

เอกสารประกอบการอบรม



Empower world class
teaching & learning experiences

การจัดการเรียนรู้สะเต็มศึกษา ด้วยการเขียนโปรแกรม **SCRATCH**



ลิขสิทธิ์โดย สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (สสวท.)



คำนำ

สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (สสวท.) เป็นหน่วยงานหลักในการยกระดับคุณภาพการศึกษาด้านวิทยาศาสตร์ คณิตศาสตร์และเทคโนโลยีของประเทศ โดยมีบทบาทในการริเริ่ม ส่งเสริมการศึกษาค้นคว้า วิจัย และพัฒนาหลักสูตร วิธีการจัดการเรียนรู้ การประเมินผลการเรียนรู้ด้านวิทยาศาสตร์ คณิตศาสตร์และเทคโนโลยี ในระดับการศึกษาขั้นพื้นฐาน

รูปแบบการจัดการเรียนรู้แบบสะเต็มศึกษา เป็นการเรียนรู้รูปแบบหนึ่งที่จะบูรณาการวิทยาศาสตร์ เทคโนโลยี วิศวกรรมศาสตร์ และคณิตศาสตร์เข้าด้วยกัน โดยเน้นการนำความรู้ไปใช้แก้ปัญหาในชีวิตจริง และการประกอบอาชีพในอนาคตผ่านประสบการณ์ในการทำกิจกรรมการเรียนรู้แบบโครงงานเป็นฐาน (Project-Based Learning) หรือกิจกรรมการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning)

สสวท. มีนโยบายในการดำเนินโครงการพัฒนาและสนับสนุนการดำเนินงานศูนย์สะเต็มศึกษาแห่งชาติ ศูนย์สะเต็มศึกษาภาค และโรงเรียนเครือข่ายสะเต็มศึกษา โดยมีความมุ่งมั่นที่จะส่งเสริมการจัดการเรียนรู้ตามแนวทางสะเต็มศึกษาให้เกิดขึ้นในทุกระดับชั้น เพื่อขับเคลื่อนการจัดการเรียนรู้วิทยาศาสตร์ คณิตศาสตร์และเทคโนโลยีตามแนวทางสะเต็มศึกษา ให้เยาวชนไทยได้พัฒนาทักษะกระบวนการคิดวิเคราะห์ การแก้ปัญหา ความคิดสร้างสรรค์ ทั้งนี้ สสวท. ซึ่งมีหน้าที่หลักในการพัฒนาหลักสูตรและกิจกรรมการเรียนรู้ในสาระเทคโนโลยีสารสนเทศและการสื่อสาร จึงได้พัฒนากิจกรรมการเรียนรู้แบบสะเต็มศึกษาด้วยการเขียนโปรแกรม Scratch ที่เน้นการนำเทคโนโลยีสารสนเทศเข้ามาบูรณาการ เพื่อให้ครูในโรงเรียน และครูผู้สนใจเห็นแนวทาง และรูปแบบการจัดกิจกรรมสะเต็มศึกษา และสามารถนำรูปแบบกิจกรรมไปขยายผล หรือประยุกต์ใช้ในชั้นเรียนต่อไปได้

การจัดทำเอกสารชุดนี้ได้รับความร่วมมืออย่างดียิ่งจากครูผู้สอนวิชาเทคโนโลยีสารสนเทศและการสื่อสาร วิทยาศาสตร์ และคณิตศาสตร์ ผู้ทรงคุณวุฒิ และนักวิชาการ สสวท. จึงขอขอบคุณไว้ ณ ที่นี้ และหวังเป็นอย่างยิ่งว่าเอกสารฉบับนี้จะเป็นประโยชน์แก่สถานศึกษา และผู้ที่เกี่ยวข้องทุกฝ่ายที่จะนำหลักสูตรนี้ไปปรับใช้กับกิจกรรมการเรียนรู้ในชั้นเรียนตามความเหมาะสม และหากมีข้อเสนอแนะใดที่จะให้เอกสารนี้สมบูรณ์ยิ่งขึ้น โปรดแจ้ง สสวท. ทราบด้วย จักขอบพระคุณยิ่ง

สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี

มีนาคม 2559



คำชี้แจง

เอกสารฉบับนี้ เป็นเอกสารประกอบการอบรมครู หลักสูตรการจัดการเรียนรู้สะเต็มศึกษา ด้วยการเขียนโปรแกรม Scratch ประกอบด้วยเนื้อหาหลัก 2 ส่วน คือ

ส่วนที่ 1 เรื่องการเขียนโปรแกรม Scratch ซึ่งเน้นเนื้อหาเกี่ยวกับการใช้งาน และฝึกทักษะการโปรแกรมด้วยโปรแกรม Scratch

ส่วนที่ 2 เรื่อง การจัดการเรียนรู้สะเต็มศึกษาด้วยการเขียนโปรแกรม Scratch โดยจะเป็นรูปแบบ การจัดการเรียนรู้ตามแนวทางสะเต็มศึกษา ด้วยการนำโปรแกรม Scratch มาเป็นเครื่องมือในการจัดกิจกรรมที่จะบูรณาการกับวิทยาศาสตร์และคณิตศาสตร์ ซึ่งรูปแบบกิจกรรมนี้จะเป็นเพียงแนวทางให้ครูนำไปประยุกต์ใช้ต่อไป

ครูสามารถดาวน์โหลดไฟล์เอกสารพร้อมทั้งไฟล์สื่อที่ใช้ประกอบการดำเนินกิจกรรม ได้ที่ <http://oho.ipst.ac.th>

สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี
มีนาคม 2559

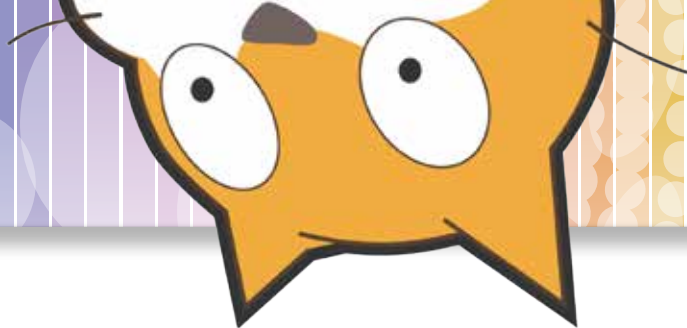


สารบัญ

การเขียนโปรแกรม Scratch

กิจกรรมที่ 1 รู้จักโปรแกรม Scratch	2
ใบงานที่ 1.1 มารู้จักโปรแกรม Scratch กันเถอะ	5
ใบงานที่ 1.2 โปรเจกต์แรกของฉัน	7
ใบความรู้ที่ 1.1 รู้จักโปรแกรม Scratch	12
ใบความรู้ที่ 1.2 การสร้างภาพเคลื่อนไหวอย่างง่าย	20
กิจกรรมที่ 2 การสร้างภาพเคลื่อนไหว	21
ใบงานที่ 2.1 เพิ่มและลบฉาก	24
ใบงานที่ 2.2 พิกัดและทิศทาง	27
ใบความรู้ที่ 2.1 ฉากและตัวละคร	30
ใบความรู้ที่ 2.2 พิกัดและทิศทาง	36
กิจกรรมที่ 3 สนุกกับเสียง	38
ใบงานที่ 3.1 เล่นกับเสียง	40
ใบงานที่ 3.2 ดนตรีจินตนาการ	43
ใบความรู้ 3.2 โน้ตดนตรี	45
กิจกรรมที่ 4 บทละครโต้ตอบ	47
ใบงานที่ 4.1 สร้างบทละครโต้ตอบ	49
ใบความรู้ที่ 4.1 คำสั่งบล็อก say บล็อก ask และบล็อก join	56
กิจกรรมที่ 5 กระจายสารและรับสาร	58
ใบงานที่ 5.1 กระจายสารและรับสาร	60
ใบงานที่ 5.2 สนทนาพาเพลิน	63
ใบความรู้ที่ 5.1 การกระจายสารและรับสาร	65
กิจกรรมที่ 6 มุมและทิศทาง	70
ใบงานที่ 6.1 เส้นตรงหลงทาง	72
ใบงานที่ 6.2 มุมและทิศทาง	75
ใบความรู้ที่ 6.1 มุมและทิศทาง	80
กิจกรรมที่ 7 ภาพสวยด้วยมือเรา	83
ใบงานที่ 7.1 ลากเส้นเล่นลาย	86
ใบงานที่ 7.2 ฟังก์ชัน	90
ใบงานที่ 7.3 ตัวป้อน	91





ใบความรู้ที่ 7.1 การสั่งงานแบบวนซ้ำ	94
ใบความรู้ที่ 7.2 ฟังก์ชัน	97
แบบฝึกหัดเพิ่มเติม	99
กิจกรรมที่ 8 เงื่อนไขและตัวแปร	104
ใบงานที่ 8.1 มารู้จักเงื่อนไขกันเถอะ	107
ใบงานที่ 8.2 เกมทายตัวเลข	109
ใบความรู้ที่ 8.1 เงื่อนไข	111
ใบความรู้ที่ 8.2 ตัวแปร	113
กิจกรรมที่ 9 เกมสร้างสรรค์	115
ใบงานที่ 9.1 สัมผัสสนุก	118
ใบความรู้ที่ 9.1 สัมผัสสนุก	125
แบบฝึกหัดเพิ่มเติม	127
 การจัดการเรียนรู้สะเต็มศึกษาด้วยโปรแกรม Scratch	
กิจกรรมที่ 1 รู้จัก IPST LINK	132
ใบงานที่ 1.1 มาตรวจสอบอุปกรณ์กัน	135
ใบงานที่ 1.2 กดกระโดด	136
ใบงานที่ 1.3 ซ้ายหัน ขวาหัน	137
ใบงานที่ 1.4 ย่อ-ขยาย	138
ใบงานที่ 1.5 จู๊ จู๊ อย่าดังนะ	139
ใบงานที่ 1.6 ปรับแสง เปลี่ยนสี	140
ใบความรู้ที่ 1.1 รู้จักชุดแผงวงจร IPST LINK	141
ใบความรู้ที่ 1.2 ทดสอบการเชื่อมต่อ	143
ใบความรู้ที่ 1.3 แผงวงจร IPST LINK และตัวตรวจจ็ับ	149
กิจกรรมที่ 2 ประยุกต์ใช้งานตัวตรวจจ็ับ	152
ใบงานที่ 2.1 E-Librarian	154
ใบงานที่ 2.2 Smart Home	156
กิจกรรมที่ 3 สร้างชิ้นงาน STEM	159
ใบงานที่ 3.1 สร้างชิ้นงาน STEM	161
แบบประเมินผลงานและการนำเสนอ	163



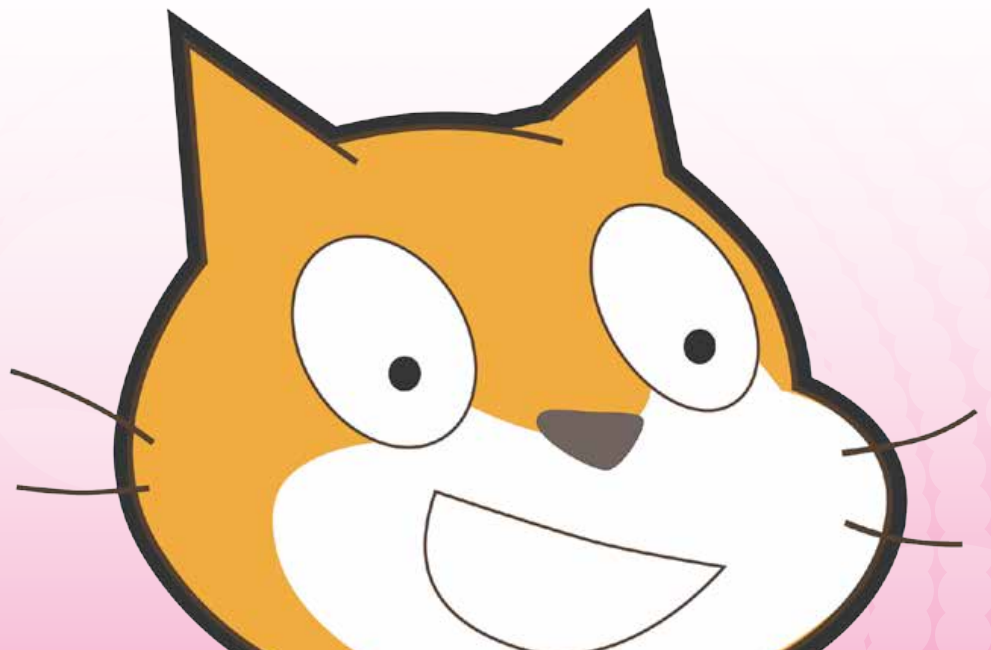
ภาคผนวก	165
ก โปรแกรม Scratch	166
ข การติดตั้งโปรแกรม Scratch	174
ค กลุ่มบล็อกคำสั่งในโปรแกรม Scratch	177
ง การส่งออกและนำเข้าตัวละคร	190
จ การแปลงไฟล์ Scratch ให้เป็น .exe	192
ฉ การใช้งานคำสั่งอื่นๆ	194
คณะผู้พัฒนาเอกสารการจัดการเรียนรู้สะเต็มศึกษาด้วยโปรแกรม Scratch ครั้งที่ 1	218





การเขียนโปรแกรม

SCRATCH



กิจกรรมที่ 1 รู้จักโปรแกรม Scratch

1. ผลการเรียนรู้

- 1.1 อธิบายขั้นตอนและเขียนโปรแกรมสั่งงานคอมพิวเตอร์
- 1.2 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม

2. สารการเรียนรู้

- 2.1 ความหมายของการโปรแกรม
- 2.2 ส่วนประกอบของโปรแกรม Scratch
- 2.3 การกำหนดค่าในบล็อกของโปรแกรม Scratch
- 2.4 การเขียนสคริปต์ให้ตัวละครเคลื่อนที่

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 อธิบายหลักการโปรแกรมเบื้องต้น
- 3.2 บอกส่วนประกอบของโปรแกรม Scratch
- 3.3 ใช้งานโปรแกรมเบื้องต้นและการกำหนดค่าในบล็อกของโปรแกรม Scratch
- 3.4 เขียนสคริปต์ให้ตัวละครเคลื่อนที่ด้วยกลุ่มบล็อก Motion เช่น บล็อก move, If on edge bounce
- 3.5 เปลี่ยนชุดตัวละครเพื่อให้มองเห็นเป็นภาพเคลื่อนไหวด้วยกลุ่มบล็อก Looks เช่น บล็อก Next costume

4. แนวคิด

การโปรแกรม หมายถึง การสร้างชุดคำสั่งเพื่อสั่งให้คอมพิวเตอร์ทำงาน Scratch เป็นภาษาคอมพิวเตอร์ที่มีคำสั่งสำหรับสั่งให้คอมพิวเตอร์ทำงานต่างๆ เช่น ภาพเคลื่อนไหว แบบจำลองทางวิทยาศาสตร์ เกม ดนตรี ศิลปะ สื่อแบบมีปฏิสัมพันธ์ การโปรแกรมภาษา Scratch จะใช้บล็อกคำสั่งมาวางต่อกันแทนการพิมพ์คำสั่งมีตัวละครและฉากที่ใช้สร้างเรื่องราวต่างๆ จึงเหมาะกับการสอนเขียนโปรแกรมที่ส่งเสริมความคิดริเริ่มสร้างสรรค์ การคิดอย่างมีระบบและการทำงานร่วมกัน นอกจากนี้ยังสามารถแลกเปลี่ยนและเผยแพร่ผลงานผ่านทางเว็บไซต์ <http://scratch.mit.edu/> ได้อีกด้วย

ชิ้นงานในโปรแกรม Scratch เรียกว่าโปรเจกต์ จะประกอบด้วย 3 ส่วนสำคัญ ได้แก่ เวที (Stage) ตัวละคร (Sprite) และสคริปต์ (Script) โดยแต่ละส่วนเกี่ยวข้องสัมพันธ์กันและทำงานไปพร้อมๆ กัน

การสร้างโปรเจกต์ทำได้โดยการเขียนสคริปต์ (ชุดคำสั่ง) ซึ่งใช้การลากวางบล็อก (คำสั่ง) เรียงต่อกันเพื่อสั่งตัวละคร ณ ตำแหน่งต่างๆ บนเวที ให้ทำงานต่างๆ

คำสั่งพื้นฐานที่สั่งให้ตัวละครเคลื่อนไหว แสดงออก ควบคุมขั้นตอนการทำงานของคำสั่ง จะใช้กลุ่มบล็อก Motion, Looks, และ Control ตามลำดับ

การบันทึกโปรเจกต์เพื่อเก็บไว้ใช้งานหรือแก้ไข สามารถทำได้โดยคลิกเมนู File -> Save หรือ Save As โปรเจกต์จะถูกบันทึกไว้ที่โฟลเดอร์ Documents\Scratch Projects

5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
1.1	มารู้จักโปรแกรม Scratch กันเถอะ	15
1.2	โปรเจกต์แรกของฉัน	20

5.2 ใบความรู้

- ใบความรู้ที่ 1.1 เรื่อง รู้จักโปรแกรม Scratch
- ใบความรู้ที่ 1.2 เรื่อง การสร้างภาพเคลื่อนไหวอย่างง่าย

6. วิธีดำเนินการ

6.1 การจัดเตรียม

6.1.1 เครื่องคอมพิวเตอร์พร้อมติดตั้งโปรแกรม Scratch

6.1.2 ใบความรู้ที่ 1.1-1.2 ตามจำนวนผู้เรียน

6.1.3 ใบงานที่ 1.1-1.2 ตามจำนวนผู้เรียน

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนชี้แจงข้อตกลงในการเรียน และการวัดและการประเมินผล

6.2.2 ผู้สอนอธิบายเกี่ยวกับที่มาของโปรแกรม Scratch และการนำไปสร้างโปรเจกต์ต่างๆ

6.2.3 อธิบายการเปิดโปรแกรม Scratch พร้อมกับสาธิตการเข้าใช้งานโปรแกรม

6.2.4 ผู้เรียนศึกษาใบความรู้ที่ 1.1 เรื่องรู้จักโปรแกรม Scratch และทำใบงานที่ 1.1 เรื่องมารู้จักโปรแกรม Scratch กันเถอะ

6.2.5 ผู้เรียนและผู้สอนร่วมเฉลยใบงานที่ 1.1 และสรุปสิ่งที่ได้จากการทำใบงานที่ 1.1

6.2.6 ผู้สอนกระตุ้นความคิดของผู้เรียนด้วยคำถาม “เราจะทำอย่างไรให้ตัวละครแมวเคลื่อนไหวที่ได้ และเดินอย่างเป็นธรรมชาติ” รอคำตอบจากผู้เรียน (คำตอบเช่น ทำให้แมวมีการขยับขา เปลี่ยนแปลงท่าเวลาเดิน) และพูดกระตุ้นให้ผู้เรียนค้นหาวิธีการโดยศึกษาใบความรู้ที่ 1.2 เรื่อง การสร้างภาพเคลื่อนไหวอย่างง่าย

- 6.2.7 ผู้เรียนศึกษาใบความรู้ที่ 1.2 และทำใบงาน ที่ 1.2 เรื่อง โปรเจกต์แรกของฉัน
- 6.6.8 ผู้สอนสุ่มผู้เรียนนำเสนอคำตอบในใบงานที่ 1.2 และผู้สอนเปิดโอกาสให้ผู้เรียนคนอื่นได้แสดงความคิดเห็น และเสนอแนะผลงานเพื่อนที่นำเสนอ
- 6.2.9 ผู้เรียนและผู้สอนร่วมกันสรุปคำสั่งและบล็อกที่เรียนในใบงานนี้

7. การวัดผลประเมินผล

- 7.1 ตรวจสอบคำตอบในใบงานและผลงาน
- 7.2 สังเกตการร่วมกิจกรรมในการระดมความคิดของผู้เรียน
- 7.3 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

- 8.1 <http://scratch.mit.edu/> (01/03/59)
- 8.2 <https://scratch.mit.edu/scratch2download/> (01/03/59)


9. ข้อเสนอแนะ

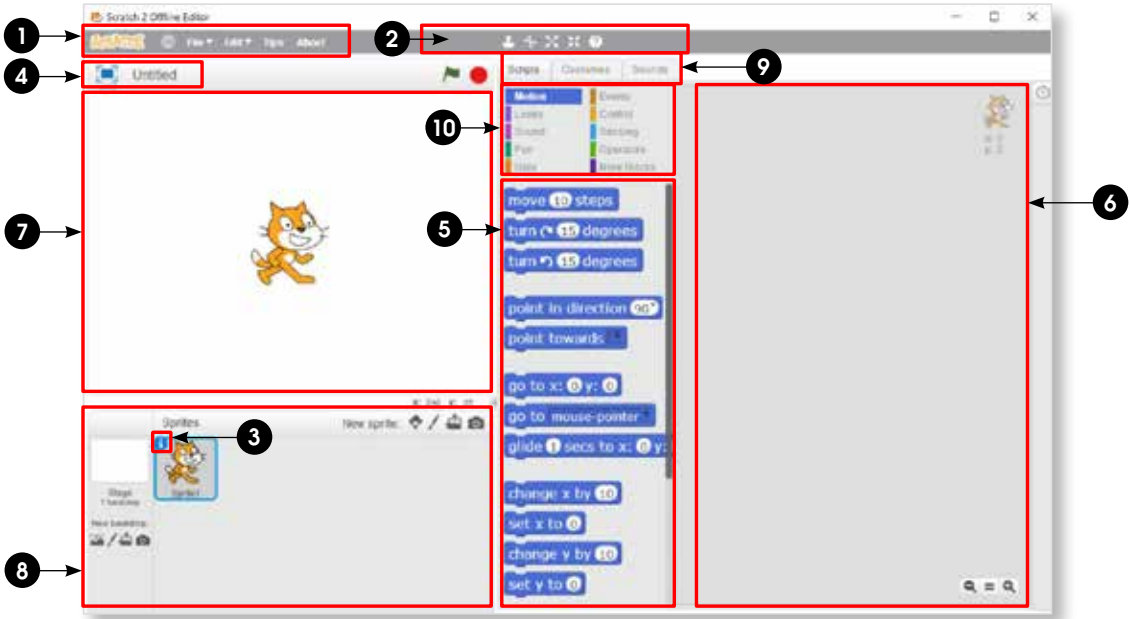
- 9.1 ถ้าโรงเรียนไม่มีเครื่องคอมพิวเตอร์เพียงพอสำหรับผู้เรียน 1 คน ต่อ 1 เครื่อง ผู้สอนสามารถจัดการเรียนการสอนโดยแบ่งผู้เรียนออกเป็นกลุ่ม ตามความเหมาะสม
- 9.2 ผู้สอนชี้แจงเรื่อง การบันทึกไฟล์งานกับผู้เรียนอย่างชัดเจน ในส่วนของโฟลเดอร์ที่ใช้จัดเก็บไฟล์ ตั้งชื่อให้สอดคล้องและอ้างอิงถึงงานของผู้เรียนแต่ละคนอย่างเป็นระบบ
- 9.3 ผู้สอนแนะนำให้ผู้เรียนสมัครสมาชิกของเว็บไซต์ <http://scratch.mit.edu/> เพื่อให้ผู้เรียนสามารถเผยแพร่และแลกเปลี่ยนผลงานผ่านเว็บไซต์




ใบงานที่ 1.1 มารู้จักโปรแกรม Scratch กันเถอะ

ชื่อ-สกุล เลขที่

- 1) ศึกษาใบความรู้ที่ 1.1 เรื่องรู้จักโปรแกรม Scratch 
- 2) เปิดโปรแกรม Scratch โดยดับเบิลคลิกที่เมนูลัดรูปแมวบนเดสก์ท็อป จะปรากฏหน้าต่างโปรแกรม ดังรูป ให้ใส่หมายเลข 1 ถึง 10 ในช่องว่างของแต่ละข้อที่มีรายละเอียดสัมพันธ์กัน

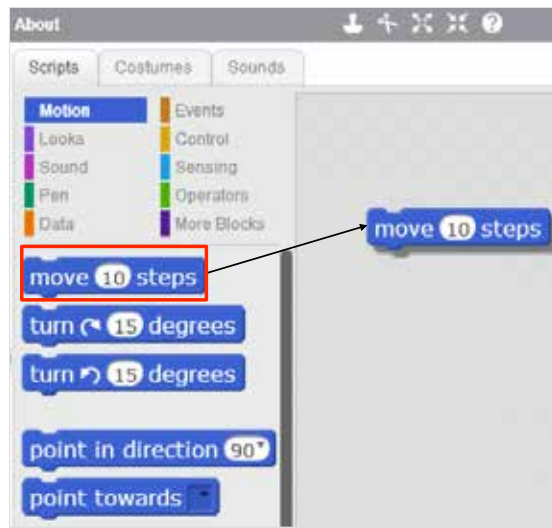


- 1) หมายเลข มีแถบเมนูที่ใช้เปลี่ยนภาษาของโปรแกรม Scratch ได้
- 2) การเขียนสคริปต์ให้ตัวละคร ต้องวางบล็อกที่หมายเลข
- 3) เมื่อคลิก  การแสดงผลจะปรากฏที่หมายเลข
- 4) หมายเลข ประกอบด้วยกลุ่มบล็อกต่างๆ ที่ใช้ในการเขียนสคริปต์
- 5) ถ้าคลิกที่กลุ่มบล็อก Looks จะแสดงบล็อกในกลุ่มที่หมายเลข
- 6) หมายเลข เป็นพื้นที่แสดงรายการตัวละคร และเวที
- 7) หมายเลข แสดงรายละเอียดข้อมูลของตัวละครที่เลือก
- 8) เครื่องมือที่ใช้ในการประทับตราตัวละครอยู่ที่หมายเลข
- 9) แถบเมนูแสดงข้อมูลสคริปต์ costumes และเสียงของตัวละครหรือเวที อยู่ที่หมายเลข
- 10) เมื่อกดหมายเลข พื้นที่การแสดงผลการทำงานของโปรแกรมจะมีขนาดเปลี่ยนแปลง

3 สร้างโปรเจกต์ใหม่ทุกครั้ง จะปรากฏตัวละครหลัก 1 ตัว ตัวละครหลักนั้นมีชื่อว่า

4 เขียนสคริปต์ให้ตัวละครแมวเดิน 10 หน่วย โดยทำตามขั้นตอน ดังนี้

4.1 คลิกปุ่มบล็อก Motion แล้วลากบล็อก **move 10 steps** ไปวางบนพื้นที่เขียนสคริปต์



4.2 คลิกปุ่มบล็อก Events แล้วลากบล็อก **when green flag clicked** มาวางต่อเชื่อมด้านบนของบล็อก **move 10 steps** ดังรูป



คลิก  ที่มุมขวาบนของเวที ปรากฏผลลัพธ์คือ

ทดลองเปลี่ยนแปลงค่าตัวเลขใน **move 10 steps** แล้วคลิก  สังเกตผลลัพธ์ที่เกิดขึ้น

5 ถ้าต้องการให้ตัวละครแมวเดินถอยหลังจะทำอย่างไร

6 บันทึกโปรเจกต์ชื่อ activity1_1

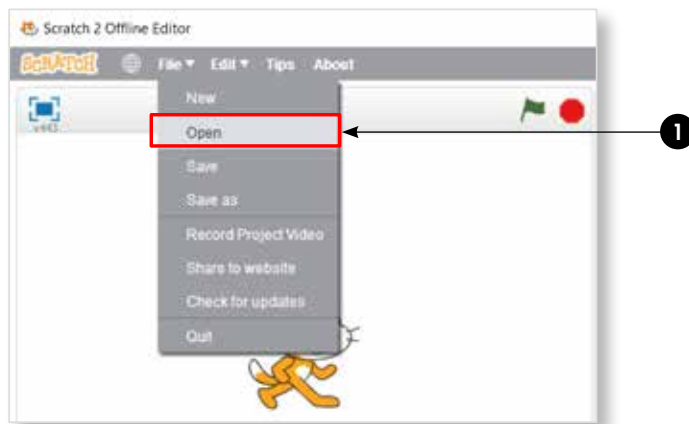
7 ปิดหน้าต่างโปรแกรม Scratch

ใบงานที่ 1.2 โปรเจกต์แรกของฉัน

ชื่อ-สกุล เลขที่

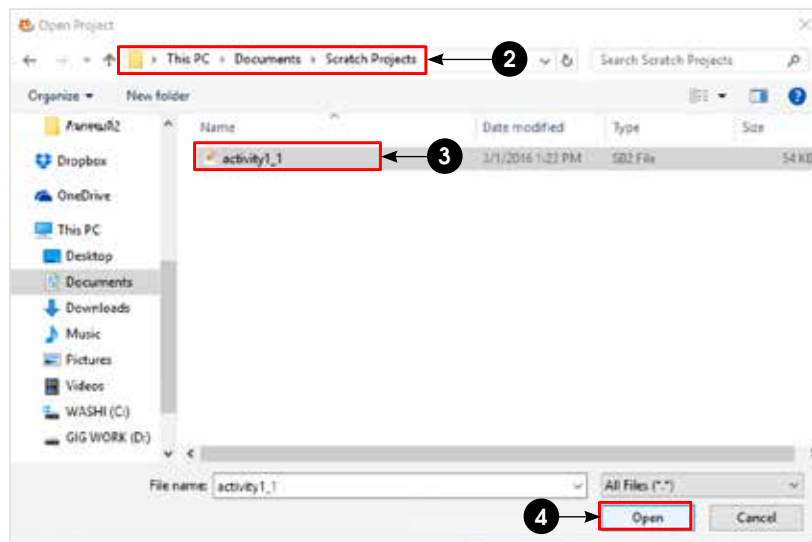
- 1 ศึกษาใบความรู้ที่ 1.2 เรื่อง การสร้างภาพเคลื่อนไหวอย่างง่าย
- 2 เปิดโปรเจกต์ชื่อ activity1_1 ที่ได้สร้างไว้จากใบงานที่ 1.1 ดังนี้

1 คลิกเมนู File -> Open



2 เปิดโฟลเดอร์ที่เก็บงาน ในที่นี้คือ โฟลเดอร์ Documents\Scratch Projects

3 เลือกไฟล์ activity1_1




4 คลิกปุ่ม


3) คลิกที่แท็บ Costumes ให้สังเกตเห็นตัวละคร Sprite1

3.1 ตัวละคร Sprite1 มีชุดตัวละคร ชุด

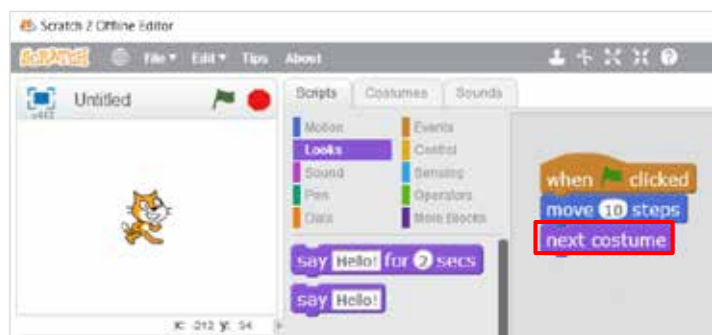
3.2 ชุดตัวละครแต่ละชุดมีชื่ออะไรบ้าง

1.
2.

 **Animation (แอนิเมชัน)**
 Animation หมายถึง การนำภาพนิ่งหลายๆ ภาพ มาแสดงต่อเนื่องกันทำให้เห็นเป็นภาพเคลื่อนไหว โดยมีการกำหนดช่วงเวลาของการแสดงภาพนิ่งแต่ละภาพ




3.3 คลิกกลุ่มบล็อก Looks เพิ่มบล็อก **next costume** คลิก  แล้วสังเกตและบันทึกผลลัพธ์ผลลัพธ์คือ




4) ถ้าต้องการให้แมวเคลื่อนที่ต่อเนื่องไปเรื่อยๆ ทำได้โดยเขียนสคริปต์ให้ตัวละครแมวดังนี้


1) คลิกที่กลุ่มบล็อก Control




2) ลากบล็อก  กลุ่มบล็อก 



สังเกตการแทรกบล็อก เมื่อมีการลากบล็อก มาแทรกระหว่างบล็อก when clicked กับ move 10 steps จะปรากฏแถบสีขาว

4.1 คลิก  สังเกตผลลัพธ์ที่เกิดขึ้น คือ

4.2 คลิก  สังเกตผลลัพธ์ที่เกิดขึ้น คือ

5) คลิกกลุ่มบล็อก Control เพิ่มบล็อก  คลิก  แล้วสังเกตและบันทึกผลลัพธ์



ผลลัพธ์

.....

.....


6) ทดลองเปลี่ยนตัวเลขในบล็อก 

6.1 เปลี่ยนตัวเลขที่มีค่ามากกว่า 1 เช่น , 3 ผลลัพธ์ที่ได้

6.2 เปลี่ยนตัวเลขที่มีค่าน้อยกว่า 1 เช่น 0.1, 0.5 ผลลัพธ์ที่ได้

6.3 เปลี่ยนตัวเลขที่มีค่าติดลบ เช่น -1, -2 ผลลัพธ์ที่ได้



แยแล้ว!!! แมวเหลือแต่หาง ...ไม่ต้องตกใจ...
 วิธีที่ 1 ให้คลิก  เพื่อหยุดการทำงานของสคริปต์ แล้วดึงตัวละครมาไว้กลางเวทีเหมือนเดิม
 วิธีที่ 2 นำบล็อก  มาวางบนพื้นที่เขียนสคริปต์ แล้วดับเบิลคลิกที่บล็อกนั้น เพียงเท่านั้น แมวก็จะกลับมาที่เดิม

7) คลิกกลุ่มบล็อก Motion เพิ่มบล็อก  คลิก  แล้วสังเกตและบันทึกผลลัพธ์

.....







8 ทดลองเปลี่ยนทิศทางการหมุนของตัวละคร ดังนี้

8.1 คลิก  ใน






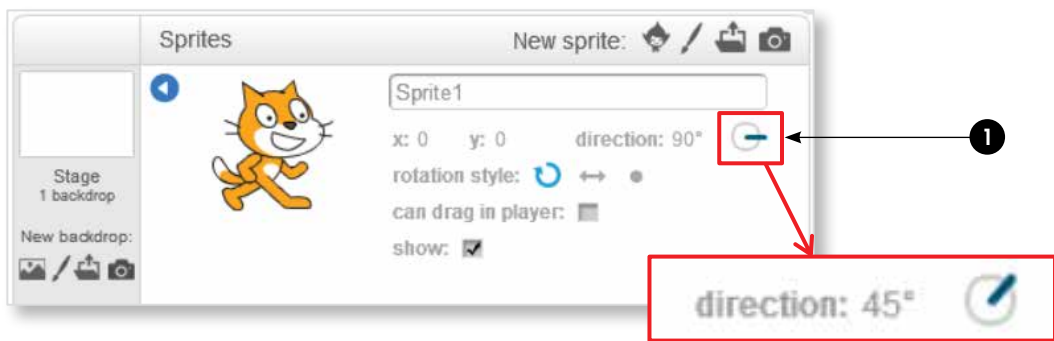
จะปรากฏหน้าต่างรายละเอียดของตัวละคร แล้วสังเกต rotation style



8.2 หัวข้อ rotation style เป็นรูปแบบการหมุนที่มี 3 ลักษณะ (  ) จากสคริปต์ในข้อ 7 ให้ทดลองเปลี่ยนรูปแบบการหมุน จากนั้นคลิก  สังเกตผลลัพธ์ แล้ววงกลมล้อมรอบรูปตัวละครที่เป็นผลลัพธ์หลังชนขอบเวที

รูปแบบการหมุน	หลังจากชนขอบด้านขวา	หลังจากชนขอบด้านซ้าย
8.1 ปุ่ม 		
8.2 ปุ่ม 		
8.3 ปุ่ม 		

- 9) คลิกที่  เพื่อหยุดการทำงาน เลือกการหมุนแบบ  แล้วหมุนเส้นสีน้ำเงิน **1** ที่ปรากฏที่ข้อมูลของตัวละคร (Sprite1) ให้ direction เท่ากับ 45 จากนั้นคลิก  สังเกตผลลัพธ์



- 10) บันทึกโปรเจกต์ชื่อ activity1_2

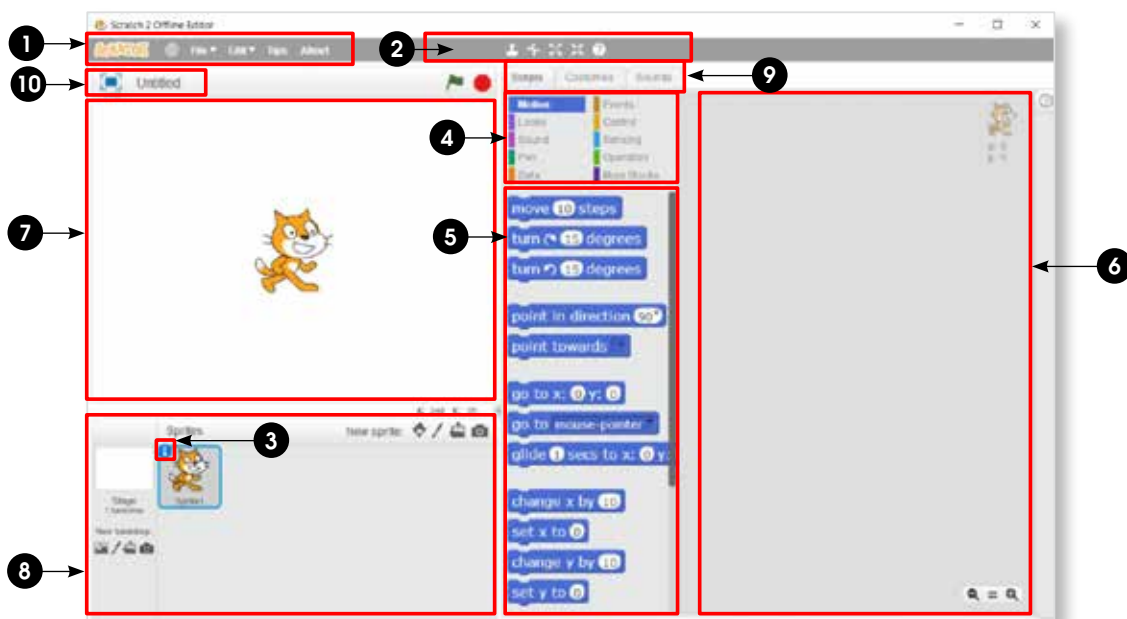


ใบความรู้ที่ 1.1 รู้จักโปรแกรม Scratch


โปรแกรม **Scratch** (อ่านว่า สะ - แครช) เป็นโปรแกรมภาษาคอมพิวเตอร์ ที่ผู้เรียนสามารถสร้างชิ้นงานได้อย่างง่าย เช่น นิทานที่สามารถโต้ตอบกับผู้อ่านได้ ภาพเคลื่อนไหว เกม ดนตรี และศิลปะ และเมื่อสร้างเป็นชิ้นงานเสร็จแล้ว สามารถนำชิ้นงานที่สร้างสรรค์นี้ แสดง และแลกเปลี่ยนความคิดเห็นร่วมกับผู้อื่นบนเว็บไซต์ได้ ทำให้ผู้เรียนได้เรียนรู้หลักการและแนวความคิดการโปรแกรมไปพร้อมๆ กับการคิดอย่างสร้างสรรค์ มีเหตุผล และเป็นระบบ


1. ส่วนประกอบหลักของโปรแกรม


หน้าต่างการทำงานของโปรแกรม Scratch มีส่วนประกอบหลักดังนี้





1 แถบเมนูเครื่องมือ (Toolbar) File Edit Tips About

 ปุ่มเปลี่ยนภาษา

 File เมนู เปิด-บันทึกโปรเจกต์ บันทึกวีดิโอ แชรเว็บไซต์ เช็คอัปเดต และปิดโปรแกรม

 Edit เมนู แก้ไขการตั้งค่า

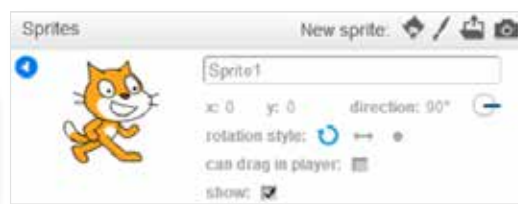
 Tips เมนู แนะนำโปรแกรมตัวอย่าง

 About เมนู เว็บไซต์ <http://scratch.mit.edu/>

2 เครื่องมือเวที (Stage Toolbar)

- ปุ่มปรับทิศทางตัวละคร
- ปุ่มลบตัวละคร
- ปุ่มเพิ่มขนาดตัวละคร
- ปุ่มลดขนาดตัวละคร
- ปุ่มแสดงความช่วยเหลือ

