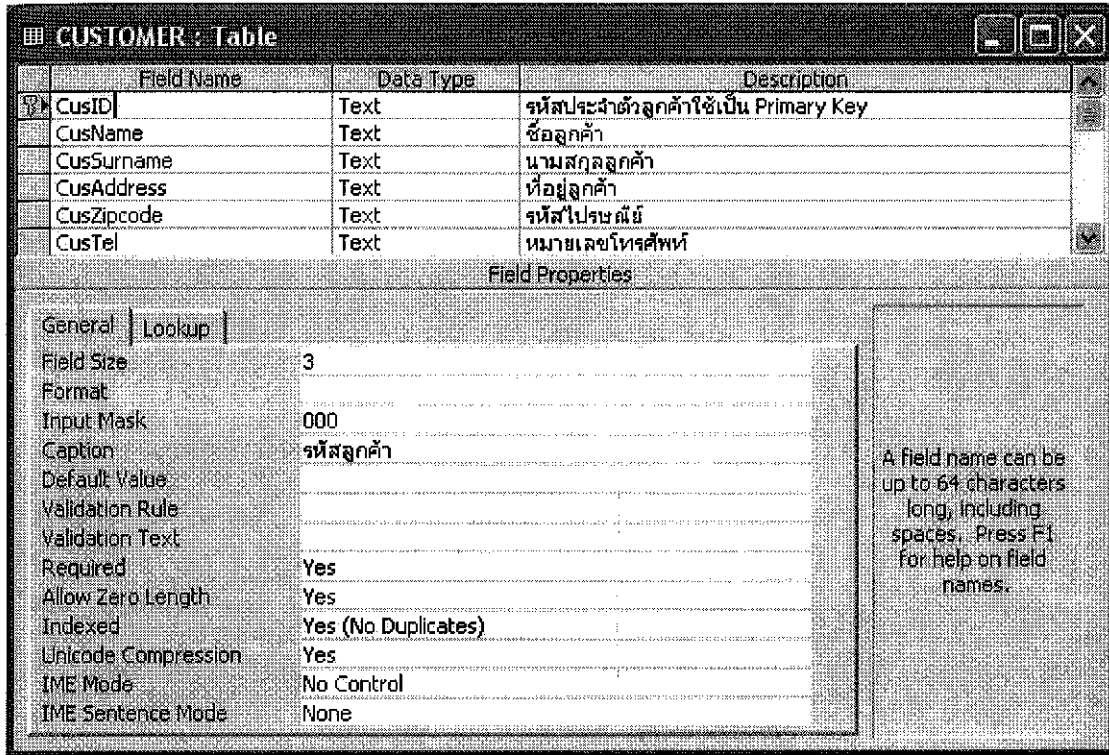
หรือ คลิกที่ไอคอน  บนแถบเครื่องมือ

1. คลิกเมนูคำสั่ง Edit



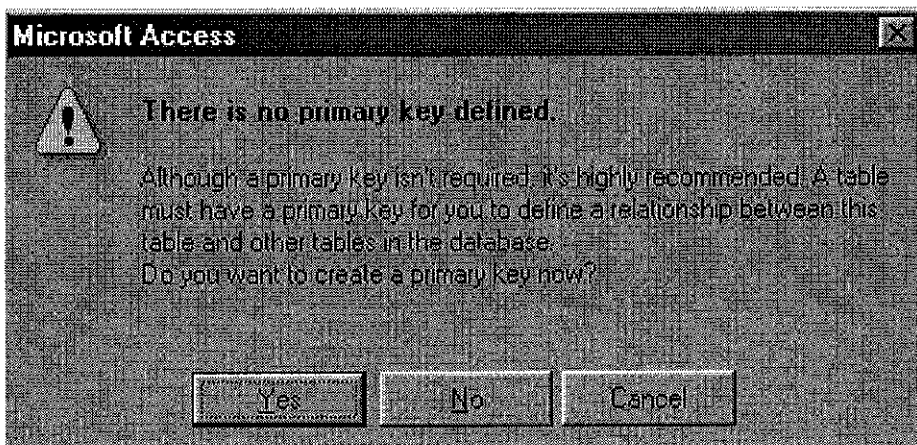
2. คลิกคำสั่ง Primary Key

3. ปรากฏรูปถูกแฉที่ฟิลด์ CusID และที่ Field Properties: Indexed จะเปลี่ยนรูปแบบ
 ดรชชนี้จาก Yes (Duplicates OK) เป็น Yes (No Duplicates) ดังรูป



ข้อควรระวังในการกำหนดคีย์หลัก (Primary Key : PK)

กรณีที่ตารางใดตารางหนึ่งไม่มีการระบุให้มีการกำหนดคีย์หลัก หรือกรณีที่นักศึกษา
 ลืมกำหนดคีย์หลักก่อนการบันทึกตารางที่สร้างขึ้น เมื่อคลิกปุ่มคำสั่ง Save เพื่อบันทึกตาราง
 ดังกล่าวโปรแกรม Microsoft Access 2002 จะปรากฏกล่องข้อความเตือน (Message Box)
 ขึ้นมาถามผู้ใช้งานว่าต้องการกำหนดคีย์หลัก หรือไม่ ดังรูป



กรณีที่คลิกเลือกปุ่มคำสั่ง Yes

- ☞ ให้โปรแกรม Microsoft Access 2002 กำหนดคีย์หลัก (PK) โดยอัตโนมัติกับตารางนั้น โดยโปรแกรมจะสร้างฟิลด์ขึ้นใหม่เพื่อใช้เป็นคีย์หลัก

กรณีที่คลิกเลือกปุ่มคำสั่ง No

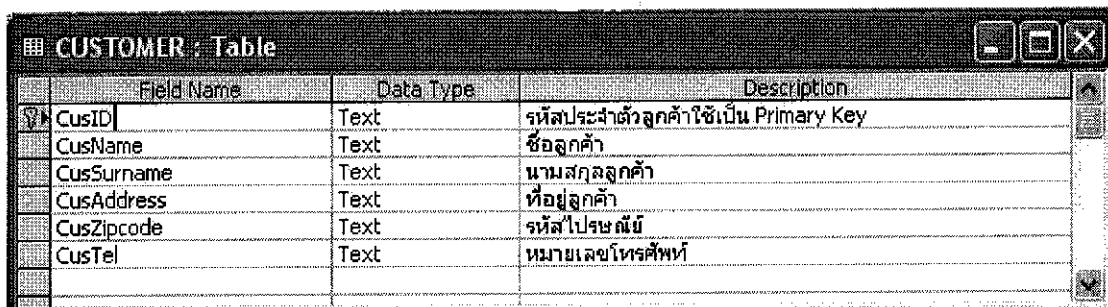
- ☞ ไม่ต้องการกำหนดคีย์หลัก (PK) กับตารางดังกล่าว

กรณีที่คลิกเลือกปุ่มคำสั่ง Cancel

- ☞ ยกเลิกการ Save ตารางดังกล่าว และกลับไปยังหน้าต่างตารางดังกล่าว

การยกเลิกคีย์หลัก (Primary Key : PK)

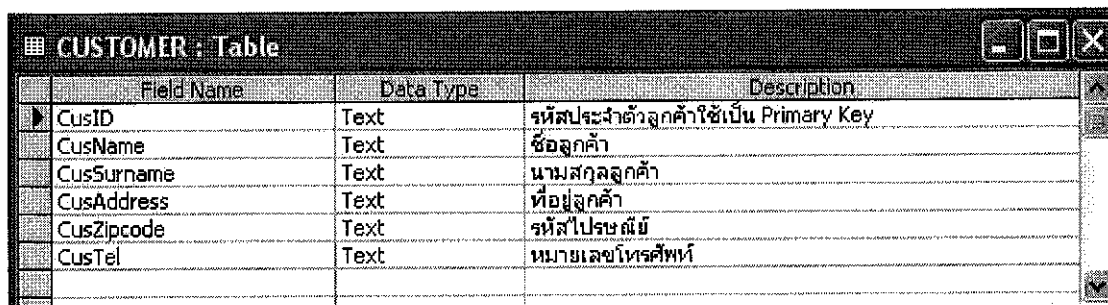
1. คลิกเลือกฟิลด์ที่กำหนดคีย์หลัก และต้องการยกเลิกคีย์หลัก ดังรูป



Field Name	Data Type	Description
CusID	Text	รหัสประจำตัวลูกค้าใช้เป็น Primary Key
CusName	Text	ชื่อลูกค้า
CusSurname	Text	นามสกุลลูกค้า
CusAddress	Text	ที่อยู่ลูกค้า
CusZipcode	Text	รหัสไปรษณีย์
CusTel	Text	หมายเลขโทรศัพท์

2. คลิกเลือกที่ Icon รูป  ที่แถบเครื่องมือ (Toolbar)

3. รูปกุญแจหน้าฟิลด์ CusID ก็จะหายไป ดังรูป



Field Name	Data Type	Description
CusID	Text	รหัสประจำตัวลูกค้าใช้เป็น Primary Key
CusName	Text	ชื่อลูกค้า
CusSurname	Text	นามสกุลลูกค้า
CusAddress	Text	ที่อยู่ลูกค้า
CusZipcode	Text	รหัสไปรษณีย์
CusTel	Text	หมายเลขโทรศัพท์

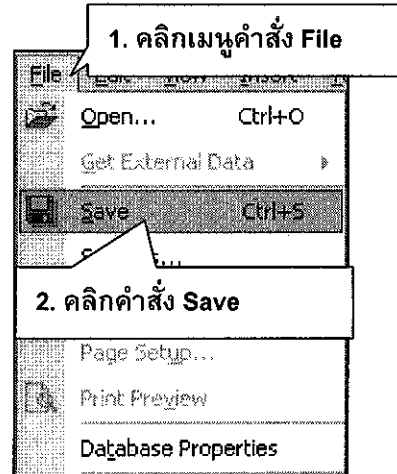
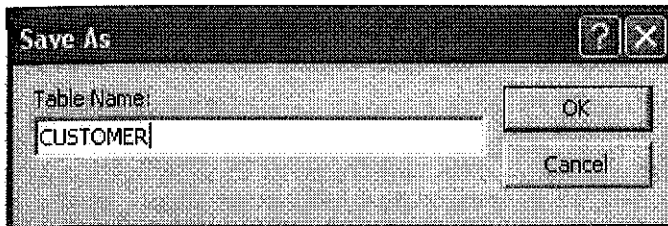
การบันทึกตาราง (Save)

1. คลิกเลือกคำสั่ง File จากแถบเมนูคำสั่ง จากนั้นเลือกคำสั่ง Save

หรือ กดปุ่ม Alt + F จากนั้นกดปุ่ม S

หรือ คลิกที่ไอคอน  บนแถบเครื่องมือ

2. ปรากฏหน้าต่าง Save As ให้ตั้งชื่อตารางว่า CUSTOMER จากนั้นคลิกปุ่มคำสั่ง OK ดังรูป



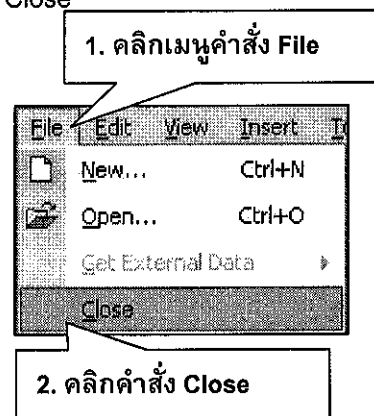
3. ตารางจะเปลี่ยนชื่อ Table1 เป็น CUSTOMER โดยอัตโนมัติ

การปิดหน้าต่างกำหนดโครงสร้างตาราง

1. คลิกเลือกคำสั่ง File จากแถบเมนูคำสั่ง จากนั้นเลือกคำสั่ง Close

หรือ กดปุ่ม Alt + F จากนั้นกดปุ่ม C

หรือ คลิกที่ไอคอน  บนหน้าต่างตาราง CUSTOMER



การแก้ไขโครงสร้างตาราง

กรณีที่ฟิลด์ใดฟิลด์หนึ่งไม่ได้กำหนดคุณสมบัติบางอย่างหรือต้องการลบบางฟิลด์ออกจากตารางให้นักศึกษาดำเนินการดังนี้ (เมื่อมีการแก้ไขต้องมีการ Save ด้วยทุกครั้ง โดยปกติเมื่อสั่งปิดหน้าต่างโครงสร้างตารางเครื่องจะถามว่า Save หรือไม่)

1. เลือก object : Table แล้วเลือกชื่อตารางที่ต้องการแก้ไข โดยคลิกที่ชื่อตารางนั้น แล้วเลือกคำสั่ง Design จะเข้าสู่หน้าต่างกำหนดโครงสร้างตาราง
2. หากต้องการลบฟิลด์ ให้เลือกฟิลด์นั้น แล้วเลือกเมนู **Edit** แล้วเลือกคำสั่ง **Delete**
3. หากต้องการแทรกฟิลด์ ให้เลือกฟิลด์ที่ต้องการให้ฟิลด์ใหม่อยู่ข้างบน แล้วเลือกเมนู **Insert** เลือกคำสั่ง **Rows**
4. หากต้องการย้ายฟิลด์ใดฟิลด์หนึ่งไปยังตำแหน่งอื่น ๆ ให้เลือกฟิลด์นั้นแล้วเลือกเมนู **Edit** เลือกคำสั่ง **cut** และไปยังตำแหน่งที่ต้องการให้ฟิลด์เข้าอยู่ แล้วเลือกเมนู **Edit** เลือกคำสั่ง **Past**
5. หากต้องการคัดลอกฟิลด์ ให้ทำตามข้อ 2 แต่เปลี่ยนคำสั่ง **cut** เป็น **copy** เท่านั้น

การออกจากโปรแกรม

1. คลิกเลือกคำสั่ง File จากแถบเมนูคำสั่ง
 2. จากนั้นเลือกคำสั่ง Exit
- หรือ กดปุ่ม Alt + F จากนั้นกดปุ่ม X

หรือ คลิกที่ไอคอน



บนหน้าต่างโปรแกรม Microsoft Access 2002

1. คลิกเมนูคำสั่ง File

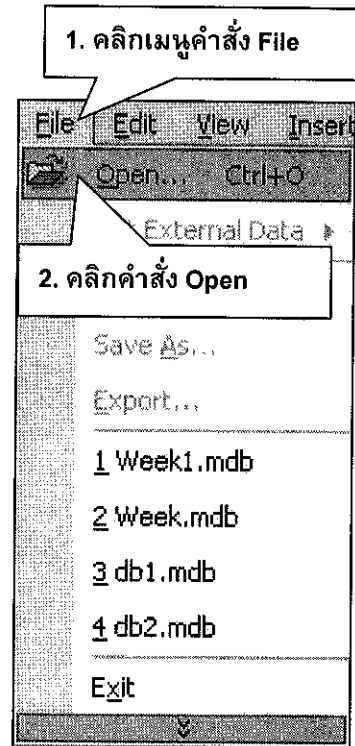


2. คลิกคำสั่ง Exit

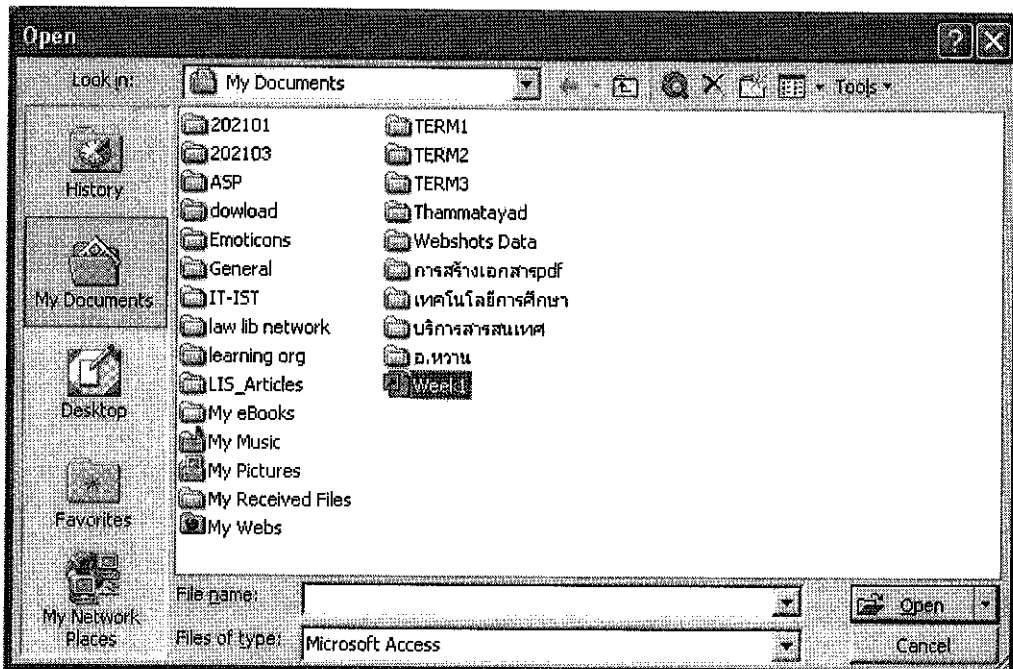
การเปิดฐานข้อมูลเก่าขึ้นมาใช้งาน

ในการเปิดฐานข้อมูลเก่าที่เคยสร้างขึ้นให้ปฏิบัติดังนี้

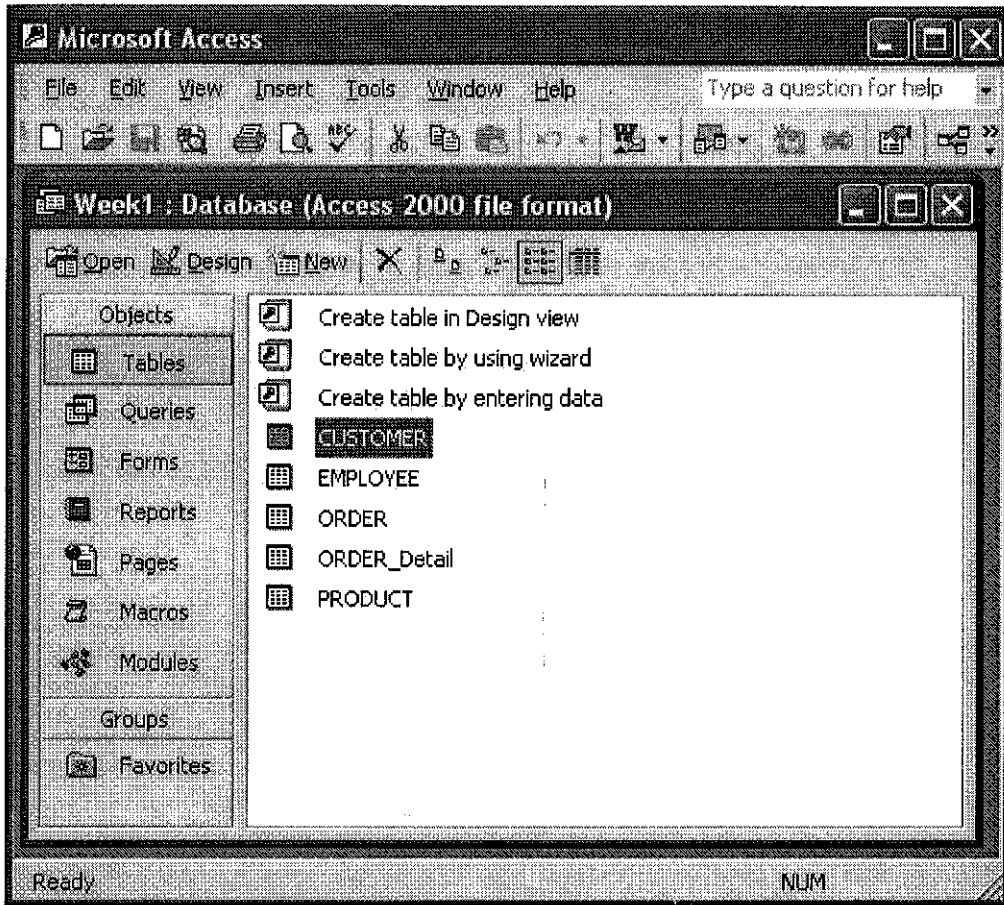
1. เปิดโปรแกรม Microsoft Access 2002
2. คลิกเลือกคำสั่ง File จากแถบเมนูคำสั่ง
จากนั้นเลือกคำสั่ง Open
หรือ กดปุ่ม Alt + F จากนั้นกดปุ่ม O
หรือ กดปุ่ม Ctrl + O



3. ปรากฏหน้าต่าง Open ดังรูป



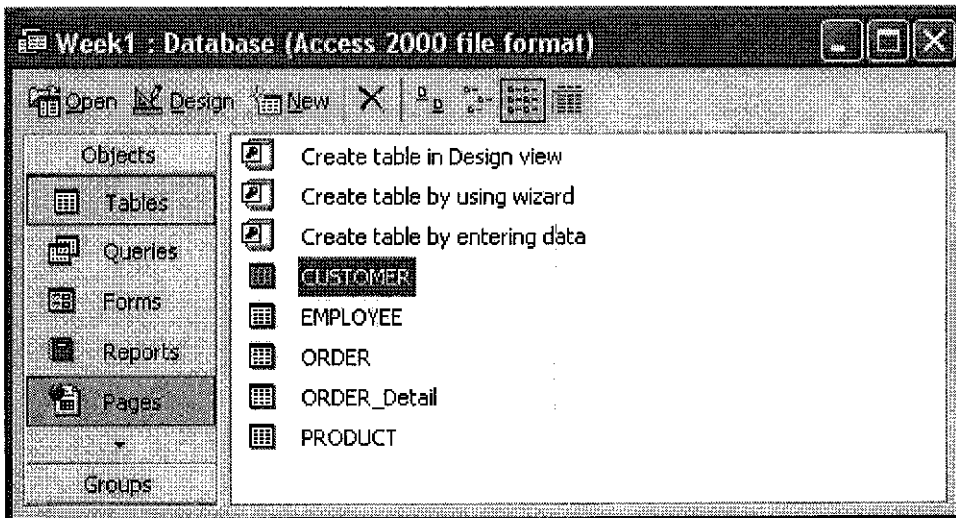
4. คลิกเลือกแฟ้มข้อมูลที่ต้องการเปิด ในที่นี้ให้เลือก Week1.mdb
5. คลิกปุ่ม Open จะปรากฏหน้าต่าง Database : Week1 ขึ้นมา ดังรูป



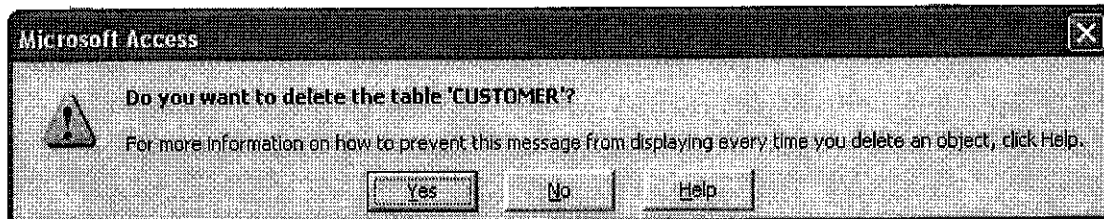
การลบตาราง

จากหน้าต่างฐานข้อมูล หากผู้ใช้ไม่ต้องการใช้งานตารางใดตารางหนึ่ง ผู้ใช้สามารถลบตารางดังกล่าวได้ โดยดำเนินการดังนี้

1. คลิกเลือกตารางที่ต้องการลบ เช่น กรณีที่ต้องการลบตารางชื่อ CUSTOMER ให้คลิกเมาส์ที่ชื่อตาราง CUSTOMER



2. เลือกเมนูคำสั่ง Edit จากแถบเมนูคำสั่ง จากนั้นคลิกเลือกคำสั่ง Delete หรือ กดปุ่ม Delete บนแป้นพิมพ์ (Keyboard)
3. ปรากฏหน้าต่างแสดงข้อความสอบถามผู้ใช้ว่าต้องการลบตาราง CUSTOMER หรือไม่
 - คลิกที่ปุ่ม YES กรณีที่ต้องการลบตาราง CUSTOMER
 - คลิกที่ปุ่ม NO กรณีที่ยกเลิกการลบตาราง CUSTOMER
 - คลิกที่ปุ่ม HELP กรณีที่ต้องการคำแนะนำอื่นๆ เพิ่มเติมจากโปรแกรม



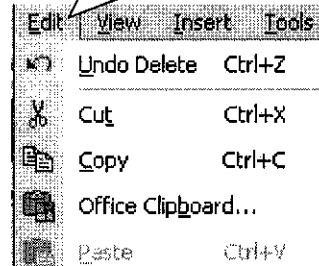
การเปลี่ยนชื่อตาราง

กรณีนี้ผู้ใช้ต้องการเปลี่ยนแปลงชื่อตารางสามารถทำได้ดังนี้

1. จากหน้าต่าง Database : Week1 คลิกเลือกตารางที่ต้องการเปลี่ยนชื่อ
2. เลือกเมนูคำสั่ง Edit จากแถบเมนูคำสั่ง
จากนั้นคลิกเลือกคำสั่ง Rename
หรือ คลิกเมาส์ด้านขวา เลือกคำสั่ง Rename

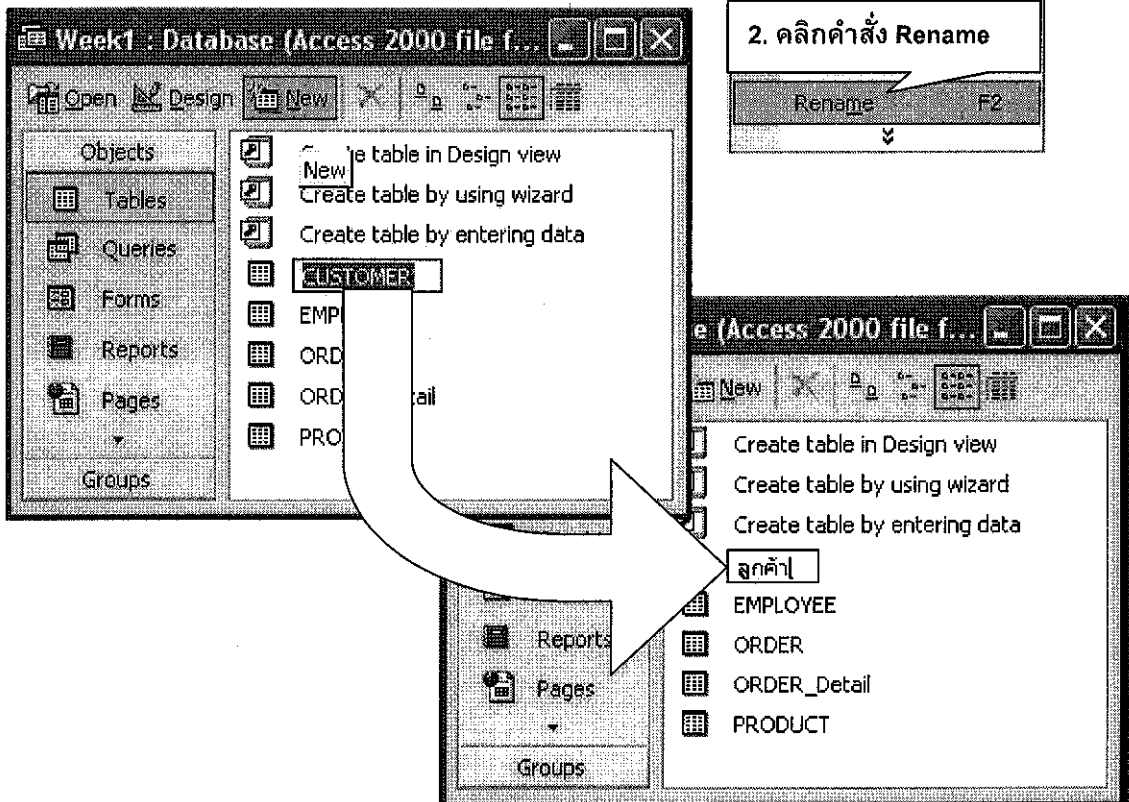
3. พิมพ์ชื่อใหม่ลงแทนชื่อเดิม แล้วกดปุ่ม Enter ดังรูป

1. คลิกเมนูคำสั่ง Edit



2. คลิกคำสั่ง Rename

Rename F2



ภาคผนวก

Table : CUSTOMER

CusID = Primary key

Field Name	Data Type	Description	Field Size	Format	Decima I Place	Input Mask	Caption	Default Value	Validation Value	Validation Text	Required	Indexed
CusID	Text	ใช้เป็น PK	3			000	รหัสลูกค้า				Yes	Yes (NO Duplicate)
CusName	Text	ชื่อลูกค้า	20				ชื่อ				Yes	
CusSurname	Text	นามสกุลลูกค้า	20				นามสกุล				Yes	
CusAddress	Text	ที่อยู่ลูกค้า	50				ที่อยู่				Yes	
CusZipcode	Text	รหัสไปรษณีย์	5			00000	รหัสไปรษณีย์	30000				
CusTel	Text	เบอร์โทรศัพท์ลูกค้า	13			9-9999-9999	เบอร์โทรศัพท์					

Table : PRODUCT

ProID = Primary key

Field Name	Data Type	Description	Field Size	Format	Decima I Place	Input Mask	Caption	Default Value	Validation Value	Validation Text	Required	Indexed
ProID	Text	ใช้เป็น PK	3			000	รหัสสินค้า				Yes	Yes (NO Duplicate)
ProName	Text	ชื่อสินค้า	20				ชื่อสินค้า				Yes	
Price	Currency	ราคา		Standard	Auto		ราคาต่อหน่วย	0	< = 5000	คุณป้อน ข้อมูลผิด	Yes	
ProDes	Memo	รายละเอียด					รายละเอียด สินค้า					

Table : EMPLOYEE

EmID = Primary key

Field Name	Data Type	Description	Field Size	Format	Decima l Place	Input Mask	Caption	Default Value	Validation Value	Validation Text	Required	Indexed
EmID	Text	ใช้เป็น PK	3			000	รหัสพนักงาน				Yes	Yes (NO Duplicate)
EmName	Text	ชื่อ	20				ชื่อ				Yes	
EmSurname	Text	นามสกุล	20				นามสกุล				Yes	
EmPosition	Text	ตำแหน่ง	20				ตำแหน่ง	พนักงาน ชาย			Yes	
EmAddress	Text	ที่อยู่	50				ที่อยู่					
EmZipcode	Text	รหัสไปรษณีย์	5			00000	รหัสไปรษณีย์					
EmTel	Text	เบอร์โทรศัพท์	13			9-9999-9999	เบอร์โทรศัพท์					
StartingDate	Date/Time	วันที่เข้าทำงาน		General Date			วันที่บรรจุ				Yes	
EmSalary	Currency	เงินเดือน		General Number	2		เงินเดือน	0	Between 8000 and 30000	คูณป้อน ข้อมูลผิด	Yes	

Table : ORDER

OrderID = Primary key

Field Name	Data Type	Description	Field Size	Format	Decima l Place	Input Mask	Caption	Default Value	Validation Value	Validation Text	Required	Indexed
OrderID	Text	ใช้เป็น PK	3			000	รหัสการสั่ง				Yes	Yes (NO Duplicate)
CusID	Text	รหัสลูกค้า	3			000	รหัสลูกค้า				Yes	
OrderDate	Dat/Time	วันที่สั่งซื้อ		General Date			วันสั่ง				Yes	
ShippedDate	Dat/Time	วันที่ส่ง		General Date			วันสั่ง				Yes	
EmID	Text	รหัสพนักงาน	3			000	รหัสพนักงาน				Yes	

Table : ORDER-detail

ไม่มีฟิลด์ใดเป็น Primary key

Field Name	Data Type	Description	Field Size	Format	Decimal Place	Input Mask	Caption	Default Value	Validation Value	Validation Text	Required	Indexed
OrderID	Text	รหัสการสั่ง	3			000	รหัสการสั่ง				Yes	Yes (Duplicate OK)
ProID	Text	รหัสสินค้า	3			000	รหัสสินค้า				Yes	Yes (Duplicate OK)
ProCount	Number	จำนวน	Integer		0		จำนวน	0			Yes	
Discount	Number	ส่วนลด	Double	Percent	0		ส่วนลด	0.05			Yes	

