



การสร้างแบบทดสอบคำสั่งสืบค้นข้อมูล และตรวจคำตอบโดยอัตโนมัติ

Quiz Generating for SQL Command and Checking Answer Automatically

อุไรวรรณ รักผกาวงศ์* และ พงษ์เทพ รักผกาวงศ์

สาขาคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏพิบูลสงคราม จังหวัดพิษณุโลก 65000

*Corresponding Author, E-mail: urairux@psru.ac.th

บทคัดย่อ

คำสั่งสืบค้นข้อมูลเป็นคำสั่งพื้นฐานที่สำคัญในการพัฒนาโปรแกรมประยุกต์ ซึ่งในการเรียนการสอน จำเป็นต้องมีตัวอย่างโจทย์จำนวนมาก เพื่อให้ผู้เรียนใช้ฝึกเขียนคำสั่ง และฝึกหาผลของการใช้คำสั่งสืบค้นข้อมูลนั้น โดยผู้เรียนสามารถใช้โปรแกรมระบบจัดการฐานข้อมูลทั่วไป ในการฝึกทดลองคำสั่งได้ด้วยตนเอง แต่อาจพบปัญหาข้อผิดพลาดที่ผิด และไม่สามารถแก้ไขคำสั่งให้ถูกต้องได้ด้วยตนเอง เพราะระบบจัดการฐานข้อมูลแจ้งเพียงส่วนคำสั่งที่ผิดที่พบในจุดแรกสุดเท่านั้น จากปัญหาข้างต้น ในงานวิจัยนี้ จึงพัฒนาระบบ 2 ระบบ คือ 1) ระบบการสร้างแบบทดสอบคำสั่งสืบค้นข้อมูล จากโจทย์ต้นแบบ เพื่อใช้เป็นตัวอย่างโจทย์ให้ฝึกปฏิบัติ จากนั้นใช้ระบบที่ 2 คือ ระบบตรวจคำตอบ โดยจะตรวจคำสั่งแต่ละส่วน (SELECT, FROM, WHERE, GROUP BY, ORDER BY) และแสดงแต่ละจุดที่ผิด เพื่อให้ผู้เรียนได้ทราบคำสั่งที่ถูกต้อง จากการออกแบบ 8 โจทย์ต้นแบบ นำไปทดลองใช้กับผู้เรียนจำนวน 43 คน สามารถสร้างโจทย์คำสั่งให้ผู้เรียนได้ฝึกทำ คนละ 2 ครั้ง ครั้งละ 11 โจทย์ รวมสร้างโจทย์ทั้งหมด 946 โจทย์ ที่ไม่เหมือนกัน ใช้เวลาสร้างโจทย์เฉลี่ย โจทย์ละ 2.56 วินาที และใช้ระบบตรวจคำตอบโดยเฉลี่ยโจทย์ละ 12.85 วินาทีได้อย่างถูกต้อง

ABSTRACT

Structure Query Language (SQL) Command is a basic outstanding statement for developing application program. In a class room, variety of quiz examples is necessary for practice creating command and searching answer. Students can use the product of Database Management System (DBMS) as self-learn tutorial. However, they put incorrect command and cannot edit by themselves because DBMS shows only the first error. Thus, this research aims to develop 2 systems, 1) quiz generating system for SQL Command from quiz master as examples for practice, and then 2) answer checking system for searching each part (SELECT, FROM, WHERE, GROUP BY, ORDER BY) and show each error in order to let student know the correct command. The eight quiz masters were tried out with 43 students with 2 times per a student and 11 quizzes per time as well as create individual 946 quizzes. Average creating a quiz time was 2.56 seconds and checking answer time was 12.85 seconds.

คำสำคัญ: คำสั่งสืบค้นข้อมูล แบบทดสอบต้นแบบ ตรวจคำตอบ

Keywords: SQL command, Prototype quiz, Check answer

บทนำ

การเรียนการสอนในรายวิชาการระบบจัดการฐานข้อมูล จะศึกษาเกี่ยวกับ สถาปัตยกรรมฐานข้อมูล ระดับภายใน ระดับแนวคิด และระดับภายนอก พีชคณิตเชิงสัมพันธ์ การออกแบบฐานข้อมูล โดยใช้แผนภาพอี-อาร์ (E-R diagram) การทำให้อยู่ในบรรทัดฐาน (Normalization) และคำสั่งระบบจัดการฐานข้อมูล (อุไรวรรณ, 2560) ซึ่งการศึกษาคำสั่งระบบจัดการฐานข้อมูลนี้ มีความสำคัญเพราะจะเป็นพื้นฐานที่สำคัญในการพัฒนาโปรแกรมประยุกต์ใช้งานด้านต่างๆ เช่น ระบบบัญชี ระบบบริหารสินค้าคงคลัง ระบบบุคลากร ระบบลงทะเบียนเรียน เป็นต้น

การศึกษาคำสั่งระบบจัดการฐานข้อมูล ในเบื้องต้นเป็นการศึกษาคำสั่งที่เป็นมาตรฐาน (ISO Standards, 2003) เช่น คำสั่งสร้างตาราง (CREATE TABLE) คำสั่งเพิ่มข้อมูล (INSERT) คำสั่งปรับปรุงข้อมูล (UPDATE) คำสั่งสืบค้นข้อมูล (SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY) เป็นต้น ส่วนการทดลองใช้คำสั่งในภาคปฏิบัติของผู้เรียน ผู้สอนอาจใช้ผลิตภัณฑ์ระบบจัดการฐานข้อมูลตามที่จัดหาได้ มาเป็นแบบฝึกทดลองใช้คำสั่ง เช่น Oracle RDBMS, MySQL, IBM DB2, Informix, Microsoft SQL Server, Microsoft Access, SAP HANA หรืออื่นๆ ซึ่งระบบจัดการฐานข้อมูลจะรับคำสั่ง และแสดงผลของคำสั่งให้ทราบ แต่ถ้ามีส่วนใดของคำสั่งที่ผิด ระบบจะแจ้งจุดที่พบคำสั่งผิด หยุดการทำงาน เพื่อให้ผู้ใช้แก้ไขคำสั่งให้ถูกต้องก่อน แล้วจึงจะทำงานตามคำสั่งที่ต้องการ กล่าวโดยสรุปคือ ระบบจัดการฐานข้อมูลจะรับคำสั่งที่ต้องการเท่านั้น ทำให้ผู้เรียนต้องฝึกใช้คำสั่ง ในแต่ละส่วนให้ถูกต้อง ถ้ามีบางส่วนผิดและไม่สามารถแก้ไขให้ถูกต้องได้ ก็จะไม่สามารถทราบได้ว่าคำสั่งส่วนอื่นๆที่เหลือ ถูก หรือผิด หรือไม่อย่างไร

การฝึกใช้คำสั่ง ถ้ามีโจทย์ตัวอย่างให้ฝึกทดลองใช้คำสั่ง ในหลากหลายสถานการณ์ จะเป็นโอกาสให้ผู้เรียนได้ฝึกใช้คำสั่งเพื่อหาคำตอบในสถานการณ์ต่างๆได้อย่างดี จึงเป็นความร่วมมือระหว่างผู้เรียน ที่จะต้องไขว่คว้าหาโจทย์มาฝึกทำ และความร่วมมือของผู้สอน ที่จะต้องนำโจทย์ที่ง่ายในระดับเริ่มต้น เพื่อให้ผู้เรียนเกิดความเข้าใจ มีความคุ้นเคยกับโจทย์คำสั่งในการหาคำตอบ รู้สึกไม่ยากจนเกินไป จากนั้นพัฒนาขึ้น โดยนำโจทย์ที่ซับซ้อนมากขึ้น เหมาะกับชั้นกลาง และชั้นสูงต่อไปตามลำดับความเหมาะสม ซึ่งภาระที่ตามมาของผู้สอน คือการตรวจคำตอบของผู้เรียนว่ามีการใช้คำสั่ง แต่ละส่วนถูกต้อง หรือไม่อย่างไร ซึ่ง