3 ข้อมูลของตัวละครที่ถูกเลือก

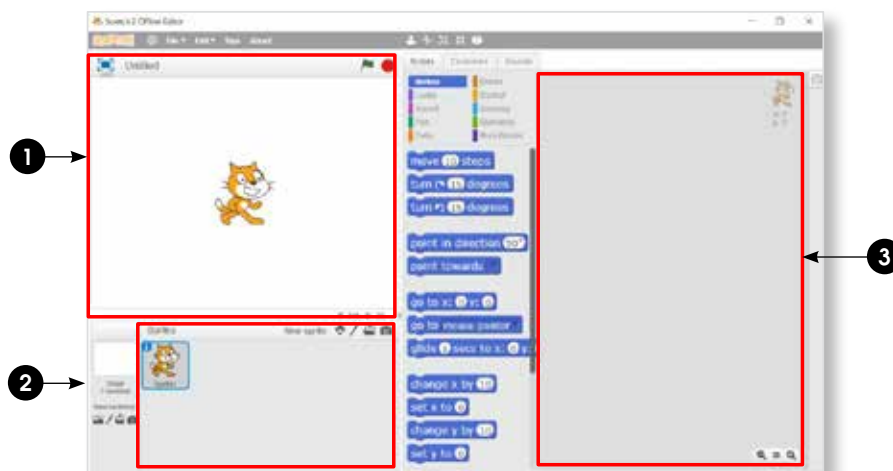


- 4** กลุ่มบล็อก
- 5** บล็อกในกลุ่มที่เลือก
- 6** พื้นที่ทำงาน
- 7** เวที (Stage)
- 8** รายการตัวละคร และเวทีที่ใช้ในโปรเจกต์ปัจจุบัน
- 9** แถบเมนูแสดงข้อมูลสคริปต์ costumes และเสียงของตัวละครหรือเวที
- 10** พื้นที่การแสดงผลของการทำงานของโปรแกรมที่มีขนาดที่เปลี่ยนแปลง

2. รู้จักโปรเจกต์

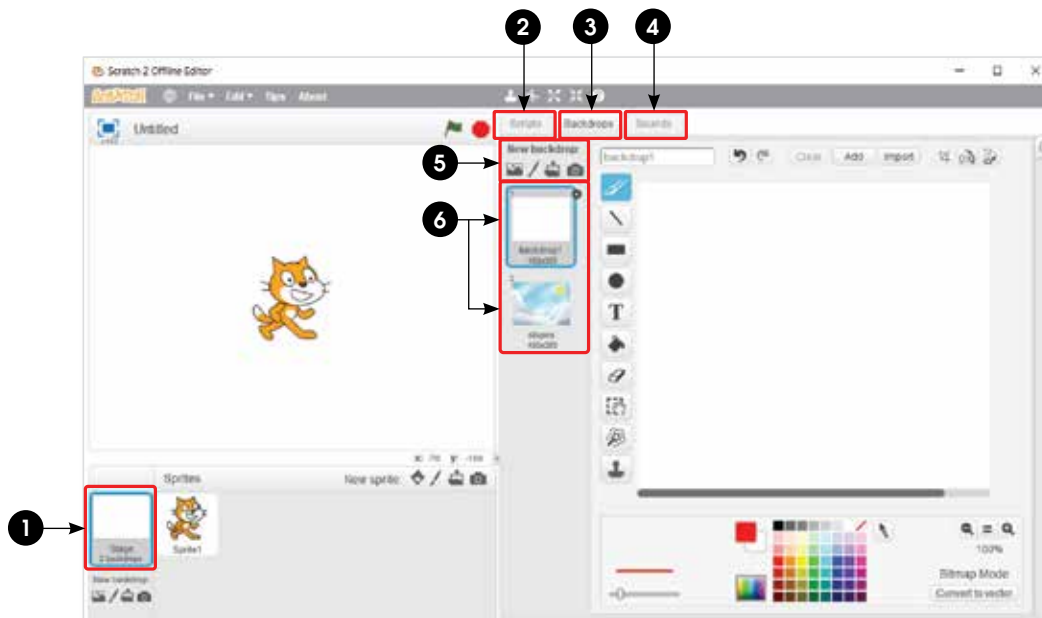
โปรเจกต์ใน Scratch มีโครงสร้าง 3 ส่วนประกอบด้วย

- 1** เวที (Stage)
- 2** ตัวละคร (Sprite)
- 3** สคริปต์ (Script)



2.1 เวที

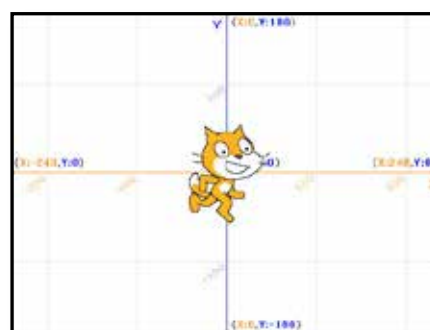
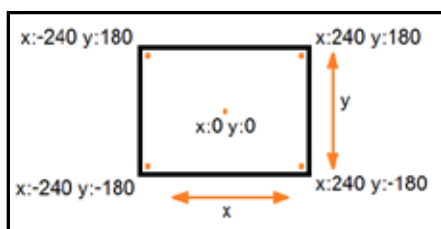
เวทีมีขนาดกว้าง 480 หน่วย สูง 360 หน่วย ในแต่ละโปรเจกต์มีเวทีเดียว จึงมีชื่อเดียว และไม่สามารถเปลี่ยนชื่อได้ เวทีใช้แสดงผลการทำงานของสคริปต์ (script) เสียง (sound) หรือฉาก (backdrop) ได้



รายละเอียดของเวที

1. คลิก Stage เพื่อดูรายละเอียดของเวที
2. แท็บ Scripts ของเวที
3. แท็บ Backdrops
4. แท็บ Sounds
5. เพิ่มฉากใหม่ New Backdrop
6. ฉากทั้งหมดที่มีอยู่บนเวที

การบอกตำแหน่งใดๆ บนเวทีจะบอกโดยใช้ค่า (x, y) เช่น ตำแหน่งกลางเวที จะมีค่า (x, y) เป็น (0, 0)

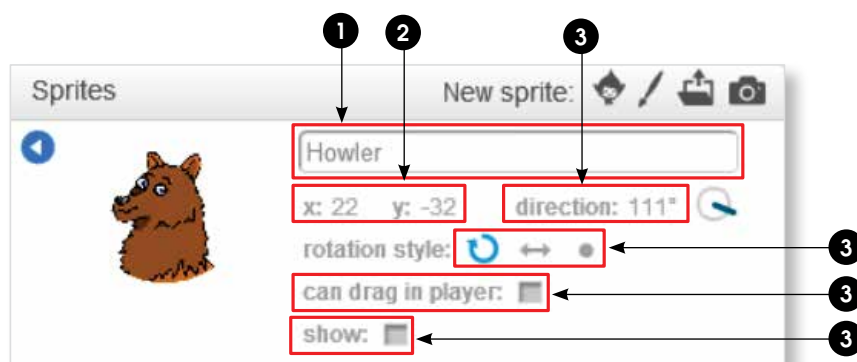


2.2 ตัวละคร

ตัวละครแต่ละตัวจะมีข้อมูลแตกต่างกัน โดยสามารถคลิก ⓘ ที่ตัวละคร เพื่อดูข้อมูลของตัวละครนั้น เช่น ตัวละครหมาป่ามีข้อมูลดังตาราง

2.2.1 ชื่อตัวละคร


โปรแกรมจะตั้งชื่อตัวละครให้เป็น Sprite1, Sprite2, Sprite3 ตามลำดับที่สร้างขึ้น โดยอัตโนมัติ ถ้าต้องการเปลี่ยนชื่อตัวละครให้พิมพ์ชื่อใหม่บนแถบชื่อหมายเลข ❶



หมายเลข	ข้อมูล	รายละเอียดข้อมูลตัวละคร
❶	ชื่อตัวละคร	Howler
❷	ตำแหน่งบนเวที	x: 22 y: -32
❸	ทิศทางการเคลื่อนที่ (direction)	111 องศา
❹	รูปแบบการหมุนของตัวละคร มี 3 ลักษณะ 🔄 หมุนได้รอบทิศทาง ↔ หันได้เฉพาะซ้ายหรือขวา ● ห้ามหมุน	🔄 หมุนได้รอบทิศทาง
❺	<input checked="" type="checkbox"/> can drag in player: การลากตัวละครในโหมดนำเสนอ หรือบนเว็บไซต์	<input type="checkbox"/> can drag in player: ลากตัวละครไม่ได้
❻	<input checked="" type="checkbox"/> show: การแสดงตัวละครบนเวที ในขณะที่ออกแบบ	<input type="checkbox"/> show: ตัวละครบนเวทีไม่ปรากฏ ในขณะที่ออกแบบ

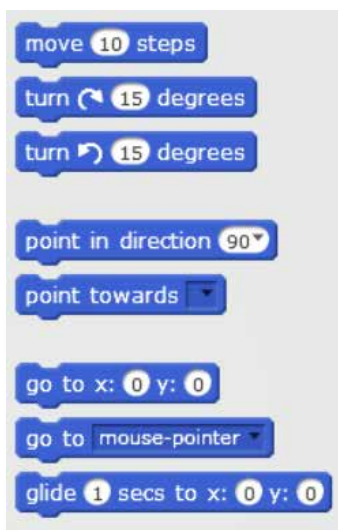
2.3 สคริปต์

สคริปต์ คือ ชุดคำสั่งสำหรับตัวละครหรือเวที เพื่อสั่งให้ตัวละครหรือเวทีทำงานตามวัตถุประสงค์ที่ต้องการ โดยการเลือกสคริปต์จากกลุ่มบล็อก ซึ่งแบ่งเป็น 8 กลุ่ม ดังนี้

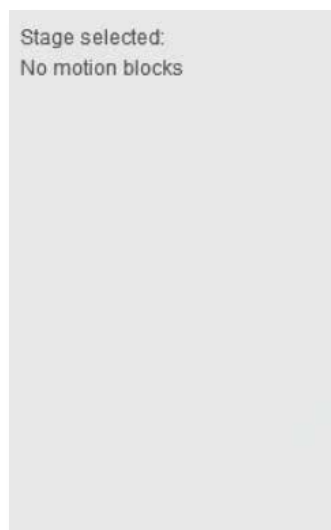
กลุ่มบล็อก	การทำงาน
Motion	การเคลื่อนไหว เช่น เคลื่อนที่ไปข้างหน้า หมุนไปทางซ้ายหรือขวา
Control	การควบคุม เช่น การวนซ้ำ การตรวจสอบเงื่อนไข
Looks	การแสดง เช่น พูด คิด เปลี่ยนขนาด
Sensing	การรับรู้ เช่น การสัมผัส คลิกเมาส์ จับเวลา
Sound	การแสดงเสียง เช่น เสียงสัตว์ กลอง ไม้ตดนตรี
Operators	ตัวดำเนินการ เช่น บวก และ หรือ
Pen	ปากกา เช่น ยกปากกาขึ้น ตั้งค่าสีปากกา
Data	ตัวแปร เช่น สร้างค่าตัวแปร
Events	เหตุการณ์ เช่น เมื่อคลิก 
More Blocks	บล็อกอื่นๆ เช่น การสร้างฟังก์ชัน

บล็อกของตัวละครกับบล็อกของเวที ที่อยู่ภายใต้กลุ่มบล็อกเดียวกัน จะมีความแตกต่างกัน เช่น กลุ่มบล็อก Motion ของตัวละครจะมีบล็อกดังรูปด้านซ้าย ส่วนรูปด้านขวาเป็นของเวทีซึ่งไม่มีบล็อก Motion เนื่องจากเวทีเคลื่อนที่ไม่ได้

บล็อก Motion สำหรับตัวละคร



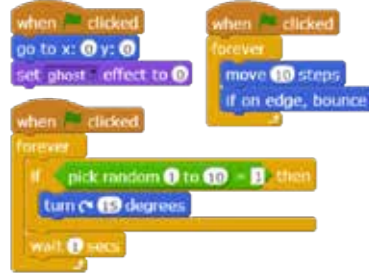
ไม่มีบล็อก Motion สำหรับเวที





สคริปต์หนึ่งๆ ประกอบไปด้วยบล็อกมาเรียงต่อกันเป็นกลุ่ม บางบล็อกสามารถซ้อนอยู่บนบล็อกอื่นได้

ตัวอย่างการเขียนสคริปต์สั่งให้ Dinodane เคลื่อนที่ ก่อนเขียนสคริปต์ต้องเลือกตัวละคร แล้วนำบล็อกที่ต้องการมาเรียงต่อกันในพื้นที่สำหรับเขียนสคริปต์



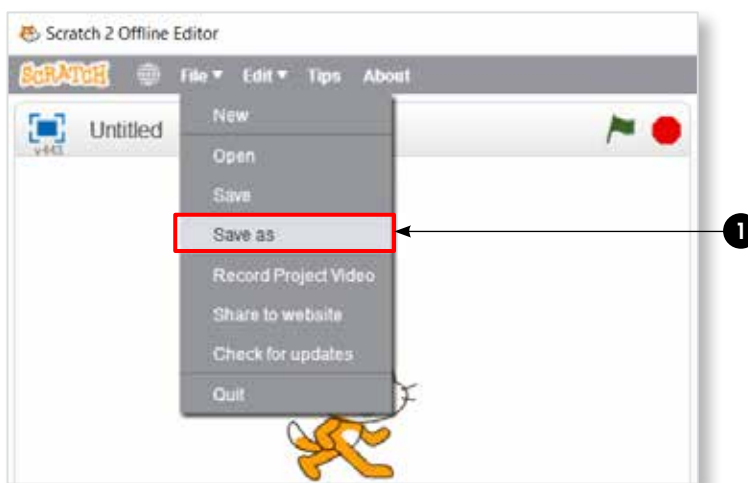
3. การสั่งให้โปรเจกต์เริ่มทำงานและหยุดทำงาน
 การสั่งให้โปรเจกต์เริ่มทำงาน ทำได้โดยใช้บล็อกในกลุ่ม
 บล็อก Events ที่มีเหตุการณ์ต่างๆ เช่น

บล็อก	คำอธิบาย
	เมื่อคลิก บล็อกคำสั่งต่างๆ ที่อยู่ภายใต้บล็อก ก็จะเริ่มทำงาน
	เมื่อกดแป้น space bar บล็อกคำสั่งต่างๆ ที่อยู่ภายใต้บล็อก ก็จะเริ่มทำงาน (สามารถเปลี่ยนเป็นตัวอักษรใดๆ บนคีย์บอร์ดได้)
	เมื่อคลิกที่ตัวละคร บล็อกคำสั่งต่างๆ ที่อยู่ภายใต้บล็อก ก็จะเริ่มทำงาน
	ถ้าฉากปัจจุบันชื่อ backdrop1 บล็อกคำสั่งต่างๆ ที่อยู่ภายใต้บล็อก ก็จะเริ่มทำงาน

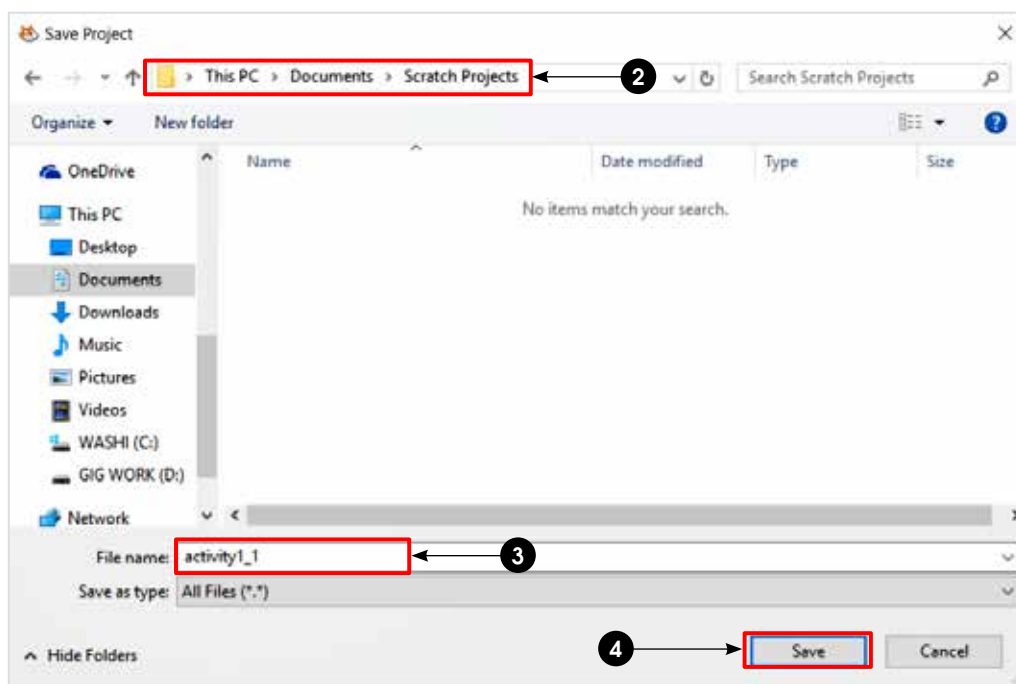
กรณีที่ใช้คำสั่ง Forever ถ้าต้องการหยุดการทำงานทั้งโปรเจกต์ ให้คลิก

4. การบันทึกโปรเจกต์

1. คลิกเมนู File -> Save หรือ Save As



2. เลือกโฟลเดอร์ที่ต้องการบันทึก (เลือกโฟลเดอร์ Documents\Scratch Projects)
3. พิมพ์ชื่อโปรเจกต์ (ตัวอย่าง บันทึกชื่อ activity1_1)



4. คลิกปุ่ม โปรเจกต์จะถูกบันทึกไว้ที่โฟลเดอร์ Documents\Scratch Projects



ใบความรู้ที่ 1.2 การสร้างภาพเคลื่อนไหวอย่างง่าย

โปรเจกต์ในโปรแกรม Scratch สามารถสร้างตัวละครได้หลายตัว ซึ่งตัวละครแต่ละตัว จะมีชุดตัวละครอย่างน้อยหนึ่งชุด และสามารถเพิ่มเติมได้ การทำให้ตัวละครเคลื่อนไหวจะใช้วิธีเปลี่ยนสลับชุดตัวละครไปมาอย่างรวดเร็ว พร้อมกับสั่งให้ตัวละครเคลื่อนที่ในลักษณะใดลักษณะหนึ่ง โดยมีการเขียนสคริปต์จากกลุ่มบล็อก ดังต่อไปนี้

1. กลุ่มบล็อก Motion

บล็อกที่ใช้ในตัวละคร	ความหมาย
	ไปข้างหน้า หรือถอยหลัง ตัวอย่าง ไปข้างหน้า 10 หน่วย
	ถ้าตัวละครชนขอบให้สะท้อนกลับ

2. กลุ่มบล็อก Looks

บล็อกที่ใช้ในตัวละคร	ความหมาย
	เปลี่ยนชุดตัวละครเป็นชุดถัดไปที่มีอยู่ในรายการ

3. กลุ่มบล็อก Control

บล็อกที่ใช้ในตัวละคร	ความหมาย
	การทำงานซ้ำไม่รู้จบ
	หยุดรอเป็นเวลา 1 วินาที



กิจกรรมที่ 2 การสร้างภาพเคลื่อนไหว

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม

2. สารการเรียนรู้

- 2.1 การเพิ่มและลบฉาก
- 2.2 การบอกตำแหน่งใดๆ บนเวที และรูปแบบการหมุนตัวละคร
- 2.3 การย้ายตัวละครไปยังตำแหน่งที่ระบุ
- 2.4 การเพิ่มตัวละครใหม่ และส่งออกชุดตัวละครออกไปเป็นไฟล์
- 2.5 การลบฉากและตัวละคร

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 เพิ่มและสร้างตัวละครใหม่ Paint new sprite และ Save to local file ตัวละครออกไปเป็นไฟล์
- 3.2 เพิ่มและลบฉากด้วยบล็อก next backdrop และ switch backdrop to ในกลุ่มบล็อก Looks
- 3.3 ลบตัวละครและฉาก
- 3.5 ระบุตำแหน่งด้วยค่า (x, y) และรูปแบบการหมุนของตัวละคร

4. แนวคิด

ตัวละครในโปรเจกต์มีได้หลายตัว โดยตัวละครแต่ละตัวจะมีสคริปต์การทำงานที่เป็นของตนเอง การเพิ่มตัวละครอาจทำได้ทั้งการนำเข้าจากไฟล์ตัวละครที่มีอยู่แล้ว หรือวาดขึ้นใหม่เอง นอกจากนี้ยังสามารถส่งออกตัวละครที่มีอยู่ในโปรเจกต์ออกไปเป็นไฟล์ เพื่อการนำไปใช้งานต่อไปได้ด้วย

การเคลื่อนที่ของตัวละคร จะใช้ความรู้เกี่ยวกับตำแหน่งหรือพิกัดบนพื้นที่แสดงผลการทำงาน ส่วนการเคลื่อนที่ไปในทิศทางต่างๆ จะต้องกำหนดให้ตัวละครหันหน้าไปยังทิศทางที่ต้องการก่อนแล้วจึงเคลื่อนที่ ซึ่งอาจใช้บล็อกคำสั่ง turn หรือ point in direction

เวทีจะมีได้เพียงหนึ่งเวที แต่สามารถเปลี่ยนภาพฉากของเวทีได้หลายภาพ การเปลี่ยนชุดตัวละคร รวมไปถึงการเปลี่ยนภาพฉากนี้ สามารถเขียนสคริปต์ให้ตัวละครหรือฉากแยกส่วนกัน เพื่อสั่งงานให้เกิดเหตุการณ์ที่ต่างกัน แต่แสดงผลการทำงานไปพร้อมๆ กัน

5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
2.1	เพิ่มและลบฉาก	5
2.2	พิกัดและทิศทาง	15

5.2 ใบความรู้

- ใบความรู้ที่ 2.1 เรื่อง ฉากและตัวละคร
- ใบความรู้ที่ 2.2 เรื่อง พิกัดและทิศทาง

6. วิธีดำเนินการ

6.1 การจัดเตรียม

6.1.1 ใบงานที่ 2.1- 2.2 ตามจำนวนผู้เรียน

6.1.2 ใบความรู้ที่ 2.1- 2.2 ตามจำนวนผู้เรียน

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนตั้งคำถามให้ผู้เรียนคิดว่า “การที่ตัวละครเคลื่อนไหวและเปลี่ยนทิศทางเกิดขึ้นได้อย่างไร ผู้เรียนสังเกตเห็นอะไรเปลี่ยนแปลงอีกบ้าง” และพุดกระตุ้นผู้เรียนว่าผู้เรียนสามารถสร้างชิ้นงานเหมือนตัวอย่างได้ด้วยโปรแกรม Scratch

6.2.2 ผู้เรียนศึกษาใบความรู้ที่ 2.1 เรื่องฉากและตัวละคร และทำใบงานที่ 2.1 เรื่อง เพิ่มและลบฉาก

6.2.3 ผู้สอนและผู้เรียนร่วมกันเฉลยคำตอบในใบงานที่ 2.1

6.2.4 ผู้เรียนศึกษาใบความรู้ที่ 2.2 ทำใบงานที่ 2.2 เรื่อง พิกัดและทิศทาง

6.2.5 ผู้สอนสุ่มผู้เรียนนำเสนอคำตอบในใบงานที่ 2.2 โดยสุ่มผู้เรียนเพื่อนำเสนอคำตอบแต่ละข้อไม่ให้ซ้ำคนเดิมจนครบทุกข้อ

6.2.6 ผู้เรียนและผู้สอนร่วมกันสรุปความรู้เรื่องการเพิ่มลบ ตัวละคร การกำหนดพิกัดและทิศทาง

7. การวัดผลประเมินผล

- 7.1 ตรวจสอบใบงานและผลงาน
- 7.2 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

- 8.1 <http://scratch.mit.edu/> (01/03/59)
- 8.2 <https://scratch.mit.edu/scratch2download/> (01/03/59)

9. ข้อเสนอแนะ

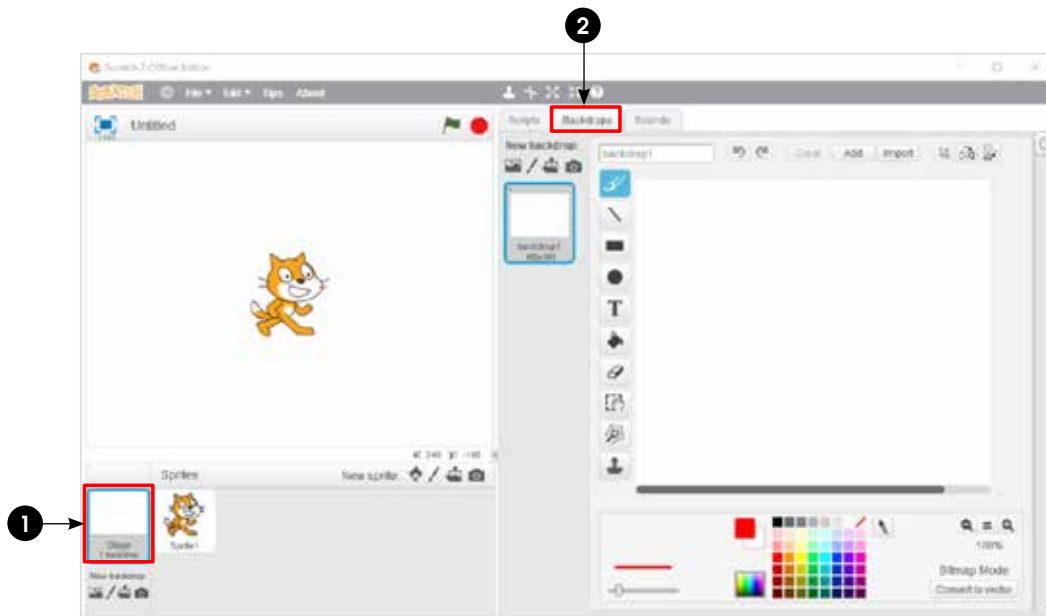
- 9.1 ให้ผู้เรียนสังเกตชิ้นงานตัวอย่างที่สร้างด้วยโปรแกรม Scratch ช่วยให้ผู้เรียนสามารถสร้างชิ้นงานเลียนแบบตัวอย่างได้




ใบงานที่ 2.1 เพิ่มและสลับฉาก

ชื่อ-สกุล เลขที่

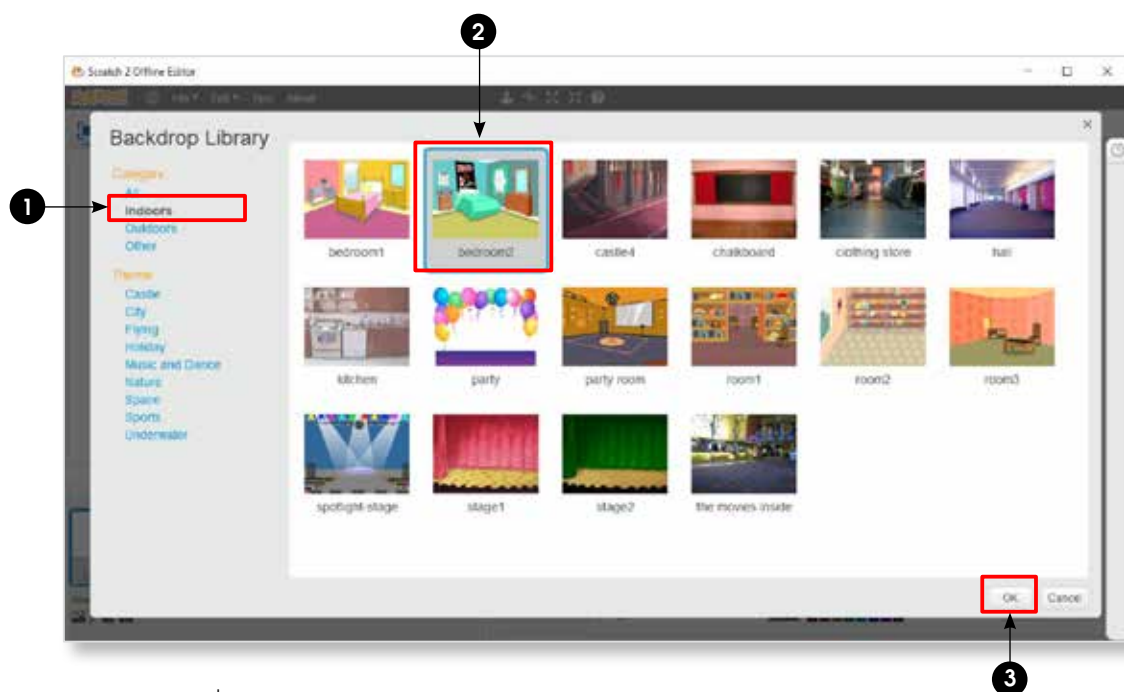
- 1 สร้างโปรเจกต์ใหม่
- 2 เพิ่มฉากของเวที ดังนี้
 - 1 คลิกที่เวที (Stage)
 - 2 คลิกแท็บฉาก (Backdrops)



การเลือกฉากจากไฟล์ภาพ

- 3 คลิกไอคอน  (Choose backdrop from library) จะปรากฏหน้าต่างต่าง Backdrop Library ซึ่งมี Theme ภาพกลุ่มต่างๆ ในที่นี้ให้เลือกภาพ bedroom2 จาก Category : Indoors ดังนี้

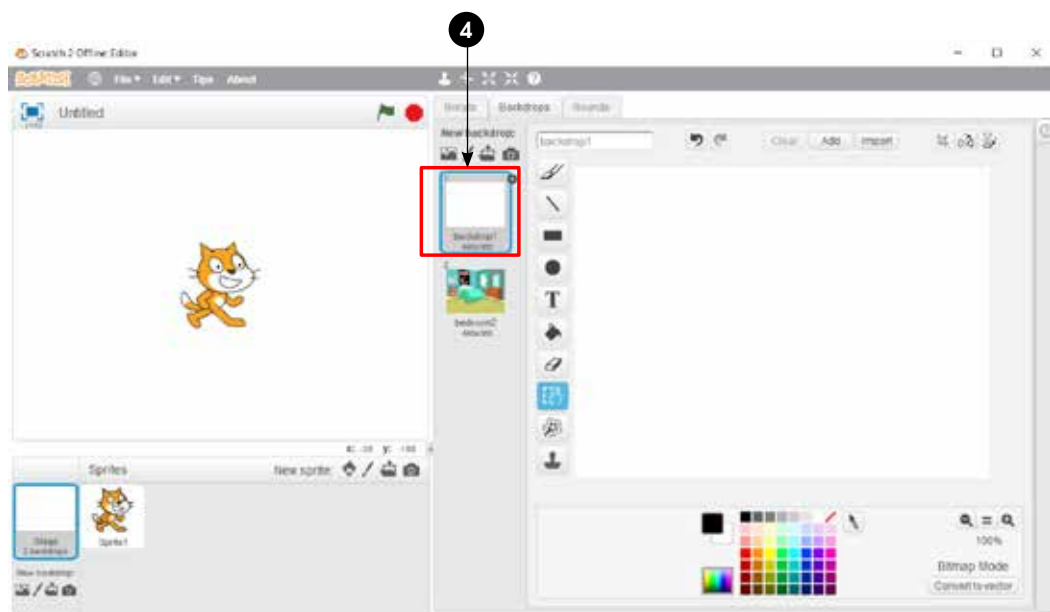
1 คลิกที่ Indoors จะปรากฏภาพต่างๆ ที่เป็นภาพภายในห้อง



2 คลิกที่ภาพ bedroom2

3 คลิกปุ่ม ฉากที่เลือกจะปรากฏบนเวที

4 ลบฉาก backdrop1 โดยคลิกปุ่ม สิ่งปรากฏคือ



4 เพิ่มฉาก blue sky จาก Category : Outdoors ด้วยวิธีการเช่นเดียวกับข้อ 3 ที่ผ่านมา

5 เขียนสคริปต์ให้กับเวที ดังนี้

```
when clicked
wait 5 secs
next backdrop
```

ผลลัพธ์ที่ได้ คือ

เพิ่มสคริปต์ให้กับเวที ดังนี้

```
when clicked
forever
wait 3 secs
next backdrop
```

ผลลัพธ์ที่ได้ คือ

6 เพิ่มฉากให้กับเวที อีก 2 ฉาก แล้วเขียนสคริปต์ให้เวทีแสดงเฉพาะฉากที่ 1 และฉากที่ 4 สคริปต์มีดังนี้

.....

.....

.....

.....

.....

7 บันทึกเป็นโปรเจกต์ใหม่ ชื่อ activity2_1

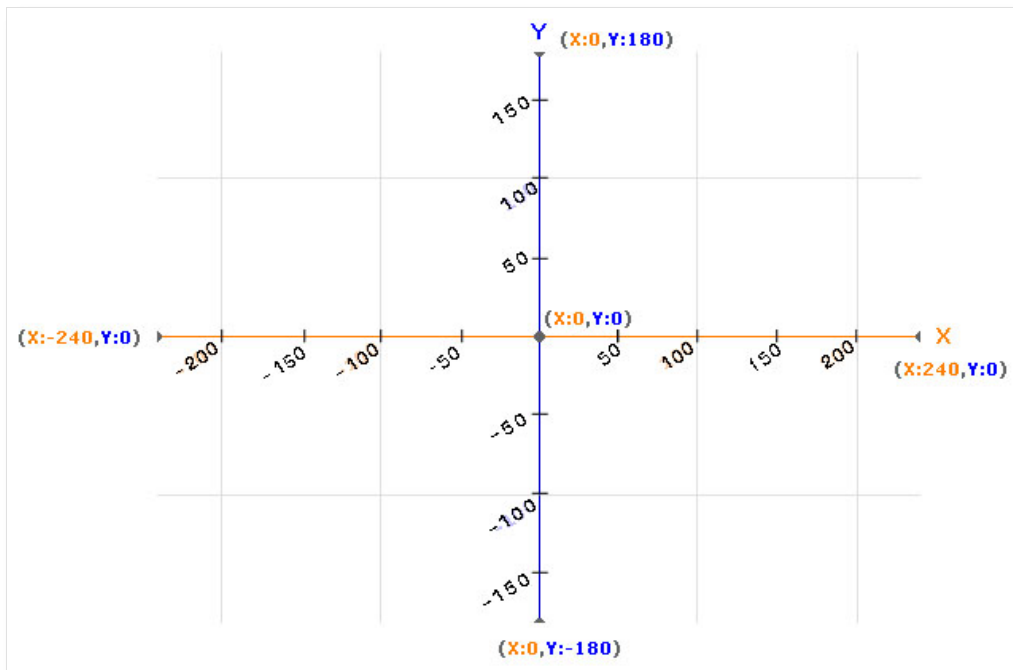
ใบงานที่ 2.2 พิกัดและทิศทาง

ชื่อ-สกุล เลขที่

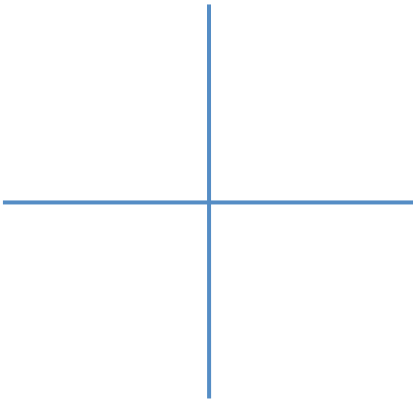
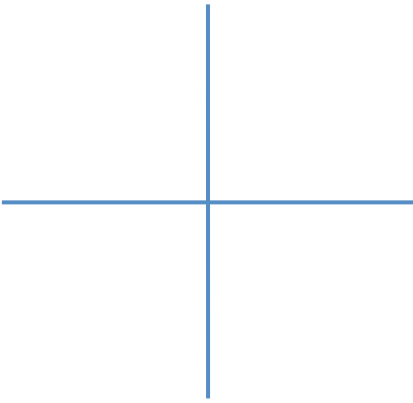
- 1 ศึกษาใบความรู้ที่ 2.2 เรื่อง พิกัดและทิศทาง
- 2 สร้างโปรเจกต์ใหม่ โดยให้มีตัวละครอยู่ในตำแหน่งต่อไปนี้

ตัวละคร		ตำแหน่ง
Duck		$x = -120$ และ $y = 100$
Fox		$x = 200$ และ $y = 60$
Lion		$x = 120$ และ $y = -140$
Mouse1		$x = -200$ และ $y = -130$

- 3 จากโจทย์ข้อ 2 ให้ทำเครื่องหมาย × ระบุตำแหน่งพิกัดและทิศทาง




4 วัดเส้นทิศทางของตัวละคร ตามที่ระบุลงในช่องผลลัพธ์

ทิศทาง	ผลลัพธ์
4.1 -45 องศา	
4.2 75 องศา	



การกำหนดทิศทาง

- 0 องศาจะมีทิศทางการเคลื่อนที่ไปด้านบน
- 90 องศาจะมีทิศทางการเคลื่อนที่ไปทางขวา
- 90 องศาจะมีทิศทางการเคลื่อนที่ไปทางซ้าย

- 5) เพิ่มสคริปต์ต่อไปนี้ให้กับตัวละคร Mouse1 สังเกตผลที่ได้ เมื่อคลิก 

```

when clicked
go to x: -200 y: -70
forever
  wait 1 secs
  move 10 steps
  turn 45 degrees
  
```

- 6) เพิ่มสคริปต์ให้กับตัวละคร Duck สังเกตผลที่ได้ เมื่อคลิก 

```

when clicked
go to x: -120 y: 100
forever
  wait 1 secs
  move 10 steps
  point in direction 90°
  
```

การใช้คำสั่ง  และ  แตกต่างกันอย่างไร

.....

.....