ปฏิบัติการที่ 2

การเชื่อมความสัมพันธ์ระหว่างตาราง

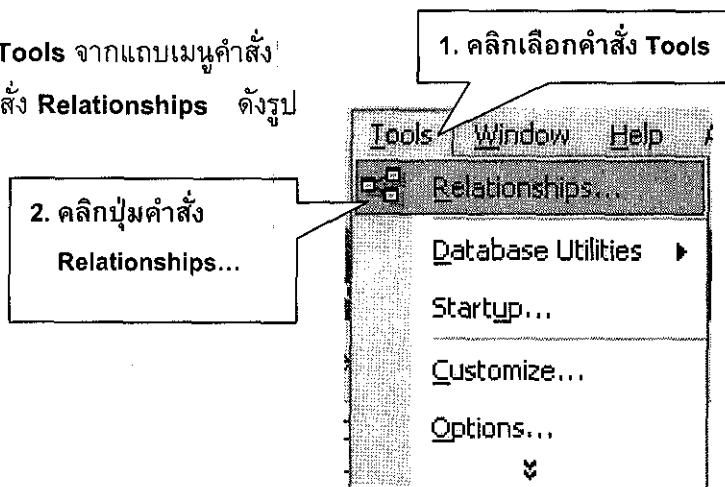
วัตถุประสงค์ : นักศึกษาสามารถ

1. เชื่อมความสัมพันธ์ระหว่างตารางได้
2. กรอกข้อมูลลงใน Datasheet ได้

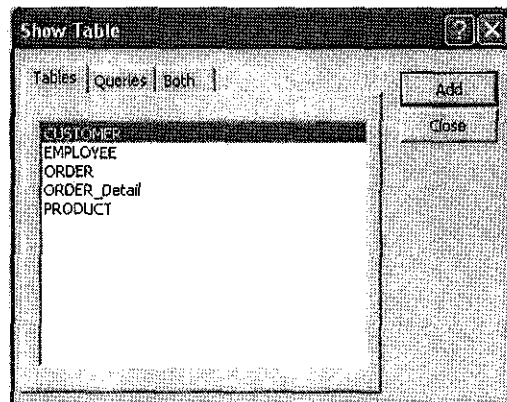
ขั้นตอนปฏิบัติที่ 2.1

การเชื่อมความสัมพันธ์ระหว่างตาราง

1. คลิกเลือกเมนูคำสั่ง **Tools** จากแถบเมนูคำสั่ง!
จากนั้นคลิกเลือกคำสั่ง **Relationships** ดังรูป



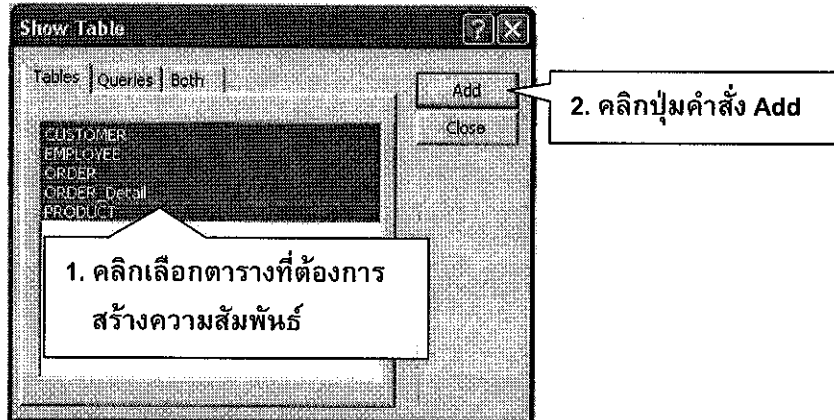
2. ปรากฏหน้าต่าง Show Table ดังรูป
ผู้ใช้สามารถกำหนดการเชื่อม
ความสัมพันธ์ของ Table, Queries
หรือ Both (ทั้ง Table และ Queries
ร่วมกัน)



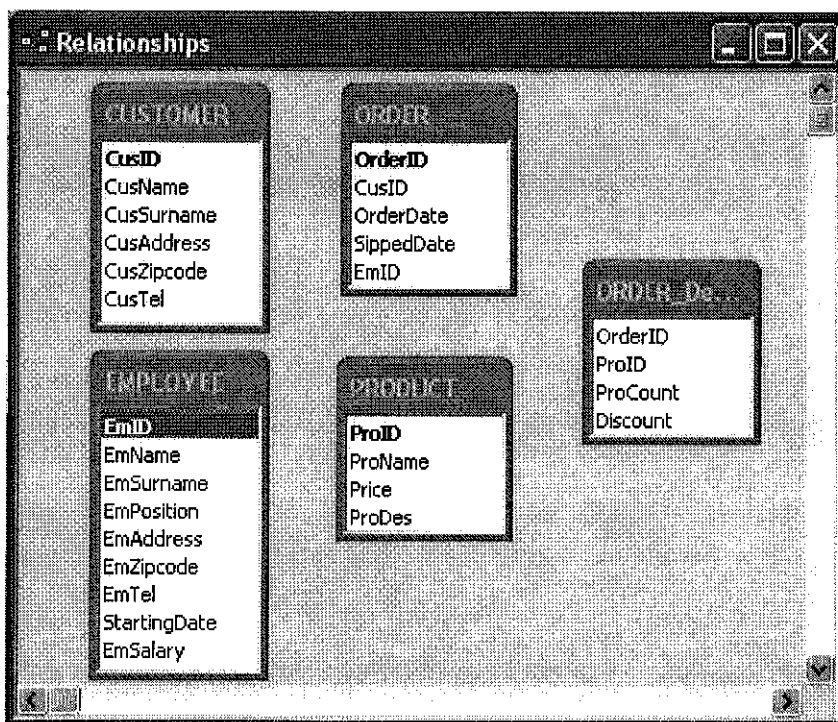
3. เลือกว่าต้องการเชื่อมความสัมพันธ์จากข้อมูลใน Table, Queries หรือทั้งสองอย่าง เช่น
กรณีที่1 หากต้องการเชื่อมความสัมพันธ์ระหว่างตาราง (Table) ด้วยกัน
ให้คลิกเลือกที่แถบ Tables
กรณีที่2 หากต้องการเชื่อมความสัมพันธ์ระหว่างคิวรี (Queries) ด้วยกัน
ให้คลิกเลือกที่แถบ Queries
กรณีที่3 หากต้องการเชื่อมความสัมพันธ์ทั้งตาราง (Table) และคิวรี (Queries)
ด้วยกันให้คลิกเลือกที่แถบ Both

ในที่นี้ให้คลิกเลือกที่แถบ Table เพราะจะทำการเชื่อมความสัมพันธ์ระหว่างทุกตาราง (CUSTOMER, PRODUCT, EMPLOYEE, ORDER และ ORDER_Detail) เข้าด้วยกัน

4. คลิกเลือกชื่อตารางที่ต้องการเชื่อมความสัมพันธ์ จากนั้นคลิกที่ปุ่มคำสั่ง Add ดังรูป



5. ปรากฏหน้าต่าง Relationships ภายในจะแสดงชื่อตาราง (Table) ที่เลือกเอาไว้เพื่อใช้ในการสร้างความสัมพันธ์ ดังรูป



6. คลิกปุ่มคำสั่ง Close บนหน้าต่าง Show Table

การลบตารางที่ไม่ต้องการออก

คลิกเมาส์ที่ตารางนั้น เลือกเมนู Edit เลือกคำสั่ง Delete หรืออาจคลิกเมาส์ที่ตารางนั้น แล้วกดเป็นพิมพ์ Delete

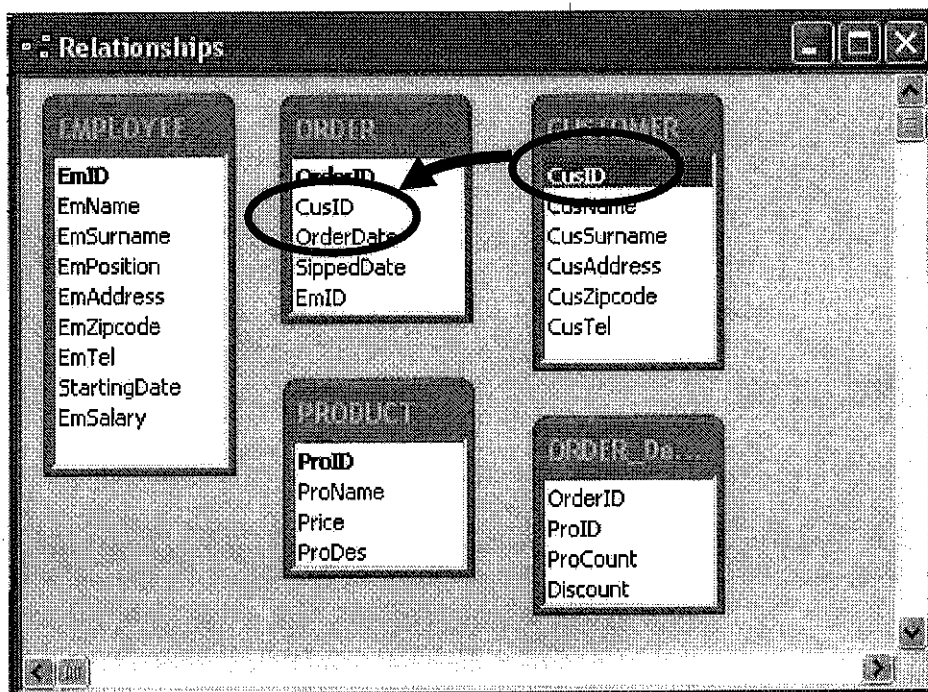
การเพิ่มตาราง

หากต้องการเลือกบางตารางเพิ่มให้ Add Table ใหม่ โดยเลือกเมนู Relationship และคลิกที่คำสั่ง Show Table แล้วดำเนินการเลือกตารางใหม่

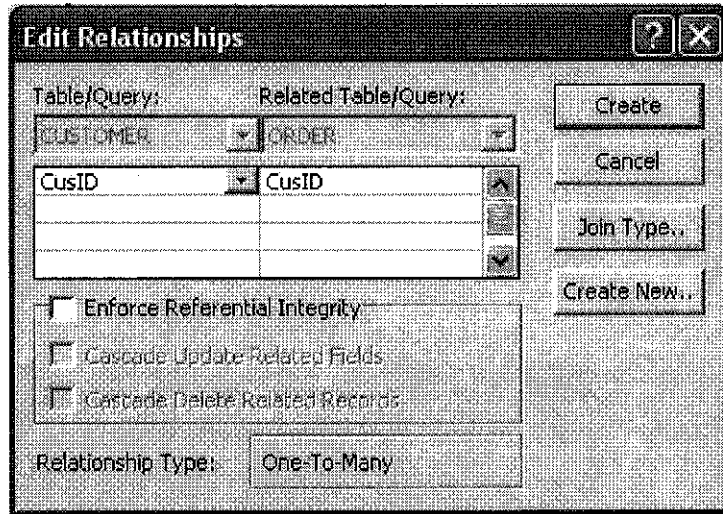
7. การเชื่อมความสัมพันธ์ระหว่างฟิลด์ต่างๆ เช่น กรณีที่ต้องการเชื่อมความสัมพันธ์ระหว่างฟิลด์ CusID ในตาราง CUSTOMER กับฟิลด์ CusID ในตาราง ORDER

ขั้นตอน

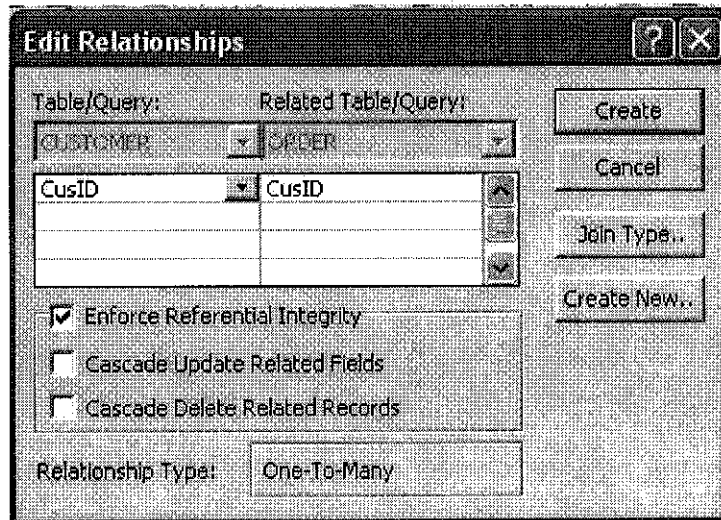
1. คลิกเลือกฟิลด์ CusID ในตาราง CUSTOMER
2. คลิกเมาส์ค้างเอาไว้แล้วลากเมาส์ไปที่บนชื่อฟิลด์ CusID ในตาราง ORDER แล้วปล่อยเมาส์



3. ปรากฏหน้าต่าง Edit Relationships ดังรูป



4. คลิกเลือกข้อความ Enforce Referential Integrity ดังรูป



การกำหนด Enforce Referential Integrity มี 2 แบบ

☛ Cascade Update Related Fields

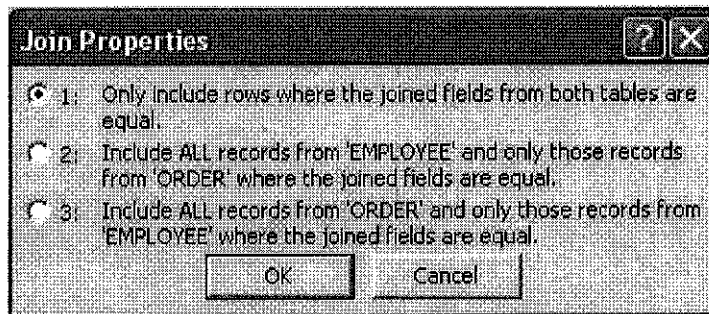
เป็นการกำหนดว่า เมื่อค่าของฟิลด์ที่เชื่อมกันในตารางหลักมีการเปลี่ยนแปลงให้เปลี่ยนแปลงที่ตารางอีกฝั่งด้วย หรือไม่

☛ Cascade Delete Related Records

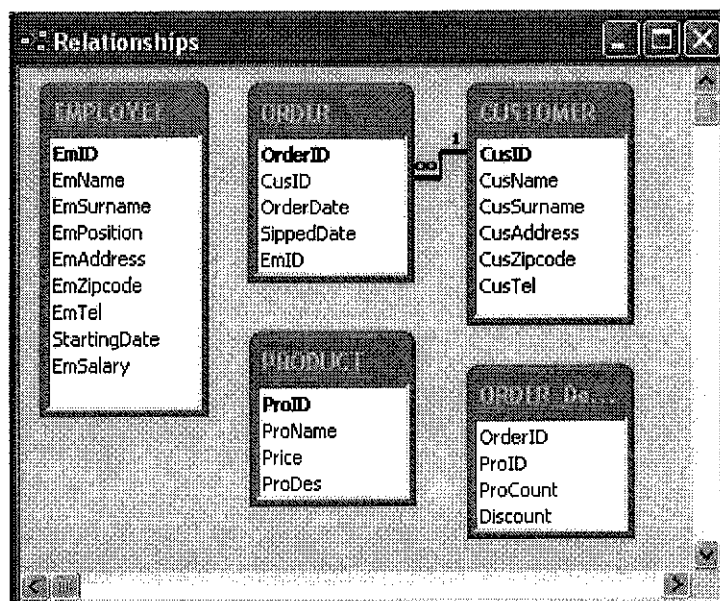
เป็นการกำหนดว่า เมื่อเรคคอร์ดที่เชื่อมกันในตารางหลักถูกลบ ก็จะลบที่ตารางอีกฝั่งด้วยหรือไม่

การกำหนด Join Type

- ตัวเลือกที่ 1 เชื่อมเฉพาะเรคอร์ดที่มีค่าของฟิลด์ตรงกันเท่านั้น
- ตัวเลือกที่ 2 เชื่อมแบบนำเรคอร์ดของตารางด้าน One ทั้งหมดมา และนำเฉพาะเรคอร์ดของตารางฝั่ง many ที่มีค่าของฟิลด์ตรงกันกับตารางฝั่ง One เท่านั้น
- ตัวเลือกที่ 3 เชื่อมแบบนำเรคอร์ดของตารางด้าน many ทั้งหมดมา และนำเฉพาะเรคอร์ดของตารางฝั่ง One ที่มีค่าของฟิลด์ตรงกันกับตารางฝั่ง many เท่านั้น



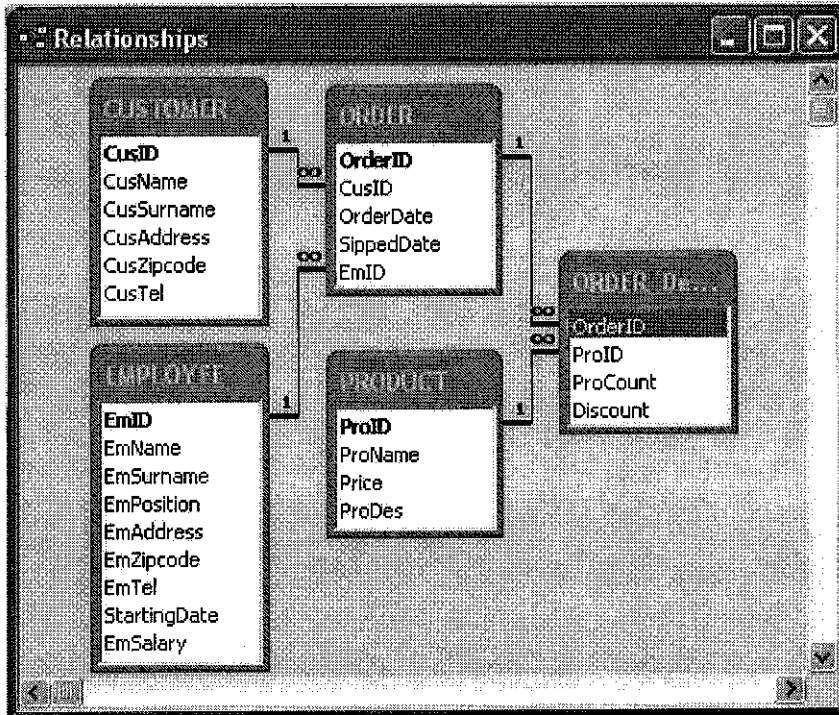
5. คลิกที่ปุ่ม Create เพื่อสร้างความสัมพันธ์ระหว่างฟิลด์ทั้งสองที่เลือกไว้
6. ที่หน้าต่าง Relationships จะปรากฏเส้นเชื่อมความสัมพันธ์ระหว่างฟิลด์ CusID ตาราง CUSTOMER กับฟิลด์ CusID ตาราง ORDER พร้อมรูปแบบความสัมพันธ์แบบ One to Many (1 : M) ดังรูป



8. ให้นักศึกษาเชื่อมความสัมพันธ์ดังนี้

- ฟیلด์ EmID ในตาราง EMPLOYEE กับฟیلด์ EmID ในตาราง ORDER
- ฟیلด์ ProID ในตาราง PRODUCT กับฟیلด์ ProID ในตาราง ORDER-DETAIL
- ฟیلด์ OrderID ในตาราง ORDER กับฟیلด์ OrderID ในตาราง ORDER-DAIL

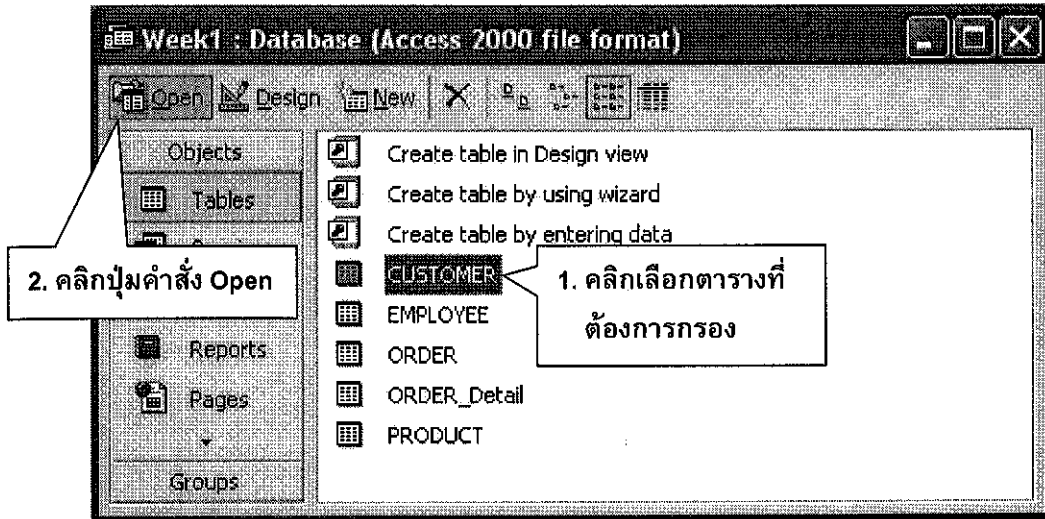
9. เมื่อเชื่อมความสัมพันธ์ครบทุกข้อแล้วจะปรากฏดังรูป



10. ทำการบันทึก Relationships ที่สร้างขึ้น

การกรอกข้อมูลใน Datasheet (ตาราง)

1. เรียก Database ที่ชื่อ Week1.mdb ขึ้นมาเพื่อกรอกข้อมูลลงใน Datasheet ของตาราง
2. จากหน้าจอ Database ให้เลือกตาราง "Customer" แล้วคลิกที่ปุ่ม Open



3. ปรากฏหน้าต่าง Table : CUSTOMER ดังรูป

รหัสลูกค้า	ชื่อ	นามสกุล	ที่อยู่	รหัสไปรษณีย์	เบอร์โทรศัพท์
				30000	

Record: 1 of 1

4. พิมพ์ข้อความลงในช่องฟิลด์ต่างๆ ดังนี้

1. ฟิลด์รหัสลูกค้า ให้พิมพ์ข้อความ "001"

รหัสลูกค้า	ชื่อ	นามสกุล	ที่อยู่	รหัสไปรษณีย์	เบอร์โทรศัพท์
001				30000	
*				30000	

Record: 1 of 1

2. กดปุ่ม Tab เพื่อเลื่อนไปยังฟิลด์ถัดไป จนใส่ข้อมูลครบทุกตารางดังนี้

รหัสลูกค้า	ชื่อ	นามสกุล	ที่อยู่	รหัสไปรษณีย์	เบอร์โทรศัพท์
001	เงิน	ดี	213 อ.เมือง จ.นครราชสีมา	30000	0-4422-2222
*				30000	

Record: 1 of 1

Table: CUSTOMER

รหัสลูกค้า	ชื่อ	นามสกุล	ที่อยู่	รหัสไปรษณีย์	หมายเลขโทรศัพท์
001	เงิน ดี		213 อ.เมือง จ.นครราชสีมา	30000	0-4422-2222
002	ทองแดง		111 อ.เมือง จ.ขอนแก่น	40000	0-4333-2222
003	นาถ งาม		9 พญาไท กรุงเทพฯ	10300	0-1001-1009
004	เพชร สวย		99 อ.เมือง จ.นครราชสีมา	30000	0-4422-0001

Table: PRODUCT

รหัสสินค้า	ชื่อสินค้า	ราคาสินค้าต่อหน่วย	รายละเอียดสินค้า
100	กางเกง	600.00	กางเกงขาสั้น
200	เสื้อยืด	200.00	ใส่แล้วดูดี
300	เข็มขัด	300.00	หนังกบ
400	รองเท้า	900.00	รองเท้าหนังแกะ

Table: EMPLOYEE

รหัสพนักงาน	ชื่อ	นามสกุล	ตำแหน่ง	ที่อยู่	รหัสไปรษณีย์	หมายเลขโทรศัพท์	StartDate	เงินเดือน
111	เก่ง วย		พนักงานขาย	2 อ.เมือง จ.นครราชสีมา	30000	0-4422-3333	11/01/1998	20000
222	เลิศ หู		พนักงานขาย	11 อ.เมือง จ.ขอนแก่น	40000	0-4333-5555	11/12/1998	10000
333	ดี งาม		พนักงานส่งเสริม	19 พญาไท กรุงเทพฯ	10300	0-1111-2222	12/05/1999	8000

Table: ORDER

รหัสการสั่งซื้อ	รหัสลูกค้า	วันสั่ง	วันส่ง	รหัสพนักงาน
001	003	12/12/1998	18/12/1998	111
002	004	15/12/1998	20/12/1998	222
003	003	30/12/1998	05/01/1999	111

Table: Order_Detail

รหัสการสั่งซื้อ	รหัสสินค้า	จำนวน	ส่วนลด
001	100	20	5%
002	100	15	5%
003	300	5	5%
003	400	30	5%
001	200	30	10%
002	200	20	5%

ปฏิบัติการที่ 3

การสร้าง Query

วัตถุประสงค์ นักศึกษาสามารถ

1. เข้าใจหลักการในการทำ Query
2. สร้าง Query เพื่อการใช้งานฐานข้อมูลได้สะดวกยิ่งขึ้น

Query : ประเภทและมุมมองในการสร้าง Query

คิวรี (Query) เป็นคุณสมบัติพิเศษของ Microsoft Access ที่ทำหน้าที่ค้นหาข้อมูลตามความต้องการของผู้ใช้งาน นอกจากนี้คิวรียังใช้ในการจัดการกับข้อมูลในฐานข้อมูล ซึ่งมีความสามารถต่างๆ ได้แก่ การสร้างตารางข้อมูลขึ้นใหม่ โดยการนำข้อมูลจากตารางที่มีอยู่แล้วมาจัดการสร้างเป็นตารางใหม่, การปรับปรุงข้อมูลในตาราง และใช้ฟังก์ชันต่างๆ ในการจัดการข้อมูล

ประเภทของ Query

Query สามารถแบ่งประเภทตามลักษณะการทำงานได้ดังนี้

ประเภท	ความหมาย	
Select Query	Query ที่ใช้ในการเลือกข้อมูลจากตาราง โดยแสดงตามเงื่อนไขที่กำหนด โดยข้อมูลที่นำมาแสดงอาจมาจาก 1 ตาราง หรือหลายตารางที่มีความสัมพันธ์กัน	
Action Query	Query ที่ใช้กระทำกรอย่างใดอย่างหนึ่งกับข้อมูลหรือตาราง ซึ่งจำแนกตามความสามารถได้ดังนี้	
	<input type="checkbox"/> Make-Table Query	สร้างตารางขึ้นใหม่โดยอัตโนมัติ โดยโครงสร้างและข้อมูลของตารางได้จากข้อมูลที่ Query เลือกไว้ หรือได้จากการคำนวณโดย Query
	<input type="checkbox"/> Append Query	เพิ่มข้อมูลลงในตารางโดยนำข้อมูลที่ Query เลือกไว้ไปใส่ต่อท้ายข้อมูลที่มีอยู่ในตารางที่กำหนด
	<input type="checkbox"/> Update Query	แก้ไข-ปรับปรุงข้อมูลจำนวนมากๆ ในตาราง ในครั้งเดียว
<input type="checkbox"/> Delete Query	ลบข้อมูลทั้งหมดออกจากตารางที่กำหนด	
Crosstab Query	Query ที่นำข้อมูลแถวไปสร้างเป็นคอลัมน์ และนำข้อมูลคอลัมน์ไปสร้างเป็นแถว เพื่อใช้ในการวิเคราะห์ข้อมูล	
Union Query	Query ที่นำข้อมูลจากหลายๆ ตารางมารวมกันเป็นตารางเดียว ซึ่ง Query ประเภทนี้จะต้องสร้างด้วยคำสั่งภาษา SQL	
Parameter Query	Query ที่มีการเรียกให้ผู้ใช้กรอกข้อมูลที่เป็นเงื่อนไขในการเลือกข้อมูล หรือใส่ค่าตัวแปรในการคิดคำนวณ	

มุมมองในการสร้าง Query

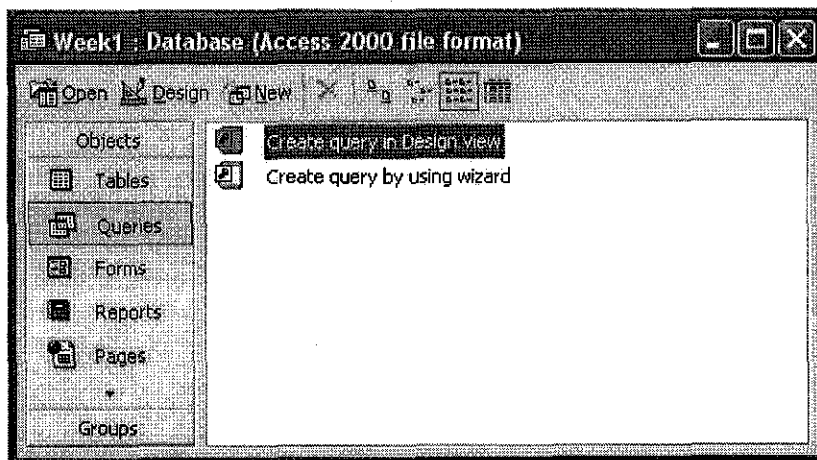
Microsoft Access 2002 กำหนดมุมมองในการสร้าง Query 3 มุมมอง ดังนี้

มุมมอง	คำอธิบาย
Design View	มุมมองที่ใช้สำหรับออกแบบและสร้าง Query ซึ่งประกอบด้วยเงื่อนไขและกฎเกณฑ์ต่างๆ ที่ต้องนำมาใช้ในการค้นหาหรือแสดงข้อมูลตามที่ต้องการ
Datasheet View	มุมมองที่ใช้สำหรับแสดงข้อมูลที่ได้จากการสืบค้น ตามเงื่อนไขที่กำหนดไว้ใน Query (Design View) โดยแสดงในลักษณะเช่นเดียวกับตาราง (Table)
SQL View	มุมมองที่ใช้สำหรับการออกแบบหรือสร้างคำสั่ง Query ด้วยภาษา SQL ซึ่งถือเป็นภาษาที่ใช้ในการจัดการกับข้อมูลที่มีอยู่ในฐานข้อมูล

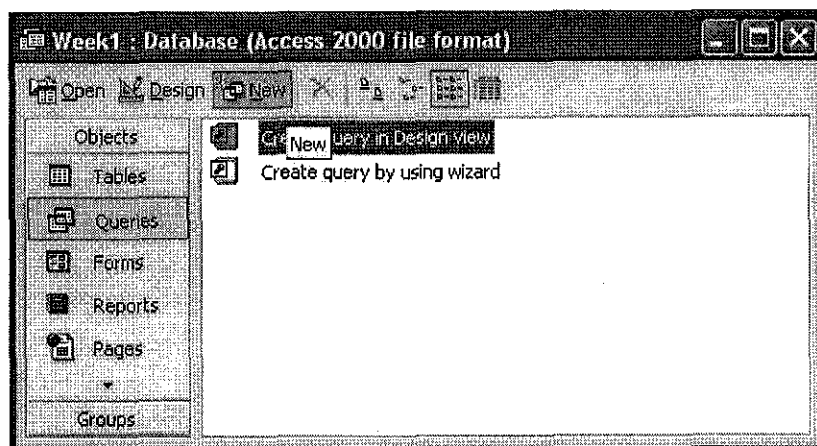
ขั้นตอนการปฏิบัติที่ 3.1

การสร้าง Query จากตารางเดียว

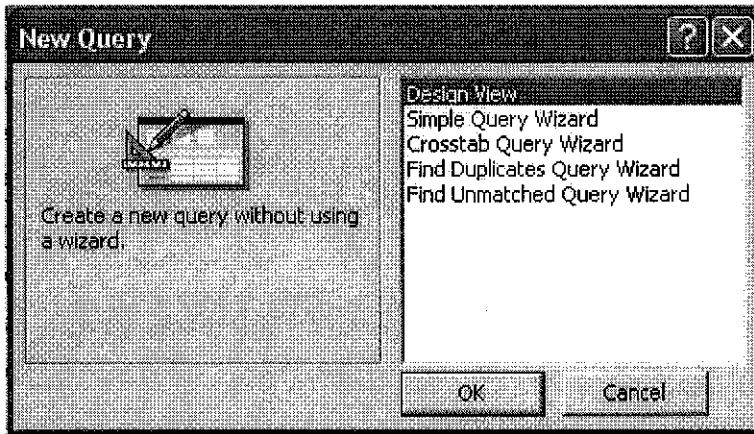
1. ให้นักศึกษาเรียกฐานข้อมูล **Week1.mdb** ขึ้นมาใช้งาน
2. จากหน้าต่าง **Week1:Database** ให้คลิกเลือก **Object** ชื่อ **Query**
3. หน้าต่าง Database จะเปลี่ยนเป็นหน้าต่างการทำงานของ Query ดังรูป



4. คลิกที่คำสั่ง **New** เพื่อสร้าง Query สำหรับการเรียกข้อมูลจากฐานข้อมูล



5. ปรากฏหน้าต่าง New Query

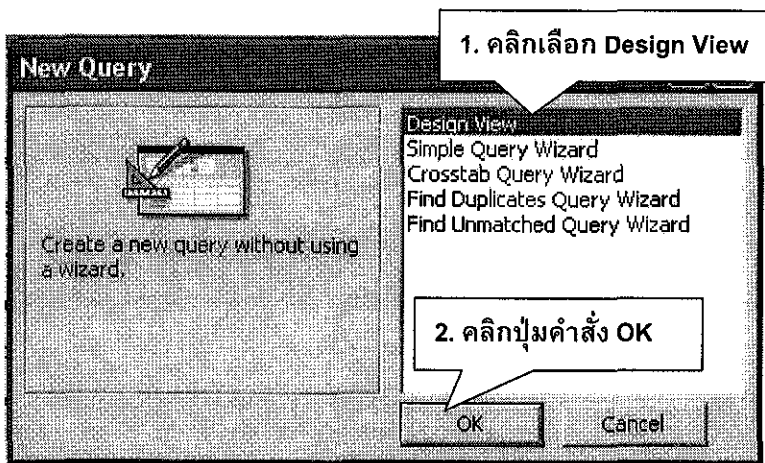


รูปแบบในการสร้าง Query

โปรแกรม Microsoft Access 2002 ได้กำหนดรูปแบบที่ใช้ในการสร้าง Query ดังนี้

รูปแบบ	คำอธิบาย
Design View	การสร้าง Query ด้วยตนเอง เป็นมุมมองที่เหมาะสมสำหรับผู้ใช้ที่เริ่มต้นการใช้งานฐานข้อมูล ซึ่งผู้ใช้สามารถกำหนดฟิลด์และตารางที่ต้องการสืบค้นข้อมูลได้ โดยสามารถเลือกจาก 1 ตารางหรือมากกว่า นอกจากนี้ยังสามารถกำหนดเงื่อนไขต่างๆ ในการสืบค้นข้อมูลได้อีกด้วย
Simple Query Wizard	การสร้าง Query ที่ผู้ใช้สามารถสร้างได้ 2 รูปแบบ คือ <input type="radio"/> สร้างเพื่อค้นหาหรือแสดงรายละเอียด (Detail) <input type="radio"/> สร้างเพื่อแสดงค่าสรุป (Summary) ของข้อมูลในตารางที่กำหนด
Crosstab Query Wizard	การสร้าง Query โดยแสดงความสัมพันธ์ของข้อมูล 2 ชุดในลักษณะของตารางที่มีทั้งแนวตั้งและแนวนอน พร้อมทั้งคำนวณสรุปผลข้อมูลและนำผลที่ได้มาแสดง
Find Duplicates Query Wizard	การสร้าง Query เมื่อต้องการหาข้อมูลที่ซ้ำกันในฟิลด์ที่กำหนด โดยเรคคอร์ดทั้งหมดที่มีข้อมูลซ้ำกันในฟิลด์ที่ระบุนั้นจะถูกนำมาแสดงเป็นกลุ่มต่อเนื่อง
Find Unmatched Query Wizard	การสร้าง Query เพื่อหาข้อมูลจากตารางหนึ่งที่ไม่มีข้อมูลที่สัมพันธ์กันเลย (มักใช้กับฐานข้อมูลที่ซับซ้อนมากขึ้นและมีหลาย Table)

