

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

**ระบบฐานข้อมูล และการวิเคราะห์กีฬาเทนนิส
DATABASE SYSTEM AND ANALYSIS OF TENNIS**

โดย
นาย ตันศักดิ์ จารุพุทธิกร
นาย พชร เจริญวัฒนะตระกูล

๒/๓๑.
๐๗/๑๕๕
๗๕/๕

เลขหมู่.....
เลขทะเบียน.....**83086**
วัน,เดือน,ปี.....- 5 ส.ค. 2551

b. 11๑๕408X
i.....

**ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2550**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATABASE SYSTEM AND ANALYSIS OF TENNIS

BY

Mr. TONSAK JARUPUTHIKORN

Mr. POTCHARA CHAROENWATTANATRAKORN



**A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR THE DEGREE OF
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์ ระบบฐานข้อมูล และการวิเคราะห์กีฬาเทนนิส
ชื่อนักศึกษา นาย ดันศักดิ์ จารุพุทธิกร รหัสประจำตัว 47010259
นาย พชร เจริญวัฒนะตระกูล รหัสประจำตัว 47010480
อาจารย์ที่ปรึกษา ร.ศ. นภพินท์ อนันตรศิริชัย
ระดับการศึกษา ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมสารสนเทศ
ภาควิชา วิศวกรรมสารสนเทศ
ปีการศึกษา 2550

บทคัดย่อ

วิทยานิพนธ์นี้กล่าวถึงระบบฐานข้อมูล และการวิเคราะห์กีฬาเทนนิส ซึ่งเนื้อหาประกอบด้วยประวัติและสถิติการได้รับถ้วยรางวัลของตำนานนักเทนนิส การแสดงผลกราฟเปรียบเทียบกับรางวัลของตำนานนักเทนนิสจากระบบการค้นหา การค้นหาข้อมูลประวัตินักเทนนิสอาชีพ ประวัติสนามการแข่งขันรายการแกรนด์สแลม และวิธีการเล่น กฎกติกาต่าง ๆ ซึ่งนำเสนอผ่านทางเว็บไซต์ในรูปแบบแพลตฟอร์มอินเทอร์เน็ต รูปภาพ และวิดีโอ

Thesis Title Database system and Analysis of tennis
Student Mr. Tonsak Jaruputhikorn ID. 47010259
Mr. Potchara Charoenwattanatrakoon ID. 47010480
Advisor Ms. Noppin Anantrasirichai
Graduate Level Bachelor Degree of Information Engineering
Department Information Engineering
Academic Year 2007

ABSTRACT

In this thesis described the database system and analysis of tennis. The contents contain of history and trophies statistic of legendary tennis players, simulations of graphs to compare trophies of legendary tennis players from a searching system, searching of professional tennis player's profiles, the history of the Grandslam tournaments, how to play and rules. All of these represented in a form of flash contents, picture and video via a website.

กิตติกรรมประกาศ

ปริญญาโทฉบับนี้คงไม่อาจสำเร็จไปได้ หากไม่ได้รับความช่วยเหลือและความร่วมมือจากหลาย ๆ ฝ่ายด้วยกัน เริ่มจากอาจารย์ที่ปรึกษาปริญญาโท ร.ศ. นภพินท์ อนันตรศิริชัย คอยช่วยเหลือ ให้คำปรึกษา คำแนะนำต่าง ๆ ตลอดระยะเวลาในการทำโครงการนี้ ซึ่งทำให้ทั้งโครงการและปริญญาโทเล่มนี้ผ่านลุล่วงไปได้ด้วยดี

สุดท้ายนี้คณะผู้จัดทำขอกราบขอบพระคุณผู้มีพระคุณของข้าพเจ้าทั้งหลาย บิดา มารดา ครู อาจารย์ ที่สั่งสอนถ่ายทอดความรู้ ให้คำแนะนำอันมีค่าทั้งหลาย ตลอดจนช่วยสนับสนุนในทุก ๆ ด้านจนข้าพเจ้ามีวันนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ช
สารบัญตาราง	ฅ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของปริญญาโท	1
1.2 จุดมุ่งหมายของปริญญาโท	1
1.3 แนวความคิดและที่มา	1
1.4 ขอบเขตของงาน	2
1.4.1 เนื้อหา	2
1.4.2 ส่วนการทำงาน	2
1.4.3 รูปแบบ	2
บทที่ 2 ทฤษฎีที่ใช้ในโครงการ	3
2.1 การใช้งาน Dreamweaver 8	3
2.1.2 เมนู	3
2.1.3 รูปแบบการแสดงผล	3
2.1.4 การสร้างตาราง (Table)	5
2.1.5 การสร้างลิงค์	6
2.1.6 การสร้างฟอร์ม	7
2.1.7 การใช้งานเฟรม (Frame)	8
2.1.8 การปรับแต่งข้อความ และการเลือกรูปแบบตัวอักษร	9
2.2 การประยุกต์ใช้ Flash Professional 8	9
2.2.1 การทำงาน	9
2.2.2 สเตรจ	9
2.2.3 พาเนล Properties	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

2.2.4	ไทม์ไลน์	10
2.2.5	พาเนล Library	11
2.2.6	พาเนล Actions	12
2.2.7	พาเนล Behaviors	12
2.2.8	พาเนล Components	13
2.2.9	พาเนล Components Inspector	13
2.3	ภาษา SQL	14
2.3.1	ประเภทของคำสั่งภาษา SQL	14
2.3.2	ชุดคำสั่ง DDL	14
2.3.3	ชุดคำสั่ง DML	20
2.3.4	ชุดคำสั่ง DCL	26
2.4	ภาษา PHP	29
2.4.1	การเขียน โค้ด PHP	29
2.4.2	กฎในการกำหนดชื่อให้แก่ตัวแปร	30
2.4.3	คำสั่งที่ใช้รับส่งตัวแปร	30
2.4.4	การใช้งาน PHP ร่วมกับ MySQL	32
2.5	ทฤษฎีระบบฐานข้อมูล	34
2.5.1	โครงสร้างข้อมูล	34
2.5.2	การควบคุมความถูกต้องให้กับข้อมูล	35
บทที่ 3	การออกแบบโครงงาน	37
3.1	การออกแบบภาพรวม	37
3.2	การออกแบบในระดับถัดไป	38
3.3	การออกแบบในระดับการประมวลผล	38
3.4	การออกแบบฐานข้อมูล	39
3.5	การนำฐานข้อมูลที่ออกแบบมาใช้งาน	44
บทที่ 4	ผลการทดลอง	46
4.1	ส่วนของเมนูหลัก	46
4.2	หัวข้อ Legend & History	46
4.3	หัวข้อ Academy	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

4.4 หัวข้อ BBS	57
บทที่ 5 สรุป	58
5.1 การพัฒนาโครงการ	58
5.2 ปัญหาระหว่างการพัฒนา	58
5.3 แนวทางในการพัฒนาต่อ	58
เอกสารอ้างอิง	59



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
บทที่ 2	
รูปที่ 2.1 แสดงเมนู	3
รูปที่ 2.2 แสดงโค้ดวิววี	4
รูปที่ 2.3 แสดงโค้ดและดีไซน์วิววี	4
รูปที่ 2.4 แสดงดีไซน์วิววี	5
รูปที่ 2.5 แสดงตาราง	6
รูปที่ 2.6 แสดงการกำหนดจุดที่ต้องการจะไป	6
รูปที่ 2.7 แสดงแท็บฟอร์ม	7
รูปที่ 2.8 แสดงรูปแบบของเฟรม	8
รูปที่ 2.9 แสดงการปรับแต่งข้อความ	9
รูปที่ 2.10 แสดงสแตจ	10
รูปที่ 2.11 แสดงพานอล properties	10
รูปที่ 2.12 แสดงไทม์ไลน์	11
รูปที่ 2.13 แสดงพานอล Library	11
รูปที่ 2.14 แสดงพานอล Actions	12
รูปที่ 2.15 แสดงพานอล Behaviors	12
รูปที่ 2.16 แสดงพานอล Components	13
รูปที่ 2.17 แสดงพานอล Components Inspector	13
รูปที่ 2.18 แสดงโครงของตาราง	16
รูปที่ 2.17 แสดงพานอล Components Inspector	13
รูปที่ 2.19 ภาพแสดงตารางและรายละเอียด	34
รูปที่ 2.20 แสดงตารางและความสัมพันธ์ของคีย์แต่ละคีย์	36
บทที่ 3	
รูปที่ 3.1 ภาพแผนภาพระดับสูงสุด (context diagram)	37
รูปที่ 3.2 ภาพแผนภาพระดับล่าง (diagram 0)	38
รูปที่ 3.3 ภาพกระแสข้อมูลระดับ 1 (level-1 diagram)	39

สารบัญรูป(ต่อ)

รูปที่	หน้า
รูปที่ 3.4	40
รูปที่ 3.5	40
รูปที่ 3.6	41
รูปที่ 3.7	42
รูปที่ 3.8	42
รูปที่ 3.9	43
บทที่ 4	
รูปที่ 4.1	46
รูปที่ 4.2	47
รูปที่ 4.3	47
รูปที่ 4.4	48
รูปที่ 4.5	49
รูปที่ 4.6	49
รูปที่ 4.7	50
รูปที่ 4.8	50
รูปที่ 4.9	51
รูปที่ 4.10	51
รูปที่ 4.11	52
รูปที่ 4.12	52
รูปที่ 4.13	53
รูปที่ 4.14	53
รูปที่ 4.15	53
รูปที่ 4.16	54
รูปที่ 4.17	55
รูปที่ 4.18	55
รูปที่ 4.19	56
รูปที่ 4.20	57
รูปที่ 4.21	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
บทที่ 2	
ตารางที่ 2.1 ตัวอย่างชนิดข้อมูล (Data type)	16-17
ตารางที่ 2.2 โอเปอเรเตอร์คณิตศาสตร์	24
ตารางที่ 2.3 โอเปอเรเตอร์ตรรกะ	25
ตารางที่ 2.4 คำสั่งการเชื่อมต่อ	32
ตารางที่ 2.5 คำสั่งสร้างและกำหนดใช้งานฐานข้อมูล	32
ตารางที่ 2.6 การประมวลผลคำสั่ง SQL	33
บทที่ 3	
ตารางที่ 3.1 ข้อมูลนักเทนนิสที่ทำการรวบรวมได้	37
ตารางที่ 3.2 ข้อมูลรายการแข่งขัน	41
ตารางที่ 3.3 PlayName	44
ตารางที่ 3.4 Player	44
ตารางที่ 3.5 LegendPlayer	44
ตารางที่ 3.6 PlayerProfile	44
ตารางที่ 3.7 TournamentName	45
ตารางที่ 3.8 TourProfile	45
ตารางที่ 3.9 PrizePoint	45
ตารางที่ 3.10 Champion	45

บทที่ 1

บทนำ

1.1 ความเป็นมาของปริญาานิพนธ์

โครงการนี้ได้นำแนวคิดมาจากการจัดหาระบบฐานข้อมูลเพื่อนำฐานข้อมูลขึ้นมาใช้งาน หรือเพื่อทำการค้นคว้า รวมทั้งใช้ซอฟต์แวร์ต่าง ๆ ในการนำเสนอข้อมูลเหล่านี้ผ่านทางระบบ อินเทอร์เน็ตให้ดึงดูดน่าสนใจและมีประสิทธิภาพ โดยนำมาพัฒนาและจัดทำอยู่ในรูปของเว็บไซต์

1.2 จุดมุ่งหมายของปริญาานิพนธ์

- เพื่อเป็นการนำความรู้ที่ได้ศึกษามาประยุกต์ใช้งานจริง
- เพื่อให้บุคคลที่ชื่นชอบในกีฬาเทนนิสเข้ามาหาข้อมูลของตำนานนักเทนนิส นักเทนนิส คนอื่น ๆ ประวัติความเป็นมาของสนามแข่งขัน รวมทั้งสถิติการได้รับถ้วยรางวัลที่ แข่งขันในรายการใหญ่
- เพื่อให้บุคคลที่สนใจที่จะเริ่มเล่นกีฬาเทนนิสเข้ามาศึกษาหาข้อมูลวิธีการเล่น ตลอดจน กฎกติกาที่ใช้ในการเล่นแบบต่าง ๆ
- เพื่อเป็นแหล่งนัดพบของคนที่รักกีฬาเทนนิส

1.3 แนวความคิดและที่มา

โครงการนี้ได้แนวความคิดมาเนื่องจากผู้จัดทำสนใจ และชื่นชอบในกีฬาเทนนิส ประกอบ กับความนิยมในกีฬานี้ในประเทศไทยเริ่มสูงขึ้นอย่างต่อเนื่อง และในยุคปัจจุบันนี้เป็นยุคของ คอมพิวเตอร์ผู้คนส่วนใหญ่ใช้เทคโนโลยีการติดต่อสื่อสาร หาข้อมูลกันผ่านอินเทอร์เน็ตกันอย่าง กว้างขวาง แต่อย่างไรก็ตามเว็บไซต์ภาษาไทยของกีฬานี้ใด ๆ ที่จัดทำเกี่ยวกับสถิติการได้รับถ้วย รางวัลของนักเทนนิสอาชีพในการแข่งขันรายการใหญ่ ๆ ต่าง ๆ มาเปรียบเทียบกัน ประกอบกับ เนื้อหาต่าง ๆ เกี่ยวกับกีฬาเทนนิสที่นำเสนอในเว็บไซต์เหล่านั้น ตลอดจนการจัดองค์ประกอบยังทำ ได้ไม่ดีเท่าที่ควร จึงเป็นเหตุให้ผู้จัดทำคิดริเริ่มสร้างสรรค์โครงการชิ้นนี้ขึ้นมา

1.4 ขอบเขตของงาน

1.4.1 เนื้อหา

- ประวัติและข้อมูลของตำนานนักเทนนิส และนักเทนนิสอาชีพ
- สถิติการได้รับถ้วยรางวัลในการแข่งขันรายการใหญ่ต่างๆ (แกรนด์สแลม) ของตำนานนักเทนนิส
- ประวัติความเป็นมาสนามการแข่งขันรายการใหญ่ต่างๆ (แกรนด์สแลม)
- วิธีการเล่น การจับกริปแบบต่างๆ การตีโฟร์แฮนด์ การตีแบคแฮนด์ การเสิร์ฟ
- กฎกติกาต่างๆ การนับแต้ม
- การเตรียมพร้อม การวอร์มอัพ รวมทั้งอาการบาดเจ็บที่อาจเกิดขึ้น
- ข้อมูลข่าวสาร เกร็ดความรู้ต่างๆ จากการเล่นเทนนิส
- ข้อมูลเรีกเกต อุปกรณ์ต่างๆ ของนักเทนนิสที่ได้รับความนิยม
- รูปภาพแกลลอรี่
- กระดานข้อมูลข่าวสารแลกเปลี่ยนความคิดเห็น

1.4.2 ส่วนการทำงาน

- การค้นหาข้อมูล และการวิเคราะห์ผลเพื่อเปรียบเทียบสถิติการได้รับถ้วยรางวัลในการแข่งขันรายการใหญ่ต่างๆ (แกรนด์สแลม) ของตำนานนักเทนนิสจัดแสดงผลในรูปแบบกราฟ
- การค้นหาข้อมูล และประวัติของนักเทนนิส

1.4.3 รูปแบบ

- จัดทำในรูปแบบเนื้อหา แฟลช (Flash) รูปภาพ และวิดีโอ (Video)

บทที่ 2

ทฤษฎีที่ใช้ในโครงการ

2.1 การใช้งาน Dreamweaver 8

Dreamweaver 8 เป็นซอฟต์แวร์ที่ช่วยในการสร้างเว็บเพจขึ้นมาได้อย่างสะดวก และง่ายกว่าการสร้างเว็บเพจแบบปกติ เนื่องจากในบางครั้งผู้ใช้ไม่จำเป็นที่จะต้องพิมพ์โค้ด เพื่อที่จะกำหนดรูปแบบเว็บเพจ แต่เราสามารถเลือกรูปแบบ หรือสร้างอ็อบเจกต์ (object) ขึ้นมาจากโปรแกรมที่มีให้ได้ โดยโปรแกรมก็จะทำการแปลงรูปแบบ หรืออ็อบเจกต์ที่เราใช้งานนั้นออกมาเป็นโค้ด ภาษาต่าง ๆ ซึ่งมีให้เลือกหลายภาษาอาทิเช่น HTML Actionscript Javascript XML ASP.NET C# ColdFusion JSP หรือ PHP

2.1.2 เมนู

เป็นแถบที่รวบรวมคำสั่งพื้นฐานทั้งหมดที่ใช้ในการกำหนดรูปแบบการแสดงผลลักษณะต่าง ๆ ของโปรแกรมเอง รวมทั้งใช้ในการจัดการและแสดงผลต่าง ๆ ของเว็บไซต์ที่เรากำลังพัฒนา ไม่ว่าจะเป็นการแทรกข้อมูลอาทิเช่น แฟลช อ็อบเจกต์ (Flash object) ภาพ หรือ การสร้างฟอร์ม และตาราง การกำหนดรูปแบบของตัวอักษร และสีต่าง ๆ การพรีวิว (preview) ข้อมูล เป็นต้น

Macromedia Dreamweaver 8 - [C:\wwwroot\Untitled-2.php (XHTML)*]

File Edit View Insert Modify Text Commands Site Window Help

รูปที่ 2.1 แสดงเมนู

2.1.3 รูปแบบการแสดงผล

Dreamweaver 8 จะแบ่งการแสดงผลเป็น 3 แบบหลัก ๆ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.1 Code

คือการ Show Code view ซึ่งเป็นการแสดงผลในรูปแบบภาษา แสดงออกมาเป็นโค้ดทั้งหมด

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html;
6 charset=iso-8859-1" />
7 <title>Untitled Document</title>
8 <script src="Scripts/AC_RunActiveContent.js" type="text/javascript">
9 </script>
10 </head>
11 <body>
12 <table width="2016" height="179" border="0">
13 <tr>
14 <td width="127" scope="row"></td>

```

รูปที่ 2.2 แสดงโค้ดวิว

2.1.3.2 Split

คือการ Show Code and Design views ซึ่งเป็นการแสดงผลในรูปแบบภาษา และในส่วนของการออกแบบ ซึ่งเราสามารถควบคุมไปกับการสร้างเว็บเพจได้

```

21 <td width="44">&nbsp;</td>
22 <td width="1821"><p class="style1">This is test page</p>
23 <p>&nbsp;</p></td>
24 </tr>
25 <tr>
26 <th scope="row">&nbsp;</th>