- 7) บันทึกเป็นไฟล์ใหม่ ชื่อ activity2_2

ใบความรู้ที่ 2.1 ฉากและตัวละคร


การสร้างโปรเจกต์เพื่อให้มีการเปลี่ยนภาพฉาก (Backdrops) ของเวที (Stage) ได้หลายภาพ สามารถเขียนสคริปต์ให้ตัวละครหรือฉากแยกส่วนกัน เพื่อสั่งงานให้เกิดเหตุการณ์ที่ต่างกัน แต่แสดงผลการทำงานไปพร้อมๆ กัน ซึ่งบล็อกที่ใช้มีดังต่อไปนี้

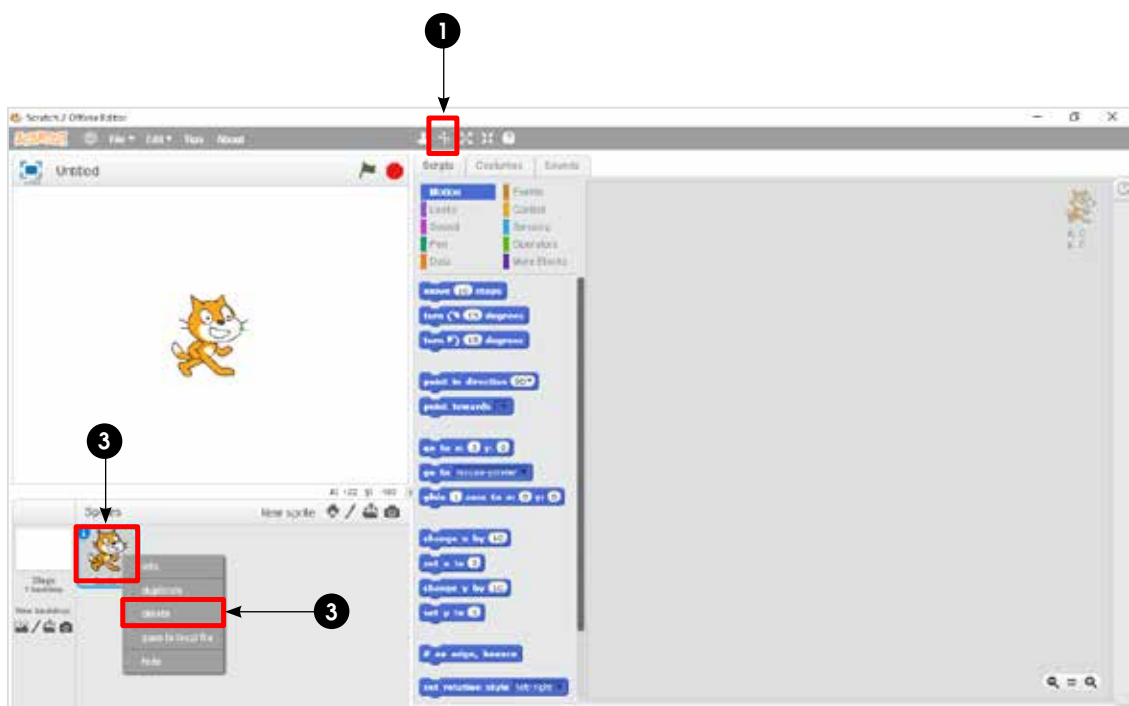
บล็อกในกลุ่มบล็อก **Looks** ของเวที เพื่อสลับฉาก

switch backdrop to backdrop1	สลับฉาก ตัวอย่าง สลับฉากเป็นฉาก backdrop1
next backdrop	สลับฉากเป็นฉากถัดไปที่มีอยู่ในรายการ

การเพิ่ม ลบตัวละคร

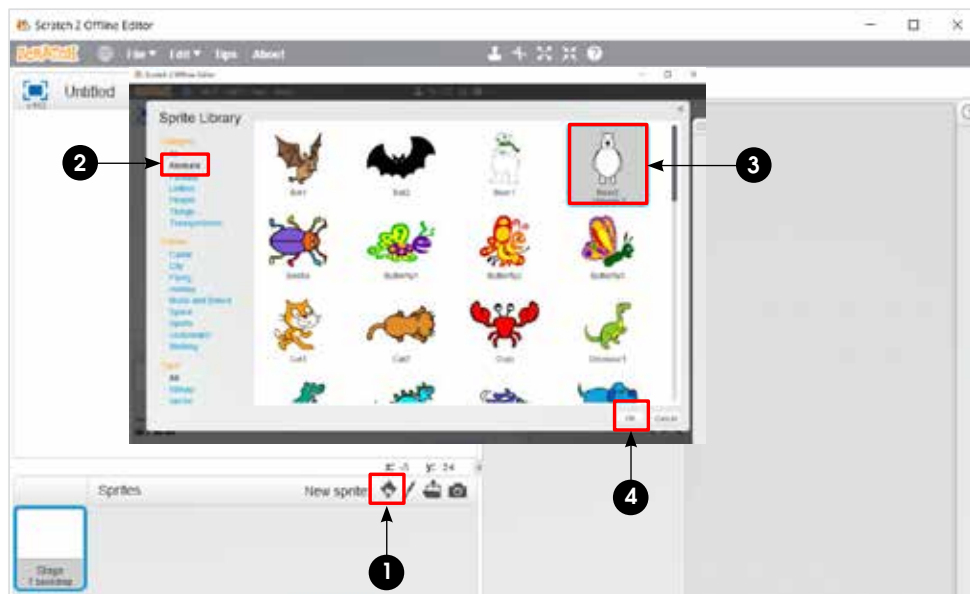
1. การลบตัวละคร

- 1 วิธีที่ 1 คลิกเครื่องมือ  (Delete)
- 2 หรือ วิธีที่ 2 คลิกขวาที่ตัวละคร
- 3 คลิกคำสั่ง delete



2. การเพิ่มตัวละคร (Sprite) จากไฟล์ภาพ ดังนี้

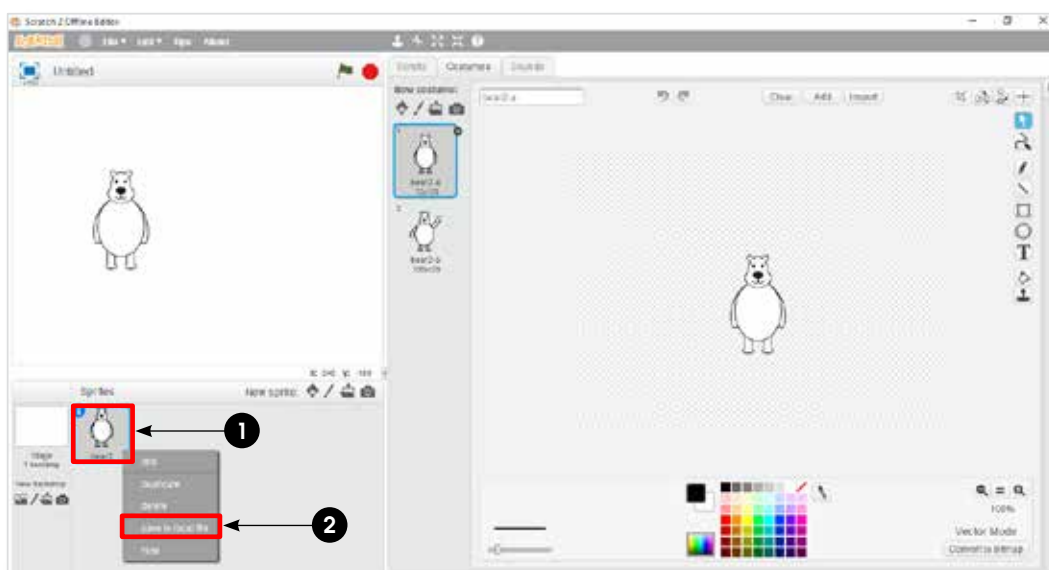
1. คลิกที่ไอคอน  (Choose sprite from library) จะปรากฏหน้าต่าง Sprite Library ดังรูป



2. คลิก Category : Animals
3. เลือกชุดตัวละคร (Sprite)
4. คลิกปุ่ม

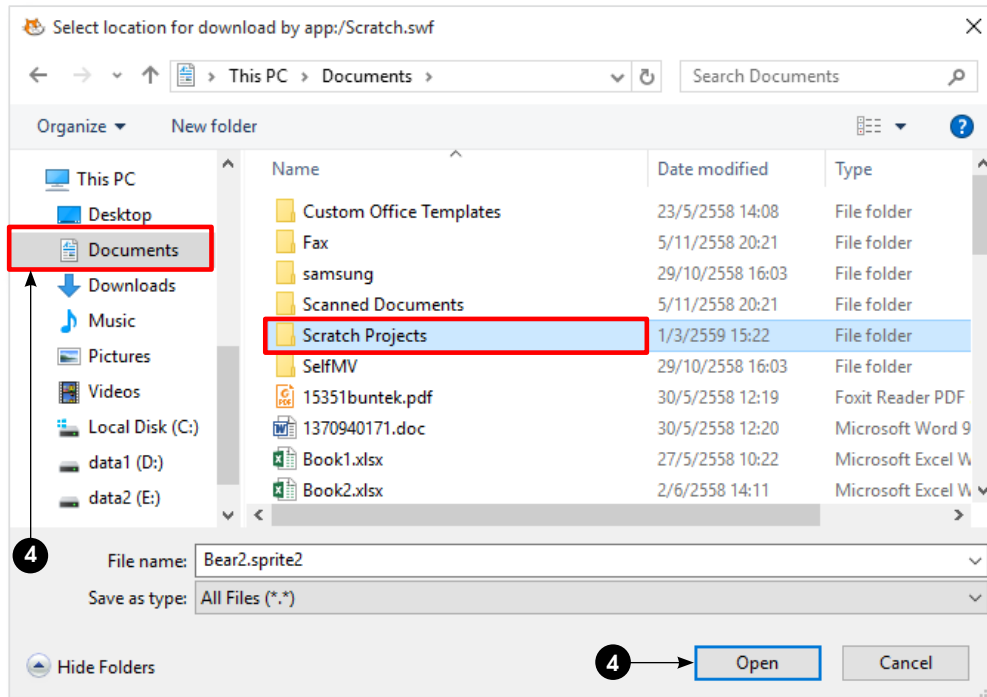
การส่งออกตัวละคร

1. คลิกขวาที่ตัวละคร
2. เลือกคำสั่ง Save to local file



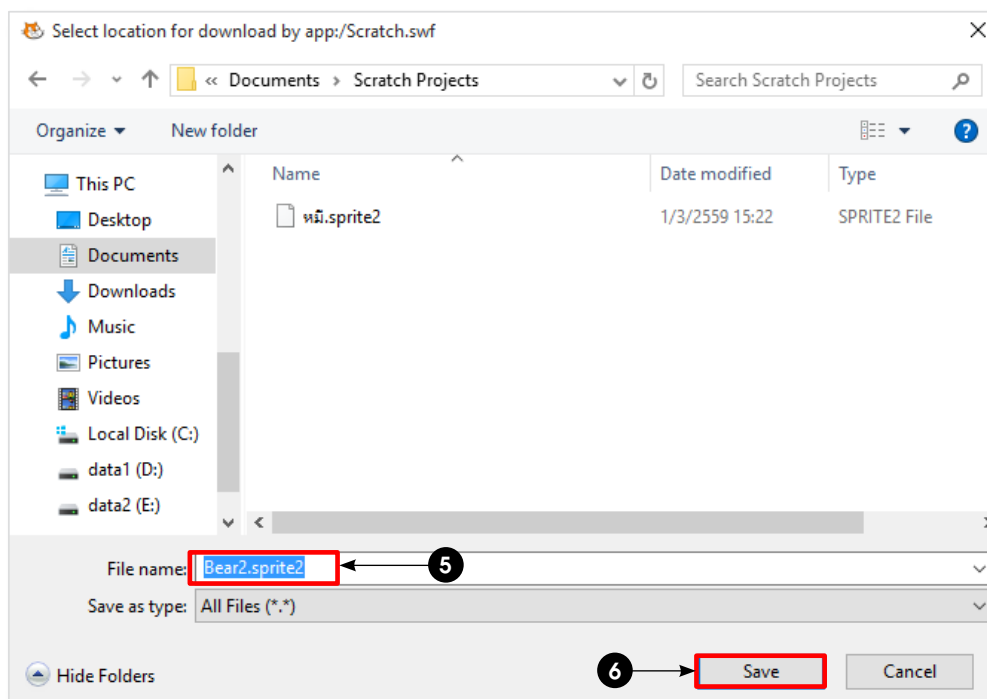
3 คลิกเลือกโฟลเดอร์ที่ต้องการจัดเก็บ Documents / Scratch Projects

4 คลิกปุ่ม




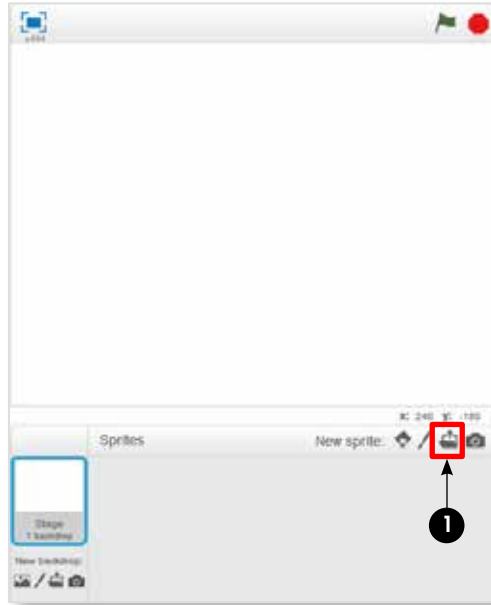
5 ตั้งชื่อไฟล์ตัวละคร

6 คลิกปุ่ม

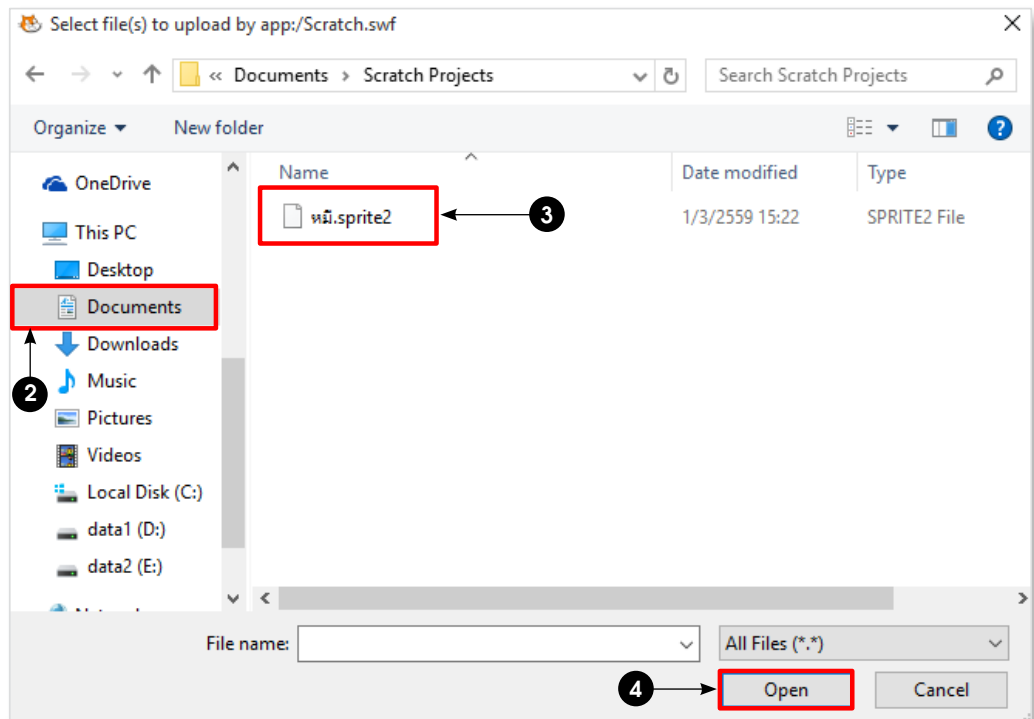


การเพิ่มตัวละคร

- 1 คลิกไอคอน New sprite : Upload sprite from file 

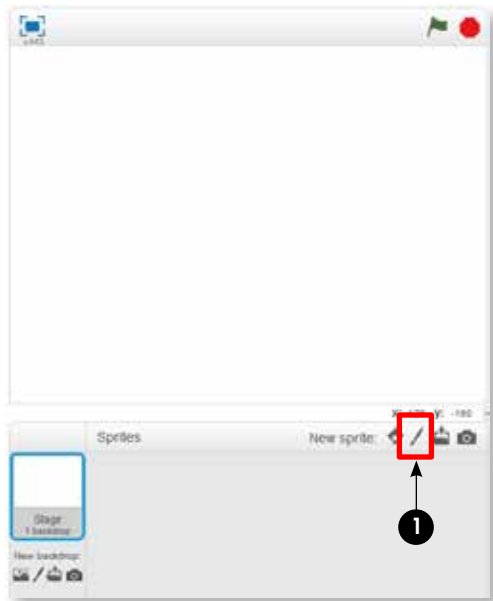


- 2 เลือกโฟลเดอร์ที่ได้เก็บบันทึกไว้ Documents/Scratch Projects
- 3 เลือกไฟล์ตัวละคร
- 4 คลิกปุ่ม

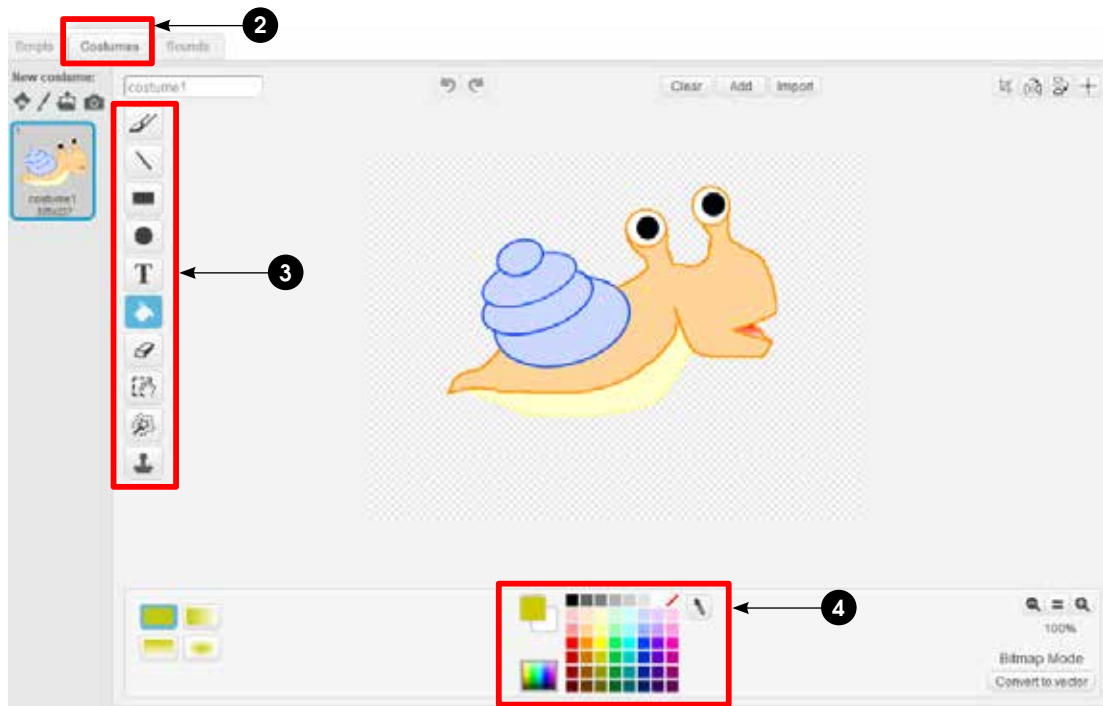


การเพิ่มตัวละครแบบวาดขึ้นเอง

- 1 คลิกไอคอน New sprite : Paint new sprite 

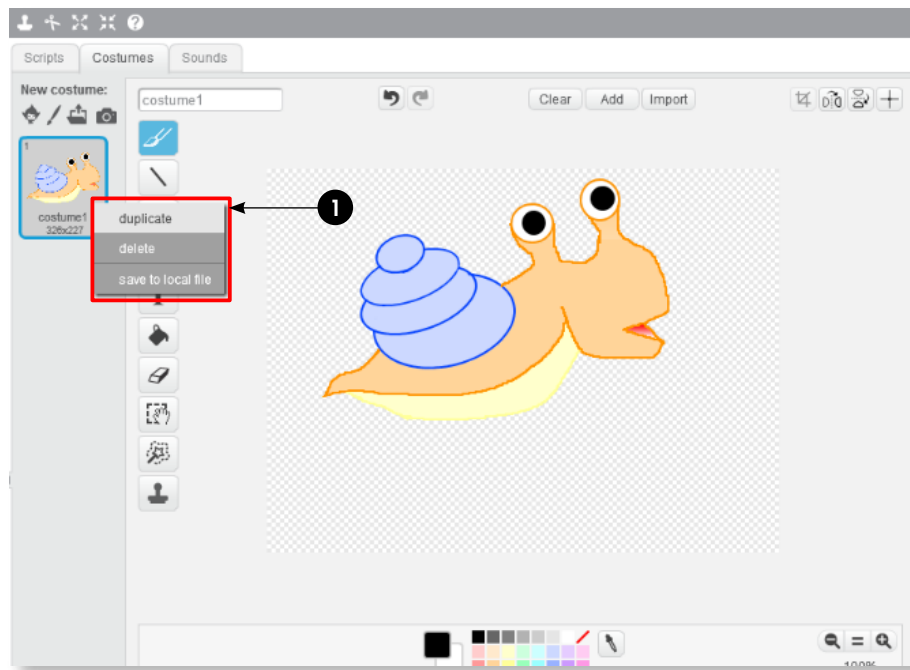



- 2 ปรากฏหน้าต่างของแท็บ Costumes ตัวอย่างเป็นการสร้างตัวละครใหม่เป็นรูปหอยทาก
- 3 เมนูเครื่องมือในการวาด
- 4 กล่องสี

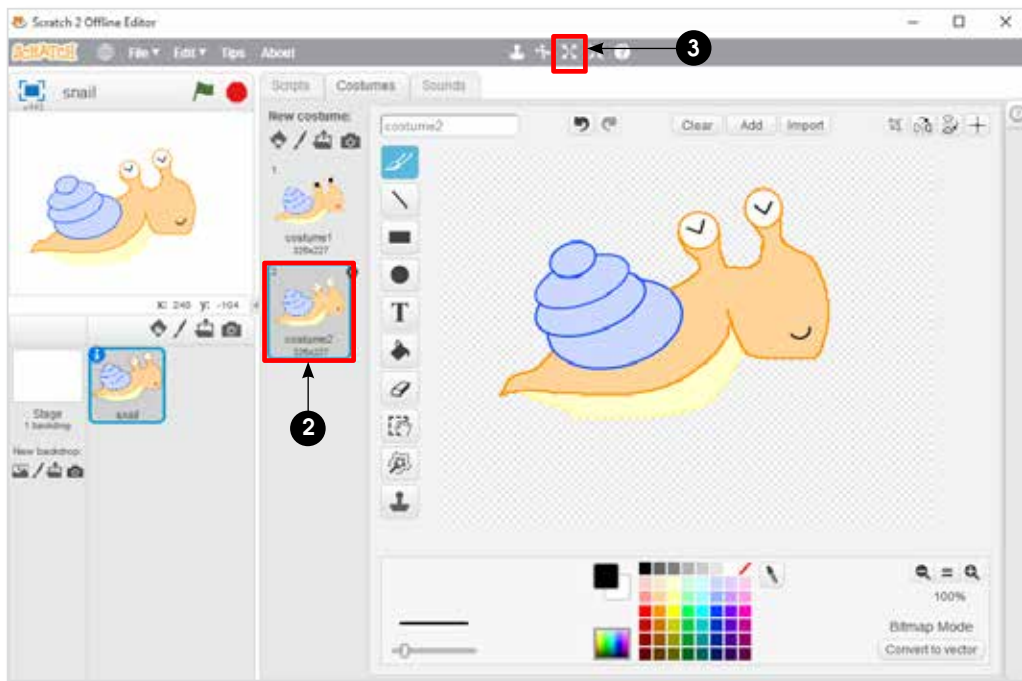


การเพิ่มชุดตัวละคร

- 1 คลิกขวาที่รูปภาพ costume1 เลือกคำสั่ง duplicate เพื่อสร้างชุดตัวละครใหม่เหมือนกับชุดตัวละคร costume1



- 2 คลิกรูปภาพ costume2 เพื่อแก้ไขชุดตัวละคร
- 3 คลิกเครื่องมือ  (Grow) เพื่อขยายขนาดรูปภาพเพื่อปรับแก้ชุดตัวละคร



ใบความรู้ที่ 2.2 พิกัดและทิศทาง

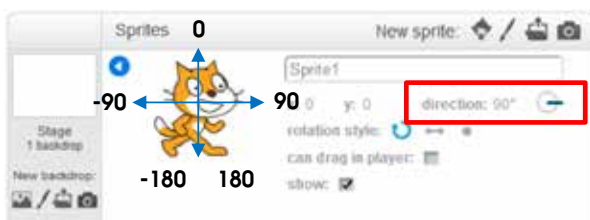
การทำให้ตัวละคร (Sprite) สามารถเคลื่อนที่ไปได้ในทิศทางต่างๆ สามารถเขียนสคริปต์ให้ตัวละคร เพื่อให้การแสดงผลการเคลื่อนที่ของตัวละครมีลักษณะเป็นธรรมชาติมากที่สุด ซึ่งบล็อกที่ใช่มีดังต่อไปนี้

กลุ่มบล็อก **Motion** เพื่อให้ตัวละครเคลื่อนที่ไปในทิศทางต่างๆ

	หมุนตามเข็มนาฬิกา ตัวอย่าง หมุนตามเข็มนาฬิกา 15 องศา
	หมุนทวนเข็มนาฬิกา ตัวอย่าง หมุนทวนเข็มนาฬิกา 15 องศา
	ระบุทิศทางที่ต้องการ ตัวอย่าง ระบุทิศทาง 90 องศา
	ไปที่ตำแหน่ง (x, y) บนเวที ตัวอย่าง ไปตรงกลางเวที (x=0, y=0)
	ตั้งค่าตำแหน่งตัวละครไปด้านซ้ายหรือขวา ตัวอย่าง ตั้งค่าตัวละครไปที่ตำแหน่ง $x = 0$ คือ กลางเวที $x = 240$ คือ ขอบขวาสุด $x = -240$ คือ ขอบซ้ายสุด
	ตั้งค่าตำแหน่งตัวละครไปด้านบนหรือด้านล่าง ตัวอย่าง ตั้งค่าตัวละครไปที่ตำแหน่ง $y = 0$ คือ กลางเวที $y = 180$ คือ ขอบบนสุด $y = -180$ คือ ขอบล่างสุด

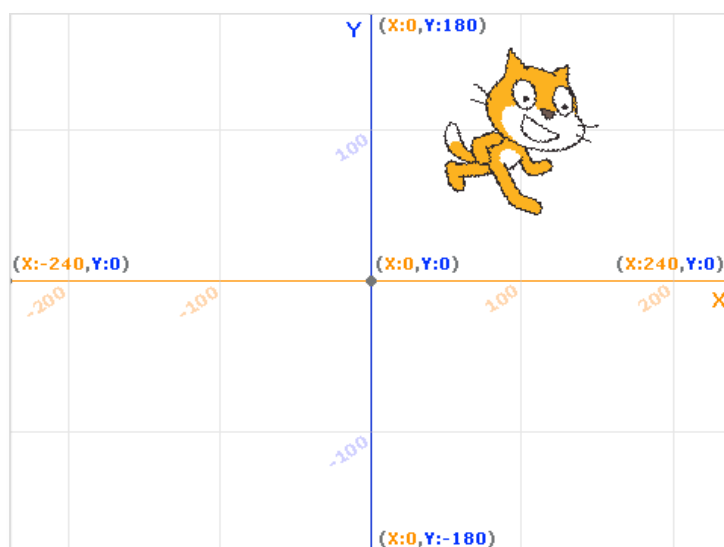
การกำหนดทิศทางและตำแหน่งการเคลื่อนที่ของตัวละคร

เส้นสีน้ำเงินเป็นตัวกำหนดทิศทางการเคลื่อนที่ของตัวละคร ในภาพแมวมามีทิศทางการเคลื่อนที่ 108 องศา ทิศทางการเคลื่อนที่ที่ระบุในโปรแกรม Scratch เช่น



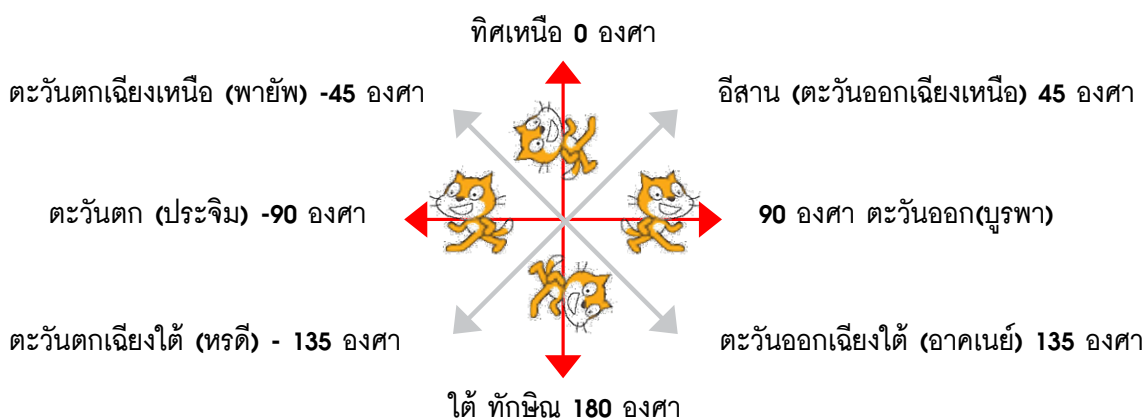
0 องศาจะมีทิศทางการเคลื่อนที่ไปด้านบน
90 องศาจะมีทิศทางการเคลื่อนที่ไปทางขวา
-90 องศาจะมีทิศทางการเคลื่อนที่ไปทางซ้าย
180, -180 องศาจะมีทิศทางการเคลื่อนที่ไปด้านล่าง

ตัวอย่าง ตัวละครแมวอยู่บนเวทีที่ ตำแหน่ง $x: 100$ $y:100$ บนเวที และมีทิศทางการเคลื่อนที่ 108 องศา



การหมุนของตัวละคร ถ้าค่าเป็นบวก จะเป็นการหมุนตามเข็มนาฬิกา ถ้าค่าเป็นลบจะหมุนทวนเข็มนาฬิกา เช่น -180 มีทิศทางการเคลื่อนที่ไปด้านล่างเหมือนกับทิศ 180 องศา แต่เป็นการหมุนทวนเข็มนาฬิกา

ตัวอย่าง การระบุทิศทางของตัวละครในโปรแกรม Scratch



การเคลื่อนที่ไปได้ในทิศทางต่างๆ ของตัวละครจะไม่เกี่ยวข้องกับลักษณะการหันหน้าของตัวละคร ดังนั้นในการทำให้ตัวละครเคลื่อนที่ ผู้เรียนควรทำความเข้าใจกับประเด็นนี้ เพื่อให้การแสดงผลการเคลื่อนที่ของตัวละครมีลักษณะเป็นธรรมชาติมากที่สุด

กิจกรรมที่ 3 สนุกกับเสียง

1. ผลการเรียนรู้

1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม

2. สารการเรียนรู้

2.1 การเพิ่มเสียงให้ตัวละครและเวที

2.2 การเขียนสคริปต์เพื่อแสดงเสียง

3. จุดประสงค์ ให้ผู้เรียนสามารถ

3.1 เพิ่มเสียงให้ตัวละครและเวทีโดยการแทรกไฟล์เสียง

3.2 เพิ่มเสียงให้ตัวละครโดยใช้โปรแกรมบันทึกเสียง

3.3 เขียนสคริปต์เพื่อเล่นเสียงด้วยกลุ่มบล็อก Sound โดยใช้บล็อก Play sound, Play note

4. แนวคิด

การเพิ่มเสียงลงในชิ้นงานจะช่วยเพิ่มความน่าสนใจให้กับชิ้นงานนั้นๆ เช่น เพิ่มเสียงให้ตัวละครขณะเคลื่อนไหว เพิ่มเสียงเมื่อเปลี่ยนฉาก การเพิ่มเสียงทำได้โดยแทรกไฟล์เสียงซึ่งมีส่วนขยายเป็น .wav หรือ .mp3 ที่โปรแกรมมีมาให้ หรือดาวน์โหลดจากแหล่งอื่นที่ไม่ผิดลิขสิทธิ์ หรือสร้างขึ้นเองจากเครื่องมือบันทึกเสียง (Sound Recorder) จากนั้นเขียนสคริปต์ โดยใช้บล็อก Play sound ในกลุ่มบล็อก Sound เพื่อเล่นเสียงดังกล่าว นอกจากนี้ยังมีบล็อกที่สามารถกำหนดเสียงของเครื่องดนตรีชนิดต่างๆ ได้ซึ่งสามารถนำมาสร้างสรรค์เป็นเพลงในจังหวะต่างๆ ตามความสนใจ

5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
3.1	เล่นกับเสียง	10
3.2	ดนตรีจินตนาการ	90

5.2 ใบความรู้

- ใบความรู้ที่ 3.1 โน้ตดนตรี

5.3

- อื่นๆ

6. วิธีดำเนินการ

6.1 การจัดเตรียม

6.1.1 ใบงานที่ 3.1 และ ใบงานที่ 3.2 ตามจำนวนผู้เรียน

6.1.2 ใบความรู้ที่ 3.1 ตามจำนวนผู้เรียน

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนแนะนำบล็อกในกลุ่มบล็อก Sound เพื่อทำกิจกรรมในครั้งนี้

6.2.2 ผู้เรียนแต่ละคนทำใบงานที่ 3.1 เรื่อง เล่นกับเสียง

6.2.3 ผู้สอนสุ่มผู้เรียนนำเสนอคำตอบ และร่วมกันสรุปเกี่ยวกับการใส่เสียงให้กับตัวละคร และเวที

6.2.4 ผู้เรียนศึกษาใบความรู้ที่ 3.1 เรื่อง โน้ตดนตรี

6.2.5 ผู้เรียนแต่ละคนทำใบงานที่ 3.2 เรื่อง ดนตรีจินตนาการ

6.2.6 ผู้สอนสุ่มผู้เรียนนำเสนอคำตอบ และร่วมกันสรุปการสร้างสรรค์ชิ้นงานที่มีเสียงดนตรีประกอบ

7. การวัดและประเมินผล

7.1 ตรวจคำตอบในใบงานและผลงาน

7.2 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

8.1 <https://scratch.mit.edu/> (01/03/2559)

8.2 <https://scratch.mit.edu/scratch2download/> (01/03/2559)

9. ข้อเสนอแนะ

-

ใบงานที่ 3.1 เล่นกับเสียง

ชื่อ-สกุล เลขที่

- 1) สร้างโปรเจกต์ใหม่ เขียนสคริปต์ให้ตัวละครแมวทั้ง 3 แบบ แล้วสังเกตผลลัพธ์แต่ละสคริปต์

1.1



1.2

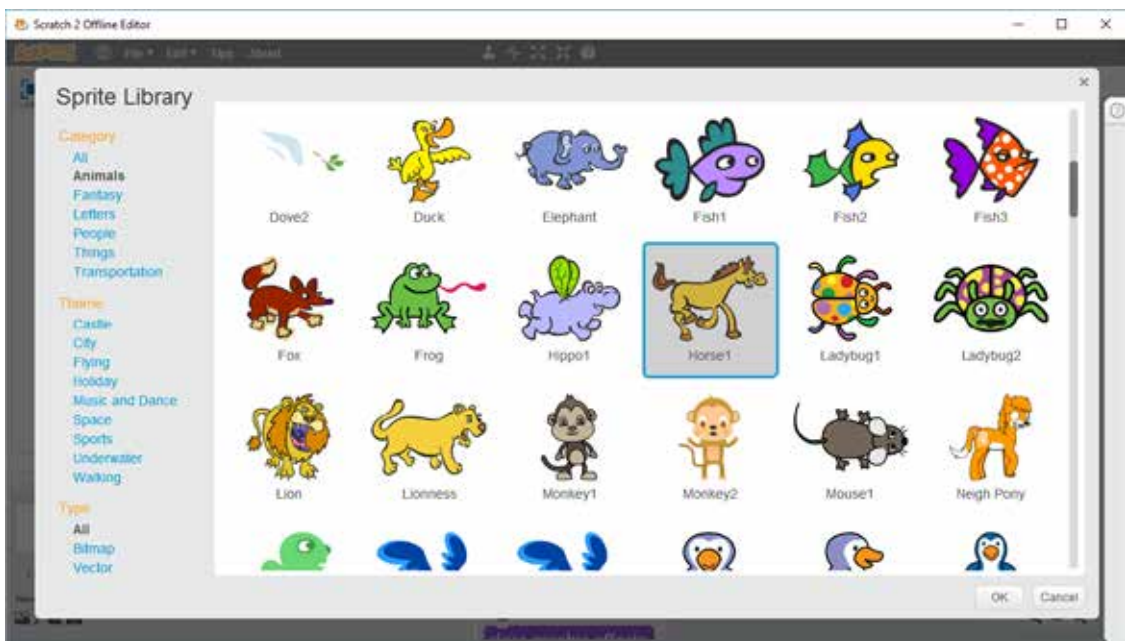


1.3

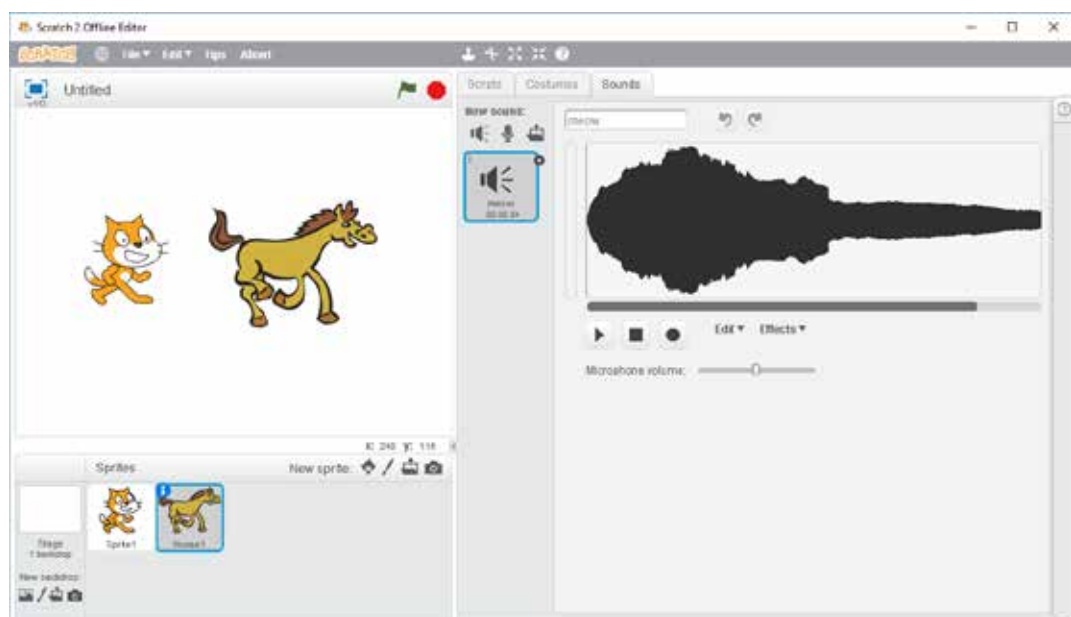


ผลลัพธ์ของ 3 สคริปต์นี้แตกต่างกันอย่างไร

- 2) สร้างตัวละครม้าและใส่เสียงม้า ดังนี้
2.1 เพิ่มตัวละครม้า เลือกประเภท Animals แล้วเลือก Horse1



2.2 เลือกแท็บ Sounds แล้วลบไฟล์เสียง meow ออก



2.3 เพิ่มเสียง horse

2.4 เขียนสคริปต์ เมื่อนำเมาส์ไปคลิกที่ตัวละครม้า ให้ส่งเสียงม้าร้อง

when this sprite clicked
play sound horse

2.5 เพิ่มตัวละครใหม่ และใส่เสียงให้กับตัวละคร พร้อมกับเขียนสคริปต์ ให้มีเสียงเมื่อคลิกที่ตัวละคร

2.6 เพิ่มฉากใหม่ และใส่เสียงให้ฉาก พร้อมกับเขียนสคริปต์ ให้มีเสียงเมื่อคลิกที่ฉากหลัง (หมายเหตุ : ใช้บล็อก when Stage clicked เพื่อกำหนดเหตุการณ์)

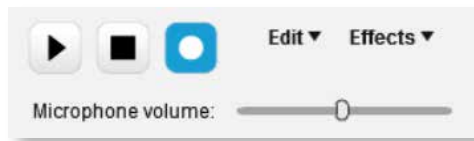
3 เพิ่มเสียงให้ตัวละครโดยใช้เครื่องมือบันทึกเสียง

3.1 คลิกแท็บ Sounds

3.2 คลิกปุ่ม

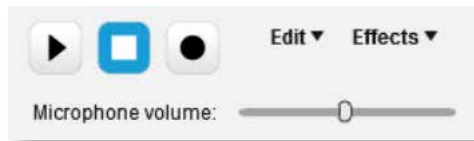


3.3 คลิกปุ่ม



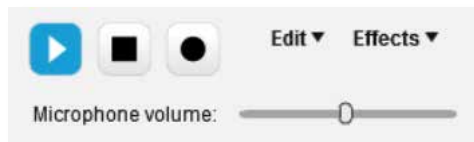
เพื่อบันทึกเสียงผ่านไมโครโฟน

3.4 คลิกปุ่ม



เพื่อสิ้นสุดการบันทึกเสียง

3.5 คลิกปุ่ม



เพื่อฟังเสียงที่อัดไว้



ใบงานที่ 3.2 กนตรีจินตนาการ

ชื่อ-สกุล เลขที่

1 เขียนสคริปต์ แล้วตอบคำถามต่อไปนี้

```

when clicked
  play note 55 for 0.3 beats
  play note 55 for 0.3 beats
  play note 52 for 0.5 beats
  play note 55 for 0.3 beats
  play note 57 for 0.3 beats
  play note 60 for 0.5 beats
  rest for 0.2 beats
  play note 62 for 0.4 beats
  play note 60 for 0.2 beats
  play note 57 for 0.5 beats
  play note 55 for 0.4 beats
  play note 55 for 0.2 beats
  play note 52 for 0.5 beats
    
```

ผลลัพธ์ที่ได้ คือ เพลง

เขียนโน้ตดนตรีของท่อนเพลงดังกล่าว

ปรับปรุงสคริปต์ เพื่อให้จังหวะดนตรีสมบูรณ์ขึ้น

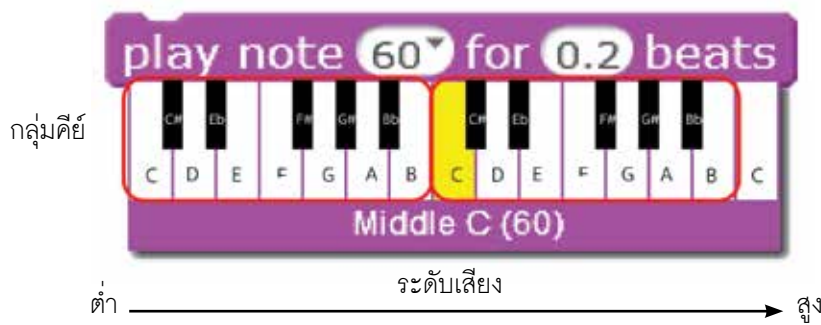
2 เขียนสคริปต์หรือระบุชนิดเครื่องดนตรี ลงในตารางต่อไปนี้