การตรวจคำตอบนี้ จะช่วยให้ผู้เรียนสามารถรู้ข้อบกพร่องของคำสั่ง และปรับปรุงให้ถูกต้องได้ อันจะทำให้สามารถเรียนรู้การใช้คำสั่งที่ซับซ้อนในขั้นสูงขึ้นไปเป็นลำดับ

จากปัญหาการเรียนการสอนที่ผู้สอนจะต้องเตรียมโจทย์คำสั่งจำนวนมาก และตรวจคำตอบจำนวนมากเป็นทวีคูณตามจำนวนผู้เรียน ผู้วิจัยจึงมีแนวคิดการสร้างแบบทดสอบคำสั่งสืบค้นข้อมูล และตรวจคำตอบโดยอัตโนมัติ เพื่อช่วยในการเรียนการสอน ทำให้ผู้เรียนมีโจทย์ให้ฝึกทำ เพื่อพัฒนาทักษะการใช้คำสั่งสืบค้นข้อมูลได้อย่างถูกต้อง สามารถนำไปใช้ประกอบการพัฒนาโปรแกรมประยุกต์ด้านอื่นๆต่อไป

ในบทความนี้แบ่งออกเป็น 6 ส่วน ส่วนที่ 1 บทนำอธิบายเกี่ยวกับปัญหาและที่มาของงานวิจัย ซึ่งได้นำเสนอข้างต้น ส่วนที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง จะได้กล่าวถึง คำสั่งสืบค้นข้อมูล que สร้างเป็นแบบทดสอบ นิพจน์ปรกติ (Regular expression) ที่ถูกนำมาใช้ในการตรวจคำตอบ และงานวิจัยอื่นๆ ที่นำเสนอมาในอดีต ส่วนที่ 3 วิธีการดำเนินการวิจัย จะได้อธิบายถึงขั้นตอนการดำเนินการวิจัย ที่แบ่งเป็น การสร้างแบบทดสอบ การทำแบบทดสอบ และการตรวจคำตอบ ส่วนที่ 4 เป็นผลการวิจัย ส่วนที่ 5 เป็นการวิจารณ์ผลการวิจัย และสุดท้ายคือสรุปงานวิจัยนี้ ดังจะได้นำเสนอที่ละหัวข้อ และอธิบายรายละเอียดต่อไป

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในงานวิจัยนี้ มีทฤษฎีที่น่าสนใจมาใช้ในการสร้างแบบทดสอบ คือ รูปแบบคำสั่งสืบค้นข้อมูล นิพจน์ปรกติ (Regular Expression) และงานวิจัยอื่นๆที่เกี่ยวข้อง ดังนี้

รูปแบบคำสั่งสืบค้นข้อมูล

คำสั่งสืบค้นข้อมูล เป็นคำสั่งในการสืบค้นข้อมูลจากฐานข้อมูล โดยแบบทดสอบที่ใช้ในงานวิจัยนี้ ใช้รูปแบบคำสั่งสืบค้นข้อมูลที่สำคัญ ดังนี้

SELECT รายชื่อฟิลด์ หรือฟังก์ชัน

FROM ชื่อตาราง

WHERE เงื่อนไข

GROUP BY ชื่อฟิลด์ที่จัดกลุ่ม

ORDER BY ชื่อฟิลด์ [ASC, DESC]

รายชื่อฟิลด์ เป็นรายชื่อฟิลด์ที่ต้องการนำมาแสดงผล

ฟังก์ชัน เป็นฟังก์ชันประมวลผลกับข้อมูล เช่น ฟังก์ชัน sum เป็นการหาผลรวม ฟังก์ชัน max เป็นการหาค่ามากที่สุด ฟังก์ชัน min เป็นการหาค่าน้อยที่สุด ฟังก์ชัน average เป็นการหาค่าเฉลี่ย และฟังก์ชัน std เป็นการหาค่าส่วนเบี่ยงเบนมาตรฐาน และยังมีฟังก์ชันอื่นๆอีก แต่ในงานวิจัยนี้ มีแบบทดสอบใช้ฟังก์ชันเฉพาะดังกล่าวข้างต้น

ชื่อตาราง เป็นชื่อตารางที่ต้องการนำข้อมูลมาแสดงผล เงื่อนไข เป็นเงื่อนไขในการเลือกเรคคอร์ดเฉพาะที่ตรงกับเงื่อนไขมาแสดงผล

ชื่อฟิลด์ที่จัดกลุ่ม เป็นการจัดกลุ่มข้อมูลในฟิลด์ ที่มีค่าเหมือนกัน เพื่อประมวลผลฟังก์ชัน ดังนั้น คำสั่งส่วน GROUP BY

ตารางที่ 1 รูปแบบนิพจน์ปรกติ

รูปแบบ	ความหมาย
$\wedge n$	สายอักขระ n เป็นสายอักขระขึ้นต้นย่อหน้า
$n\$\wedge$	สายอักขระ n เป็นสายอักขระสุดท้ายของย่อหน้า
$?!n$	อักขระใดๆ ที่ไม่ตามด้วยสายอักขระ n
$\backslash s$	ช่องว่าง
n^*	มีสายอักขระ n เกิดขึ้น หลายครั้ง หรือไม่มีเลยก็ได้
$/i$	การพิจารณาไม่สนใจอักขระตัวเล็กหรือตัวใหญ่ กล่าวคืออักขระตัวเล็กหรือตัวใหญ่เหมือนกัน

งานวิจัยที่เกี่ยวข้อง

ระบบช่วยสร้างและวิเคราะห์ข้อสอบ (ปัฐมา, ชลนิสา และณัฐธนนท์, 2552) การพัฒนาเครื่องมือสร้างข้อสอบแบบหลายชุดโดยอัตโนมัติ (อรยา และสุดา, 2557) เป็นการเก็บรวบรวมข้อสอบ วิเคราะห์คุณภาพข้อสอบ และสามารถนำมาสร้างเป็นชุดข้อสอบชุดใหม่ ที่สลับตำแหน่งตัวเลือกได้

การสร้างโจทย์คำถาม แบบการจัดกลุ่ม (พินิจนันต์ และพงษ์เทพ, 2559) แบบสร้างตัวเลือกถูก-ผิด (สนาพร และพงษ์เทพ, 2559) การเลือกค่าแปรผันในโจทย์ ที่มีผลต่อวิธีการหาคำตอบ (สนาพร และพงษ์เทพ, 2560) การสร้างโจทย์คำถามจากโจทย์ต้นแบบ เพื่อใช้เป็นแบบทดสอบที่ไม่ซ้ำกัน (ทิพย์สุดา และพงษ์เทพ, 2560) เป็นการเตรียมโจทย์ต้นแบบ ที่มีค่าคงที่ ค่าแปรผัน และตัวแปร เมื่อสุ่มเลือกค่าแปรผัน และค่าตัวแปรใหม่ จะได้โจทย์ใหม่ที่ไม่ซ้ำเดิม

วิธีการดำเนินการวิจัย

วิธีการดำเนินการวิจัย แบ่งออกเป็น 3 ขั้นตอนหลัก 1. การสร้างแบบทดสอบ 2. การทำแบบทดสอบ และ 3. การตรวจคำตอบ แสดงขั้นตอนการดำเนินการวิจัย ได้ดังรูปที่ 1