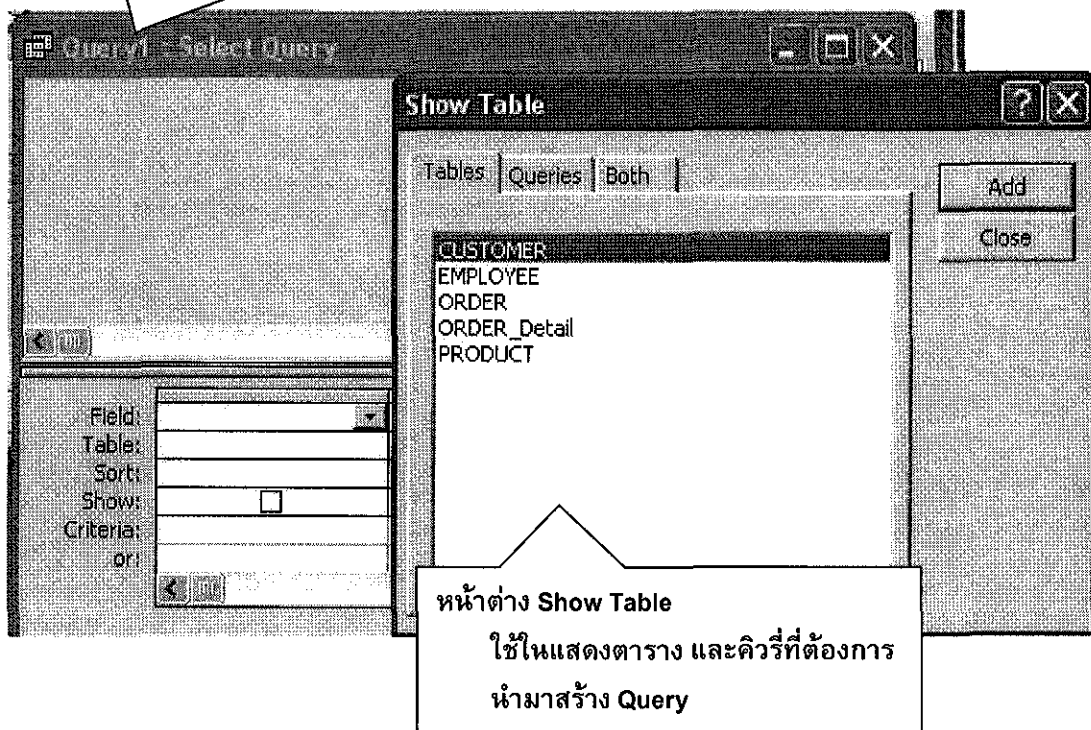
6. คลิกเลือกคำสั่ง Design View แล้วคลิกปุ่ม OK



7. ปรากฏหน้าต่าง 2 หน้าต่างขึ้นมาพร้อมกันดังรูป ประกอบด้วย

- 1) หน้าต่าง Select Query : Query1 ซึ่งต่อไปจะเรียกว่า หน้าต่างออกแบบ Query
- 2) หน้าต่าง Show Table

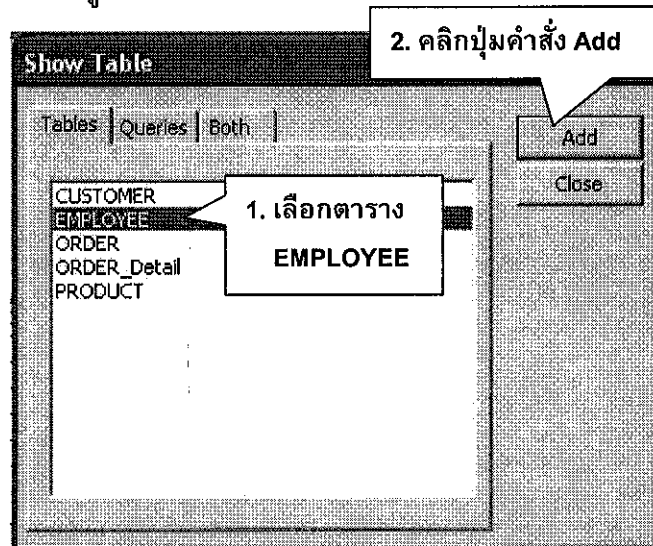
หน้าต่าง Select Query
ใช้ในการกำหนดฟิลด์และตาราง รวมถึงเงื่อนไขที่ใช้ในการสืบค้นข้อมูล



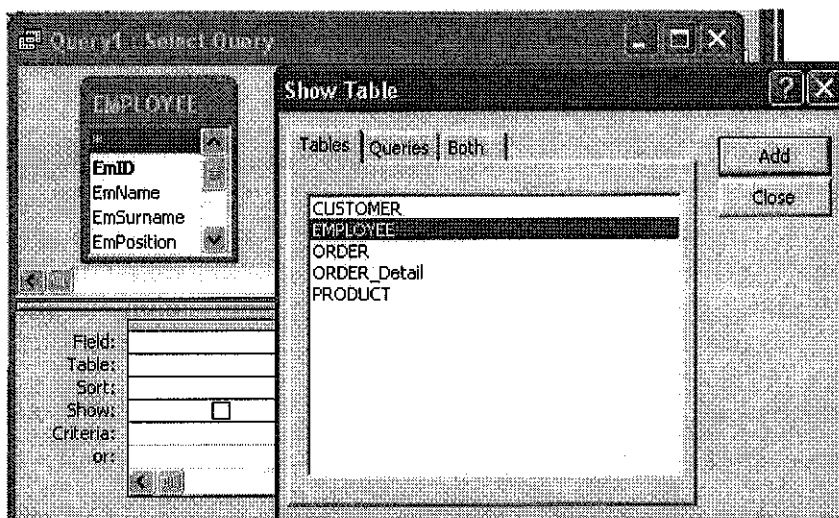
ตัวอย่าง : ต้องการทราบรหัส, ชื่อ-นามสกุล, ตำแหน่ง และเงินเดือนของพนักงานทั้งหมด

8. จากตัวอย่างด้านบนพบว่า ข้อมูลที่ต้องการได้แก่ รหัส, ชื่อ-นามสกุล, ตำแหน่ง และเงินเดือนของพนักงานทั้งหมด ซึ่งข้อมูลต่างๆ นี้อยู่ในตาราง EMPLOYEE ซึ่งมีขั้นตอนดังนี้

- 1) คลิกเลือกตาราง EMPLOYEE จากแถบ Tables ในหน้าต่าง Show Table จากนั้นคลิกคำสั่ง Add ดังรูป



- 2) ปรากฏตาราง EMPLOYEE ในหน้าต่าง Select Query : Query1 ดังรูป



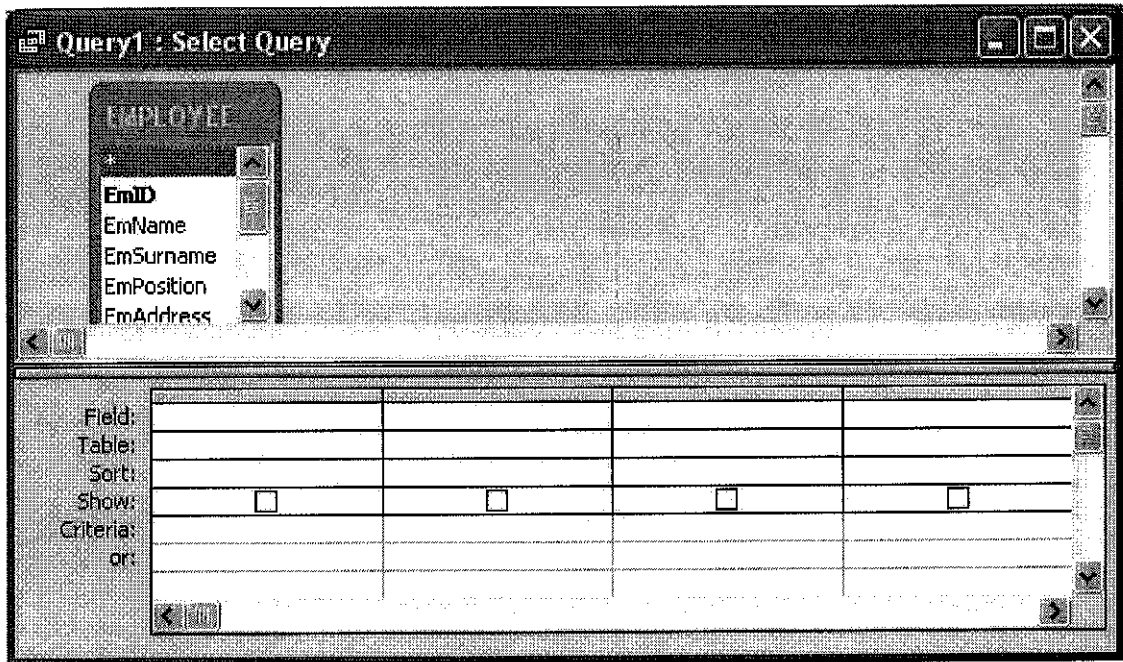
Tip การเพิ่มหรือลบตาราง

หากนักศึกษา เลือกตารางผิดหรือต้องการเพิ่มตารางที่จะนำมาสร้าง Query ให้ดำเนินการ เหมือนกับการสร้างความสัมพันธ์ระหว่างตาราง

การเพิ่มตาราง ให้เลือกเมนู **Query** เลือกคำสั่ง **Show Table**

การลบ ให้เลือกตารางที่ต้องการลบ แล้วเลือกเมนู **Edit** เลือกคำสั่ง **Delete**

- 3) คลิกที่ปุ่ม **Close** ที่หน้าต่าง Show Table เพื่อเปิดหน้าต่าง Show Table จะเหลือเฉพาะหน้าต่าง Select Query: Query1 เท่านั้น ดังรูป



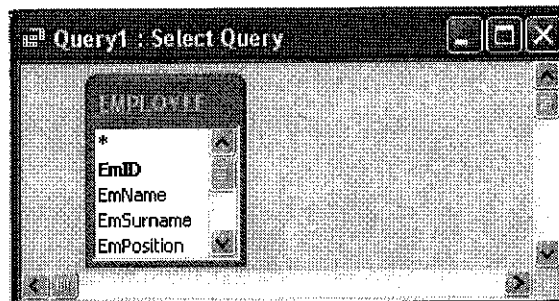
ส่วนประกอบของมุมมอง Query Design

องค์ประกอบ

คำอธิบาย

Field List Pane


ส่วนที่ใช้แสดงแหล่งข้อมูลที่ใช้ในการสร้าง Query และเป็นส่วนที่ใช้แสดง การเชื่อมโยงความสัมพันธ์ของแหล่งข้อมูล



ส่วนที่ใช้ระบุฟิลด์ข้อมูลที่ต้องการนำมาแสดงเป็นผลลัพธ์ของ Query ประกอบด้วย

☞ **Field:** ระบุชื่อฟิลด์ที่ต้องการให้แสดงผลลัพธ์ หรือใส่ชื่อฟิลด์ที่ตั้งขึ้นใหม่ นอกจากนี้ยังสามารถใส่สูตรคำนวณ หรือฟังก์ชันที่ใช้กำหนดค่าข้อมูลของฟิลด์ได้

☞ **Tables:** ระบุชื่อตารางที่มาของฟิลด์

☞ **Total:** ระบุฟังก์ชันที่ใช้ในการคำนวณ (ปรากฏเมื่อคลิกปุ่ม  เพื่อเรียกใช้ฟังก์ชันคำนวณ)

☞ **Sort:** กำหนดรูปแบบการเรียงลำดับของข้อมูลภายในฟิลด์นั้นๆ รูปแบบการเรียงลำดับมี 3 รูปแบบดังนี้

- ☞ **Ascending** กำหนดการเรียงลำดับจากน้อยไปหามาก
- ☞ **Descending** กำหนดการเรียงลำดับจากมากไปหาน้อย
- ☞ **Not Sorted** ไม่กำหนดการเรียงลำดับ

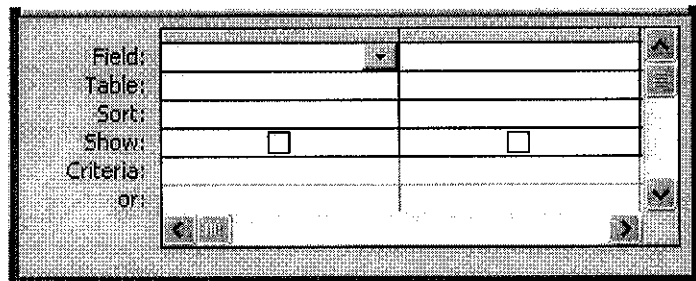
ในกรณีที่มีการกำหนดให้เรียงลำดับมากกว่า 1 ฟิลด์ จะยึดฟิลด์ที่อยู่ทางด้านซ้ายเป็นหลัก

☞ **Show:** กำหนดให้แสดงหรือไม่แสดงข้อมูลภายในฟิลด์ โดยคลิกที่กล่องสี่เหลี่ยม () ถ้ามีเครื่องหมายถูก () หมายถึงกำหนดให้แสดงข้อมูล

☞ **Criteria:** ระบุเงื่อนไข สูตร และฟังก์ชันต่างๆ ที่เป็นตัวกำหนดผลลัพธ์ของคิวรี

☞ **Or:** ระบุเงื่อนไขที่สอง ในลักษณะของคำสั่ง OR ของการเปรียบเทียบค่าในทางตรรกศาสตร์ (ศึกษาเพิ่มเติมจากการสร้างเงื่อนไขแบบนิพจน์ หน้า)

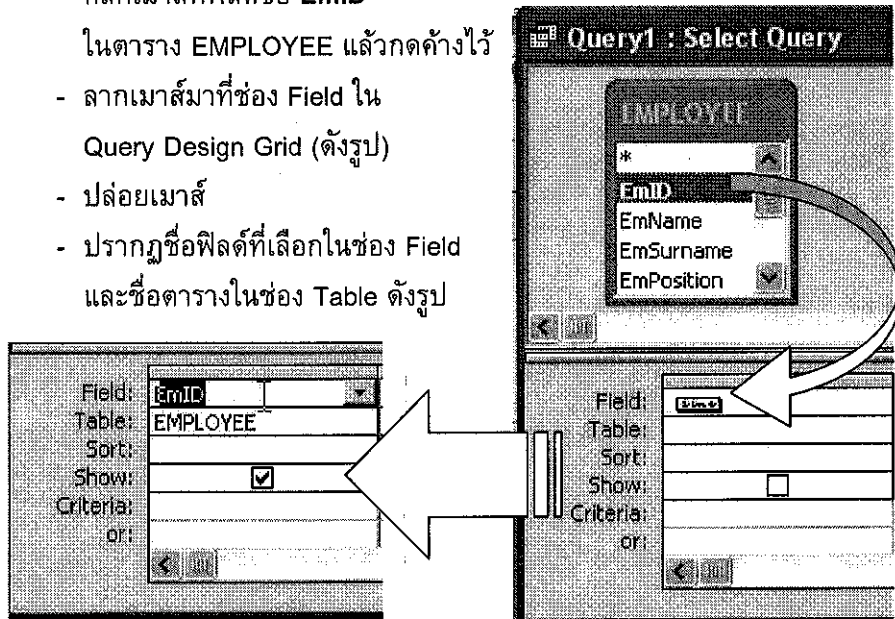
Query Design Grid



- 4) เลือกฟิลด์ที่ต้องการแสดงผลการสืบค้นโดยการคลิกเมาส์ที่ชื่อฟิลด์ที่ต้องการในตารางที่ปรากฏอยู่ใน Field List Pane เช่น กรณีที่เลือกแสดงฟิลด์ชื่อ **EmID** ในตาราง EMPLOYEE เป็นต้น ซึ่งมีขั้นตอนดังนี้

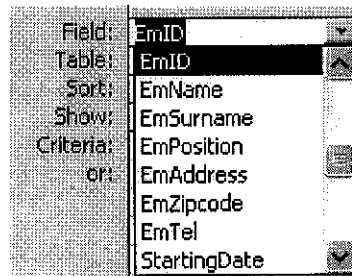
วิธีที่ 1

- คลิกเมาส์ที่ฟิลด์ชื่อ **EmID** ในตาราง EMPLOYEE แล้วกดค้างไว้
- ลากเมาส์มาที่ช่อง Field ใน Query Design Grid (ดังรูป)
- ปล่อยเมาส์
- ปรากฏชื่อฟิลด์ที่เลือกในช่อง Field และชื่อตารางในช่อง Table ดังรูป

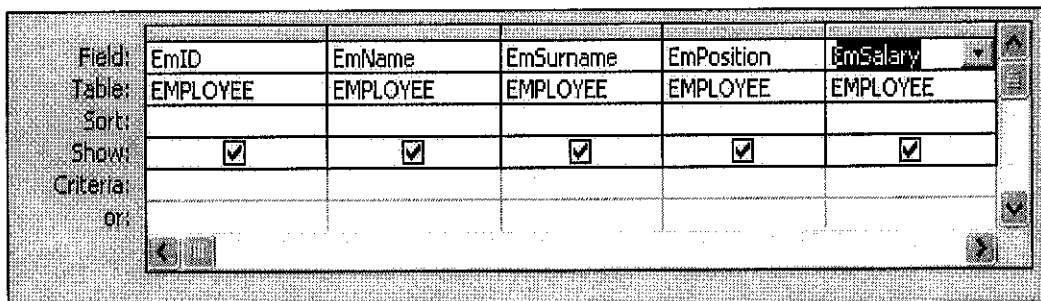


วิธีที่ 2

- คลิกที่เครื่องหมาย  ด้านหลังช่อง Field ใน Query Design Grid จะปรากฏรายชื่อฟิลด์ ดังรูป
- คลิกเลือกฟิลด์ **EmID**
- ปรากฏชื่อฟิลด์ที่เลือกในช่อง Field และชื่อตารางในช่อง Table



- 5) ให้นักศึกษาดำเนินการเลือกฟิลด์ชื่อ EmName, EmSurname, EmPosition และ EmSalary มาเรียงต่อกันไว้ตามลำดับ ดังรูป

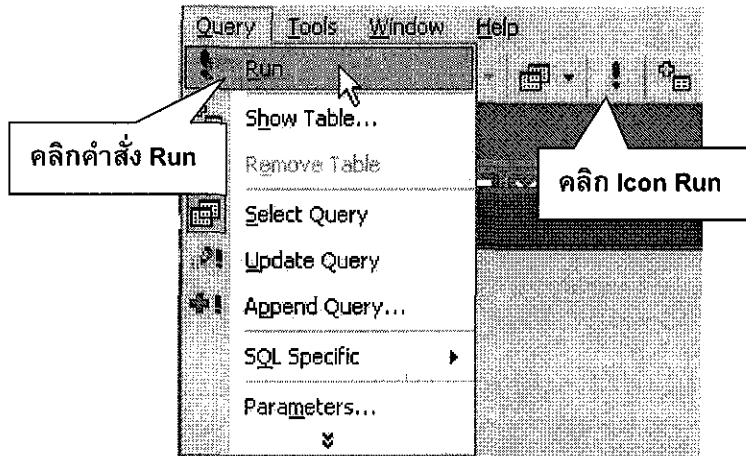


การดูผลลัพธ์ Query

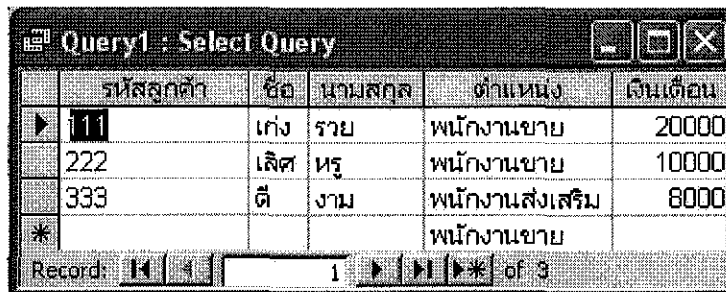
การดูผลลัพธ์การสร้าง Query มีขั้นตอนดังนี้

1. คลิกที่เมนูคำสั่ง Query จากนั้นเลือกคำสั่ง Run

หรือ คลิกที่ Icon  บนแถบเครื่องมือ



2. ปรากฏผลลัพธ์จาก Query ที่สร้างขึ้น ดังรูป




รหัสนลูกค้า	ชื่อ	นามสกุล	ตำแหน่ง	เงินเดือน
111	เก่ง	ราย	พนักงานขาย	20000
222	เลิศ	หรุ	พนักงานขาย	10000
333	ดี	งาม	พนักงานส่งเสริม	8000
*			พนักงานขาย	

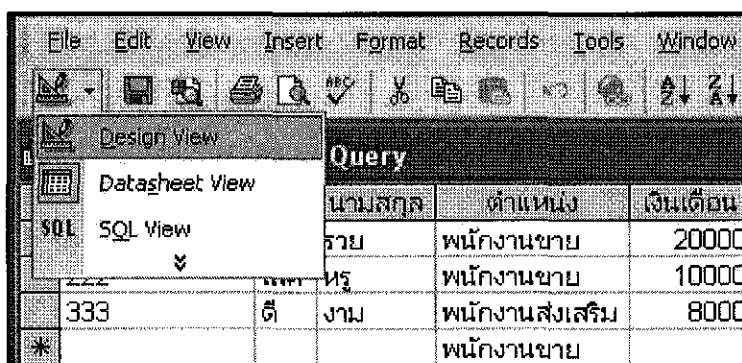
Record: 1 of 3

การกลับหน้าตา Query Design

การกลับสู่หน้าตา Query Design มีขั้นตอนดังนี้

1. คลิกเมนูคำสั่ง View จากนั้นเลือกคำสั่ง Design View หรือ

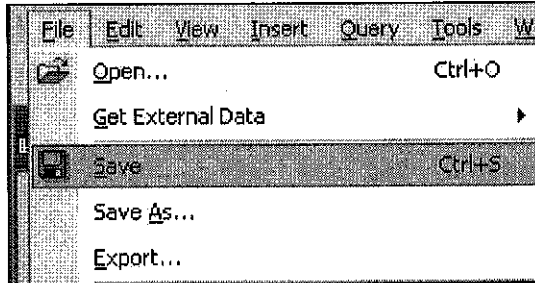
2. หรือคลิกที่ Icon  จากนั้นเลือกคำสั่ง Design View ดังรูป




การบันทึก Query

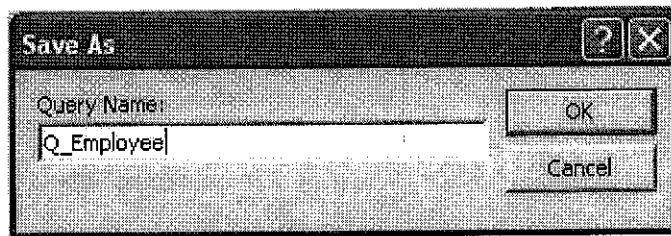
การบันทึก Query มีขั้นตอนดังนี้

1. คลิกเมนูคำสั่ง File เลือกคำสั่ง Save จะปรากฏหน้าต่าง Save as

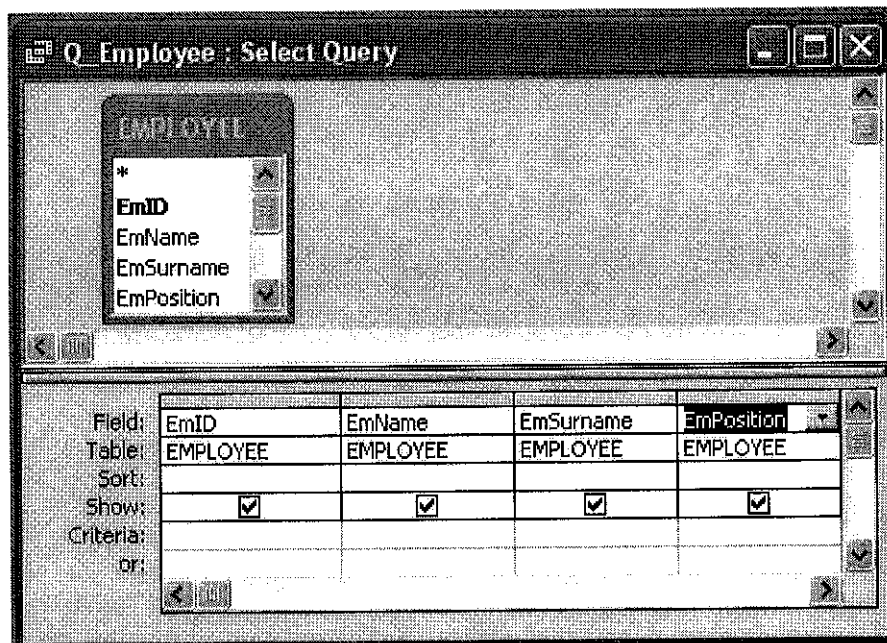


หรือ คลิก Icon  บนแถบเครื่องมือ (Toolbar)

2. พิมพ์ชื่อ Query ว่า **Q_Employee** แล้วคลิกปุ่ม OK ดังรูป



3. หน้าจอ Select Query : จะเปลี่ยนชื่อจาก Query1 เป็น Q_Employee ดังรูป



การแก้ไขการออกแบบ Query

การลบฟิลด์ใน Query

ตัวอย่าง กรณีที่ต้องการลบฟิลด์ EmSalary มีขั้นตอนดังนี้

1. เลื่อนเมาส์ไปด้านบนของคำว่า EmSalary จะปรากฏลูกศร ↓

EmPosition	EmSalary
EMPLOYEE	EMPLOYEE
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

2. คลิกเมาส์ 1 ครั้ง จะปรากฏแถบสีดำที่ช่องฟิลด์นั้น ดังรูป
3. คลิกเลือกคำสั่ง Edit จากนั้นเลือกคำสั่ง Delete หรือคลิกปุ่ม Delete บนแป้นพิมพ์

การแทรกฟิลด์ใน Query

ตัวอย่าง กรณีที่ต้องการแทรกฟิลด์ EmTel ไว้ด้านหน้าฟิลด์ EmPosition มีขั้นตอนดังนี้

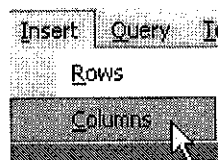
1. เลื่อนเมาส์ไปด้านบนของคำว่า EmPosition จะปรากฏลูกศร ↓

EmSurname	EmPosition	EmSalary
EMPLOYEE	EMPLOYEE	EMPLOYEE
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

2. คลิกเมาส์ 1 ครั้ง จะปรากฏแถบสีดำที่ช่องฟิลด์นั้น ดังรูป

EmSurname	EmPosition	EmSalary
EMPLOYEE		EMPLOYEE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

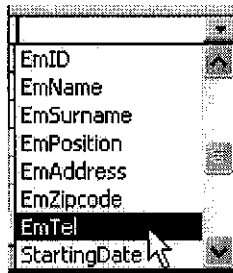
3. คลิกเลือกคำสั่ง Insert จากนั้นเลือกคำสั่ง Columns ดังรูป



4. ปรากฏคอลัมน์ใหม่ระหว่างฟิลด์ EmSurname กับ EmPosition ดังรูป

EmSurname		EmPosition
EMPLOYEE		EMPLOYEE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

5. เลือกฟิลด์ที่ต้องการแสดง (ฟิลด์ EmTel) ดังรูป



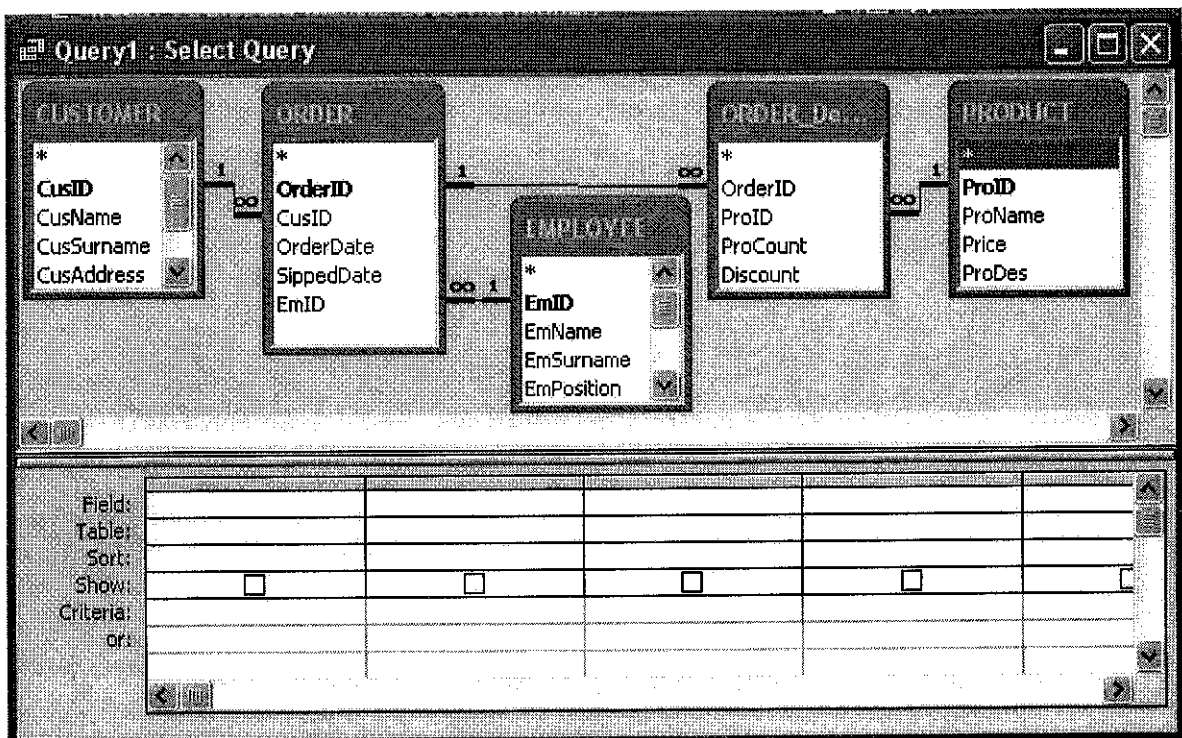
EmSurname	EmTel	EmPosition
EMPLOYEE	EMPLOYEE	EMPLOYEE
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

ขั้นตอนการปฏิบัติที่ 3.2

การสร้าง Query จาก 2 ตารางขึ้นไป

การสร้าง Query จาก 2 ตารางขึ้นไป หรือการสร้าง Query แบบเชื่อมความสัมพันธ์ระหว่างตาราง เป็นการสร้าง Query โดยการใช้ฟิลด์ในตารางต่างๆ ร่วมกัน เช่น ต้องการทราบรายการสั่งซื้อของลูกค้าแต่ละคน พร้อมรายละเอียดการสั่งซื้อ มีขั้นตอนดังนี้

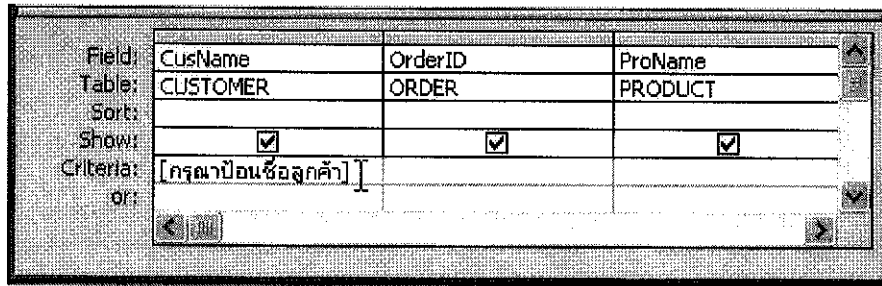
1. คลิกเลือก **Object: Query** จากหน้าต่าง **Database: Week1** แล้วสร้าง Query ใหม่
2. ปรากฏหน้าต่าง **New Query** จากนั้นให้คลิกเลือกคำสั่ง **Design View** แล้วคลิกปุ่ม **OK**
3. ปรากฏหน้าต่าง **Show Table** ให้เลือกตารางทุกตาราง
4. คลิกปุ่ม **Close** เพื่อปิดหน้าต่าง **Show Table**
5. ปรากฏหน้าต่าง **Select Query: Query1** ดังรูป



6. เลือกฟิลด์จากตารางตามลำดับดังนี้

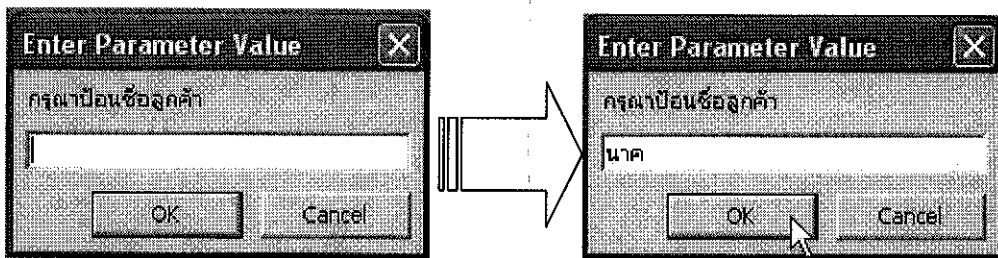
CusName	จากตาราง	CUSTOMER
OrderID	จากตาราง	ORDER
ProName	จากตาราง	PRODUCT
Price	จากตาราง	PRODUCT
ProCount	จากตาราง	Order-Detail

7. ที่ช่อง Criteria ของฟิลด์ CusName ให้พิมพ์ข้อความว่า “ [กรุณาป้อนชื่อลูกค้า] ” ดังรูป!