สคริปต์	ชนิดเครื่องดนตรี
2.1 <code>set instrument to 1</code>	
2.2 <code>set instrument to 5</code>	
2.3	Saxophone
2.4	Steel Drum

ใบความรู้ที่ 3.1 น้้กนตรี

โครงสร้างของคีย์บอร์ดในเปียโนจะแบ่งเป็นกลุ่มคีย์ออกเป็นกลุ่มๆ แต่ละกลุ่มประกอบด้วยกลุ่มที่มีคีย์สีขาว 7 ตัว (C-โด D-เร E-มี F-ฟา G-ซอล A-ลา B-ที) และคีย์สีดำ 5 ตัว (C#, Eb, F#, G#, Bb) เสียงดนตรีในกลุ่มคีย์สีขาวทั้ง 7 เสียงนั้น ถือเป็นระดับเสียงตามธรรมชาติของมนุษย์ ซึ่งระดับเสียงจะห่างกัน 1 เสียงเต็ม ส่วนเสียงดนตรีในกลุ่มคีย์สีดำไม่ใช่เสียงเต็มแบบธรรมชาติ จึงต้องมีการสร้างสัญลักษณ์เพื่อให้โน้ตนั้นมีระดับเสียงสูงขึ้นหรือต่ำลงครึ่งเสียงและสัญลักษณ์ดังกล่าวก็คือเครื่องหมายชาร์ป (#) และแฟลท (b)

ในโปรแกรม Scratch ใช้บล็อก play note ระบุตัวเลขเพื่อแทนระดับเสียงดนตรี ดังตัวอย่างในตาราง



ตารางแสดงเสียงดนตรีในกลุ่มคีย์สีขาวกับค่าตัวเลขในบล็อก play note

ระบบซอล-ฟา	ระบบตัวอักษร	ค่าตัวเลขในบล็อก play note
โด (ต่ำ)	C	48
เร (ต่ำ)	D	50
มี (ต่ำ)	E	52
ฟา (ต่ำ)	F	53
ซอล (ต่ำ)	G	55
ลา (ต่ำ)	A	57
ที (ต่ำ)	B	59
โด	C	60
เร	D	62
มี	E	64
ฟา	F	65
ซอล	G	67
ลา	A	69
ที	B	71
โด (สูง)	C	72

- เครื่องหมาย ชาร์ป (Sharp: #) เมื่อปรากฏที่โน้ตตัวใดจะทำให้โน้ตนั้นมีระดับเสียงสูงขึ้นครึ่งเสียงเช่น C# อ่านว่า ซี-ชาร์ป จะมีระดับเสียงสูงกว่า C (โด) อยู่ครึ่งเสียง ซึ่งใน Scratch จะเป็นการเพิ่มค่าตัวเลข 1 ค่า ตัวอย่างเช่น C มีค่าตัวเลขในบล็อก play note เป็น 48 C# จะมีค่าเป็น 49

- เครื่องหมาย แฟล็ต (Flat: b) ตรงกันข้ามกับ # เมื่อปรากฏที่โน้ตตัวใด จะทำให้โน้ตนั้นมีระดับเสียงต่ำลงครึ่งเสียง เช่น Eb อ่านว่า อี-แฟล็ต จะมีระดับเสียงต่ำกว่า E (มี) อยู่ครึ่งเสียง ซึ่งใน Scratch จะเป็นการเพิ่มค่าตัวเลข 1 ค่า ตัวอย่างเช่น E มีค่าตัวเลขในบล็อก play note เป็น 52 Eb จะมีค่าเป็น 51



กรณีที่ต้องการโน้ตที่มีค่าสูงหรือต่ำกว่าค่าในตาราง สามารถกรอกตัวเลขในบล็อก play note ให้มีค่าสูงต่ำกว่านั้นได้ตามต้องการ

กิจกรรมที่ 4 ทละครโต้ตอบ

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม
- 1.2 ใช้งานตัวแปร

2. สารการเรียนรู้

- 2.1 การใช้บล็อก say
- 2.2 การใช้บล็อก ask
- 2.3 การใช้บล็อก join

3. จุดประสงค์ ให้ผู้เรียนสามารถ

เขียนสคริปต์ด้วยบล็อก say, ask และ join

4. แนวคิด

บล็อก say ใช้สำหรับแสดงคำพูด บล็อก think ใช้สำหรับแสดงความคิด ทั้งสองบล็อกนี้อยู่ในกลุ่มบล็อก Looks

บล็อก ask อยู่ในกลุ่มบล็อก Sensing ใช้สำหรับแสดงข้อความและรับข้อมูล ซึ่งข้อมูลที่ได้รับจะถูกนำมาเก็บไว้ในตัวแปรชื่อ answer โดยอัตโนมัติ

บล็อก join อยู่ในกลุ่มบล็อก Operators ใช้สำหรับผสมข้อความกับข้อความ ข้อมูลกับข้อมูลที่เก็บในตัวแปร หรือข้อมูลที่เก็บไว้ในตัวแปรต่างๆ โดยบล็อก join จะถูกนำไปใช้ร่วมกับบล็อกอื่นๆ

5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
4.1	สร้างบทละครโต้ตอบ	60

5.2 ใบความรู้

- ใบความรู้ที่ 4.1 เรื่อง คำสั่งบล็อก say บล็อก ask และบล็อก join

5.3 อื่นๆ

-

6. วิธีดำเนินการ

6.1 การจัดเตรียม

6.1.1 ใบงานที่ 4.1 และใบความรู้ที่ 4.1 ตามจำนวนผู้เรียน

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนกระตุ้นความสนใจของผู้เรียนโดยให้ผู้เรียนช่วยกันคิดว่าจะสามารถนำโปรแกรม Scratch มาสร้างโปรเจกต์ได้อย่างไร

6.2.2 ผู้เรียนศึกษาใบความรู้ที่ 4.1 เรื่อง บล็อก ask บล็อก say และบล็อก join

6.2.3 ผู้สอนแนะนำให้ผู้เรียนรู้จักกับสคริปต์และบล็อกที่ใช้ในการรับข้อความ การนำข้อความที่รับเข้าไปแสดงผล

6.2.4 ผู้เรียนทำใบงานที่ 4.1 เรื่องสร้างบทละครโต้ตอบ ตอนที่ 1 โดยให้ผู้เรียนทำพร้อมกันทั้งห้อง ทีละข้อ และผู้สอนถามคำตอบในแต่ละข้อ

6.2.5 ผู้เรียนทำใบงานที่ 4.1 ตอนที่ 2 โดยผู้สอนคอยให้คำแนะนำ จากนั้นสุ่มผู้เรียนนำเสนอ โปรเจกต์ที่ได้จากการทำใบงาน

6.2.6 ผู้เรียนและผู้สอนร่วมกันสรุปเกี่ยวกับการเขียนสคริปต์โดยใช้บล็อก say, ask และ join

7. การวัดและประเมินผล

7.1 ตรวจคำตอบในใบงาน

7.2 สังเกตการร่วมกิจกรรมในการระดมความคิดของผู้เรียน

7.3 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

8.1 <http://scratch.mit.edu/> (01/03/59)

8.2 <https://scratch.mit.edu/scratch2download/> (01/03/59)

9. ข้อเสนอแนะ

9.1 การทำใบงานที่ 4.1 ผู้สอนควรให้ผู้เรียนทุกคนทำใบงานทีละข้อพร้อมๆ กัน และโดยให้ผู้เรียนแต่ละคนทำหน้าที่ดูแลเพื่อนที่อื่นๆ ช่างๆ เพื่อทำใบงานไปพร้อมๆ กัน หรือช่วยเหลือกัน

ใบงานที่ 4.1 สร้างบทละครโต้ตอบ

ชื่อ-สกุล เลขที่

ตอนที่ 1 เขียนบทสนทนาให้ตัวละคร

- สร้างโปรเจกต์ใหม่ เพิ่มตัวละครและเขียนสคริปต์ให้ตัวละคร ดังนี้



เมื่อคลิก สิ่งที่เกิดขึ้น แล้วตอบคำถาม
 การทำงานของ บล็อก say และ บล็อก ask แตกต่างกันอย่างไรร

- ลบตัวละครในข้อ 1 แล้วสร้างตัวละครเด็กหญิงในชุดบัลเลย์ และเขียนสคริปต์ดังต่อไปนี้



answer
 บล็อก answer เป็นคู่หูของบล็อก ask มี ask ที่ไหน มี answer ที่นั่น เมื่อบล็อก ask แสดงคำถาม ให้ผู้ใช้ตอบ บล็อก answer จะทำหน้าที่เก็บคำตอบ ที่ผู้ใช้พิมพ์ลงไป

- คลิก สิ่งเกิดผลการทำงาน และตอบคำถามของเด็กหญิง
 ชื่อที่นักเรียนตอบคือ

นักเรียนคิดว่าชื่อหรือคำที่นักเรียนได้พิมพ์เป็นคำตอบ ข้อมูลนี้จะถูกเก็บไว้ในบล็อกใด

ask เราชื่ออะไรจ๊ะ and wait answer

- 4) ปรับปรุงสคริปต์ในข้อ 2 โดยใช้บล็อก `join hello world` ร่วมกับ `say Hello! for 2 secs` ดังต่อไปนี้



```

when clicked
say สวัสดีค่ะ for 2 secs
ask เธอชื่ออะไรจะ and wait
say join ยินดีที่ได้รู้จัก answer for 2 secs
  
```

- 5) คลิก  สิ่งเกิดผลการทำงาน

บล็อก join อยู่ในกลุ่มบล็อก Operators เป็นบล็อกแห่งมิตรภาพ เพราะสามารถนำไปวางแทรกไว้ในบล็อกต่างๆ เพื่อเป็นตัวกลางเชื่อมสัมพันธ์ระหว่างข้อความกับบล็อก หรือบล็อกกับบล็อกก็ได้



หลังจากใช้บล็อกคำสั่ง `say Hello! for 2 secs` ร่วมกับ `join hello world` ตัวละครเด็กหญิง จะแสดงข้อความ

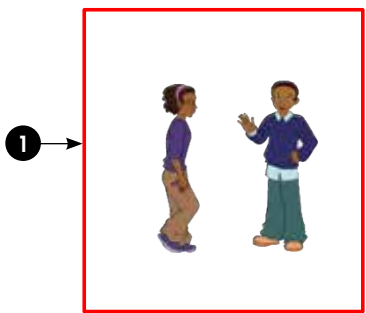
- 6) โยงเส้นจับคู่ระหว่างบล็อกและความหมายที่สัมพันธ์กัน

<code>say Hello!</code>	<input type="radio"/>	<input type="radio"/>	ถามคำถามที่กำหนด และรอผู้ใช้พิมพ์คำตอบ
<code>say Hello! for 2 secs</code>	<input type="radio"/>	<input type="radio"/>	ใช้สำหรับเชื่อมข้อความเข้าด้วยกัน
<code>ask What's your name? and wait</code>	<input type="radio"/>	<input type="radio"/>	แสดงข้อความที่กำหนด
<code>join hello world</code>	<input type="radio"/>	<input type="radio"/>	แสดงข้อความที่กำหนดเป็นเวลา 2 วินาที

- 7) บันทึกเป็นโปรเจกต์ใหม่ ชื่อ Activity 4_1



ตอนที่ 2 ถามตอบกันเถอะ

- 1) สร้างโปรเจกต์ใหม่ โดยมีตัวละคร 2 ตัว ดังภาพ 1
- 2) เพิ่มฉากให้กับเวที 1 ฉาก กำหนดให้ใช้เป็นฉากเริ่มต้น ดังภาพ 2



3 เขียนสคริปต์เริ่มต้นให้กับตัวละครต่างๆ ให้มีการทำงาน ดังนี้

ภาพ	บทสนทนา
	<p>เมื่อคลิกธงเขียว ให้ตัวละครผู้หญิง และตัวละครผู้ชายปรากฏขึ้นมา</p>
	<p>จากนั้นตัวละครทั้งสองเดินเข้ามาเจอกันตรงกลางหน้าจอ</p>

ตัวละครผู้หญิง	ตัวละครผู้ชาย
<p>เมื่อคลิก </p> <p>ไป ณ ตำแหน่ง $x = -196$ และ $y = -10$ ไป ณ ตำแหน่ง $x = 50$ $y = -10$ ภายใน 1 วินาที เขียนเป็นสคริปต์ ได้ดังนี้</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	<p>เมื่อคลิก </p> <p>ไป ณ ตำแหน่ง $x = 181$, $y = -10$ ไป ณ ตำแหน่ง $x = 100$, $y = -10$ ภายใน 1 วินาที เขียนเป็นสคริปต์ ได้ดังนี้</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>

4 เพิ่มสคริปต์ให้กับตัวละครผู้หญิง และตัวละครผู้ชาย ให้มีการทำงานตามเรื่องราว ดังนี้

ภาพ	บทสนทนา
	<p>ตัวละครผู้หญิง กล่าวคำทักทาย</p>
	<p>จากนั้น ตัวละครผู้ชาย ทักทายตอบ</p>
	<p>ตัวละครผู้หญิง ถามว่า “เธอรู้หรือเปล่าว่า ประเทศใดในอาเซียนที่มีทะเลทราย ?”</p>
	<p>ตัวละครผู้หญิง แสดงคำตอบที่ผู้ใช้ป้อนลงในกล่องข้อความ</p>

ภาพ	บทสนทนา
	<p>หลังจากแสดงคำตอบเสร็จแล้ว ตัวละครผู้หญิงพูดว่า “ลองกดแป้น Space bar ดูซิจ๊ะ”</p>



มาเพิ่มสคริปต์ให้กับตัวละครผู้หญิง และตัวละครผู้ชาย ให้ทำงานตามเรื่องราวข้างต้นกัน

ตัวละครผู้หญิง	ตัวละครผู้ชาย
<p>พูด “สวัสดีจ๊ะ” เป็นเวลา 4 วินาที รอ 2 วินาที</p> <p>ถาม “เธอรู้หรือเปล่าว่า ประเทศใด ในอาเซียนที่มีทะเลทราย?”</p> <p><i>จะปรากฏกล่องข้อความให้ผู้ไขป้อนคำตอบ</i></p> <p>พูด “คำตอบของเธอคือ ___” ตามด้วย ข้อความที่ผู้ใช้ป้อนลงไปในกลุ่มข้อความ และ แสดงข้อความนี้เป็นเวลา 2 วินาที</p> <p>พูด “ลองกดแป้น Space bar ดูซิจ๊ะ” เป็นเวลา 4 วินาที</p> <p>เขียนเป็นสคริปต์ ได้ดังนี้</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	<p>พูด “สวัสดีครับ” เป็นเวลา 2 วินาที</p> <p>เขียนเป็นสคริปต์ ได้ดังนี้</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>


5 เมื่อมีการกดแป้น space bar แล้วมีเหตุการณ์ต่อเนื่องดังต่อไปนี้

ภาพ	บทสนทนา
	<p>เมื่อมีเหตุการณ์กดแป้น space bar ตัวละครผู้ชาย “จะถามคำถามอะไรดีนะ อ้อ....คิดออกแล้ว”</p>
	<p>ตัวละครผู้ชาย ถามว่า “เธอรู้หรือไม่ว่า ประเทศใดในอาเซียนที่มีขนาดพื้นที่เล็กที่สุด”</p>
	<p>ตัวละครผู้ชาย แสดงคำตอบที่ผู้ใช้ป้อนลง ในกล่องข้อความ</p>
	<p>ตัวละครผู้ชาย กล่าวคำอำลา</p>



when space key pressed

บล็อก when space key pressed อยู่ในกลุ่มบล็อก Events ใช้สำหรับ รับค่าการกดแป้น space bar บนคีย์บอร์ด เมื่อผู้ใช้มีการกดแป้น space bar บล็อกคำสั่งต่างๆ ที่อยู่ภายใต้บล็อก when space key pressed ก็จะเริ่มทำงาน

นอกจากนี้ ถ้าต้องการกำหนดเหตุการณ์ให้สามารถรับค่าการกดแป้นตัวอักษรใดๆ บนคีย์บอร์ด สามารถทำได้โดยการคลิกเลือก  เพื่อเปลี่ยนเป็นแป้นตัวอักษรใดๆ บนคีย์บอร์ดได้



ให้เขียนสคริปต์ของตัวละครผู้ชาย เมื่อมีการกดแป้น space bar

การทำงาน	สคริปต์ของตัวละครผู้ชาย
เมื่อกดแป้น space bar
คิด “จะถามคำถามอะไรดีนะ อ้อ...คิดออกละ” เป็นเวลา 2 วินาที
ถาม “เธอหละรู้หรือเปล่าว่า ประเทศใดในอาเซียนที่มีขนาดพื้นที่เล็กที่สุด”
จะปรากฏกล่องข้อความให้ผู้ใช้ป้อนคำตอบ
พูด “คำตอบของเธอคือ ___” ตามด้วย
ข้อความที่ผู้ใช้ป้อนลงไปในกลุ่มข้อความ และแสดงข้อความนี้เป็นเวลา 2 วินาที
พูด “Bye Bye...” เป็นเวลา 2 วินาที

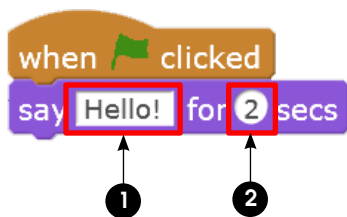
6) บันทึกไฟล์ชื่อ activity4_2

ใบความรู้ที่ 4.1 บล็อก say บล็อก ask และบล็อก join

บล็อก **say Hello! for 2 secs**

บล็อก say อยู่ในกลุ่มบล็อก Looks ใช้สำหรับให้ตัวละครพูด พร้อมกับระบุระยะเวลาในการแสดงข้อความที่พิมพ์ลงไปในช่วง say

สคริปต์



เวที

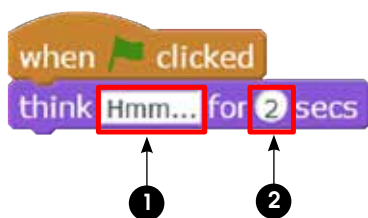


- 1 การใช้บล็อก say เพื่อแสดงข้อความที่ต้องการให้ตัวละครพูด
- 2 ช่องรับค่าตัวเลขแสดงระยะเวลาที่ต้องการให้ข้อความแสดง หน่วยเป็นวินาที

บล็อก **think Hmm... for 2 secs**

บล็อก think อยู่ในกลุ่มบล็อก Looks ใช้สำหรับให้ตัวละครคิด พร้อมกับระบุระยะเวลาในการแสดงข้อความที่พิมพ์ลงไปในช่วง think ลักษณะการใช้งานจะเหมือนกับบล็อก say

สคริปต์



เวที



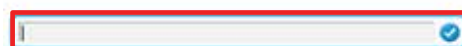
- 1 การใช้บล็อก think เพื่อแสดงข้อความที่ต้องการให้ตัวละครคิด
- 2 ช่องรับค่าตัวเลขแสดงระยะเวลาที่ต้องการให้ข้อความแสดง หน่วยเป็น วินาที

บล็อก ask What's your name? and wait

บล็อก ask อยู่ในกลุ่มบล็อก Sensing ใช้สำหรับแสดงคำถามและรอให้ผู้ใช้พิมพ์คำตอบ เช่น

สคริปต์

```
when clicked
ask What's your name? and wait
```



- 1 การใช้นบล็อก ask เพื่อแสดงคำถาม
 - 2 ช่องรับคำตอบ ซึ่งผู้ใช้จะต้องพิมพ์คำตอบแล้วกดแป้น Enter หรือคลิกที่เครื่องหมาย ✓
- สังเกตได้ว่าส่วนท้ายของบล็อก ask จะมีคำว่า and wait อยู่ด้วย ส่วนนี้เป็นการรอคำตอบเมื่อผู้ใช้พิมพ์คำตอบลงไปในช่วงรับคำตอบ คำตอบนั้นจะถูกนำไปเก็บไว้ที่บล็อก `answer` โดยอัตโนมัติเพื่อนำไปใช้ในการแสดงผลร่วมกับบล็อกอื่นๆ เช่น

```
when clicked
ask What's your name? and wait
say answer for 2 secs
```



บล็อก join hello world

บล็อก join อยู่ในกลุ่มบล็อก Operators ใช้สำหรับเชื่อมต่อความเข้าด้วยกัน เราอาจใช้นบล็อก say หรือ บล็อก think เพื่อแสดงข้อความจากบล็อก join ได้ เช่น

```
say join พิมพ์ข้อความตรงนี้ answer for 2 secs
```

```
think join answer ข้อความของฉัน
```

ตัวอย่างการใช้นบล็อก join เพื่อแสดงข้อความและคำตอบ เช่น

```
1. when clicked
ask What's your name? and wait
say join Your name is answer for 10 secs
```



```
2. when clicked
ask ใครคือเพื่อน ที่สนิทที่สุด and wait
say join answer น่ารักที่สุด for 10 secs
```



กิจกรรมที่ 5 กระจายสารและรับสาร

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม
- 1.2 สร้างชิ้นงานจากจินตนาการ

2. สารการเรียนรู้

- 2.1 บล็อก broadcast และบล็อก broadcast and wait
- 2.2 บล็อก when I receive
- 2.3 บล็อก show และ hide

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 เขียนสคริปต์ด้วยบล็อก broadcast, broadcast and wait และ when I receive

4. แนวคิด

บล็อก broadcast อยู่ในกลุ่มบล็อก Events ใช้สำหรับกระจายสารให้กับตัวละครต่างๆ เหมือนการสื่อสารกันไปยังตัวละคร/เวที โดยใช้งานร่วมกับบล็อก when I receive ซึ่งใช้รับสารเพื่อดำเนินการต่อ

บล็อก show ใช้สำหรับแสดงตัวละคร และบล็อก hide ใช้สำหรับซ่อนตัวละคร

5. สื่ออุปกรณ์

- 5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
5.1	กระจายสารและรับสาร	30
5.2	สนทนาพาเพลิน	90

5.2 ใบความรู้

- ใบความรู้ที่ 5.1 เรื่อง การกระจายสารและรับสาร

5.3 อื่นๆ

- บัตรคำ broadcast และการ when I receive (อาจเตรียมเท่ากับจำนวนที่แสดงในตัวอย่าง โดยไม่ต้องเตรียมเท่าจำนวนนักเรียนทั้งหมดในห้อง)
- ตัวอย่างสรุปเกมกระจายสาร

6. วิธีดำเนินการ

6.1 การจัดเตรียม

6.1.1 ใบงานที่ 5.1-5.2 ตามจำนวนผู้เรียน

6.1.2 ใบความรู้ที่ 5.1 ตามจำนวนผู้เรียน

6.1.3 บัตรคำ broadcast และบัตรคำ when I receive จำนวน 1 ชุด

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนนำอธิบายเกี่ยวกับการกระจายสารและรับสารว่า ตัวละครต่างๆ สามารถส่งสารระหว่างกันได้ และบทบาทของตัวละครหลังจากการได้รับสารอาจแตกต่างกัน

6.2.2 เล่นเกมกระจายสาร โดยผู้สอนแจก บัตรคำ when I receive ให้กับผู้เรียน โดยสุ่มแจกบัตรคำ ผู้สอนกระจายสารตามบัตรคำ broadcast และสังเกตผลพร้อมสรุปผลเกมกระจายสาร

6.2.3 ผู้เรียนศึกษาใบความรู้ที่ 5.1 เรื่อง กระจายสารและรับสาร จากนั้นทำใบงานที่ 5.1 เรื่องกระจายสารและรับสาร และใบงานที่ 5.2 สนทนาพาเพลิน แล้วสุ่มผู้เรียนออกมานำเสนอคำตอบ

6.2.4 ผู้เรียนและผู้สอนร่วมกันสรุปการใช้บล็อก broadcast และ when I receive

7. การวัดและประเมินผล

7.1 ตรวจสอบคำตอบในใบงานและผลงาน

7.2 สังเกตการร่วมกิจกรรมในการระดมความคิดของผู้เรียน

7.3 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

-

9. ข้อเสนอแนะ

-

ใบงานที่ 5.1 กระจายสารและรับสาร

ชื่อ-สกุล เลขที่

1 โครงร่างบทสนทนาการกระจายสารและรับสาร

ตัวละคร/เวที	เหตุการณ์เมื่อคลิก 	เหตุการณ์เมื่อมีการรับสาร
	พูด กระจายสาร	-
	ซ่อนตัวละครนักเรียน เปลี่ยนชุดตัวละครเริ่มต้น	แสดงตัวละครนักเรียน เปลี่ยนชุดตัวละครตามบัตรคำ
	เปลี่ยนฉากเป็น 	เปลี่ยนฉากเป็น 

2 สร้างโปรเจกต์ใหม่ มีตัวละคร 2 ตัว คือ ครูและนักเรียน โดยเพิ่มตัวละคร ดังนี้



Avery



Champ99

3 เขียนสคริปต์ให้กับตัวละครครู



บล็อก broadcast ใช้สำหรับกระจายข่าวสาร ให้ตัวละครต่างๆ ได้รับรู้

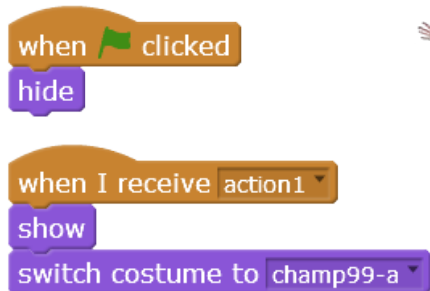
broadcast

วิธีกำหนดข้อความด้านท้ายบล็อก broadcast เพื่อกระจายสารคำว่า "action1"

- 1) เลือกบล็อก broadcast มาวาง
- 2) คลิกที่ลูกศรหลัง message1 หลังจากนั้นเลือก new message
- 3) พิมพ์คำว่า action1 และคลิก



4 เขียนสคริปต์ให้กับตัวละครนักเรียน



บล็อก when I receive เป็นคู่กันกับบล็อก broadcast ใช้สำหรับรับสารแล้วทำคำสั่งต่างๆ

บล็อก hide ใช้ซ่อนตัวละคร


บล็อก show ใช้แสดงตัวละคร

** อยากให้เห็นก็ show ไม่อยากให้เห็นก็ hide **

show **hide**



วิธีกำหนดข้อความด้านท้ายบล็อก when I receive เพื่อรับสารทำได้โดยคลิกที่ลูกศรหลัง message1 หลังจากนั้นเลือกชื่อสารที่ต้องการรับ



มาแปลสคริปต์กัน

ตัวละคร/เวที	สคริปต์	เหตุการณ์
	<pre> when clicked say action1 for 2 secs broadcast action1 </pre>	<p>เมื่อคลิก </p> <p>พูด "action1" เป็นเวลา 2 วินาที กระจายสาร "action1"</p>
	<pre> when clicked hide when I receive action1 show switch costume to champ99-a </pre>	<p>เมื่อคลิก </p> <p>ซ่อนตัวละครนักเรียน</p> <p>เมื่อได้รับสาร "action1" แสดงตัวละครนักเรียน เปลี่ยนชุดตัวละครเป็น champ99-a</p>
<p>เวที</p> 	<pre> when clicked switch backdrop to room3 when I receive action1 switch backdrop to school2 </pre>	<p>เมื่อคลิก </p> <p>เปลี่ยนฉากเป็น room3</p> <p>เมื่อได้รับสาร "action1" เปลี่ยนฉากเป็น school2</p>

ตัวละครใด/เวที เป็นผู้ส่งสาร

ตัวละครใด/เวที เป็นผู้รับสาร

สารที่กระจายคือคำว่าอะไร

สารที่รับคือคำว่าอะไร

ใช้เหตุการณ์ใดบ้างที่ทำให้บทสนทนาทำงาน

5 บันทึกไฟล์ชื่อ activity5_1.sb

ใบงานที่ 5.2 สกนนาพาพลิน



ชื่อ-สกุล เลขที่

- 1 อ่านบทสนทนาด้านล่าง แล้วออกแบบโครงร่างตามข้อ 2

ภาพ	บทสนทนา
	<p>เมื่อเราคลิก  จะปรากฏม้าและสิงโตอยู่ในป่าด้วยกัน แล้วสิงโตพูด : “ สิ่งโตมาแล้วจ้า ”</p>
	<p>ม้าคิดในใจว่า : “ สิงโต ใจดี เราไม่กลัว สิงโต ”</p>
	<p>สิงโตคำรามใส่ม้า แล้วพูดว่า : “ ฮ่า !!! ”</p>
	<p>ม้าตกใจ แล้วพูดว่า : “ จิ้ง!!! ม้ากลัวสิงโต ”</p>

ภาพ	บทสนทนา
	<p>จากนั้นมันก็วิ่งหายไป</p>

2) ออกแบบโครงร่างบทสนทนา เพื่อนำไปเขียนสคริปต์ให้กับตัวละคร เวที

ตัวละคร/เวที	เหตุการณ์ เมื่อคลิก 	เหตุการณ์ เมื่อคลิก 

3) เขียนสคริปต์ ตามที่ได้ออกแบบไว้

4) บันทึกไฟล์ชื่อ activity5_2.sb

ใบความรู้ที่ 5.1 การกระจายสารและรับสาร



จากภาพด้านบนจะสังเกตได้ว่า ผู้กระจายสารคือ  ซึ่งสังเกตได้จาก `broadcast walk`
 โดยกระจายสารว่า "walk" และมีผู้รับสารคือ  ซึ่งสังเกตได้จาก `when I receive walk`
 เมื่อได้รับสาร แล้วจะทำตามคำสั่งที่อยู่ใต้บล็อก `when I receive walk` ในที่นี้คือ `next costume`

กรณีที่ต้องการให้มีผู้รับสารเดียวกันหลายๆ ตัวละคร
 ให้ใช้บล็อก `when I receive` รับสารชื่อเดียวกันที่ตัวละครนั้นๆ



การใช้บล็อก **broadcast** เพื่อกระจายสาร

- 1) เลือกบล็อก `broadcast` มาวางในสคริปต์ของตัวละครที่ต้องการกระจายสาร
- 2) คลิกที่ลูกศรหลัง `message1` หลังจากนั้นเลือก `broadcast message1`
`message1`
`new message...`
- 3) พิมพ์ข้อความที่ต้องการกระจายสาร เช่น คำว่า "walk" และคลิกปุ่ม `OK`

New Message

Message Name:

การใช้บล็อก **when I receive** เพื่อรับสาร

- 1) เลือกบล็อก **when I receive** มาวางในสคริปต์ของตัวละครที่ต้องการให้รับสาร
- 2) คลิกที่ลูกศรหลัง message1 หลังจากนั้นเลือกสารที่ต้องการรับ
- 3) เพิ่มสคริปต์เพื่อให้เวทีหรือตัวละครทำงานหลังจากได้รับสารแล้ว เช่น

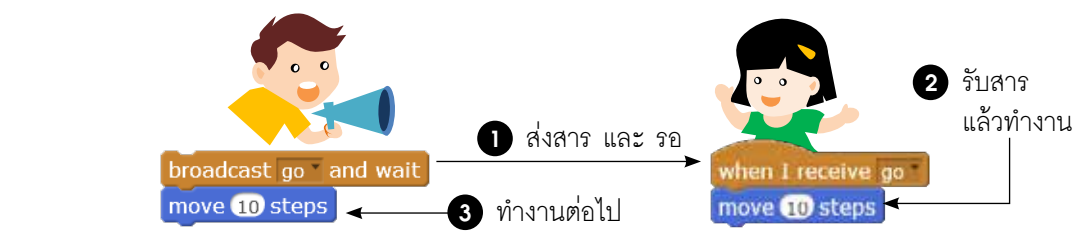
```
when I receive walk
  next costume
```

แล้ว **broadcast** และ **broadcast and wait** ต่างกันอย่างไร? 

broadcast คือ การกระจายสารโดยเมื่อกระจายสารแล้วผู้ส่งสารก็ยังคงทำงานของตนเองต่อไป



broadcast and wait คือ การกระจายสารโดยเมื่อกระจายสารแล้วผู้ส่งสารจะรอให้ผู้รับสารทำงานให้เสร็จก่อนแล้วผู้ส่งสารจึงจะทำงานต่อไป



ถ้ามีผู้รับสารมากกว่า 1 คน ผู้กระจายสารด้วย **broadcast and wait** จะต้องรอจนผู้รับสารทั้งหมดทำงานเสร็จ จึงจะทำงานต่อไป 

บัตรคำ when I receive



when I receive action1

ทำตามตามรูป



when I receive action2

ทำตามตามรูป



when I receive action3

ทำตามตามรูป



when I receive action4

ทำตามตามรูป



when I receive action5

ทำตามตามรูป



when I receive action6

ทำตามตามรูป



บัตรคำ broadcast



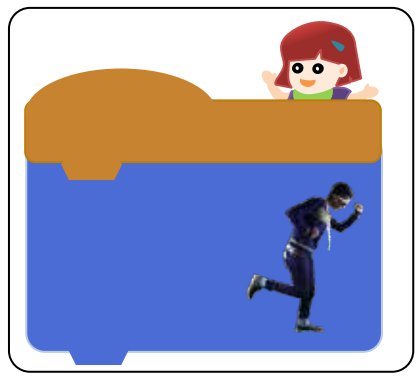
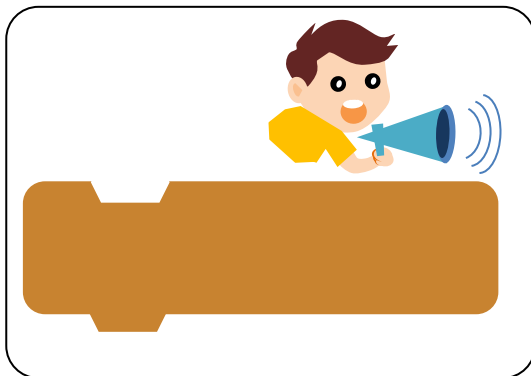
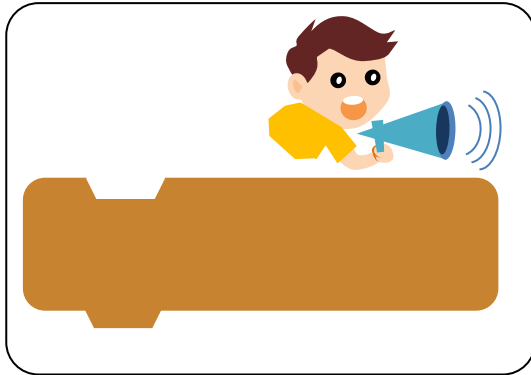
ตัวอย่างการสรุปเกมกระจายสาร

ครู (ผู้กระจายสาร)



นักเรียน (ผู้รับสาร)





การกระจายสาร () จะถูกกระจายจากครู แล้วให้นักเรียนซึ่งเป็นผู้รับสาร () หากข้อสารตรงกับสารที่ครูส่ง นักเรียนต้องปฏิบัติตามคำสั่งในสารนั้น

กิจกรรมที่ 6 มุมและทิศทาง

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรมและการวาดเส้น
- 1.2 เข้าใจมุมและทิศทางของเส้นตรง
- 1.3 เข้าใจการหมุนและลากเส้นไปยังทิศทางที่ต้องการ

2. สารการเรียนรู้

- 2.1 มุมและทิศทาง
- 2.2 กลุ่มบล็อก Pen

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 สร้างชิ้นงานโดยใช้คำสั่งในกลุ่มบล็อก Pen
- 3.2 สร้างรูปเรขาคณิตเบื้องต้น

4. แนวคิด

กลุ่มบล็อก Pen ใช้ในการวาดรูป และกลุ่มบล็อก Motion ใช้เพื่อควบคุมทิศทางการเคลื่อนที่ของปากกา ในกิจกรรมนี้ เน้นให้ผู้เรียนใช้งานปากกาได้ เช่น การวางปากกา ณ ตำแหน่งที่ต้องการ การลากเส้นไปในทิศทางที่ต้องการ และการยกปากกาเพื่อหยุดการวาดรูป

5. สื่ออุปกรณ์

- 5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
6.1	เส้นตรงหลงทาง	10
6.2	มุมและทิศทาง	10

5.2 ใบความรู้

- ใบความรู้ที่ 6.1 เรื่อง มุมและทิศทาง

5.3 อื่นๆ

-

6. วิธีดำเนินการ

6.1 การจัดเตรียม

6.1.1 ใบงานที่ 6.1 ใบงานที่ 6.2 ตามจำนวนผู้เรียน

6.1.2 ใบความรู้ที่ 6.1 ตามจำนวนผู้เรียน

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนกล่าวถึงความสำคัญในการทำความเข้าใจเรื่องมุมและทิศทางในโปรแกรม Scratch

6.2.2 ผู้เรียนศึกษาใบความรู้ที่ 6.1 เรื่อง มุมและทิศทาง

6.2.3 ผู้เรียนทำใบงานที่ 6.1 เรื่อง เส้นตรงหลงทาง หลังจากนั้นผู้เรียนและผู้สอนร่วมกันเฉลยใบงาน

6.2.4 ผู้เรียนทำใบงานที่ 6.2 มุมและทิศทาง

6.2.5 ผู้สอนสรุปแนวคิดเกี่ยวกับทิศทาง การหมุนของปากกา และการวาดรูปเรขาคณิตเบื้องต้น เช่น รูปสามเหลี่ยม หรือ รูปสี่เหลี่ยม

7. การวัดและประเมินผล

7.1 ตรวจคำตอบในใบงาน

7.2 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

-

9. ข้อเสนอแนะ

9.1 ผู้สอนควรทบทวนเรื่องทิศทาง ก่อนการทำใบงานที่ 6.1 และใบงานที่ 6.2

9.2 ผู้สอนควรเน้นให้ผู้เรียนเข้าใจวิธีการเคลื่อนที่ของปากกาและการหมุน ซึ่งจะไม่เท่ากับมุมระหว่างส่วนของเส้นตรง

ใบงานที่ 6.1 เส้นตรงสองทาง

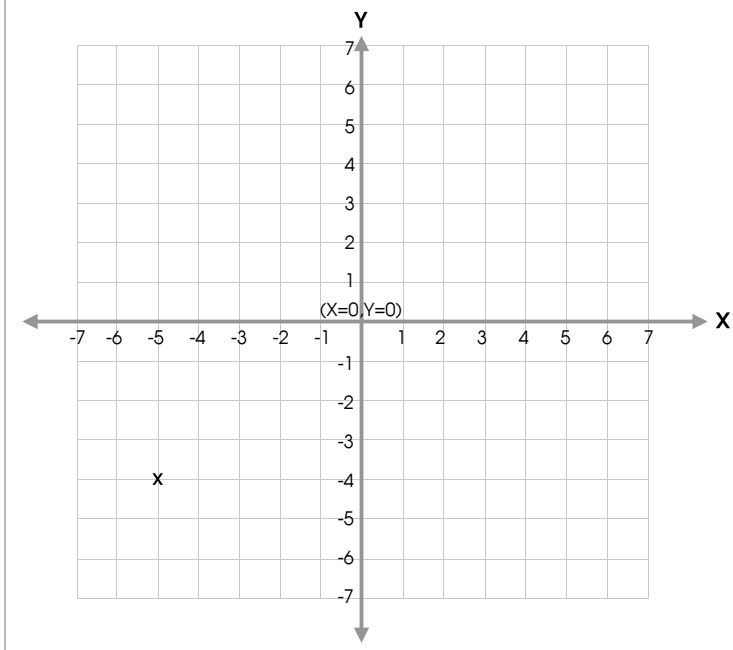
ชื่อ-สกุล เลขที่

1) ทำเครื่องหมายกากบาท ณ ตำแหน่งพิกัดที่ระบุ

ตัวอย่าง:

ตำแหน่ง $x = -5, y = -4$

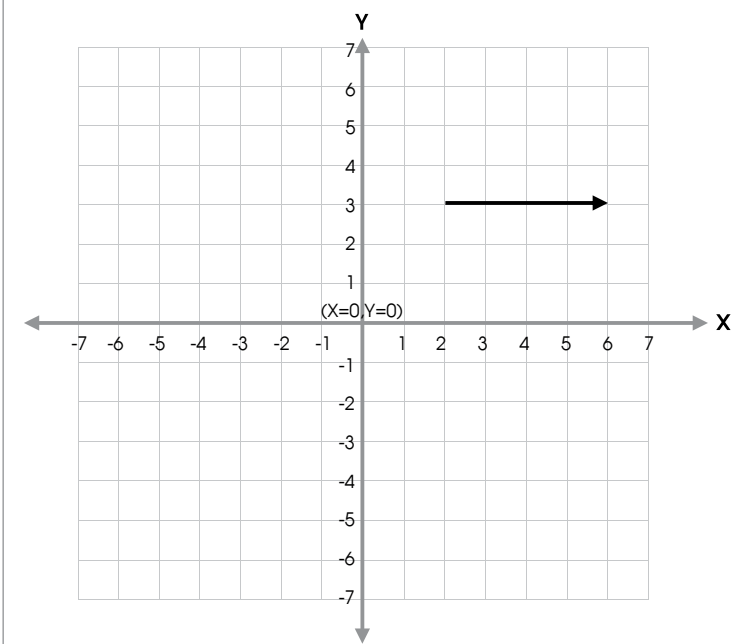
- 1) ตำแหน่ง $x = 0, y = 5$
- 2) ตำแหน่ง $x = -4, y = 0$
- 3) ตำแหน่ง $x = 2, y = 3$
- 4) ตำแหน่ง $x = 5, y = 4$
- 5) ตำแหน่ง $x = 3, y = -3$
- 6) ตำแหน่ง $x = -5, y = -1$