มักจะมีฟังก์ชัน ที่อยู่ในส่วน SELECT เสมอ

ชื่อฟิลด์ที่เรียงลำดับ เป็นการกำหนดให้ข้อมูลที่นำมาแสดงผล ต้องเรียงลำดับเรคคอร์ดตามข้อมูลที่อยู่ในฟิลด์ ถ้ากำหนดวิธีการเรียงลำดับเป็น ASC หมายถึงการเรียงจากน้อยไปมาก ถ้ากำหนดเป็น DESC หมายถึงการเรียงจากมากไปน้อย แต่ถ้าไม่กำหนดวิธีการเรียงลำดับ จะหมายถึงให้เรียงจากน้อยไปมาก

นิพจน์ปรกติ

นิพจน์ปรกติ (Alfred and Jeffrey, 1992) เป็นวิธีการตรวจสอบอักขระว่ามีการเรียงอักขระตรงตามที่กำหนดในรูปแบบหรือไม่ แสดงความหมายของรูปแบบที่ใช้ในงานวิจัยนี้ดังตารางที่ 1

1. การสร้างแบบทดสอบ

แบบทดสอบประกอบด้วยโจทย์คำถามและช่องคำตอบ ถ้ามีการออกแบบแบบทดสอบดี โดยกำหนดช่องคำตอบที่ผู้ทำแบบทดสอบต้องตอบไว้อย่างเหมาะสม จะทำให้การตรวจคำตอบสามารถค้นหาคำตอบของผู้ทำแบบสอบถามได้ง่าย และนำไปตรวจคำตอบได้อย่างถูกต้อง ในงานวิจัยนี้แบ่งการสร้างแบบทดสอบเป็น 3 ขั้นตอน คือ แบบทดสอบต้นแบบ ข้อมูลตารางที่ใช้เป็นโจทย์ และการรวมเป็นแบบทดสอบ ดังจะได้อธิบายรายละเอียดทีละหัวข้อต่อไป

1.1 แบบทดสอบต้นแบบ

แบบทดสอบต้นแบบ สามารถมีได้หลายโจทย์ ดังนั้นต้องออกแบบช่องคำถาม และช่องคำตอบให้ผู้ทำแบบทดสอบอ่านคำถาม และตอบคำถามลงในช่องคำตอบของแต่ละข้อได้อย่างถูกต้อง

โจทย์แต่ละโจทย์อาจนำมาจากโจทย์ต้นแบบที่ต่างกัน ในงานวิจัยนี้มีโจทย์ต้นแบบไว้ 8 โจทย์ต้นแบบ แสดงดังตารางที่ 2 ในแต่ละโจทย์ต้นแบบ ประกอบด้วยตารางข้อมูลและโจทย์คำถาม โดยให้ผู้ทำแบบทดสอบตอบ 2 คำตอบ คือ 1) เขียน

คำสั่งสืบค้นข้อมูล และ 2) แสดงผลการสืบค้นข้อมูล แสดงตัวอย่างโจทย์ดังรูปที่ 2

จากโจทย์ต้นแบบตามจุดประสงค์ที่ 1-6 ในตารางที่ 2 สร้างเป็นโจทย์ข้อที่ 1-6 แต่จุดประสงค์ที่ 7 และ 8 ซึ่งเป็นแบบทดสอบเกี่ยวกับการหาผลฟังก์ชัน คือ ฟังก์ชันหาผลรวม (sum) ฟังก์ชันหาค่ามากที่สุด (max) ฟังก์ชันหาค่าน้อยที่สุด (min) ฟังก์ชันหาค่าเฉลี่ย (average) และฟังก์ชันหาค่าส่วนเบี่ยงเบนมาตรฐาน (std) รวม 5 ฟังก์ชัน จะถูกสุ่มการหาผลฟังก์ชัน เพื่อเป็นโจทย์ตามจุดประสงค์ที่ 7 และ 8 จำนวน 2 และ 3 โจทย์ตามลำดับ รวม 5 ข้อ

จากตารางที่ 2 ในส่วนของโจทย์ต้นแบบ เครื่องหมาย {ชื่อตัวแปร} จะถูกแทนที่ด้วยข้อความที่เหมาะสมจากคลังตาราง ในขั้นตอนการรวมเป็นแบบทดสอบ

1.2 คลังตารางที่ใช้เป็นโจทย์

จากแบบทดสอบต้นแบบมีการเตรียม {ชื่อตัวแปร} ไว้ ซึ่งพร้อมจะเปลี่ยนเป็นข้อความที่เหมาะสมในแต่ละโจทย์คำถาม ดังนั้นข้อความที่เหมาะสมดังกล่าว จะต้องเตรียมไว้ให้พร้อม เพื่อนำไปใช้ในขั้นตอนการรวมเป็นแบบทดสอบได้ต่อไป

จากแบบทดสอบที่ต้องการสร้างโจทย์เกี่ยวกับคำสั่งสืบค้นข้อมูล ดังนั้น ชื่อตาราง ชื่อฟิลด์ เงื่อนไขการกรองข้อมูล การเรียงลำดับตามชื่อฟิลด์ จะต้องกำหนดให้มีความสัมพันธ์กัน สามารถหาคำตอบได้ ซึ่งต้องเลือกชื่อตารางก่อน แล้วจึงเลือกชื่อฟิลด์ในตารางนั้น มากำหนดเป็นเงื่อนไขการกรองข้อมูล การเรียงลำดับ หรืออื่นๆได้

ตารางที่ใช้ในงานวิจัยนี้มี 7 ตาราง คือ Customers, invoices, Lists, Students, Grades, Subjects และ People โดย ตาราง People จะใช้เฉพาะจุดประสงค์ 7 และ 8 เพราะการหาผลฟังก์ชัน จะต้องเป็นการหาผลจากฟิลด์ประเภทเลขจำนวน และมีบางฟิลด์ที่มีข้อมูลซ้ำกัน ดังนั้น จะกำหนดให้ตาราง People มีฟิลด์ประเภทเลขจำนวนไว้โดยเฉพาะ เช่น ส่วนสูง, น้ำหนัก, เกรดเฉลี่ย และฟิลด์ที่มีค่าซ้ำกัน เช่น หลักสูตร, เพศ, จังหวัด, โรงเรียน เป็นต้น รายละเอียดของตารางแสดงได้ดังตารางที่ 3 ส่วนข้อมูลในแต่ละตารางจะมี 8-10 เรคอร์ดตามความเหมาะสม โดยข้อมูลในตารางจะถูกนำมาสลับข้อมูลฟิลด์ไปอยู่ต่างเรคอร์ด เพื่อใช้เป็นตารางข้อมูลเริ่มต้น ของแต่ละโจทย์คำถาม และบางข้อมูลจะถูกเลือก เพื่อใช้เป็นเงื่อนไขในการกรองข้อมูล จะได้พบทั้งเรคอร์ดที่ตรงตามเงื่อนไข และไม่ตรงตาม

เงื่อนไข ซึ่งจะทำให้โจทย์คำถามมีความเหมาะสม เกิดความท้าทายในการทำแบบทดสอบที่ไม่ซ้ำแบบกัน

จากตัวอย่าง 7 ตารางในคลังตาราง สามารถเพิ่มตารางเพื่อให้เป็นโจทย์เพิ่มเติมได้ โดยกำหนดรายละเอียดที่จำเป็นในการสร้างโจทย์คำถาม เพื่อให้ระบบสร้างโจทย์นำไปใช้ได้ถูกต้อง เช่น ชื่อตาราง ชื่อฟิลด์ ฟิลด์คำถาม ฟิลด์ LIKE และ ข้อมูลประมาณ 8-10 เรคอร์ด