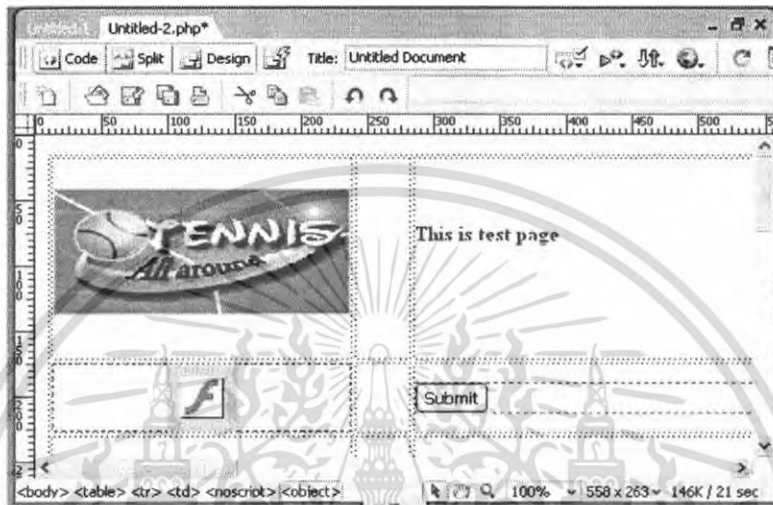
```

รูปที่ 2.3 แสดงโค้ดและดีไซน์วิว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.3 Design

คือการ Show Design view ซึ่งเป็นการแสดงผลในส่วนของการออกแบบ ซึ่งจะแสดงออกมาในรูปแบบกราฟฟิก (graphic) และอีอบเจกต์ต่าง ๆ



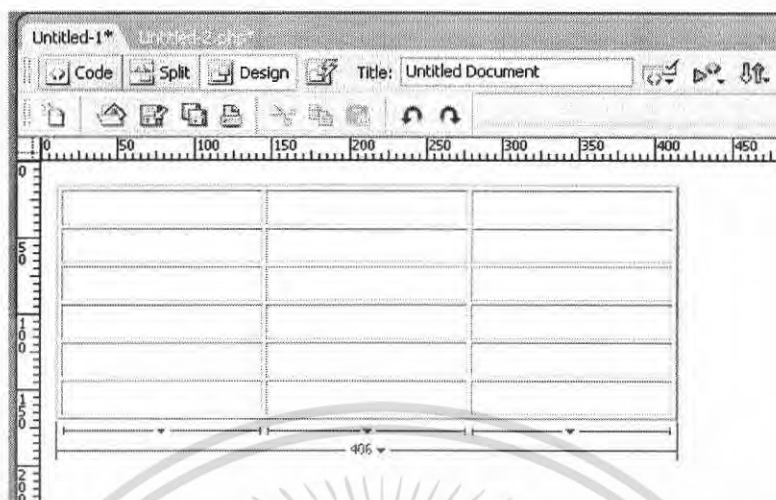
รูปที่ 2.4 แสดงดีไซน์วิว

2.1.4 การสร้างตาราง (Table)

การสร้างตารางจะมีประโยชน์มากสำหรับการจัดรูปแบบเนื้อหาต่าง ๆ ภายในเว็บไซต์ ไม่ว่าจะเป็นรูปภาพ บทความ อีอบเจกต์ ต่าง ๆ ซึ่งจะสามารถทำได้ง่ายกว่าการไม่ใช้ตาราง เราสามารถสร้างตารางได้โดยไปที่ปุ่ม menu Insert > Table แล้วจะปรากฏหน้าต่างขึ้นมาให้กำหนดค่าต่าง ๆ ซึ่งค่าหลัก ๆ มีดังนี้

Rows	หมายถึง จำนวนแถวของตาราง
Columns	หมายถึง จำนวนคอลัมน์ของตาราง
Table width	ใช้กำหนดความกว้างของตาราง
Border thickness	ใช้กำหนดความหนาของขอบตาราง
Cell padding	ใช้กำหนดระยะห่างระหว่างข้อความกับขอบตาราง
Cell spacing	ใช้กำหนดระยะห่างระหว่างเซลล์ของตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงตาราง

ในการจัดรูปแบบเนื้อหาต่างๆ ภายในเว็บไซต์ถ้าเราต้องการไม่ให้ขอบของตารางปรากฏในหน้าเว็บเพจ ก็ให้กำหนดค่า Border thickness เป็น 0

2.1.5 การสร้างลิงค์

การสร้างลิงค์เป็นการเชื่อมโยงเว็บเพจ จากหน้าหนึ่งไปยังอีกหน้าหนึ่ง ซึ่งการเชื่อมโยงนั้นมีได้หลายแบบด้วยกัน ยกตัวอย่างเช่น การลิงค์ไปยังอีเมล การลิงค์ภายในหน้าเดียวกัน การลิงค์ไปที่ไฟล์อื่น การลิงค์ไปยังเว็บไซต์อื่น

2.1.5.1 การลิงค์ภายในหน้าเดียวกัน

จะใช้สำหรับเว็บเพจที่มีเนื้อหาภายในหน้ายาวมาก ๆ การใช้การสกรอล (scroll) ลงมาเรื่อย อาจเป็นวิธีที่ไม่สะดวกนัก ดังนั้นวิธีนี้จะทำให้การดูเนื้อหาภายในหน้านั้นมีความสะดวกมากขึ้น

การใช้งานโดยการเลื่อนเคอร์เซอร์ไปยังตำแหน่งที่ต้องการจะไปของหน้านั้นๆ แล้วเลือกที่ปุ่มรูปสมอสีทองตรงแท็บ Common ดังรูป เพื่อเป็นการตั้งชื่อให้จุดที่ต้องการจะไป



รูปที่ 2.6 แสดงการกำหนดจุดที่ต้องการจะไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสร็จแล้วให้เลือกออบเจกต์ที่เราต้องการจะลิงค์ไปยังจุดนั้น ใน properties ด้านล่าง ให้กำหนดตรงช่อง Link เป็น “ # (แล้วตามด้วยชื่อจุดที่ต้องการจะไปซึ่งตั้งชื่อไว้แล้วในตอนต้น) ” ซึ่งในตัวอย่างจะพิมพ์ว่า “ #best ”

2.1.5.2 การลิงค์ไปยังไฟล์อื่น

เป็นการเชื่อมโยงไปยังเว็บเพจหน้าอื่น ๆ ที่เราสร้างขึ้น ซึ่งหน้าเว็บเพจจะเปลี่ยนหน้าใหม่ขึ้นมา โดยการใช้งานให้เราเลือกออบเจกต์ที่เราต้องการจะลิงค์ไปแล้วกำหนดตรง properties ด้านล่างตรงช่อง Link ให้เป็นชื่อไฟล์ที่เราต้องการจะไป

2.1.5.3 การลิงค์ไปยังเว็บไซต์อื่นและการลิงค์ไปยังอีเมล

เป็นการเชื่อมโยงไปยังเว็บไซต์อื่นๆ และการเชื่อมโยงไปยังอีเมล วิธีการจะเหมือนกันคือกำหนดตรง Link เพียงแต่ถ้าเป็นเว็บไซต์ก็ให้พิมพ์ http:// นำหน้า ส่วนถ้าเป็นอีเมล ให้พิมพ์ mailto: นำหน้า

2.1.6 การสร้างฟอร์ม

เว็บเพจที่มีการกรอกข้อมูลเพื่อส่งข้อมูลกลับไป จะมีการใช้แท็บฟอร์มดังรูป



รูปที่ 2.7 แสดงแท็บฟอร์ม

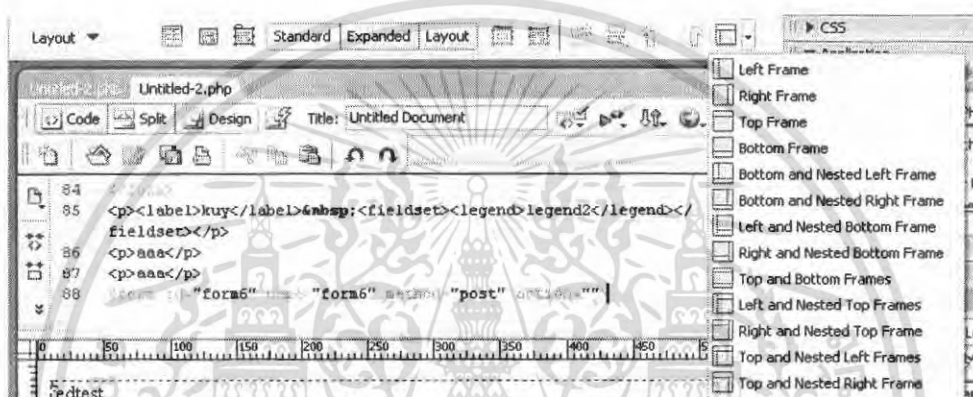
ซึ่งจะมีเมนูให้เราใช้งานหลัก ๆ ดังนี้

- Text Field เป็นช่องใส่ข้อความแบบแถวเดียว
- Hidden Field ใช้ในกรณีต้องการส่งข้อมูลบางชนิดไปประมวลผลต่อ ซึ่งส่วนนี้จะไม่ปรากฏในหน้าเว็บเพจ
- Textarea เป็นช่องใส่ข้อความแบบหลายบรรทัด
- Checkbox เป็นการแสดงข้อมูลแบบตัวเลือก ซึ่งสามารถเลือกได้มากกว่าหนึ่งตัวเลือก
- List/Menu เป็นการแสดงข้อมูลแบบรายการให้เลือก
- Image Field เป็นส่วนที่จะแสดงรูปภาพ
- File Field เป็นส่วนที่สำหรับ browse ไฟล์ที่จะอัปโหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 การใช้งานเฟรม (Frame)

เฟรมเป็นการแบ่งหน้าเว็บเพจหน้าหนึ่ง ๆ ออกเป็นส่วน ๆ ซึ่งแต่ละส่วนจะเป็นอิสระต่อกัน รูปแบบของเฟรมจะมีให้เลือกหลายแบบด้วยกันขึ้นอยู่กับความต้องการใช้งานของผู้ใช้ ส่วนประกอบของหลักของเฟรมจะประกอบด้วย เฟรมหลัก (frameset) และเฟรมย่อย (frame) เสมอ โดยเฟรมหลักมีหน้าที่ควบคุมลักษณะของการแบ่งเฟรมของเฟรมย่อย ส่วนเฟรมย่อยจะควบคุมลักษณะของตัวเอง รวมทั้งข้อมูลที่จะใส่ลงในเฟรม รูปแบบที่มีให้เลือกแสดงดังรูป



รูปที่ 2.8 แสดงรูปแบบของเฟรม

2.1.7.1 การเชื่อมโยงระหว่างเฟรม

ในหัวข้อนี้เราสามารถสร้างการลิงค์ไปยังเพจอื่น ๆ ที่เราสร้างขึ้นมาได้ โดยการเปลี่ยนหน้าเฉพาะเฟรมหลัก(main frame) อย่างเดียวโดยที่เฟรมซ้าย(left frame) หรือเฟรมบน(top frame) ไม่เปลี่ยนแปลงไป ยกตัวอย่างเราจะสร้างอ็อบเจกต์ไว้ทาง leftframe แล้วกำหนด properties ของอ็อบเจกต์ในส่วน Link เป็นหน้าเว็บเพจที่เราต้องการจะไป และในส่วน Target ให้เลือกเป็น mainframe เพียงเท่านั้น หน้า mainframe ก็จะเปลี่ยนไปเมื่อเราคลิกเลือกที่อ็อบเจกต์ แต่ว่าเฟรมที่เหลือไม่เปลี่ยนแปลงตาม

2.1.7.2 การลบเฟรม

เมื่อเฟรมที่สร้างขึ้นผู้พัฒนาไม่จำเป็นที่จะต้องใช้อีกต่อไปแล้ว หรือว่าผู้พัฒนาต้องการเปลี่ยนรูปแบบของเฟรม เราต้องทำการลบเฟรมซึ่งสามารถทำได้ง่าย ๆ โดยการคลิกเลือกขอบของเฟรมที่ไม่ต้องการแล้วค้างไว้ แล้วเลื่อนจนไปชิดขอบหน้าต่าง จะทำให้เฟรมนั้นหายไปทันที

ข้อดีของการใช้เฟรมคือ จะแบ่งการทำงานของเว็บเพจเป็นเฟรม แต่ละเฟรมเป็นอิสระต่อกัน ทำให้เว็บเพจในเฟรมหนึ่งจะไม่กระทบกระเทือนกับเว็บเพจในอีกเฟรมหนึ่งที่แสดงอยู่ในหน้าเดียวกัน

2.1.8 การปรับแต่งข้อความ และการเลือกรูปแบบตัวอักษร

เราสามารถปรับแต่งข้อความ และรูปแบบตัวอักษรได้อย่างง่ายดายโดยการใช้เมาส์คลิกลากให้ครอบคลุมข้อความแล้วทำการปรับแต่งข้อความได้ทันทีในส่วนของ properties ไม่ว่าจะเป็นการเปลี่ยนชนิด หรือขนาดของตัวอักษร ชิดเส้นใต้ ตัวเอียง ตัวหนา การจัดข้อความกึ่งกลาง การใส่สีให้ตัวอักษร เราสามารถทำได้ในส่วนนี้ ดังแสดงในรูปข้างล่าง



รูปที่ 2.9 แสดงการปรับแต่งข้อความ

2.2 การประยุกต์ใช้ Flash Professional 8

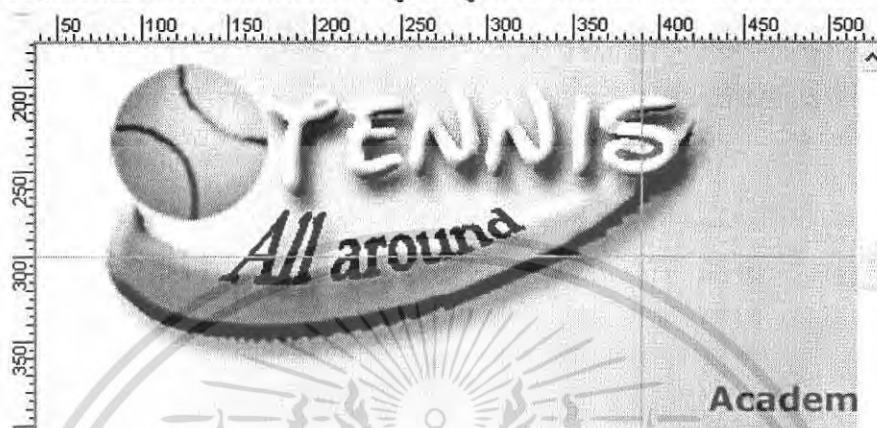
ในการสร้างเว็บไซต์ การที่จะสร้างความสนใจ ความประทับใจเพื่อดึงดูดผู้คนที่เข้ามาใช้งานเว็บไซต์ของเรานั้น สิ่งหนึ่งที่สำคัญก็คือความสวยงาม และสนุกสนาน ที่มีในเว็บเพจ และที่นิยมอย่างมากก็คือภาพเคลื่อนไหว และนี่คือที่มาของการใช้โปรแกรม Flash Professional 8 นำมาประยุกต์กับการสร้างเว็บไซต์ ซึ่งโปรแกรมนี้สามารถสร้างลูกเล่นอันน่าตื่นตาตื่นใจให้กับเว็บไซต์ของเรานั้นได้อย่างมากมาย

2.2.1 การทำงาน

โปรแกรม Flash ได้ยืมแนวคิดจากการสร้างภาพยนตร์มาใช้ โดยชิ้นงานที่คุณสร้างเพื่อนำเสนอต่อผู้เข้าชมเว็บไซต์เปรียบได้กับ ภาพยนตร์หรือมูฟวี่ (Movie) แต่ละส่วนที่ร้อยเรียงเข้าด้วยกันเป็นภาพยนตร์ก็คือ ฉากหรือซีน (Scene) ขณะที่ข้อมูลจะเปรียบเสมือนนักแสดงที่อยู่บนเวทีหรือสแตจ (Stage) และการเคลื่อนไหวต่างๆ ก็จะถูกจำกัดอยู่ภายใน ไทม์ไลน์ (Timeline)

2.2.2 สเตจ

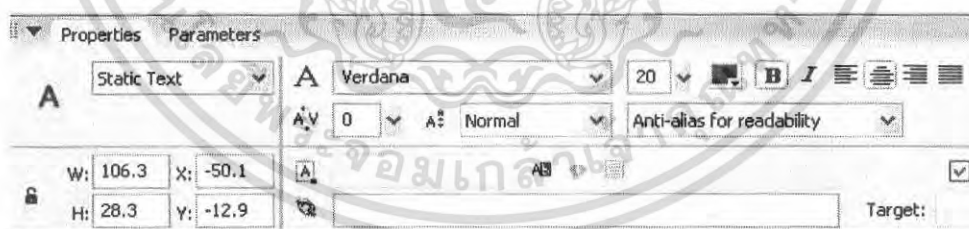
เป็นพื้นที่สำหรับจัดวางและจัดเรียงองค์ประกอบต่าง ๆ ของเว็บไซต์ลงไป โดยพื้นที่ที่เหลื่อมของสเตจนี้เป็นตัวกำหนดเนื้อหาของมูฟวี่ที่ผู้ชมจะได้รับชม ดังภาพข้างล่าง



รูปที่ 2.10 แสดงสเตจ

2.2.3 พาเนล Properties

เป็นส่วนที่ใช้สำหรับตรวจสอบและเปลี่ยนแปลงคุณลักษณะต่าง ๆ ของอีอบเจกต์ โดยตัวเลือกและเครื่องมือควบคุมที่มีในพาเนล Properties นี้จะผันแปรไปตามองค์ประกอบที่ถูกเลือก ขณะนั้น ซึ่งเราสามารถเปลี่ยนแปลงแอททริบิวต์หรือลักษณะของข้อความ กราฟฟิก เฟรม แอนิเมชัน และองค์ประกอบอื่นๆ ผ่านพาเนลนี้ได้

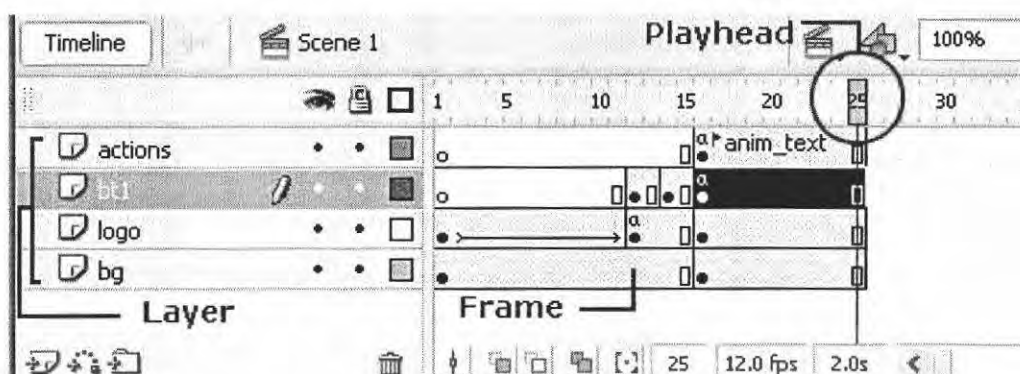


รูปที่ 2.11 แสดงพาเนล properties

2.2.4 ไทม์ไลน์

คือตัวควบคุมลำดับ เวลา และการเคลื่อนไหวของมูฟวี่ ซึ่งส่วนประกอบหลัก ๆ ของพาเนลมีอยู่ 3 ส่วนด้วยกัน ได้แก่ เฟรม เลเซอร์ และ Playhead

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงไทม์ไลน์

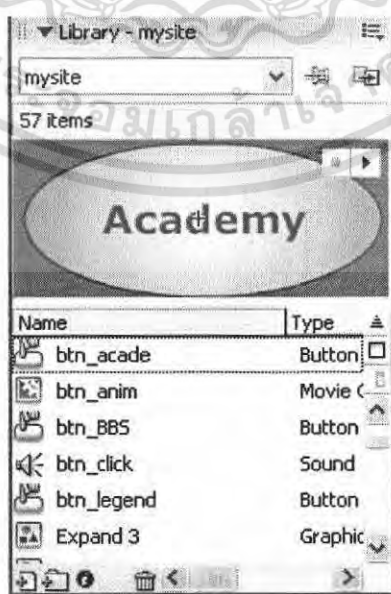
เฟรม ที่อยู่ในไทม์ไลน์แทนถึงการเปลี่ยนแปลงของข้อมูลในแต่ละช่วงเวลา

Playhead เป็นสิ่งที่บ่งบอกถึงเฟรมปัจจุบันที่แสดงอยู่ในสเตรจ ณ ขณะนั้น

เลเยอร์ เปรียบเสมือนแผ่นฟิล์มที่แยกเป็นอิสระจากกัน ในแต่ละแผ่นจะมีองค์ประกอบเป็นตัวของมันเอง สามารถนำมาเรียงซ้อนทับกับแผ่นอื่น ๆ และรวมกันออกมาเป็นเฟรม ๆ หนึ่งได้

2.2.5 พาเนล Library

จะใช้จัดเก็บซิมไบลต่าง ๆ ในมูฟวี่ ซึ่งหากเปรียบองค์ประกอบในมูฟวี่ของเราเป็นนักแสดง ซิมไบลก็อาจเทียบได้กับดาวเด่นในมูฟวี่เลยก็ว่าได้ เพราะการใช้ซิมไบลจะช่วยลดขนาดไฟล์ ประหยัดเวลาในการแก้ไขและจัดการกับไฟล์ของเราได้เป็นอย่างดี

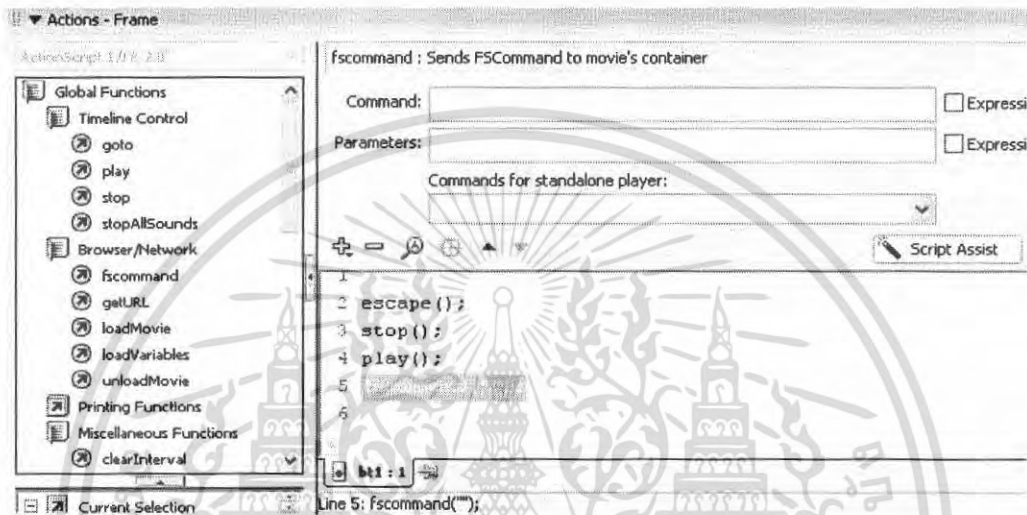


รูปที่ 2.13 แสดงพาเนล Library

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.6 พาเนล Actions

ใช้ในการเพิ่ม ActionScript ให้มูฟวี่ โดย ActionScript เป็นภาษาสคริปต์ของโปรแกรม Flash ที่ใช้สำหรับเพิ่มการปฏิสัมพันธ์ (Interactivity) แบบที่มีความซับซ้อน ควบคุมการท่องไปยังส่วนต่าง ๆ ภายในเว็บไซต์ รวมถึงการกำหนดฟังก์ชันขั้นสูงหลาย ๆ อย่างที่มีในโปรแกรม Flash



รูปที่ 2.14 แสดงพาเนล Actions

2.2.7 พาเนล Behaviors

ช่วยให้เราควบคุม ActionScript ได้อย่างมีประสิทธิภาพโดยไม่ต้องลงมือเขียนโค้ดเอง Behavior คือชุดของ ActionScript ที่ได้ถูกจัดเตรียมเอาไว้ให้แล้ว ซึ่งแม้ว่าแอ็คชั่นที่เราต้องการใช้จะมีความซับซ้อน แต่เราก็สามารถหยิบไปใช้งานได้อย่างสะดวกและง่ายดาย

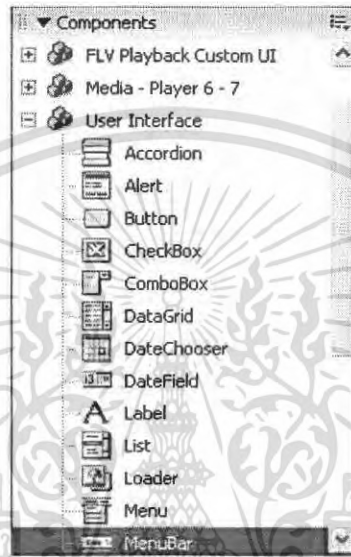


รูปที่ 2.15 แสดงพาเนล Behaviors

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.8 พาเนล Components

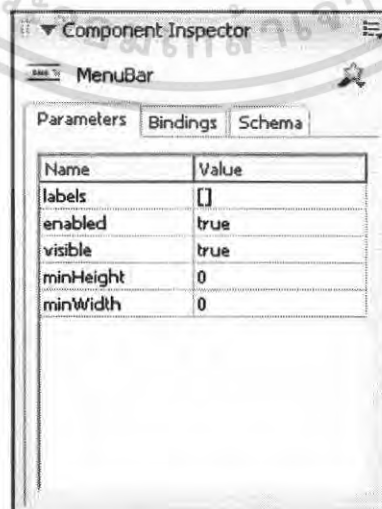
จะเป็นชุดของอ็อบเจกต์พิเศษที่รวม ActionScript สำหรับควบคุม Behavior เอาไว้ คอมพิวเตอร์ที่ประกอบไปด้วยตัวควบคุมอย่างง่าย เช่น ปุ่ม (Button) เช็คบ็อกซ์ (Checkbox) และยังรวมไปถึงตัวควบคุมแบบซับซ้อน เช่น หน้าต่างแบบที่สามารถเลื่อนดูเนื้อหาได้ เป็นต้น



รูปที่ 2.16 แสดงพาเนล Components

2.2.9 พาเนล Components Inspector

จะใช้กำหนดค่าพารามิเตอร์ให้กับคอมพิวเตอร์ที่แบบต่างๆ เพื่อให้ตรงกับความต้องการด้านการออกแบบของเราให้มากที่สุด



รูปที่ 2.17 แสดงพาเนล Components Inspector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ภาษา SQL

ภาษา SQL อ่านว่า “เอสคิวแอล” (SQL) มาจาก Structured Query Language เป็นภาษาทางด้านฐานข้อมูลที่สามารถสร้างและปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์ (relational database) โดยเฉพาะและเป็นภาษาที่มีลักษณะคล้ายกับภาษาอังกฤษ ถูกพัฒนาขึ้นจากแนวคิดของ relational calculus และ relational algebra เป็นหลัก ภาษา SQL เริ่มพัฒนาครั้งแรกโดย almaden research center ของบริษัท IBM โดยมีชื่อเริ่มแรกว่า “ซีเควล” (Sequel) ต่อมาได้เปลี่ยนชื่อเป็น “เอสคิวแอล” (SQL) ภายหลังในปีค.ศ. 1986 American National Standards Institute (ANSI) จึงได้กำหนดมาตรฐานของ SQL ขึ้นและเป็นที่ยอมรับกันอย่างแพร่หลายในปัจจุบัน

2.3.1 ประเภทของคำสั่งของภาษา SQL

ภาษา SQL เป็นภาษาที่ใช้งาน แบ่งออกเป็น 3 ประเภท คือ

- Data Definition Language : DDL

ประกอบด้วยคำสั่งที่ใช้ในการกำหนดโครงสร้างข้อมูลว่ามีคอลัมน์อะไร แต่ละคอลัมน์เก็บข้อมูลประเภทใด รวมถึงการเพิ่มคอลัมน์ การกำหนดดัชนี การกำหนดตารางเสมือนของผู้ใช้

- Data Manipulation Language : DML

ประกอบด้วยคำสั่งที่ใช้ในการเรียกใช้ข้อมูล การเปลี่ยนแปลงข้อมูล การเพิ่มหรือลบข้อมูล

- Data Control Language : DCL

ประกอบด้วยคำสั่งที่ใช้ในการควบคุม การเกิดภาวะพร้อมกัน หรือการป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกัน และคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูลด้วยการกำหนดสิทธิ์ของผู้ใช้ที่แตกต่างกัน

2.3.2 ชุดคำสั่ง DDL

2.3.2.1 การสร้างฐานข้อมูล

การสร้างฐานข้อมูลต้องใช้คำสั่ง