2) วาดส่วนของเส้นตรงตามใจทย์ (ทิศทางตามทิศของโปรแกรม Scratch)

ตัวอย่าง: เส้นตรงยาว 4 หน่วย
เริ่มต้นที่ตำแหน่ง $x=2, y=3$ ไปทาง
ทิศ 90 องศา

- 1) เส้นตรงยาว 3 หน่วยเริ่มต้นที่
ตำแหน่ง $x=1, y=0$ ไปทางทิศ
0 องศา
- 2) เส้นตรงยาว 4 หน่วยเริ่มต้นที่
ตำแหน่ง $x = -3, y = -3$ ไปทางทิศ
90 องศา
- 3) เส้นตรงยาว 4 หน่วยเริ่มต้นที่
ตำแหน่ง $x = -5, y = 2$ ไปทางทิศ
45 องศา





3) ศึกษาการทำงานจากคำสั่งต่อไปนี้ แล้ววาดรูปผลลัพธ์

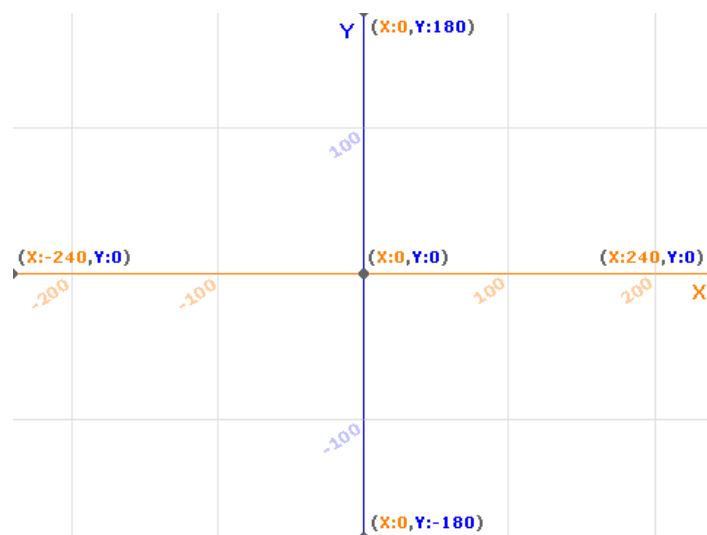
3.1 แนวคิด

- 1) ซ่อนตัวละคร
- 2) ล้างภาพวาด
- 3) กำหนดตำแหน่ง $x = 50$, $y = -50$
- 4) กำหนดทิศทางเป็น 90
- 5) วางปากกา
- 6) เดินหน้า 100 หน่วย
- 7) หมุนตามเข็มนาฬิกา 90 องศา
- 8) เดินหน้า 100 หน่วย
- 9) ยกปากกา

สคริปต์ในภาษา Scratch สำหรับแนวคิดทางซ้ายมือ

-
- 1) hide
 - 2) clear
 - 3) go to x: 50 y: -50
 - 4) point in direction 90
 - 5) pen down
 - 6) move 100 steps
 - 7) turn 90 degrees
 - 8) move 100 steps
 - 9) pen up

วาดรูปผลลัพธ์ที่ได้



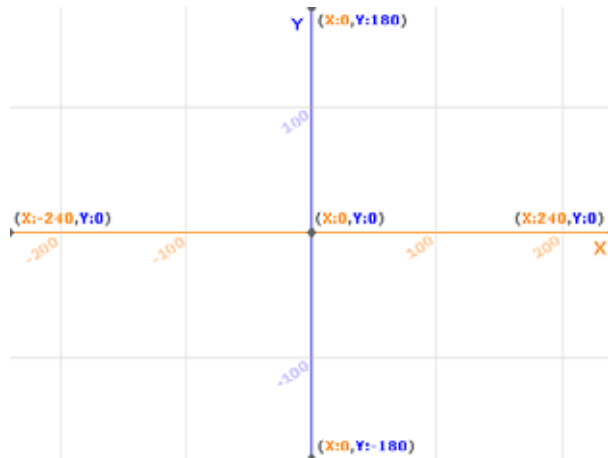


3.2

แนวคิด

- 1) ช้อนตัวละคร
- 2) ล้างภาพวาด
- 3) กำหนดตำแหน่ง $x=0, y=0$
- 4) กำหนดทิศทางเป็น 90
- 5) วางปากกา
- 6) เดินหน้า 100 หน่วย
- 7) หมุนตามเข็มนาฬิกา 120 องศา
- 8) เดินหน้า 100 หน่วย
- 9) หมุนตามเข็มนาฬิกา 120 องศา
- 10) เดินหน้า 100 หน่วย
- 11) ยกปากกา

ให้วาดรูปผลลัพธ์จากโปรแกรม Scratch ของแนวคิดทางด้านซ้ายมือ



ใบงานที่ 6.2 มุมและทิศทาง

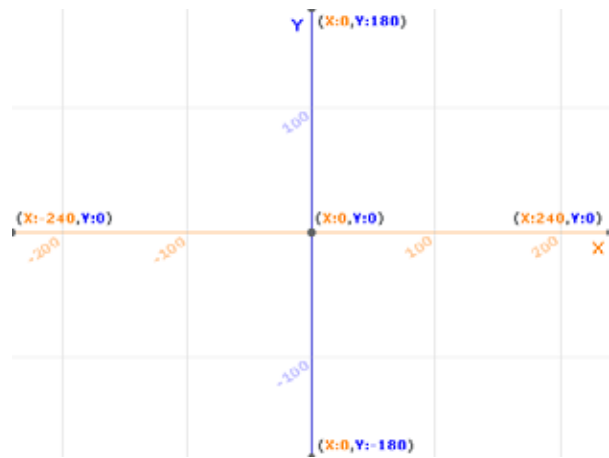
ชื่อ-สกุล เลขที่

1) ศึกษาใบความรู้ที่ 6.1 มุมและทิศทาง แล้วศึกษาการทำงานจากสคริปต์ด้านล่าง แล้วตอบคำถาม

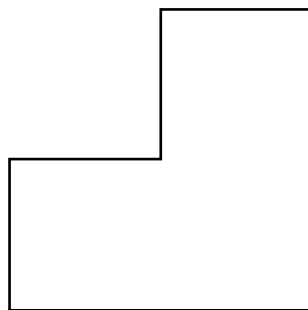
1.1

- 1) กำหนดตำแหน่ง $x = -100$, $y = -100$
- 2) กำหนดทิศทางเป็น 0
- 3) วางปากกา
- 4) เดินหน้า 100 หน่วย
- 5) หมุนตามเข็มนาฬิกา 90 องศา
- 6) เดินหน้า 100 หน่วย
- 7) หมุนทวนเข็มนาฬิกา 90 องศา
- 8) เดินหน้า 100 หน่วย
- 9) หมุนตามเข็มนาฬิกา 90 องศา
- 10) เดินหน้า 100 หน่วย

1.2 วาดรูปผลลัพธ์จากสคริปต์ด้านซ้ายมือ



1.3 หากต้องการให้ผลลัพธ์ที่ได้เป็นดังรูปด้านล่างทางขวา จะต้องเพิ่มเติมคำสั่งใดบ้าง



2 จากสคริปต์การทำงานด้านล่างทางซ้าย ให้ตอบคำถามข้อ 2.1 – 2.2

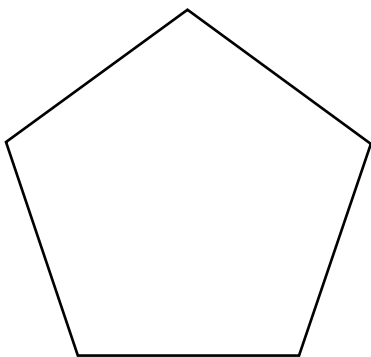
- 1) ซ่อนตัวละคร
- 2) กำหนดตำแหน่ง $x = 0$, $y = 0$
- 3) กำหนดทิศทางเป็น 90
- 4) วางปากกา
- 5) เดินหน้า 100 หน่วย
- 6) หมุนตามเข็มนาฬิกา 120 องศา
- 7) เดินหน้า 100 หน่วย
- 8) หมุนตามเข็มนาฬิกา 120 องศา
- 9) เดินหน้า 100 หน่วย
- 10) ยกปากกา

2.1 ถ้าต้องการลดขนาดรูปสามเหลี่ยมให้เหลือด้านละ 50 หน่วย ต้องแก้คำสั่งในบรรทัดใดบ้างอย่างไร

2.2 ถ้าต้องการขยายขนาดรูปสามเหลี่ยมให้เป็นด้านละ 200 หน่วย ต้องแก้คำสั่งในบรรทัดใดบ้างอย่างไร

3 ลองคิดว่าถ้าจะวาดรูปห้าเหลี่ยมด้านเท่ามุมเท่าจะเขียนแนวคิดอย่างไร

รูปห้าเหลี่ยมด้านเท่ามุมเท่า



แนวคิด



4 เติมช่องว่างให้สมบูรณ์ เพื่อให้โปรแกรมวาดรูปที่ต้องการ

ผลลัพธ์ที่ต้องการ	สคริปต์	อธิบาย
	<p>ตัวอย่าง</p> <ol style="list-style-type: none"> กำหนดตำแหน่ง $x=50$, $y=50$ กำหนดทิศทางเป็น 90 องศา วางปากกา เดินหน้า 100 หน่วย หมุน ทวน เข็มนาฬิกา เป็นมุม 120 องศา เดินหน้า 100 หน่วย ยกปากกา 	<p>เส้นตรงสองเส้นทำมุม 180 องศา ถ้าเราต้องการสร้างเส้นมุม 60 คำนวณโดยนำ $180-60$ จะได้มุมที่ใช้หมุนทวนเข็มนาฬิกา 120 องศา</p>

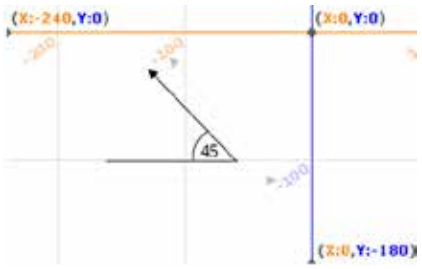
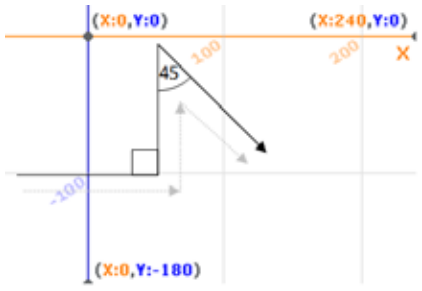
ผลลัพธ์ที่ต้องการ	สคริปต์
	<p>4.1</p> <ol style="list-style-type: none"> กำหนดตำแหน่ง $x=50$, $y=50$ กำหนดทิศทางเป็น 90 องศา (ขวา) วางปากกา เดินหน้า 100 หน่วย หมุน เข็มนาฬิกา เป็นมุม เดินหน้า 100 หน่วย ยกปากกา
	<p>4.2</p> <ol style="list-style-type: none"> กำหนดตำแหน่ง $x=50$, $y=125$ กำหนดทิศทางเป็น 90 องศา (ขวา) วางปากกา เดินหน้า 100 หน่วย หมุน เข็มนาฬิกา เป็นมุม เดินหน้า 100 หน่วย ยกปากกา

ผลลัพธ์ที่ต้องการ	สคริปต์
	<p>4.3</p> <ol style="list-style-type: none"> 1) กำหนดตำแหน่ง $x=-20, y=50$ 2) กำหนดทิศทางเป็น -90 องศา (ซ้าย) 3) วางปากกา 4) เดินหน้า 100 หน่วย 5) หมุน เจริมนาฬิกา เป็นมุม 6) เดินหน้า 100 หน่วย 7) หมุน เจริมนาฬิกา เป็นมุม 8) เดินหน้า 100 หน่วย

5) ศึกษาการทำงานจากสคริปต์ด้านล่าง แล้วตอบคำถาม

<ol style="list-style-type: none"> 1 clear 2 go to x: -50 y: 0 3 point in direction -90 4 pen down 5 move 100 steps 6 turn ↺ 120 degrees 7 move 100 steps 8 turn ↺ 120 degrees 9 move 100 steps 10 turn ↺ 120 degrees 11 pen up 12 go to x: 150 y: 0 13 pen down 14 move 100 steps 15 turn ↺ 120 degrees 16 move 100 steps 17 turn ↺ 120 degrees 18 move 100 steps 19 pen up 	<p>5.1 วาดรูปผลลัพธ์ที่ได้จากสคริปต์ด้านซ้ายมือ</p> <p>5.2 ถ้าลบคำสั่ง pen up (บรรทัดที่ 11) ออก แล้ววาดรูปผลลัพธ์ที่ได้คือ</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------

- 6 เขียนสคริปต์ให้ตัวละครวาดภาพตามผลลัพธ์ที่กำหนด (ไม่ต้องวาดรูปลูกศร และมุม) แล้วเติมช่องว่างในแนวคิดให้สมบูรณ์

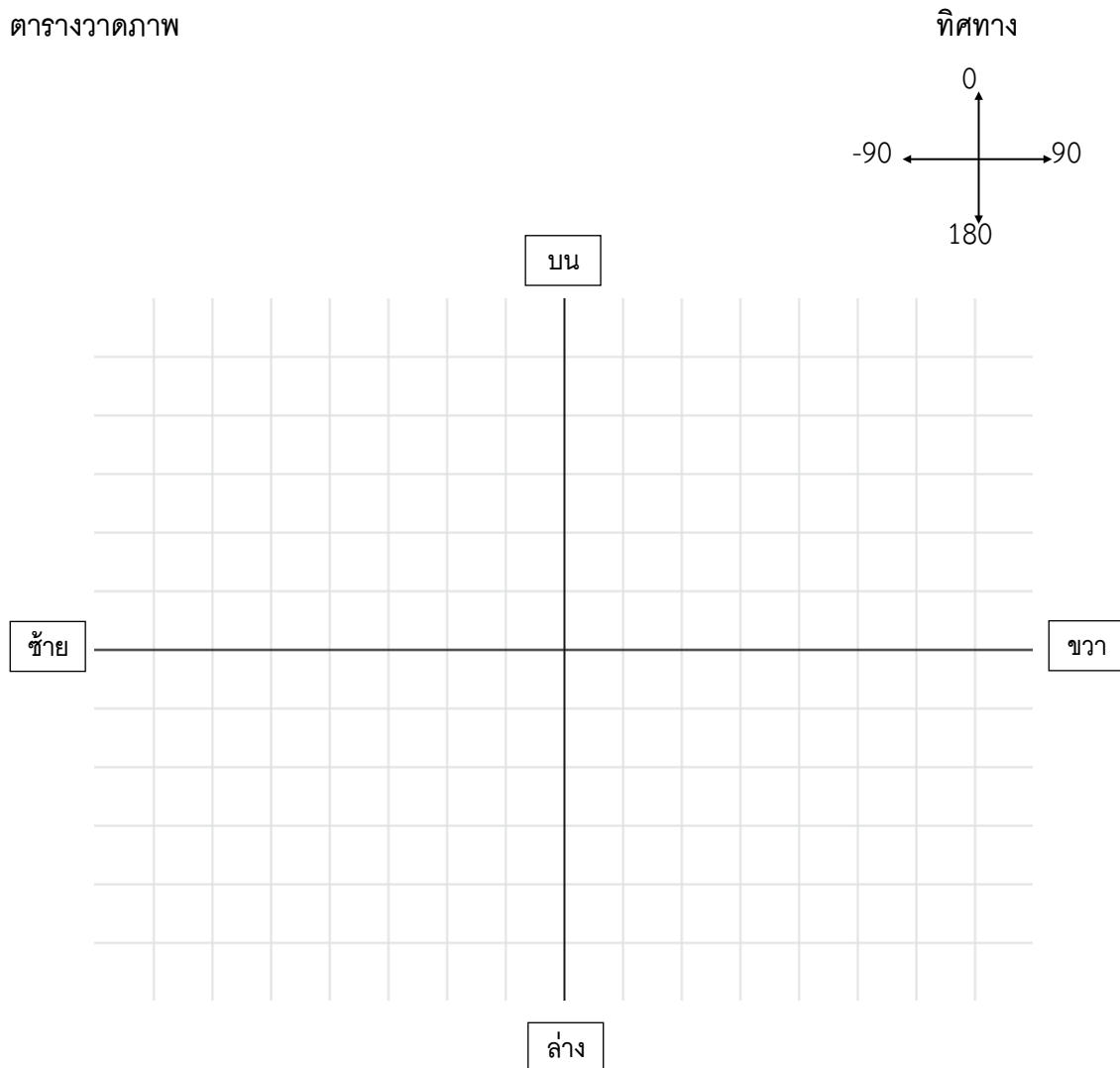
ผลลัพธ์	แนวคิด
	<p>6.1</p> <ol style="list-style-type: none"> 1) กำหนดตำแหน่ง $x=-175, y=-100$ 2) 3) วางปากกา 4) เดินหน้า 100 หน่วย 5) หมุน เข็มนาฬิกา เป็นมุม 6) เดินหน้า 100 หน่วย 7) ยกปากกา
	<p>6.2</p> <ol style="list-style-type: none"> 1) กำหนดตำแหน่ง $x=-50, y=-100$ 2) 3) วางปากกา 4) เดินหน้า 100 หน่วย 5) หมุน เข็มนาฬิกา เป็นมุม 6) เดินหน้า 100 หน่วย 7) หมุน เข็มนาฬิกา เป็นมุม 8) เดินหน้า 100 หน่วย 9) ยกปากกา

ใบความรู้ที่ 6.1 มุมและทิศทาง

ทิศทาง

การอ้างอิงทิศทางในโปรแกรม Scratch จะหมายถึงค่าตัวเลขที่แสดงถึงองศา ทิศทางสามารถมีค่าเป็นได้ทั้งบวกและลบ โดยค่าจะเพิ่มขึ้นตามแนวเข็มนาฬิกา ตัวอย่างเช่น ทิศตะวันตกคือ ทิศ -90 องศา (หันหน้าไปทางซ้าย), ทิศเหนือคือทิศ 0 องศา (หันหน้าไปด้านบน), ทิศตะวันออกคือ ทิศ 90 องศา (หันหน้าไปทางขวา), และทิศใต้คือทิศ 180 องศา (หันหน้าไปด้านล่าง) เป็นต้น การใช้คำสั่งวางปากกา (pen down) ร่วมกับคำสั่งเคลื่อนที่ (move) และคำสั่งหมุน (turn) จะทำให้เราสามารถวาดรูปเรขาคณิตต่างๆ ตามที่ต้องการได้

ตารางวาดภาพ



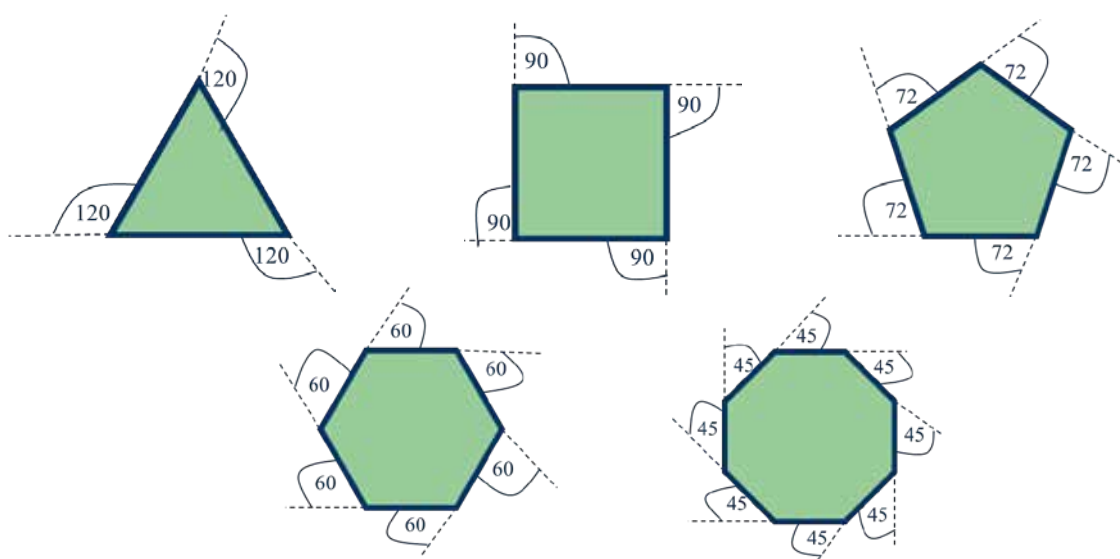
ตัวอย่างการวาดภาพสามเหลี่ยมด้านเท่าด้วยโปรแกรม Scratch

สคริปต์	คำอธิบายคำสั่ง	ภาพที่เกิดขึ้น
<pre> hide clear go to x: 0 y: 0 point in direction 90 pen down move 100 steps turn 120 degrees move 100 steps turn 120 degrees move 100 steps pen up </pre>	<ol style="list-style-type: none"> 1) ซ่อนตัวละครจากเวที 2) ลบรอยปากกาทั้งหมดจากเวที 3) เริ่มต้นที่ตำแหน่ง 0, 0 4) เริ่มต้นหันไปที่ทิศทาง 90 องศา (ขวา) 5) วางปากกากลาง เพื่อเริ่มต้นการวาด 6) เดินหน้า 100 หน่วย 7) หมุนตามเข็มนาฬิกา 120 องศา 8) เดินหน้า 100 หน่วย 9) หมุนตามเข็มนาฬิกา 120 องศา 10) เดินหน้า 100 หน่วย 11) ยกปากกาขึ้น 	

มุมของรูปหลายเหลี่ยมด้านเท่ามุมเท่า




Scratch จะใช้มุมภายนอกในการหมุนปากกาเพื่อสร้างรูปหลายเหลี่ยมใดๆ โดยมีหลักการว่า ผลบวกของมุมภายนอกจากการหมุนปากกาเพื่อให้ได้รูปหลายเหลี่ยมใดๆ มีค่าเท่ากับ 360 องศา และมุมภายนอกของรูปหลายเหลี่ยมด้านเท่ามุมเท่า ทุกมุมมีค่าเท่ากัน เราสามารถนำ 360 หารด้วยจำนวนมุม เช่น วาดรูปสามเหลี่ยมด้านเท่า $360/3 = 120$ องศา 120 องศา คือ มุมองศาที่จะใช้สร้างรูปสามเหลี่ยมด้านเท่า

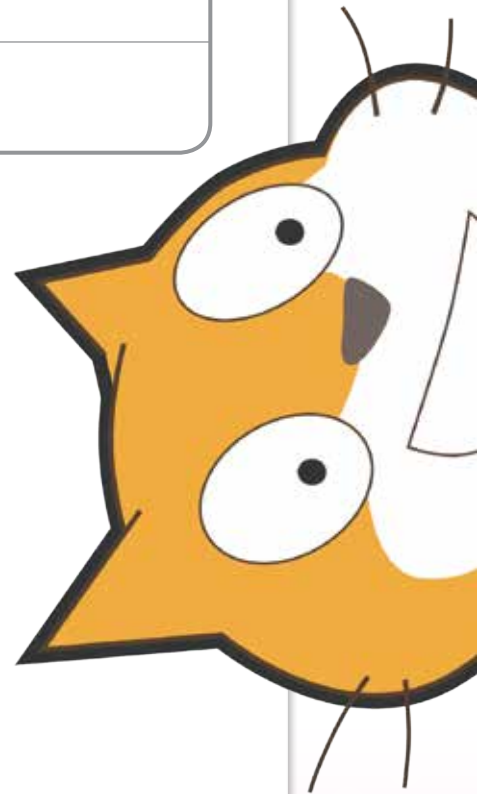
ตัวอย่าง แสดงมุมภายนอกของรูปหลายเหลี่ยมด้านเท่ามุมเท่า



คำสั่งพื้นฐาน

กลุ่มบล็อก Pen ซึ่งเปรียบเสมือนปากกาที่ช่วยในการวาดรูปต่างๆ ในกิจกรรมนี้จะใช้บล็อกในกลุ่มบล็อก Pen ดังนี้

กลุ่มบล็อก Pen	ความหมาย
 clear	ล้างภาพวาด
 pen down	วางปากกา
 pen up	ยกปากกา



กิจกรรมที่ 7 ภาพสวยด้วยมือเรา

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการวาดรูป
- 1.2 ใช้โครงสร้างแบบลำดับ เงื่อนไข และวนซ้ำ
- 1.3 ใช้คำสั่งในการสร้างฟังก์ชัน

2. สารการเรียนรู้

- 2.1 บล็อก Repeat
- 2.2 กลุ่มบล็อก Pen
- 2.3 บล็อก Pick Random
- 2.4 บล็อก Make a Block

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 ใช้บล็อก Repeat ในการสร้างรูปทรงเรขาคณิตเบื้องต้น
- 3.2 สร้างชิ้นงานโดยใช้คำสั่งในกลุ่มบล็อก Pen
- 3.3 สร้างและเรียกใช้ฟังก์ชัน

4. แนวคิด

กลุ่มบล็อก Pen ใช้ในการวาดรูปโดยสามารถกำหนดคุณลักษณะต่างๆ ของปากกาได้ เช่น การวางหรือยกปากกา ขนาดและสีปากกา สร้างการประทับตราให้ตัวละคร

บล็อก Repeat เป็นบล็อกคำสั่งที่ใช้ในการทำซ้ำแบบระบุจำนวนรอบ ช่วยให้โปรแกรมมีขนาดเล็กลง

ในการเขียนโปรแกรมที่มีขนาดใหญ่ ควรจะแบ่งการทำงานย่อยเป็นฟังก์ชัน เพื่อให้ง่ายต่อการทำความเข้าใจ การแก้ไข และการเรียกใช้งานภายหลัง ใน scratch มีบล็อก Make a Block ที่ใช้ในการสร้างฟังก์ชัน



5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
7.1	ลากเส้นเล่นลาย	20
7.2	ฟังก์ชัน	20
7.3	ตัวปั๊ม	20

5.2 ใบความรู้

- ใบความรู้ที่ 7.1 เรื่อง การสั่งงานแบบวนซ้ำ
- ใบความรู้ที่ 7.2 เรื่อง ฟังก์ชัน

5.3 อื่นๆ

- กระดาษต้นแบบ

6. วิธีดำเนินการ

6.1 การจัดเตรียม

- 6.1.1 ใบงานที่ 7.1 ตามจำนวนผู้เรียน
- 6.1.2 ใบความรู้ที่ 7.1 ตามจำนวนผู้เรียน
- 6.1.3 กระดาษต้นแบบ ตามจำนวนกลุ่ม

6.2 ขั้นตอนการดำเนินการ

- 6.2.1 ผู้สอนทบทวนเรื่องมุมและทิศทาง
- 6.2.2 ผู้เรียนทำใบงานที่ 7.1 เรื่อง ลากเส้นเล่นลาย ข้อ 1
- 6.2.3 ผู้เรียนศึกษาใบความรู้ที่ 7.1 เรื่อง การสั่งงานแบบวนซ้ำ แล้วทำใบงานที่ 7.1 ข้อ 2
- 6.2.4 ผู้สอนสรุปแนวคิดคำสั่ง Repeat
- 6.2.5 ผู้เรียนทำใบงานที่ 7.2 เรื่อง ฟังก์ชัน ข้อที่ 1
- 6.2.6 ให้ผู้เรียนศึกษาใบความรู้ที่ 7.2 เรื่อง ฟังก์ชัน แล้วทำใบงานที่ 7.2 ที่เหลือ
- 6.2.7 ผู้สอนแบ่งผู้เรียนออกเป็นกลุ่ม กลุ่มละ 4 คน ตรวจสอบคำตอบจากการทำใบงาน
- 6.2.8 ให้ผู้เรียนแต่ละคนทำใบงานที่ 7.3 ตัวปั๊ม หลังจากผู้สอนสุ่มผู้เรียนนำเสนอ
- 6.2.9 ผู้เรียนและผู้สอนสรุปการใช้คำสั่ง Repeat Make a block Stamp

7. การวัดและประเมินผล

- 7.1 ตรวจสอบใบงานและผลงาน
- 7.2 สังเกตการร่วมกิจกรรมในการระดมความคิดของผู้เรียน
- 7.3 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

-

9. ข้อเสนอแนะ

- 9.1 การวาดรูปวงกลมในคอมพิวเตอร์ เป็นการวาดรูปหลายเหลี่ยมที่มีเหลี่ยมจำนวนมากทำให้มองเห็นเป็นวงกลม
- 9.2 ผู้สอนอาจให้ผู้เรียนฝึกทักษะโดยทำแบบฝึกหัดเพิ่มเติมเป็นการบ้าน



ใบงานที่ 7.1 ลากเส้นเล่นลาย

ชื่อ-สกุล เลขที่

1) ศึกษาการทำงานจากคำสั่งต่อไปนี้ แล้วตอบคำถาม

ผลลัพธ์ที่ต้องการ	สคริปต์
1.1 <ol style="list-style-type: none"> 1) กำหนดตำแหน่ง $x = 0, y = 0$ 2) กำหนดทิศทางเป็น 90 3) วางปากกา 4) เดินหน้า 100 หน่วย 5) หมุนตามเข็มนาฬิกา 90 องศา 6) เดินหน้า 100 หน่วย 7) หมุนตามเข็มนาฬิกา 90 องศา 8) เดินหน้า 100 หน่วย 9) หมุนตามเข็มนาฬิกา 90 องศา 10) เดินหน้า 100 หน่วย 11) หมุนตามเข็มนาฬิกา 90 องศา 12) ยกปากกา 	1.2 <ol style="list-style-type: none"> 1) กำหนดตำแหน่ง $x = 0, y = 0$ 2) กำหนดทิศทางเป็น 90 3) วางปากกา 4) ทำข้อ 4.1 และ 4.2 ซ้ำ 4 รอบ <ol style="list-style-type: none"> 4.1) เดินหน้า 100 หน่วย 4.2) หมุนตามเข็มนาฬิกา 90 องศา 5) ยกปากกา
วาดรูปผลลัพธ์ที่ได้ 	วาดรูปผลลัพธ์ที่ได้

1.3 ข้อสรุปจากการทำงานในข้อ 1.1 และ 1.2 เหมือนหรือต่างกันอย่างไร

.....

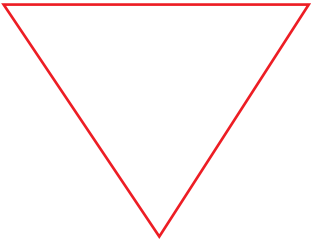
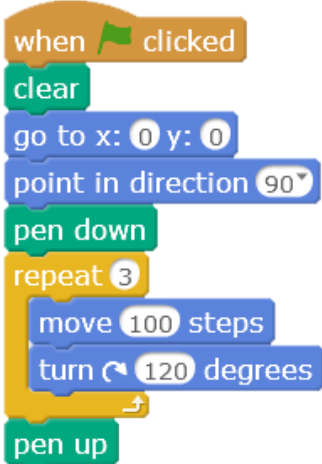
.....

.....

.....

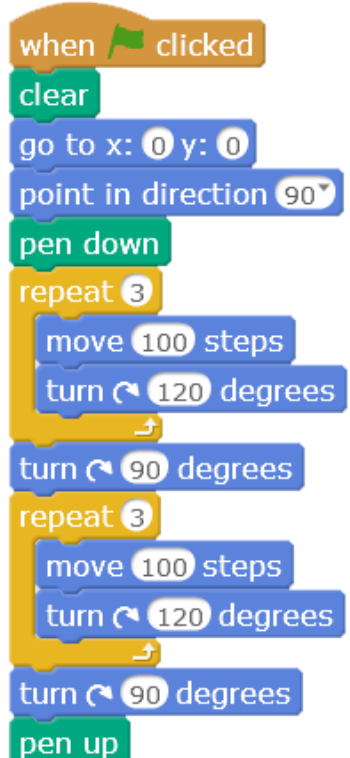
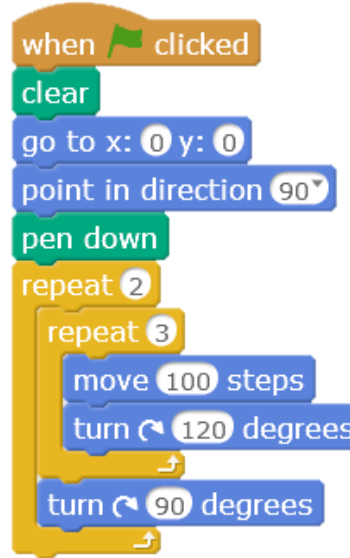
.....

2) สร้างโปรเจกต์ชื่อ Art1
วาดรูปสามเหลี่ยม โดยเขียนสคริปต์ดังนี้

ผลลัพธ์	แนวคิด	สคริปต์
<p>2.1 สามเหลี่ยม 1 รูป</p> 	<ol style="list-style-type: none"> 1) กำหนดพิกัด $x = 0, y = 0$ 2) กำหนดทิศทางเป็น 90 3) วางปากกา 4) วาดซ้ำ 3 รอบ <ol style="list-style-type: none"> 4.1) เดินหน้า 100 หน่วย 4.2) หมุนตามเข็มนาฬิกา 120 องศา 5) ยกปากกา 	



2.2 เขียนสคริปต์ตามตัวอย่างต่อไปนี้

แบบที่ 1	แบบที่ 2
<p>1)</p>  <pre> when clicked clicked clear go to x: 0 y: 0 point in direction 90 pen down repeat 3 move 100 steps turn 120 degrees turn 90 degrees repeat 3 move 100 steps turn 120 degrees turn 90 degrees pen up </pre>	<p>2)</p>  <pre> when clicked clicked clear go to x: 0 y: 0 point in direction 90 pen down repeat 2 repeat 3 move 100 steps turn 120 degrees turn 90 degrees </pre>
<p>ผลลัพธ์</p>	<p>ผลลัพธ์</p>



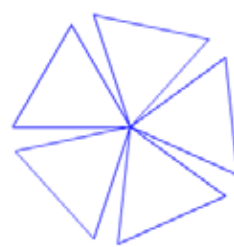
รู้หรือไม่ว่า...