1.3 การรวมเป็นแบบทดสอบ

แต่ละโจทย์ต้นแบบ จะเตรียมข้อมูลโจทย์ โดยสุ่มเลือกตารางจากคลังตารางมา 1 ตาราง นำรายละเอียดของตารางมาสุ่มเลือกข้อมูล กำหนดในรายการตัวแปร ตามที่โจทย์ต้นแบบต้องการ เช่น ชื่อฟิลด์ในตารางมี 5 ฟิลด์ จะสุ่มเลือกชื่อฟิลด์มา 3 ชื่อ กำหนดเป็น {field1} {field2} และ {field3} ประโยชน์ที่ได้จากการสุ่ม จะทำให้แม้โจทย์ต้นแบบ เลือกได้ตารางเดียวกัน แต่ชื่อฟิลด์ที่กำหนดเป็น {field1} {field2} และ {field3} ก็อาจจะไม่เหมือนกัน ทำให้ผู้ทำแบบทดสอบจะต้องสนใจโจทย์คำถามเฉพาะโจทย์นั้นๆ ไม่สามารถคัดลอกคำตอบจากโจทย์อื่นที่แม้เป็นชื่อตารางเดียวกันได้

ขั้นตอนการเตรียมข้อมูลโจทย์ มีขั้นตอนดังนี้

1. สุ่มเลือกตารางจากคลังตารางมา 1 ตาราง กำหนดให้เป็น Table
2. นำชื่อตารางของ Table กำหนดเป็น {table_name}
3. สุ่มเลือกชื่อฟิลด์มาจาก Table จำนวน 3 ฟิลด์ กำหนดเป็น {field1} {field2} และ {field3}
4. สุ่มเลือกชื่อฟิลด์มาจาก Table มา 2 ฟิลด์ กำหนดเป็น {field_where1} และ {field_where2}
5. สุ่มเลือกข้อมูลในฟิลด์ {field_where1} มา 1 ข้อมูล กำหนดเป็น {value_where1}
6. สุ่มเลือกข้อมูลในฟิลด์ {field_where2} มา 1 ข้อมูล กำหนดเป็น {value_where2}
7. สุ่มเลือกคำในกลุ่ม ("และ", "หรือ") มา 1 คำ กำหนดให้เป็น {op_where1} เพื่อใช้เป็นคำเชื่อมเงื่อนไข
8. สุ่มเลือกชื่อฟิลด์ LIKE ที่ไม่ซ้ำกับชื่อฟิลด์ใน {field_where1} หรือ {field_where2} มากำหนดเป็น {field_like1}

9. สุ่มเลือกคำในกลุ่ม ("ขึ้นต้นด้วย", "ลงท้ายด้วย", "มีคำว่า") มา 1 คำ กำหนดให้เป็น {like1}

10. สุ่มเลือกข้อมูลในฟิลด์ {field_like1} มา 1 ข้อมูล พร้อมตัดอักขระบางส่วนเหลือ 3 ตัวอักษร กำหนดเป็น {value_like1}

11. สุ่มเลือกชื่อฟิลด์จาก {field1}, {field2} หรือ {field3} มา 2 ฟิลด์ กำหนดเป็น {field_order1} และ {field_order2}

12. สุ่มเลือกคำในกลุ่ม ("น้อยไปมาก", "มากไปน้อย") กำหนดเป็น {order_method1} และ {order_method2}

13. กรณี Table มีฟิลด์ฟังก์ชัน (ในที่นี้เฉพาะตาราง People)

13.1 สุ่มเลือกคำในกลุ่ม ("ค่ามากที่สุด", "ค่าน้อยที่สุด", "ผลรวม", "ค่าเฉลี่ย", "ส่วนเบี่ยงเบนมาตรฐาน") กำหนดเป็น {functionmath}

13.2 สุ่มเลือกชื่อฟิลด์ GROUP BY มากำหนดเป็น {groupfield}

14. เปลี่ยน {field_where1}, {field_where2}, {field_like1}, {field_order1} และ {field_order2} จากชื่อฟิลด์ เป็นฟิลด์คำถาม เพื่อให้เหมาะสมกับโจทย์ภาษาไทย

จากข้อมูล {ชื่อตัวแปร} ข้างต้น นำไปแทนที่ในโจทย์ต้นแบบ ซึ่งเพียงพอกับโจทย์ต้นแบบที่ต้องการตัวแปรมากที่สุด จะได้โจทย์คำถามใหม่ พร้อมให้ผู้ทำแบบทดสอบได้ตอบคำถามต่อไป

2. การทำแบบทดสอบ

หลังจากสร้างแบบทดสอบเสร็จแล้ว จะส่งให้ผู้ทำแบบทดสอบโดยวิธีแบ่งปันไฟล์ โดยอนุญาตให้ผู้ทำแบบทดสอบสามารถแก้ไขไฟล์ได้ ในเวลาที่กำหนด เมื่อหมดเวลาทำแบบทดสอบจะยกเลิกการอนุญาตให้แก้ไข เปลี่ยนให้ผู้ทำแบบทดสอบมาเป็นผู้ดูได้แทน ซึ่งผู้ดู มีคุณสมบัติสามารถดูเนื้อหา แต่ไม่สามารถแก้ไขได้ ซึ่งเป็นการเหมาะสมเพราะเมื่อได้มีการตรวจคำตอบ ผู้ทำแบบทดสอบ ซึ่งขณะนี้มีสถานะเป็นผู้ดู ก็จะสามารถดูคำตอบ และทราบผลการตรวจได้

ในงานวิจัยนี้ใช้ ภูเก็ตเอกสาร (Google, 2017) ซึ่งผู้ทำแบบทดสอบต้องมีบัญชีของภูเก็ต เพื่อแบ่งปันไฟล์แบบทดสอบให้แก้ไข และยกเลิกการแก้ไขในเวลาที่กำหนดได้ โดยจำกัดไม่ให้ผู้แก้ไขมีสิทธิเพิ่มผู้แก้ไขอื่นเพิ่มเติม

3. การตรวจคำตอบ

การตรวจคำตอบจะนำไฟล์คำตอบมาตัดแยกที่ละโจทย์ คำถาม และนำแต่ละโจทย์คำถามมาตัดแยกเป็นส่วนต่างๆ เพื่อพิจารณาแต่ละส่วน ว่าส่วนใดถูก ส่วนใดผิด เพื่อแสดงผลการตรวจในแต่ละส่วนได้อย่างชัดเจน โดยนำไปเปรียบเทียบกับคำตอบที่ถูกต้องที่ได้เตรียมไว้ มีขั้นตอนย่อย 4 ขั้นตอน คือ การค้นหาคำตอบ การตัดแยกคำตอบ การเตรียมคำตอบที่ถูกต้อง และการตรวจคำตอบ ดังจะได้อธิบายรายละเอียดต่อไป

3.1 การค้นหาคำตอบของผู้ทำแบบทดสอบ

การค้นหาคำตอบในเอกสารแบบทดสอบ จะค้นหาจุดเริ่มต้นของโจทย์คำถามก่อน คือ "ข้อ i" โดย i เป็นหมายเลขโจทย์ที่ต้องการพิจารณา จากจุดเริ่มต้นของโจทย์นี้ จึงค้นหาคำว่า "คำสั่ง SQL" และคำว่า "ผลของคำสั่ง" ซึ่งในช่องถัดไปจะพบคำตอบที่เป็นคำสั่งสืบค้นข้อมูล และตารางผลของคำสั่ง SQL ที่ผู้ทำแบบทดสอบได้หาผลไว้

คำว่า "ข้อ i", "คำสั่ง SQL" และ "ผลของคำสั่ง" เป็นข้อความบรรทัดเดียวจึงกำหนด รูปแบบการค้นหาข้อความขึ้นต้น และลงท้าย คือ `/^ข้อ i$/`, `/^คำสั่ง SQL$/` และ `/^ผลของคำสั่ง$/` เมื่อกำหนด i เป็นหมายเลขข้อที่ต้องการค้นหา ซึ่งรูปแบบการค้นหาใช้เครื่องหมายแบบเดียวกับนิพจน์ปรกติ และเมื่อได้คำตอบของผู้ทำแบบทดสอบแล้ว จึงนำไปตัดแยกคำตอบในขั้นตอนต่อไป