```
CREATE DATABASE databaseName;
```

```
databaseName = ชื่อฐานข้อมูล
```

2.3.2.2 การสร้างตาราง

การสร้างตารางในภาษา SQL จะใช้คำสั่ง CREATE TABLE ซึ่งเป็นคำสั่งที่ใช้ในการสร้างตารางขึ้นมาใหม่ คำสั่ง CREATE TABLE จะกำหนดชื่อตารางและกำหนดลักษณะข้อมูลเป็นคอลัมน์ต่างๆที่ตั้งขึ้นในตารางรวมไปถึงชนิดของข้อมูลของแต่ละคอลัมน์นั้น ในโครงสร้างของคำสั่งการสร้างตารางมีรูปแบบดังต่อไปนี้

```
CREATE TABLE <table name>
(<column name> [<size>] [[ constraint <constraint_name>]constraint_type]
[,<column name> <data type>[<size>],.....]);
```

CREATE TABLE เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการสร้างตาราง

table name	=	ชื่อตารางที่ต้องการสร้าง
column name	=	ชื่อของคอลัมน์แต่ละคอลัมน์
data type	=	ชนิดข้อมูลของคอลัมน์นั้นๆ
constraint	=	ข้อกำหนดของคอลัมน์
constraint_name	=	ชื่อของข้อกำหนดที่ต้องการสร้างให้กับคอลัมน์
constraint_type	=	ประเภทของข้อกำหนด

ตัวอย่าง การสร้างตารางพนักงานขาย

```
CREATE TABLE SALESTAB
(SALENO integer,
SALENAME char (10),
ADDRESS char (10),
SALECOM decimal);
```

จากคำสั่งจะทำให้ได้ตารางพนักงานขายที่มีคอลัมน์ SALENO มีชนิดข้อมูลเป็น integer คอลัมน์ SALENAME มีชนิดข้อมูลเป็น char มีความยาว 10 ตัวอักษร คอลัมน์ ADDRESS มีชนิดข้อมูลเป็น char มีความยาว 10 ตัวอักษรและคอลัมน์ SALECOM มีชนิดข้อมูลเป็น decimal

ผลของคำสั่งการสร้างตารางจะได้ตารางพนักงานขายที่ยังไม่มีข้อมูลใดๆ เป็นเพียงแต่โครงของตารางเท่านั้นดังนี้

SALENO	SALENAME	ADDRESS	SALECOM
--------	----------	---------	---------

รูปที่ 2.18 แสดงโครงของตาราง

ตารางที่ 2.1 ตัวอย่าง ชนิดข้อมูล (Data type)

ชื่อ	ขนาด (byte)	ความหมาย
VARCHAR(M)	M	ข้อมูลประเภทตัวอักษรที่มีความยาว 1 – 255 เหมาะกับข้อมูลประเภท ชื่อ-นามสกุล ที่อยู่ เบอร์โทรศัพท์ สามารถกำหนด attribute เป็น BINARY ที่มีผลกับการค้นหาไม่คำนึงถึงขนาดตัวพิมพ์
TINYINT	1	ข้อมูลตัวเลขขนาด 8 บิต สามารถกำหนด attribute ได้เป็น <ul style="list-style-type: none"> - UNSIGNED เก็บค่าตัวเลขแบบไม่มีเครื่องหมายนำหน้า ตั้งแต่ 0 – 255 - UNSIGNED ZEROFILL เหมือนข้างบนแต่ถ้าหลักไม่ครบตามจำนวนที่กำหนด จะมีการเติม 0 ให้ครบเอง เช่น 35 เป็น 035 ถ้าไม่เลือกจะกำหนดเป็น SIGNED และเสียไปบิตสำหรับเก็บเครื่องหมายไป ทำให้เก็บได้ในช่วง -128 ถึง 128
TEXT	มากกว่า 64 KB	สามารถเก็บข้อมูลได้ถึง 65535 ตัวอักษร หรือ 64 Kbit เหมาะสำหรับเก็บข้อมูลขนาดใหญ่ เช่น บทความ
DATE	3	ข้อมูลประเภทวันที่ YYYY-MM-DD
INT	4	ข้อมูลเลขจำนวนเต็มขนาด 32 bit กรณีคิดเครื่องหมายมีค่าตั้งแต่ -2147483648 ถึง 2147483647 ไม่คิดเครื่องหมายมีค่า 0 ถึง 4294967295
DECIMAL[(M,D)]	M+1 หรือ M+2	ข้อมูลชนิดเลขจำนวนจริง โดยต้องกำหนดความยาวข้อมูลในรูปแบบ (M,D) คือ ตัวเลขจำนวนเต็มก็หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 (ต่อ) ตัวอย่าง ชนิดข้อมูล (Data type)

		ทศนิยมที่หลัก
DATETIME	8	ข้อมูลวันที่และเวลา YYYY-MM-DD HH:MM:SS
YEAR[(2/4)]	1	กำหนด 2 จะเป็นแบบ YY กำหนด 4 เป็นแบบ YYYY
CHAR(M)	M	ข้อมูลประเภทสตริงมีความยาวไม่เกิน 255 ตัวอักษร ไม่สามารถปรับระดับความยาวได้ attribute เป็น BINARY จะไม่คำนึงถึงขนาดตัวพิมพ์
MEDIUMTEXT	16 MB	เหมือนข้อมูล TEXT แต่เก็บได้ 16 MB

2.3.2.3 ข้อจำกัด (constraints)

- NOT NULL คือ การกำหนดข้อมูลของคอลัมน์ใด คอลัมน์หนึ่งมีค่าว่างไม่ได้โดยใช้คำว่า “NOT NULL” เช่น คอลัมน์ที่เป็นคีย์หลัก (primary key) หรือต้องการให้ลูกค้าทุกคนในตาราง ลูกค้าต้องมีข้อมูลชื่อ โดยทั่วไปการสร้างตารางถ้าในคอลัมน์ไม่ระบุคำว่า “NOT NULL” คอลัมน์นั้นจะถูกระบุให้เป็นค่า NULL โดยปริยาย(DEFAULT)

ตัวอย่าง สมมติว่าต้องการกำหนดให้ตารางพนักงานขายในคอลัมน์ SALENO และ คอลัมน์ SALENAME ไม่ให้เป็นค่าว่าง (NOT NULL) จะสามารถสร้างตารางพนักงานด้วยคำสั่ง ดังนี้

```
CREATE TABLE SALESTAB
(SALENO integer NOT NULL,
SALENAME char(10) NOT NULL,
ADDRESS char(10),
SALECOM decimal);
```

- UNIQUE คือ การสร้างตารางโดยกำหนดให้คอลัมน์นั้นทั้งตารางไม่ให้มีค่าซ้ำกัน โดยใช้คำว่า “UNIQUE” เช่น คอลัมน์รหัสพนักงานที่เป็นคีย์หลัก และ ไม่ต้องการให้มีค่าซ้ำ

ตัวอย่าง สมมติว่าต้องการกำหนดให้ตารางพนักงานขายในคอลัมน์ SALENO และ คอลัมน์ SALENAME ไม่ให้เป็นค่าว่าง (NOT NULL) และไม่ให้มีค่าซ้ำกัน จะสามารถสร้างตารางพนักงานด้วยคำสั่ง ดังนี้

```
CREATE TABLE SALESPEOPLE
(SALENO integer NOT NULL UNIQUE,
SALENAME char(10) NOT NULL UNIQUE,
```

ADDRESS char(10),

SALECOM decimal);

- PRIMARY KEY คือ การกำหนดคีย์หลัก สามารถกำหนดได้ 2 วิธีคือ

1) การกำหนดให้คอลัมน์เดียวเป็นคีย์หลัก

ตัวอย่าง สมมติว่าต้องสร้างตารางพนักงานที่กำหนดให้คอลัมน์ SALENO เป็น คีย์หลัก (primary key) โดยไม่ให้มีค่าซ้ำกัน คอลัมน์ SALENAMEไม่ให้เป็น ค่าว่าง (NOT NULL) และไม่ให้มีค่าซ้ำกัน จะสามารถสร้างตารางด้วยคำสั่งดังนี้

```
CREATE TABLE SALESPEOPLE
```

```
(SALENO integer NOT NULL UNIQUE PRIMARY KEY,
```

```
SALENAME char(10) NOT NULL UNIQUE,
```

```
ADDRESS char(10),
```

```
SALECOM decimal);
```

2) การกำหนดให้คอลัมน์มากกว่า 1 คอลัมน์เป็นคีย์หลัก

ตัวอย่าง สมมติว่าต้องสร้างตาราง NAMEFIELD โดยกำหนดให้คอลัมน์ FIRSTNAME และคอลัมน์ LASTNAME เป็น คีย์หลัก(primary key) จะสามารถสร้างตาราง NAMEFIELD ด้วยคำสั่งดังนี้

```
CREATE TABLE NAMEFIELD
```

```
(FIRSTNAME char(10) NOT NULL ,
```

```
LASTNAME char(10) NOT NULL UNIQUE,
```

```
CITY char(10),
```

```
PRIMARY KEY (FIRSTNAME LASTNAME ));
```

- FOREIGN KEY คือ คีย์นอกเป็นคอลัมน์ของตารางหนึ่งที่ใช้เชื่อมโยงหรืออ้างอิง

ข้อมูลกับอีกตารางหนึ่งที่มีคอลัมน์ที่มีชื่อคอลัมน์เดียวกัน เช่น ลูกค้าในตารางลูกค้า แต่ละคนมีคอลัมน์ SALENO ที่อยู่ตารางพนักงานขายหากกำหนดคีย์นอกเป็นข้อจำกัดในระดับคอลัมน์จะใช้คำสั่ง REFERENCE ต่อท้ายประเภทและขนาดของคอลัมน์ที่เป็นคีย์นอจากตาราง แสดงการอ้างอิงข้อมูล โดยคอลัมน์ที่มีชื่อเดียวกัน เป็นตารางลูกค้าและตารางพนักงานขาย

ตัวอย่าง CUSTOMERSTAB โดยกำหนดให้คอลัมน์ CUSNO เป็น PRIMARY KEY และคอลัมน์SALENO เป็นคีย์นอก (foreign key) ที่ใช้เชื่อมโยงหรืออ้างอิงข้อมูลกับตารางพนักงานขาย (SALESTAB) โดยใช้คำสั่งดังนี้

```
CREATE TABLE CUSTOMERSTAB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(CUSNO integer NOT NULL PRIMARY KEY,
 CUSNAME char(10) ,ADDRESS char(10),SALENO integer,
 FOREIGN KEY (SALENO) REFERENCES SALESTAB(SALENO));

2.3.2.4 การลบตาราง

เมื่อต้องการลบโครงสร้างตารางที่ถูกสร้างขึ้นจะสามารถทำได้ด้วยคำสั่ง DROP TABLE ซึ่งมีรูปแบบทั่วไปดังนี้

DROP TABLE <table name>[CASCADE CONSTRAINTS];

DROP TABLE = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการลบโครงสร้างตาราง

table name = ชื่อตารางที่ต้องการลบ

CASCADE CONSTRAINTS = ระบบจัดการฐานข้อมูลจะทำการลบข้อจำกัด (constraint) ที่มีการอ้างอิงตารางทิ้งไปให้ด้วยทั้งหมด

ตัวอย่าง ถ้าต้องการลบตารางพนักงานขาย (SALESTAB) จะใช้คำสั่งดังนี้

DROP TABLE SALESTAB;

2.3.2.5 การเปลี่ยนแปลงโครงสร้างตาราง

เมื่อสร้างโครงสร้างตารางแล้ว ต้องการเปลี่ยนแปลงโครงสร้างตารางที่มีการสร้างไว้ใหม่ เช่น ต้องการเพิ่มหรือลบบางคอลัมน์ที่เป็นโครงสร้างหลักของตารางออกหรือต้องการเปลี่ยนแปลงประเภทข้อมูลของคอลัมน์สามารถแก้ไขโครงสร้างข้อมูลได้ด้วยคำสั่งการเปลี่ยนแปลงโครงสร้างตาราง รูปแบบของของคำสั่ง ALTER TABLE มี 2 แบบ คือ

- ALTER TABLE ที่ใช้ในการเพิ่มคอลัมน์

- ALTER TABLE ที่ใช้ในการเปลี่ยนชื่อคอลัมน์

มีรูปแบบดังนี้

ALTER TABLE <table name>Database update(<column_name> data type [SIZE]);

ALTER TABLE =เป็นคำสั่งที่ต้องมีทุกครั้งในการเปลี่ยนแปลงโครงสร้างตาราง

table name = ชื่อตารางที่จะเปลี่ยนแปลง

Database update = คำสั่งการเปลี่ยนแปลง

column_name = ชื่อคอลัมน์

data type [SIZE] = ชนิดข้อมูลและขนาดของข้อมูล

ตัวอย่าง ถ้าต้องการเปลี่ยนแปลงโครงสร้างตาราง โดยการการเพิ่มคอลัมน์ลงไปบน
โครงสร้างตารางเดิมจะใช้คำสั่งดังนี้

```
ALTER TABLE SALESPeOPLE ADD SALESTAB_FAX CHAR(15);
```

ผลของคำสั่งจะทำให้ตารางพนักงานขายมีคอลัมน์ SALESTAB_FAX ที่มีชนิดข้อมูลเป็น
ตัวอักษร (char) มีความยาว 15 ตัวอักษรเพิ่มขึ้น

ตัวอย่าง ถ้าต้องการเปลี่ยนแปลงโครงสร้างตาราง โดยการเปลี่ยนชื่อคอลัมน์จะใช้คำสั่งดังนี้

```
ALTER TABLE SALESPeOPLE RENAME ADDRESS TO COUNTRY;
```

ผลของคำสั่งจะทำให้ตารางพนักงานขาย ที่เดิมมีคอลัมน์ชื่อ ADDRESS ต้องเปลี่ยนชื่อเป็น
COUNTRY แทน

2.3.3 ชุดคำสั่ง DML

2.3.3.1 คำสั่งการเพิ่มข้อมูล

คำสั่งการเพิ่มข้อมูลในตารางจะใช้คำสั่ง INSERT จะมีอยู่ 2 รูปแบบคือ การเพิ่ม
ข้อมูลเข้าไปทีละแถวและการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล

- คำสั่งการเพิ่มข้อมูลทีละแถวโดยระบุข้อมูลที่ จะ INSERT เข้าไป โดยตรงรูปแบบของ
คำสั่ง เป็นดังนี้

```
INSERT INTO <tablename>[(column 1, column 2,...)]VALUE(<value1,value2, ...>);
```

INSERT INTO = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล

tablename = ชื่อตารางที่จะเพิ่มข้อมูล

column 1, column 2,... = คอลัมน์ที่ต้องการเพิ่มข้อมูล

value1,value2, = ค่าข้อมูลของแต่ละคอลัมน์ที่ต้องการเพิ่ม

ตัวอย่าง ถ้าต้องการจะใส่ข้อมูลทุกคอลัมน์ลงในตารางลูกค้า

```
INSERT INTO SALESTAB VALUES( 1001, "Chaiwat", "Bangkok",0.12);
```

ตัวอย่าง ถ้าต้องการจะใส่ข้อมูลบางคอลัมน์ เช่น ชื่อเมือง"Bangkok" ชื่อลูกค้า "Arlee" และ
หมายเลขลูกค้า 2001 ลงในตารางลูกค้า ใช้คำสั่งดังนี้

```
INSERT INTO CUSTOMERSTAB(ADDRESS,CUSNAME,CUSNO)
```

```
VALUES('Bangkok','Arlee', 2001);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คำสั่งการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล ในภาษา SQL สามารถใช้ คำสั่ง INSERT ในการนำค่าหรือหาค่าจากตารางหนึ่งแล้วไปใส่ไว้ในอีกตารางหนึ่งได้ โดยได้ค่านั้นมาจากการสอบถามข้อมูล รูปแบบเป็นดังนี้

```
INSERT INTO <table name>[(column 1, column 2,...)]SELECT statement;
INSERT INTO          = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล
tablename            = ชื่อตารางที่จะเพิ่มข้อมูล
SELECT statement     = ประโยคคำสั่ง SELECTที่ต้องการข้อมูลอีกตารางหนึ่ง
```

ตัวอย่าง ถ้าต้องการใส่ข้อมูลพนักงานลงในตาราง BANGKOKSTAFF โดยข้อมูลที่จะใส่ลงไปนั้น ได้มาจากตารางพนักงานขายที่อาศัยอยู่ใน “Bangkok”