เพียงแค่เรานำ 360 หารด้วยจำนวนรูปที่เราต้องการวาด เราก็จะได้ขนาดของมุมที่เราต้องหมุนไป เช่น ต้องการวาดสามเหลี่ยม 5 รูปขนาดของมุมที่เราต้องหมุนไปจะเท่ากับ $360 \div 5 = 72$ องศา

```

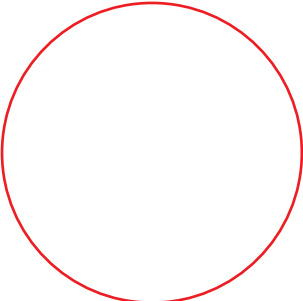
repeat 5
  repeat 3
    move 100 steps
    turn 120 degrees
    turn 72 degrees
  
```

- 1) วาดรูปสามเหลี่ยม 1 รูป
- 2) วนซ้ำ 5 ครั้ง (วาดรูปสามเหลี่ยมอีก 5 ครั้ง โดยในแต่ละรอบ หมุนไป 72 องศา)



มุมที่ต้องหมุน = 72 องศา

2.2 เขียนสคริปต์ตามตัวอย่างต่อไปนี้

ผลลัพธ์	แนวคิด	สคริปต์
<p>วงกลม</p> 	<ol style="list-style-type: none"> 1) กำหนดพิกัด $x = 100, y = 80$ 2) ลากเส้นยาว 5 หน่วย 3) หมุน 10 องศา 4) ทำซ้ำข้อ 2 และ 3 จำนวน 36 ครั้ง (หมุนครบ 360 องศา) ได้รูปวงกลม 	<pre> clear go to x: 100 y: 80 pen down repeat 36 move 5 steps turn 10 degrees pen up </pre>

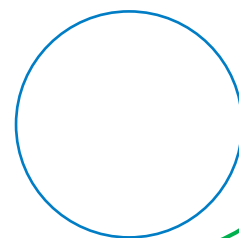


การวาดรูปวงกลม

วงกลมวาดจริงๆ ไม่ได้หรอกนะ แต่ว่าต้องวาดเป็นรูปหลายเหลี่ยมโดยจำนวนเหลี่ยมมีจำนวนมากพอก็จะมองเห็นเป็นวงกลมมากขึ้น โดยมีแนวคิดดังนี้

วาดรูป 360 เหลี่ยม

- 1) ลากเส้น 2 หน่วย
- 2) หมุน 1 องศา
- 3) ทำซ้ำข้อ 1 และ 2 จนครบ 360 องศา



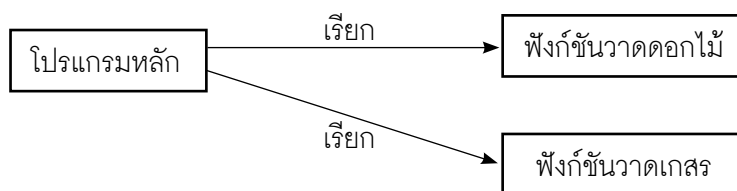
ใบงานที่ 7.2 ฟังก์ชัน

ชื่อ-สกุล เลขที่

- 1 ให้ผู้เรียนเขียนสคริปต์วาดดอกไม้และเกสร โดยใช้กลุ่มบล็อก Pen สคริปต์ในการวาดดอกไม้และเกสรมีดังนี้





- 2 ให้ผู้เรียนศึกษาใบความรู้ที่ 7.2 เรื่อง ฟังก์ชัน
- 3 จากข้อ 1 ให้ผู้เรียนปรับสคริปต์ โดยสร้างเป็นฟังก์ชัน ให้มีโครงการทำงานดังนี้

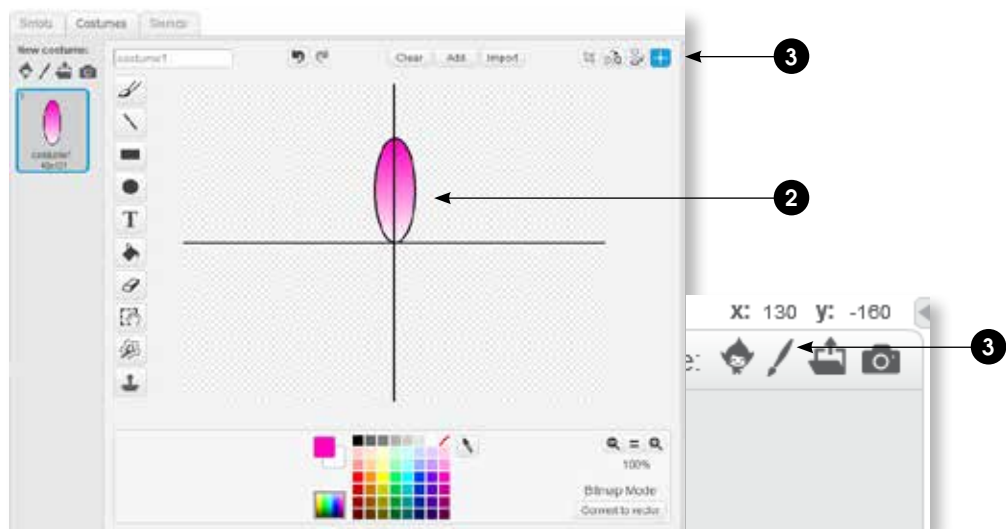


- 4 สร้างฟังก์ชันวาดก้านเพิ่มจากข้อ 3
- 5 เขียนสคริปต์ให้โปรแกรมหลักมีการสั่งวาดดอกไม้ 5 ครั้ง โดยแต่ละครั้งจะต้องสุ่มตำแหน่ง x,y

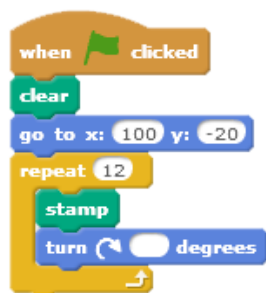
ใบงานที่ 7.3 ตัวบีม

ชื่อ-สกุล เลขที่

- 1) สร้างตัวละครใหม่ โดยใช้ Paint Editor วาดรูปกลีบดอกไม้ 1 กลีบ
 - 1) คลิก  เพื่อวาดรูปตัวละครใหม่
 - 2) วาดรูปกลีบดอกไม้ 1 กลีบ โดยใช้เครื่องมือ Costumes
 - 3) คลิก  Set costume center เพื่อเลือกศูนย์กลางภาพ ในที่นี้ให้คลิกที่โคลนกลีบ



- 2) เปลี่ยนชื่อตัวละครเป็น StampFlower
- 3) เขียนสคริปต์วาดรูปดอกไม้โดยใช้ตัวบีมดังนี้ คลิก  แล้วสังเกตผลลัพธ์




เส้นรอบวง หน่วย ทารด้วยจำนวน 12 กลีบ
เท่ากับ องศา

4 ใช้กลุ่มบล็อก Pen เขียนสคริปต์เกสรดอกไม้



เกสร

5 ปรับปรุงสคริปต์โดยสร้างฟังก์ชันเกสรดอกไม้ โดยเพิ่มบล็อก  เพื่อสร้างฟังก์ชันเกสรให้ตัวละครดอกไม้

เกสรดอกไม้	สคริปต์ของฟังก์ชันเกสรดอกไม้ คือ	ดอกไม้ที่มีเกสร	สคริปต์ของโปรแกรมหลัก คือ
			

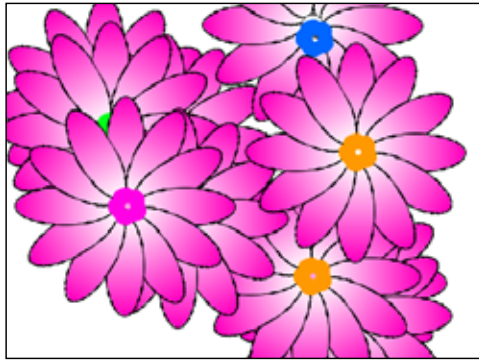
6 บันทึกโปรเจกต์ชื่อ flower

7 เปิดโปรเจกต์ใหม่ แล้วเขียนสคริปต์ให้ตัวละครแมวดังนี้แล้วบันทึกผลลัพธ์

สคริปต์	ผลลัพธ์
	เกิดอะไรขึ้นเมื่อคลิก  หลายๆ ครั้ง
	เกิดอะไรขึ้นเมื่อคลิก 

8 แก้วโปรเจกต์ flower ให้ดอกไม้ไปปรากฏที่ตำแหน่งต่างๆ บนเวทีแบบสุ่ม

ตัวอย่าง



ถ้าทำได้แล้วลองปรับปรุงชิ้นงานนี้ให้มีความน่าสนใจมากขึ้นสิจ๊ะ



ใบความรู้ที่ 7.1 การสั่งงานแบบวนซ้ำ

การสั่งงานแบบวนซ้ำใช้สำหรับเขียนสคริปต์ที่สั่งให้การทำงานซ้ำกันไปมาหลายๆ รอบ ให้พิจารณาสคริปต์ที่สั่งให้ตัวละครเดินบนเวทีเป็นรูปสี่เหลี่ยมโดยเคลื่อนที่ไปข้างหน้า 100 ก้าว จากนั้นหมุนทิศตามเข็มนาฬิกาไป 90 องศาทำแบบนี้ซ้ำกัน 4 รอบดังนี้

```

when clicked
  move 100 steps
  turn 90 degrees
  move 100 steps
  turn 90 degrees
  move 100 steps
  turn 90 degrees
  move 100 steps
  turn 90 degrees
  
```

ตัวอย่างที่ 1 การเขียนสคริปต์แบบลำดับ

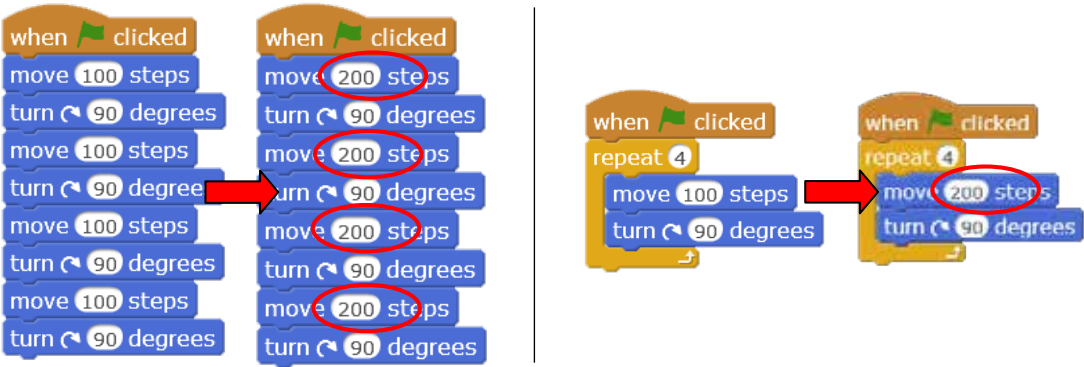
สคริปต์ดังกล่าวมีบล็อกที่ใช้คำสั่งซ้ำกันจำนวนมากสามารถเขียนสคริปต์ในรูปแบบของคำสั่งวนซ้ำ จะได้สคริปต์ที่สั้นลง ดังนี้

```

when clicked
  repeat 4
    move 100 steps
    turn 90 degrees
  
```

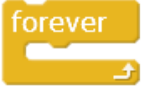

ตัวอย่างที่ 2 การเขียนสคริปต์แบบวนซ้ำ

การแก้ไขสคริปต์ที่เขียนแบบลำดับทำให้เสียเวลาในการแก้ไขเช่น ถ้าต้องการให้ตัวละครเดินเป็นรูปสี่เหลี่ยมที่กว้างขึ้นก็ต้องมีการแก้ไขสคริปต์ถึง 4 จุดถ้าใช้การทำงานแบบวนซ้ำการแก้ไข สคริปต์จะทำได้ง่ายขึ้น



ตัวอย่างที่ 3 เปรียบเทียบการแก้ไขสคริปต์แบบลำดับและแบบวนซ้ำ

การควบคุมลำดับการทำงานแบบวนซ้ำมีหลายลักษณะตามแต่ละชนิดของบล็อก Control เช่น

บล็อก	คำอธิบาย	ตัวอย่าง
	การทำงานซ้ำไม่รู้จบ	 <p>เดินไปข้างหน้าครั้งละ 10 หน่วย เมื่อชนขอบเวทีให้เดินกลับ ทำซ้ำแบบนี้ไปเรื่อยๆ ไม่รู้จบ</p>
	การทำงานซ้ำตามจำนวนรอบที่กำหนด	 <p>เดินไปข้างหน้า 50 หน่วย หมุนตามเข็มนาฬิกา 36 องศา ทำซ้ำแบบนี้ 10 รอบ</p>






กลุ่มบล็อก Pen

โปรแกรม Scratch มีกลุ่มบล็อก Pen ซึ่งเปรียบเสมือนปากกาที่ช่วยในการวาดรูปต่างๆ โดยสามารถกำหนดลักษณะปากกาได้ เช่น กำหนดสี ขนาดเส้น การยกปากกา การวางปากกา กลุ่มบล็อก Pen มีดังนี้

บล็อก	ความหมาย
	ปั๊มตัวละคร
	กำหนดสีปากกา โดยสามารถดูสีจากวัตถุที่เราต้องการ
	เพิ่มหรือลดค่าสีปากกา โดยระบุค่าสีเป็นตัวเลข
	กำหนดค่าสีปากกา
	เพิ่มหรือลดค่าเฉดสีปากกา
	กำหนดเฉดสีของปากกา โดยระบุค่าเฉดสีเป็นตัวเลข
	เพิ่มหรือลดขนาดของปากกา
	กำหนดขนาดปากกา โดยระบุขนาดเป็นตัวเลข

ป้อนค่าสุ่ม (pick random)

ในการเขียนสคริปต์โปรแกรมนอกจากการใช้คำสั่งพื้นฐาน เพื่อสร้างการเคลื่อนที่อย่างง่ายแล้ว โปรแกรม Scratch ยังมีคำสั่งสำหรับการสุ่มตัวเลข มีรูปแบบและการใช้งาน ดังนี้

บล็อก	คำอธิบาย	ตัวอย่าง
	การสุ่มตัวเลขจาก 1 ถึง 10	 ให้ตัวละครเคลื่อนที่ไปข้างหน้าเท่ากับตัวเลขที่ได้จากการสุ่มเลข 1 ถึง 10
		 ให้ตัวละครแสดงตัวเลขโดยการสุ่มเลขจาก 1 ถึง 10
		 ให้ตัวละครเคลื่อนที่ไปข้างหน้าเท่ากับตัวเลขที่ได้จากการสุ่มเลข 1 ถึง 10 โดยทำซ้ำคำสั่ง move เป็นจำนวน 5 ครั้ง
		 เคลื่อนที่ไปข้างหน้า 10 หน่วย โดยทำซ้ำคำสั่ง move เป็นจำนวนครั้งตามตัวเลขที่สุ่มได้

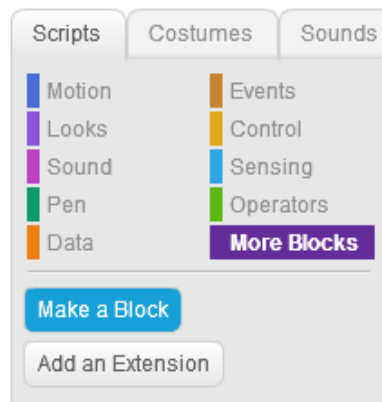



ใบความรู้ที่ 7.2 ฟังก์ชัน

ฟังก์ชัน (function) หรือโปรแกรมย่อย (subroutine) เป็นกลุ่มของคำสั่งที่มีการนิยามชื่อที่ใช้สำหรับอ้างอิงแทนกลุ่มคำสั่งเหล่านั้นเพื่อให้สามารถเรียกทำงานได้ตามต้องการ กลุ่มของคำสั่งที่ประกอบกันขึ้นเป็นฟังก์ชันจะทำหน้าที่อย่างใดอย่างหนึ่งโดยเฉพาะ

การสร้างฟังก์ชันสามารถทำได้ ดังนี้

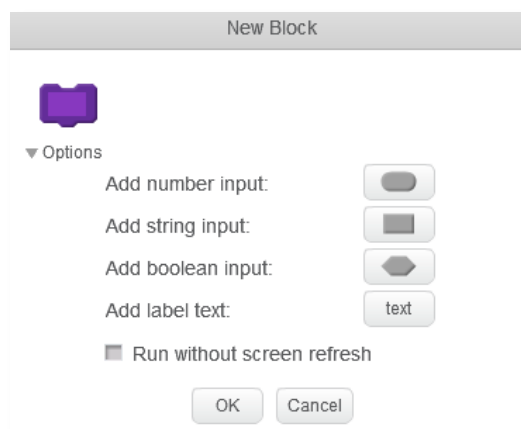
- 1) คลิกกลุ่มบล็อก More Blocks แล้วเลือกบล็อก Make a Block



- 2) จะปรากฏหน้าต่าง New Block ให้พิมพ์ชื่อฟังก์ชันที่ต้องการ ในช่อง 



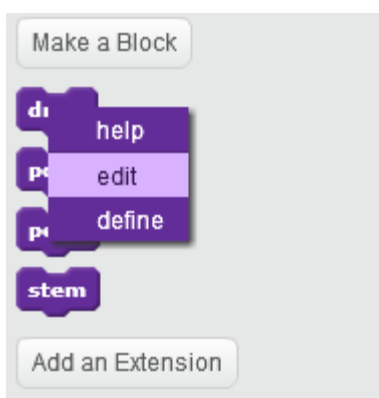
- 3) หากต้องการกำหนดการรับค่าข้อมูล หรือการแสดงผลข้อความ สำหรับฟังก์ชัน ให้คลิกที่คำสั่ง Options ซึ่งในที่นี้เราสามารถกำหนดการรับเข้าข้อมูลได้หลายค่า



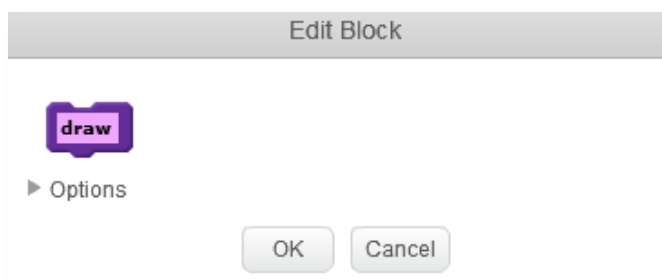
Add number input	เพิ่มการรับข้อมูลเข้าที่เป็นจำนวน
Add string input	เพิ่มการรับข้อมูลเข้าที่เป็นสตริง
Add Boolean input	เพิ่มการรับข้อมูลเข้าที่มีเป็นค่าจริงหรือเท็จ
Add label text	เพิ่มข้อความเพื่ออธิบายฟังก์ชัน
Run without screen refresh	แสดงผลหน้าจอเหมือนกับการใช้ Turbo mode

การแก้ไขชื่อฟังก์ชันและการกำหนดค่า สามารถทำได้ดังนี้

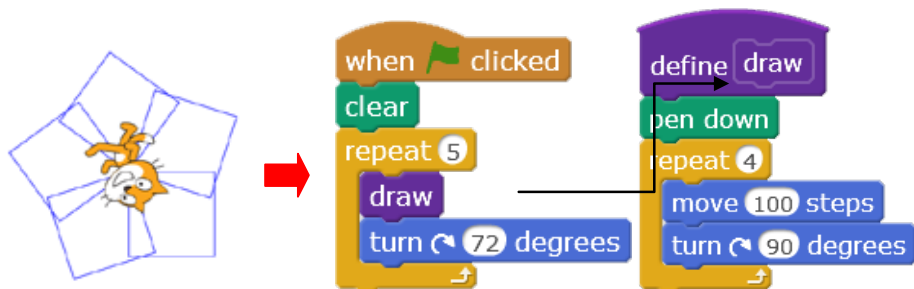
- 1) คลิกขวาที่ชื่อฟังก์ชันแล้วเลือกคำสั่ง edit



- 2) จะปรากฏหน้าต่างต่าง Edit Block สามารถแก้ไขชื่อและการกำหนดค่าต่างๆ ได้



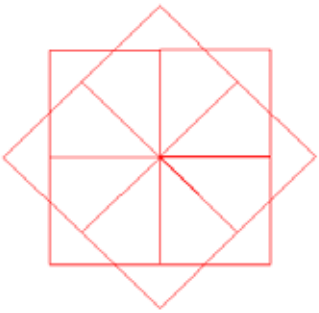
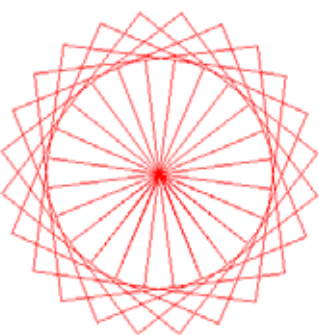

ตัวอย่างการใช้งานฟังก์ชัน

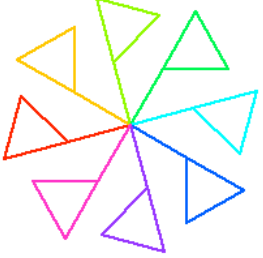
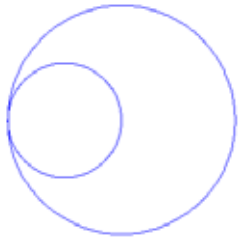

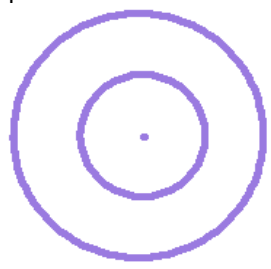


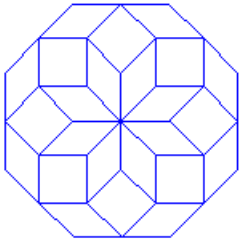
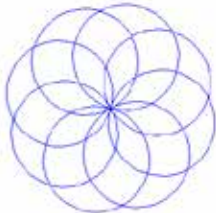

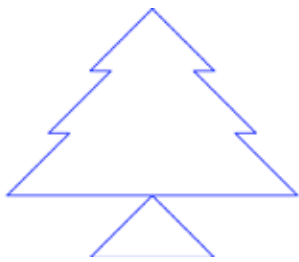


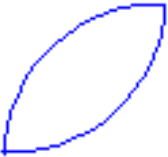
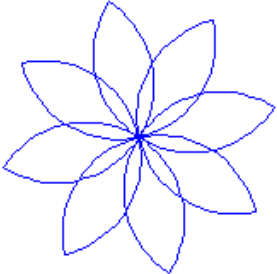

แบบฝึกหัดเพิ่มเติม

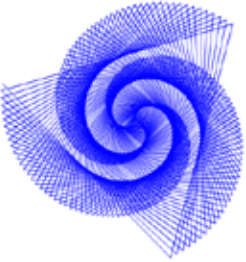

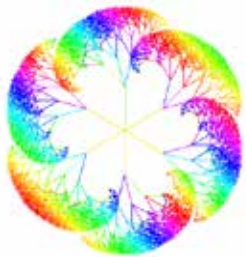
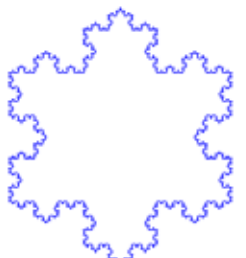
สร้างโปรเจกต์ชื่อ ArtPractice แล้วเขียนแนวคิด และสคริปต์ให้ตัวละครวาดรูปเพื่อให้ได้ผลลัพธ์ดังตารางต่อไปนี้

ผลลัพธ์	แนวคิด	สคริปต์
<p>สี่เหลี่ยม 8 รูป</p> 	<ol style="list-style-type: none"> 1) วาดรูปสี่เหลี่ยม 1 รูป 2) กำหนดมุมในการวาดสี่เหลี่ยมรูปต่อไป 45 องศา 3) ทำซ้ำข้อ 1 และ 2 จนครบ 8 ครั้ง 	
<p>กงจักร</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>ธง</p> 	<p>แนวคิดที่ 1</p> <ol style="list-style-type: none"> 1) วาดด้ามธง 2) วาดธงรูปสามเหลี่ยม <p>แนวคิดที่ 2</p> <ol style="list-style-type: none"> 1) วาดธงรูปสามเหลี่ยม 2) วาดด้ามธง 	

ผลลัพธ์	แนวคิด	สคริปต์
<p>ธงหลากสี</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>วงกลม 2 วง</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>คลื่น</p> 	<p>1) วาดครึ่งวงกลมด้านบน วาดรูปวงกลมด้านล่าง</p> <p>2) ทำซ้ำข้อ 1 และ 2 จนครบ 2 ครั้ง</p>	
<p>เป้า</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	

ผลลัพธ์	แนวคิด	สคริปต์
<p>ดอกไม้สามมิติ</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>วงกลมดอกไม้</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>นกขี้ไมโท</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>ต้นไม้</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	

ผลลัพธ์	แนวคิด	สคริปต์
<p>กลีบดอกไม้</p> 	<ol style="list-style-type: none"> 1) กำหนดพิกัด $x = -100$, $y = -80$ 2) ลากเส้นสั้นๆ 3) กำหนดมุมหมุน 1 องศา 4) ทำซ้ำข้อ 2 และ 3 ไปจน 90 รอบ ได้กลีบดอกไม้ ครึ่งกลีบ 5) 6) 	<pre> clear pen up go to x: -100 y: -80 pen down repeat 18 turn 1 degrees move 5 steps turn 90 degrees repeat turn 1 degrees move 5 steps </pre>
<p>ดอกไม้</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>เลิฟ</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	

ผลลัพธ์	แนวคิด	สคริปต์
<p>สามเหลี่ยมมัทศจรรย์</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>ต้นไม้</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>ดอกไม้สายรุ้ง</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	
<p>เกล็ดน้ำแข็ง</p> 	<p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	

กิจกรรมที่ 8 เงื่อนไขและตัวแปร

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม
- 1.2 กำหนดและใช้งานตัวแปร
- 1.3 ใช้โครงสร้างแบบลำดับ เงื่อนไข และวนซ้ำ

2. สารการเรียนรู้

- 2.1 การใช้บล็อก if-else
- 2.2 การใช้บล็อกตัวดำเนินการเปรียบเทียบ
- 2.3 การใช้งานกลุ่มบล็อก Data

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 สร้างประโยคเงื่อนไขโดยใช้บล็อกตัวดำเนินการ $>$, $<$ และ $=$
- 3.2 เขียนสคริปต์ในการตรวจสอบเงื่อนไขโดยใช้บล็อก if และ if-else
- 3.3 ใช้กลุ่มบล็อก Data

4. แนวคิด

คำสั่ง if ใช้ตรวจสอบเงื่อนไข ถ้าเงื่อนไขหลัง if เป็นจริง จะทำคำสั่งภายในบล็อก if
คำสั่ง if-else ใช้ตรวจสอบเงื่อนไข ถ้าเงื่อนไขหลัง if เป็นจริง จะทำคำสั่งภายในบล็อก if
แต่ถ้าเงื่อนไขเป็นเท็จ จะทำคำสั่งภายในบล็อก else สำหรับเงื่อนไขที่ใช้ในการตรวจสอบจะนำตัว
ดำเนินการมาช่วยในการเปรียบเทียบ เช่น $>$, $<$ และ $=$

การสร้างตัวแปรและเรียกใช้ตัวแปร อยู่ในกลุ่มบล็อก Data บล็อก Make a Variable
เมื่อมีการสร้างตัวแปรใหม่ โปรแกรมจะสร้างบล็อกต่างๆ ที่เกี่ยวข้องกับตัวแปรนั้นเพิ่มเติม สำหรับ
ใช้ในการทำงานต่างๆ

5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
8.1	มารู้จักเงื่อนไขกันเถอะ	30
8.2	เกมทายตัวเลข	40

5.2 ใบความรู้

- ใบความรู้ที่ 8.1 เรื่อง เงื่อนไข
- ใบความรู้ที่ 8.2 เรื่อง ตัวแปร

5.3 อื่นๆ

- ไฟล์ตัวอย่างเกมที่สร้างจากโปรแกรม Scratch ชื่อ Maze.sb2 ในโฟลเดอร์ \MediaScratch\Act8

6. วิธีดำเนินการกิจกรรม

6.1 การจัดเตรียม

6.1.1 ใบงานที่ 8.1 ตามจำนวนกลุ่ม และใบงานที่ 8.2 ตามจำนวนผู้เรียน

6.1.2 ใบความรู้ที่ 8.1-8.2 ตามจำนวนผู้เรียน

6.2 ขั้นตอนการดำเนินการ

6.2.1 ผู้สอนสาธิตการเล่นเกมเขาวงกต ที่สร้างจากโปรแกรม Scratch ชื่อไฟล์ Maze.sb2 ให้ผู้เรียนชมและร่วมกันสังเกตลักษณะเงื่อนไขของเกมว่าน่าจะมีเงื่อนไขอะไรบ้าง

6.2.2 ผู้สอนยกตัวอย่างสถานการณ์ในชีวิตประจำวันที่มีการตัดสินใจเลือกอย่างใดอย่างหนึ่ง อยู่เสมอ เพื่ออธิบายเรื่องเงื่อนไข โดยตัวอย่างของสถานการณ์ ให้มีการตัดสินใจว่า จะต้องทำอะไรก่อนหรือหลัง หรือตัดสินใจเลือกทำสิ่งใดสิ่งหนึ่งตามเหตุและผล จากนั้นเชื่อมโยงไปหาการสร้างประโยคเงื่อนไข

6.2.3 ผู้สอนอธิบายเรื่องการเขียนคำสั่งเงื่อนไขแบบ if-else และการใช้กลุ่มบล็อก Operators โดยให้ผู้เรียนศึกษาใบความรู้ที่ 8.1 เรื่อง เงื่อนไข ประกอบการอธิบายของผู้สอน

- 6.2.4 ผู้สอนแบ่งผู้เรียนออกเป็นกลุ่ม กลุ่มละ 2 คนแล้วให้แต่ละกลุ่มช่วยกันทำใบงานที่ 8.1 เรื่องมารู้จักเงื่อนไขกันเถอะจากนั้นผู้เรียนและผู้สอนร่วมกันเฉลยคำตอบในใบงานที่ 8.1
- 6.2.5 ให้ผู้เรียนศึกษาใบความรู้ที่ 8.2 เรื่องตัวแปร และให้ผู้เรียนแต่ละคนทำใบงานที่ 8.2 เรื่องเกมทายตัวเลข
- 6.2.6 ผู้เรียนและผู้สอนร่วมกันสรุปความรู้เรื่องเงื่อนไข และตัวแปร
- 6.2.7 ผู้เรียนและผู้สอนร่วมกันสรุป อภิปรายถามตอบความรู้ที่ได้จากการทำกิจกรรม

7. การวัดและประเมินผล

- 7.1 ตรวจคำตอบในใบงานและผลงาน
- 7.2 สังเกตการมีส่วนร่วมในการทำกิจกรรมกลุ่ม

8. แหล่งข้อมูลเพิ่มเติม

- 8.1 ผลงานที่สร้างด้วยโปรแกรม Scratch เว็บไซต์ [http://scratch.mit.edu/\(2/03/2559\)](http://scratch.mit.edu/(2/03/2559))
- 8.2 เทคนิคการใช้โปรแกรม Scratch เว็บไซต์ [http://scratched.gse.harvard.edu/\(2/03/2559\)](http://scratched.gse.harvard.edu/(2/03/2559))

9. ข้อเสนอแนะ

- 9.1 การยกตัวอย่างสถานการณ์ในชีวิตประจำวันให้ผู้เรียนต้องตัดสินใจ ในข้อ 6.2.2 เพื่อเป็นการเชื่อมโยงความรู้เข้าสู่การสอนเรื่องเงื่อนไข ยกตัวอย่างเช่น การตัดสินใจเลือกซื้อของให้แม่ถามคำถามให้ผู้เรียนคิดว่าจะเลือกเลือกซื้อสิ่งใดระหว่างนาฬิกา หรือกระเป๋า ลองถามผู้เรียน 2-3 คน และถามเหตุผลของผู้เรียนแต่ละคนว่าเหตุใดจึงตัดสินใจเลือกสิ่งนั้น (ผู้เรียนอาจจะตอบ เลือกซื้อนาฬิกาเพราะ แม่มีกระเป๋าแล้ว หรืออื่นๆ) จากนั้นถามคำถามเชื่อมโยงคำถามอื่นๆ อีก เช่น จะนั่งรถเมล์ หรือ รถแท็กซี่ไปโรงเรียนดี (ผู้เรียนอาจจะตอบ เลือกแท็กซี่เพราะเร็วกว่า หรือเลือกรถเมล์เพราะไม่รีบ แล้วเชื่อมโยงคำตอบนี้เข้าสู่การสอนเรื่องเงื่อนไข คือ รีบหรือไม่ ถ้ารีบก็รถแท็กซี่ ถ้าไม่รีบก็เลือกรถเมล์) จากนั้นถามคำถามอื่นเพื่อให้ผู้เรียนคิดเป็นเงื่อนไข
- ชื่อของขวัญให้แม่ เงื่อนไขคือ มีเงินหรือไม่ (ถ้ามีเงินซื้อนาฬิกา ถ้าไม่มีเงินซื้อกระเป๋า)
 - ดูหนังเรื่องอะไรดี ระหว่างสไปเดอร์แมน กับซูเปอร์แมน เงื่อนไขคือ หนังเรื่องไหนฉายก่อน (ถ้าสไปเดอร์แมนฉายก่อน เลือกดูสไปเดอร์แมน)
- 9.2 ผู้สอนอาจให้ผู้เรียนช่วยกันเขียนโปรแกรมเพื่อตรวจสอบคำตอบในข้อ 2 ของใบงานที่ 8.1 โดยศึกษาวิธีการสร้างตัวแปรจากใบความรู้ที่ 8.2 ก่อน

ใบงานที่ 8.1 มารู้จักเงื่อนไขกันเถอะ

รายชื่อสมาชิกในกลุ่มที่

1 2

1) พิจารณาเงื่อนไข จากสถานการณ์ แล้วทำเครื่องหมาย ✓ ลงในช่องวงกลมพร้อมบันทึกผลลัพธ์

ลำดับ	ขั้นตอนวิธีการ	สถานการณ์	พิจารณาเงื่อนไข	ผลลัพธ์
1.1	ถ้า ตำแหน่ง $y > 0$ แล้ว กำหนดสีปากกาเป็นสีแดง	ตำแหน่ง $x,y =$ $(-20, 50)$	<input type="radio"/> จริง <input type="radio"/> เท็จ	สีปากกาคือ <input type="radio"/> สีดำ <input type="radio"/> สีแดง
1.2	ถ้า ตัวแปร no มีค่า > 5 แล้ว กำหนดสีปากกาเป็นสีน้ำเงิน มีฉะนั้น กำหนดสีปากกาเป็นสีเขียว	ตัวแปร no มีค่า เป็น 3	<input type="radio"/> จริง <input type="radio"/> เท็จ	สีปากกาคือ <input type="radio"/> สีน้ำเงิน <input type="radio"/> สีเขียว
1.3	กำหนดสีปากกาคือสีเขียว ถ้า ตำแหน่ง $x = 30$ แล้ว ปากกาเป็นสีน้ำเงิน มีฉะนั้น ปากกาเป็นสีชมพู	ตำแหน่ง $x,y =$ $(-30, 20)$	<input type="radio"/> จริง <input type="radio"/> เท็จ	สีปากกาคือ <input type="radio"/> สีน้ำเงิน <input type="radio"/> สีเขียว <input type="radio"/> สีชมพู
1.4	ถ้า ตัวแปร count มีค่า < 30 แล้ว พูดว่า น้อยกว่า 30 มีฉะนั้น พูดว่า มากกว่าหรือเท่ากับ 30	ตัวแปร count มีค่าเป็น 40	<input type="radio"/> จริง <input type="radio"/> เท็จ	พูดว่า
1.5	ถ้า ตัวแปร age มีค่า > 15 แล้ว พูดว่า คุณโตแล้ว มีฉะนั้น พูดว่า คุณยังเด็ก	ตัวแปร age มีค่า เป็น 10	<input type="radio"/> จริง <input type="radio"/> เท็จ	พูดว่า
1.6	ถ้า ตัวแปร answer = เอก แล้ว พูดว่า สวัสดีครับเอก มีฉะนั้น พูดว่า ยินดีที่รู้จักครับ	ตัวแปร answer มีค่าเป็น นุ่น	<input type="radio"/> จริง <input type="radio"/> เท็จ

2) ให้ผู้เรียนศึกษาใบความรู้ที่ 8.1 จากนั้นพิจารณาสคริปต์ต่อไปนี้ แล้วตอบคำถาม

สคริปต์	อธิบายคำสั่ง	ผลลัพธ์
<p>2.1</p> 	<p>1. กำหนดตัวแปร guess เป็น 10</p> <p>2. ถ้า ตัวแปร guess น้อยกว่า 10 แล้ว ตัวละครพูดว่า Hello</p>	<p>ตัวละคร ไม่พูดอะไร</p> <p>เพราะว่า ตัวแปร guess มีค่าเป็น 10 ทำให้เงื่อนไข เป็น <input type="radio"/> จริง <input checked="" type="radio"/> เท็จ</p> <p>ดังนั้นจึง ไม่พูดว่า Hello</p>
<p>2.2</p> 	<p>1. กำหนดตัวแปร guess เป็น</p> <p>2. ถ้า ตัวแปร guess แล้ว ตัวละครพูดว่า มิฉะนั้น ตัวละครพูดว่า rabbit</p>	<p>ตัวละคร พูดว่า</p> <p>เพราะว่า ตัวแปร guess มีค่าเป็น ทำให้เงื่อนไข เป็น <input type="radio"/> จริง <input type="radio"/> เท็จ</p> <p>ดังนั้นจึงพูดว่า rabbit</p>
<p>2.3</p> 	<p>1. กำหนดตัวแปร guess เป็น</p> <p>2. ถ้า ตัวแปร guess แล้ว ตัวละครพูดว่า มิฉะนั้น ตัวละครพูดว่า rabbit</p>	<p>ตัวละคร พูดว่า</p> <p>เพราะว่า ตัวแปร guess มีค่าเป็น ทำให้เงื่อนไข เป็น <input type="radio"/> จริง <input type="radio"/> เท็จ</p> <p>ดังนั้นจึง</p>
<p>2.4</p> 	<p>1. เป็น</p> <p>2. ถ้า แล้ว มิฉะนั้น</p>	<p>ตัวละคร พูดว่า</p> <p>เพราะว่า มีค่าเป็น ทำให้เงื่อนไข เป็น <input type="radio"/> จริง <input type="radio"/> เท็จ</p> <p>ดังนั้นจึง</p>
<p>2.5</p> 	<p>1. เป็น</p> <p>2. ถ้า แล้ว มิฉะนั้น</p> <p>3. หยุดรอ วินาที</p> <p>4. ถ้า แล้ว มิฉะนั้น</p>	<p>ตัวละคร พูดว่า</p> <p>หยุดรอ แล้วพูดว่า</p> <p>เพราะว่า ตัวแปร guess เป็น มีค่าเป็น ทำให้เงื่อนไขแรก เป็น <input type="radio"/> จริง <input type="radio"/> เท็จ</p> <p>ดังนั้นจึง</p> <p>หยุดรอ วินาที ทำให้เงื่อนไขที่สองเป็น <input type="radio"/> จริง <input type="radio"/> เท็จ</p> <p>ดังนั้นจึง</p>

ใบงานที่ 8.2 เกมทายตัวเลข

ชื่อ-สกุล เลขที่

สร้างโปรเจกต์ใหม่ เกมทายตัวเลข (guess) จากตัวเลขที่โปรแกรมสุ่มมาให้ระหว่าง 1 - 10 โดยเขียนสคริปต์ให้ตัวละครแมว ต่อไปนี้

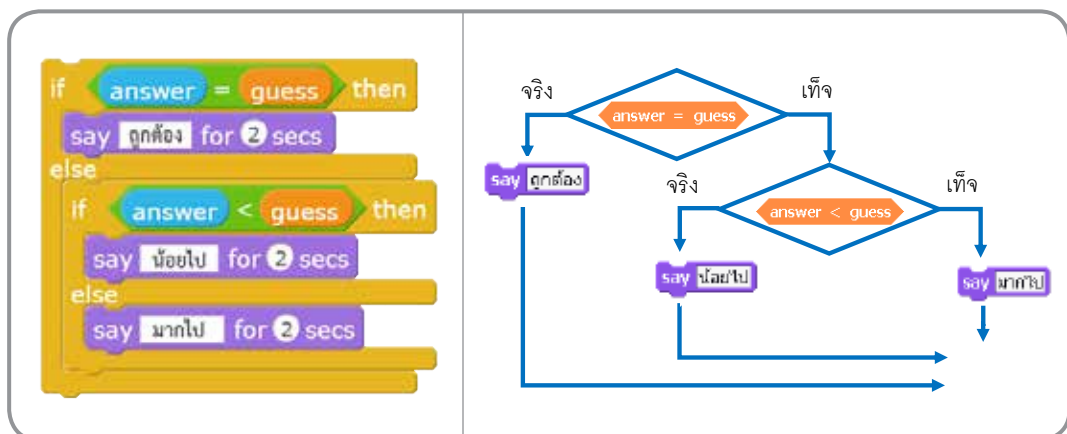
- สร้างตัวแปร guess โดยคลิกเลือกกลุ่มบล็อก Data จากนั้นคลิก **Make a Variable** พิมพ์ชื่อตัวแปร guess ในกล่องโต้ตอบ New Variable แล้วคลิกปุ่ม OK
สังเกตที่เวที สิ่งที่พบคือ
ถ้าหน้าเครื่องหมาย ✓ ออก guess สังเกตที่เวที สิ่งที่พบคือ
- เขียนสคริปต์ กำหนดค่าให้กับตัวแปร guess ด้วยการให้โปรแกรมสุ่มตัวเลข ระหว่าง 1 - 10




- เพิ่มบล็อก ask เพื่อให้ผู้ใช้ ทายค่าของตัวแปร guess



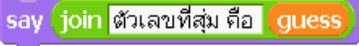




- สร้างเงื่อนไขเพื่อตรวจสอบตัวเลขที่รับเข้ามา กับตัวแปร guess



- คลิก แล้วทดสอบใส่ค่าตัวเลข

- 6) โปรแกรมดังกล่าวจะสามารถทายตัวเลขได้ 1 ครั้ง แต่ถ้าต้องการให้ทายตัวเลขได้ 3 ครั้ง จะต้องเลือกใช้บล็อกใด ให้เลือกระหว่างบล็อก forever และ repeat โดยทดลองใช้บล็อก ทั้ง 2 เพิ่มเข้าไปในสคริปต์ในข้อ 4 แล้วคลิก  คำตอบคือ



- 7) เพิ่มบล็อก  ต่อท้ายสคริปต์ทั้งหมด (หลังจากการใช้บล็อก repeat ในข้อ 6) คลิก  แล้วทายตัวเลข
 สิ่งที่เกิดขึ้น
- 8) ให้แสดงค่าตัวแปร guess ดู   ค่าตัวแปร guess แล้วทายค่าตัวเลขให้ถูกต้อง
 บันทึกผลลัพธ์ที่ได้
- 9) เพิ่มบล็อก stop all ลงในสคริปต์เดิม (ในข้อ 4) ตามตำแหน่งคลิก  แล้วทายค่าตัวเลข
 ให้ถูกต้อง



- บันทึกผลลัพธ์ที่ได้
- 10) จากเกมทายตัวเลข ให้สร้างฟังก์ชันเปลี่ยนแปลงขนาดของตัวละคร โดยมีการรับข้อมูลเข้า เป็นขนาดของตัวละคร ในการทายตัวเลข ถ้าทายผิดให้เรียกฟังก์ชันเปลี่ยนแปลงขนาดเพื่อ เปลี่ยนให้ตัวละครมีขนาดเล็กลง ถ้าทายถูกให้เรียกฟังก์ชันเปลี่ยนแปลงขนาดเพื่อเปลี่ยนให้ ตัวละครมีขนาดใหญ่ขึ้น
- 11) บันทึกโปรเจกต์ชื่อ Guess_Number

ใบความรู้ที่ 8.1 เงื่อนไข

บล็อก if, if-else



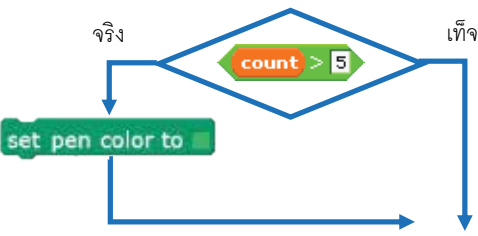
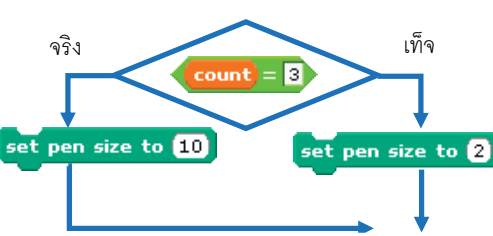

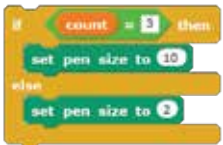
บล็อก if ใช้ในการทำงานแบบมีเงื่อนไข (condition) เพื่อควบคุมทิศทางการทำงาน ในการตัดสินใจอย่างใดอย่างหนึ่ง โดยถ้าเงื่อนไขเป็นจริง จะทำคำสั่งภายในบล็อก if แต่ถ้าเงื่อนไขเป็นเท็จ จะไปทำคำสั่งที่ต่อบล็อก if

ในกรณีที่ เป็น บล็อก if else ถ้าหากเงื่อนไขเป็นจริง จะทำคำสั่งภายในบล็อก if แต่ถ้าเงื่อนไขเป็นเท็จ จะทำคำสั่งภายในบล็อก else