3.2 การตัดแยกคำตอบ

คำตอบที่เป็นคำสั่งสืบค้นข้อมูล

คำสั่งสืบค้นข้อมูล แบ่งเป้าหมายของคำสั่งที่ถูกต้อง ที่วางอยู่หลังคำเฉพาะ 5 คำ คือ SELECT (1) FROM (2) WHERE (3) GROUP BY (4) ORDER BY (5) สามารถใช้นิพจน์ปรกติโดยใช้รูปแบบ (.*?) เพื่อจะได้ข้อความหลังคำเฉพาะนั้นได้ จากนั้นนำข้อความที่ได้มาตัดแยกเป็นคำตอบของส่วนนั้นๆต่อไป

กำหนดรูปแบบนิพจน์ปรกติ (R_i) เพื่อตัดแยกข้อความหลังคำเฉพาะดังนี้

U_{SQL} หมายถึงคำตอบของคำสั่งสืบค้นข้อมูล ของผู้ทำแบบทดสอบ

$R_{select} = /SELECT\s(.*)\sFROM/i$

$R_{from} = /FROM\s(?:WHERE|ORDER BY|GROUP BY)\s(.*)\s(?:WHERE\s|ORDER BY\s|GROUP BY\s|;|\Z|)\s(.*)\s/i$

$R_{where} = /WHERE\s(.*)?(GROUP BY|ORDER BY);|\$/i$

$R_{group} = /GROUP BY\s(.*)?(ORDER BY);|\$/i$

$R_{order} = /ORDER BY\s(.*)?;|\$/i$

การคัดแยกคำตอบจะนำรูปแบบของ R_i ไปทำการคัดแยกคำตอบ (U_{SQL}) โดยใช้ฟังก์ชัน Extract ซึ่งจะได้คำตอบส่วนต่างๆของผู้ทำแบบทดสอบ (U_i) เพื่อนำไปใช้ในการตรวจคำตอบต่อไป

i คือส่วนของคำตอบ คือส่วน SELECT, FROM, WHERE, GROUP BY หรือ ORDER BY

$U_{select} = \text{Extract}(R_{select}, U_{SQL}, [';', ['AS']]);$

$U_{from} = \text{Extract}(R_{from}, U_{SQL}, [';', []]);$

$U_{where} = \text{Extract}(R_{where}, U_{SQL}, ['AND', 'OR'], ['=', '>', '>=', '<', '<=', 'LIKE']);$

$U_{group} = \text{Extract}(R_{group}, U_{SQL}, [';', []]);$

$U_{order} = \text{Extract}(R_{order}, U_{SQL}, [';', []]);$

ฟังก์ชัน Extract จะรับค่ามา 4 ค่า คือ รูปแบบนิพจน์ปรกติ (r) ข้อความที่ต้องการคัดแยก (t) ตัวแบ่งระดับ 1 ($d1$) และ ตัวแบ่งระดับ 2 ($d2$) โดยมีขั้นตอนดังนี้

1. ใช้ r คัดแยก t เก็บไว้ที่ m ($m = r.exec(t)$)
2. ถ้า m ไม่มีข้อความที่คัดแยกได้จะคืนค่า null และออกจากฟังก์ชัน
3. นำรายการตัวแบ่งใน $d1$ คัดแยก m ออกเป็นส่วนๆ ถ้าไม่มีการคัดแยก m จะมีค่าเต็ม
4. นำรายการตัวแบ่งใน $d2$ คัดแยกแต่ละส่วนของ m ถ้า m_i ไม่มีการคัดแยก ส่วน m_i นั้นจะมีค่าเต็ม
5. คืนค่า m เป็นผลของฟังก์ชัน

คำตอบที่เป็นผลของคำสั่งสืบค้นข้อมูล

ผลของคำสั่งสืบค้นข้อมูล จะตอบอยู่ในรูปของตาราง โดยบรรทัดแรกสุดเป็นชื่อฟิลด์ บรรทัดอื่นๆที่เหลือ เป็นเรคอร์ดที่ตรงตามเงื่อนไข

3.3 การเตรียมคำตอบที่ต้องการ

การเตรียมคำตอบที่ต้องการ จะนำตัวแปรที่ใช้ในการตั้งโจทย์คำถาม มาสร้างเป็นคำสั่งสืบค้นข้อมูลที่ต้องการ โดยเปลี่ยนค่าตัวแปรที่ตอนแรก กำหนดให้เหมาะสมเป็นโจทย์คำถาม ต้องเปลี่ยนให้เป็นชื่อฟิลด์ตามรูปแบบคำสั่งสืบค้นข้อมูล

1. เปลี่ยน $\{field_where1\}$, $\{field_where2\}$, $\{field_like1\}$, $\{field_order1\}$ และ $\{field_order2\}$ ที่นำฟิลด์คำถามมาใช้ในการตั้งโจทย์คำถาม ต้องเปลี่ยนเป็นชื่อฟิลด์ที่ตรงตามชื่อฟิลด์ในตาราง

2. กรณี $\{like1\}$ เป็นคำว่า "ขึ้นต้นด้วย", "ลงท้ายด้วย" หรือ "มีคำว่า" จะเปลี่ยน $\{value_like1\}$ เป็น $\{value_like1\}\%$, $\%\{value_like1\}$ หรือ $\%\{value_like1\}\%$ เพื่อให้ตรงตามรูปแบบคำสั่ง LIKE ที่เครื่องหมาย % แทนอักขระใดๆ

1. กรณีการเรียงลำดับที่ $\{order_method1\}$ หรือ $\{order_method2\}$ เป็นคำว่า "น้อยไปมาก" หรือ "มากไปน้อย" จะเปลี่ยนเป็น ASC หรือ DESC ตามลำดับ

กำหนด C_{SQL} ดังนี้

SELECT $\{field1\}$, $\{field2\}$, $\{field3\}$

FROM $\{table_name\}$

WHERE $\{field_where1\} = \{value_where1\}$

$\{op_where1\} \{field_like1\}$ LIKE $\{field_like1\}$

GROUP BY $\{groupfield\}$

ORDER BY $\{field_order1\} \{order_method1\}$,

$\{field_order2\} \{order_method2\}$

การสร้างคำสั่งสืบค้นข้อมูลที่ต้องการนี้ จะทำบนพื้นฐาน {ชื่อตัวแปร} ที่นำไปใช้ในโจทย์ต้นแบบ ซึ่ง {ชื่อตัวแปร} ใด ที่ไม่ได้นำไปใช้สร้างโจทย์จะถูกทิ้งไป เมื่อแทนค่า {ชื่อตัวแปร} ใน C_{SQL} กำหนดเป็น C_{SQL}

การคัดแยกคำตอบแต่ละส่วนที่ต้องการ โดยนำ C_{SQL} ไปผ่านการคัดแยกคำตอบ (ขั้นตอน 3.2) จะได้ C_i เป็นคำตอบที่ต้องการของแต่ละส่วน