```
INSERT INTO BANGKOKSTAFF
SELECT *
FROM SALESTAB
WHERE ADDRESS = 'Bangkok';
```

ผลของคำสั่งนี้จะทำให้ได้ข้อมูลพนักงานที่อยู่ในเมือง Bangkok(ADDRESS = ‘Bangkok’) ทั้งหมดไปใส่ไว้ในตาราง BANGKOKSTAFF โดยตาราง BANGKOKSTAFF” ได้ถูกสร้างไว้แล้ว ด้วยคำสั่ง CREATE TABLE การสร้างตาราง BANGKOKSTAFF จะต้องสร้างให้มี 4 คอลัมน์และมีชนิดข้อมูลตรงกับคอลัมน์ของตารางพนักงานขาย

2.3.3.2 คำสั่งปรับปรุงแถวข้อมูล

หลังจากที่ป้อนข้อมูลเข้าไปเก็บไว้ในตารางแล้วกรณีที่ต้องการปรับปรุงแก้ไขข้อมูลสามารถทำได้ด้วยภาษาเอสคิวแอล การปรับปรุงแถวข้อมูลเป็นการปรับปรุงหรือแก้ไขค่าคอลัมน์ ซึ่งในคำสั่งปรับปรุงข้อมูลอาจมีมากกว่า 1 คอลัมน์ในแถวทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลังคำว่า WHERE รูปแบบของคำสั่งปรับปรุงแถวข้อมูลมี ดังนี้

```
UPDATE<tablename>SET<column1>[,column2,...]=<expression >WHERE<condition>;
UPDATE          = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการปรับปรุงข้อมูล
table name      = ชื่อตารางที่ต้องการปรับปรุง
SET <column >  = ชื่อคอลัมน์ที่ต้องการปรับปรุง
expression      = ค่าข้อมูลที่ต้องการปรับปรุง
WHERE<condition> = เงื่อนไขในการปรับปรุง
```

ตัวอย่าง ถ้าต้องการเปลี่ยนค่า “RATING” ของลูกค้าทั้งหมดในตารางลูกค้าให้เป็น 200 จะต้อง ป้อนคำสั่ง ดังนี้

```
UPDATE CUSTOMERSTAB
```

```
SET RATING = 200;
```

ตัวอย่าง ถ้าต้องการจะเปลี่ยนค่า RATING ให้กับลูกค้าทั้งหมด ที่มีหมายเลขประจำตัวพนักงานขาย(SALENO) เป็น 1001 ให้มีค่า RATING เป็น 200

```
UPDATE CUSTOMERSTAB
```

```
SET RATING = 200
```

```
WHERE SALENO = 1001;
```

2.3.3.3 คำสั่งการลบข้อมูลทิ้งแล้ว

คำสั่งในการลบแถวข้อมูล เป็นคำสั่งที่ใช้ในการลบแถวข้อมูลทุกแถวที่มีเงื่อนไข สอดคล้องกับที่ระบุไว้หลัง WHERE คำสั่งการลบข้อมูลมีรูปแบบทั่วไปดังนี้

```
DELETE FROM <table name>[WHERE<condition>];
```

DELETE FROM = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการลบข้อมูล

table name = ชื่อตารางที่ต้องการลบข้อมูล

WHERE<condition> = เงื่อนไขในการลบข้อมูล

ตัวอย่าง ถ้าต้องการลบแบบมีเงื่อนไข เช่น ต้องการลบพนักงานหมายเลขพนักงาน (SALENO)=1003 ออกจากตารางจะใช้คำสั่งว่า

```
DELETE FROM SALESTAB
```

```
WHERE SALENO = 1003;
```

2.3.3.4 การเรียกค้นดูคอลัมน์ในตาราง

1. การเรียกค้นดูทุกคอลัมน์ในตาราง

คำสั่ง SELECT แบบง่ายมีรูปแบบดังนี้

```
SELECT * FROM <table name>;
```

SELECT * = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูลทุกคอลัมน์

FROM = เป็นการกำหนดว่าให้เรียกดูข้อมูล ได้จากตารางใดบ้าง

table name = ชื่อตารางที่ต้องการเรียกค้นข้อมูล

การเรียกดูข้อมูลสามารถเรียกดูได้มากกว่า 1 คอลัมน์ขึ้นไป โดยถ้ามีมากกว่า 1 คอลัมน์แต่ละคอลัมน์จะต้องคั่นด้วยเครื่องหมายคอมม่า(,) และถ้าต้องการดูทุกคอลัมน์จะใช้เครื่องหมายดอกจัน(*) หลัง SELECT การใช้คำสั่ง SELECT จะใช้ควบคู่กับคำสั่ง FROM เสมอในการเลือกตาราง

ตัวอย่าง ถ้าต้องการดูทุกคอลัมน์ในตารางก็จะใช้เครื่องหมายดอกจัน(*) แทนรายการคอลัมน์ ได้ทั้งหมดได้ดังนี้

```
SELECT * FROM CHECKS;
```

ผลลัพธ์จากคำสั่ง select * จะเป็นการบอกให้นำข้อมูลทั้งจากตาราง CHECKS มาแสดง (from checks) โดยลำดับตามคอลัมน์ในฐานข้อมูล

2. การเรียกค้นข้อมูลเฉพาะคอลัมน์ใดๆในตารางและการเปลี่ยนลำดับคอลัมน์

การใช้คำสั่ง SELECT ในการเรียกค้นข้อมูลเฉพาะคอลัมน์ที่สนใจทำได้โดยใช้เฉพาะคอลัมน์ที่ต้องการดูในส่วน of คำสั่ง SELECT มีรูปแบบ ดังนี้

```
SELECT <column 1, column 2,...> FROM <table name>;
```

SELECT = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล

column 1, column 2,... = เป็นคอลัมน์ที่ต้องการเรียกค้น

FROM = เป็นการกำหนดว่าให้เรียกดูข้อมูลได้จากตารางใดบ้าง

table name = ชื่อตารางที่ต้องการเรียกค้นข้อมูล

ตัวอย่าง ถ้าต้องการแสดงข้อมูลบางคอลัมน์จะใช้ เช่น ถ้าต้องการดูคอลัมน์ CHECK# และ AMOUNT ใช้คำสั่งดังนี้

```
SELECT CHECK#, amount from checks;
```

ผลลัพธ์จะพบว่าเราสามารถใช้อักษรตัวใหญ่และตัวเล็กปนกันในคำสั่งได้ซึ่งอักษรตัวใหญ่และตัวเล็กจะไม่มี ความแตกต่างกันเมื่อนำมาประมวลผล

ตัวอย่าง ถ้าต้องแสดงข้อมูลโดยการเปลี่ยนลำดับคอลัมน์ของข้อมูล จะใช้คำสั่งดังนี้

```
SELECT PAYEE, REMARKS, AMOUNT, CHECK# FROM checks;
```

3. การใช้คำสั่ง SELECT กับ WHERE

```
SELECT <column 1, column 2,...>FROM <table name>[WHERE<condition>];
```

SELECT = เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล

column 1, column 2,... = คอลัมน์ที่ต้องการเรียกค้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FROM = เป็นการกำหนดว่าให้เรียกดูข้อมูลได้จากตารางใดบ้าง
 table name = ชื่อตารางที่ต้องการเรียกค้นข้อมูล
 WHERE<condition> = ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล

4. โอเปอเรเตอร์

เงื่อนไขตามหลักของภาษาเอสคิวแอล จะอยู่หลังคำสั่ง WHERE ซึ่งสามารถเปรียบเทียบตามโอเปอเรเตอร์ในภาษาเอสคิวแอล อาจแบ่งโอเปอเรเตอร์ได้เป็น 4 กลุ่มคือ

- โอเปอเรเตอร์คณิตศาสตร์(Arithmetic Operators)
- โอเปอเรเตอร์เปรียบเทียบ(Comparison Operators)
- โอเปอเรเตอร์อักขระ(Character Operators)
- โอเปอเรเตอร์ตรรกะ(Logical Operators)

- โอเปอเรเตอร์คณิตศาสตร์(Arithmetic Operators) ได้แก่ โอเปอเรเตอร์ที่เป็น plus (+) minus (-), divide (/), multiply (*), and modulo (%)

ตารางที่ 2.2 โอเปอเรเตอร์คณิตศาสตร์

ชื่อ	สัญลักษณ์	ความหมาย
Plus	+	เป็นคำสั่งที่ใช้รวมค่า 2 ค่าเข้าด้วยกัน
Minus	-	ใช้ได้เป็น 2 กรณีคือ 1. การเปลี่ยนเครื่องหมายจากบวกเป็นลบและจากลบเป็นบวก 2. การนำข้อมูลของคอลัมน์หนึ่งไปลบออกจากข้อมูลของอีกคอลัมน์หนึ่ง
Divide	*	เป็นคำสั่งที่ใช้ในการหารข้อมูล
Multiply	/	เป็นคำสั่งที่ใช้ในคูณค่าของข้อมูล
Modulo	%	เป็นคำสั่งที่ได้ผลลัพธ์เป็นเศษที่ได้จากการหาร

- โอเปอเรเตอร์เปรียบเทียบ(Comparison Operators) เป็นโอเปอเรเตอร์ที่จะให้ค่าออกมา 3 ค่า คือ ถูก(TRUE) ผิด (FALSE) ไม่รู้ (Unknown) การไม่รู้หมายถึง ถ้านำข้อมูลที่มีค่าไปเปรียบเทียบกับข้อมูลที่เป็น NULL ตัวเปรียบเทียบจะให้ค่าไม่รู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โอเปอเรเตอร์ตัวอักษร (Character Operators) โอเปอเรเตอร์ LIKE เป็นการค้นหาข้อมูลของคอลัมน์ที่เก็บข้อมูลประเภทตัวอักษร โดยไม่ทราบค่าข้อมูลทั้งหมดหรือรู้เพียงบางตัวอักษรเท่านั้น โอเปอเรเตอร์ LIKE จะระบุต่อท้ายชื่อคอลัมน์ที่เป็นเงื่อนไข ใช้สัญลักษณ์ที่เป็นตัวช่วยในการค้นหาข้อมูลที่เรียกว่า ไวลด์การ์ด (WILD Card) สัญลักษณ์ดังกล่าวประกอบด้วย % และ _ (underscore) โดยข้อมูลบางส่วนที่ใช้ในการค้นหาพร้อมกับสัญลักษณ์ทั้งสองนี้จะต้องมีเครื่องหมาย ‘ ‘ กำกับเสมอ

- โอเปอเรเตอร์ตรรกะ (Logical Operator) เป็นตัวโอเปอเรเตอร์ที่ใช้ในการเปรียบเทียบเชื่อมโยงค่า 2 ค่า

ตารางที่ 2.3 โอเปอเรเตอร์ตรรกะ

ชื่อ	ความหมาย
AND	เป็นตัวโอเปอเรเตอร์ ที่ใช้เชื่อมโยงค่า 2 ค่า โดยถ้าค่าหนึ่งเป็น TRUE อีกค่าหนึ่งเป็น TRUE จะให้ค่า TRUE ออกมา แต่ถ้ามีค่าใดค่าหนึ่งเป็น FALSE จะให้ค่าเป็น FALSE
OR	ถ้าค่าหนึ่งเป็น TRUE ไม่ว่าอีกค่าหนึ่งเป็นอะไร จะให้ค่า TRUE ออกมา แต่ถ้ามีเป็น FALSE ทั้งหมดจะให้ค่าเป็น FALSE
NOT	ในการเปรียบเทียบถ้าสิ่งทีนำมาเปรียบเทียบเป็นจริงจะได้ผลลัพธ์ออกมาเป็นเท็จ แต่ถ้าสิ่งทีนำมาเปรียบเทียบเป็นเท็จผลที่ได้จะออกมาเป็นจริง
IN และ BETWEEN	เป็นการกำหนดเซตของสิ่งทีต้องการค้นหา โดยการกำหนดชื่อของสมาชิกเซตลงไปในช่วงเล็บและแยกจากกันด้วยคอมม่า
BETWEEN...AND...	เป็นการกำหนดเงื่อนไขของคอลัมน์ระหว่างค่าสองค่า ซึ่งค่าสองทีอยู่ระหว่างคำสั่ง BETWEEN...AND... นั้นจะมีความหมายว่าเท่ากับหรือมากกว่าและเท่ากับหรือน้อยกว่า

5. การเรียกดูข้อมูลแบบซ้อนกัน

การเรียกดูข้อมูลแบบซ้อนกัน(subqueries) เป็นการสร้างคำสั่ง SELECT ซ้อนกัน การเรียกดูข้อมูลแบบซ้อนกันมีจุดประสงค์เพื่อลดภาระในการเชื่อมตารางที่ต้องใช้ หน่วยความจำเป็นจำนวนมาก คำสั่งย่อยนี้สามารถสร้างหลังคำสั่ง WHERE มีรูปแบบ ดังนี้

```
SELECT [*] <column 1, column 2,...>
```

```
FROM <table name>
```

```
[WHERE<column list = <Select Statement>]
```

SELECT = คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล

column 1, column 2,... = คอลัมน์ที่ต้องการเรียกค้น

FROM = การกำหนดว่าให้เรียกดูข้อมูล ได้จากตารางใดบ้าง

table name = ชื่อตารางที่ต้องการเรียกค้นข้อมูล

WHERE<condition> = ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล

Select Statement = ส่วนของคำสั่งที่เรียกค้นข้อมูลตามเงื่อนไข

การทำงานของคำสั่งย่อยที่ใช้ในการระบุเงื่อนไขหรือเรียกข้อมูลจะทำจากคำถาม ย่อยด้านในสุด ผลที่ได้จะเป็นค่ากลับมาให้กับค่าที่อยู่หน้าเครื่องหมายเท่ากับ เพื่อ เรียกค้นข้อมูล(SELECT) ตามต้องการ

ตัวอย่าง ถ้าต้องการหาคำสั่งซื้อของพนักงานขายที่อาศัยอยู่ใน Bangkok

```
SELECT *
FROM ORDERSTAB
WHERE SALENO IN
(SELECT SALENO
FROM SALETAB
WHERE ADDRESS = 'Bangkok');
```

2.3.4 ชุดคำสั่ง DCL

เป็นชุดคำสั่งที่ใช้ในการควบคุม ความถูกต้องของข้อมูล หรือการกำหนดสิทธิให้แก่ผู้ใช้แต่ละคนในการเข้าถึงฐานข้อมูล

2.3.4.1 การสร้างรายชื่อและการกำหนดสิทธิ์

สามารถทำได้ 2 วิธี โดยการกำหนดสิทธิ์โดยตรงและการกำหนดผ่านทางคำสั่ง GRANT

- GRANT เป็นการกำหนดสิทธิ์ให้แก่ผู้ใช้ที่ไม่มีรายชื่ออยู่ในฐานข้อมูลของ MySQL ซึ่งเราสามารถกำหนดสิทธิ์ของผู้ใช้งานได้ 4 ระดับ

1. Global level การกำหนดสิทธิ์ในระดับโกลบอลมีผลต่อทุกๆ ฐานข้อมูลในเซิร์ฟเวอร์ โดยจะเก็บข้อมูลในเทเบิล user ในฐานข้อมูล ชื่อ mysql
2. Database level การกำหนดสิทธิ์ในระดับฐานข้อมูลมีผลต่อทุกๆ เทเบิลในฐานข้อมูล โดยจะเก็บข้อมูลในเทเบิล db และ host ในฐานข้อมูล ชื่อ mysql
4. Table level การกำหนดสิทธิ์ในระดับเทเบิลมีผลต่อทุกๆ คอลัมน์ในเทเบิล โดยจะเก็บข้อมูลในเทเบิล tables_priv ในฐานข้อมูล ชื่อ mysql
5. Column level การกำหนดสิทธิ์ในระดับคอลัมน์มีผลสำหรับแต่ละคอลัมน์บนเทเบิล โดยจะเก็บข้อมูลในเทเบิล column_priv ในฐานข้อมูล ชื่อ mysql

รูปแบบคำสั่ง

```
GRANT priv_type [(column_list)] [, priv_type [(column_list)] ...]
```

```
ON { tbl_name | * | *.* | db_name.* }
```

```
TO user_name [ IDENTIFIED BY 'password' ]
```

```
[, user_name [IDENTIFIED BY 'password'] ...]
```

```
[ WITH GRANT OPTION ]
```

priv_type = ประเภทของสิทธิ์ (privileges) ที่ต้องการกำหนดให้แก่ผู้ใช้ซึ่งแบ่งออกเป็น 3 กลุ่ม คือ สิทธิ์สำหรับผู้ใช้ ผู้บริหารระบบและสิทธิ์อื่นๆ

column_list = ชื่อคอลัมน์

tbl_name = ชื่อเทเบิล

db_name = ชื่อฐานข้อมูล

user_name = ชื่อผู้ที่ต้องการสร้างหรือกำหนดสิทธิ์

WITH GRANT OPTION ถ้ามีการกำหนดจะทำให้ผู้ใช้มอบสิทธิ์ที่ตนมีแก่ผู้อื่น

ตัวอย่างคำสั่ง

```
GRANT ALL ON *.* TO username@localhost IDENTIFIED by 'password';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างนี้เป็นการกำหนดสิทธิ์ให้แก่ผู้ใช้ ชื่อ “username” (ในกรณีที่ยังไม่มีชื่ออยู่ในฐานข้อมูลจะทำการสร้างขึ้น) โดยกำหนดให้รับสิทธิ์ ALL คือ มีสิทธิทุกๆสิทธิ์ ON *.* คือ ในทุกเทเบิลและทุกฐานข้อมูลมีพาเสิร์ฟิด ส่วน “localhost” คือ การกำหนดเครื่องที่อนุญาตให้ทำการเชื่อมต่อได้ อาจกำหนดเป็นหมายเลขไอพีหรือเครื่องหมาย ไม้ล่การ์ด(wild card)

- REVOKE

เป็นคำสั่งที่ใช้ในการถอนสิทธิ์จากผู้ใช้ รูปแบบคล้ายกับ คำสั่ง GRANT

```
GRANT priv_type [(column_list)] [, priv_type [(column_list)] ...]
ON { tbl_name | * | *.* | db_name. * }
FROM user_name [, user_name ...]
```

เช่น REVOKE ALL ON DB1. FROM username;

การถอนสิทธิ์ในการใช้งานฐานข้อมูล DB1 ออกจากผู้ใช้ ชื่อ “username”

2.3.4.2 การลบรายชื่อผู้ใช้

โดยใช้คำสั่ง DELETE เพื่อทำการลบรายชื่อผู้ใช้ออกจากเทเบิล “user” ฐานข้อมูล “mysql”

```
DELETE FROM user WHERE user='username' AND host='localhost';
```

2.4 ภาษา PHP

PHP เป็นภาษาสคริปต์แบบเซิร์ฟเวอร์ไซด์ (server-side script language) หมายถึง การประมวลผลจะกระทำที่บนเครื่องเซิร์ฟเวอร์แล้วจึงสร้างผลลัพธ์เป็นภาษา HTML ส่งมาให้กับเครื่องไคลเอนต์(client) ทำการแสดงผล เพื่อเป็นการลดภาระของเครื่องไคลเอนต์ โดยที่สามารถเขียนโค้ด PHP ลงในโค้ด HTML ได้เลย ด้วยการใส่แท็ก <?php เป็นแท็กเปิด และ ?> เป็นแท็กปิด หรือเขียนโดยใช้โค้ด PHP อย่างเดียว และทำการบันทึกเป็นไฟล์นามสกุล .php .php3 หรือ .phtml ขึ้นอยู่กับการที่ได้กำหนดลงบนเซิร์ฟเวอร์

ซึ่ง PHP เป็นภาษาที่ง่ายในการเขียนและมีความสามารถในการนำข้อมูลมาจากเซิร์ฟเวอร์ฐานข้อมูลมาแสดงผลบนเว็บเพจ จึงเหมาะกับการนำมาใช้ในทำเว็บบอร์ด ทำเว็บเพจเพื่อการพาณิชย์ ตลอดจนการสร้างเว็บแอปพลิเคชันต่าง ๆ

2.4.1 การเขียนโค้ด PHP

เริ่มเปิดซิ่นแท็ก (Syntax) ด้วย "<?php" และปิดท้ายด้วย ">" ในส่วนของการคอมเมนต์ใน PHP จะใช้ "/*" เพื่อละเว้นคำสั่งหลังจากเครื่องหมายในบรรทัดนั้นหรือใช้ "/*(ข้อความ)... */" สำหรับละเว้นคำสั่ง โดยที่บรรทัดแต่ละบรรทัดที่จบชุดโค้ดของ PHP จะปิดด้วยตัวซีมีโคลอน(";")

ตัวอย่าง สคริปต์ PHP จะเป็นการส่งข้อความ "Hello!" พร้อมทั้งคอมเมนต์คำสั่ง "echo "Hi Friend" " และ "echo "Hi everyone" "ไปที่เบราเซอร์

```
<?php
echo "Hello!";
//echo "Hi Friend";
/*echo "Hi everyone"*/
?>
```

-**ตัวแปร (variables)** มีไว้ใช้ในการเก็บค่าทั้งตัวเลข อักษร หรือผลลัพธ์ ตัวแปรทั้งหมดใน PHP จะเริ่มต้นด้วยสัญลักษณ์ "\$" ซึ่งตัวแปรเหล่านี้อาจจะประกอบไปด้วย ตัวอักษร ตัวเลข หรือ อารเรย์