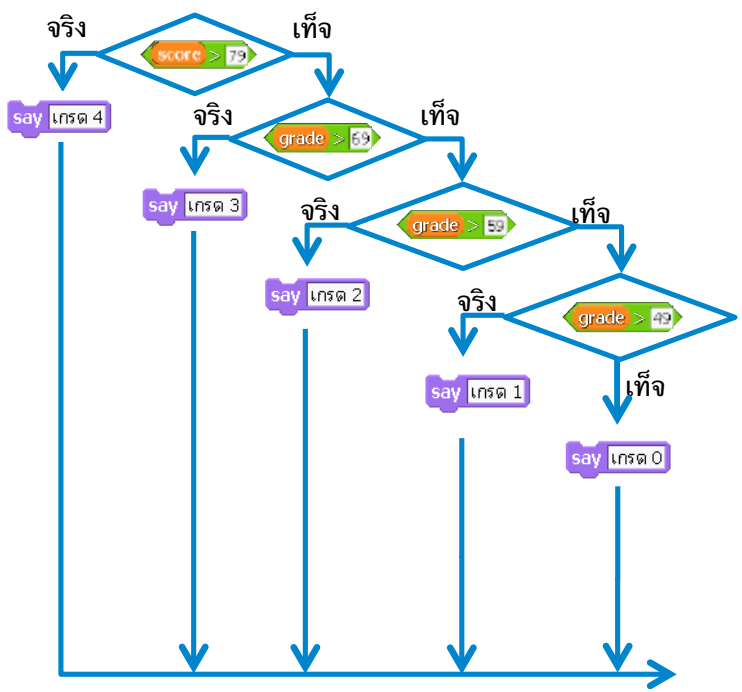
การสร้างประโยคเงื่อนไขใน Scratch จะใช้บล็อกในกลุ่ม Operators เป็นตัวดำเนินการเพื่อช่วยในการประเมินหรือเปรียบเทียบเพื่อให้ได้ผลเป็นจริง หรือเท็จ ใน Scratch มีกลุ่มบล็อก Operators ที่มีตัวดำเนินการเพื่อนำมาใช้สร้างเงื่อนไขได้ ตัวอย่างเช่น

บล็อก	ความหมาย	ตัวอย่างการใช้งาน	ผลลัพธ์
	น้อยกว่า		เท็จ
	มากกว่า		จริง
	เท่ากัน		เท็จ

ตัวอย่างการใช้งานบล็อก if และ if-else

บล็อก if	บล็อก if-else
	
	
	
ถ้า ตัวแปร count มีค่ามากกว่า 5 แล้ว กำหนดปากกาเป็นสีแดง	ถ้า ตัวแปร count มีค่าเท่ากับ 3 แล้ว กำหนดขนาดปากกาเป็นขนาด 10 มิฉะนั้น กำหนดขนาดปากกาเป็นขนาด 2

ตัวอย่าง การเขียนสคริปต์โปรแกรมตัดเกรด



```

if grade > 79 then
  say เกรด 4
else
  if grade > 69 then
    say เกรด 3
  else
    if grade > 59 then
      say เกรด 2
    else
      if grade > 49 then
        say เกรด 1
      else
        say เกรด 0
      end
    end
  end
end
  
```



ใบความรู้ที่ 8.2 ตัวแปร




กลุ่มบล็อก Data

ตัวแปร (variables) คือ ชื่อที่กำหนดขึ้นเพื่อใช้เก็บข้อมูล และเรียกใช้ในการทำงานต่างๆ การเรียกใช้ตัวแปรในกลุ่มบล็อก Data จะต้องสร้างขึ้นเองโดยคลิก **Make a Variable** จากนั้นตั้งชื่อตัวแปร ชื่อตัวแปรควรมีความหมายที่สอดคล้องกับข้อมูล เช่น name ใช้สำหรับเก็บค่าชื่อ, score ใช้สำหรับเก็บค่าคะแนน และ count ใช้สำหรับการนับค่า

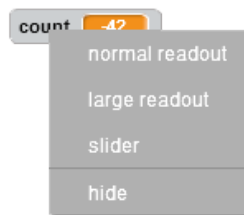
เมื่อสร้างตัวแปร count เสร็จแล้วจะมีบล็อกเกิดขึ้นมาอีก 5 บล็อก คือ

บล็อก	คำอธิบาย
<input type="checkbox"/> count	<input checked="" type="checkbox"/> count แสดงค่าตัวแปรที่เวที <input type="checkbox"/> count ซ่อนค่าตัวแปรที่เวที
set count to 0	set count to -5 กำหนดค่าเริ่มต้นให้กับตัวแปร count ให้มีค่าเป็น -5 set count to 8 กำหนดค่าเริ่มต้นให้กับตัวแปร count ให้มีค่าเป็น 8
change count by 1	change count by 1 เปลี่ยนค่าตัวแปร count เพิ่มขึ้นครั้งละ 1 หน่วย change count by -3 เปลี่ยนค่าตัวแปร count ลดลงครั้งละ 3 หน่วย
show variable count	แสดงตัวแปร count และข้อมูล
hide variable count	ซ่อนตัวแปร count และข้อมูล

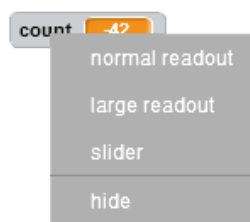
รูปแบบการแสดงผลข้อมูลตัวแปรบนเวทีมี 3 แบบดังนี้


รูปแบบ	คำอธิบาย	การแสดงผลบนเวที
1. normal readout	แบบเริ่มต้น	
2. large readout	แบบขนาดใหญ่	
3. slider	แบบแถบเลื่อนโดยสามารถ กำหนดค่าต่ำสุดและสูงสุดของแถบ เลื่อนได้	

การกำหนดรูปแบบการแสดงผลข้อมูลตัวแปรทำได้โดย คลิกขวาที่ตัวแปรบนเวทีจากนั้นให้เลือกรูปแบบที่ต้องการ



หากต้องการซ่อนการแสดงผลข้อมูลตัวแปรบนเวทีให้คลิกขวาที่ตัวแปรบนเวที จากนั้นให้เลือก hide



ตัวแปรบนเวทีจะถูกซ่อน และ เครื่องหมาย ✓ หน้าชื่อตัวแปรก็จะหายไป 

กิจกรรมที่ 9 เกมสร้างสรรค์

1. ผลการเรียนรู้

- 1.1 ใช้คำสั่งพื้นฐานในการเขียนโปรแกรม
- 1.2 ใช้โครงสร้างแบบลำดับ เงื่อนไข และวนซ้ำ
- 1.3 กำหนดและใช้งานตัวแปร
- 1.4 ใช้งานรายการ (List)

2. สารการเรียนรู้

- 2.1 บล็อก touching
- 2.2 กลุ่มบล็อก Data
- 2.3 บล็อกวนซ้ำแบบมีเงื่อนไข repeat until

3. จุดประสงค์ ให้ผู้เรียนสามารถ

- 3.1 สร้างชิ้นงานโดยใช้คำสั่ง touching
- 3.2 สร้างตัวแปรเพื่อใช้สำหรับการนับ
- 3.3 เขียนสคริปต์วนซ้ำแบบมีการตรวจสอบเงื่อนไขโดยใช้ บล็อก repeat until
- 3.4 สร้างรายการ (List) เพื่อใช้สำหรับนำเข้าหรือเก็บข้อมูล

4. แนวคิด

กลุ่มบล็อก Sensing ใช้ในการตรวจสอบการรับรู้ แล้วทำงานตามที่กำหนด สามารถนำบล็อกเหล่านี้มาประยุกต์ใช้เพื่อสร้างชิ้นงานได้

บล็อก change อยู่ในกลุ่มบล็อก Data ใช้สำหรับเพิ่มหรือลดค่าของตัวแปรที่กำหนด การทำงานแบบวนซ้ำ แบ่งได้เป็น 2 ลักษณะ คือ การทำซ้ำแบบไม่มีเงื่อนไข (forever) และการทำซ้ำจนกว่าเงื่อนไขจะเป็นจริง (repeat until)

การใช้งานรายการ (List) เมื่อต้องการเก็บข้อมูลแบบเดียวกันจำนวนมาก

5. สื่ออุปกรณ์

5.1 ใบงาน

ใบงานที่	เรื่อง	เวลา (นาที)
9.1	สัมผัสสนุก	40

5.2 ใบความรู้

- ใบความรู้ที่ 9.1 เรื่อง สัมผัสสนุก
- ความรู้ในภาคผนวกเรื่อง การใช้งานตัวแปรอย่างง่าย

5.3 อื่นๆ

- ไฟล์เกมชื่อ matching game ไฟล์ตัวละครชื่อ giraffe.sprite2, giraffe shadow.sprite2, Magnifier.sprite2 ไฟล์พื้นหลังชื่อ bg.jpg และไฟล์รายชื่อชื่อ name.txt ในโฟลเดอร์ \MediaScratch\Act9

6. วิธีดำเนินการ

6.1 การจัดเตรียม

- 1) ใบงานที่ 9.1 ตามจำนวนผู้เรียน
- 2) ใบความรู้ที่ 9.1 ตามจำนวนผู้เรียน
- 3) ความรู้ในภาคผนวกเรื่อง การใช้งานตัวแปรอย่างง่าย ตามจำนวนผู้เรียน
- 4) ไฟล์เกม ชื่อ matching game.sb2 สำหรับผู้สอน
- 5) ไฟล์ตัวละครชื่อ giraffe.sprite2, giraffe shadow.sprite2, Magnifier.sprite2 ไฟล์พื้นหลังชื่อ bg.jpg และไฟล์รายชื่อชื่อ name.txt เก็บไว้บนเดสก์ท็อปเครื่องคอมพิวเตอร์ของผู้เรียน ภายในโฟลเดอร์ Act9

6.2 ขั้นตอนการดำเนินการ

- 1) ผู้สอนนำเข้าสู่บทเรียนโดยเปิดตัวอย่างเกม Matching Game ให้ผู้เรียนดู โดยให้ผู้เรียนสังเกตการสัมผัสของตัวละครแต่ละตัว และเหตุการณ์ที่เกิดขึ้นหลังจากการสัมผัส
- 2) ผู้เรียนช่วยกันบอกสิ่งที่สังเกตได้จากเกม และให้ผู้เรียนช่วยกันสำรวจว่าจะใช้กลุ่มบล็อกและบล็อกใด ในการตรวจสอบการสัมผัสกันของตัวละคร เมาส์ หรือ สีที่พบเห็นจากเกม

- 3) ผู้เรียนศึกษาใบความรู้ที่ 9.1 เรื่อง สัมผัสสนุก แล้วทำใบงานที่ 9.1 เรื่อง สัมผัสสนุก
- 4) ผู้สอนเลือกผู้เรียนที่สร้างโปรเจกต์ที่มีความน่าสนใจ 1-2 คน ให้ออกมานำเสนอหน้าชั้นเรียน
- 5) ผู้สอนหาอาสาสมัครนำเสนอคำตอบในข้อ 8.5 เพื่อตรวจสอบความเข้าใจและสรุปเกี่ยวกับคำสั่งวนซ้ำที่มีการตรวจสอบเงื่อนไข และประเมินความเข้าใจผู้เรียนทั้งห้อง
- 6) ผู้สอนสรุปการใช้งานบล็อก touching, repeat until และ change

7. การวัดและประเมินผล

- 7.1 ตรวจสอบใบงานและผลงาน
- 7.2 สังเกตการร่วมกิจกรรมในการระดมความคิดของผู้เรียน
- 7.3 สังเกตการทำงานและการทำกิจกรรมร่วมกันในชั้นเรียน

8. แหล่งข้อมูลเพิ่มเติม

-

9. ข้อเสนอแนะ

ครูผู้สอนอาจให้ผู้เรียนทำแบบฝึกหัดเพิ่มเติมในเรื่องการใช้รายการหากมีเวลาเพียงพอหรือให้เป็นงานเสริม โดยทำตามขั้นตอนต่อไปนี้

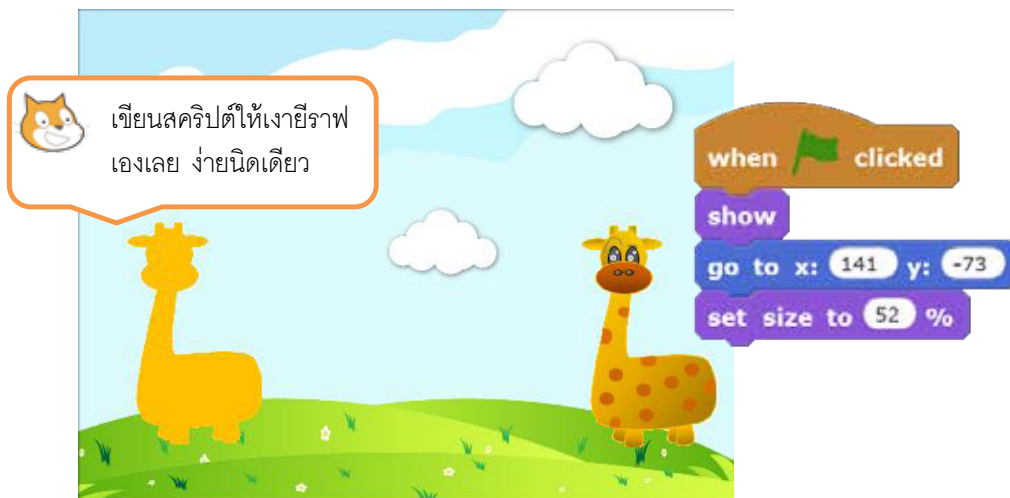
- 1) ครูผู้สอนทบทวนเรื่องตัวแปร ถึงหลักการทำงานและหน้าที่ของตัวแปร แล้วยกตัวอย่างสถานการณ์ที่มีการเก็บค่าข้อมูลแบบเดียวกันในจำนวนมาก เช่น ชื่อหรือคะแนนของนักเรียนในห้อง พร้อมแนะนำคำสั่งรายการ
- 2) ผู้เรียนศึกษาความรู้ในภาคผนวกเรื่อง การใช้งานตัวแปรอย่างง่าย และทดลองทำแบบฝึกหัดเพิ่มเติม
- 3) ผู้สอนสรุปการใช้งาน List สามารถนำมาใช้ในการเก็บข้อมูลอย่างง่ายได้

ใบงานที่ 9.1 สัมผัสสนุก


ชื่อ-สกุล เลขที่

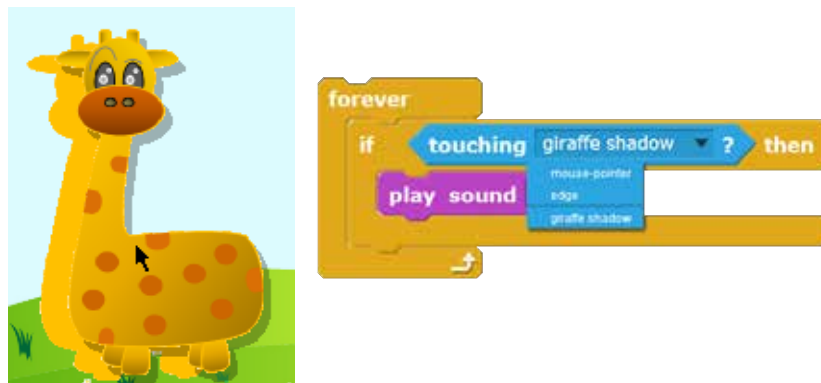
คำชี้แจง ใบงานนี้ผู้เรียนจะได้สร้างเกมจับคู่ระหว่างยีราฟตัวจริงและเงา โดยถ้าจับคู่ถูกต้องจะได้คะแนนครั้งละ 1 คะแนน แต่ถ้าจับผิดตัวคะแนนจะติดลบ 1


- 1 สร้างโปรเจกต์ใหม่ โดยนำเข้าตัวละครยีราฟ เงา และพื้นหลังจากไฟล์ในโฟลเดอร์ชื่อ Act9 บนเดสก์ท็อปของเครื่องคอมพิวเตอร์ผู้เรียน
- 2 เขียนสคริปต์ให้ตัวละครยีราฟและเงาปรากฏตัว โดยวางอยู่ที่ตำแหน่ง (141,-73) และ (-161,-73) ตามลำดับ และปรับขนาดให้เล็กลง 52% ดังรูป



การสัมผัสตัวละคร

- 3 เขียนสคริปต์เพิ่มให้ยีราฟ เพื่อสร้างเงื่อนไขว่า เมื่อใดก็ตามถ้าเงายีราฟมาสัมผัสกับเงาให้ร้องเสียง Rattle แล้วคลิก  ส่งเกตผลลัพธ์



- 4) เพิ่มสคริปต์ต่อจากเดิม โดยเมื่อตัวละครยีราฟสัมผัสกับเงาแล้วให้หายไปจากเวที ณ จุดเดิม
คลิก  สังเกตผลลัพธ์

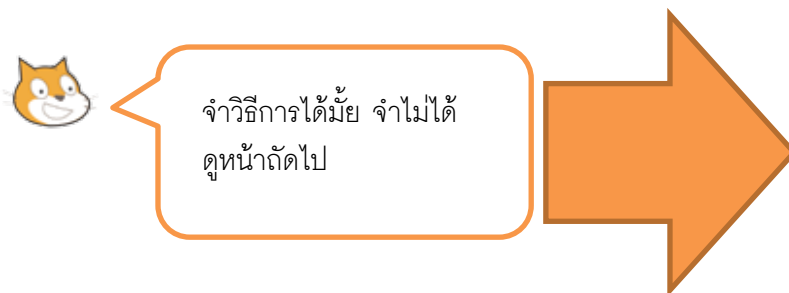


```
forever
  if touching giraffe shadow ? then
    play sound Rattle
    hide
    wait 1 secs
    go to x: 141 y: -73
    show
```

- 5) เพิ่มตัวละครยีราฟอีก 1 ตัว โดยการ duplicate Sprite (คลิกขวาที่ตัวละครยีราฟในส่วน
ของแถบรายการตัวละคร)



- 6) ปรับตัวละครยีราฟตัวใหม่ให้หันอีกด้านโดยการ Edit Costumes



วิธีปรับตัวละครให้หันกลับด้าน

- 1 คลิกเลือก giraffe2
- 2 คลิกแท็บ Costumes
- 3 คลิกปุ่ม Flip left-right



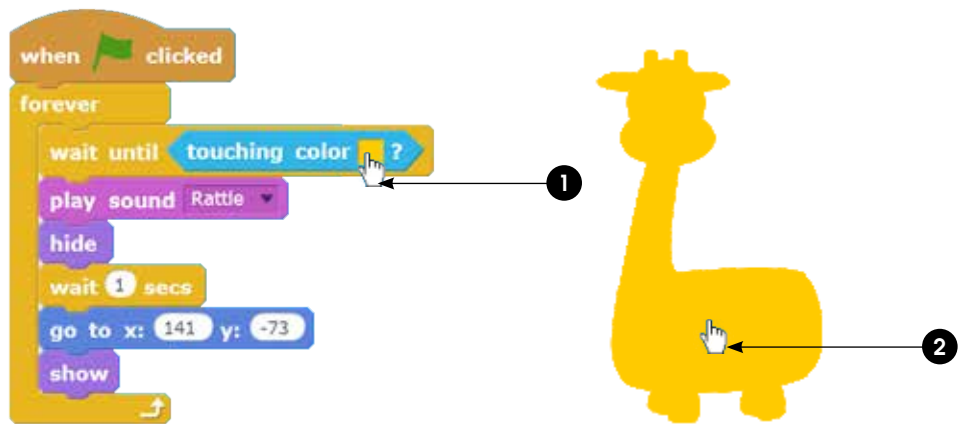
- 7 ตัวละครยี่ราฟตัวใหม่จะมีสคริปต์จากตัวละครยี่ราฟตัวเดิมติดมาด้วย ให้ปรับแก้สคริปต์ดังนี้
7.1 ปรับตำแหน่งของตัวละครยี่ราฟตัวใหม่ให้เริ่มต้นปรากฏตัวที่ตำแหน่ง (25, -73) บนเวที

7.2 เปลี่ยนบล็อก  เป็น 
คือ รอจนกระทั่งเงื่อนไขเป็นจริงและวนซ้ำไปเรื่อยๆ

การสัมผัสสี

7.3 แก๊วสคริปต์จากบล็อก `touching giraffe shadow` เป็น `touching color` โดยการเปลี่ยนสีทำได้ดังนี้

- 1 คลิกที่สี จะปรากฏ  สำหรับดูสี
- 2 เลื่อนเมาส์ไปดูสีที่ตัวละครเงายีราฟบนเวที





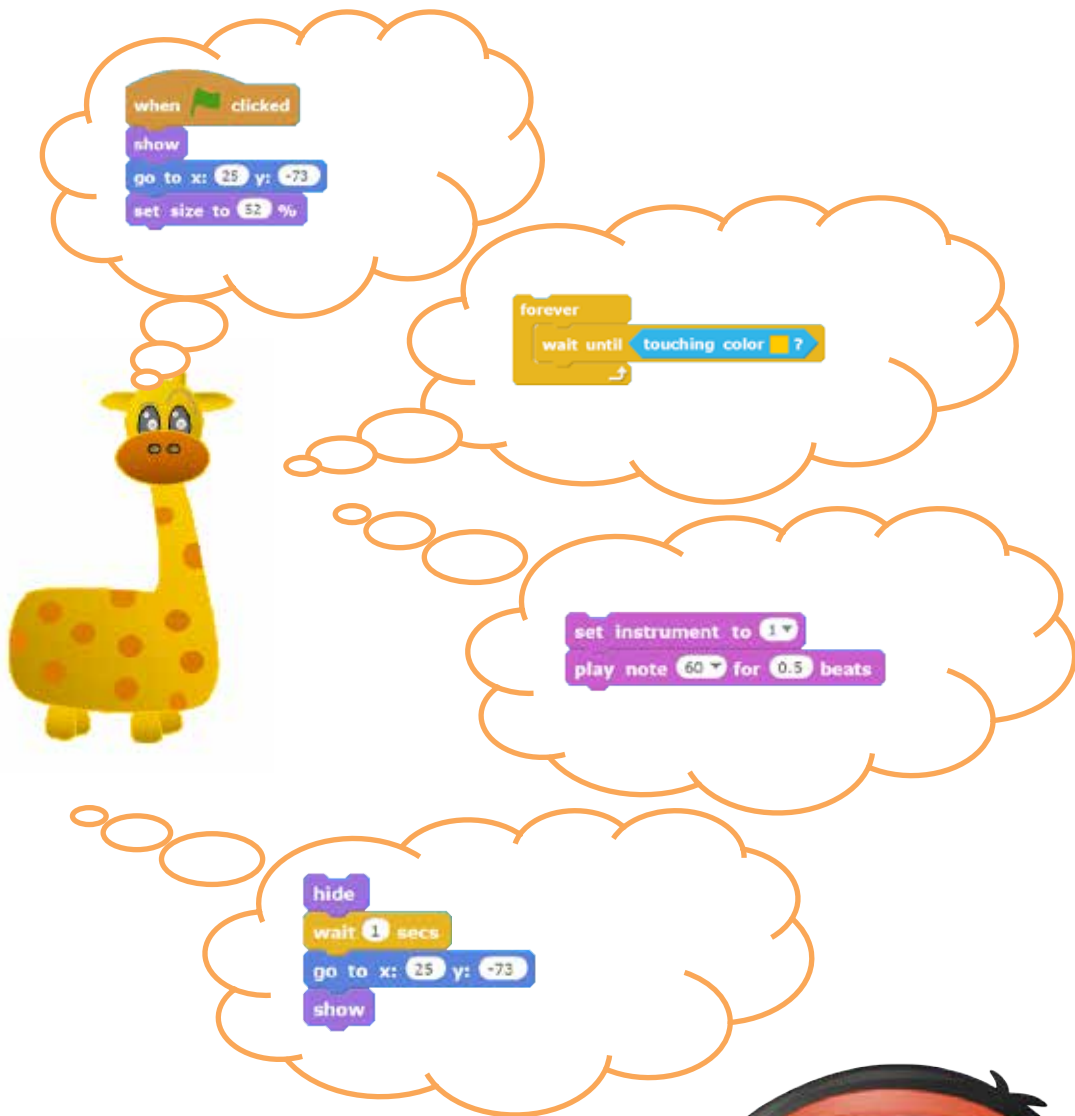
The image shows a Scratch script on the left and a yellow giraffe character on the right. The script starts with a 'when green flag clicked' block, followed by a 'forever' loop containing: 'wait until touching color?' (with a hand cursor icon and a circled '1' pointing to it), 'play sound Rattle', 'hide', 'wait 1 secs', 'go to x: 141 y: -73', and 'show'. The giraffe character has a hand cursor icon on its body with a circled '2' pointing to it.

7.4 เปลี่ยนเสียงเมื่อตัวละครยีราฟตัวใหม่สัมผัสกับสีของเงายีราฟ เป็นเสียงเครื่องดนตรี 1 เล่นโน้ตตัวที่ 60 เป็นเวลา 0.5 จังหวะ (beat)

7.5 เมื่อตัวละครยีราฟตัวใหม่สัมผัสเงาแล้วให้หายไปแล้วมาปรากฏที่ตำแหน่งเดิม



 สรุปคำสั่งในข้อ 7 ลองเขียนสคริปต์ของยีราฟตัวใหม่ให้ถูกต้อง คลิก  สังเกตผลลัพธ์

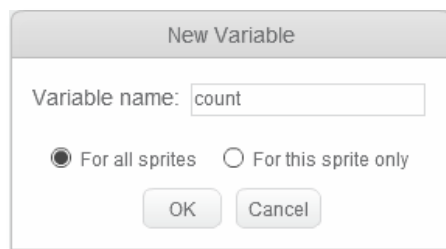


- 8 สร้างตัวแปรชื่อ count สำหรับเก็บข้อมูลจำนวนครั้งในการสัมผัส โดยมีเงื่อนไขดังนี้
 “เมื่อยีราฟตัวแรกสัมผัสกับเงา ให้นำจำนวนการสัมผัสในแต่ละครั้ง”



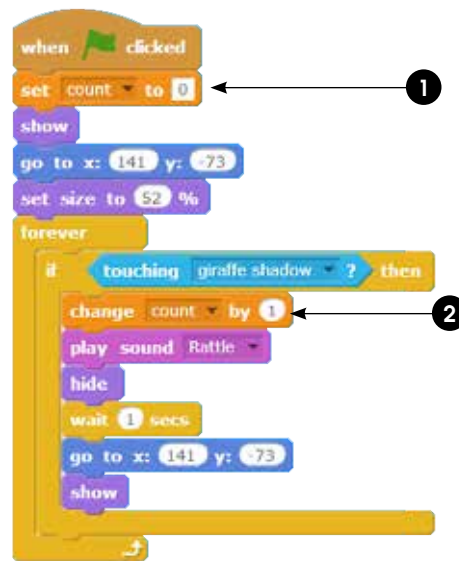
การสร้างตัวแปร **count** เพื่อเก็บค่าการนับ

- 8.1 เลือกกลุ่มบล็อก Data **Data**
 8.2 คลิก **Make a Variable** จากนั้นตั้งชื่อตัวแปรว่า count

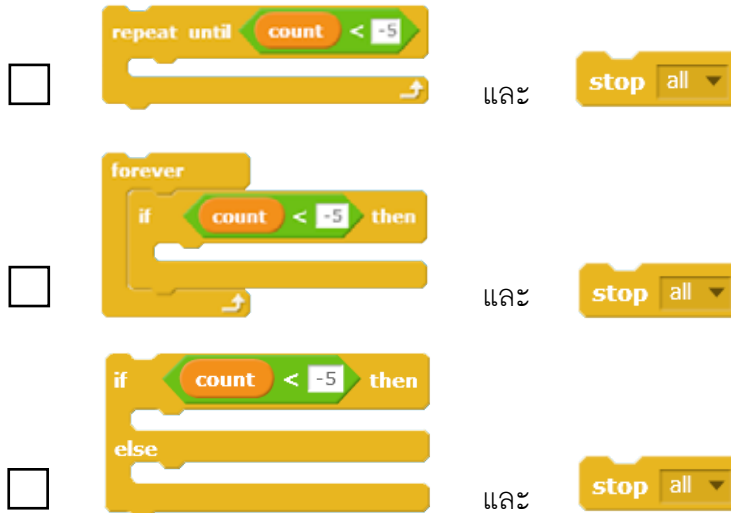


8.3 เขียนสคริปต์เพิ่มให้กับยีราฟตัวแรก

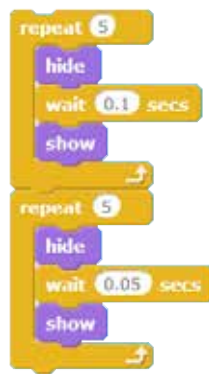
- 1 กำหนดค่าเริ่มต้นให้กับตัวแปร count ให้เท่ากับ 0
- 2 เปลี่ยนค่า count เพิ่มขึ้นครั้งละ 1 หน่วย เมื่อยีราฟตัวแรกมีการสัมผัสกับเงา จากนั้นใส่เครื่องหมาย **count** เพื่อให้แสดงค่าการนับบนเวที



- 8.4 เขียนสคริปต์ให้ยีราฟตัวใหม่ โดยเมื่อยีราฟตัวใหม่สัมผัสกับเงา หรือยีราฟตัวเดิม ให้ลดค่าการนับลง 1 หน่วย และหายไปอยู่ ณ ตำแหน่งเดิม
- 8.5 หากเพิ่มเติมเงื่อนไขว่าต้องการให้หยุดการเล่นสคริปต์ทั้งหมดเมื่อค่าของ count < -5 ต้องแก้ไขโปรแกรมโดยใช้บล็อกในข้อใด ให้เขียนสคริปต์เพื่อทดสอบเงื่อนไขดังกล่าว



- 9 สร้างฟังก์ชัน (function) กระทบให้ตัวละครยิปโป โดยเรียกใช้ฟังก์ชันกระทบ เมื่อตัวละครยิปโปหายไป และเรียกใช้ฟังก์ชันกระทบอีกครั้งก่อนจะปรากฏตัว ณ ตำแหน่งเดิม
- 9.1 ตัวอย่างสคริปต์สั่งให้ตัวละครกระทบ





- 10 ปรับปรุงโปรเจกต์ของตนเองให้มีความสุข และท้าทาย เช่น
- เพิ่มตัวละครและเงา หลากๆ คู่ หรือสร้างเป็นเกมจับคู่
 - สุ่มที่อยู่ของตัวละครแต่ละตัว
 - มีการเปลี่ยนด้านหากทำคะแนนได้ตามที่กำหนด
 - เพิ่มเสียงดนตรีหรือเสียงโต้ตอบ
 - มีการโต้ตอบถามชื่อผู้เล่นเกม





ใบความรู้ที่ 9.1 สัมผัสสนุก

การวนซ้ำแบบมีเงื่อนไข

การทำงานวนซ้ำแบบมีเงื่อนไข (conditional loop) คือ การทำซ้ำจนกว่าเงื่อนไขจะเป็นจริง (repeat until)

บล็อก	ความหมาย	ตัวอย่าง
	ทำซ้ำจนกว่าเงื่อนไขจะเป็นจริง หากเงื่อนไขเป็นจริงจะจบการทำงานภายในลูป และไปทำคำสั่งถัดไป	 <p>เดินไปข้างหน้าทีละ 10 หน่วย จนกว่าจะมีการกดแป้นพิมพ์ Space Bar บนคีย์บอร์ด และจะหยุดการทำงานถึงแม้ว่าจะปล่อยแป้นพิมพ์บน Space bar แล้ว ก็จะไม่เดินต่อ</p>

นอกจากนี้ ยังมีบล็อกที่เป็นคำสั่งใช้งานเฉพาะ โดยมีการตรวจสอบเงื่อนไข เช่น บล็อก wait until


บล็อก	ความหมาย	ตัวอย่าง
	หยุดรอเวลา จนกระทั่งเงื่อนไขเป็นจริง จึงจะทำคำสั่งต่อไป	 <p>หยุดรอเวลา จนกว่าจะมีการกดแป้นพิมพ์ Space Bar แล้วจึงหมุน 15 องศา</p>





กลุ่มบล็อก Sensing

มีบล็อกที่ใช้ตรวจสอบการรับรู้ต่างๆ เช่น การสัมผัสสี การสัมผัสตัวละคร การตรวจสอบตำแหน่งเมาส์ การตรวจสอบการกดแป้นพิมพ์

1. บล็อก **touching mouse-pointer** ? ใช้ตรวจสอบว่าตัวละครนั้นสัมผัสกับ ตัวชี้เมาส์ ขอบเวที หรือตัวละครอื่นหรือไม่ ตัวอย่างการใช้งาน

	<p>คลิก  ผลลัพธ์ที่ได้ คือ เมื่อทดลองเลื่อนเมาส์สัมผัสตัวละคร ตัวละครจะแสดงคำพูดว่า Oh! Touching me แต่เมื่อเลื่อนเมาส์ออกจะแสดงคำพูดว่า Away</p>
------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. บล็อก **touching color** ? ตรวจสอบว่าตัวละครนั้นสัมผัสกับสีที่กำหนดหรือไม่ ตัวอย่างการใช้งาน

	<p>คลิก  แล้วทดลองเลื่อนตัวละครไปทับสีม่วง ผลลัพธ์ที่ได้คือ จะแสดงคำพูดว่า purple เป็นเวลา 2 วินาที</p>
-------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

แบบฝึกหัดเพิ่มเติม



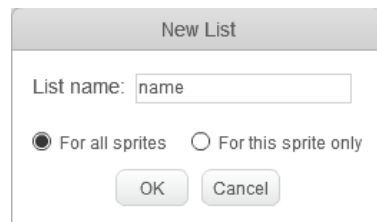
รายการ

1. สร้างโปรเจกต์ใหม่ โดยเพิ่มตัวละครคุณครู และเวที จาก library ดังภาพ

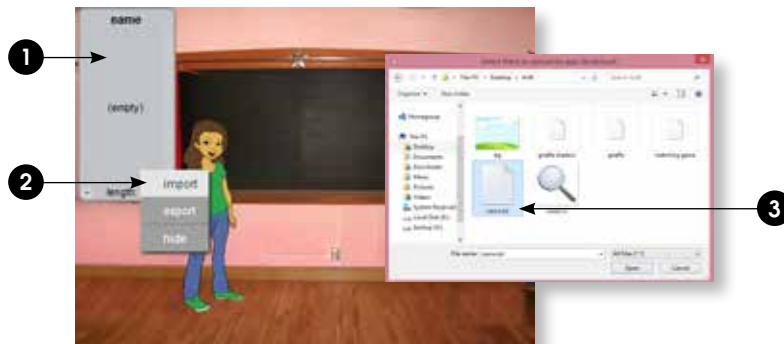



การนำเข้าข้อมูลรายการ

2. เลือกกลุ่มบล็อก Data **Data**
 - 2.1 คลิก **Make a List** สร้างรายการ (List) ชื่อ name




3. นำเข้าข้อมูลรายชื่อนักเรียนโดยวิธีการทำได้ดังนี้
 1. คลิกขวาที่กรอบ List
 2. เลือกเมนู Import
 3. เลือกไฟล์ name.txt ในโฟลเดอร์ชื่อ Act9 บนเดสก์ท็อปของเครื่องคอมพิวเตอร์ผู้เรียน




- เขียนสคริปต์ให้ตัวละครคุณครูพูดทักทายชื่อนักเรียนคนที่ 1 จากรายการ name แล้วคลิก  สังเกตผลลัพธ์




```
when  clicked  
say join สวัสดี item 1 of name for 2 secs
```

- เขียนสคริปต์เพิ่มเติมให้ครูพูดทักทายชื่อนักเรียนทั้งหมดในรายการ name ทีละชื่อ

การค้นหาในรายการ

- เพิ่มตัวละครแว่นขยาย ในโฟลเดอร์ชื่อ Act9 บนเดสก์ท็อปของเครื่องคอมพิวเตอร์ ผู้เรียน และเขียนสคริปต์เพื่อค้นหาชื่อ “สนใจ” ในรายการ name โดยใช้บล็อก  ในการตรวจสอบเงื่อนไข

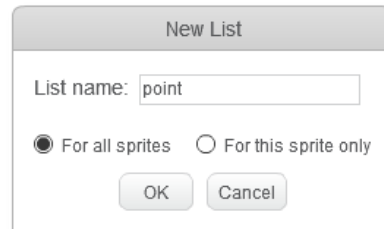


```
when this sprite clicked  
if  then  
say นักเรียนเรียนอยู่ห้องนี้จ้า for 2 secs  
else  
say ไม่มีนักเรียนชื่อในชั้นเรียน for 2 secs
```

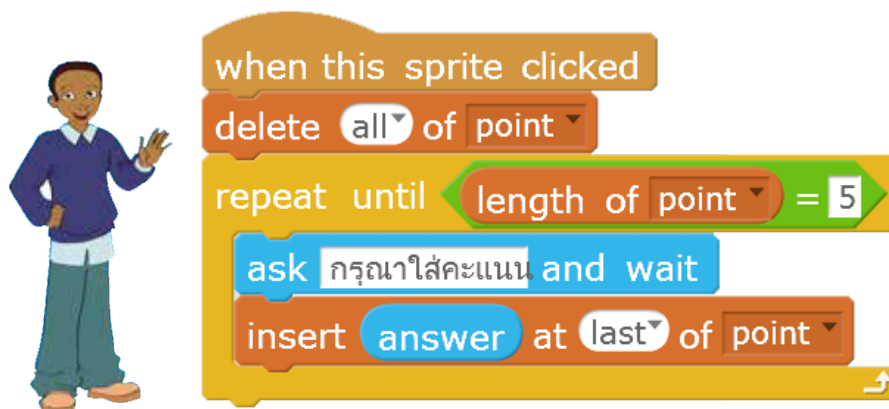
- เพิ่มเติมสคริปต์ให้กับตัวละครแว่นขยาย โดยให้ผู้ใช้สามารถพิมพ์รายชื่อนักเรียนที่ต้องการค้นหาได้เอง

การใส่ค่าข้อมูลเข้ารายการ

- สร้างรายการใหม่ 1 รายการ ชื่อว่า point เพื่อใช้เก็บค่าคะแนน



- เพิ่มตัวละครผู้ชายและเขียนสคริปต์เพื่อรับค่าคะแนน เก็บไว้ในรายการ point จำนวน 5 ครั้ง เมื่อคลิกที่ตัวละครผู้ชาย จากนั้น ทดสอบกรอกคะแนนตามจำนวนครั้งที่กำหนด และสังเกตผลลัพธ์ในกรอบรายการ point



- เพิ่มเติมสคริปต์ให้กับละครผู้ชาย โดยหลังจากรับค่าคะแนนครบตามจำนวนครั้งที่กำหนดแล้วให้คำนวณหาค่าเฉลี่ยจากคะแนนทั้งหมดที่รับเข้ามา
- ปรับปรุงโปรเจกต์ของตนเองให้มีความสนุก และท้าทาย เช่น
 - เพิ่ม ลบ รายชื่อนักเรียนในรายการ name
 - หาค่าต่ำสุด, สูงสุด หรือสถิติอื่นๆ จากรายการ point
 - เรียงลำดับคะแนนจากน้อยไปมากจากรายการ point
 - นำข้อมูลจากรายการ point มาสร้างเป็นกราฟแสดงคะแนน



การจัดการเรียนรู้สะเต็มศึกษา
ด้วยการเขียนโปรแกรม

SCRATCH



กิจกรรมที่ 1 รู้จัก IPST LINK

ระดับชั้น

มัธยมศึกษาตอนปลาย

แนวคิด

โปรแกรม Scratch สามารถเชื่อมต่อกับอุปกรณ์ภายนอกได้โดยทำงานร่วมกับแผงวงจร IPST LINK เพื่ออ่านค่าการวัดในรูปแบบที่หลากหลาย แล้วนำมากำหนดพฤติกรรมให้กับโปรแกรม โดยใช้ชุดบล็อกคำสั่ง    และ  ที่อยู่ในกลุ่ม PicoBoard

แผงวงจร IPST LINK ประกอบด้วยอุปกรณ์วัดเบื้องต้น ได้แก่ ตัวตรวจจับแสง (Light) ตัวตรวจจับเสียง (Sound) ตัวเลื่อน (Slide) และปุ่มกด (Button) นอกจากนี้ยังมีจุดเชื่อมต่อตัวตรวจจับอีก 4 จุด (A-D)

ตัวชี้วัด

- อธิบายกระบวนการรับ-ส่งสารสนเทศและเปรียบเทียบการสื่อสารด้วยสัญญาณแอนะล็อกกับสัญญาณดิจิทัล
- แก้ปัญหาด้วยเทคโนโลยีสารสนเทศ
- สร้างหรือพัฒนาผลิตภัณฑ์ หรือวิธีการ อย่างมีความคิดสร้างสรรค์ ตามกระบวนการเทคโนโลยี โดยการถ่ายทอดความคิดเกี่ยวกับผลิตภัณฑ์เป็นภาพร่างอย่างง่าย หรือภาพร่าง 3 มิติ รวมถึงอาจมีการสร้างแบบจำลองก่อนการสร้างชิ้นงาน หรือการถ่ายทอดความคิดเกี่ยวกับวิธีการออกมาเป็นผังงาน
- บอกค่าความจริงของนิเสธของประพจน์และประพจน์ที่มีตัวเชื่อม “และ” “หรือ” “ถ้า...แล้ว” “ก็ต่อเมื่อ”