8. สั่ง Run Query โดยเลือกเมนู Query แล้วเลือกคำสั่ง Run หรือคลิกที่ปุ่ม ! สีแดง ที่แถบ Toolbar

9. ปรากฏ Dialog box: กรุณาป้อนชื่อลูกค้า ให้นักศึกษาป้อนชื่อ นาค แล้วกดปุ่ม OK



10. ปรากฏตารางแสดงผลการสืบค้นข้อมูลจากคำว่า "นาค" ดังรูป

ชื่อ	รหัสการสั่ง	ชื่อสินค้า	ราคาต่อหน่วย	จำนวน
▶ นาค	001	กางเกง	600.00	20
นาค	001	เสื้อยืด	200.00	30
นาค	003	เข็มขัด	300.00	5
นาค	003	รองเท้า	900.00	30
*				

Record: 1 of 4

11. บันทึก Query ที่สร้างขึ้นใหม่ว่า Q_Order1

ขั้นตอนการปฏิบัติที่ 3.3 : การเพิ่มความสามารถให้กับ Query

การสร้างนิพจน์เพื่อการคำนวณ

นิพจน์ (Expression) หมายถึง ข้อความที่สร้างขึ้นจากฟิลด์, ข้อมูลภายในฟิลด์, ตาราง และเครื่องหมายในการคำนวณต่างๆ ทั้งการคำนวณทางคณิตศาสตร์และตรรกศาสตร์ ประมวลผลออกมาเป็นข้อมูลที่เราต้องการ

การสร้างนิพจน์จะเป็นการเขียนสูตรให้เกิดการกระทำกับข้อมูลในรูปแบบที่เรา กำหนด ซึ่งการสร้างนิพจน์เหมือนกับการเขียนสูตรคำนวณทั่วไป เช่น

$$\text{ราคารวม} = \text{ราคาสินค้า} \times \text{จำนวนสินค้า}$$

ซึ่งเราสามารถเขียนลงในบรรทัดฟิลด์ (Field) ได้ทันที โดยการใส่เครื่องหมายต่างๆ เพื่อใช้ในการอ้างอิงถึงแหล่งที่มาของข้อมูล โดยรายละเอียดการอ้างอิงถึงแหล่งข้อมูลมีดังนี้

รูปแบบ	คำอธิบาย	ตัวอย่าง
[ชื่อฟิลด์]	อ้างอิงชื่อฟิลด์ข้อมูลที่ต้องการ	[Price]
[ชื่อตาราง.ชื่อฟิลด์]	อ้างอิงชื่อฟิลด์ในกรณีที่มีชื่อฟิลด์นั้นมีอยู่ในหลายตาราง ให้ใส่ชื่อตารางกำกับ	[Product.Price]
ชื่อวัตถุ!ชื่อตาราง!ชื่อฟิลด์	อ้างอิงองค์ประกอบย่อยของวัตถุ	Table!Product!Price
ชื่อวัตถุ!ชื่อตาราง!ชื่อฟิลด์.คุณสมบัติ	อ้างอิงถึงคุณสมบัติของฟิลด์	Table!Product!Price.Format

สำหรับเครื่องหมายที่ใช้ในการคำนวณในการสร้างนิพจน์นั้นมีดังนี้

1. เครื่องหมายคำนวณทางคณิตศาสตร์

รูปแบบ	คำอธิบาย	ตัวอย่าง
+	รวมค่าของฟิลด์ที่หนึ่งเข้ากับฟิลด์ที่สอง	[ผลรวม] : [Number1] + [Number2]
-	ลบค่าของฟิลด์ที่สองออกจากฟิลด์ที่หนึ่ง	[ผลรวม] : [Number1] - [Number2]
*	คูณค่าของฟิลด์ที่หนึ่งกับฟิลด์ที่สอง	[ผลรวม] : [Number1] * [Number2]
/	ฟิลด์ที่หนึ่งหารด้วยฟิลด์ที่สอง	[ผลรวม] : [Number1] / [Number2]
\	ฟิลด์ที่หนึ่งหารด้วยฟิลด์ที่สอง แต่ผลลัพธ์ที่ได้คือ จำนวนเต็มที่ไม่นับรวมเศษ (เอาเฉพาะจำนวนเต็ม)	[ผลรวม] : [Number1] \ [Number2]
Mod	ฟิลด์ที่หนึ่งหารด้วยฟิลด์ที่สอง แต่ผลลัพธ์ที่ได้คือเศษที่ได้จากการหารไม่นับจำนวนเต็ม (เอาเฉพาะเศษ)	[ผลรวม] : [Number1] Mod [Number2]
^	ยกกำลังฟิลด์ที่กำหนด	[ผลรวม] : [Number1] ^ 2

2. เครื่องหมายการเปรียบเทียบค่าในทางตรรกศาสตร์

รูปแบบ	คำอธิบาย	ตัวอย่าง
IIf	ผลลัพธ์ที่ได้จากการเปรียบเทียบเงื่อนไข โดยมีการกำหนดให้ทำงานตามคำสั่งที่ 1 เมื่อข้อมูลตรงกับเงื่อนไข หากไม่ตรงกับเงื่อนไขให้ทำงานตามคำสั่งที่ 2	IIf (เงื่อนไข, คำสั่งเมื่อเงื่อนไขเป็นจริง, คำสั่งเมื่อเงื่อนไขเป็นเท็จ) IIf ([Product.ProID] = [Order_Detail.ProID], [Order_Detail.Count], 0)

ตัวอย่าง : ต้องการสร้างนิพจน์เพื่อคำนวณหาราคารวมของสินค้าที่ขายไป

$$\text{ราคารวม} = \text{ราคาสินค้า} \times \text{จำนวนสินค้าที่ขาย}$$

การสร้างนิพจน์มีขั้นตอนดังนี้

1. สร้างคอลัมน์ต่อท้ายจากคอลัมน์ ProCount
2. พิมพ์ข้อความ " ราคารวม: [Price]* ในช่อง Field ที่ต่อจากช่องฟิลด์ ProCount ดังรูป

ProCount	ราคารวม: [Price]*[ProCount]
ORDER_Detail	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

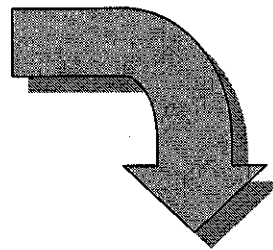
3. สั่ง Run Query ใหม่ จะพบว่าช่องราคารวมจะมีผลการคำนวณให้เรียบร้อยแล้ว

Enter Parameter Value

กรุณาป้อนข้อมูลค่า

ขนาด

OK Cancel



Query1 : Select Query

ชื่อ	รหัสการสั่ง	ชื่อสินค้า	ราคาต่อหน่วย	จำนวน	ราคารวม
▶ ขนาด	001	กางเกง	600.00	20	฿12,000.00
ขนาด	001	เสื้อยืด	200.00	30	฿6,000.00
ขนาด	003	เข็มขัด	300.00	5	฿1,500.00
ขนาด	003	รองเท้า	900.00	30	฿27,000.00
*					

Record: 1 of 4

4. บันทึก Query ที่สร้างขึ้นใหม่ว่า Q_Order2

การสร้าง Query เพื่อค้นหาข้อมูลแบบมีเงื่อนไข

การกำหนดเงื่อนไขใน Query ถือเป็นหลักสำคัญในการเรียกใช้ข้อมูล เนื่องจากสามารถทำให้ผู้ชมมองข้อมูลได้หลายมุมมอง และสามารถนำข้อมูลไปใช้ได้หลากหลายรูปแบบของเงื่อนไขใน Query สามารถกำหนดให้แสดงผลลัพธ์ได้ทั้งที่ตรงกับเงื่อนไขและไม่ตรงกับเงื่อนไข โดยสามารถจำแนกเป็นกลุ่มตามประเภทของเงื่อนไขได้ดังนี้

1. กลุ่มการเปรียบเทียบค่า

เครื่องหมาย	เงื่อนไข / คำอธิบาย	ตัวอย่าง
<	น้อยกว่าค่าที่กำหนด	<200
<=	น้อยกว่าหรือเท่ากับค่าที่กำหนด	<=200
>	มากกว่าค่าที่กำหนด	>200
>=	มากกว่าหรือเท่ากับค่าที่กำหนด	>=200
=	เท่ากับค่าที่กำหนดเท่านั้น	=200
<>	ไม่เท่ากับค่าที่กำหนด	<>200

2. กลุ่มการคำนวณ

รูปแบบ	คำอธิบาย	ตัวอย่าง
+	รวมค่าของฟิลด์ที่หนึ่งเข้ากับฟิลด์ที่สอง	[ผลรวม] : [Number1] + [Number2]
-	ลบค่าของฟิลด์ที่สองออกจากฟิลด์ที่หนึ่ง	[ผลรวม] : [Number1] - [Number2]
*	คูณค่าของฟิลด์ที่หนึ่งกับฟิลด์ที่สอง	[ผลรวม] : [Number1] * [Number2]
/	ฟิลด์ที่หนึ่งหารด้วยฟิลด์ที่สอง	[ผลรวม] : [Number1] / [Number2]
\	ฟิลด์ที่หนึ่งหารด้วยฟิลด์ที่สอง แต่ผลลัพธ์ที่ได้คือ จำนวนเต็มที่ไม่นับรวมเศษ (เอาเฉพาะจำนวนเต็ม)	[ผลรวม] : [Number1] \ [Number2]
Mod	ฟิลด์ที่หนึ่งหารด้วยฟิลด์ที่สอง แต่ผลลัพธ์ที่ได้คือเศษที่ได้จากการหารไม่นับจำนวนเต็ม (เอาเฉพาะเศษ)	[ผลรวม] : [Number1] Mod [Number2]
^	ยกกำลังฟิลด์ที่กำหนด	[ผลรวม] : [Number1] ^ 2

ลำดับการทำงานของเครื่องหมายในกลุ่มการคำนวณ

ลำดับที่	เครื่องหมาย
1	() วงเล็บ
2	^
3	*, /, \, Mod
4	+, -

3. กลุ่มการเชื่อมข้อความ

เครื่องหมาย	เงื่อนไข / คำอธิบาย	ตัวอย่าง
[ชื่อฟิลด์] & [ชื่อฟิลด์]	กำหนดให้แสดงข้อมูลโดยนำข้อมูลในฟิลด์ที่สองมาต่อท้ายข้อมูลฟิลด์แรก	[ProName] & [Price]
[ชื่อฟิลด์] + [ชื่อฟิลด์]	เหมือนเครื่องหมาย & (กำหนดให้แสดงข้อมูลโดยนำข้อมูลในฟิลด์ที่สองมาต่อท้ายข้อมูลฟิลด์แรก) <u>จะต้องเป็นฟิลด์ที่เก็บข้อมูลประเภท Text เท่านั้น</u>	[ProdID] + [ProName]

4. กลุ่มตรรกศาสตร์

เครื่องหมาย	เงื่อนไข / คำอธิบาย	ตัวอย่าง
And	ผลลัพธ์ที่ได้จะต้องอยู่ในเงื่อนไขที่กำหนดไว้เท่านั้น ถ้าต้องการข้อมูลเฉพาะที่ตรงกับค่าที่กำหนด ให้ใส่คำสั่ง Between เพิ่มข้างหน้าคำสั่ง And	> 50 And <120 หรือ Between "001" And "005" หรือ Between #04/05/2549# And #30/05/2549#
Or	ผลลัพธ์ที่ได้จะต้องประกอบด้วยค่าใดค่าหนึ่งที่กำหนด	"002" Or "006" หรือ 2,000 Or 3,000
Not	ผลลัพธ์ที่ได้จะต้องไม่ตรงกับค่าที่กำหนด	Not 8,500

5. กลุ่มค้นหาข้อมูล

เครื่องหมาย	เงื่อนไข / คำอธิบาย	ตัวอย่าง
"ข้อความ"	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลตรงกับข้อความ	"สมชาย"
Not "ข้อความ"	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลไม่ตรงกับข้อความ	Not "นครราชสีมา"
Not ข้อความ*	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลที่ไม่ได้ขึ้นต้นด้วยข้อความ	Not นคร*
Like "ข้อความ"	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลที่ขึ้นต้นด้วยข้อความ	Like "นคร*"
Like "ข้อความ?"	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลที่ขึ้นต้นด้วยข้อความและต่อท้ายด้วยตัวเลขหรือตัวอักษร 1 ตัว	Like "นคร?"
Like "ข้อความ#"	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลที่ขึ้นต้นด้วยข้อความและต่อท้ายด้วยตัวเลข 1 ตัว	Like "นคร#"

เครื่องหมาย	เงื่อนไข / คำอธิบาย	ตัวอย่าง
#วันเดือนปี#	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลตรงกับวันที่ที่กำหนด	#26/05/2549#
Between #วันเดือนปี# And #วันเดือนปี#	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลอยู่ระหว่างวันที่เริ่มต้นและวันที่สิ้นสุด	Between #26/05/2549# And #26/06/2549#
Date()	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลตรงกับวันที่ปัจจุบัน	Date()
In ("ข้อความ", "ข้อความ")	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดมีข้อมูลตรงกับข้อความใดข้อความหนึ่งในวงเล็บ	In ("นครราชสีมา", "ขอนแก่น")
Is Null	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดเป็นค่าว่าง	Is Null
Is Not Null	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดไม่มีค่าว่าง	Is Not Null
" "	แสดงเฉพาะเรคคอร์ดที่ฟิลด์ที่กำหนดไม่มีค่าข้อมูล	" "

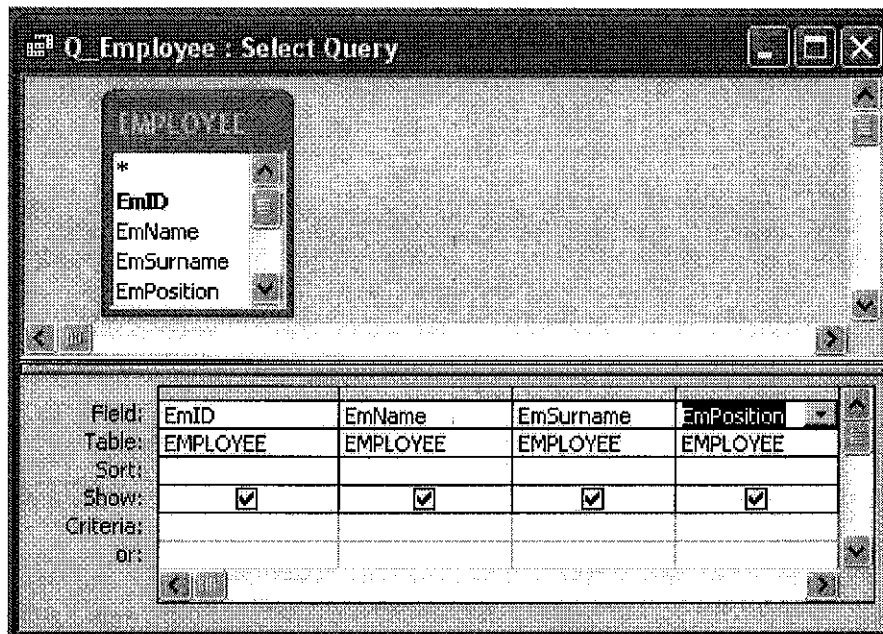
เครื่องหมายที่ใช้ในการค้นหาข้อมูล

เครื่องหมาย	คำอธิบาย
?	แทนตัวอักษร หรือตัวเลข 1 ตัว
*	แทนตัวอักษร หรือตัวเลข มีความยาวตั้งแต่ 0 ตัวขึ้นไป สามารถใช้เป็นตัวอักษรแรก หรือตัวอักษรสุดท้ายก็ได้
#	แทนตัวเลข 1 ตัว

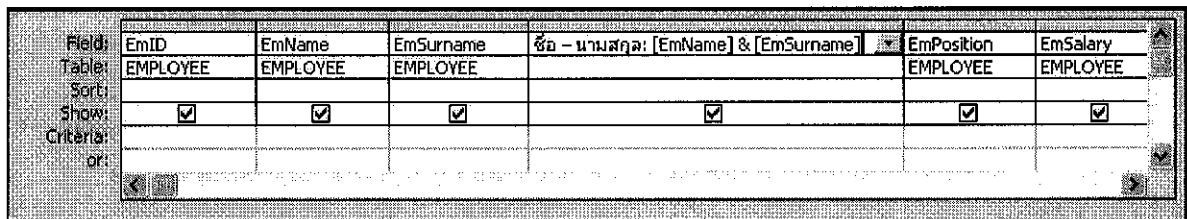
ตัวอย่าง : ต้องการเชื่อมข้อความของฟิลด์ EmName และ EmSurname ให้เป็นฟิลด์ใหม่

ขั้นตอนปฏิบัติ

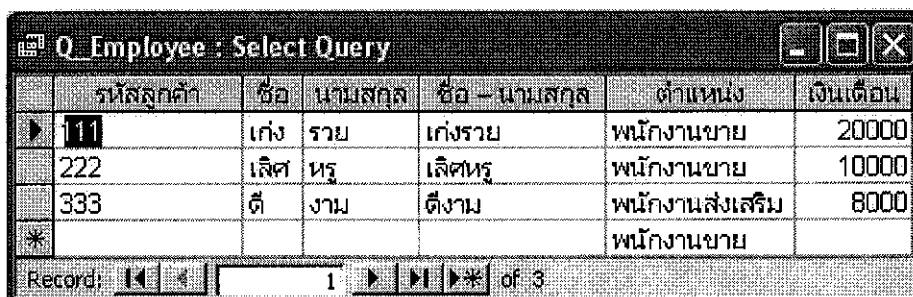
1. จากหน้าต่าง **Database : Week1** ให้เลือก **Object : Query** จากนั้นคลิกเลือก **Query : Q_Employee**
2. คลิกเลือกคำสั่ง **Design** จะปรากฏหน้าต่าง Q_Employee : Select Query ดังรูป



3. แทรกคอลัมน์ใหม่หลังฟิลด์ EmSurname แล้วพิมพ์ในฟิลด์ช่องใหม่ว่า **ชื่อ - นามสกุล: [EmName]&[EmSurname]** ดังรูป



4. สั่ง **Run** จะปรากฏผลลัพธ์ดังภาพ



ข้อสังเกต :

ชื่อและนามสกุลของลูกค้าจะติดกัน

ให้นักศึกษาแก้ไขใหม่ โดยกลับเข้าไปที่หน้าต่าง Design แล้วแก้ไขเป็น

ชื่อ - นามสกุล: [EmName]& " "&[EmSurname] แล้ว Run ผลดูใหม่อีกครั้ง

รหัสลูกค้า	ชื่อ	นามสกุล	ชื่อ - นามสกุล	ตำแหน่ง	เงินเดือน
111	เก่ง	รวย	เก่ง รวย	พนักงานขาย	20000
222	เลิศ	หรุ	เลิศ หรุ	พนักงานขาย	10000
333	ดี	งาม	ดี งาม	พนักงานส่งเสริม	8000
*				พนักงานขาย	

รหัสลูกค้า	ชื่อ	นามสกุล	ชื่อ - นามสกุล	ตำแหน่ง	เงินเดือน
111	เก่ง	รวย	เก่ง รวย	พนักงานขาย	20000
222	เลิศ	หรุ	เลิศ หรุ	พนักงานขาย	10000
333	ดี	งาม	ดี งาม	พนักงานส่งเสริม	8000
*				พนักงานขาย	

5. คลิกที่ช่อง show ในช่องฟิลด์ Emname และ EmSurname เครื่องหมาย ✓ จะไม่ปรากฏ (การไม่แสดงฟิลด์ดังกล่าวในการแสดงผลการค้นข้อมูล)

Field:	EmID	EmName	EmSurname	ชื่อ - นามสกุล: [EmName] & " "& [EmSurname]	EmPosition	EmSalary
Table:	EMPLOYEE	EMPLOYEE	EMPLOYEE		EMPLOYEE	EMPLOYEE
Sort:						
Show:	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:						
or:						

6. Run ผลลัพธ์ดูอีกครั้ง จะพบว่าฟิลด์ชื่อและนามสกุลจะหายไป ดังรูป

รหัสลูกค้า	ชื่อ - นามสกุล	ตำแหน่ง	เงินเดือน
111	เก่ง รวย	พนักงานขาย	20000
222	เลิศ หรุ	พนักงานขาย	10000
333	ดี งาม	พนักงานส่งเสริม	8000
*		พนักงานขาย	

ตัวอย่าง : ต้องการค้นหาข้อมูลของลูกค้าชื่อ ดี (ใช้คำสั่ง Like "ข้อความ")

ขั้นตอนปฏิบัติ

1. จากหน้าต่าง **Database : Week1** ให้เลือก **Object : Query** จากนั้นคลิกเลือก **Query : Q_Employee**
2. คลิกเลือกคำสั่ง **Design** จะปรากฏหน้าต่าง Q_Employee : Select Query
3. ให้นักศึกษาพิมพ์เงื่อนไขในช่อง Criteria ที่ช่องฟิลด์ EmName ว่า Like "ดี" ดังรูป

Field:	EmID	EmName	EmSurname	ชื่อ - นามสกุล: [EmName] & ' '& [EmSurname]	EmPosition	EmSalary
Table:	EMPLOYEE	EMPLOYEE	EMPLOYEE		EMPLOYEE	EMPLOYEE
Sort:						
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		Like "ดี"				
or:						

4. สั่ง **Run** ผลลัพธ์ใหม่อีกครั้ง แล้วสังเกตผลที่เกิดขึ้นจะพบว่า ตารางจะแสดงเฉพาะคนที่ชื่อ "ดี" เท่านั้น ดังรูป

	รหัสลูกค้า	ชื่อ	นามสกุล	ชื่อ - นามสกุล	ตำแหน่ง	เงินเดือน
▶	838	ดี	งาม	ดี งาม	พนักงานส่งเสริม	8000
*					พนักงานขาย	

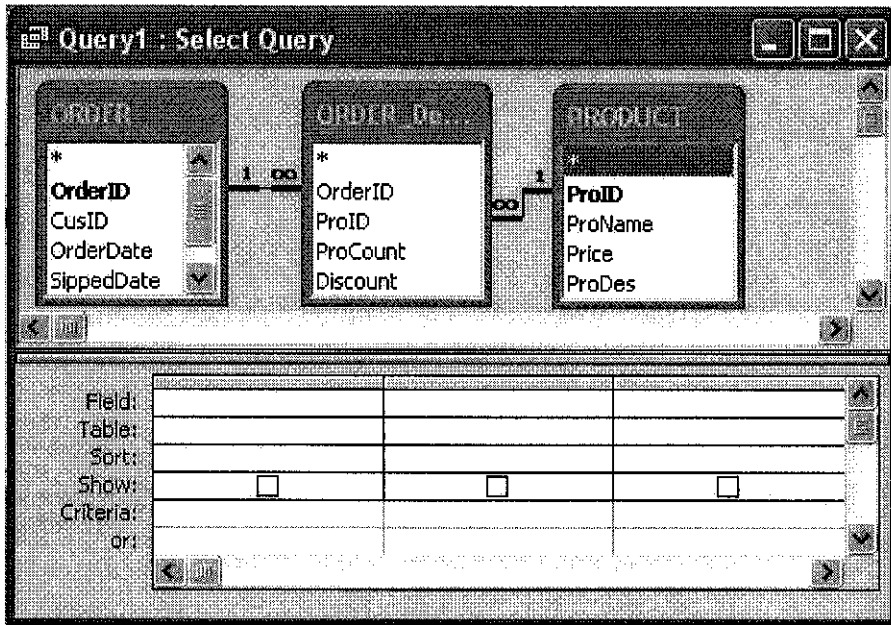
Record: 1 of 1

5. ให้นักศึกษา Save as Query ชื่อว่า Q_Em

ตัวอย่าง : ต้องการค้นหารายการสั่งซื้อสินค้าในแต่ละวัน

ขั้นตอนปฏิบัติ

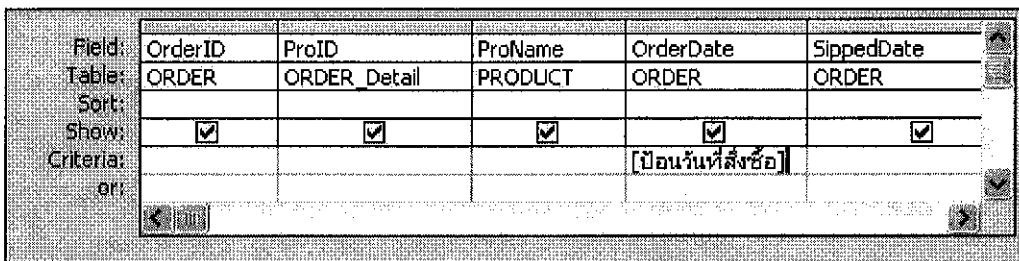
1. จากหน้าจอ **Database: Week1** คลิกเลือก **Object : Query**
2. คลิกที่คำสั่ง **New** เพื่อสร้าง Query ใหม่
3. ปรากฏหน้าต่าง **New Query** จากนั้นให้คลิกเลือกคำสั่ง **Design View** แล้วคลิกปุ่ม **OK**
4. จากหน้าต่าง **Add Table** คลิกที่แถบ **Table** คลิกเลือกตาราง **Order, Order_Detail** และ **Product** ดังรูป



5. จากนั้นเลือกฟิลด์ต่างๆ ตามลำดับดังนี้ เพื่อสร้าง Query

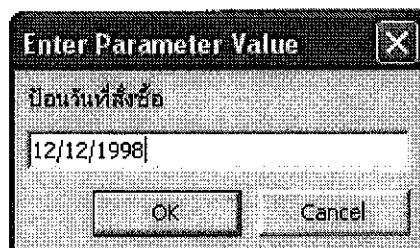
OrderID	จากตาราง	ORDER
ProID	จากตาราง	Order_Detail
ProName	จากตาราง	PRODUCT
OrderDate	จากตาราง	ORDER
ShippedDate	จากตาราง	ORDER

6. พิมพ์เงื่อนไข Criteria ในช่องฟิลด์ OrderDate ว่า [ป้อนวันที่สั่งซื้อ] ดังรูป



6. ให้นักศึกษาสั่ง Save ชื่อ Query ว่า Q_Order2

7. สั่ง Run ผลลัพธ์ จะปรากฏ Dialog box : ป้อนวันที่สั่งซื้อ ให้นักศึกษาทดลองใส่วันที่ "12/12/1998"



7. ปรากฏผลการสืบค้นข้อมูลจากวันที่ ที่ระบุไว้ ดังรูป

รหัสการสั่ง	รหัสสินค้า	ชื่อสินค้า	วันสั่ง	วันส่ง
001	100	กางเกง	12/12/1998	18/12/1998
001	200	เสื้อยืด	12/12/1998	18/12/1998

ตัวอย่าง : ต้องการค้นหาข้อมูลของลูกค้าชื่อ ดี (ใช้คำสั่ง Like "ข้อความ")

ขั้นตอนปฏิบัติ

1. จากหน้าต่าง **Database : Week1** ให้เลือก **Object : Query** จากนั้นคลิกเลือก **Query : Q_Order2**
2. คลิกเลือกคำสั่ง **Design** จะปรากฏหน้าต่าง **Q_Order2 : Select Query**
3. ให้นักศึกษาลบเงื่อนไขเดิมออก
4. พิมพ์เงื่อนไขในช่องฟิลต์ OrderID ว่า **between 002 and 003**

Field:	OrderID	ProID	ProName	OrderDate	SippedDate
Table:	ORDER	ORDER_Detail	PRODUCT	ORDER	ORDER
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	Between "002" And "003"				

5. สั่ง **Save As** ชื่อ Query ใหม่ โดยตั้งชื่อว่า **Q_Order3**
6. สั่ง **Run** ผลลัพธ์จะพบว่าจะแสดงตารางการสั่งซื้อเลขที่ 002-003 เท่านั้น ดังรูป

รหัสการสั่ง	รหัสสินค้า	ชื่อสินค้า	วันสั่ง	วันส่ง
002	100	กางเกง	15/12/1998	20/12/1998
003	300	เข็มขัด	30/12/1998	5/1/1999
003	400	รองเท้า	30/12/1998	5/1/1999
002	200	เสื้อยืด	15/12/1998	20/12/1998

ปฏิบัติการที่ 4

การสร้างแบบฟอร์ม

วัตถุประสงค์ นักศึกษาสามารถ

1. มีความรู้เกี่ยวกับการสร้างแบบฟอร์ม
2. สร้างฟอร์มที่ใช้ในการกรอกข้อมูลแทนการกรอกใน Datasheet ได้
3. สร้างปุ่มคำสั่งต่างๆ เพื่อการทำงานที่สะดวกของฟอร์มได้

ฟอร์ม (Form)

ฟอร์ม (Form) เป็นตัวกลางระหว่างผู้ใช้งานฐานข้อมูลกับตาราง ซึ่งทำหน้าที่รับข้อมูลและนำข้อมูลใส่ลงในตารางแทนการบันทึกข้อมูลลงใน Datasheet ของตาราง และการนำเสนอข้อมูลจากการสืบค้นข้อมูลด้วย Query

ประเภทของ Form

Form สามารถแบ่งประเภทได้ดังนี้

ประเภท	คำอธิบาย
Design View	การสร้างฟอร์มด้วยตนเอง โดยใช้เครื่องมือหรือที่เรียกว่า "คอนโทรล" (Control) ในลักษณะต่างๆ นามาวางบนฟอร์มเพื่อเป็นตัวกลางในการรับข้อมูลจากผู้ใช้บันทึกลงในตาราง และแสดงข้อมูลจากตารางสู่ผู้ใช้
Form Wizard	การสร้างฟอร์มโดยการเรียกใช้วิซาร์ด (Wizard) ซึ่งโปรแกรมจะถามคำถามต่างๆ เกี่ยวกับลักษณะของฟอร์มที่ต้องการ พร้อมแสดงภาพตัวอย่างประกอบทุกขั้นตอน แล้วสร้างออกมาเป็นฟอร์มตามที่ต้องการ
Auto Forms: Columnar	การสร้างฟอร์มจากตารางที่ต้องการโดยอัตโนมัติ ซึ่งรูปแบบฟอร์มที่ได้จะอยู่ในรูปแบบการเรียงลำดับเรคคอร์ดจากบนลงล่าง และแสดง 1 เรคคอร์ดต่อ 1 หน้า
Auto Forms: Tabular	การสร้างฟอร์มจากตารางที่ต้องการโดยอัตโนมัติ ซึ่งรูปแบบฟอร์มที่ได้จะอยู่ในรูปแบบการเรียงลำดับเรคคอร์ดจากซ้ายไปขวา โดย 1 หน้าจะแสดงหลายเรคคอร์ด
Auto Forms: Datasheet	การสร้างฟอร์มจากตารางที่ต้องการโดยอัตโนมัติ ซึ่งรูปแบบฟอร์มที่ได้จะอยู่ในรูปแบบเดียวกันกับมุมมอง Datasheet ของตาราง
Auto Forms: PivotTable	การเรียกใช้คุณสมบัติ PivotTable จากตารางที่ต้องการโดยอัตโนมัติ
Auto Forms: PivotChart	การเรียกใช้คุณสมบัติ PivotChart จากตารางที่ต้องการโดยอัตโนมัติ






ประเภท	คำอธิบาย
Chart Wizard	การสร้างแผนภูมิโดยใช้วิซาร์ด (Wizard) สร้างแผนภูมิ ซึ่งโปรแกรมจะถามคำถามเกี่ยวกับลักษณะของแผนภูมิที่ต้องการ พร้อมแสดงภาพตัวอย่างประกอบทุกขั้นตอน แล้วสร้างออกมาเป็นแผนภูมิในรูปแบบตามต้องการ
PivotTable Wizard	การสร้าง PivotTable โดยใช้วิซาร์ด (Wizard) สร้าง PivotTable ซึ่งโปรแกรมจะมีคำแนะนำและวิธีการใช้งาน PivotTable ที่ต้องการ พร้อมเข้าสู่หน้าต่างการสร้าง PivotTable โดยอัตโนมัติ
