ตัวอย่างวิธีรวม 2 ตัวแปรเข้าด้วยกัน ให้ใช้เครื่องหมาย (.) ดังตัวอย่างข้างล่าง

```
<?php
$text="Hello Friend!!";
$text2="How R U?";
echo $text . " " . $text2 ;
?>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์ที่ได้ คือ Hello Friend!! How R U?

2.4.2 กฎในการกำหนดชื่อให้แก่ตัวแปร

- ชื่อตัวแปร อักขระตัวแรกต้องเป็นตัวอักษร หรือ underscore "_" ห้ามเป็นตัวเลข
- ชื่อตัวแปร ประกอบด้วยตัว a-Z 0-9 และ _ (underscores) เท่านั้น
- การใช้ตัวพิมพ์เล็กและพิมพ์ใหญ่นั้นจะไม่เหมือนกัน
- ชื่อตัวแปร ที่มีมากกว่า 1 คำ ให้ใช้ "_" เป็นตัวเชื่อม (\$my_string) หรือใช้ตัวอักษร

ตัวใหญ่ แทน (\$myString)

2.4.3 คำสั่งที่รับส่งตัวแปร

ตัวอย่าง ฟอรัม HTML

```
<html><body><form method="POST" action="register.php" >
กรอกชื่อ: <input type="text" name="name" />
กรอกอายุ: <input type="text" name="age" />
<input type="submit" value="Submit"/>
</form></body></html>
```

ตัวอย่างฟอรัม HTML ข้างบนนี้มี 2 ช่องรับข้อความ (text field) กับปุ่มส่งค่า (submit button) เมื่อผู้ใช้เติมค่าเข้าไปและกดปุ่ม submit ไฟล์ register.php จะถูกเรียกขึ้นมา ตัวอย่างโค้ดภายในไฟล์ "register.php"

```
<?php
echo $_POST["name"];
echo $_POST["age"];
?>
```

ผลลัพธ์ของสคริปต์ตัวอย่าง

(ชื่อที่กรอกลงใน text name)(อายุที่กรอก text age)

- ตัวแปร \$_GET

ตัวแปร \$_GET เป็นตัวแปรที่รับและส่งค่าตัวแปร โดย HTTP GET การส่งค่าจากฟอร์มด้วยวิธี \$_GET ซึ่งค่าของตัวแปรห้ามส่งเกิน 100 ตัวอักษร โดยที่ \$_GET จะแสดงค่าตัวแปรไว้ที่ URL ให้ผู้ใช้เห็นจึงไม่สมควรนำมาใช้ในการส่งข้อมูลที่เป็นความลับ

```
<html><body>
<form method="GET" action="register.php" >
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<input type="text" name="name" />
<input type="text" name="age" />
<input type="submit" value="Submit"/>
</form></body></html>

```

URL ของหน้าที่ทำหน้าที่ในการรับค่าตัวแปร คือไฟล์ "register.php" จะแสดงให้เห็นเป็น "www.php.net/register.php?name=potchara&age=21" ตัวแปรจะถูกค้นไว้ด้วยเครื่องหมายคำถาม "?" และถ้ามีตัวแปรมากกว่าหนึ่งตัวจะมีเครื่องหมาย "&" คั่นระหว่างตัวแปรแต่ละตัว ไฟล์ register.php จะทำหน้าที่รับค่าและแสดงผล โดยจะต้องใช้ตัวแปร \$_GET เพื่อแสดงค่าที่ได้มา ตัวอย่าง register.php

```

<?php
echo $_GET["name"];
echo $_GET["age"];
?>

```

ผลของสคริปต์ตัวอย่างข้างบนนี้เป็น
potchara21

- ตัวแปร \$_POST

ตัวแปร \$_POST เป็นตัวแปรที่รับและส่งโดย HTTP POST จะแตกต่างจาก HTTP GET ซึ่งผู้ใช้จะไม่สามารถเห็นค่าที่ส่งไปได้และสามารถส่งค่าได้ไม่จำกัด ตัวอย่างแบบฟอร์มเขียนด้วยวิธี \$_POST

```

<html><body>
<form method="POST" action="register.php" >
<input type="text" name="name" />
<input type="text" name="age" />
<input type="submit" value="Submit"/>
</form></body></html>

```

ไฟล์ register.php ซึ่งเป็นหน้ารับค่าและแสดงผล และใช้ตัวแปร \$_POST เพื่อแสดงค่าที่รับมา