3.4 การตรวจคำตอบ

การตรวจคำสั่งสืบค้นข้อมูล

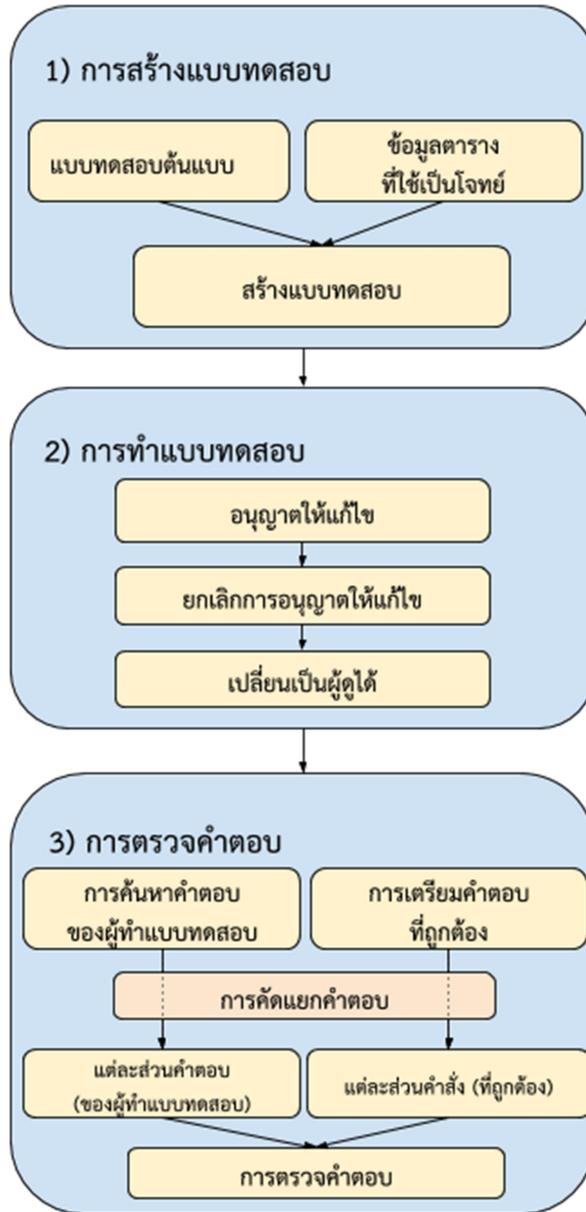
การตรวจคำสั่งสืบค้นข้อมูล จะนำเซตคำตอบของผู้ทำแบบสอบถาม (U_i) เปรียบเทียบกับเซตของคำตอบที่ต้องการ (C_i) ซึ่งจะได้เซตของคำตอบที่ตอบถูก (S_i)

จำนวนสมาชิกของ S_i คือจำนวนที่ตอบถูกต้อง

การตรวจสอบการตอบเกินคำตอบ โดยนำจำนวนสมาชิกของ U_i ลบด้วยจำนวนสมาชิกของ C_i กำหนดเป็น n_{O_i}

ถ้า $nQ_i = 0$ หมายถึง ตอบตามจำนวนที่ควรตอบ
 $nQ_i < 0$ หมายถึง ตอบน้อยกว่าจำนวนที่ควรตอบ
 $nQ_i > 0$ หมายถึง ตอบมากกว่าจำนวนที่ควรตอบ จะ

นำค่า nQ_i นี้ ไปหักคะแนน เพื่อป้องกันผู้ทำแบบทดสอบ ตอบ
แบบหวนคำตอบ



รูปที่ 1 ขั้นตอนการดำเนินการวิจัย

ข้อ 6	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ title lastname และ school จากตาราง Students เฉพาะคนที่มีคำนำหน้า นางสาว และเป็น คนที่รหัสนักศึกษา ที่มีค่าว่า 600 โดยเรียงลำดับตามฟิลด์ title จากมากไปน้อย ถ้าฟิลด์ title มีค่าเท่ากัน ให้เรียงลำดับตามฟิลด์ name จากน้อยไปมาก พร้อมทั้งแสดงผลของคำสั่งนั้น						
คำสั่ง SQL	<pre>SELECT title, lastname, school FROM Students WHERE title LIKE "นางสาว%" AND id LIKE "%600%" ORDER BY title DESC, name ASC</pre> <p style="text-align: center;">✓ ✗ (</p> <p style="text-align: center;">[เฉลย=SELECT title, lastname, school FROM Students WHERE title = "นางสาว" AND id LIKE "%600%" ORDER BY title DESC, name ASC]) ((SELECT=9/9, FROM=1/1, WHERE=5/15, GROUP BY=1/1, ORDER BY=6/6, รวม 22/32 คะแนน))</p>						
ผลของคำสั่ง	<table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>title</th> <th>lastname</th> <th>school</th> </tr> </thead> <tbody> <tr> <td>นางสาว</td> <td>คำอ่อนดี</td> <td>บ้านไร่พิทยาคม</td> </tr> </tbody> </table> <p style="text-align: center;">✓ ((FIELDS=6/6, ROWS=15/15, 21/21 คะแนน) (FIELDS=title,lastname,school, ROWS=(นางสาว,คำอ่อนดี,บ้านไร่พิทยาคม)))</p>	title	lastname	school	นางสาว	คำอ่อนดี	บ้านไร่พิทยาคม
title	lastname	school					
นางสาว	คำอ่อนดี	บ้านไร่พิทยาคม					

รูปที่ 2 ตัวอย่างโจทย์คำถาม คำตอบของผู้ทำแบบทดสอบ และแสดงผลการตรวจ

ตารางที่ 2 โจทย์ต้นแบบและจุดประสงค์

จุดประสงค์	โจทย์ต้นแบบ
(1.1) ผู้เรียนสามารถเลือกฟิลด์ที่จะแสดงผลได้	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ {field1} {field2} และ {field3} จากตาราง {table_name} โดย
(1.2) ผู้เรียนสามารถเรียงลำดับข้อมูลได้	เรียงลำดับตามฟิลด์ {field_order1} จาก{order_method1} พร้อมทั้งแสดงผลของคำสั่งนั้น
(2.1) เหมือน 1.1	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ {field1} {field2} และ {field3} จากตาราง {table_name} เฉพาะ
(2.2) ผู้เรียนสามารถกรองข้อมูลเฉพาะเรคอร์ดได้ โดยใช้ตัวดำเนินการเท่ากับ (=) เพียง 1 เงื่อนไข	{field_where1} {value_where1} พร้อมทั้งแสดงผลของคำสั่งนั้น
(3.1) เหมือน 1.1	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ {field1} {field2} และ {field3} จากตาราง {table_name} เฉพาะ
(3.2) ผู้เรียนสามารถใช้ตัวดำเนินการ and, or ได้ โดยเพิ่ม 2.2 ให้มี 2 เงื่อนไข	{field_where1} {value_where1} {op_where1} {field_where2} {value_where2} พร้อมทั้งแสดงผลของคำสั่งนั้น
(4.1) เหมือน 1.1	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ {field1} {field2} และ {field3} จากตาราง {table_name} เฉพาะ
(4.2) เหมือน 1.2	{field_like1} {like1} {value_like1} โดยเรียงลำดับตามฟิลด์ {field_order1} จาก{order_method1}
(4.3) เหมือน 2.2 แต่ใช้ตัวดำเนินการ LIKE	พร้อมทั้งแสดงผลของคำสั่งนั้น
(5.1) เหมือน 1.1	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ {field1} {field2} และ {field3} จากตาราง {table_name} เฉพาะ
(5.2) เหมือน 4.3	{field_like1} {like1} {value_like1} โดยเรียงลำดับตามฟิลด์ {field_order1} จาก{order_method1} ถ้า
(5.3) เหมือน 1.2 และเพิ่มการเรียงลำดับรองกรณีแรกมีค่าเท่ากัน	ฟิลด์ {field_order1} มีค่าเท่ากัน ให้เรียงลำดับตามฟิลด์ {field_order2} จาก{order_method2} พร้อมทั้งแสดงผลของคำสั่งนั้น
(6.1) เหมือน 1.1	จงเขียนคำสั่ง SQL เพื่อแสดงข้อมูลฟิลด์ {field1} {field2} และ {field3} จากตาราง {table_name} เฉพาะ
(6.2) เหมือน 3.2 แต่ใช้เงื่อนไข = และ LIKE	{field_where1} {value_where1} {op_where1} {field_like1} {like1} {value_like1} โดยเรียงลำดับตาม
(6.3) เหมือน 5.3	ฟิลด์ {field_order1} จาก{order_method1} ถ้าฟิลด์ {field_order1} มีค่าเท่ากัน ให้เรียงลำดับตามฟิลด์ {field_order2} จาก{order_method2} พร้อมทั้งแสดงผลของคำสั่งนั้น
(7.1) ผู้เรียนสามารถหาผลฟังก์ชันการรวมได้	จากตาราง {table_name} จงเขียนคำสั่ง SQL เพื่อหา{functionmath} ของ{field1} พร้อมทั้งแสดงผลของคำสั่งนั้น
(8.1) เหมือน 7.1 และเพิ่มการจัดกลุ่ม	จากตาราง {table_name} จงเขียนคำสั่ง SQL เพื่อหา{functionmath} ของ{field1} โดยจำแนกในแต่ละ {groupfield} พร้อมทั้งแสดงผลของคำสั่งนั้น