องค์ประกอบของพื้นที่การออกแบบฟอร์ม (Form Design Area)

ส่วน	คำอธิบาย
Form Header / Form Footer	พื้นที่สำหรับวางวัตถุต่างๆ เพื่อใช้ในการตกแต่งให้ฟอร์มมีความสวยงาม และเพื่อแสดงทุกครั้งที่ฟอร์มขึ้นหน้าใหม่ ส่วนมากนิยมใส่เครื่องหมาย, สัญลักษณ์ หรือข้อความต่างๆ
Detail	พื้นที่สำหรับวางฟิลด์และคอนโทรลต่างๆ เพื่อรับข้อความจากผู้ใช้ และแสดงข้อมูลทั้งหมดจากตาราง

คอนโทรล (Controls)

คอนโทรลแต่ละตัวสามารถทำงานภายใต้คุณสมบัติที่แตกต่างกันไป ซึ่งรายละเอียดของคอนโทรลชนิดต่างๆ มีดังนี้

สัญลักษณ์	ชื่อคอนโทรล	คำอธิบาย
	Select Object	ใช้คลิกเลือกคอนโทรล หรือเปลี่ยนจากเมาส์ที่ใช้สร้างคอนโทรลเป็นเมาส์ที่ใช้งานปกติ
	Control Wizard	คอนโทรลบางตัวที่มีวิธีการสร้างที่ซับซ้อน หากแต่การเข้าใจ เครื่องมือนี้จะช่วยเรียกวิซาร์ด (Wizard) ขึ้นมาใช้งานในการสร้างคอนโทรล
	Label	ใช้เขียนข้อความลงบนฟอร์มในลักษณะของคำอธิบายซึ่งไม่เกี่ยวกับข้อมูลและตาราง
	Text Box	ใช้รับข้อมูลจากผู้ใช้บันทึกลงตาราง หรือแสดงข้อมูลจากตาราง สามารถใส่ฟังก์ชันในการคำนวณลงไปและแสดงออกมาเป็นผลลัพธ์ได้
	Option Group	กรอกล้อมกลุ่มตัวเลือกที่สามารถเลือกได้ตัวเลือกเดียว

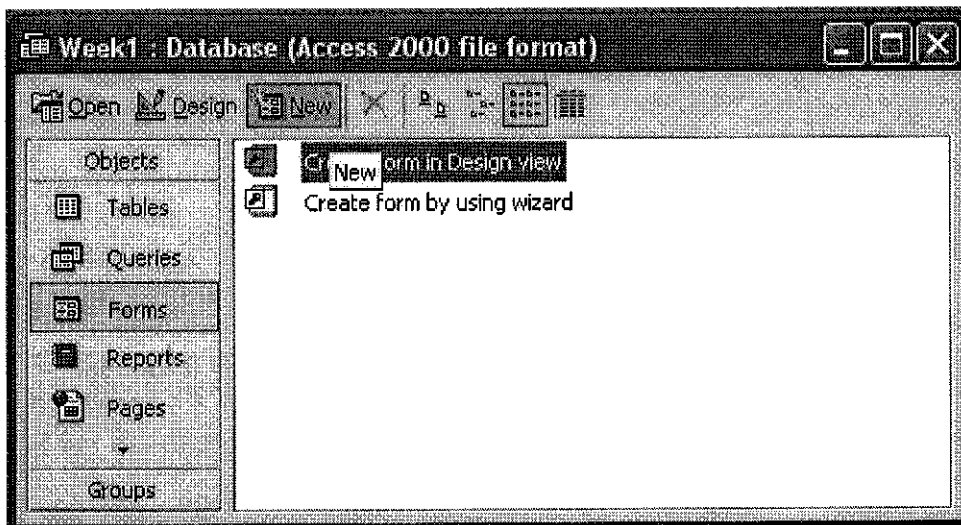
สัญลักษณ์	ชื่อคอนโทรล	คำอธิบาย
	Toggle Button	ปุ่มแบบคลิกเลือก สามารถเลือกได้เพียงตัวเลือกเดียว
	Option Button	ตัวเลือกแบบคลิกเลือก สามารถเลือกได้เพียงตัวเลือกเดียว
	Check Box	ตัวเลือกแบบคลิกเลือก สามารถเลือกได้หลายตัวเลือก
	Combo Box	ใช้คลิกเพื่อเลือกรายการข้อมูลที่ซ่อนไว้ สามารถเลือกได้รายการเดียว
	List Box	คล้ายกับ Combo Box แต่ไม่ซ่อนรายการข้อมูล
	Command Button	ปุ่มที่บรรจุคำสั่งในการทำงาน เช่น คลิกปุ่มเพื่อเพิ่มข้อมูล เป็นต้น
	Image	แสดงรูปภาพที่ไม่ได้จัดเก็บอยู่ในตาราง
	Unbound Object Frame	แสดงข้อมูลชนิด Object ที่ไม่ได้บันทึกอยู่ในตาราง
	Bound Object Frame	แสดงและรับข้อมูลชนิด Object ลงในตาราง
	Page Break	ใช้ในกรณีที่ฟอร์มมีความยาวมากกว่า 1 หน้า 79 จาก ใช้เพื่อกำหนดตำแหน่งที่ฟอร์มจะเลื่อนขึ้นลงเมื่อกดปุ่ม Page up หรือ Page Down หรือขึ้นหน้าใหม่เมื่อพิมพ์ฟอร์ม
	Tab Control	ใช้แสดงฟอร์มอื่นๆ ในลักษณะของแท็บ
	Subform / Subreport	ใช้แสดงฟอร์มอื่นๆ ในฟอร์มหลักที่มีข้อมูลในตารางที่มีความสัมพันธ์กัน
	Line	ใช้สร้างเส้นในฟอร์ม
	Rectangle	ใช้วาดสี่เหลี่ยมในฟอร์ม
	More Controls	ปุ่มสำหรับเรียกใช้คอนโทรลอื่นๆ ที่ซ่อนอยู่

ขั้นตอนการปฏิบัติที่ 4.1

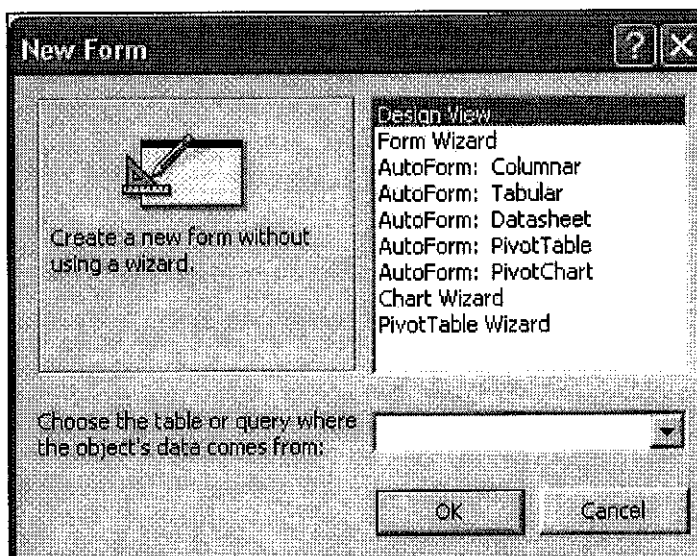
การสร้าง Form ด้วย Form Wizard

คำสั่ง : จงสร้างฟอร์มเพื่อใช้ในการรับค่าและนำเสนอข้อมูลลูกค้า (CUSTOMER)

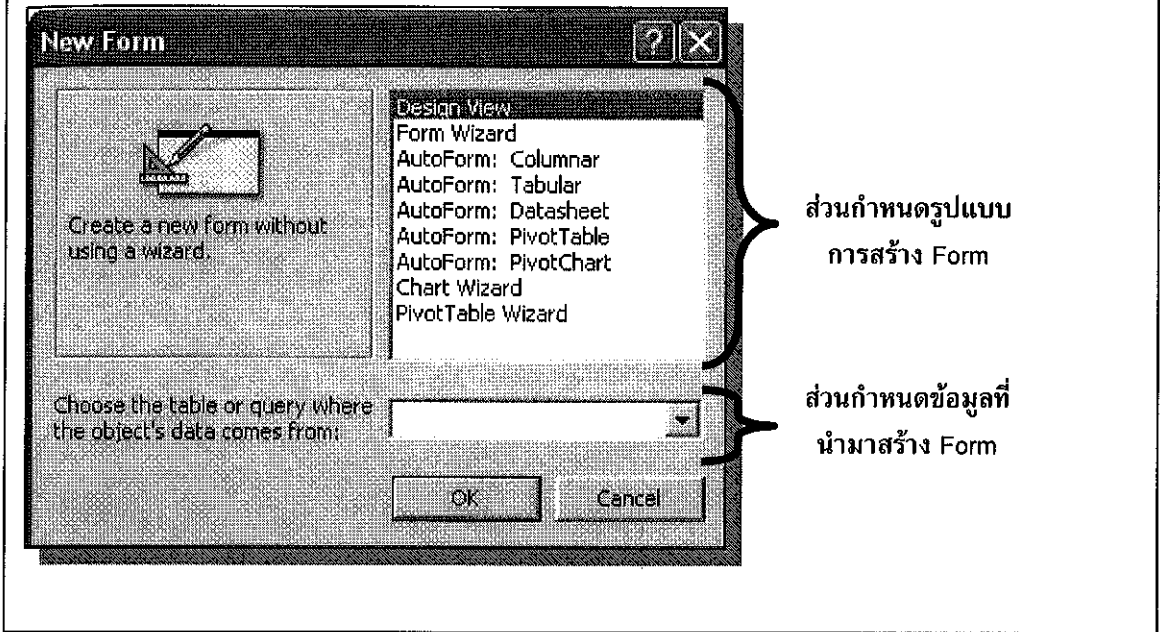
1. เลือก Database: Week1.mdb ขึ้นมาใช้งาน
2. จากหน้าจอ Database Week1 ให้เลือก Object : Form
3. เลือกคำสั่ง New ดังรูป



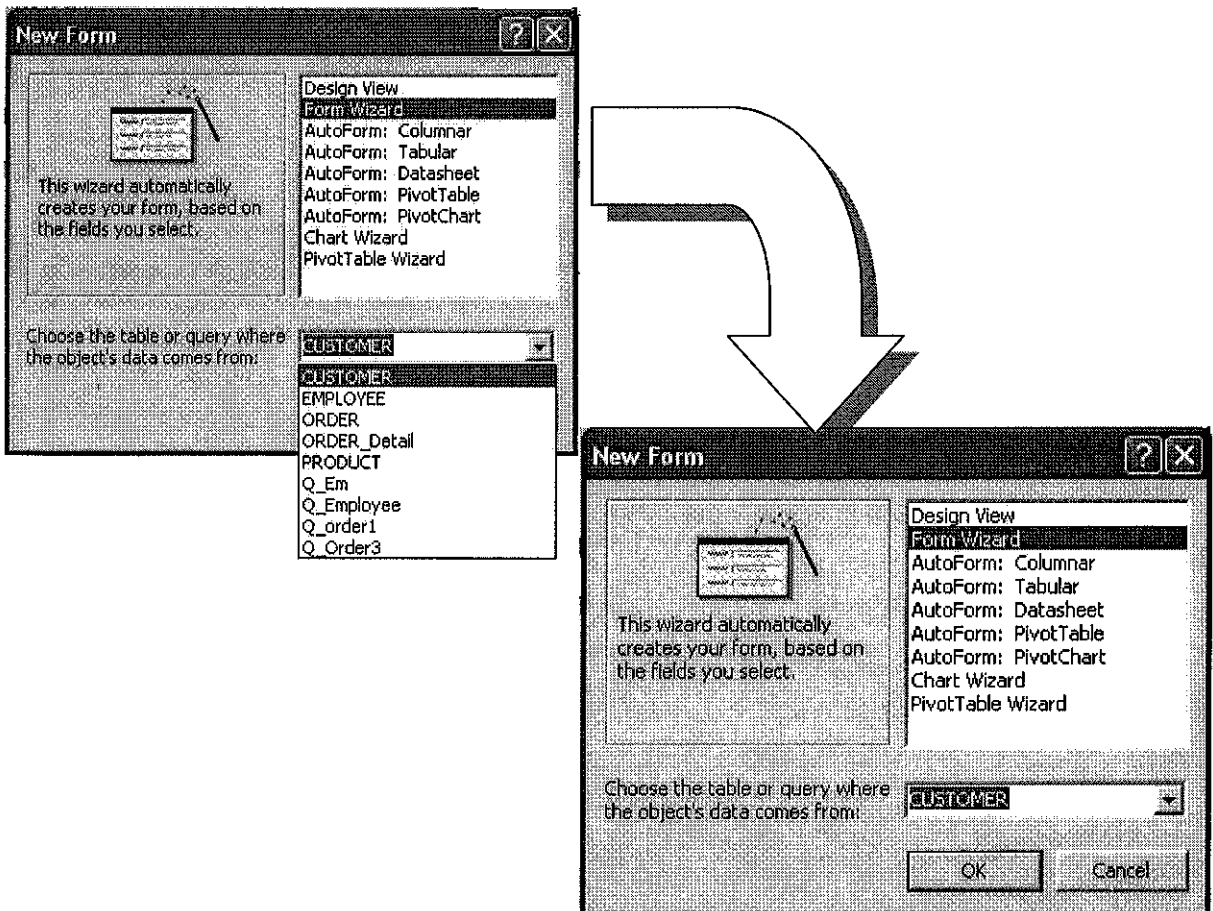
4. ปรากฏหน้าต่างต่าง New Form ดังรูป



หน้าต่าง New Form ประกอบด้วย 2 ส่วนคือ

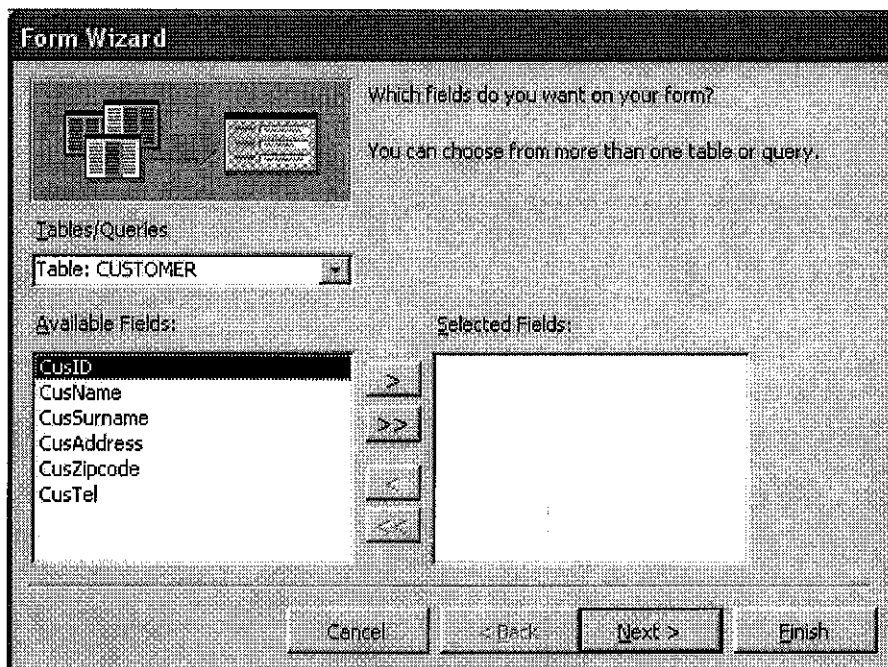


5. คลิกเลือกรูปแบบการสร้าง Form แบบ Form Wizard และคลิกเลือกชื่อตาราง CUSTOMER โดยคลิกที่ปุ่มลูกศรด้านข้าง ดังรูป

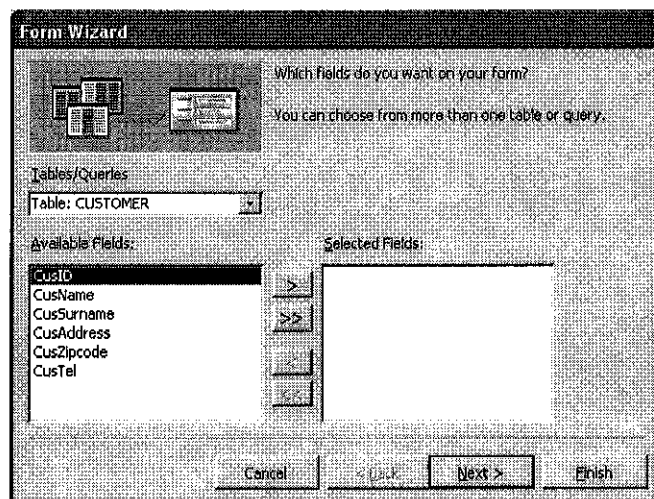


6. คลิกปุ่มคำสั่ง OK

7. ปรากฏหน้าต่าง Form Wizard ดังรูป




หน้าต่าง Form Wizard ประกอบด้วย 3 ส่วนคือ



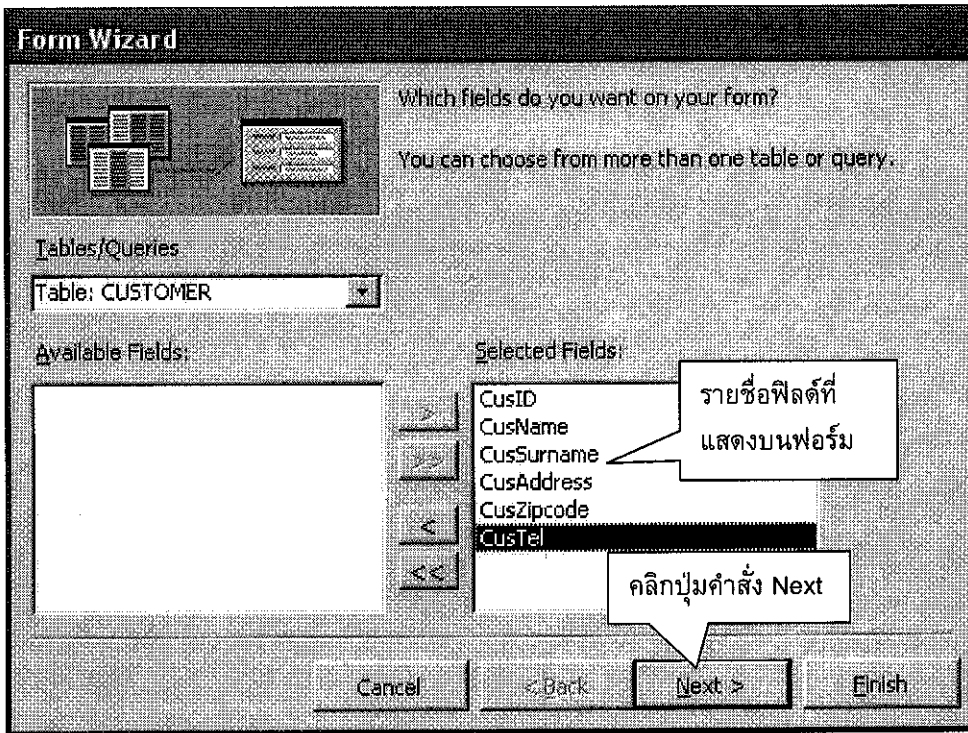
ส่วน **Tables / Queries** กำหนดชื่อ Table หรือ Query ที่ต้องการสร้าง Form

ส่วน **Available Fields** รายชื่อฟิลด์ต่างๆ ของ Table หรือ Query ที่ต้องการสร้าง Form

ส่วน **Selected Fields** รายชื่อฟิลด์ที่เลือกให้ปรากฏบน Form

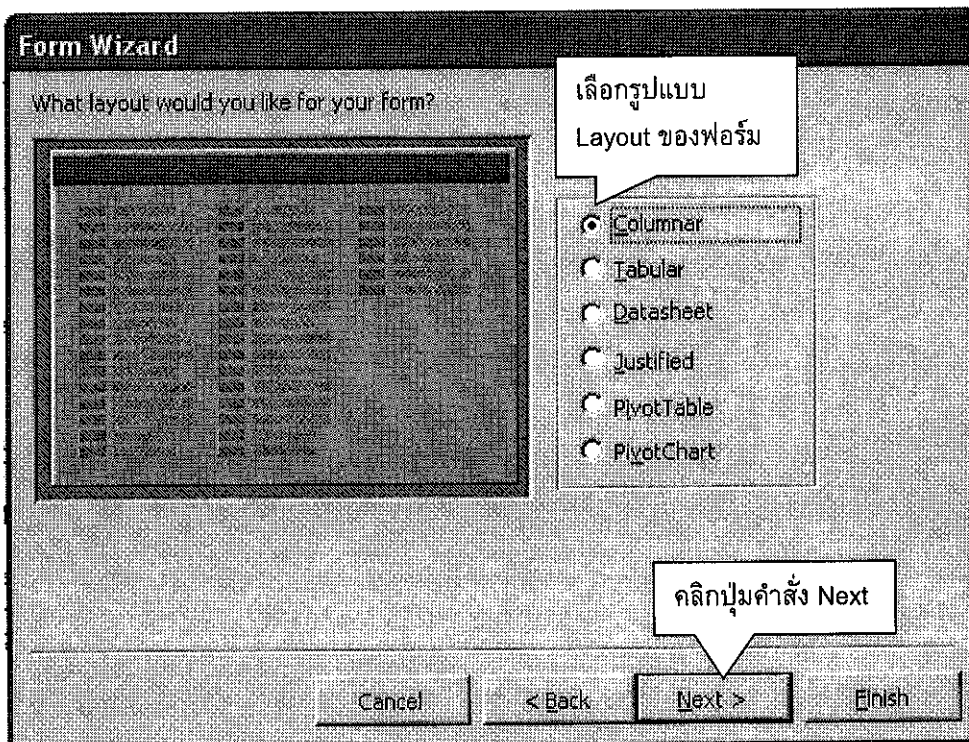
8. เลือกฟิลด์ที่ต้องการแสดงบน Form ในที่นี้ให้เลือกทุกฟิลด์ โดยคลิกที่ปุ่ม 

9. ปรากฏชื่อฟิลด์ทุกชื่อในช่อง Select Field ดังรูป

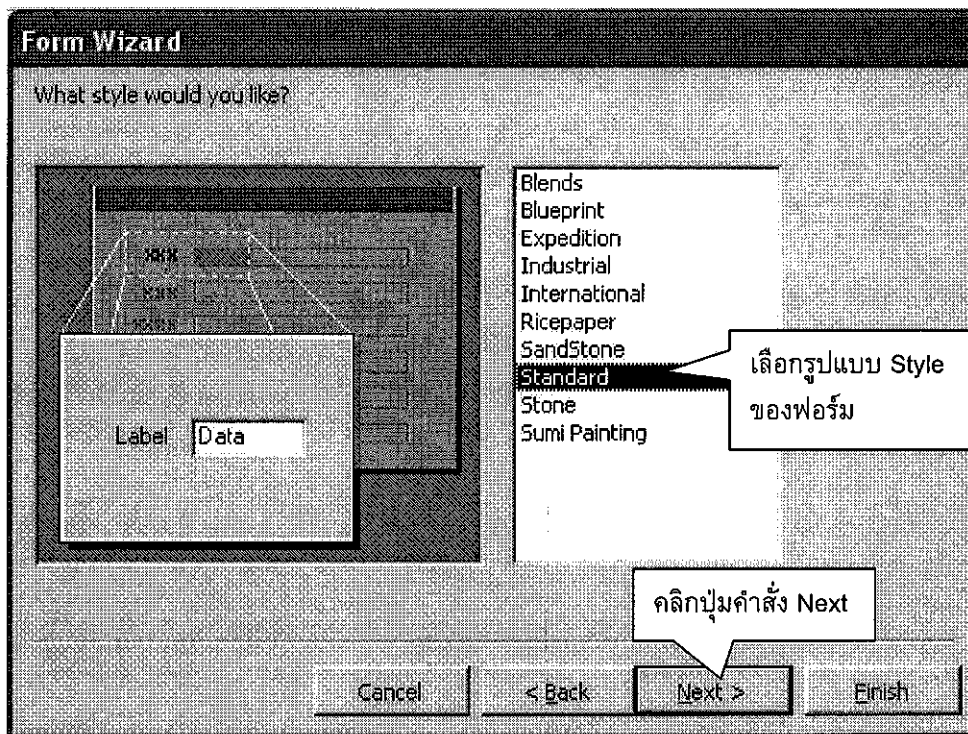


10. คลิกปุ่มคำสั่ง Next

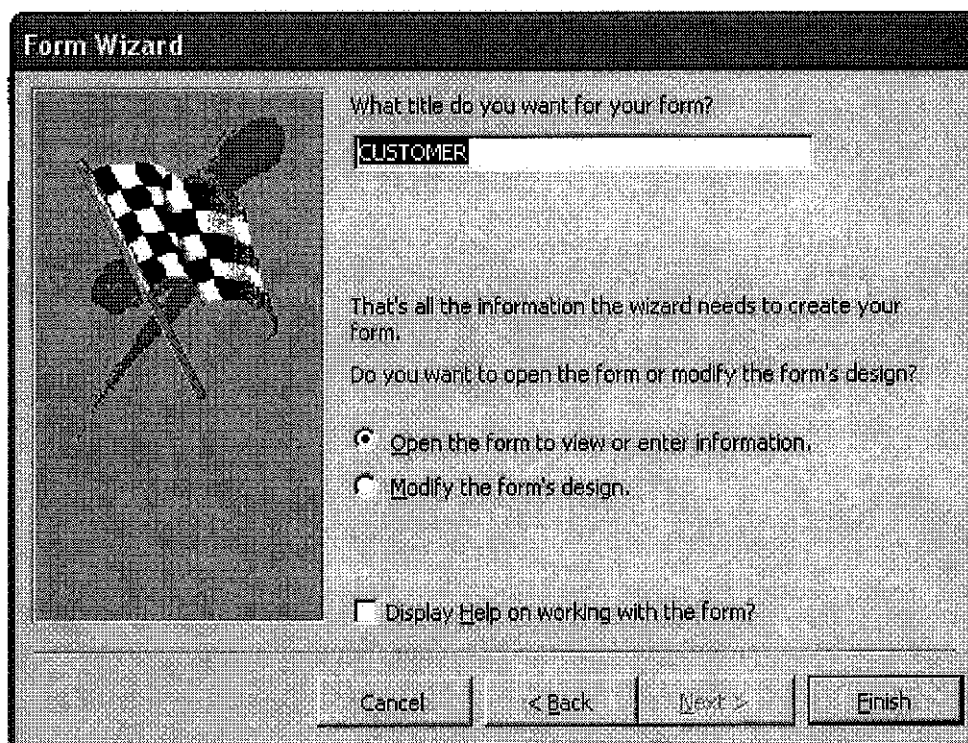
11. ปรากฏหน้าจอถัดไป เพื่อกำหนดเลือก layout ของ Form ให้นักศึกษาเลือกรูปแบบใดก็ได้ ในที่นี้เลือก แบบ Columnar ดังรูป จากนั้นให้คลิกปุ่ม Next



12. ปรากฏหน้าจอดีไป Style ของ Form ให้นักศึกษาเลือกรูปแบบใดก็ได้ ในที่นี้เลือกแบบ Standard ดังรูป จากนั้นให้คลิกปุ่ม Next



13. ปรากฏหน้าต่างให้ตั้งชื่อของ Form ให้นักศึกษาตั้งชื่อฟอร์มว่า CUSTOMER ดังรูป



โดยทั่วไปแล้วการสร้างฟอร์มด้วย Form Wizard จะตั้งชื่อฟอร์มที่สร้างขึ้นให้เหมือนกับชื่อ Table หรือ Query ที่ใช้ในการสร้างฟอร์ม

14. เลือกรูปแบบการแสดงผลฟอร์ม ซึ่งมีให้เลือก 2 รูปแบบ คือ

ประเภท	คำอธิบาย
Open the Form to view or enter information	เปิดฟอร์มที่สร้างขึ้นในหน้าต่าง Form View
Modify the form's design	เปิดฟอร์มที่สร้างขึ้นในหน้าต่าง Design View

ในที่นี้เลือกรูปแบบ Open the Form to view or enter information

15. คลิกปุ่ม Finish จะปรากฏ Form : Customer ดังรูป

The screenshot shows a Microsoft Access form window titled "CUSTOMER". The form contains several text boxes with the following data:

- รหัสลูกค้า: 001
- ชื่อ: เงิน
- นามสกุล: ดี
- ที่อยู่: 213 อ.เมือง จ.นครราชสีมา
- รหัสไปรษณีย์: 30000
- เบอร์โทรศัพท์: 0-4422-2222

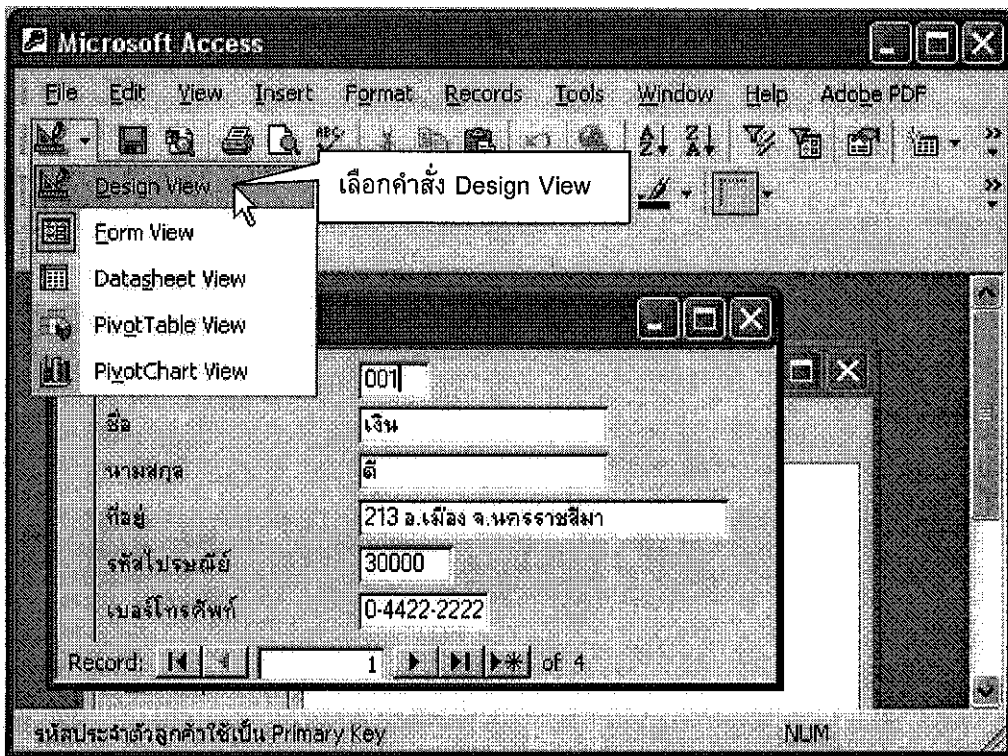
At the bottom of the form, there is a record navigation bar showing "Record: 1 of 4".