```
<?php
echo $_POST["name"];
echo $_POST["age"];
?>
```

2.4.4 การใช้งาน PHP ร่วมกับ MySQL

คำสั่งที่นำ MySQL มาใช้งานผ่านทางภาษา PHP สามารถทำได้โดยการใช้คำสั่งการเชื่อมต่อกับฐานข้อมูล ดังต่อไปนี้

ตารางที่ 2.4 คำสั่งการเชื่อมต่อ

คำสั่ง	ความหมาย
mysql_connect()	เป็นการเชื่อมต่อกันแบบชั่วคราวที่มีการเชื่อมต่อทุกครั้งที่ทำงาานกับสคริปต์และสิ้นสุดการเชื่อมต่อทันทีที่การทำงานของสคริปต์นั้นสิ้นสุดลง หรือพบกับคำสั่ง mysql_close()
mysql_pconnect()	เป็นการเชื่อมต่อแบบถาวร ที่มีการเชื่อมต่อเพียงครั้งเดียว โดยไม่สนใจว่าสคริปต์นั้นทำงานเสร็จสิ้นหรือยัง ซึ่งทำการค้นหาค่าการเชื่อมต่อก่อนหน้านี้ ถ้าพบว่ามี การเชื่อมต่ออยู่แล้วจะทำการใช้ค่าเชื่อมต่อนั้นทันทีโดยไม่มีการต่อซ้ำ

การเลือกฐานข้อมูล

คำสั่งที่ใช้ในการกำหนดฐานข้อมูลที่นำมาเชื่อมต่อ

ตารางที่ 2.5 คำสั่งสร้างและกำหนดใช้งานฐานข้อมูล

คำสั่ง	ความหมาย
mysql_select_db()	โดยส่งค่าเป็น TRUE เมื่อเลือกฐานข้อมูลสำเร็จแต่ถ้าไม่สำเร็จจะคืนค่าเป็น FALSE
mysql_create_db()	โดยส่งค่าเป็น TRUE เมื่อสร้างฐานข้อมูลสำเร็จแต่ถ้าไม่สำเร็จจะคืนค่าเป็น FALSE
mysql_drop_db()	โดยส่งค่าเป็น TRUE เมื่อลบฐานข้อมูลสำเร็จแต่ถ้าไม่สำเร็จจะคืนค่าเป็น FALSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำสั่ง

`bool mysql_select_db(string database_name [, resource link_identifier])`

`bool mysql_create_db(string database_name [, resource link_identifier])`

`bool mysql_drop_db(string database_name [, resource link_identifier])`

คำสั่งสร้างหรือลบฐานข้อมูลสามารถสร้างหรือลบโดยสั่งผ่านคำสั่ง `mysql_query()` ได้

ดังเช่น `mysql_query("CREATE DATABASE table");`

ตารางที่ 2.6 การประมวลผลคำสั่ง SQL

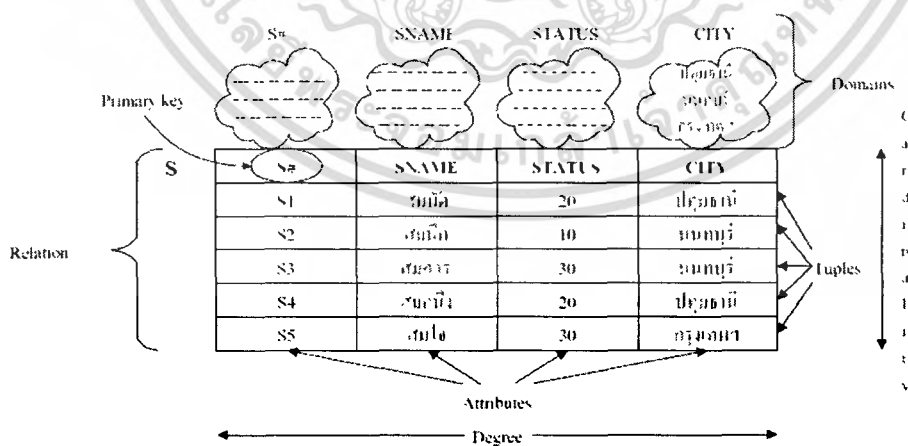
คำสั่ง	ความหมาย
<code>mysql_query()</code>	ฟังก์ชันนี้ทำหน้าที่ประมวลผลคำสั่งภาษาเอสคิวแอล โดยมีการคืนค่าที่เก็บไว้กลับมาเมื่อใช้คำสั่ง SELECT SHOW DESCRIBE หรือ EXPLAIN แล้วไม่มีปัญหา หรือเป็น FALSE เมื่อมีความผิดพลาดเกิดขึ้น แต่ถ้าคำสั่ง CREATE INSERT UPDATE DELETE DROP เมื่อไม่เกิดปัญหาขึ้นจะส่งค่าเป็น TRUE หรือฟังก์ชันนี้จะส่งค่าเป็น FALSE เมื่อผู้ใช้ไม่มีสิทธิ์ในฐานข้อมูล
<code>mysql_unbuffered_query()</code>	คล้ายกับคำสั่ง <code>mysql_query()</code> แต่จะไม่ใช้บัฟเฟอร์ ทำให้ใช้หน่วยความจำไม่มากและประมวลผล โดยที่จะแสดงผลทันทีโดยไม่ต้องรอให้ประมวลผลจบก่อน
<code>mysql_fetch_array()</code>	เป็นคำสั่งที่นำข้อมูลที่ได้จากการใช้งานฟังก์ชัน <code>mysql_query()</code> ออกมาเก็บไว้ในรูปของอาร์เรย์ และสามารถอ้างข้อมูลในฟิลด์ได้ด้วยชื่อของฟิลด์ โดยจะทำการเรียกข้อมูลออกมาทีละ 1 เรคอร์ด
<code>mysql_result()</code>	เป็นคำสั่งที่นำข้อมูลที่ได้จากการใช้งานฟังก์ชัน <code>mysql_query()</code> ซึ่งคำสั่งนี้มีชื่อแตกต่างจากคำสั่ง <code>mysql_fetch_array()</code> ตรงที่ <code>mysql_result()</code> จะทำการแสดงผลออกมาเป็นเซลล์ ๆ เดียว จึงต้องมีการกำหนดเซลล์ที่เลือกนำมาใช้งาน

2.5 ทฤษฎีระบบฐานข้อมูล (Database)

2.5.1 โครงสร้างข้อมูล (Data Structure)

โครงสร้างของฐานข้อมูลแบบสัมพันธ์เป็นการนำเสนอข้อมูลและความสัมพันธ์ระหว่างข้อมูลในรูปของรีเลชัน(relation) โดยรีเลชันจะถูกเรียกอีกอย่างหนึ่งว่า ตาราง(Table) โครงสร้างของรีเลชันประกอบด้วย

- แถว(Row) ของข้อมูล แถวข้อมูล 1 แถว หมายถึง ข้อมูล 1 รายการในแต่ละแถวของรีเลชันเรียกว่า “Tuple” ซึ่งก็คือ แถวข้อมูลในตารางโดยแต่ละแถวของข้อมูลจะประกอบไปด้วยหลายแอทริบิวต์หรือคอลัมน์ของข้อมูล จำนวนแถวข้อมูลในตารางเราเรียกว่า คาคินารี(Cardinality) และจำนวนแอทริบิวต์ทั้งหมดในตารางเรา เรียกว่า ดีกรี(Degree)
- สดมภ์ (Column) แต่ละสดมภ์ของรีเลชัน ได้แก่ คุณลักษณะของข้อมูลในแต่ละแถว ซึ่งเราเรียกว่า แอทริบิวต์(Attribute) เช่น รีเลชัน S สำหรับเก็บรายละเอียดของลูกค้า ประกอบด้วยรหัส ชื่อ สถานะ และเมือง ซึ่งรีเลชันดังกล่าวจะประกอบไปด้วย 5 แถวโดยแต่ละแถวประกอบไปด้วย 4 แอทริบิวต์ซึ่งภายในคอลัมน์จะประกอบด้วย
- โดเมน (Domain) เป็นการกำหนดขอบเขตค่าข้อมูลและชนิดข้อมูลของแต่ละแอทริบิวต์ที่เป็นไปได้ทั้งหมด เช่น จากรูปโดเมนของ S# คือกลุ่มของรหัสลูกค้าทั้งหมด ได้แก่ S1 S2 S3 S4 และ S5 กำหนดว่า STATUS ของตาราง S จะต้องมีค่าเป็น 10 20 และ 30 เท่านั้นหรือ S# จะต้องมีค่าเป็น S1 S2 S3 S4 และS5 เท่านั้น และCITY ของลูกค้าทั้งหมดจะต้องเป็น ปทุมธานี นนทบุรี และ กรุงเทพฯ



รูปที่ 2.19 ภาพแสดงตารางและรายละเอียด

- คีย์หลัก (Primary key) เป็นแอทริบิวต์หรือกลุ่มของแอทริบิวต์ที่บ่งบอกว่าข้อมูลจะต้องไม่ซ้ำกันในแต่ละแถวข้อมูลของตาราง

2.5.2 การควบคุมความถูกต้องให้กับข้อมูล (data integrity)

ในฐานะข้อมูล มีกฎการควบคุมความถูกต้องของข้อมูลคุณสมบัติหนึ่งที่สำคัญของความสัมพันธ์ก็คือ ความเป็นเอกลักษณ์ (Uniqueness property) สิ่งที่ใช้กำหนดความเป็นเอกลักษณ์ของแต่ละแถวในความสัมพันธ์ เรียกว่า คีย์ (key) ฐานข้อมูลหนึ่งจะมีข้อมูลอยู่จำนวนมากยิ่งฐานข้อมูลมีขนาดใหญ่ขึ้นก็จะมีข้อมูลจำนวนมากขึ้นเป็นเงาตามตัวข้อมูลเหล่านี้อาจมีค่าแตกต่างกัน คล้ายกัน หรือเหมือนกัน ทำให้การแยกแยะโดยอาศัยเพียงตัวข้อมูลอย่างเดียวทำได้ยากลำบาก ดังนั้นจึงมีการกำหนดคีย์ ประจำข้อมูลเพื่อทำให้การแยกแยะข้อมูลในฐานข้อมูลเป็นไปอย่างถูกต้อง

คีย์มีหลายประเภท ได้แก่

- คีย์หลัก (Primary key)

คีย์หลัก คือ คีย์หลักที่ใช้ในการอ้างอิง เอ็นติตี้(Entity) ในฐานข้อมูล การเลือกคีย์หลักสามารถเลือกได้จากข้อมูลที่ไม่มีโอกาสซ้ำซ้อนกันบนฐานข้อมูลนั้น

- คีย์รอง (Secondary Key)

คีย์สำรอง คือ คีย์เดี่ยวหรือคีย์ผสม (Single or Composite key) ซึ่งเมื่อใช้ในการค้นหาข้อมูลจากตารางความสัมพันธ์จะได้มากกว่าหนึ่งเรคคอร์ด ต่างจากคีย์หลักที่ทำให้ข้อมูลในตารางไม่ซ้ำกัน ดังนั้นคีย์รองจึงไม่จำเป็นจะต้องเป็นเอกลักษณ์

- คีย์นอก (Foreign key)

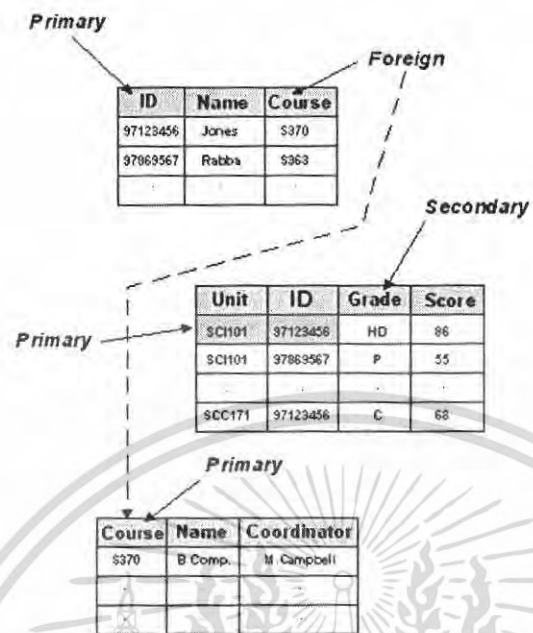
คีย์นอก คือ คีย์เดี่ยวหรือคีย์ผสม ซึ่งปรากฏเป็นคีย์ทั่วไปของความสัมพันธ์หนึ่งแต่ไปปรากฏเป็นอีกคีย์หลักในอีกความสัมพันธ์หนึ่ง คีย์นอกเป็นอีกคีย์หนึ่งที่มีความสำคัญมากในฐานข้อมูลเชิงสัมพันธ์ เนื่องจากเป็นตัวที่ใช้สร้างการเชื่อมต่อระหว่างความสัมพันธ์ จึงมีกฎและเงื่อนไขที่บังคับใช้เพื่อทำให้ข้อมูลมีความถูกต้องอยู่เสมอ

- ซุปเปอร์คีย์ (Superkey)

คือ กลุ่มของแอทริบิวต์ที่สามารถนำไปใช้ในการค้นหาข้อมูลที่เป็นเอกลักษณ์ได้

- คีย์แข่งขัน (Candidate key)

คีย์แข่งขัน ก็คือ ซุปเปอร์คีย์ และไม่มีกลุ่มย่อยของคีย์ใดในคีย์แข่งขันที่จะสามารถเป็นซุปเปอร์คีย์ได้อีก



รูปที่ 2.20 แสดงตารางและความสัมพันธ์ของคีย์แต่ละคีย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

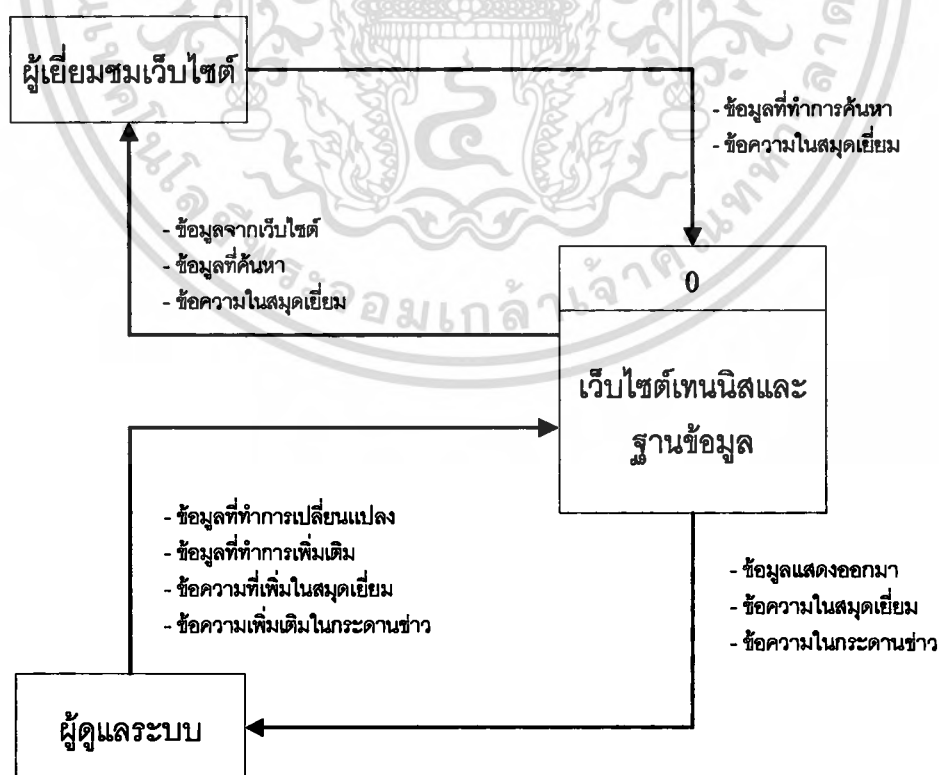
บทที่ 3

การออกแบบโครงการ

เนื่องจากโครงการที่เราได้ทำการกำหนด เป็นการนำเอาข้อมูลสถิตินักกีฬาเทนนิสในปัจจุบันและในอดีตมาเปรียบเทียบกันในรูปแบบกราฟ ในการเริ่มต้นออกแบบฐานข้อมูลจากขอบเขตที่ได้กำหนดไว้ เราจะจงในส่วนของนักกีฬาเทนนิสระดับตำนานและนักเทนนิสในปัจจุบันหนึ่งร้อยอันดับแรก อีกส่วนเป็นส่วนของการแข่งขัน โดยจะจำกัดอยู่ที่รายการแข่งขันรายการใหญ่ (Grand Slam) ทั้ง 4 รายการที่จัดขึ้นในทุกๆ ปี นำเสนอผ่านทางรูปแบบเว็บไซต์เพื่อให้สามารถนำเสนอได้อย่างกว้างขวาง โดยในส่วนที่มีการติดต่อกับฐานข้อมูลได้มีการออกแบบ ดังนี้

3.1 การออกแบบภาพรวม

จากขอบเขตในการออกแบบได้มีการกำหนดให้มีส่วนของผู้เยี่ยมชมและส่วนของผู้ดูแลระบบ ในส่วนที่มีการติดต่อกับผู้เยี่ยมชมสามารถให้ผู้เยี่ยมชมทำการแก้ไขข้อมูลได้ คือ เว็บไซต์หน้าสมุดเยี่ยม (guest book) และส่วนของผู้ดูแลระบบ คือ การแก้ไขข้อมูล และส่วนของกระดานข่าว สามารถแสดงกระบวนกรติดต่อเบื้องต้นได้ ดังรูป

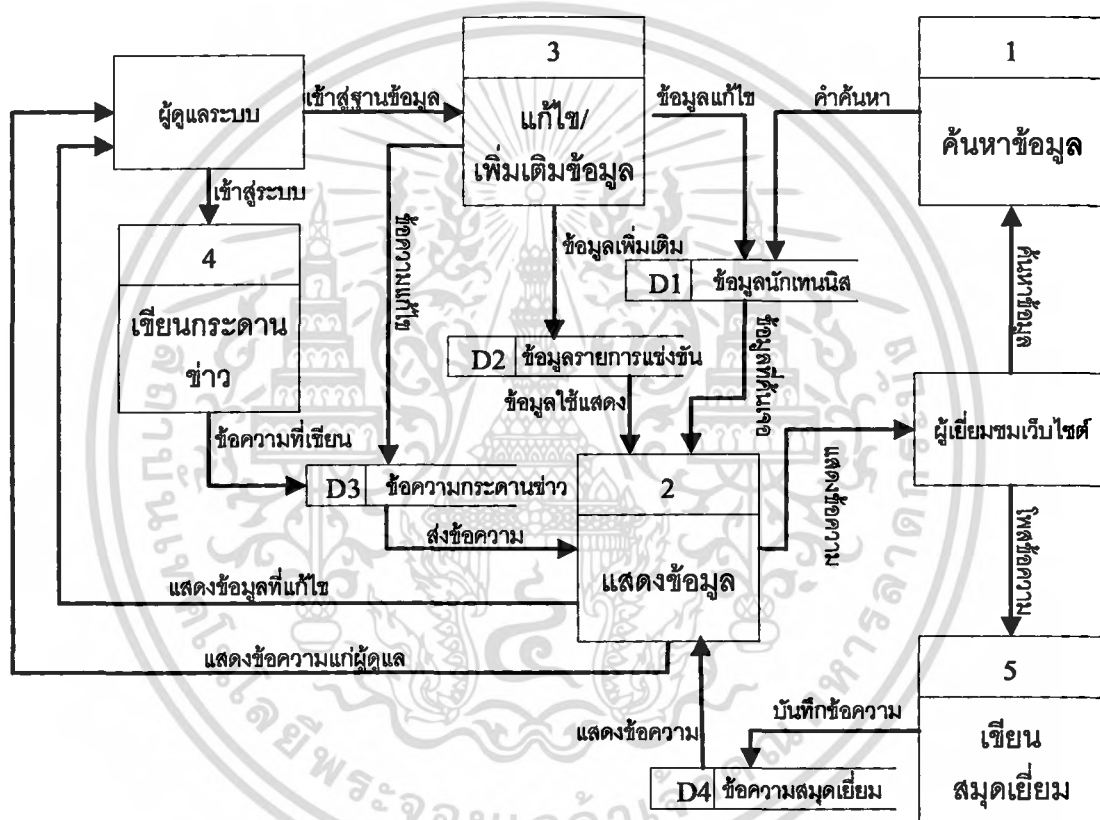


รูปที่ 3.1 ภาพแผนภาพระดับสูงสุด (context diagram)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบในระดับถัดไป

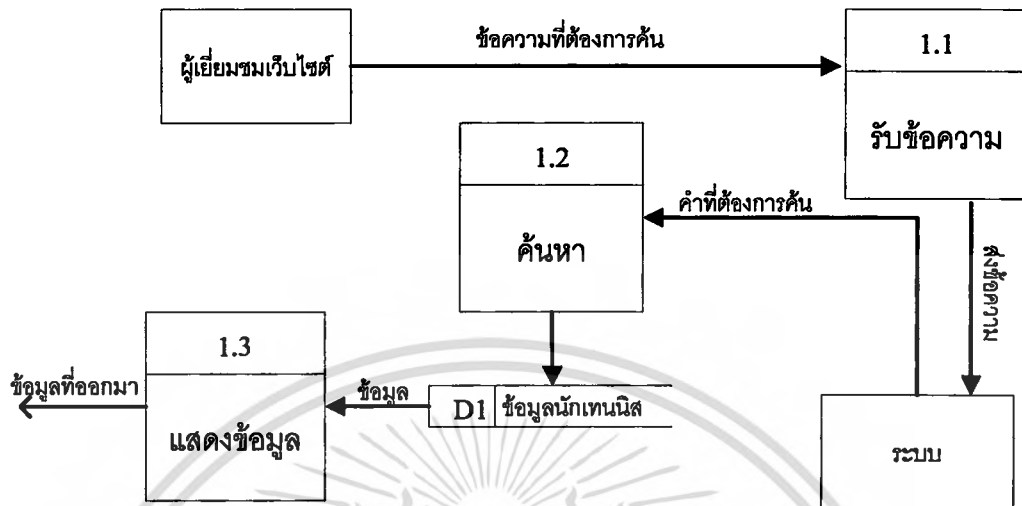
เมื่อทำการเจาะลึกถึงในระดับการติดต่อระหว่างผู้ใช้งานและฐานข้อมูล โดยผ่านกระบวนการที่ได้ทำการคิดไว้ ในการติดต่อจะมีการติดต่อในนามของผู้เยี่ยมชมและผู้ใช้ระบบ โดยทั้งคู่ถูกกำหนดสิทธิ์ให้มีการเข้าถึงฐานข้อมูลที่ต่างกัน โดยที่ผู้เยี่ยมชมถูกออกแบบให้สามารถเข้าถึงฐานข้อมูลได้ตามที่กำหนดไว้เท่านั้น เช่น สิทธิ์ในการค้นหาข้อมูลนักเทนนิส แต่ไม่สามารถทำการแก้ไขได้ แสดงได้ดังรูป



รูปที่ 3.2 ภาพแผนภาพระดับล่าง (diagram 0)

3.3 การออกแบบในระดับการประมวลผล

ในส่วนของการประมวลผลการค้นหาข้อมูล ทำได้โดยผู้เยี่ยมชมทำการพิมพ์รายชื่อนักเทนนิสที่ต้องการค้นหาหรือต้องการนำข้อมูลมาเปรียบเทียบ โดยที่เว็บไซต์จะทำการนำตัวอักษรที่ส่งต่อไปยังส่วนที่ทำการช่วยในการค้นหาข้อมูลและติดต่อไปยังฐานข้อมูล เพื่อนำข้อมูลหรือรายชื่อนักเทนนิสที่ใกล้เคียงที่สุดมาแสดงผลในรูปแบบที่เราต้องการนำไปใช้ ดังรูป



รูปที่ 3.3 ภาพกระแสข้อมูลระดับ 1 (level-1 diagram)

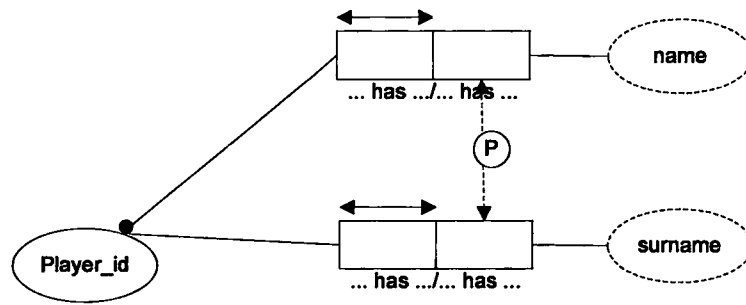
3.4 การออกแบบฐานข้อมูล