ตารางที่ 3 รายละเอียดตารางที่ใช้กำหนดเป็นโจทย์คำถาม

ชื่อตาราง	ชื่อฟิลด์	ฟิลด์คำถาม	ชื่อฟิลด์ LIKE	ชื่อฟิลด์ฟังก์ชัน	ชื่อฟิลด์ GROUP BY
Customers	customer_id, name, sex, career, email, province	รหัสลูกค้า, ชื่อลูกค้า, เพศ, อาชีพ, อีเมล, จังหวัด	customer_id, name, career, email, province	-	-
invoices	customer_id, invoice_id, seller_id, date, discount	รหัสลูกค้า, รหัสใบเสร็จ, รหัสผู้ขาย, วันที่ซื้อ, ส่วนลด	customer_id, invoice_id, seller_id, date	-	-
Lists	invoice_id, goods_id, goods, amount, price	รหัสใบเสร็จ, รหัสสินค้า, ชื่อสินค้า, จำนวน, ราคา	invoice_id, goods_id, goods	-	-
Students	id, title, name, lastname, email, school	รหัสนักศึกษา, คำนามหน้า, ชื่อ, นามสกุล, อีเมล, มาจากโรงเรียน	id, title, name, lastname, email, school	-	-
Grades	id, term, c_year, subject_id, section, subject_gr_code, grade	รหัสนักศึกษา, ปีหลักสูตร, การศึกษา, รหัสวิชา, เซกชัน, รหัสกลุ่มเรียน, เกเรตที่ได้	id, term, c_year, subject_id	-	-
Subjects	subject_id, subject_name, credit, credit_hour	รหัสวิชา, ชื่อวิชา, จำนวนหน่วยกิต, จำนวนหน่วยกิต (ชั่วโมงเรียน)	subject_id, subject_name, credit_hour	-	-
People	Id, course, GPA, sex, weight, height, province, school	รหัสผู้ใช้, หลักสูตร, เกเรดเฉลี่ยรวม, เพศ, น้ำหนัก, ส่วนสูง, จังหวัด, โรงเรียน	Id, province, school	GPA, weight, height	course, sex, province, school
ชื่อฟิลด์	- ชื่อฟิลด์ที่ใช้ในคำสั่งสืบค้นข้อมูล	ฟิลด์คำถาม	- ชื่อความที่เหมาะสมแทนชื่อฟิลด์ เพื่อใช้ในการตั้งคำถาม		
ชื่อฟิลด์ LIKE	- ชื่อฟิลด์ที่สามารถนำไปกำหนดเป็นเงื่อนไข LIKE	ชื่อฟิลด์ฟังก์ชัน	- ชื่อฟิลด์ที่สามารถนำไปกำหนดเป็นค่าผลฟังก์ชัน		
ชื่อฟิลด์ GROUP BY	- ชื่อฟิลด์ที่สามารถนำไปกำหนดเป็นการจัดกลุ่ม (GROUP BY)				

การเปรียบเทียบคำตอบ

การเปรียบเทียบคำตอบ แบ่งออกเป็น 2 แบบ คือแบบ อินเทอร์เน็ตเซกชันเซต และแบบลำดับสมาชิก

แบบอินเทอร์เน็ตเซกชันเซต เป็นการหาสมาชิก ที่เป็นสมาชิกของทั้ง 2 เซต และคี่ค่าสมาชิกที่เป็นสมาชิกของทั้ง 2 เซตนั้น แต่เนื่องจากเซตที่นำมาใช้นี้ ภายในมีสมาชิกที่ซ้ำกันได้ และเพื่อไม่ให้เปรียบเทียบสมาชิกเกิน 1 ครั้ง ในการเปรียบเทียบนี้ จึงเพิ่มลำดับสมาชิกในกลุ่มสมาชิกที่ซ้ำกัน เช่น เดิม $A = \{a, a, b, b, c, d\}$ จะปรับสมาชิกใหม่เป็น $A^* = \{a1, a2, b1, b2, c1, d1\}$ เมื่อดำเนินการอินเทอร์เน็ตเซกชันจะได้ผลถูกต้อง

ตัวอย่างเช่น คำตอบที่ถูกต้องคือ $C = \{a, b, c\}$ แต่ผู้ตอบได้ตอบคือ $D = \{c, b, b\}$ จะปรับใหม่เป็น $C^* = \{a1, b1, c1\}$ และ $D^* = \{c1, b1, b2\}$ เมื่อตรวจคำตอบ $C^* \cap D^* = \{c1, b1\}$ จะได้คำตอบถูก 2 คำตอบและผิด 1 คำตอบ

แบบลำดับสมาชิก เป็นการเปรียบเทียบสมาชิก ตามลำดับที่ตรงกัน และคี่ค่าสมาชิกเฉพาะลำดับที่ตรงกัน

การตรวจตารางข้อมูล

การตรวจตารางข้อมูล ซึ่งเป็นผลของคำสั่งสืบค้นข้อมูล แบ่งออกเป็น 2 ส่วนคือ ชื่อฟิลด์ และเรคอร์ด กรณีของชื่อฟิลด์ สามารถสลับลำดับกันได้ (ตรวจแบบอินเทอร์เน็ตเซกชันเซต) แต่ในส่วนของเรคอร์ด มี 2 กรณีคือ ถ้าคำสั่งสืบค้นข้อมูล มีส่วนของ ORDER BY จะต้องตรวจเรคอร์ดแต่ละเรคอร์ดตามลำดับ (ตรวจแบบลำดับสมาชิก) แต่ถ้าไม่มีส่วน ORDER BY จะตรวจแบบอินเทอร์เน็ตเซกชันเซต

ผลการวิจัย

การพัฒนาตัวแบบทดสอบ โดยใช้กูเกิลสคริปต์ (Google, 2017) สร้างไฟล์ในกูเกิลเอกสาร ให้เป็นแบบทดสอบ คำสั่งสืบค้นข้อมูล และทำการแบ่งปันไฟล์ให้ผู้ทำแบบทดสอบ

สามารถเข้าใช้แบบแก้ไขได้ และยกเลิกการแบ่งปันแบบแก้ไข เป็นแบบดูได้เพียงอย่างเดียว เมื่อหมดเวลาทำแบบทดสอบ