ขั้นตอนการปฏิบัติที่ 4.2

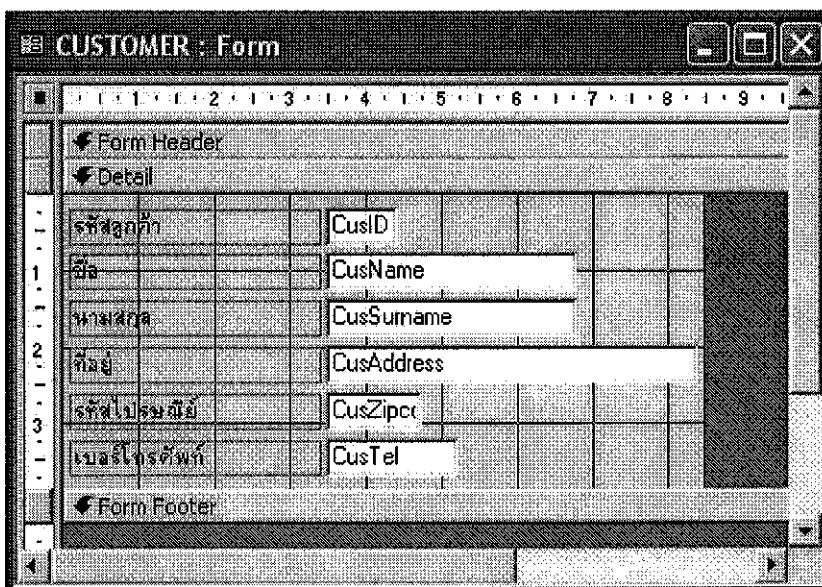
การสร้างป้ายคำอธิบายด้วยคอนโทรล Label

ขั้นตอนการปฏิบัติ มีดังนี้

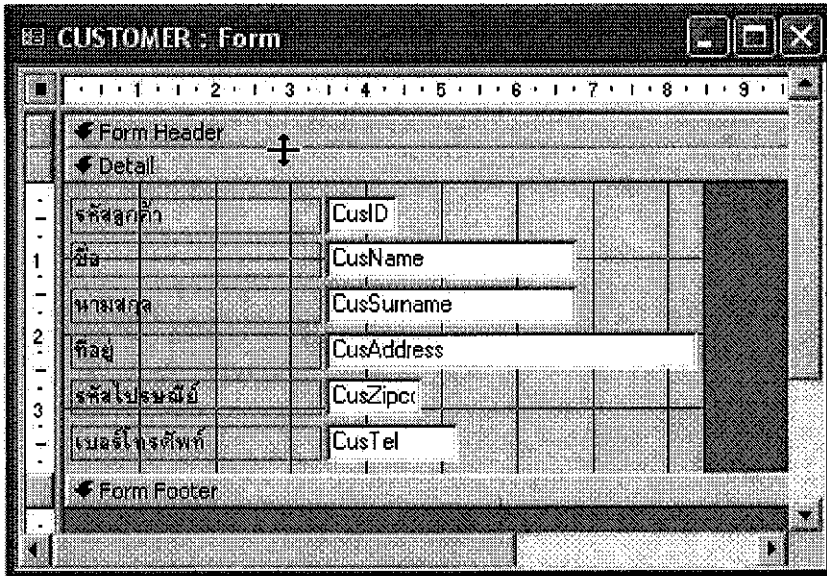
1. จากหน้าต่าง Form คลิกที่ Icon  จากนั้นเลือกคำสั่ง Design View ดังรูป



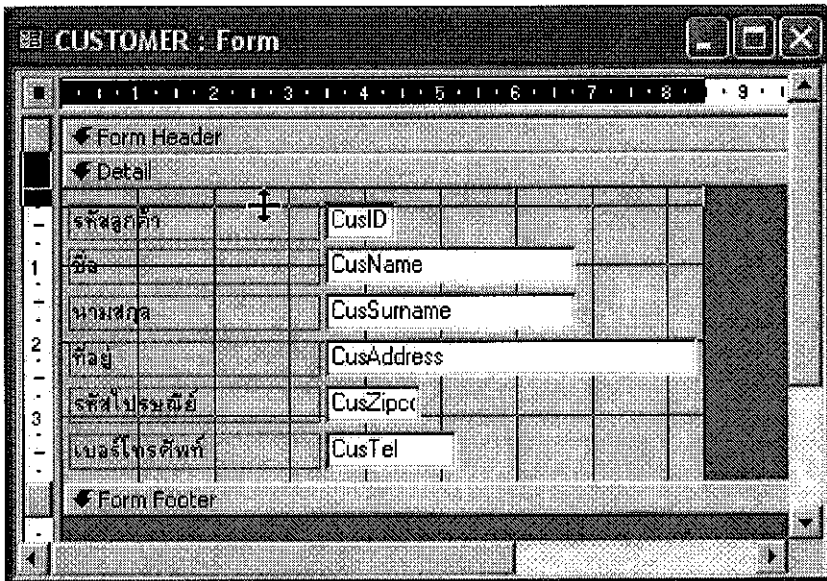
2. ปรากฏหน้าต่าง Design Form ดังรูป



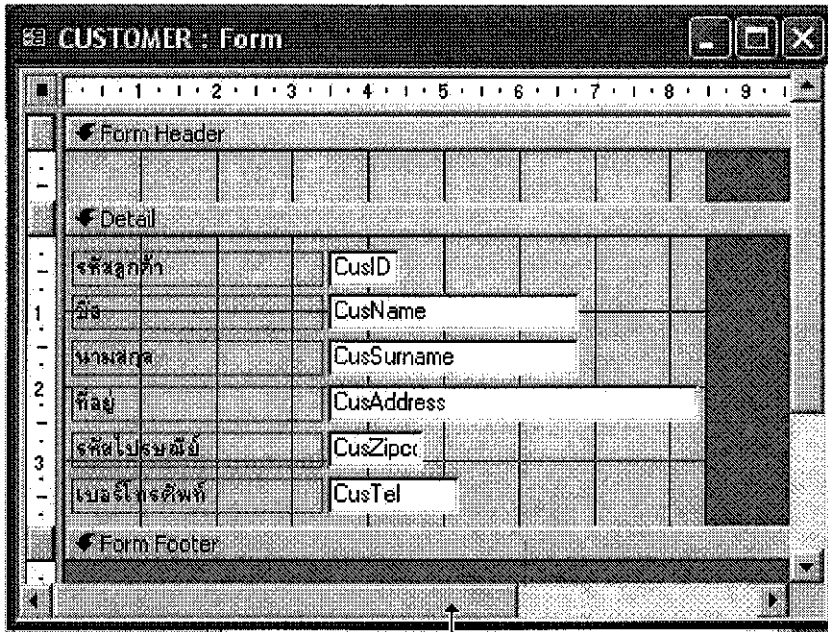
3. วางเมาส์ไว้ระหว่างส่วน Form Header และ Detail ให้เมาส์เปลี่ยนเป็นรูป  ดังรูป




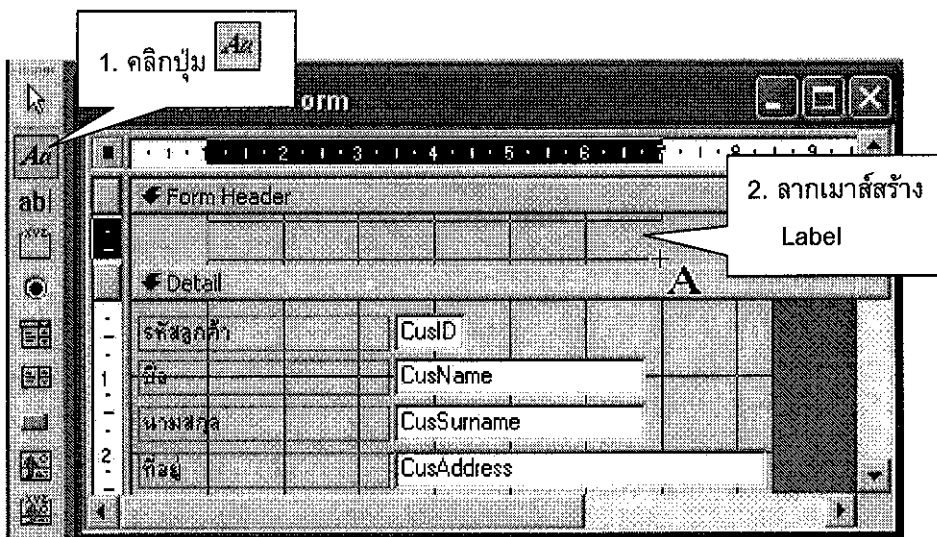
4. คลิกเมาส์ค้างไว้ แล้วลากเมาส์ลงด้านล่าง เพื่อขยายพื้นที่ของส่วน Form Header ดังรูป



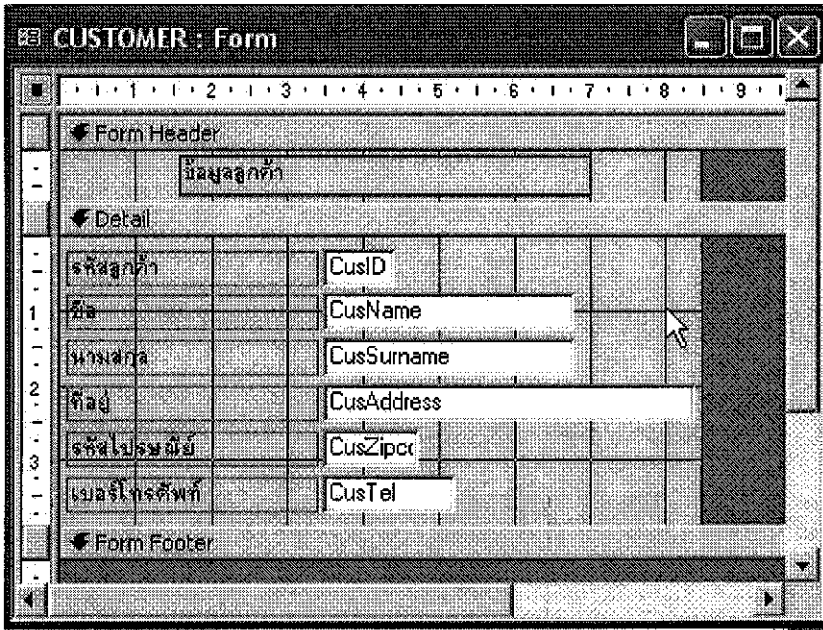
5. ปล่อยเมาส์ออก จะปรากฏพื้นที่ส่วน Form Header ดังรูป




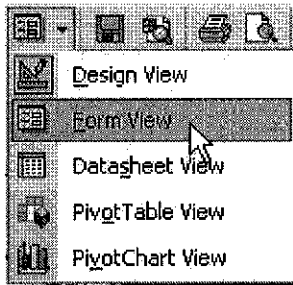
6. คลิกปุ่ม  บนแถบเครื่องมือ (Toolbox) เมาส์จะเปลี่ยนเป็นรูป 'A' จากนั้นให้คลิกวางบนพื้นที่ Form Header ที่ต้องการสร้างคอนโทรล Label แล้วลากเมาส์ให้เกิดพื้นที่สี่เหลี่ยม ดังรูป



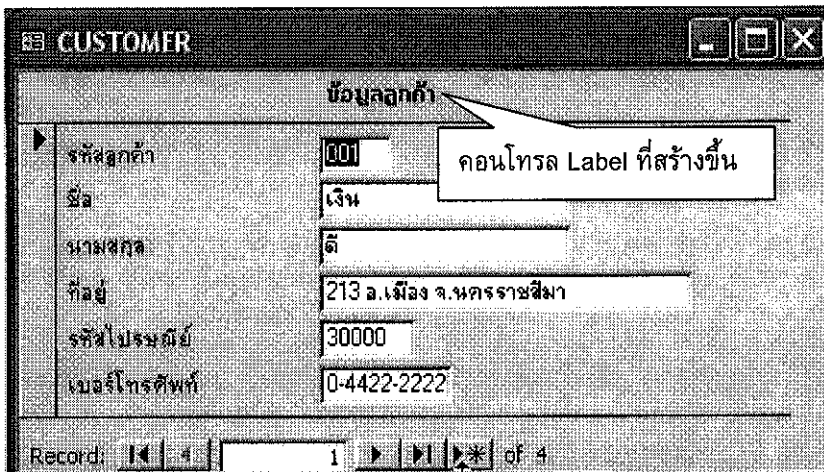
7. เมื่อปล่อยเมาส์ จะปรากฏกรอบพื้นที่ตามขนาดที่กำหนดไว้ ให้พิมพ์ข้อความที่ต้องการลงไป ในที่นี้พิมพ์ข้อความ "ข้อมูลลูกค้า" ดังรูป



8. คลิกที่ Icon  จากนั้นเลือกคำสั่ง Form View ดังรูป



9. ปรากฏหน้าต่าง Form : CUSTOMER ในมุมมอง Form View ดังรูป



การสร้างปุ่มคำสั่งด้วยคอนโทรล Command Button

Command Button เป็นคอนโทรลที่ใช้ในการสร้างปุ่มคำสั่งเพื่อควบคุมการทำงานของฟอร์ม ให้มีการทำงานต่างๆ อย่างมีประสิทธิภาพและเพิ่มความสะดวกในการใช้งาน เช่น การเปิด/ปิดฟอร์ม การพิมพ์ฟอร์ม หรือการสั่งให้เปิดคิวรีเพื่อค้นหาข้อมูล เป็นต้น ซึ่งความสามารถของคอนโทรล ประเภทนี้จำแนกได้เป็น 6 กลุ่ม (Categories) ดังนี้

1. กลุ่มคำสั่งที่ทำงานกับเรคคอร์ด (Record Navigation)

คำสั่ง (Action)	คำอธิบาย
Find Next	ค้นหาเรคคอร์ดที่ตรงกับเงื่อนไขที่กำหนดไว้
Find Record	ค้นหาเรคคอร์ดที่ตรงกับเงื่อนไข โดยเปิดหน้าต่างค้นหา (Find)
Go To First Record	แสดงเรคคอร์ดแรก
Go To Last Record	แสดงเรคคอร์ดสุดท้าย
Go To Next Record	แสดงเรคคอร์ดถัดไป
Go To Previous Record	แสดงเรคคอร์ดก่อนหน้า

2. กลุ่มคำสั่งที่จัดการเรคคอร์ด (Record Operations)

คำสั่ง (Action)	คำอธิบาย
Add New Record	เพิ่มเรคคอร์ดใหม่
Delete Record	ลบเรคคอร์ดปัจจุบัน
Duplicate Record	ทำสำเนาเรคคอร์ดปัจจุบัน
Print Record	พิมพ์เรคคอร์ดปัจจุบัน
Save Record	บันทึกเรคคอร์ดปัจจุบันลงตาราง
Undo Record	ยกเลิกการแก้ไขเรคคอร์ดปัจจุบัน

3. กลุ่มคำสั่งที่จัดการฟอร์ม (Form Operations)

คำสั่ง (Action)	คำอธิบาย
Apply Form Filter	กรองข้อมูลตามสิ่งที่เลือก
Close Form	ปิดฟอร์มที่ใช้งานอยู่
Edit Form Filter	กำหนดการกรองข้อมูล
Open Form	เปิดฟอร์มที่กำหนด
Open Page	เปิด Access Page ที่กำหนด
Print a Form	พิมพ์ฟอร์มที่กำหนด
Print Current Form	พิมพ์ฟอร์มปัจจุบัน
Refresh Form Data	ปรับปรุงข้อมูลในฟอร์ม

4. กลุ่มคำสั่งที่จัดการรายงาน (Report Operations)

คำสั่ง (Action)	คำอธิบาย
Mail Report	ส่งรายงานทางอีเมล
Preview Report	แสดงรายงานก่อนพิมพ์
Print Report	พิมพ์รายงาน
Send Report To File	พิมพ์รายงานออกเป็นไฟล์เอกสาร

5. กลุ่มคำสั่งที่ทำงานกับโปรแกรม (Applications)

คำสั่ง (Action)	คำอธิบาย
Quit Application	ออกจากโปรแกรมปัจจุบัน
Run Application	เปิดโปรแกรมที่กำหนด
Run Ms Excel	เปิดโปรแกรม Ms Excel
Run Ms Word	เปิดโปรแกรม Ms Word
Run Notepad	เปิดโปรแกรม Notepad

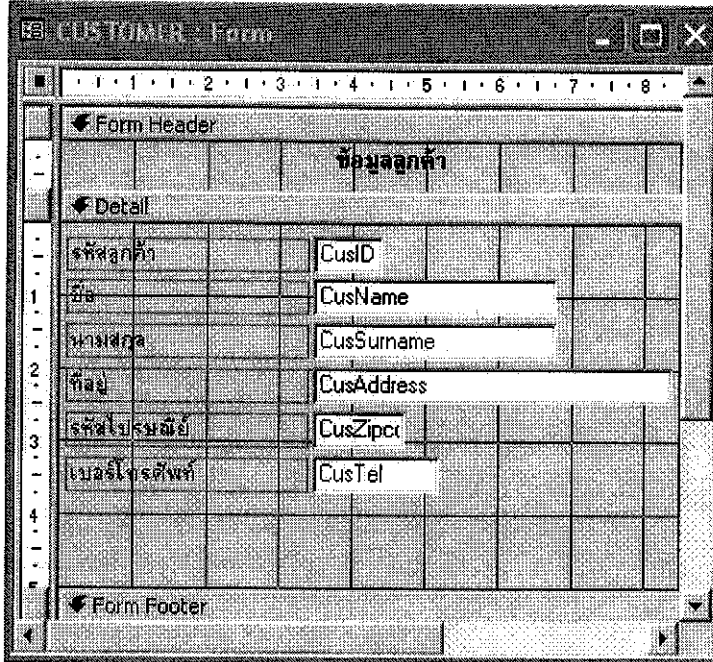
6. กลุ่มคำสั่งที่ทำงานเบ็ดเตล็ด (Miscellaneous)



คำสั่ง (Action)	คำอธิบาย
Auto Dialer	ส่งต่ออินเทอร์เน็ต
Print Table	พิมพ์ Table ที่กำหนด
Run Marco	ส่งทำงาน Marco ที่กำหนด
Run Query	ส่งทำงาน Query ที่กำหนด

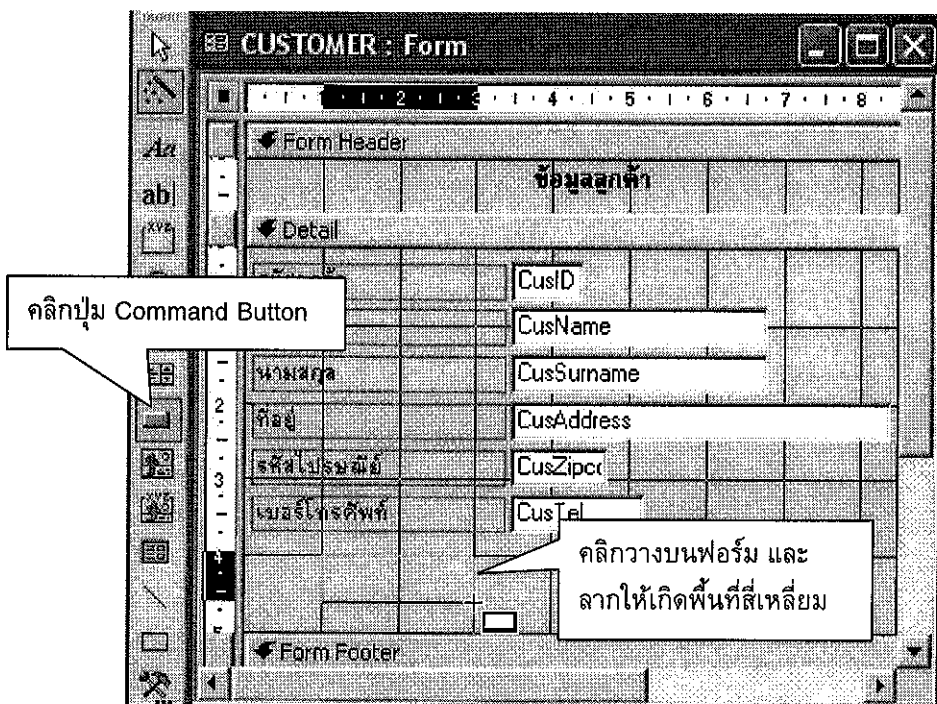
คำสั่ง : จงสร้างปุ่มคำสั่ง (Command Button) เพื่อใช้ในการเพิ่มข้อมูล

ขั้นตอนการปฏิบัติ มีดังนี้

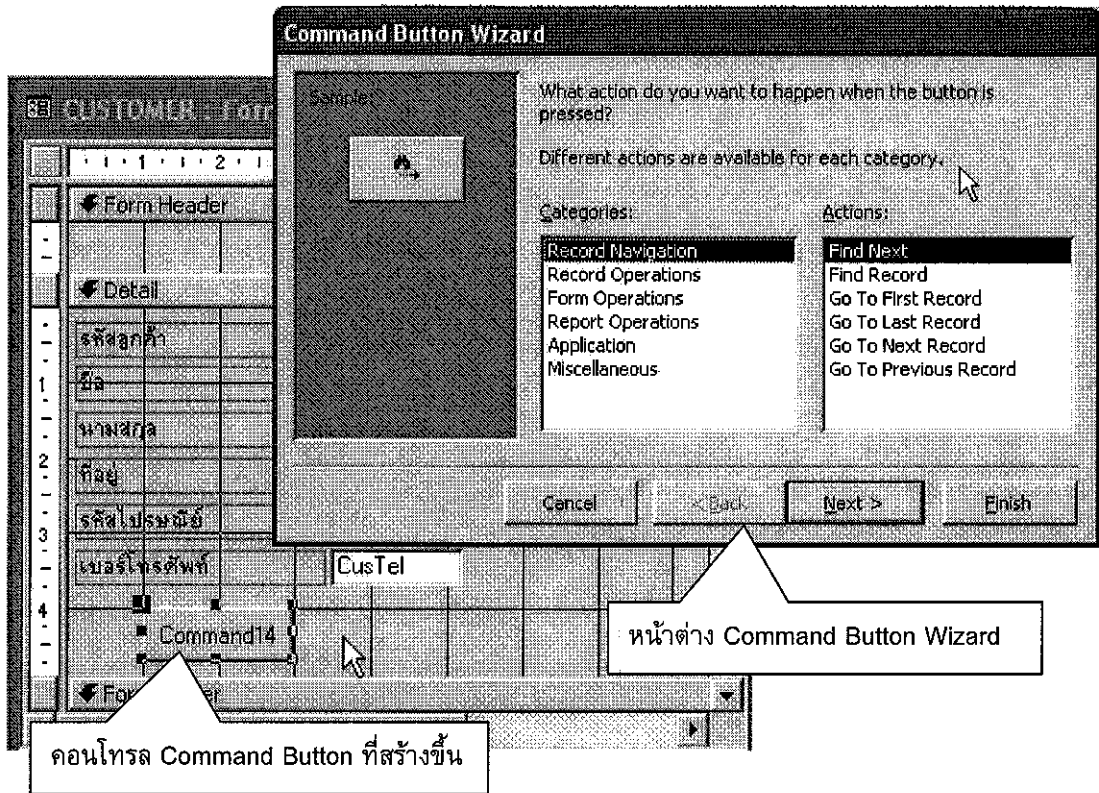
1. จากหน้าต่าง Form : CUSTOMER ในมุมมอง Design View ดังรูป



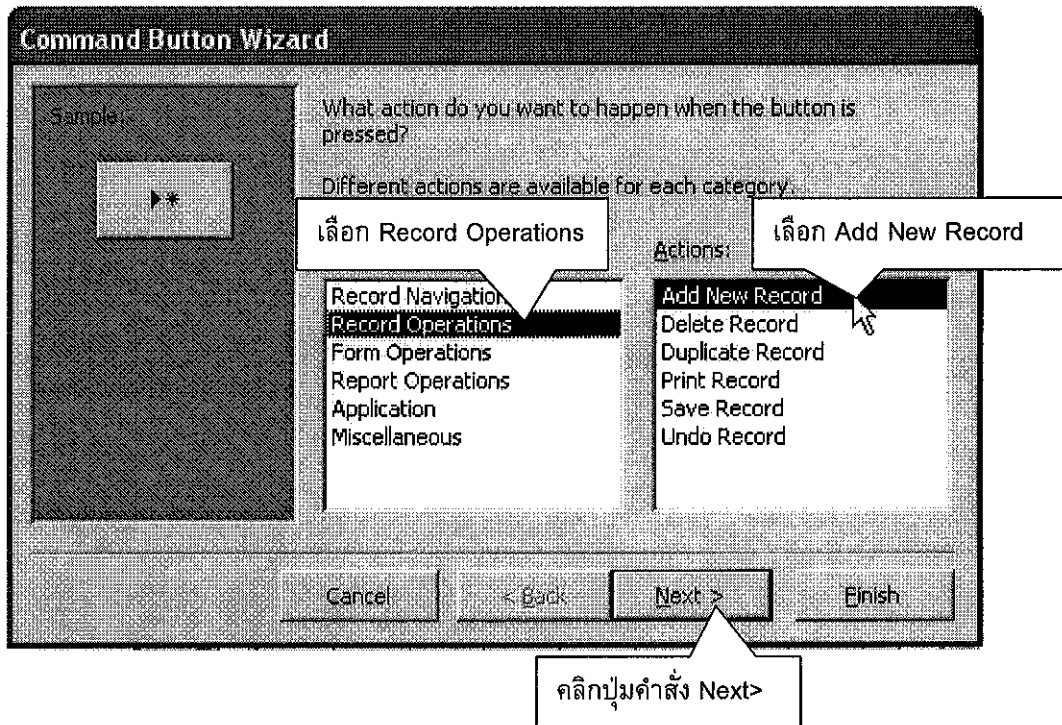
2. คลิกปุ่ม  บนแถบเครื่องมือ (Toolbox) เมาส์จะเปลี่ยนเป็นรูป  จากนั้นให้คลิกวางบนพื้นที่ Detail ที่ต้องการสร้างคอนโทรล Command Button แล้วลากเมาส์ให้เกิดพื้นที่สี่เหลี่ยม ดังรูป



3. ปรากฏคอนโทรล Command Button ชื่อ Command 14 (โปรแกรมตั้งชื่ออัตโนมัติ) บนฟอร์ม พร้อมหน้าต่าง Command Button Wizard ดังรูป



4. คลิกเลือก Categories : Record Operations และ Actions : Add New Record จากนั้นคลิกปุ่มคำสั่ง Next ดังรูป

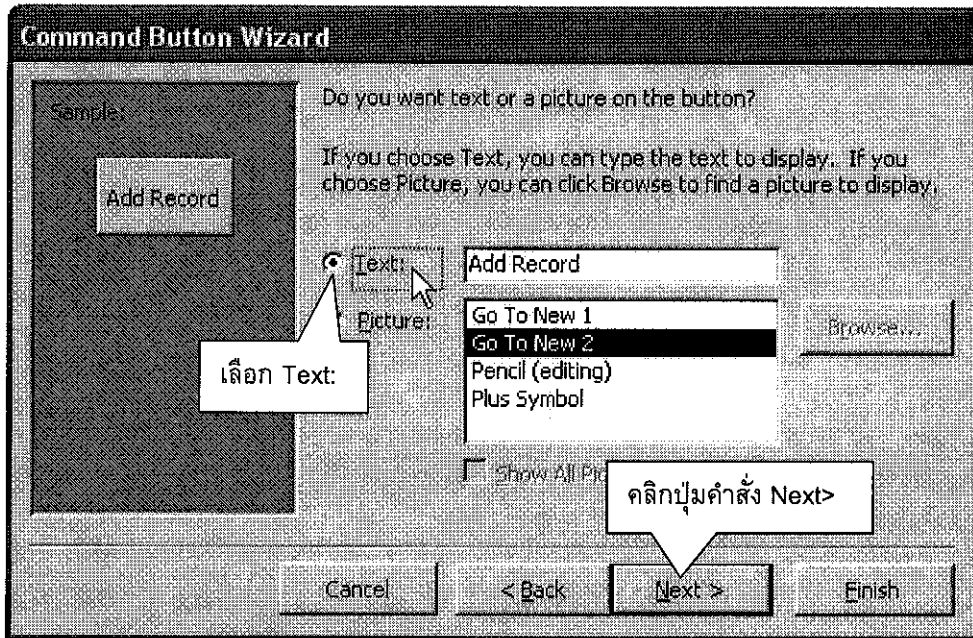


5. กำหนดรูปแบบการแสดงผลของปุ่มคำสั่ง ซึ่งมีด้วยกัน 2 รูปแบบ คือ

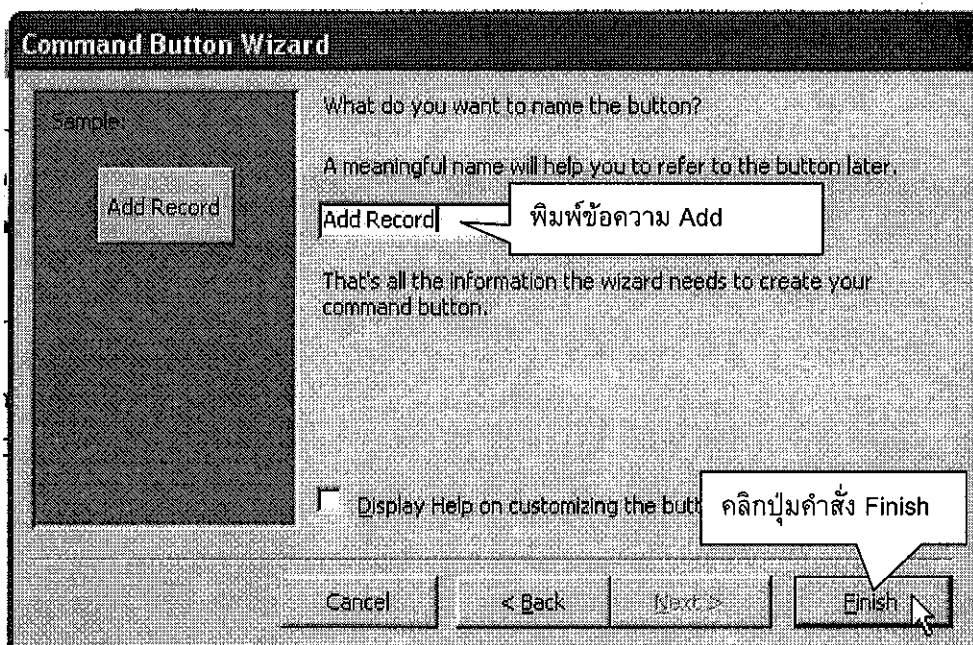
1) แบบข้อความ (Text) เป็นการแสดงข้อความบนปุ่มคำสั่ง ผู้ใช้สามารถกำหนดข้อความเองได้ โดยการพิมพ์ลงในช่องที่กำหนด

2) แบบรูปภาพ (Picture) เป็นการแสดงรูปภาพบนปุ่มคำสั่ง ผู้ใช้สามารถใช้รูปภาพที่โปรแกรมกำหนดไว้ให้ หรือกำหนดรูปภาพขึ้นเองได้ โดยคลิกที่ปุ่ม Browser

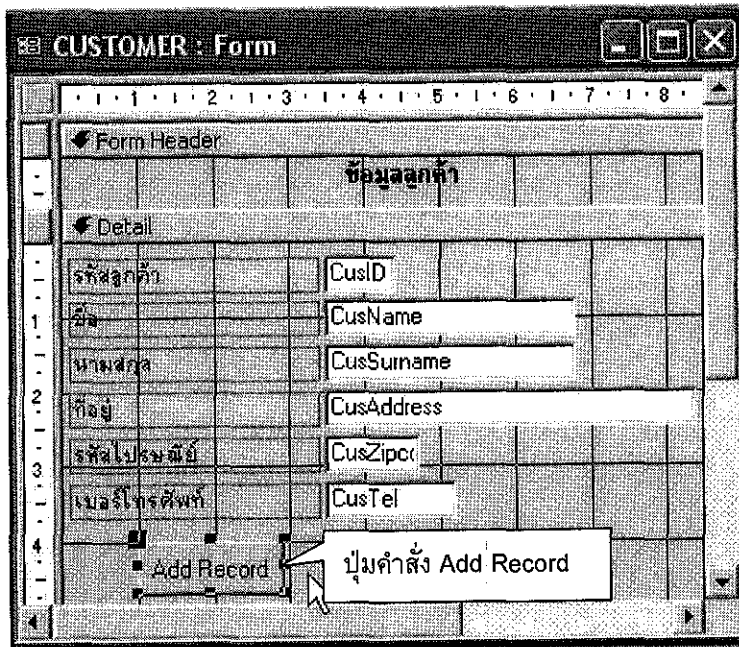
ให้เลือกรูปแบบ Text จากนั้นคลิกปุ่มคำสั่ง Next ดังรูป




6. ตั้งชื่อปุ่มคำสั่ง โดยทั่วไปโปรแกรมจะตั้งชื่อปุ่มคำสั่งมาให้โดยอัตโนมัติ แต่ในที่นี้ให้ตั้งชื่อปุ่มคำสั่งเป็นชื่อ "Add Record" เนื่องจากจะได้สื่อความหมายในการใช้งาน จากนั้นคลิกปุ่มคำสั่ง Finish ดังรูป

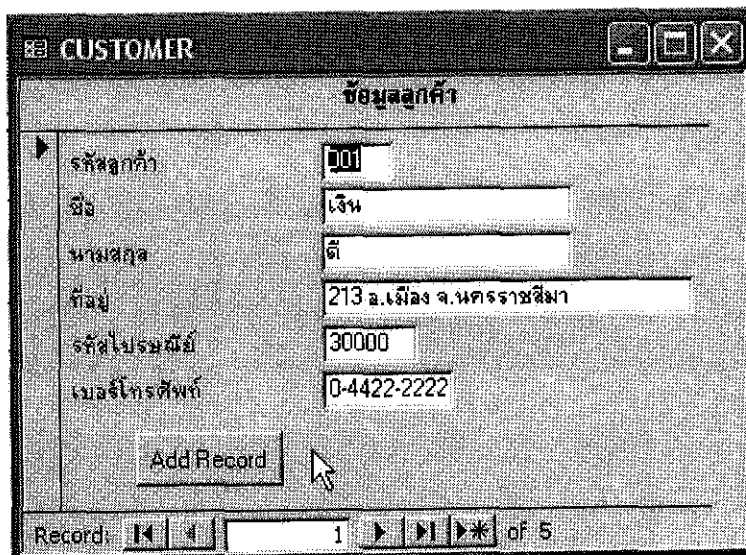


7. ปรากฏปุ่มคำสั่ง Add Record ที่สร้างขึ้นใหม่บน Form ดังรูป



8. คลิกที่ Icon  จากนั้นเลือกคำสั่ง Form View ดังรูป

9. ปรากฏหน้าต่าง Form : CUSTOMER ในมุมมอง Form View ดังรูป



10. ทดสอบปุ่มคำสั่ง Add Record เมื่อคลิกปุ่มคำสั่ง Add Record ที่ช่องรับค่าข้อมูลจะปรากฏเป็นช่องว่าง เพื่อพร้อมรับการกรอกข้อมูลลงสู่ตาราง ดังรูป

The screenshot shows a window titled 'CUSTOMER' with a form titled 'ข้อมูลลูกค้า'. The form has the following fields:

- รหัสลูกค้า (Customer ID): empty
- ชื่อ (Name): empty
- นามสกุล (Surname): empty
- ที่อยู่ (Address): empty
- รหัสไปรษณีย์ (Zip Code): 30000
- เบอร์โทรศัพท์ (Telephone): empty

 Below the fields is an 'Add Record' button. At the bottom of the window, there is a record navigation bar that says 'Record: 6 of 6'.