เมื่อทำการรวบรวมข้อมูลสามารถแบ่งข้อมูลเป็นส่วนหลัก ๆ ได้ 2 ส่วน คือ ส่วนของรายการแข่งขันและส่วนของนักกีฬาโดยส่วนของนักกีฬา เช่น ชื่อ นามสกุล สัญชาติ ลำดับที่ รายการที่ได้แชมป์ อายุ เล่นหรือไม่เล่นอยู่ เป็นต้น ในส่วนของรายการแข่งขันมีข้อมูล ชื่อ รายการแข่งขัน ประเทศที่จัด ชนิดของพื้นผิว ชื่อนักกีฬาที่ครองแชมป์ เป็นต้น

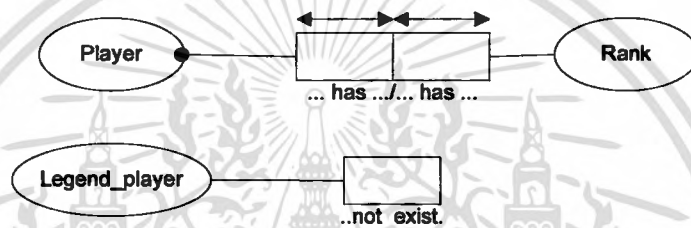
ตารางที่ 3.1 ข้อมูลนักเทนนิสที่ทำการรวบรวมได้

มีอันดับ	ชื่อ	นามสกุล	สัญชาติ	วันเกิด	อายุ	ส่วนสูง	น้ำหนัก	วันเกิด	สถานะ
----------	------	---------	---------	---------	------	---------	---------	---------	-------

จากตารางจะพบว่าข้อมูลที่ออกมาสามารถแบ่งย่อยได้อีก แบ่งเป็นนักเทนนิสที่เล่นอยู่และที่เลิกเล่นไปแล้ว โดยที่นักเทนนิสที่ยังคงเล่นอยู่จะมีการจัดลำดับ(Ranking) ส่วนที่เลิกเล่นไปแล้วจะมีนำมาจัดอันดับ เริ่มต้นโดยการกำหนดหมายเลขโอดีให้กับนักเทนนิสแต่ละคนซึ่งหมายเลขโอดีเหล่านี้สร้างขึ้นใหม่เพื่อใช้ในการอ้างอิงถึงนักเทนนิสคนนั้น ๆ

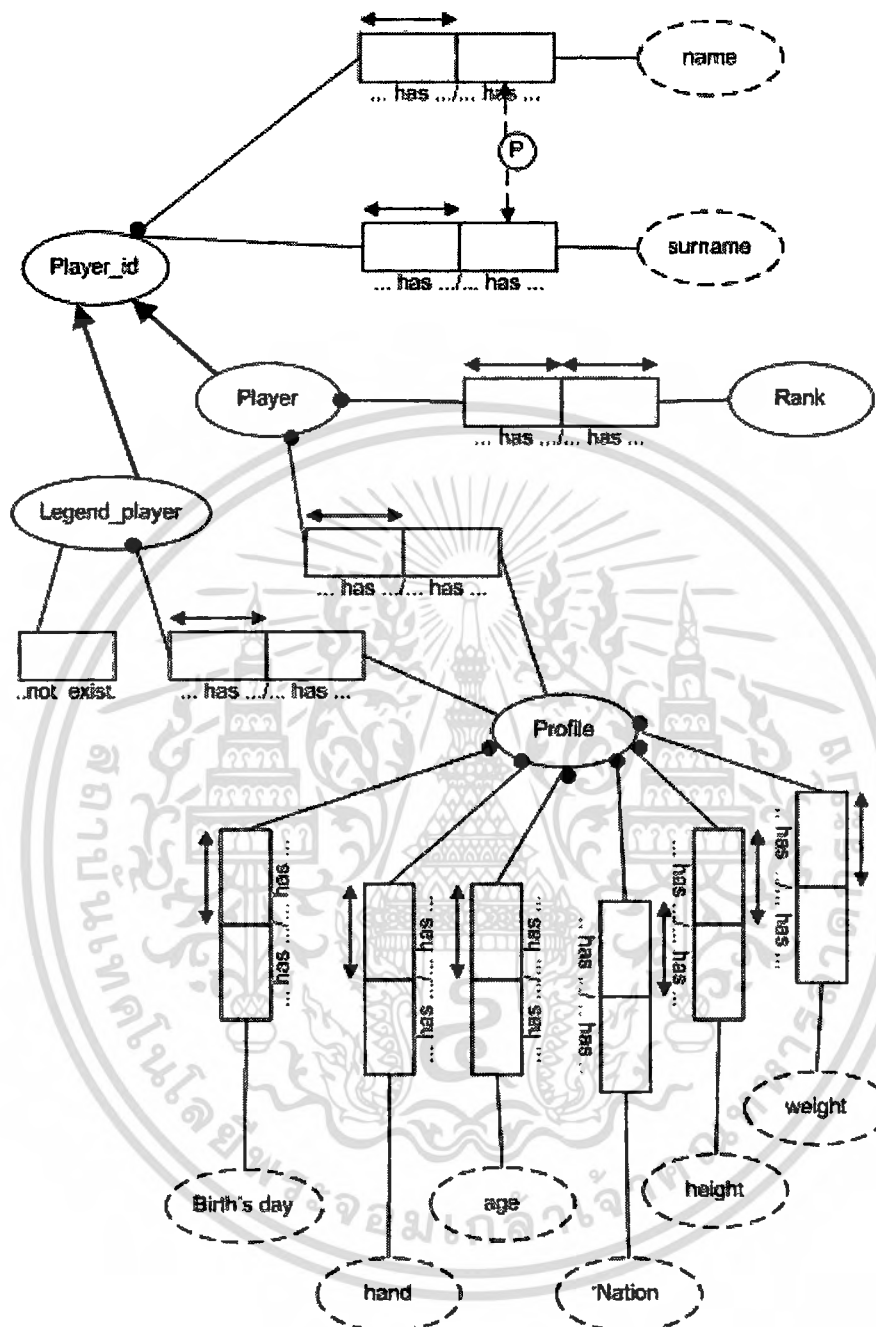


รูปที่ 3.4 ความสัมพันธ์ระหว่างหมายเลขกับชื่อนักเทนนิส



รูปที่ 3.5 ความสัมพันธ์ของผู้เล่นและอันดับ

ข้อมูลส่วนอื่นจะเป็นส่วนของรายละเอียด จึงทำการกำหนดส่วน profile(id) ขึ้นเพื่อทำการจัดเก็บข้อมูลในส่วนรายละเอียดเกี่ยวกับนักเทนนิส จากนั้นทำการเชื่อมต่อส่วนข้อมูลของนักเทนนิสทั้งหมดรวมกันได้เป็นฐานข้อมูลของนักเทนนิสทั้งหมด ซึ่งพบว่าฐานข้อมูลรายชื่อของนักเทนนิสทั้งในปัจจุบันและในอดีต ต่างเป็นส่วนหนึ่งของหมายเลขไอดีเช่นเดียวกัน จึงใช้เครื่องหมายแสดงว่าทั้งคู่ต่างเป็นส่วนหนึ่งในหมายเลขไอดี และทำการเชื่อมต่อข้อมูลทั้งสองกลุ่มเข้ากับหมายเลขไอดี โดยที่ทั้งคู่มีส่วนที่ต่างกันคือส่วนของการจัดอันดับ ซึ่งนักเทนนิสที่ได้ทำการเลิกเล่นไปแล้วจะ ไม่มีการจัดอันดับดังที่บอกไว้ข้างต้น



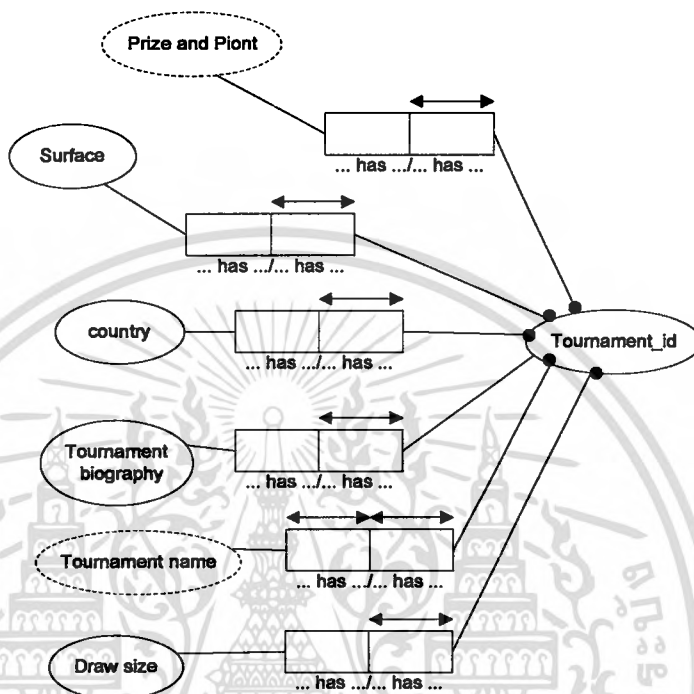
รูปที่ 3.6 ความสัมพันธ์ของข้อมูลนักเทนนิส

ตารางที่ 3.2 ข้อมูลรายการแข่งขัน

ชื่อรายการ	ชนิดคอร์ต	ประเทศ	ผู้ชนะ	คะแนน สะสม	จำนวนผู้เข้า แข่งขัน	ประวัติ	เงินรางวัล	รอบ

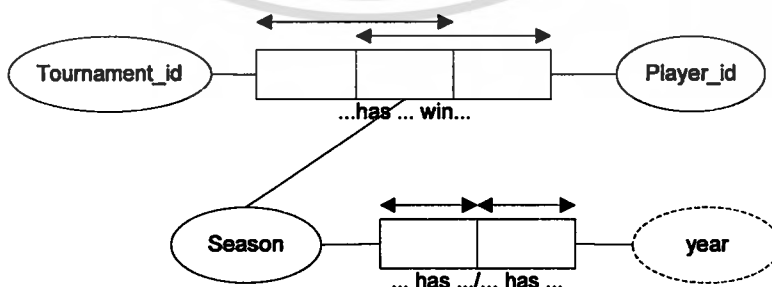
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดหมายเลขไอดีของรายการแข่งขันขึ้นให้ทำหน้าที่เป็นตัวบ่งชี้รายการแข่งขัน โดยทำการเชื่อมโยงข้อมูลเกี่ยวกับชื่อกับรายการแข่งขันทั้งหมด

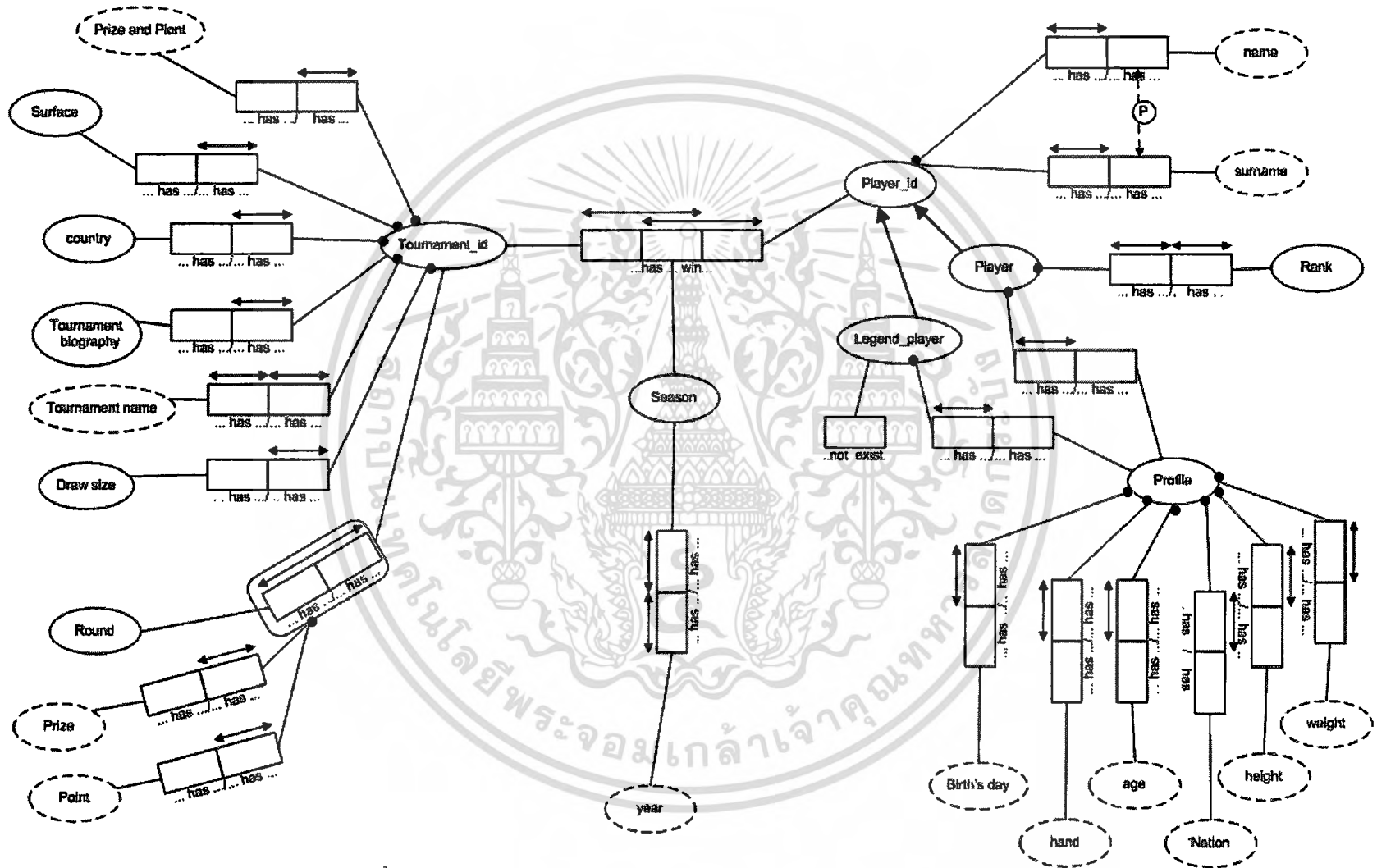


รูปที่ 3.7 ความสัมพันธ์ของรายการแข่งขันและข้อมูล

โดยที่รอบที่ผ่านเข้าไปได้ของนักเทนนิสในรายการแต่ละรายการเป็นตัวกำหนดจำนวนเงินรางวัล รางวัล และคะแนนสะสมที่จะได้รับไป และรายการแข่งขันในแต่ละฤดูกาลจะต้องมีผู้ที่ชนะรายการนั้น ๆ เพียงคนเดียว จึงได้ทำการเชื่อมโยงข้อมูลชื่อรายการแข่งขันกับฤดูกาล เพื่อชี้รายชื่อ นักเทนนิสที่เป็นผู้ชนะเลิศในการแข่งรายการนั้น หรือผู้เล่นในแต่ละปีชนะเลิศไป มีรายการใดบ้าง



รูปที่ 3.8 ความสัมพันธ์ของปี รายการแข่งขันและนักกีฬา



รูปที่ 3.9 NIAM แสดงความสัมพันธ์ภาพรวม

3.5 การนำฐานข้อมูลที่ออกแบบมาใช้งาน

ตารางที่ 3.3 PlayerName

Field	Type	Length	Comment
Player_id	VARCHAR	4	หมายเลขนักเทนนิส
P_name	VARCHAR	30	ชื่อของนักเทนนิส
P_surname	VARCHAR	30	นามสกุลของนักเทนนิส

ตารางที่ 3.4 Player

Field	Type	Length	Comment
Player_id	VARCHAR	4	หมายเลขนักเทนนิส
Rank	VARCHAR	3	ลำดับที่ของนักเทนนิส
Profile_id	VARCHAR	4	ประวัติของนักเทนนิส

ตารางที่ 3.5 LegendPlayer

Field	Type	Length	Comment
Player_id	VARCHAR	4	หมายเลขนักเทนนิส
Profile_id	VARCHAR	4	ประวัติของนักเทนนิส

ตารางที่ 3.6 PlayerProfile

Field	Type	Length	Comment
Profile_id	VARCHAR	4	ประวัติของนักเทนนิส
Nation	VARCHAR	20	สัญชาติของนักเทนนิส
Birthday	DATE	10	วันเกิด
Age	INT	3	อายุ
Hand	VARCHAR	5	มือที่ถนัด
Height	INT	3	ส่วนสูงของนักเทนนิส
Weight	INT	3	น้ำหนักของนักเทนนิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 TournamentName

Field	Type	Length	Comment
Tour_id	VARCHAR	4	หมายเลขรายการแข่งขัน
T_name	VARCHAR	30	ชื่อรายการแข่งขัน

ตารางที่ 3.8 TourProfile

Field	Type	Length	Comment
Tour_id	VARCHAR	4	หมายเลขรายการแข่งขัน
Country	VARCHAR	4	ประเทศที่ทำการจัด
Surface	VARCHAR	10	ชนิดพื้นผิว
DrawSize	INT	3	จำนวนผู้เข้าแข่งขันรอบหลัก
Tour_Bio	TEXT	2000	ข้อมูลเกี่ยวกับรายการ
Tour_prize	TEXT	500	ข้อมูลเงินรางวัลและคะแนนสะสม

ตารางที่ 3.9 PrizePoint

Field	Type	Length	Comment
Tour_id	VARCHAR	4	หมายเลขรายการแข่งขัน
Round	VARCHAR	12	รอบที่ผู้เล่นเข้าถึง
Prize	INT	10	เงินรางวัลที่ได้รับ
Point	INT	3	คะแนนสะสมที่ได้

ตารางที่ 3.10 Champion

Field	Type	Length	Comment
Tour_id	VARCHAR	4	หมายเลขรายการแข่งขัน
Player_id	VARCHAR	4	หมายเลขของผู้เล่น
Season	INT	4	ปีที่แข่งขัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

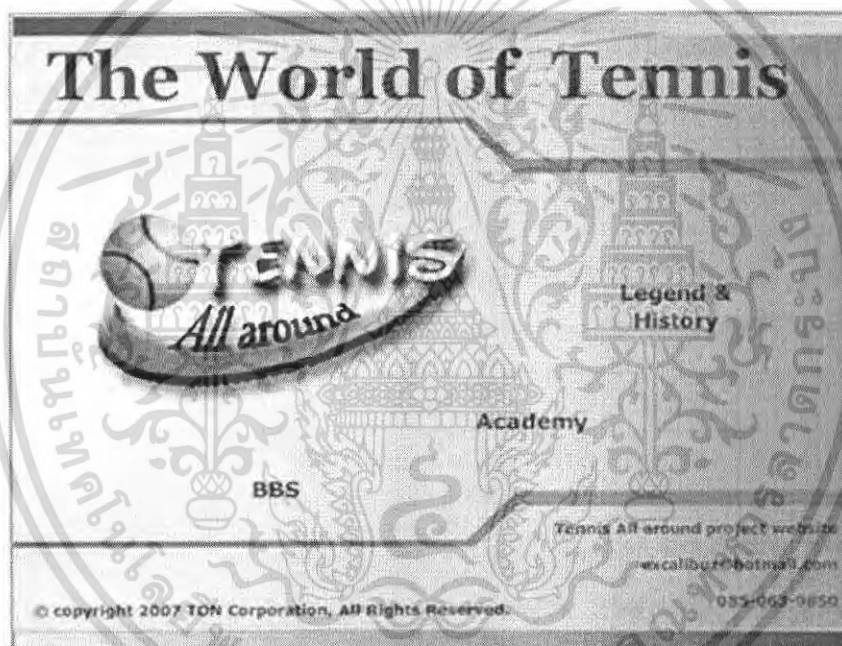
บทที่ 4

ผลงาน

ในส่วนการแสดงผลงานโครงการนี้ได้ทำการแยกข้อมูลและส่วนหลัก ๆ ออกเป็นสามส่วน ได้แก่ BBS Academy Legend & history ซึ่งแต่ละส่วนจะมีเนื้อหาและใช้ประโยชน์ต่างกันไป

4.1 ส่วนของเมนูหลัก

จะมีหัวข้อให้เลือกสามหัวข้อแยกตามลำดับ ได้ ดังนี้



รูปที่ 4.1 หน้าเมนูหลักของเว็บไซต์

<i>Legend & History</i>	เป็นส่วนที่แสดงถึงประวัติของนักเทนนิสและประวัติสนามแข่งขันแกรนด์สแลม
<i>Academy</i>	เป็นส่วนที่แสดงถึงวิธีการเล่น กฎกติกา ข่าวสารต่าง ๆ
<i>BBS</i>	เป็นส่วนแสดงเว็บบอร์ด

4.2 หัวข้อ Legend & History

หัวข้อนี้จะทำการแสดงประวัตินักเทนนิสและรายการแข่งขันแกรนด์สแลมทั้งสี่รายการตลอดจนสามารถค้นหาข้อมูลนักเทนนิสทั้งในตำนานและนักเทนนิสหนึ่งร้อย

อันดับแรกได้ โดยเมื่อทำการคลิกในส่วนนี้แล้วจะเปิดหน้าต่างขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 ส่วนบนของเว็บเพจ



รูปที่ 4.2 ภาพแสดงส่วนบนของเว็บเพจ

ด้านบนของเว็บเพจจะมีเมนูหลักให้เลือกเหมือนกับ *หน้าเมนูหลัก* แต่ว่าจะมีการเพิ่มในส่วนหัวข้อ *Guestbook* เพื่อให้ผู้เข้าชมสามารถให้คำแนะนำติชม ลงชื่อในสมุดเยี่ยมนี้ได้ซึ่งหน้านี้จะมีรายละเอียดดังต่อไปนี้

4.2.2 หัวข้อ Guestbook

เมื่อคลิกที่หัวข้อนี้แล้วจะแสดงหน้าสมุดเยี่ยมดังนี้

รูปที่ 4.3 ภาพรายละเอียดของ Guestbook

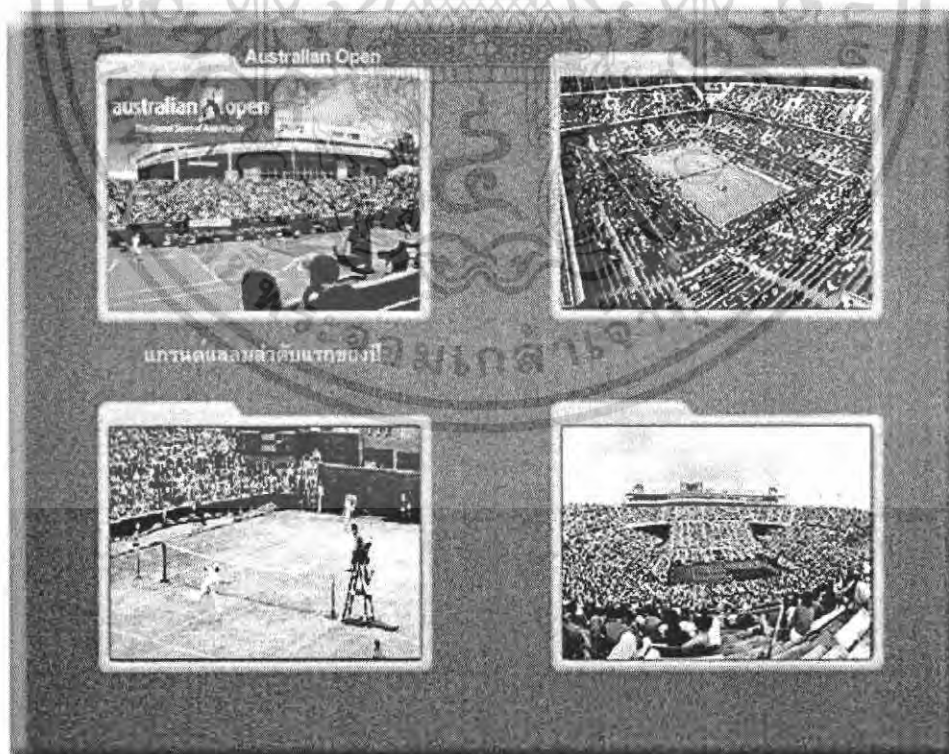
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหน้า Guestbook นี้ใช้ภาษา PHP ในการเขียน โดยจะมีการจัดเก็บข้อความที่เขียนเมื่อกดปุ่ม Send ในรูปแบบ Text ไฟล์แล้วดึงออกมาแสดงที่หน้าเว็บเพจทันที ซึ่ง Guestbook นี้มีคุณสมบัติดังต่อไปนี้

- ใส่ชื่อ หัวข้อ อีเมล URL และข้อความ ซึ่งถ้ามีการใส่ URL หรือ อีเมล จะปรากฏรูปตามที่ใส่ดังเช่นในรูป
- สามารถใส่พาสเวิร์ดในข้อความที่เราเขียนได้ตรงช่อง pass โดยระบบจะจดจำพาสเวิร์ดไว้ ซึ่งจะใช้เมื่อเราต้องการมาลบข้อความโดยการลบข้อความทำได้โดยใส่หมายเลขข้อความในช่อง No และใส่พาสเวิร์ดในช่อง Pass แล้วกดปุ่ม del ซึ่งเมื่อตรวจสอบแล้วว่าพาสเวิร์ดถูกต้องระบบจะลบข้อความนั้นให้ในทันที
- ข้อความที่เขียนลงไปจะมีการกำหนด วัน วันที่ เดือน ปีและเวลาที่ทำการบันทึกข้อความ
- มีในส่วนของ admin ซึ่งจะใช้สำหรับผู้ดูแลระบบในการจัดการกับข้อความต่าง ๆ ได้

4.2.3 ส่วนเมนูรายการแข่งขัน

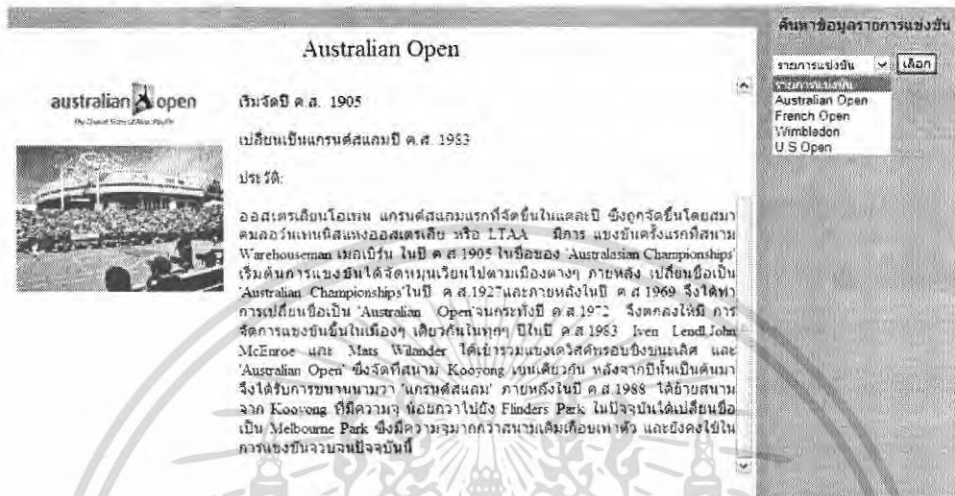
ในส่วนนี้จะทำเป็นเมนูให้เลือกเป็นสนามแข่งขันรายการแกรนด์สแลมทั้งหมด 4 สนาม ซึ่งเมื่อทำการคลิกแล้วก็จะเปิดหน้าต่างใหม่ออกมาเพื่อแสดงประวัติของสนามนั้น ๆ ดังรูป



รูปที่ 4.4 ภาพเมนูที่มีรายการแข่งขันทั้ง 4 รายการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการเลือกรายการเว็บเพจจะแสดงผลออกมาเป็นรายการนั้น ๆ ตามที่ได้เลือกโดยจะแสดงรายละเอียดเป็นประวัติความเป็นมาของรายการ สนามที่ใช้และสัญลักษณ์ของรายการ โดยทางขวาสามารถทำการเลือกแสดงรายการอื่นได้

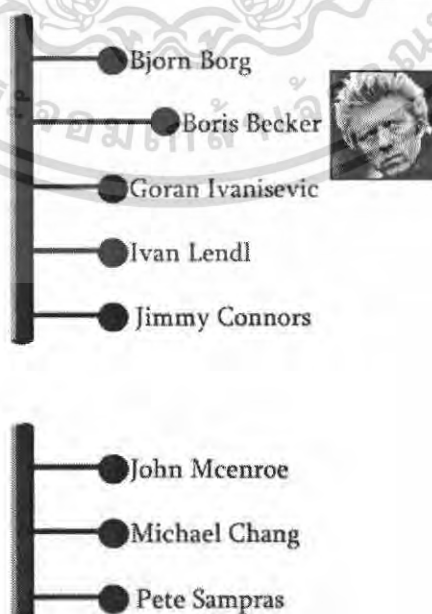


รูปที่ 4.5 หน้าแสดงข้อมูลรายการแกรนด์สแลม

4.2.4 นักเทนนิสในตำนาน

ด้านล่างของเว็บเพจจะเป็นส่วนที่แสดงเมนูให้เลือก โดยแสดงรายชื่อนักเทนนิสที่เป็นตำนานและนักเทนนิสที่มีชื่อเสียงโด่งดัง ตั้งแต่ในอดีตจนถึงปัจจุบัน

นักเทนนิสระดับตำนาน

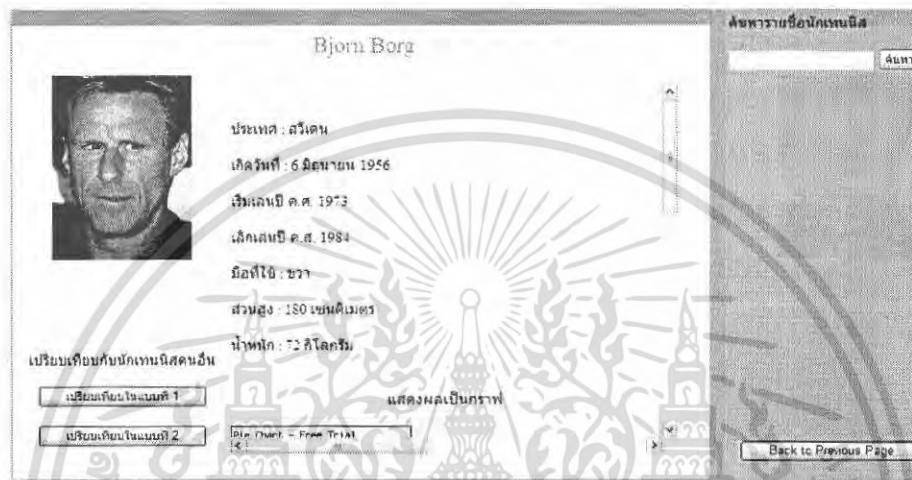


รูปที่ 4.6 เมนูเลือกดูข้อมูลนักเทนนิสในตำนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการคลิกไปยังรายชื่อนักเทนนิสจะแสดงข้อมูลนักเทนนิสและสถิติการเอาชนะในรายการแกรนด์สแลมทั้งสี่รายการ หรือสามารถทำการค้นหานักเทนนิสอื่น ๆ ในฐานข้อมูลได้ดังรูป

4.2.4.1 หน้าแสดงข้อมูลและสถิติของนักเทนนิส



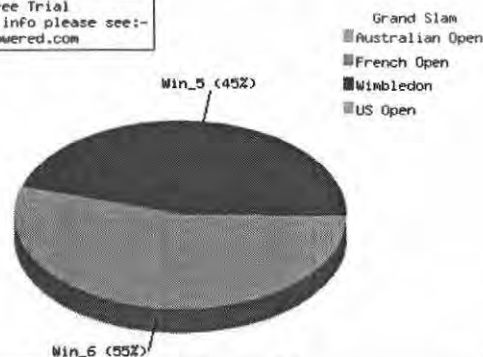
รูปที่ 4.7 หน้าแสดงข้อมูลและสถิติ

ภาพแสดงจำนวนการคว้าชัยในการแข่งขันแกรนด์สแลมแต่ละรายการและคิดเป็นเปอร์เซ็นต์เทียบกับทั้งหมด

4.2.4.2 กราฟวงกลม

ภาพกราฟวงกลมแสดงจำนวนการคว้าแชมป์ในรายการแกรนด์สแลมและเทียบเป็นอัตราส่วนซึ่งสามารถทำให้ทราบได้ว่านักเทนนิสผู้นั้นถนัดในการเล่นสนามหรือพื้นผิวแบบไหน

Pie Chart - Free Trial
For licensing info please see:-
<http://www.jpowered.com>

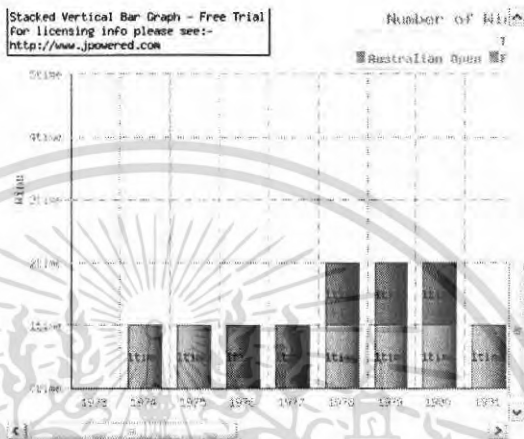