จากการทดลองนำไปใช้กับนักศึกษาจำนวน 43 คน คนละ 2 ครั้ง ครั้งที่ 1 เป็นการทดลองระบบ เพื่อตรวจสอบการตรวจคำตอบว่าสามารถตรวจคำตอบได้ถูกต้อง ตามความหลากหลายคำตอบ และหลากหลายรูปแบบคำตอบ ขึ้นอยู่กับความถนัดของผู้ทำแบบทดสอบ เช่น การเว้นวรรค การขึ้นบรรทัดใหม่ การตอบโดยใช้ตัวอักษรตัวเล็กหรือตัวใหญ่ เป็นต้น ครั้งที่ 2 แบบทดสอบเพื่อเก็บคะแนน โดยในแต่ละครั้งจะสร้างแบบทดสอบ แบ่งออกเป็น 2 ชุด ชุดที่ 1 ตรงตามจุดประสงค์ที่ 1-6 รวม 6 ข้อ และชุดที่ 2 ตรงตามจุดประสงค์ที่ 7-8 ครอบคลุมฟังก์ชันหาผลรวม หาค่ามากที่สุด น้อยที่สุด ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน รวม 5 ข้อ เป็นการสุ่มสร้างโจทย์ที่ได้โจทย์ไม่ซ้ำกัน รวมการสร้างโจทย์ทั้งสิ้น 946 โจทย์ ใช้เวลาสร้างโจทย์โดยเฉลี่ยโจทย์ละ 2.56 วินาที ให้เวลาทำแบบทดสอบชุดละ 60 นาที โดยเฉลี่ยโจทย์ละ 11 นาที เมื่อนำมาตรวจคำตอบ สามารถตรวจคำตอบได้อย่างถูกต้อง โดยเฉลี่ยโจทย์ละ 12.85 วินาที

เปรียบเทียบกับ การตรวจโดยใช้คนตรวจ สมมติ ใช้คนตรวจอย่างรวดเร็วสุด ใช้เวลาข้อละ 1 นาที จำนวน 946 ข้อ รวมใช้เวลา 946 นาที หรือใช้เวลาตรวจ 15 ชั่วโมง 46 นาที โดยไม่หยุดพัก

วิจารณ์ผลการวิจัย

จากการสร้างแบบทดสอบโดยมีโจทย์คำถามที่ไม่ซ้ำเดิม นำไปให้ผู้ทำแบบทดสอบได้ฝึกทำ และมีการตรวจคำตอบ ทำให้ผู้ทำแบบทดสอบทราบถึงข้อผิดพลาดของตนเองได้ ซึ่งเป็นข้อมูลพื้นฐานที่จะไม่ทำผิดซ้ำอีก และจากการสร้างแบบทดสอบโดยมีโจทย์คำถามที่ไม่ซ้ำเดิม ทำให้ผู้ทำแบบทดสอบไม่สามารถท่องจำคำตอบได้ เพราะเมื่อมีการทำแบบทดสอบอีกครั้ง จะได้รับโจทย์ใหม่ จะต้องฝึกหาคำตอบที่ถูกต้องตามโจทย์ที่ได้รับใหม่นั้น

การสร้างโจทย์คำถาม และการตรวจคำตอบ เป็นภาระหน้าที่ของผู้สอน แต่ในระบบนี้ได้ทำเป็นขั้นตอนวิธีให้โปรแกรมคอมพิวเตอร์สร้างโจทย์ และตรวจคำตอบ ซึ่งกระบวนการฝึกทำแบบทดสอบนี้ สามารถทำได้บ่อยครั้ง ตามความจำเป็นของผู้สอนและผู้ทำแบบทดสอบ โดยไม่เป็นการเพิ่มภาระของผู้สอนแต่อย่างใด

สรุปผลการวิจัย

การสร้างแบบทดสอบคำสั่งสืบค้นข้อมูล เป็นการสร้างโจทย์คำถาม โดยมีเป้าหมายให้ผู้ทำแบบทดสอบฝึกเขียนคำสั่งสืบค้นข้อมูล และมีการตรวจคำตอบ โดยแสดงคำตอบที่ถูกต้อง จากการทดลองใช้กับนักศึกษาจำนวน 43 คน จำนวน 2 ครั้ง ครั้งที่ 11 โจทย์ ซึ่งระบบสร้างโจทย์จำนวน 946 โจทย์ เพื่อให้แต่ละคนได้รับโจทย์ที่ไม่เหมือนกัน ใช้เวลาสร้างโจทย์เฉลี่ยโจทย์ละ 2.56 วินาที ให้เวลาผู้ทำแบบทดสอบเฉลี่ยโจทย์ละ 11 นาที เมื่อนำมาตรวจ ระบบใช้เวลาตรวจโดยเฉลี่ยโจทย์ละ 12.85 วินาที พร้อมแสดงผลการตรวจ และคำตอบที่ถูกต้องให้ผู้ทำแบบทดสอบได้ทราบ

กิตติกรรมประกาศ

งานวิจัยนี้สำเร็จลุล่วงไปได้ด้วยดี ด้วยความร่วมมือของนักศึกษาวิทยาการคอมพิวเตอร์/58 และเทคโนโลยีสารสนเทศ/58 ที่ได้ช่วยทดลองทำแบบทดสอบ และแนะนำการตรวจคำตอบ ทำให้มีแบบทดสอบที่ตรวจคำตอบได้อย่างถูกต้อง

เอกสารอ้างอิง

- ทิพย์สุตา เพ็ชรไทย และพงษ์เทพ รักภักวาศ์. (2560). การสร้างโจทย์คำถามจากโจทย์ต้นแบบ เพื่อใช้เป็นแบบทดสอบที่ไม่ซ้ำกัน. ใน: พะเยาวิจัย ครั้งที่ ๖. มหาวิทยาลัยพะเยา. พะเยา. 1572-1581.
- ธนพร สรรพรอด และพงษ์เทพ รักภักวาศ์. (2559). การสร้างโจทย์คำถาม และการสร้างตัวเลือกถูกและผิดโดยอัตโนมัติ. ใน: สวนสุนันทา วิชาการระดับชาติ ครั้งที่ 4. มหาวิทยาลัยราชภัฏสวนสุนันทา, กรุงเทพฯ. 32-39.
- ธนพร สรรพรอด และพงษ์เทพ รักภักวาศ์. (2560). การสร้างโจทย์คำถาม ที่การเลือกค่าแปรผันมีผลต่อวิธีการหาคำตอบ. ใน: บัณฑิตศึกษา ครั้งที่ 2. มหาวิทยาลัยราชภัฏมหาสารคาม, มหาสารคาม. 554-562.
- ปัฐมา กระต่ายทอง, ชลนิสา เทียมบัณฑิตย์ และณัฐธนนท์ หงส์วริทธิ์ธร. (2552). ระบบช่วยสร้างและวิเคราะห์ข้อสอบ. ใน: NCCIT2009. มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, กรุงเทพฯ. 944-950
- พินิจนันต์ สุขสวัสดิ์ และพงษ์เทพ รักภักวาศ์. (2559). การสร้างโจทย์คำถามแบบการจัดกลุ่มและการตรวจคำตอบโดยใช้เทคนิคสมาชิกเซต. ใน: ราชภัฏวิจัย ครั้งที่ ๔. มหาวิทยาลัยราชภัฏบุรีรัมย์, บุรีรัมย์. 78-86.
- อรยา ปรีชาพานิช และสุภา เจริญมนตรี. (2557). การพัฒนาเครื่องมือสร้างข้อสอบแบบหลายชุดโดยอัตโนมัติ. วารสารมหาวิทยาลัยทักษิณ 17(2): 60-68.

อุไรวรรณ รักผกาวงศ์. (2560). ระบบฐานข้อมูล. มหาวิทยาลัยราชภัฏพิบูล
สงคราม: พิษณุโลก. 17-103.

Aho, A., Ullman, V., and Jeffrey, D. (2000). Foundations of
Computer Science. C ed. New York: Computer Science
Press.

Google. (2017). Google Docs. Available on: https://gsuite.google.com/intl/en_id/products/docs/ Retrieved. April 24, 2017

ISO Standards. (2003). Information technology: Database languages: SQL. Available on: <https://www.iso.org/standard/34132.html>. Retrieved. April 24, 2017.