คำสั่ง : จะสร้างปุ่มคำสั่ง (Command Button) เพื่อใช้ในการปิดหน้าต่างฟอร์ม
CUSTOMER



ขั้นตอนการปฏิบัติ มีดังนี้

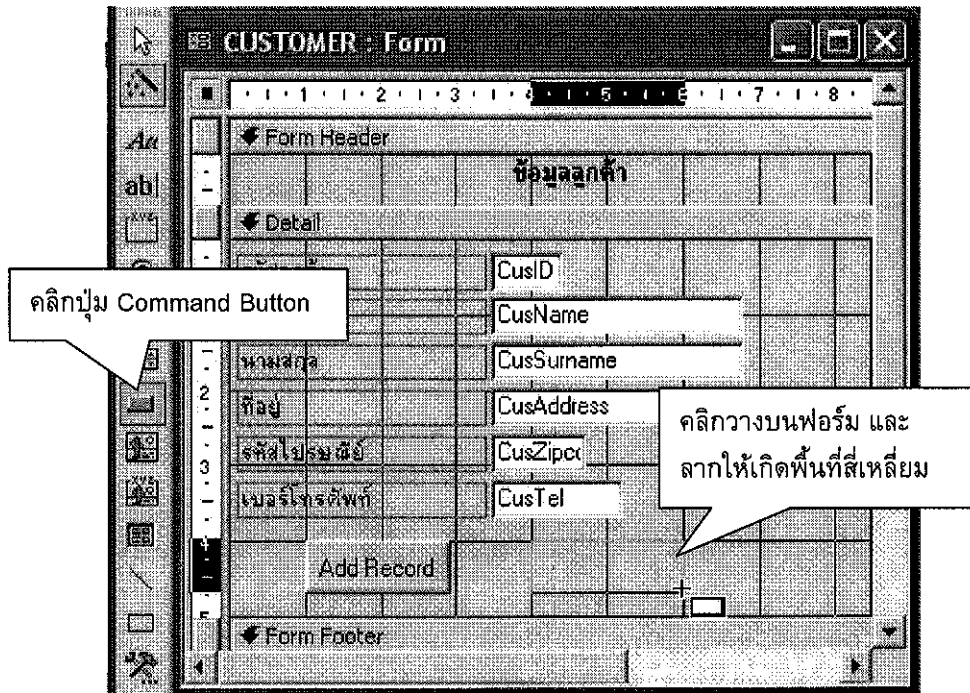
1. จากหน้าต่าง Form : CUSTOMER ในมุมมอง Design View ดังรูป

The screenshot shows the 'CUSTOMER : Form' in Design View. The form is divided into two sections:

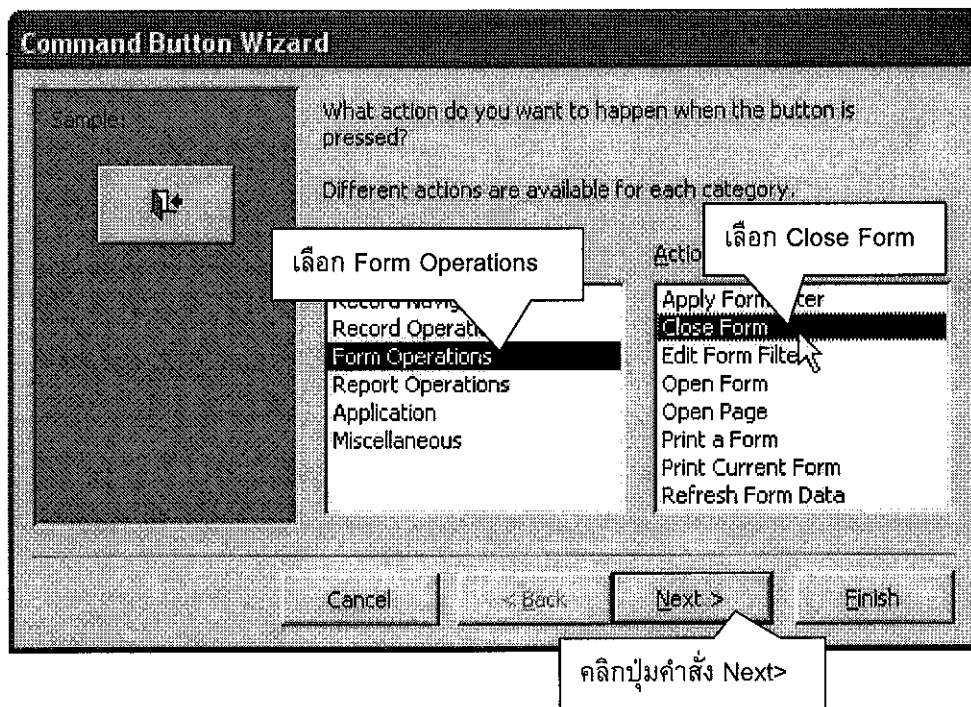
- Form Header:** Contains the title 'ข้อมูลลูกค้า'.
- Detail:** Contains the following fields:
 - รหัสลูกค้า (Customer ID): CusID
 - ชื่อ (Name): CusName
 - นามสกุล (Surname): CusSurname
 - ที่อยู่ (Address): CusAddress
 - รหัสไปรษณีย์ (Zip Code): CusZipcode
 - เบอร์โทรศัพท์ (Telephone): CusTel

 At the bottom of the form, there is an 'Add Record' button. The design view includes a grid and a ruler at the top.

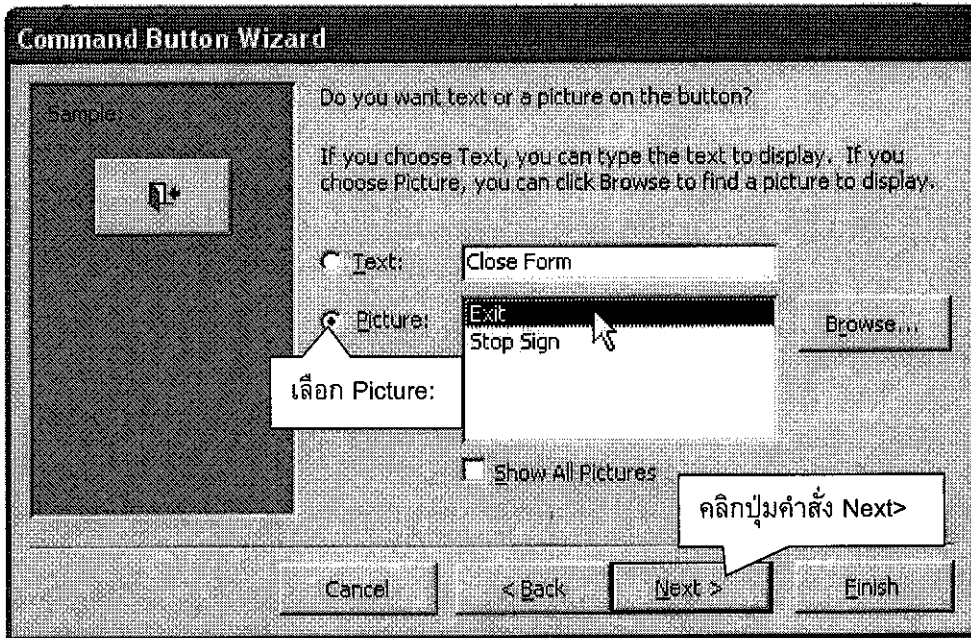
2. คลิกปุ่ม  บนแถบเครื่องมือ (Toolbox) เมาส์จะเปลี่ยนเป็นรูป  จากนั้นให้คลิกวางบนพื้นที่ Detail ที่ต้องการสร้างคอนโทรล Command Button แล้วลากเมาส์ให้เกิดพื้นที่สี่เหลี่ยม ดังรูป



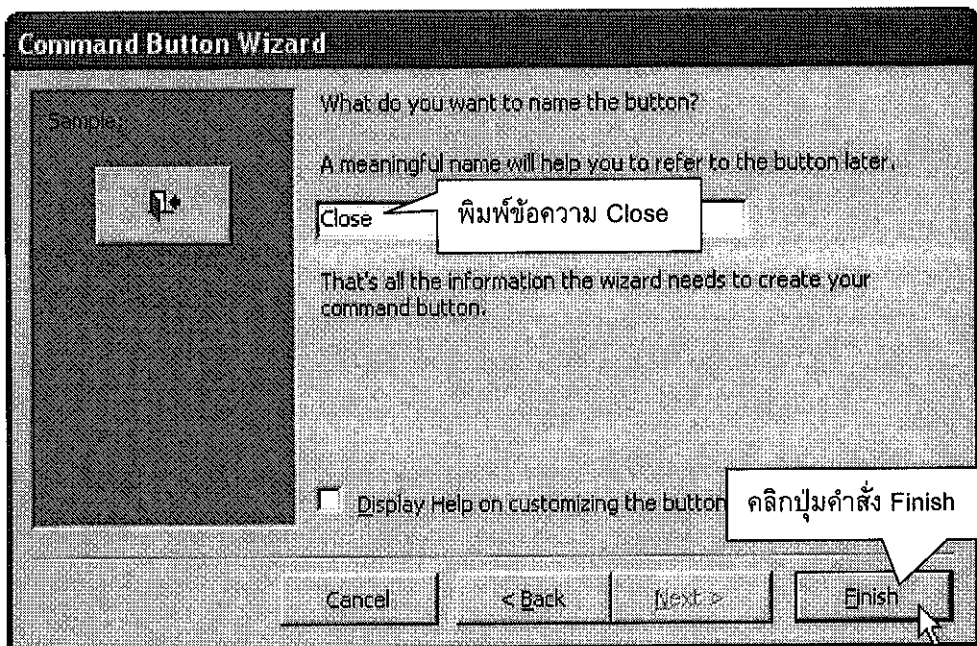
3. ปรากฏคอนโทรล Command บนฟอร์ม พร้อมหน้าต่าง Command Button Wizard
 4. คลิกเลือก Categories : Form Operations และ Actions : Close Form จากนั้นคลิกปุ่มคำสั่ง Next ดังรูป



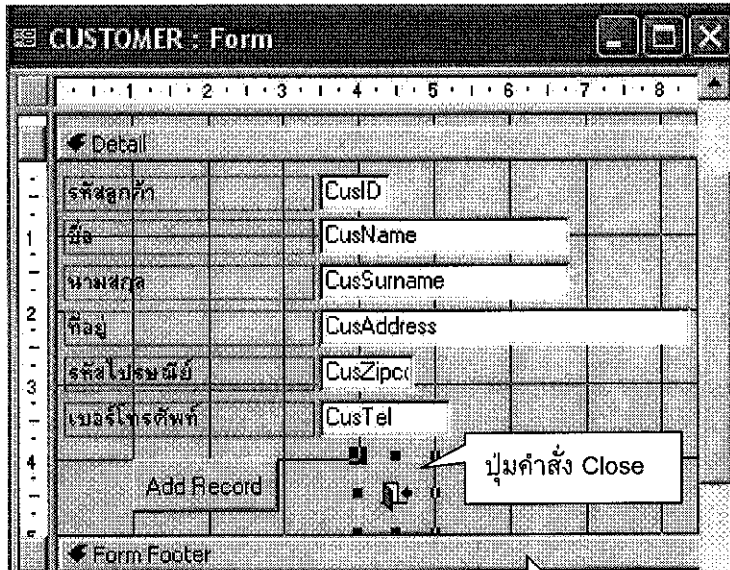
5. คลิกเลือกรูปแบบ Text จากนั้นคลิกปุ่มคำสั่ง Next ดังรูป




6. ตั้งชื่อปุ่มคำสั่ง โดยทั่วไปโปรแกรมจะตั้งชื่อปุ่มคำสั่งมาให้โดยอัตโนมัติ แต่ในที่นี้ให้ตั้งชื่อปุ่มคำสั่งเป็นชื่อ "Close" เนื่องจากจะได้สื่อความหมายในการทำงาน จากนั้นคลิกปุ่มคำสั่ง Finish ดังรูป

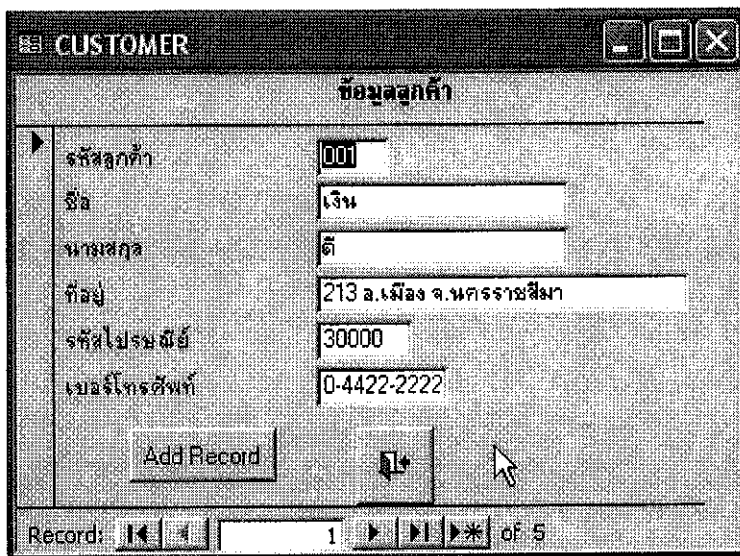


7. ปรากฏปุ่มคำสั่ง Add Record ที่สร้างขึ้นใหม่บน Form ดังรูป



8. คลิกที่ Icon  จากนั้นเลือกคำสั่ง Form View ดังรูป

9. ปรากฏหน้าต่าง Form : CUSTOMER ในมุมมอง Form View ดังรูป



10. ทดสอบปุ่มคำสั่ง Close เมื่อคลิกปุ่มคำสั่ง Close โปรแกรมจะปิดฟอร์ม CUSTOMER ลงโดยอัตโนมัติ

คำสั่ง

1. ให้นักศึกษาสร้างฟอร์มให้กับตาราง EMPLOYER, PRODUCT, ORDER และ ORDER_Detail โดยเลือกรูปแบบต่างๆ กัน เพื่อดูความแตกต่างกันของแต่ละฟอร์ม แล้วกรอกข้อมูลเพิ่มเติมตารางละ 2-3 เรคคอร์ด (กำหนดข้อมูลด้วยตนเอง)
2. ในแต่ละฟอร์มที่สร้างขึ้นในข้อที่ 1 ให้สร้างปุ่มคำสั่งดังนี้
 - ปุ่มคำสั่งเพิ่มเรคคอร์ดใหม่ (Add New Record)
 - ปุ่มคำสั่งลบเรคคอร์ด (Delete Record)
 - ปุ่มคำสั่งปิดฟอร์ม (Close Form)
3. ให้นักศึกษาสร้างฟอร์มจาก Query ชื่อ Q_Order1 โดยตั้งชื่อฟอร์มว่า F_Order1 พร้อมสร้างปุ่มคำสั่งพิมพ์ฟอร์ม (Print a Form)

๗ ปฏิบัติการที่ 5 การสร้างรายงาน

วัตถุประสงค์ นักศึกษาสามารถ

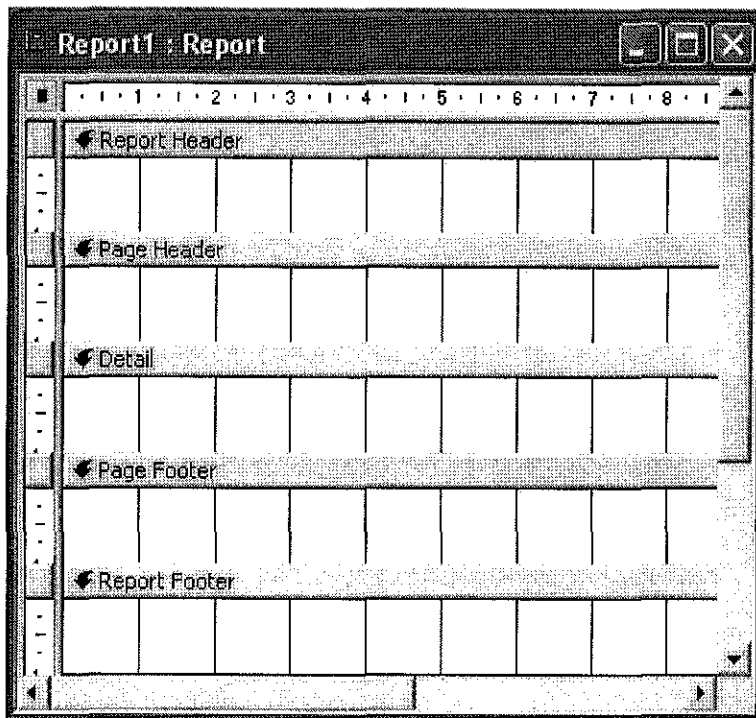
1. มีความรู้เกี่ยวกับการสร้างรายงาน
2. สร้างรายงานเพื่อใช้ในการนำเสนอข้อมูลผ่านทางเครื่องพิมพ์ได้

Report : รายงาน

รายงาน (Report) เป็นองค์ประกอบหนึ่งของโปรแกรม Microsoft Access ทำหน้าที่นำเสนอข้อมูลในรูปแบบของรายงาน ซึ่งนำเสนอข้อมูลได้หลายรูปแบบ เช่น การจัดกลุ่มข้อมูลตามต้องการ การคำนวณและรวมข้อมูลทั้งด้านแถวและคอลัมน์ การสร้างแผนภูมินำเสนอข้อมูล และการทำป้ายชื่อปิดหน้าของจดหมาย เป็นต้น

องค์ประกอบของ Report

Report ของ Microsoft Access มีองค์ประกอบดังนี้



องค์ประกอบ	คำอธิบาย
Report Header	พื้นที่ส่วนหัวของรายงาน รายละเอียดต่างๆ ที่อยู่ในส่วนนี้จะถูกแสดงเฉพาะรายงานหน้าแรกเท่านั้น
Page Header	พื้นที่ส่วนหัวของกระดาษ รายละเอียดต่างๆ ที่อยู่ในส่วนนี้จะถูกแสดงที่บนหัวกระดาษของรายงานทุกหน้า
Detail	พื้นที่สำหรับแสดงรายละเอียดของข้อมูล และถือเป็นส่วนสำคัญของรายงาน
Page Footer	พื้นที่ส่วนท้ายของกระดาษ รายละเอียดต่างๆ ที่อยู่ในส่วนนี้จะถูกแสดงที่บนหัวกระดาษของรายงานทุกหน้า
Report Footer	พื้นที่ส่วนท้ายของรายงาน รายละเอียดต่างๆ ที่อยู่ในส่วนนี้จะถูกแสดงเฉพาะรายงานหน้าแรกเท่านั้น ส่วนใหญ่จะแสดงผลรวมของข้อมูลภายในส่วน Detail ของกลุ่มข้อมูลทั้งหมดในรายงาน

มุมมองในการสร้าง Query

Microsoft Access 2002 กำหนดมุมมองในการสร้าง Report 3 มุมมอง ดังนี้

มุมมอง	คำอธิบาย
Design View	มุมมองที่ใช้สำหรับออกแบบและสร้าง Report ซึ่งมีวิธีการใช้งานเหมือนกับมุมมอง Form Design ของ Form
Print Preview	มุมมองที่ใช้สำหรับแสดงรายงานในรูปแบบภาพก่อนพิมพ์ ซึ่งสามารถกำหนดให้แสดงกี่หลายหน้า รวมถึงกำหนดขนาดในการแสดงให้แสดงเป็นภาพใหญ่หรือภาพเล็กได้ตามต้องการ
Layout View	มุมมองที่ใช้สำหรับแสดงภาพก่อนพิมพ์เช่นเดียวกับแบบ Print Preview แต่จะแสดงให้เห็นเพียงหน้าเดียวเท่านั้น และแสดงในลักษณะที่พอดีกับหน้าจอเท่านั้น

ประเภทของ Report

Report สามารถแบ่งประเภทได้ดังนี้

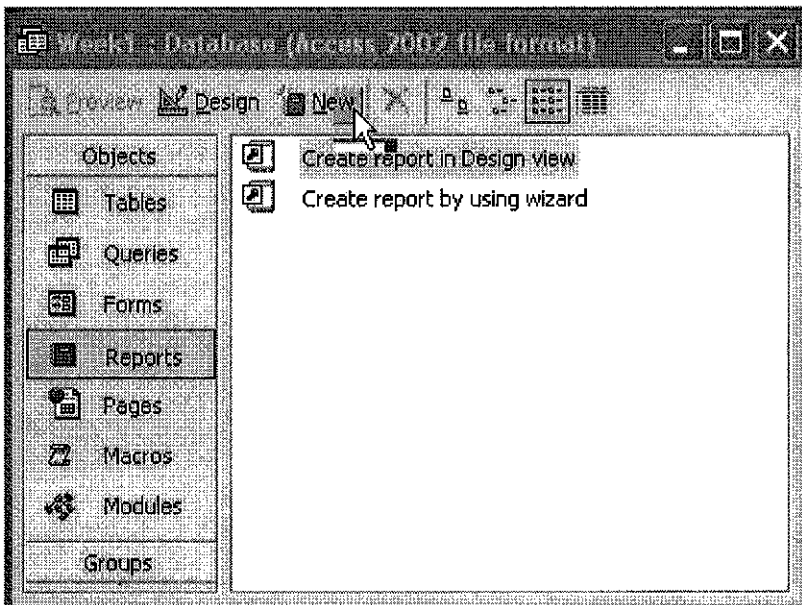
ประเภท	ความหมาย
Design View	การสร้างรายงานด้วยตนเอง โดยใช้เครื่องมือหรือที่เรียกว่า "คอนโทรล" (Control) ในลักษณะต่างๆ พร้อมระบุเงื่อนไขต่างๆ ที่ต้องการนำเสนอในรายงานตามความต้องการ
Report Wizard	การสร้างรายงานโดยการเรียกใช้วิซาร์ด (Wizard) ซึ่งโปรแกรมจะถามคำถามเกี่ยวกับลักษณะของรายงานที่ต้องการ พร้อมแสดงภาพตัวอย่างประกอบทุกขั้นตอน แล้วสร้างออกมาเป็นรายงานตามที่ต้องการ และยังสร้างรายงานที่มีการจัดกลุ่มข้อมูล เรียงลำดับข้อมูล และการสรุปผลรวมข้อมูล นอกจากนี้ยังสามารถสร้างรายงานในลักษณะพิเศษอื่นๆ เช่น รายงานในรูปแบบของแผนภูมิ (Chart) หรือป้ายชื่อ (Label)

ประเภท	ความหมาย
AutoReport : Columnar	การสร้างรายงานแบบง่าย โดยสามารถกำหนดแหล่งที่มาของข้อมูลได้จากตารางหรือคิวรี ซึ่งรูปแบบจะเหมือนกับการสร้าง Form โดยใช้ AutoForm : Coulumnar โดยแต่ละหน้าของรายงานจะแสดงข้อมูลแต่ละเรคคอร์ด โดยเรียงฟิลด์ข้อมูลจากบนลงล่างในแนวแถว ตามลำดับที่กำหนดไว้ในโครงสร้างของ Table หรือ Query
AutoReport : Tabular	การสร้างรายงานแบบง่าย โดยสามารถกำหนดแหล่งที่มาของข้อมูลได้จากตารางหรือคิวรี ซึ่งรูปแบบจะเหมือนกับการสร้าง Form โดยใช้ AutoForm : Tabular โดยแต่ละหน้าของรายงานจะแสดงข้อมูลแต่ละเรคคอร์ด โดยเรียงฟิลด์ข้อมูลจากซ้ายไปขวา ตามลำดับที่กำหนดไว้ในโครงสร้างของ Table หรือ Query
Chart Wizard	การสร้างรายงานที่แสดงข้อมูลในลักษณะแผนภูมิแบบต่างๆ
Label Wizard	การสร้างรายงานที่แสดงข้อมูลที่ต้องการพิมพ์ฉลาก เช่น ชื่อและที่อยู่ของลูกค้า เป็นต้น

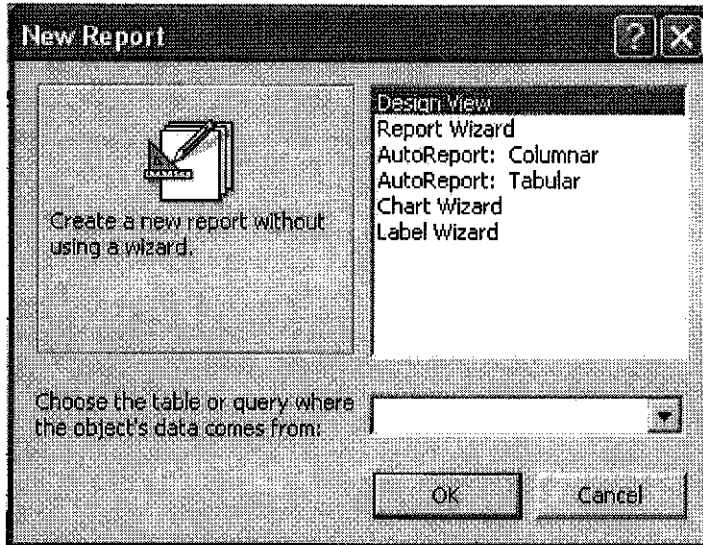
ขั้นตอนการปฏิบัติที่ 5.1

การสร้าง Report ด้วย Report Wizard

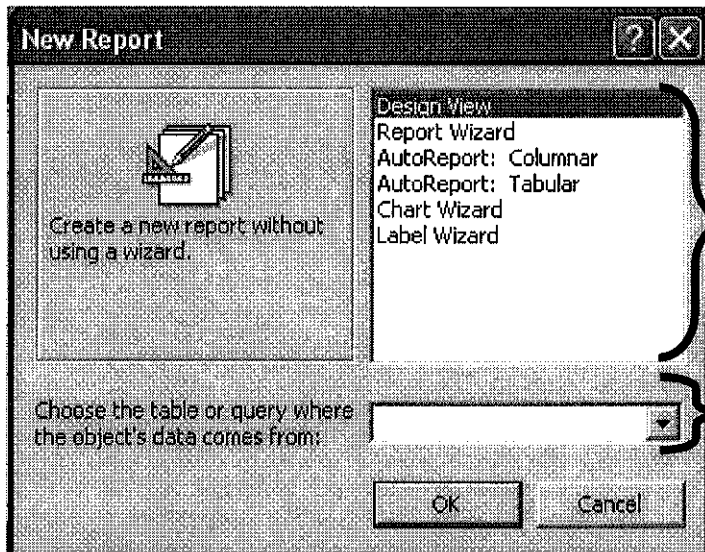
1. ให้นักศึกษาเรียกฐานข้อมูล **Week1.mdb** ขึ้นมาใช้งาน
2. จากหน้าต่าง **Week1:Database** ให้คลิกเลือก **Object** ชื่อ **Report**
3. เลือกคำสั่ง **New** ดังรูป



4. ปรากฏหน้าต่าง New Report ดังรูป



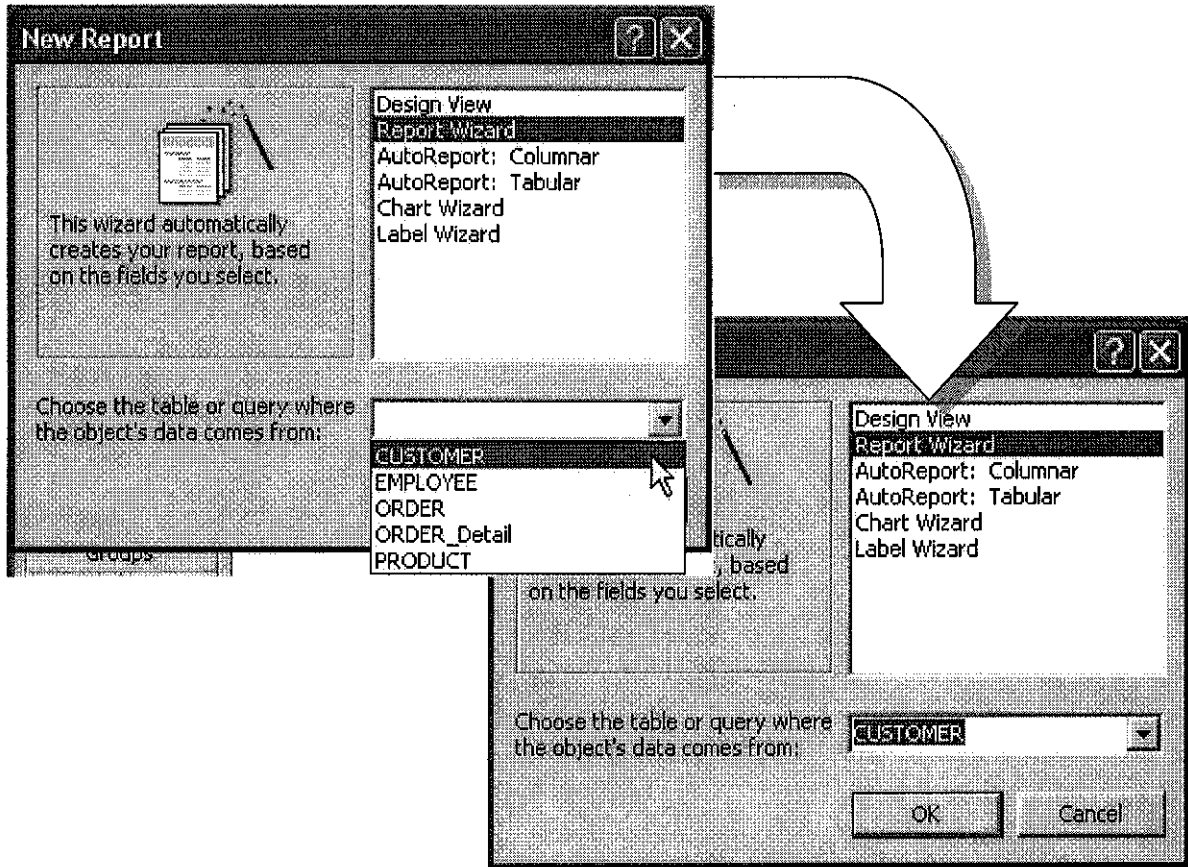
หน้าต่าง New Report ประกอบด้วย 2 ส่วนคือ



ส่วนกำหนดรูปแบบ
การสร้าง Report

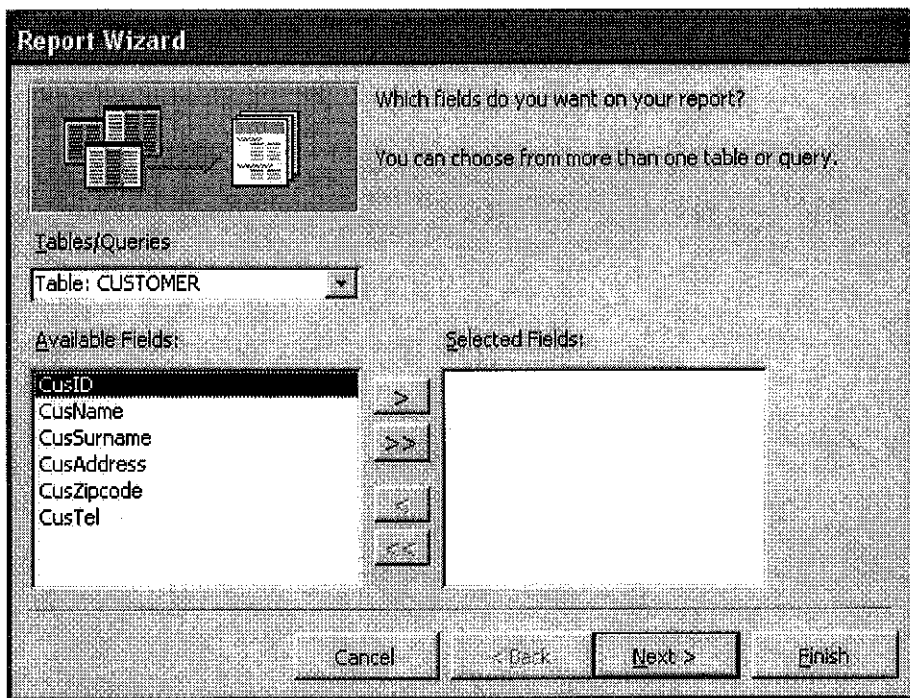
ส่วนกำหนดข้อมูลที่
นำมาสร้าง Report

5. คลิกเลือกรูปแบบการสร้าง Report แบบ Report Wizard และคลิกเลือกชื่อตาราง CUSTOMER โดยคลิกที่ปุ่มลูกศรด้านข้าง ดังรูป