รูปที่ 4.8 แสดงอัตราส่วนการคว้าแชมป์ในรายการแกรนด์สแลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4.3 แผนภูมิแท่ง

แผนภูมิแสดงสถิติการคว้าแชมป์ของนักเทนนิส โดยแต่ละแท่งจะแสดงรายการที่สามารถทำการคว้าชัยชนะได้ในปีนั้น

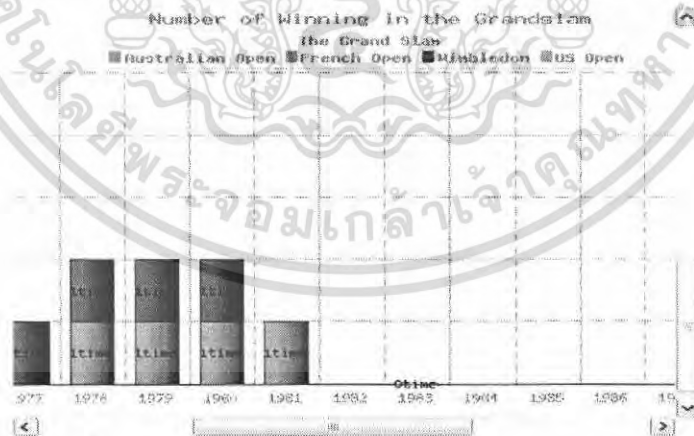


เปรียบเทียบกับนักเทนนิสคนอื่น

- เปรียบเทียบในแบบที่ 1
- เปรียบเทียบในแบบที่ 2

รูปที่ 4.9 แสดงอัตราส่วนการคว้าแชมป์ในรายการแกรนด์สแลม

โดยที่แต่ละสีในแท่งกราฟจะแสดงถึงประเภทของรายการแกรนด์สแลมที่นักเทนนิสนั้นคว้าชัยได้



รูปที่ 4.10 แสดงถึงประเภทสี่ของรายการแกรนด์สแลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าทำการคลิกปุ่ม “เปรียบเทียบในแบบที่ 1” หรือ “เปรียบเทียบในแบบที่ 2” เว็บจะทำการลิงค์ต่อไป โดยนำชื่อและข้อมูลนักเทนนิสคนนั้นไปหน้าทำการเปรียบเทียบ

	<table border="1"> <tr><td>Bjorn Borg</td></tr> <tr><td>สวีเดน</td></tr> <tr><td>52</td></tr> <tr><td>ขวา</td></tr> <tr><td>1973</td></tr> <tr><td>1984</td></tr> </table>	Bjorn Borg	สวีเดน	52	ขวา	1973	1984	<table border="1"> <tr><td>ชื่อ</td></tr> <tr><td>สัญชาติ</td></tr> <tr><td>อายุ</td></tr> <tr><td>มือที่ถนัด</td></tr> <tr><td>อันดับโลก</td></tr> <tr><td>ปีที่เริ่มเล่น</td></tr> <tr><td>เลิกเล่นปี</td></tr> </table>	ชื่อ	สัญชาติ	อายุ	มือที่ถนัด	อันดับโลก	ปีที่เริ่มเล่น	เลิกเล่นปี		
Bjorn Borg																	
สวีเดน																	
52																	
ขวา																	
1973																	
1984																	
ชื่อ																	
สัญชาติ																	
อายุ																	
มือที่ถนัด																	
อันดับโลก																	
ปีที่เริ่มเล่น																	
เลิกเล่นปี																	
Bjorn Borg กับ Goran Ivanisevic <input type="button" value="ทำการเปรียบเทียบ"/>																	

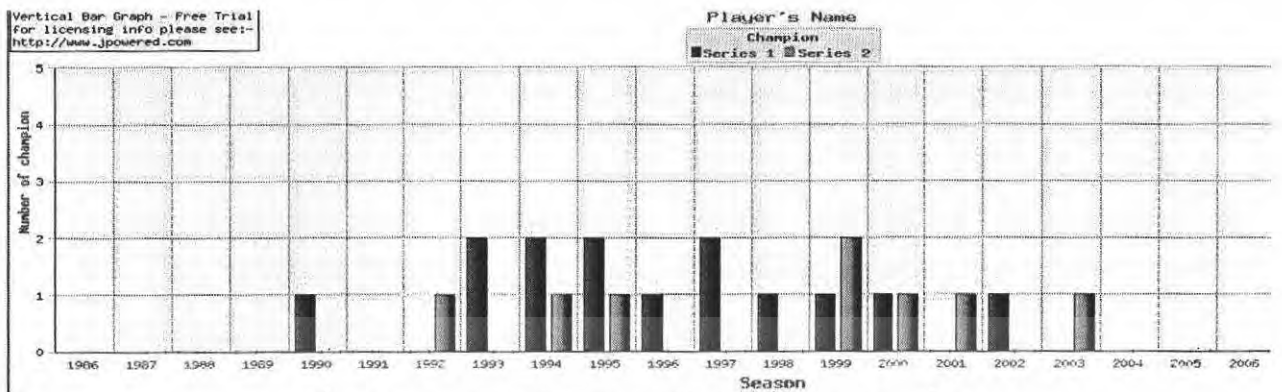
รูปที่ 4.11 แสดงหน้าต่างการแสดงผลเมื่อทำการคลิกปุ่มการเปรียบเทียบ

ในส่วนนักเทนนิสที่จะนำมาเปรียบเทียบสามารถทำการเลือกได้ใน Select box และคลิกปุ่ม “ทำการเปรียบเทียบ” เพื่อทำการแสดงผลดังรูป

	<table border="1"> <tr><td>Bjorn Borg</td></tr> <tr><td>สวีเดน</td></tr> <tr><td>52</td></tr> <tr><td>ขวา</td></tr> <tr><td>1973</td></tr> <tr><td>1984</td></tr> </table>	Bjorn Borg	สวีเดน	52	ขวา	1973	1984	<table border="1"> <tr><td>ชื่อ</td></tr> <tr><td>สัญชาติ</td></tr> <tr><td>อายุ</td></tr> <tr><td>มือที่ถนัด</td></tr> <tr><td>อันดับโลก</td></tr> <tr><td>ปีที่เริ่มเล่น</td></tr> <tr><td>เลิกเล่นปี</td></tr> </table>	ชื่อ	สัญชาติ	อายุ	มือที่ถนัด	อันดับโลก	ปีที่เริ่มเล่น	เลิกเล่นปี	<table border="1"> <tr><td>Pete Sampras</td></tr> <tr><td>สหรัฐอเมริกา</td></tr> <tr><td>37</td></tr> <tr><td>ขวา</td></tr> <tr><td>1988</td></tr> <tr><td>-</td></tr> </table>	Pete Sampras	สหรัฐอเมริกา	37	ขวา	1988	-	
Bjorn Borg																							
สวีเดน																							
52																							
ขวา																							
1973																							
1984																							
ชื่อ																							
สัญชาติ																							
อายุ																							
มือที่ถนัด																							
อันดับโลก																							
ปีที่เริ่มเล่น																							
เลิกเล่นปี																							
Pete Sampras																							
สหรัฐอเมริกา																							
37																							
ขวา																							
1988																							
-																							
Bjorn Borg กับ รายชื่อนักเทนนิส <input type="button" value="ทำการเปรียบเทียบ"/>																							

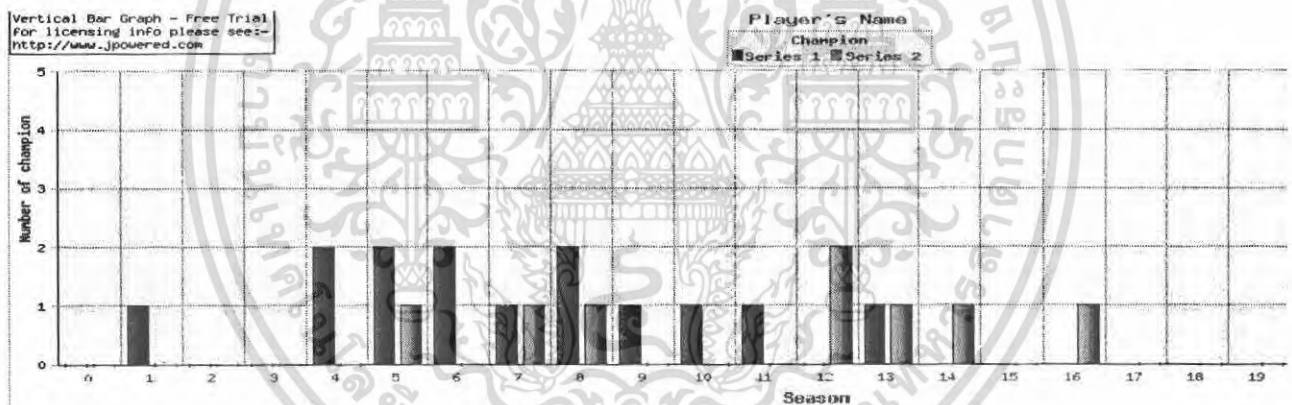
รูปที่ 4.12 แสดงการเลือกนักเทนนิสเพื่อนำมาเปรียบเทียบ

เมื่อเลือกนักเทนนิสคนที่สองแล้วคลิกปุ่ม “ทำการเปรียบเทียบ” ระบบจะดึงข้อมูลกราฟของนักเทนนิสคนที่สองขึ้นมาเพื่อเปรียบเทียบกับนักเทนนิสแรกที่เลือกไว้ก่อนหน้านี้



รูปที่ 4.13 แสดงผลลัพธ์ของกราฟเมื่อเลือกนักเทนนิสคนที่สอง

และเมื่อทำการคลิกปุ่ม “เปรียบเทียบในแบบที่ 2” รูปแบบในการแสดงผลจะเหมือนกับแบบแรก แต่จะต่างตรงที่การแสดงผลกราฟผลลัพธ์จะแสดงเป็นกราฟเปรียบเทียบตามช่วงอายุการเล่นอาชีพของนักเทนนิสแทน



รูปที่ 4.14 แสดงผลลัพธ์ของกราฟของการเปรียบเทียบในแบบที่ 2

4.3 หัวข้อ Academy

เมื่อคลิกแล้วจะเปิดหน้าต่างขึ้นมาโดยหน้าต่างนี้จะแสดงหน้าหลักของข่าวสาร ข้อมูลการวอร์มอัพ เทคนิคการเล่น กฎกติกา และข้อมูลเร็คเกตที่น่าสนใจ

4.3.1 ส่วนบนของเว็บเพจ



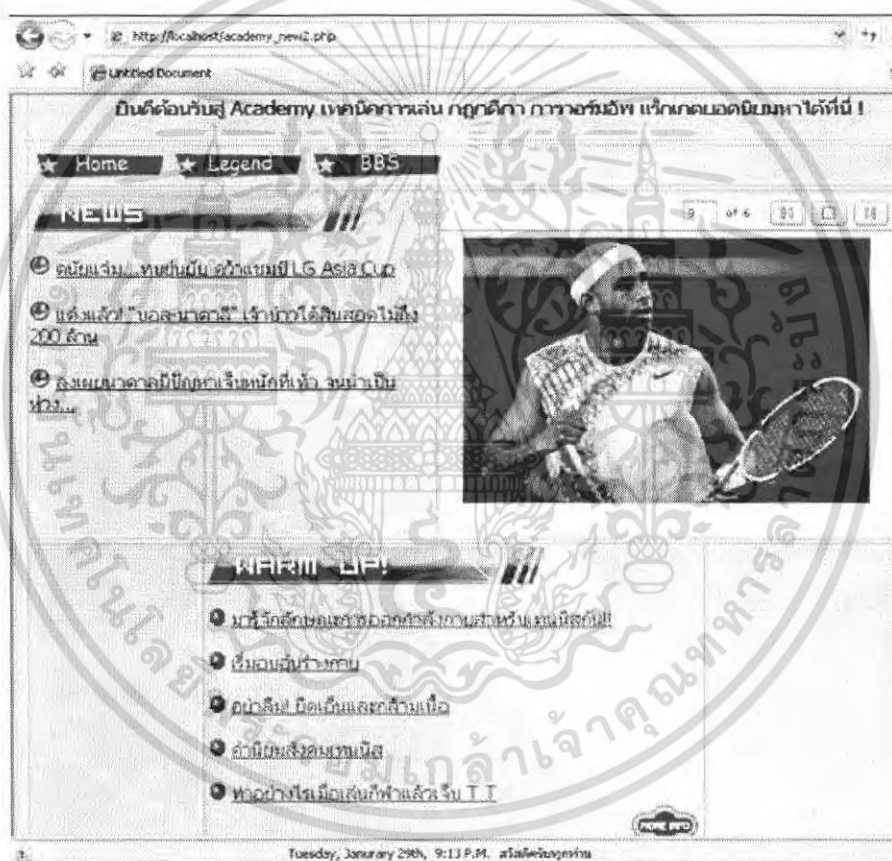
รูปที่ 4.15 แสดงรูปส่วนบนของเว็บเพจในหน้า Academy

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้านบนของหน้าเพจจะมีหัวข้อย่อให้เลือกอีกสี่หัวข้อซึ่ง ได้แก่

- WARM UP!
- TECHNIQUES
- RULES
- RACKET

โดยที่หน้าเว็บเพจมีรูปแบบดังรูป



รูปที่ 4.16 แสดงหน้าเว็บเพจหลักของ Academy

4.3.2 หัวข้อ WARM UP!

หัวข้อนี้จะแสดงวิธีที่ถูกต้องในการอบอุ่นร่างกายก่อนทำการแข่งขันหรือซ้อมเทนนิส เพื่อหลีกเลี่ยงอาการบาดเจ็บที่จะเกิดขึ้นจากการเล่นกีฬาชนิดนี้ เมื่อคลิกที่หัวข้อนี้จะแสดงข้อมูลในการอบอุ่นร่างกายดังรูป

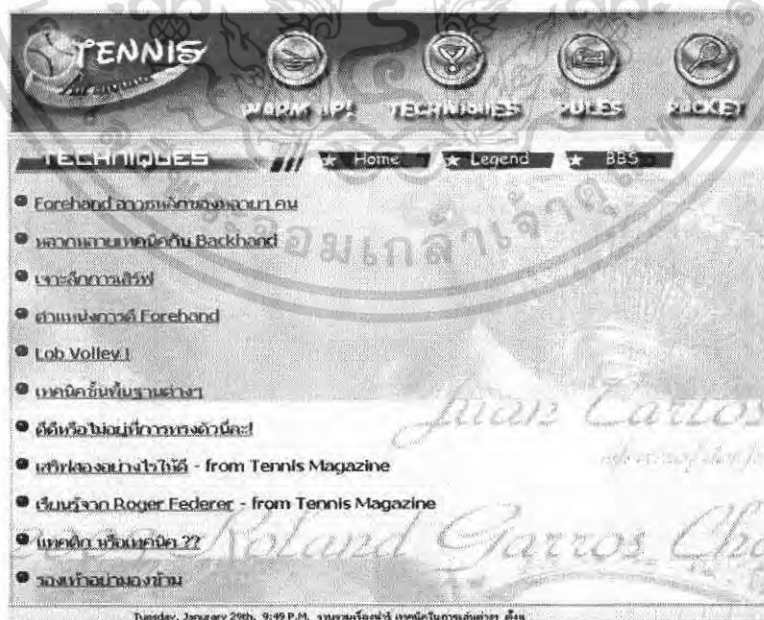
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 แสดงหน้าเว็บเพจของหัวข้อ WARM UP!

4.3.3 หัวข้อ TECHNIQUES

ส่วนนี้แสดงวิธีการเล่น ตลอดจนเทคนิคเบื้องต้นสำหรับมือใหม่ที่สนใจในกีฬาเทนนิส เมื่อคลิกที่หัวข้อนี้จะแสดงข้อมูลเทคนิคต่าง ๆ ดังรูป

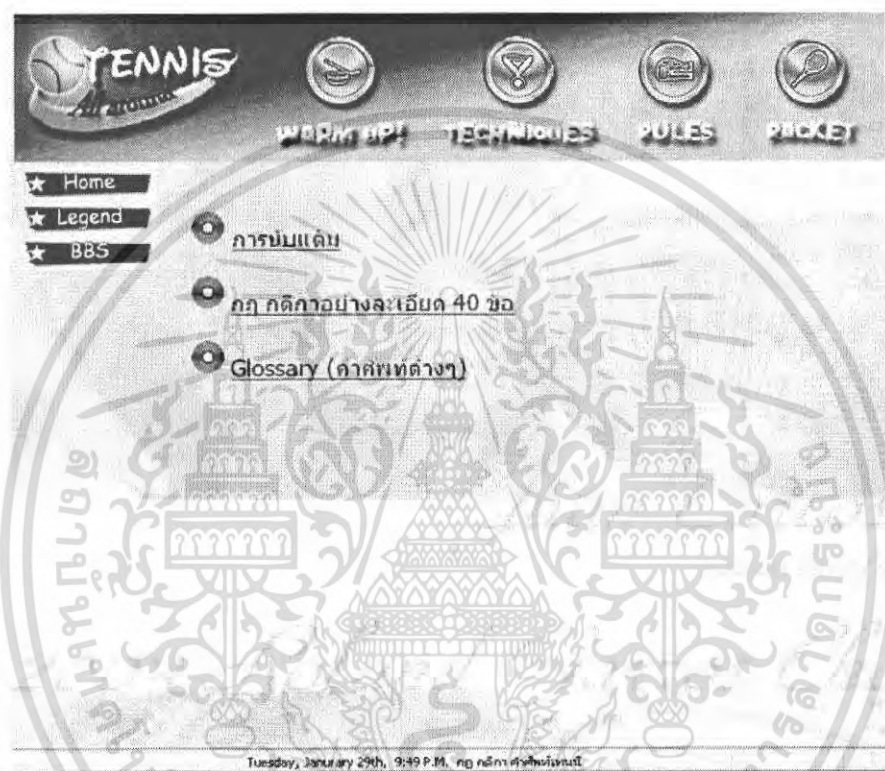


รูปที่ 4.18 แสดงหน้าเว็บเพจของหัวข้อ TECHNIQUES

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4 หัวข้อ RULES

ในส่วนนี้จะให้คำแนะนำเกี่ยวกับกติกาการนับแต้มและกติกาเบื้องต้นเกี่ยวกับกีฬาเทนนิสสำหรับผู้สนใจในกีฬานี้ เมื่อคลิกที่หัวข้อนี้จะแสดงกฎ กติกา การนับแต้ม และคำศัพท์ที่ควรรู้ต่าง ๆ ดังรูป



รูปที่ 4.19 แสดงหน้าเว็บเพจของหัวข้อ RULES

4.3.5 หัวข้อ RACKET

ในส่วนนี้จะให้คำแนะนำเกี่ยวกับแร็กเกตเทนนิสในแต่ละประเภท เพื่อให้ผู้สนใจสามารถนำไปเป็นข้อมูลประกอบการตัดสินใจเลือกซื้อไม้เทนนิส เมื่อคลิกที่หัวข้อนี้จะแสดงข้อมูลแร็กเกตยอดนิยม พร้อมทั้งการวิจารณ์แร็กเกตต่าง ๆ ดังรูป



รูปที่ 4.20 แสดงหน้าเว็บเพจของหัวข้อ RACKET

4.4 หัวข้อ BBS

เป็นส่วนที่ใช้เพื่อทำการติดต่อสอบถามหรือสนทนาของผู้เยี่ยมชม เมื่อทำการเข้าสู่หัวข้อนี้ หน้าเว็บเพจจะแสดงผลหน้ากระดาษข้อมูลข่าวสารดังรูป

ค้นหาข่าว: Search

[Home] [ติดต่อเรา] [สมัครสมาชิก] [แจ้งปัญหา]

Today: Thu 07 Feb 2008 12:50:16 +0700

จำนวนผู้เข้าชม: 130 ครั้ง จำนวน 2 คนอยู่ Online 1

หัวข้อ	หัวข้อ	วันที่	จำนวน	จำนวน	จำนวน
00002	เรื่อง: [unclear] 2008-02-01 15:31:00	8	1	2008-02-04 14:32:56	ข่าว
00001	เรื่อง: [unclear] 2008-01-22 18:08:26	27	0	2008-01-22 18:08:26	ข่าว

Board v.3.30 © 2002-2007 All rights reserved.

รูปที่ 4.21 แสดงหน้าเว็บเพจของหัวข้อ BBS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุป

5.1 การพัฒนาโครงการ

ผู้จัดทำได้ทำการศึกษาและรวบรวมข้อมูลที่เกี่ยวข้องกับกีฬาเทนนิส แล้วคิดรูปแบบการนำเสนอตลอดจนศึกษาวิธีการนำเสนอในหลาย ๆ รูปแบบ อาทิเช่นเขียนโค้ดภาษาต่าง ๆ โดยจัดทำแสดงออกมาในรูปแบบเว็บเพจ และทดสอบการทำงานของโครงการ

5.2 ปัญหาระหว่างการพัฒนา

เนื่องจากผู้พัฒนามีความชำนาญไม่เพียงพอในการใช้โปรแกรมต่าง ๆ อาทิเช่น Flash Professional 8, Dreamweaver 8 และมีประสบการณ์ในการสร้างเว็บไซต์น้อย จึงทำให้ใช้เวลานานในการศึกษาและการพัฒนาออกแบบในขั้นตอนแต่ละขั้นตอน ทั้งยังขาดประสบการณ์และความชำนาญในการใช้ภาษา PHP, MySQL ตลอดจนโปรแกรมสำเร็จรูปที่ต้องนำมาใช้ในโครงการ จึงเป็นเหตุให้ใช้เวลานานในการศึกษาเพื่อให้เกิดความเข้าใจและความชำนาญในส่วน ๆ นั้น

5.3 แนวทางการพัฒนาต่อ

สามารถพัฒนาเว็บไซต์เพิ่มเติมได้อีกหลายแนวทาง และยังสามารถพัฒนาวิธีการในการนำเสนอได้อีกมากมาย โดยอาจจะเพิ่มข้อมูลในส่วนของผู้เล่นให้มากขึ้น โดยเฉพาะข้อมูลนักเทนนิสหญิง หรือทำการเปรียบเทียบข้อมูลนักเทนนิส ตลอดจนสถิติต่าง ๆ ให้ละเอียดและสลับซับซ้อนได้มากยิ่งขึ้น แม้กระทั่งในส่วนเนื้อหาของเว็บไซต์สามารถเพิ่มในส่วนของวิดีโอหรือรูปภาพเกี่ยวกับเทนนิสที่น่าสนใจ เพื่อนำมาเป็นสื่อให้ผู้คนทั่วไปสนใจในกีฬาเทนนิส นอกจากนั้นแล้วยังสามารถนำมาเป็นตัวอย่างในการฝึกสอนวิธีการเล่นเบื้องต้นแก่ผู้เยี่ยมชมได้อีกด้วย

เอกสารอ้างอิง

David Morris. 2549. **สร้างเว็บสวยด้วย Flash 8**. พิมพ์ครั้งที่ 1. กรุงเทพฯ : ส.เอเชียเพลส.

สมศักดิ์ เถอยุกต์. 2546. **เทคนิคสร้างงาน Animation ด้วย Flash MX**. กรุงเทพฯ : ห้างหุ้นส่วนจำกัด ยงพลเทรคดิง.

ชวิศน์ช อิงชาติเจริญ. 2545. **เรียนจาวาสคริปต์เบื้องต้นด้วยตัวเอง**. กรุงเทพฯ : บริษัท ว.เพชรสกุล จำกัด.

สมศักดิ์ โชคชัยชุตติกุล. 2547. **อินไซต์ PHP 5**. กรุงเทพฯ : บริษัท โปรวิชั่น จำกัด.

จิตติมา มโนหมั่นศรีรักษา. 2548. **PHP & MySQL make it easy**. กรุงเทพฯ : โรงพิมพ์และทำปกเจริญผล.