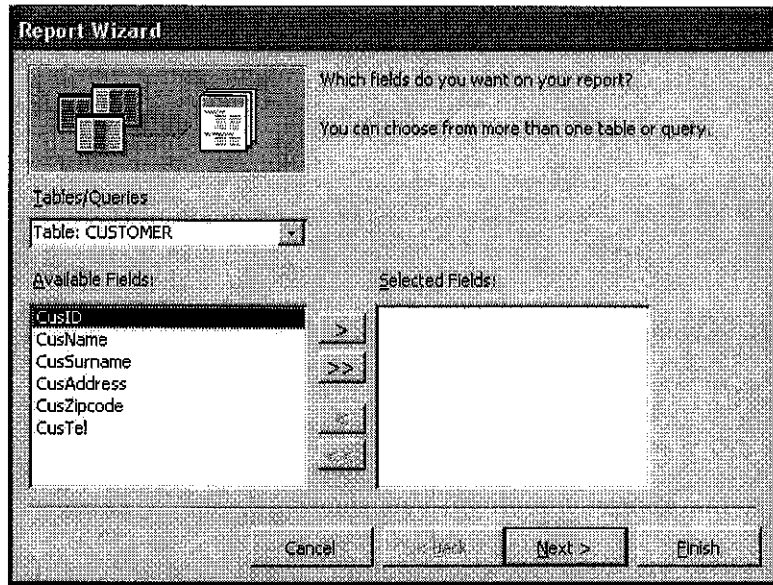


6. คลิกปุ่มคำสั่ง OK

7. ปรากฏหน้าต่าง Report Wizard ดังรูป




หน้าต่าง Report Wizard ประกอบด้วย 3 ส่วนคือ



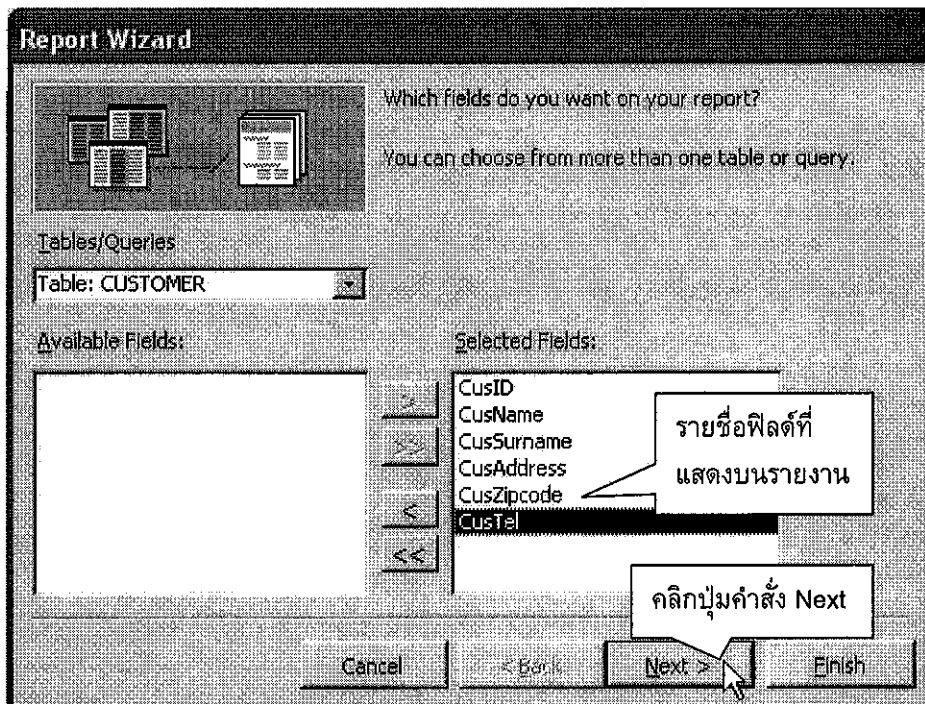
ส่วน **Tables / Queries** กำหนดชื่อ Table หรือ Query ที่ต้องการสร้าง Report

ส่วน **Available Fields** รายชื่อฟิลด์ต่างๆ ของ Table หรือ Query ที่ต้องการสร้าง Report

ส่วน **Selected Fields** รายชื่อฟิลด์ที่เลือกให้ปรากฏบน Report

8. เลือกฟิลด์ที่ต้องการแสดงบน Report ในที่นี้ให้เลือกทุกฟิลด์ โดยคลิกที่ปุ่ม 

9. ปรากฏชื่อฟิลด์ทุกชื่อในช่อง Select Field ดังรูป

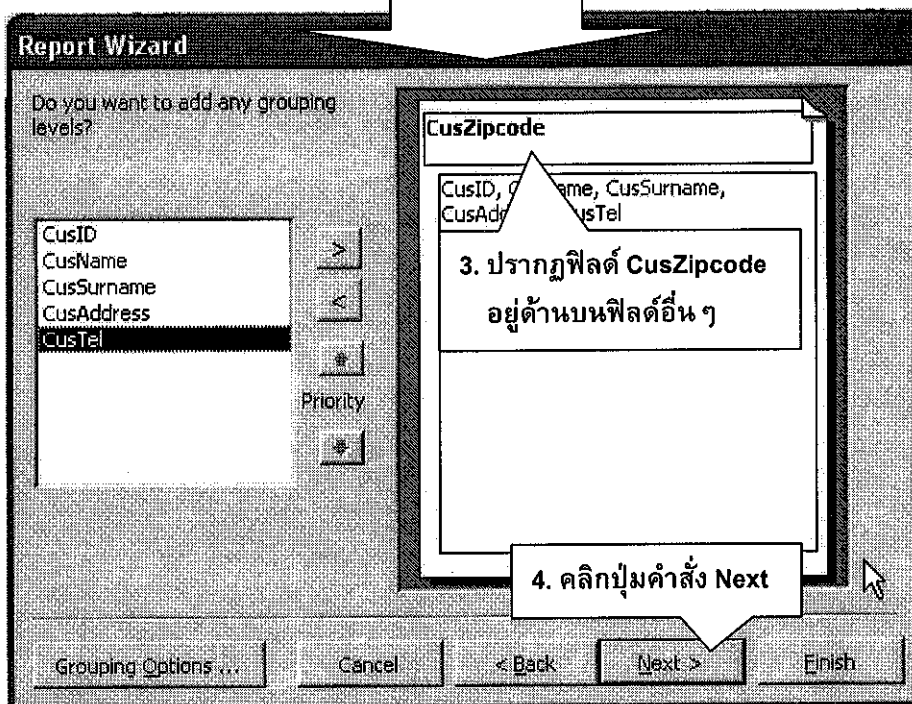
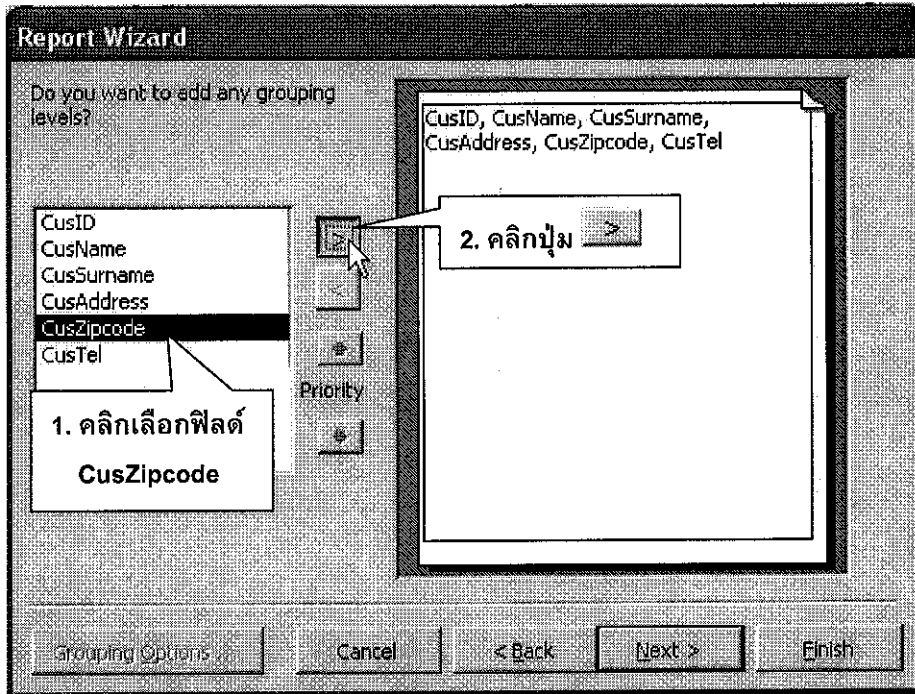


10. คลิกปุ่มคำสั่ง Next

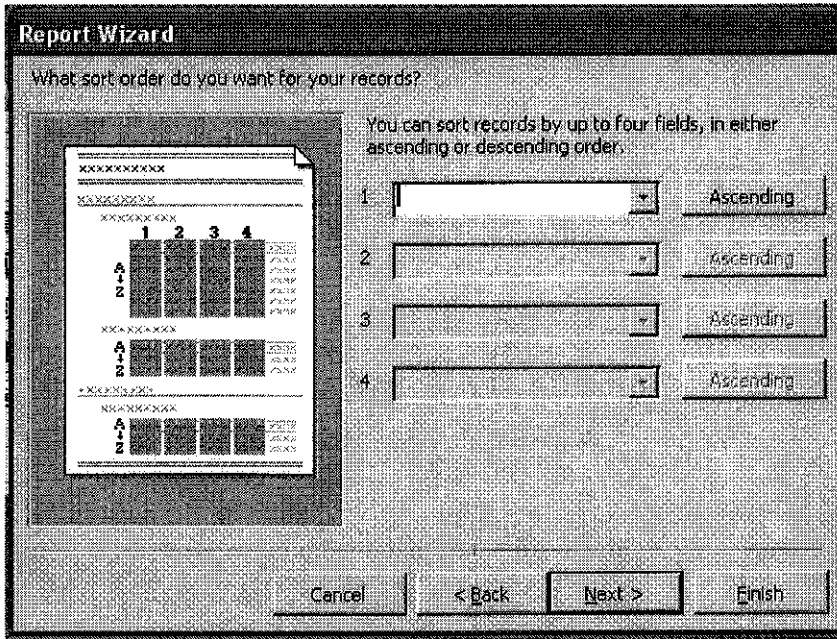
11. ปรากฏหน้าจอดังต่อไปนี้ เพื่อกำหนดการจัดกลุ่มของข้อมูลใน Report ให้นักศึกษาเลือกการจัดกลุ่มข้อมูลแบบใดก็ได้ หรือไม่จัดกลุ่มข้อมูลก็ได้

- กรณีที่เลือกการจัดกลุ่ม ให้นักศึกษาลิขเลือกที่ชื่อ Field ที่ต้องการจัดกลุ่ม จากนั้นคลิกที่เครื่องหมาย > แล้วคลิกปุ่ม Next
- กรณีที่ไม่เลือกการจัดกลุ่ม ให้นักศึกษาลิขปุ่ม Next

ในที่นี้เลือกจัดกลุ่มข้อมูลด้วยรหัสไปรษณีย์ (CusZipcode) จากนั้นให้คลิกปุ่มคำสั่ง Next ดังรูป



12. ปรากฏหน้าจอตัดไป เพื่อกำหนดการจัดเรียงข้อมูล (Sort) ในแต่ละ Record ดังรูป



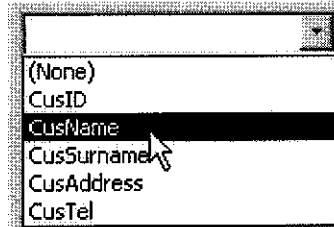
ขั้นตอนการกำหนดการจัดเรียงข้อมูล มีดังนี้

1) คลิกที่ช่องกำหนดการจัดเรียงข้อมูล เพื่อเลือกฟิลด์ที่ต้องการจัดเรียงข้อมูล

ในที่นี้คลิกเลือกฟิลด์

ชื่อลูกค้า (CusName)

ดังรูป

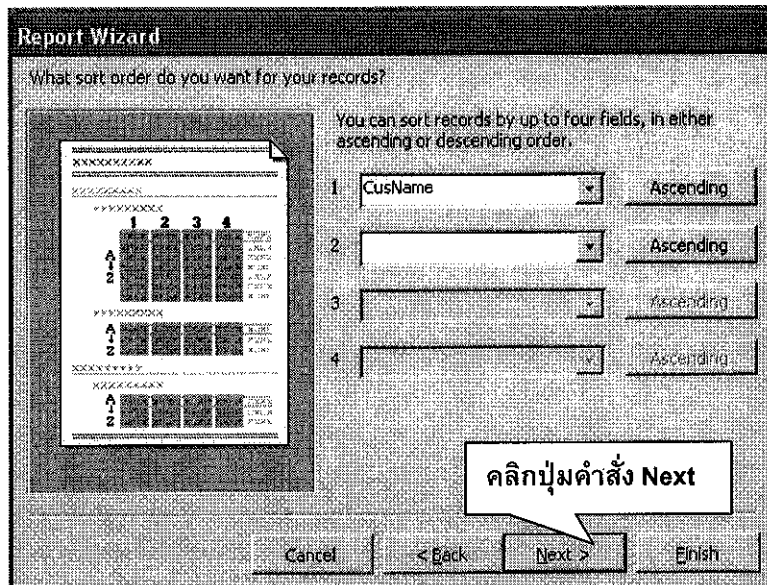


2) คลิกเลือกรูปแบบการจัดเรียงข้อมูล

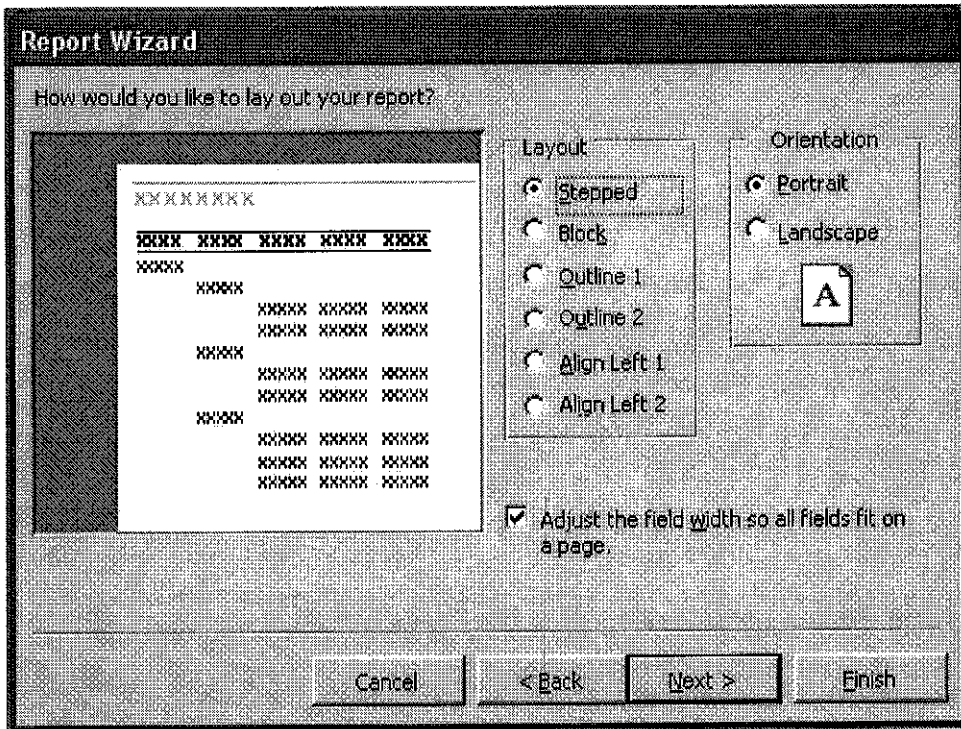
- การจัดเรียงข้อมูลแบบ Ascending เป็นการจัดเรียงข้อมูลจากน้อยไปหามาก

- การจัดเรียงข้อมูลแบบ Descending เป็นการจัดเรียงข้อมูลจากมากไปหาน้อย

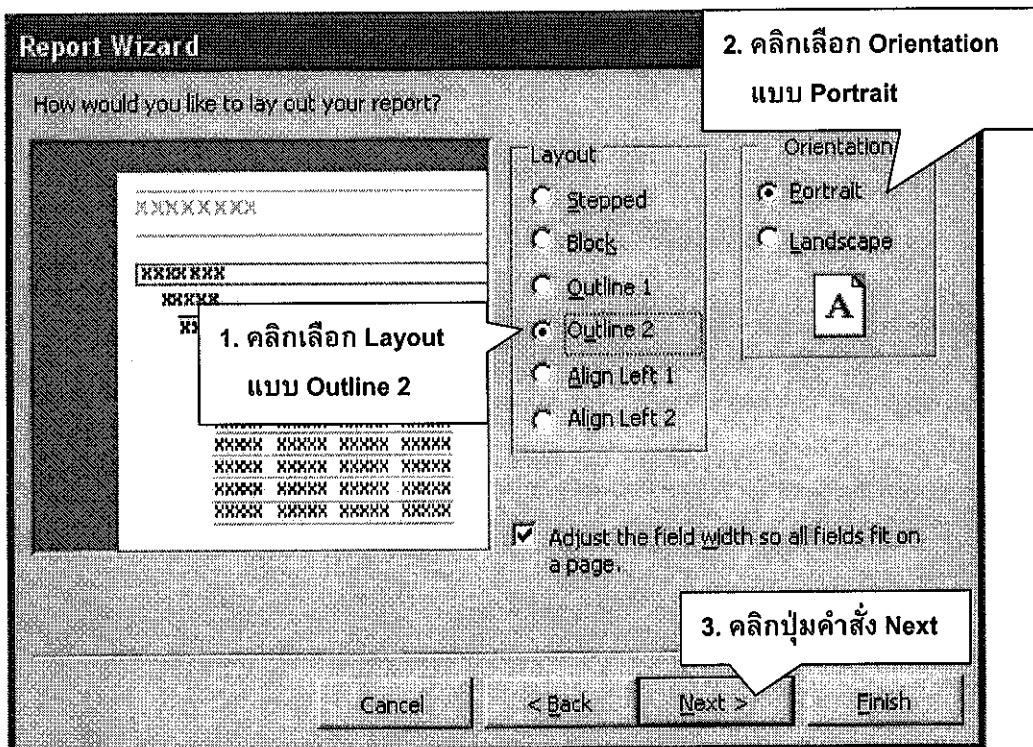
3) คลิกปุ่ม Next ดังรูป



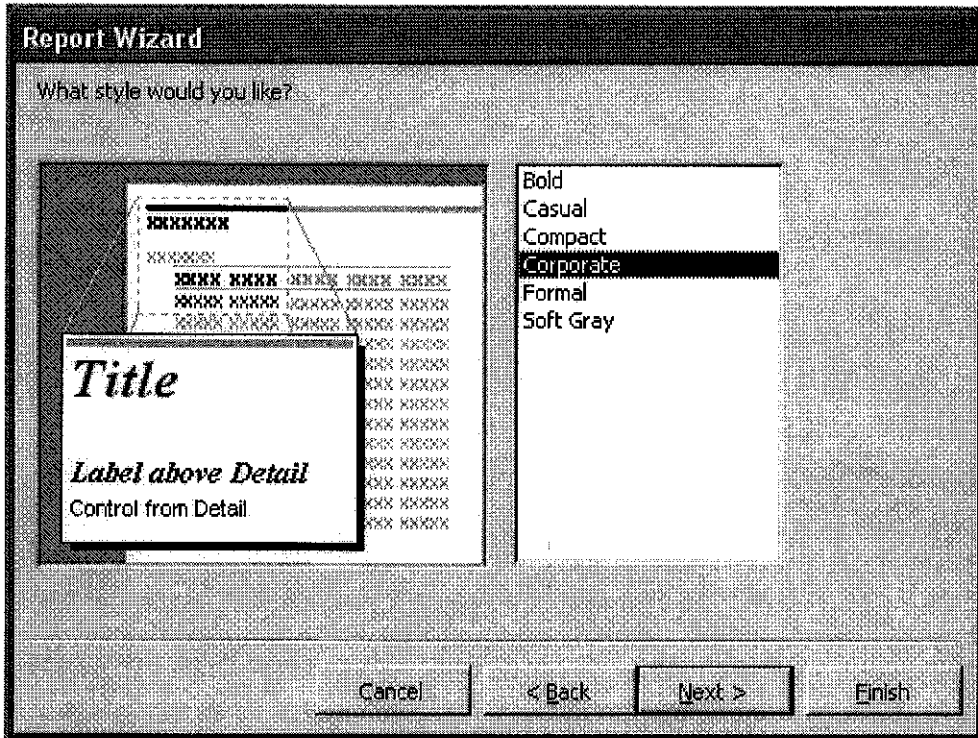
13. ปรากฏหน้าจอถัดไป เพื่อกำหนดรูปแบบของรายงาน (Layout) และหน้ากระดาษของรายงาน (Orientation) ดังรูป



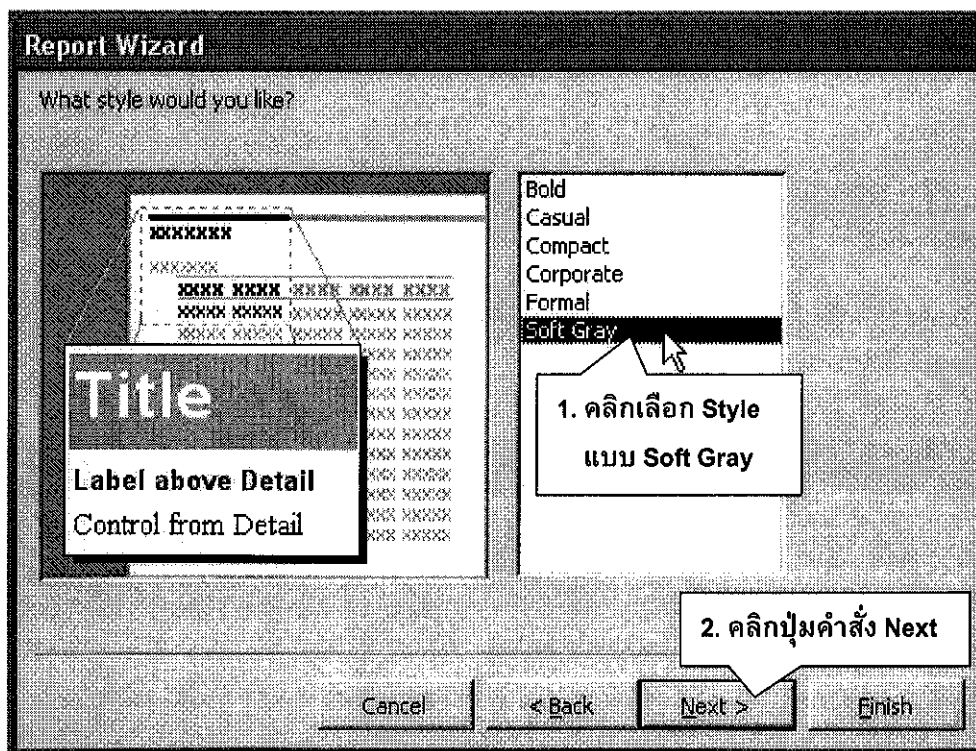
ให้นักศึกษาจะเลือกแบบไหนก็ได้ ในที่นี้เลือกรูปแบบรายงาน (Layout) แบบ Outline 2 และ เลือกหน้ากระดาษรายงาน (Orientation) แบบ Portrait ดังรูป จากนั้นให้คลิกปุ่ม Next



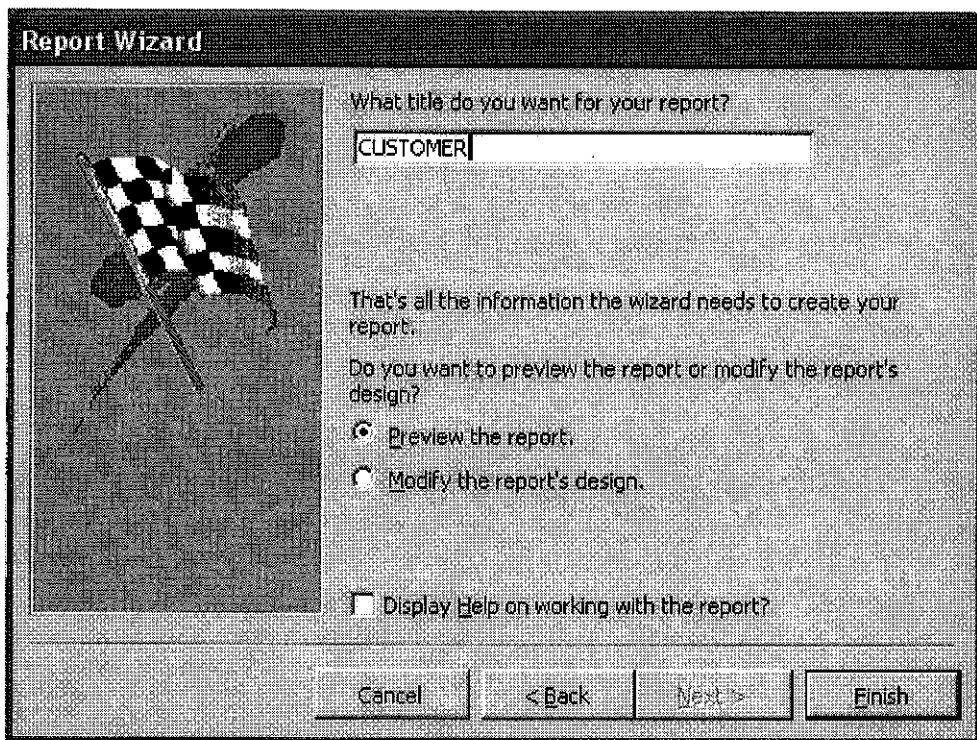
14. ปรากฏหน้าจอตัดไป เพื่อกำหนดรูปแบบตัวอักษร (Style) ให้กับส่วน Page Header



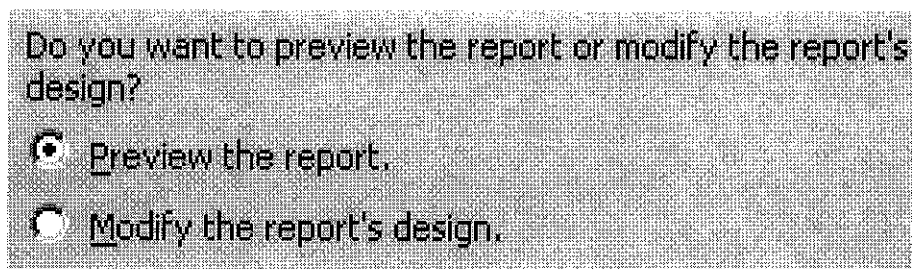
ให้นักศึกษาจะเลือกแบบไหนก็ได้ ในที่นี้เลือกรูปแบบตัวอักษร (Style) ในส่วน Page Header แบบ Soft Gray ดังรูป จากนั้นให้คลิกปุ่ม Next



15. ปราบกฏหน้าจอดีไป ให้ตั้งชื่อของ Report ให้นักศึกษาตั้งชื่อว่า CUSTOMER ดังรูป



16. คลิกเลือกรูปแบบการแสดงผลจากการสร้าง Report ด้วย Report Wizard ดังรูป



รูปแบบ	คำอธิบาย
Preview the report	แสดงรายงานในรูปแบบมุมมองที่ใช้สำหรับแสดงรายงานในรูปแบบภาพก่อนพิมพ์
Modify the report's design	แสดงรายงานในรูปแบบมุมมองที่ใช้สำหรับแก้ไข Report

ในที่นี้เลือกรูปแบบ Preview the report ดังรูป จากนั้นให้คลิกปุ่ม Finish

17. ปรากฏ Report ที่สร้างขึ้น ในมุมมองแบบก่อนพิมพ์ ดังรูป

CUSTOMER

CUSTOMER

CusZipcode	10300			
ชื่อ	นามสกุล	นามสกุล	ที่อยู่	เบอร์โทรศัพท์
นาค	003	งาม	9 พญาไท กรุงเทพฯ	0-1001-1009

CusZipcode	11110			
ชื่อ	นามสกุล	นามสกุล	ที่อยู่	เบอร์โทรศัพท์
พรตนันต์	005	เอี่ยมจรชัย	81 หมู่ 1 อ.บางบัวทอง	0-6666-6666

CusZipcode	30000			
ชื่อ	นามสกุล	นามสกุล	ที่อยู่	เบอร์โทรศัพท์
เงิน	001	ดี	213 อ.เมือง จ.นครราชสีมา	0-4422-2222
เพชร	004	สวย	99 อ.เมือง จ.นครราชสีมา	0-4422-0001

CusZipcode	40000			
ชื่อ	นามสกุล	นามสกุล	ที่อยู่	เบอร์โทรศัพท์
ทอง	002	แพง	111 อ.เมือง จ.ขอนแก่น	0-4333-2222

Page: 1/1

ขั้นตอนการปฏิบัติ

การเพิ่มส่วนหัวและส่วนท้ายของรายงาน

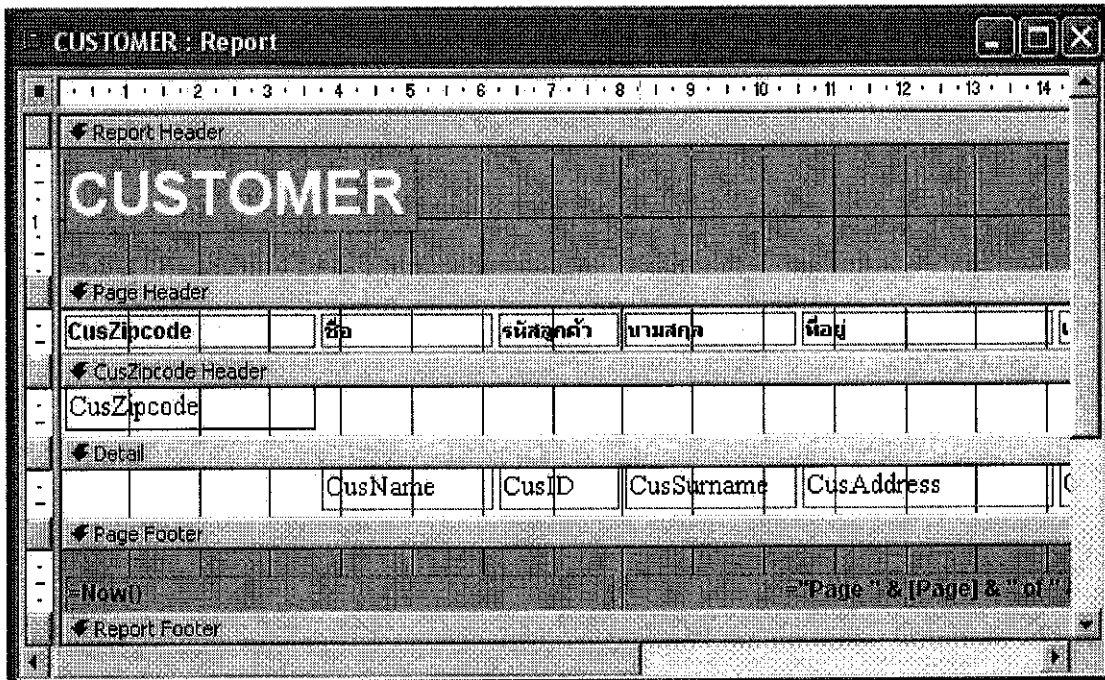
1. จากหน้าต่าง Report ดังรูป

1) เลือกเมนู View เลือกคำสั่ง Design View หรือ

2) คลิกที่ Icon 
จากนั้นเลือกคำสั่ง
Design View ดังรูป



2. ปรากฏหน้าต่าง Design Report ดังรูป



คำสั่ง

1. ให้นักศึกษาร่างรายงาน โดยใช้ข้อมูลจาก Table : CUSTOMER, ORDER, PRODUCT และ ORDER_Detail โดยให้มีการจัดเรียงข้อมูลดังตัวอย่าง โดยตั้งชื่อ Report ว่า ORDER

EMPLOYEE	
รหัสลูกค้า	111
ชื่อ	เก่ง
นามสกุล	รวย
รับส่ง	12/12/1998
รับส่ง	18/12/1998
ชื่อสินค้า	จำนวน
เสื้อยืด	30
กางเกง	20
รับส่ง	30/12/1998
รับส่ง	31/1999
ชื่อสินค้า	จำนวน
รองเท้า	30
เข็มขัด	5
รหัสลูกค้า	222
ชื่อ	เลิศ
นามสกุล	หรุ
รับส่ง	15/12/1998
รับส่ง	20/12/1998
ชื่อสินค้า	จำนวน
เสื้อยืด	20
กางเกง	15

บรรณานุกรม

- วุฒิพงศ์ พงศ์สุวรรณ และวลัยพร จรนินเทศ. (2543). **How to learn database with Microsoft Access 2002**. กรุงเทพฯ : ซอฟต์แวร์ ปาร์ค.
- สัจจะ จรัสรุ่งรวีร์ และสุวิมล วังจันทร์สุข. (2545). **คู่มือการใช้งาน Access 2002 ฉบับสมบูรณ์**. กรุงเทพฯ : อินโฟเพรส.
- สัมฤทธิ์ วงศ์เด่นดวง. (2547). **คัมภีร์การออกแบบและพัฒนาระบบฐานข้อมูลด้วย Microsoft Access เล่ม1**. กรุงเทพฯ : เคพีที คอมพ์ แอนด์ คอนซัลท์.
- Frye, Curtis. (2001). **Microsoft Access version 2002 plain & simple**. Redmond, Wash. : Microsoft Press.
- Online Training Solutions. (2001). **Microsoft Access version 2002 step by step**. Redmond, Wash. : Microsoft Press.
- Wood, Dawn Parrish. (2002). **Essentials Access 2002**. Upper Saddle River, N.J. : Prentice Hall.