สาระการเรียนรู้

- S:** ลักษณะทางกายภาพของสภาพแวดล้อม เช่น ความดังหรือความสว่าง สามารถตรวจจับได้โดยการแปลงสัญญาณให้อยู่ในรูปของสัญญาณไฟฟ้า แล้วแปลงเป็นข้อมูลเพื่อนำไปประมวลผลต่อด้วยคอมพิวเตอร์ก่อนจะกลายเป็นสารสนเทศ ข้อมูลวัดจำแนกได้เป็นสองประเภท คือ ข้อมูลวัดแบบแอนะล็อกและข้อมูลวัดแบบดิจิทัล
- T:** อุปกรณ์ตรวจจับมีเทคโนโลยีที่หลากหลาย จำเป็นต้องทราบถึงหลักการทำงานเพื่อเลือกเทคโนโลยีที่เหมาะสมกับสภาพแวดล้อมที่เราต้องการตรวจจับ

- E:** การออกแบบและพัฒนาผลิตภัณฑ์ให้แก้ไขปัญหาได้ตามที่ต้องการ ให้เกิดประสิทธิภาพและประสิทธิผล ต้องคำนึงถึงความสามารถ ชีตจำกัด และต้นทุนของเทคโนโลยีที่เลือกนำมาใช้
- M:** การกำหนดพฤติกรรมให้กับโปรแกรมเพื่อตอบสนองต่อสภาพแวดล้อมที่ตรวจจับได้สามารถทำได้โดยอาศัยหลักทฤษฎีทางตรรกศาสตร์ ซึ่งอธิบายความสัมพันธ์ของเหตุการณ์ต่างๆ ได้ด้วยตัวเชื่อม “และ” “หรือ” “นิเสธ” และ “ถ้า..แล้ว” รวมถึงใช้สมการทางคณิตศาสตร์เพื่อประเมินสถานะของสภาพแวดล้อมจากค่าที่อ่านได้

จุดประสงค์

1. รู้จักอุปกรณ์แผงวงจร IPST LINK
 - ใช้บล็อกคำสั่งอ่านค่าสถานะความสว่าง ความดังเสียง ตำแหน่งแกนเลื่อน และการกดปุ่ม

แนวการจัดกิจกรรมการเรียนรู้

1. ผู้สอนแบ่งกลุ่มผู้เรียน กลุ่มละ 4 คน
2. ผู้สอนและผู้เรียนร่วมกันอภิปรายเกี่ยวกับอุปกรณ์ที่มีตัวตรวจจับ (Sensor) เช่น เครื่องปรับอากาศ ไฟทางหลวง ประตูร้านสะดวกซื้อหรือห้างสรรพสินค้า ว่าทำงานโดยอาศัยตัวตรวจจับใด (ตัวตรวจจับ อุณหภูมิ แสง สัมผัส) และทำงานอย่างไร (เช่น ไฟส่องสว่างทางหลวงจะมีตรวจจับแสง เมื่อแสงน้อยลงในระดับที่กำหนด วงจรจะสั่งให้เปิดไฟ ในกรณีที่แสงสว่างถึงระดับที่กำหนดวงจรจะสั่งให้ไฟปิด)
3. ผู้สอนแจกชุดแผงวงจร IPST LINK ให้ผู้เรียนกลุ่มละ 1 ชุด พร้อมแนะนำอุปกรณ์
4. ผู้เรียนศึกษาใบความรู้ที่ 1.1 เรื่อง รู้จักชุดแผงวงจร IPST LINK แล้วเชื่อมต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
5. ผู้เรียนศึกษาใบความรู้ที่ 1.2 เรื่อง ทดสอบการเชื่อมต่อ แล้วทำใบงานที่ 1.1 เรื่อง มาตรวจสอบอุปกรณ์กัน
6. ผู้เรียนนำเสนอคำตอบจากการทำใบงานที่ 1.1 โดยเฉพาะแนวทางในการแก้ปัญหาที่เกิดขึ้นในการเชื่อมต่อแผงวงจร และผู้สอนให้คำแนะนำในการแก้ปัญหาคณะอื่น ๆ เพิ่มเติม
7. ผู้เรียนศึกษาใบความรู้ที่ 1.3 เรื่อง แผงวงจร IPST LINK และตัวตรวจจับ
8. ผู้เรียนทำใบงานที่ 1.2 เรื่อง กดกระโดด จากนั้นผู้สอนสุ่มผู้เรียนออกมานำเสนอ แล้วผู้เรียนและผู้สอนร่วมกันสรุป

9. ผู้เรียนทำใบงานที่ 1.3 เรื่องซ้ายหันขวาหัน จากนั้นผู้สอนสุ่มผู้เรียนออกมานำเสนอ แล้วผู้เรียนและผู้สอนร่วมกันสรุป
10. ผู้เรียนทำใบงานที่ 1.4 เรื่องย่อ-ขยาย จากนั้นผู้สอนสุ่มผู้เรียนออกมานำเสนอ แล้วผู้เรียนและผู้สอนร่วมกันสรุป
11. ผู้เรียนทำใบงานที่ 1.5 เรื่องจู้ จู้ อย่าตั้งนะ จากนั้นผู้สอนสุ่มผู้เรียนออกมานำเสนอ แล้วผู้เรียนและผู้สอนร่วมกันสรุป
12. ผู้เรียนทำใบงานที่ 1.6 เรื่องปรับแสง เปลี่ยนสี จากนั้นผู้สอนสุ่มผู้เรียนออกมานำเสนอ แล้วผู้เรียนและผู้สอนร่วมกันสรุป
13. ผู้สอนและผู้เรียนร่วมกันสรุปเกี่ยวกับการทำงานของชุดแผงวงจร IPST LINK และการใช้บล็อกคำสั่งร่วมกับตัวตรวจจับสัญญาณทั้ง 4 แบบเพื่อนำไปประยุกต์ในการสร้างชิ้นงาน

การวัดและประเมินผล

1. การลงมือปฏิบัติ
2. การตอบคำถามในใบงาน

วัสดุอุปกรณ์/สื่อ/แหล่งเรียนรู้

1. ชุดแผงวงจร IPST LINK
2. โปรแกรมสำหรับการติดตั้ง Scratch และ โปรแกรม IPST LINK CHECKER
3. ใบความรู้ที่ 1.1 - 1.3
4. ใบงานที่ 1.1 - 1.6
5. ไฟล์ IPST LINK CHECKER.ab2, angry ghost.mp4, jump and tum.mp4, left-right.mp4 และ spin color.mp4 ซึ่งอยู่ในโฟลเดอร์ MediaScratch/Actstem1

ข้อเสนอแนะ

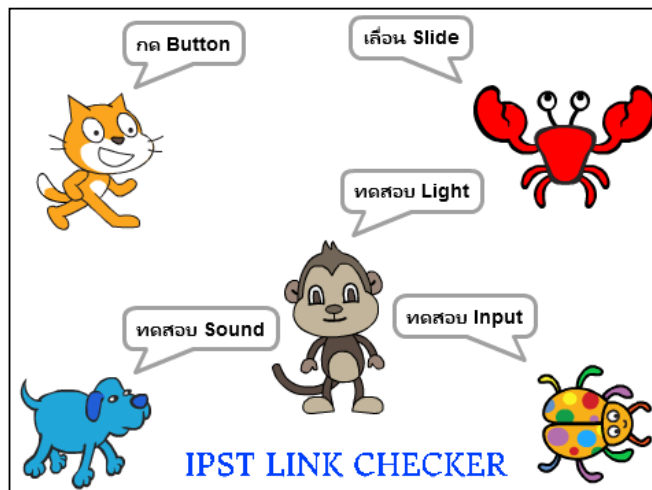
สำหรับ Scratch 2.0 แบบ offline นั้น ยังทำงานร่วมกับ IPST LINK ไม่สมบูรณ์มากนัก โรงเรียนที่มีความพร้อมด้านระบบอินเทอร์เน็ต ควรใช้โปรแกรม Scratch 2.0 แบบ online จะทำให้สามารถใช้งานกลุ่มบล็อก PicoBoard ได้อย่างสมบูรณ์มากขึ้น

ใบงานที่ 1.1 มาตรฐานสอบอุปกรณ์

ชื่อ-สกุล

เลขที่

- 1 ศึกษาใบความรู้ที่ 1.2 เรื่อง ทดสอบการเชื่อมต่อ
- 2 ต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
- 3 เปิดไฟล์ IPST LINK CHECKER.sb2 ในโฟลเดอร์ MediaScratch\ActStem1
- 4 ติดตั้ง PicoBoard
- 5 ทดสอบการทำงานของตัวตรวจจับด้วยไฟล์ IPST LINK CHECKER.sb2 ซึ่งอยู่ในโฟลเดอร์ MediaScratch/ActStem1\example_act1 ดังภาพ



- 6 บันทึกผลที่ได้ลงในตาราง

รายการ	ใช้งานได้	ใช้งานไม่ได้	บันทึกผลการทดสอบ
สายเชื่อมต่อ USB Serial Port			
ตัวตรวจจับแสง (Light)			
ตัวตรวจจับเสียง (Sound)			
สวิตช์กด (Button)			
ตัวเลื่อน (Slide)			
จุดเชื่อมต่อ A			
จุดเชื่อมต่อ B			
จุดเชื่อมต่อ C			
จุดเชื่อมต่อ D			


ใบงานที่ 1.2 กดกระโดด

ชื่อ-สกุล เลขที่

- 1 ศึกษาใบความรู้ที่ 1.3 เรื่อง แผงวงจร IPST LINK และตัวตรวจจับ
- 2 ต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
- 3 เขียนโปรแกรม Scratch ที่ตัวละครแมว ดังนี้

```
when clicked
  go to x: -180 y: -50
  point in direction 90
  forever
    move 1 steps
```

```
when button pressed
  set y to 50
  wait 0.1 secs
  set y to -50
```

- 4 คลิก  แล้วกดปุ่ม Button บนแผงวงจร IPST LINK บันทึกผลที่ได้

.....

.....

- 5 ปรับปรุงสคริปต์ เมื่อกดปุ่ม Button แล้วให้ตัวละครกระโดดหมุนตัว 1 รอบ แล้วเดินต่อ ต้องปรับปรุงสคริปต์อะไรบ้าง (ดูตัวอย่างจากไฟล์วิดีโอ jump and turn.mp4 ในโฟลเดอร์ MediaScratch\ActStem1\example_act1)

.....

.....


.....

.....

.....

.....

.....


หมายเหตุ สำหรับ Scratch 2.0 offline edition ให้คลิกเปลี่ยนการใช้  แต่อย่างไรก็ตามบล็อกนี้สามารถใช้งานได้ปกติสำหรับการใช้งานแบบ online (ข้อมูล ณ มีนาคม 2559)

ใบงานที่ 1.3 ช่ายหัน ขวาทัน

ชื่อ-สกุล เลขที่

- 1 ต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
- 2 ต่อสวิตช์เข้ากับจุดเชื่อมต่อ A และ B
- 3 ลบตัวละครแมวออก
- 4 เพิ่มตัวละคร 1080 Hip-Hop
- 5 เขียนโปรแกรม Scratch ที่ตัวละคร 1080 Hip-Hop ดังนี้

```
when clicked
go to x: 0 y: 0
forever
  if resistance-A sensor value < 50 then
    say left for 1 secs
  if resistance-B sensor value < 50 then
    say right for 1 secs
```

- 6 คลิก  แล้วกดสวิตช์ A จากนั้นบันทึกผลที่ได้
.....
กดสวิตช์ B แล้วบันทึกผลที่ได้
.....

- 7 ปรับปรุงสคริปต์ ให้ตัวละครดังนี้
เมื่อกดสวิตช์ A ให้เปลี่ยนตัวละครเป็น 1080 pop left แล้วพูดว่า "This left" จากนั้นเปลี่ยนตัวละครเป็น 1080 pop front และเมื่อกดสวิตช์ B ให้เปลี่ยนตัวละครเป็น 1080 pop right แล้วพูดว่า "This right" จากนั้นเปลี่ยนตัวละครเป็น 1080 pop front ต้องเพิ่มสคริปต์อะไรบ้าง (ดูตัวอย่างจากไฟล์วิดีโอ left - right.mp4 ในโฟลเดอร์ MediaScratch1\ActStem1\example_act1)


.....
.....
.....
.....

ใบงานที่ 1.4 ย่อ-ขยาย

ชื่อ-สกุล เลขที่

- 1 ต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
- 2 เขียนโปรแกรม Scratch ที่ตัวละครแมว ดังนี้

```
when green flag clicked
  forever loop
    say slider sensor value for 0.1 secs
    set size to slider sensor value %
```

- 3 คลิก  แล้วเลื่อนตัวเลื่อน บันทึกผลที่ได้
- 4 ปรับปรุงสคริปต์ เมื่อเลื่อนตัวเลื่อนแล้วให้ตัวละครเกิดผลตรงกันข้าม ต้องปรับปรุงสคริปต์อะไรบ้าง (ดูตัวอย่างจากไฟล์วิดีโอ zoom zoom.mp4 ในโฟลเดอร์ Act_1\example_act1)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....




ใบงานที่ 1.5 จี จี อย่างดงนะ

ชื่อ-สกุล เลขที่

- 1 ต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
- 2 ลบตัวละครแมวและเพิ่มตัวละคร ghost2
- 3 เขียนโปรแกรม Scratch ที่ตัวละคร ghost2 ดังนี้

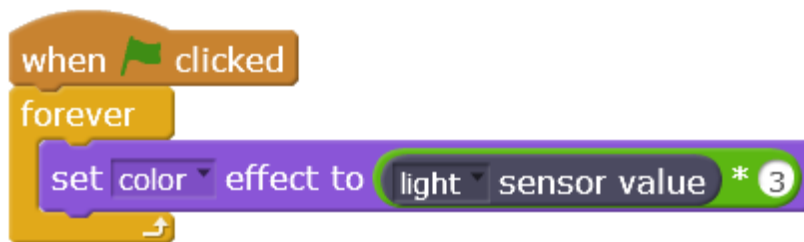
```
when clicked
  forever
    set a to sound sensor value
    if a > 50 then
      switch costume to costume1
      say a for 1 secs
```



- 4 คลิก  ล้างและส่งเสียงดังใกล้ตัวตรวจจับเสียงของแผงวงจร IPST LINK สังเกตแล้วบันทึกผลที่ได้
- 5 ปรับปรุงสคริปต์ เมื่อมีเสียงดังแล้วให้ตัวละครหมุนแบบสุ่มพร้อมทั้งเคลื่อนที่ไปข้างหน้าเมื่อชนขอบก็ให้สะท้อน และเคลื่อนที่ต่อไปอย่างน้อย 3 วินาที จากนั้นให้กลับมากลางจอภาพ ต้องเพิ่มสคริปต์อะไรบ้าง (ดูตัวอย่างจากไฟล์วิดีโอ angry_ghost.mp4 ในโฟลเดอร์ MediaScratch\AcStem1\example_act1)

ใบงานที่ 1.6 ปรับแสง เปลี่ยนสี

ชื่อ-สกุล เลขที่

- 1 ต่อแผงวงจร IPST LINK กับเครื่องคอมพิวเตอร์
- 2 เขียนโปรแกรม Scratch ที่ตัวละครแมว ดังนี้



- 3 คลิก  แล้วลองใช้มือบังตัวตรวจจับแสงของแผงวงจร IPST LINK แล้วบันทึกผลที่ได้
.....
.....
- 4 เพิ่ม  ลงในสคริปต์ เพราะอะไร
.....
.....
- 5 ปรับปรุงสคริปต์ เมื่อมีแสงสว่างตั้งแต่ 50 ขึ้นไป ให้ตัวละครเปลี่ยนสีแล้วหมุนไปทางซ้าย แต่ถ้ามีแสงสว่างน้อยกว่า 50 ให้ตัวละครเปลี่ยนสีแล้วหมุนไปทางขวา ต้องปรับปรุงสคริปต์ อะไรบ้าง (ดูตัวอย่างจากไฟล์วิดีโอ spin color ในโฟลเดอร์ MediaScratch\ActStem1\example_act1)
.....
.....
.....
.....
.....
.....

ใบความรู้ที่ 1.1 รู้จักชุดแผงวงจร IPST LINK

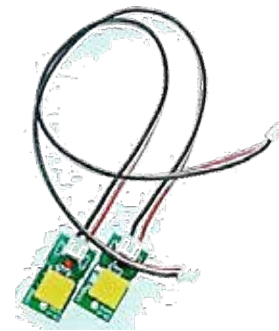
ชุดแผงวงจร IPST LINK ประกอบด้วย



1. แผงวงจร



2. สายต่อ UCON-2F
สำหรับเชื่อมต่อกับคอมพิวเตอร์
ผ่านพอร์ต USB

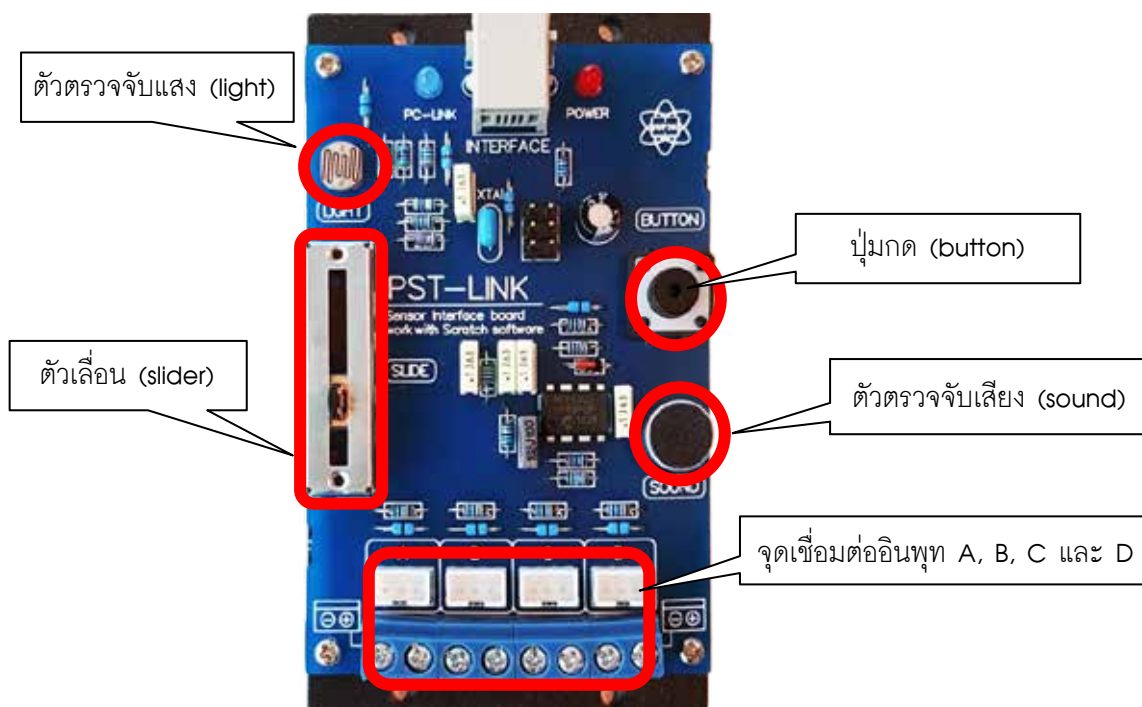


3. สวิตช์

ตัวตรวจจับสนิทพร้อมใช้งานบนแผงวงจร ดังนี้

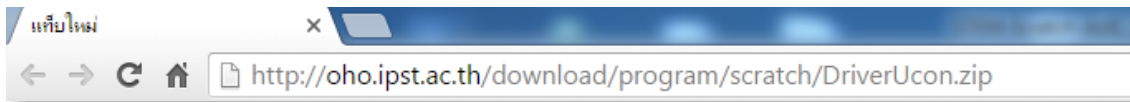
- ตัวตรวจจับสนิทแสง (light) เป็นตัวต้านทานแปรค่าตามแสงหรือ LDR
- ตัวตรวจจับสนิทเสียง (sound) ใช้คอนเดนเซอร์ ไมโครโฟน
- ตัวเลื่อน (slide) เป็นตัวต้านทานปรับค่าแบบแกนเลื่อน
- สวิตช์กด (button) ใช้สวิตช์กดติดปล่อยดับ

มีจุดเชื่อมต่ออินพุตสำหรับต่อตัวตรวจจับสนิทเพิ่มเติมและสามารถรับสัญญาณแรงดันไฟตรงอีก 4 ช่อง คือ A, B, C และ D รับแรงดันได้สูงสุด 5V ใช้จุดต่อแบบ JST 3 ขา สามารถรองรับกับตัวตรวจจับสนิทอื่น และมีจุดต่อแบบเทอร์มินอลขั้วสกรู สามารถต่อสายสัญญาณหรือขาของตัวตรวจจับสนิทได้ ดังภาพ

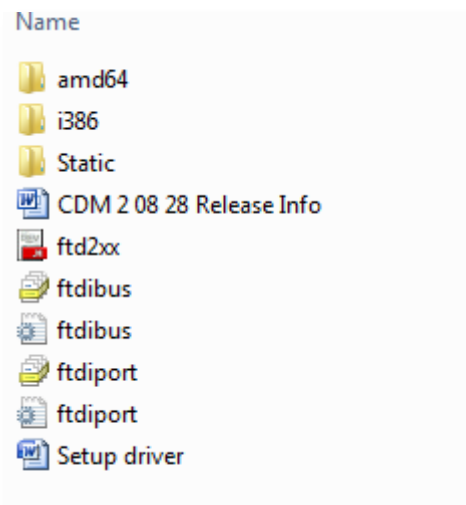


ใบความรู้ที่ 1.2 ทดสอบการเชื่อมต่อ

1. เปิดไฟล์ไดรเวอร์ Ucon-2f ในไฟล์เดอร์ Software หรือดาวน์โหลดจากเว็บไซต์ <http://oho.ipst.ac.th/download/program/scratch/DriverUcon.zip>

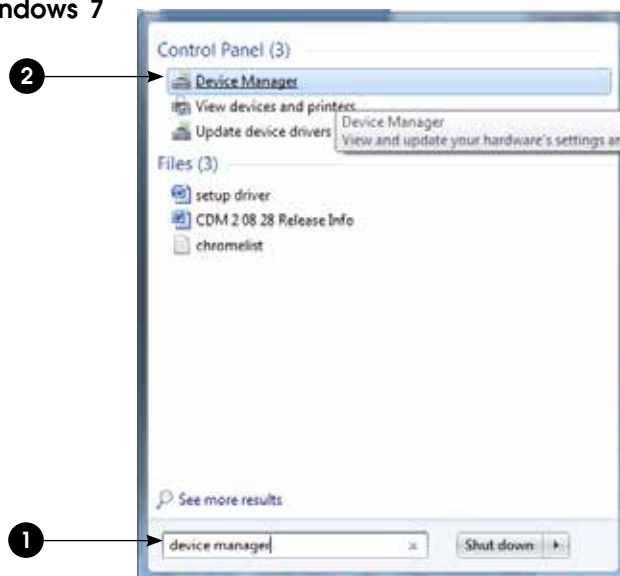


เมื่อแตกไฟล์ DriverUcon.zip จะได้ไฟล์เดอร์ Driver Ucon ซึ่งมีไฟล์และไฟล์เดอร์ ดังนี้



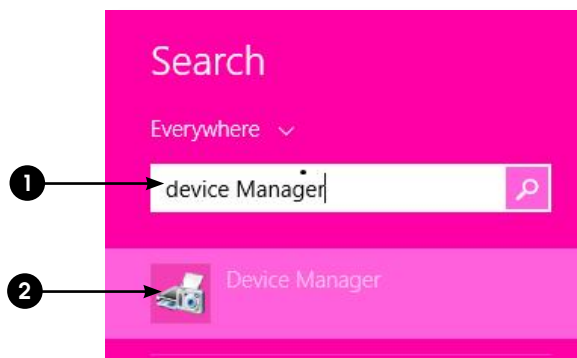
2. เชื่อมต่อแผงวงจร IPST LINK เข้ากับพอร์ต USB ของคอมพิวเตอร์โดยใช้สาย UCON-2F จากนั้นเปิดโปรแกรม Device Manager โดย

สำหรับ Windows 7

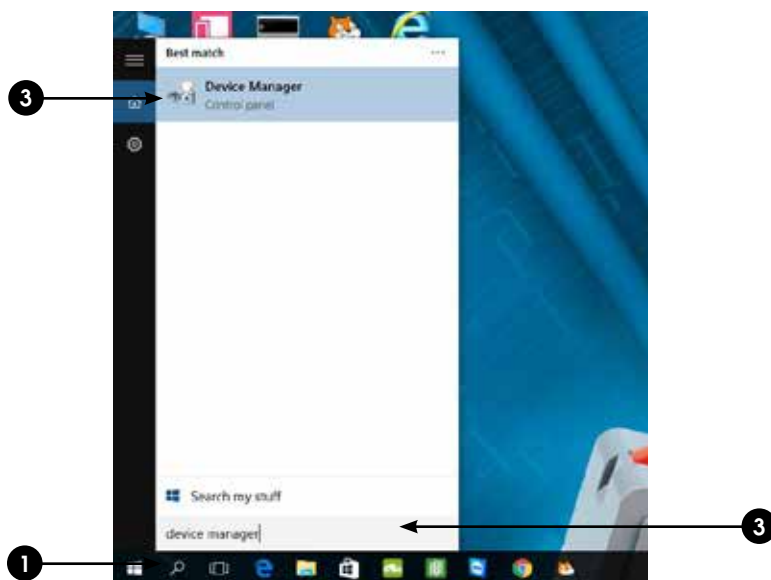



1. คลิกที่ สตาร์ทเมนู ในช่อง Search program and files พิมพ์คำว่า Device Manager
2. คลิกที่ Device Manager

สำหรับ Windows 8

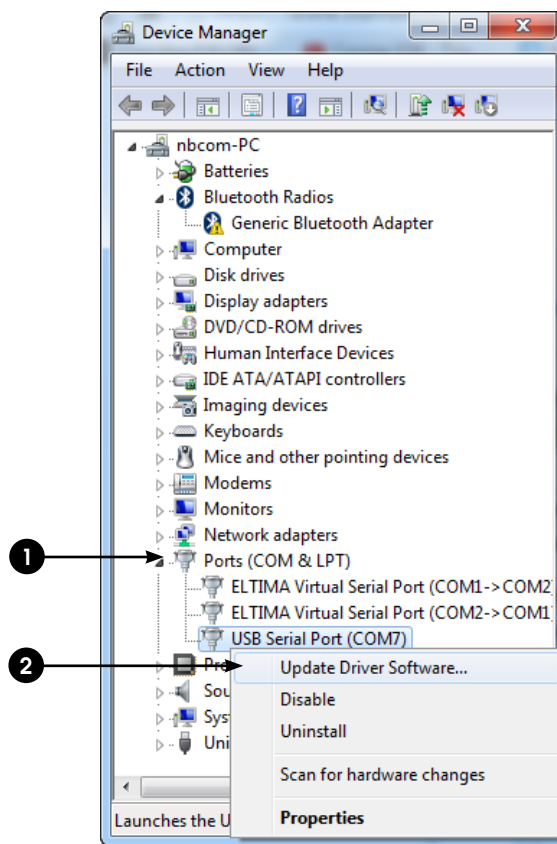


- 1 กดแป้น **Windows** + **Q** แล้วพิมพ์คำว่า Device Manager ในช่อง Search
 - 2 คลิกที่ Device Manager
- สำหรับ Windows 10

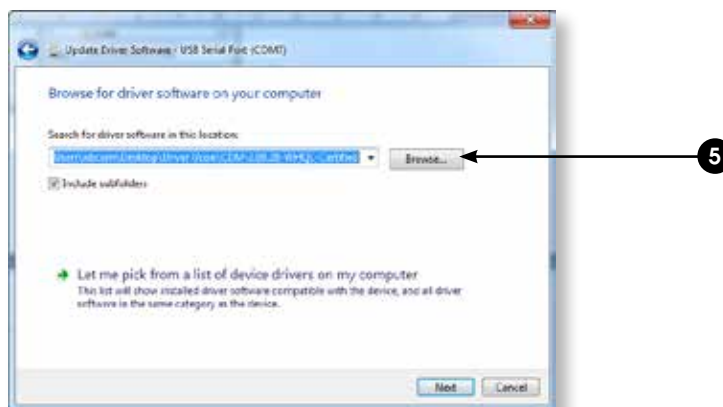


- 1 คลิกปุ่ม 
- 2 พิมพ์คำว่า Device Manager ในช่อง Search
- 3 คลิกที่ Device Manager

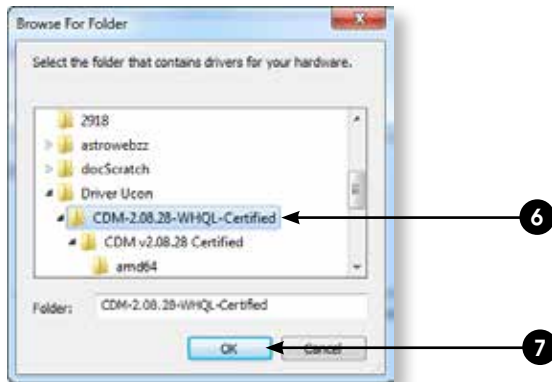
3. เมื่อเข้าสู่ Device Manager ให้ทำตาม ขั้นตอนดังนี้

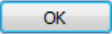
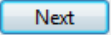


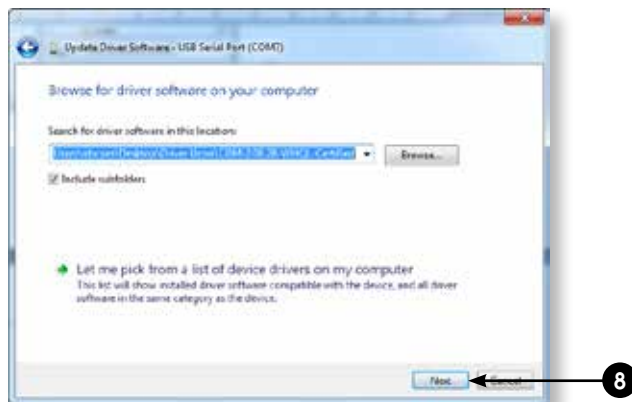
1. คลิกที่สัญลักษณ์ ▶ หน้า Port (COM & LPT)
2. คลิกขวาที่ USB Serial Port (ตัวเลขหลังคำว่า COM แต่ละเครื่องอาจจะไม่เหมือนกัน)
3. คลิกที่ Update Driver Software...
4. คลิกที่ Browse my computer for driver software
Locate and install driver software manually.

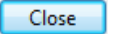


5. คลิก



- 6 เลือกโฟลเดอร์ของไดรเวอร์ จากข้อ 1
- 7 คลิกปุ่ม 
- 8 คลิกปุ่ม  และรอจนติดตั้งเสร็จ

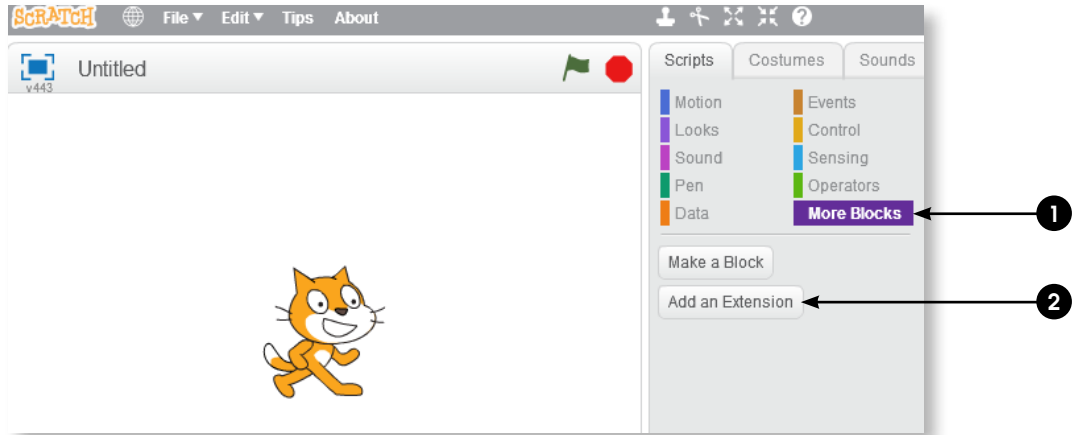


- 9 ถ้าติดตั้งสมบูรณ์จะปรากฏหน้าต่างดังรูป คลิกที่ปุ่ม  และปิด Device Manager

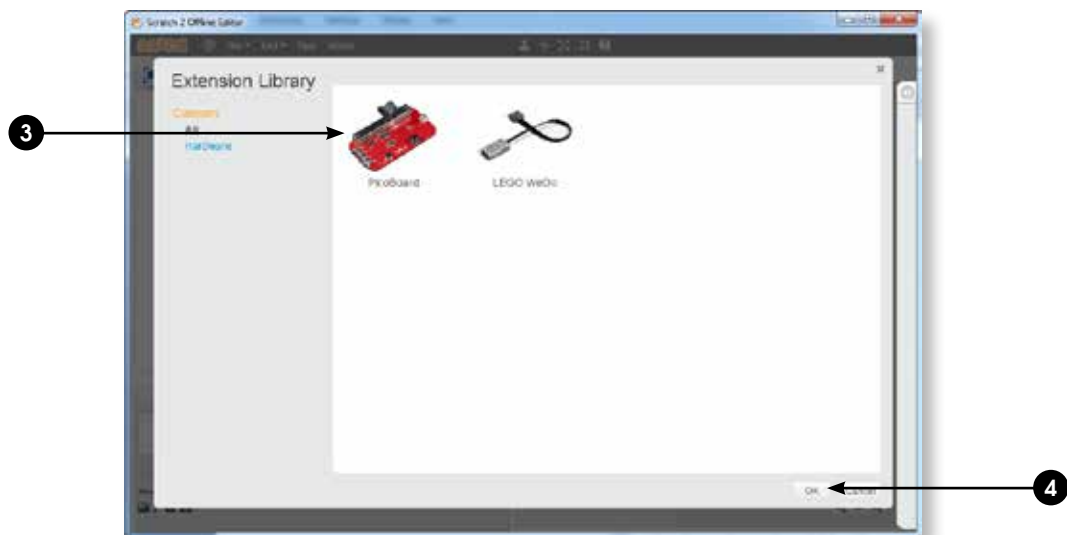


4. รีสตาร์ทคอมพิวเตอร์

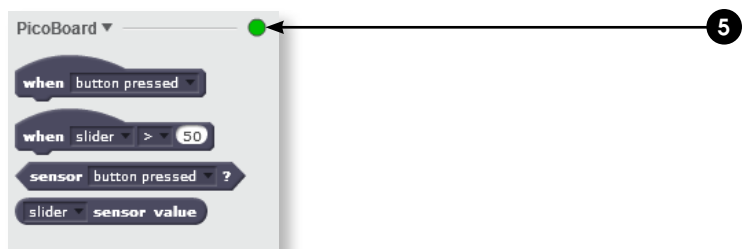
5. เปิดโปรแกรม scratch 2.0 ขึ้นมาแล้วทำตามขั้นตอน ดังนี้



- 1. คลิก **More Blocks**
- 2. คลิก **Add an Extension**

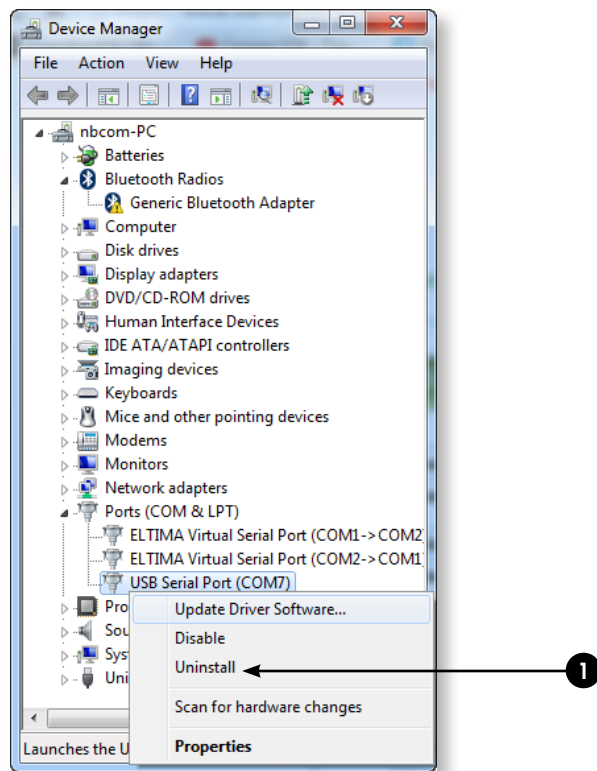


- 3. คลิกเลือกที่ PicoBoard
- 4. คลิก **OK**



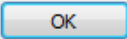
- 5. สังเกตสัญลักษณ์วงกลมจะเป็นสีเขียว แสดงว่าการเชื่อมต่อเสร็จสมบูรณ์

6. กรณีที่มีปัญหา สัญลักษณ์ไม่เป็นสีเขียว มีวิธีการแก้ไขโดย ไปที่ Device Manager อีกครั้ง และคลิกขวาที่ USB Serial Port



1. คลิก Uninstall



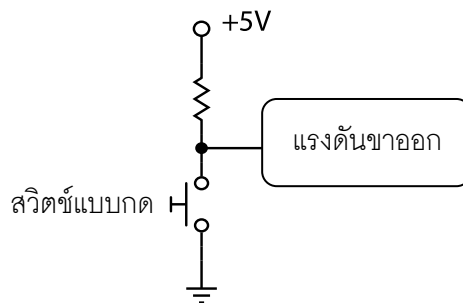
2. คลิกปุ่ม  แล้วรีสตาร์ทเครื่อง โดยให้เริ่มทำจากขั้นที่ 2 อีกครั้ง

ใบความรู้ที่ 1.3 แผงวงจร IPST LINK และตัวตรวจจับ

ตัวตรวจจับ หรือเซนเซอร์ (sensor) เป็นอุปกรณ์ที่ใช้สำหรับเปลี่ยนสถานะทางกายภาพ เช่น ความสว่าง ความดัง ความชื้น อุณหภูมิ ให้เป็นสัญญาณไฟฟ้า เพื่อใช้เป็นข้อมูลนำเข้าในการประมวลผล แผงวงจร IPST LINK ช่วยให้สคริปต์ใน Scratch รับรู้สถานะผ่านตัวตรวจจับ ได้แก่ ปุ่มกด ตัวเลื่อน ตัวตรวจจับแสง ตัวตรวจจับเสียง และจุดเชื่อมต่อตัวตรวจจับอีก 4 จุด

1. ปุ่มกด (Button)

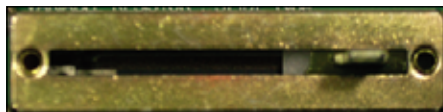
ปุ่มกด หรือเซนเซอร์สัมผัส เป็นอุปกรณ์ที่ทำหน้าที่เปลี่ยนการสัมผัสให้เป็นสัญญาณทางไฟฟ้า เซนเซอร์สัมผัสที่นิยมใช้ ได้แก่ สวิตช์แบบกลไก (mechanical switch) ซึ่งเป็นอุปกรณ์แบบกลไกที่ทำหน้าที่รับแรงกด หน้าสัมผัสของปุ่มจะนำกระแสไฟฟ้า ทำให้สามารถตรวจสอบสถานะจากสัญญาณไฟฟ้าได้ ปุ่มกดบนแผงวงจร IPST LINK มีการเชื่อมต่อเข้ากับวงจรดังภาพ



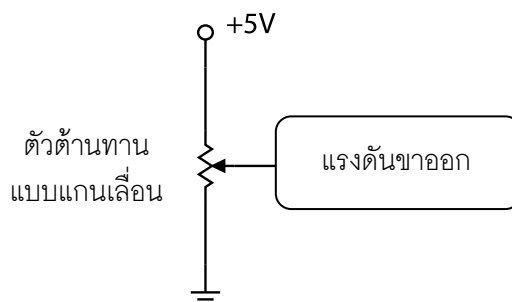
เมื่อสวิตช์ไม่ถูกกด จุดที่วัดแรงดันจะมีศักย์ไฟฟ้า 5 โวลต์ แต่ถ้าสวิตช์ถูกกด จุดวัดแรงดันจะถูกเชื่อมเข้ากับกราวด์ ทำให้มีศักย์ไฟฟ้าเป็น 0 โวลต์ ไมโครคอนโทรลเลอร์บนแผงวงจร IPST LINK จึงสามารถใช้สถานะที่แตกต่างกันนี้ ตรวจสอบสถานะการกดปุ่มของสวิตช์เพื่อส่งข้อมูลให้กับโปรแกรม Scratch ได้

2. ตัวต้านทานปรับค่าได้แบบแกนเลื่อน หรือตัวเลื่อน (slide)

ตัวเลื่อนใช้เปลี่ยนค่าความต้านทาน โดยการเปลี่ยนตำแหน่งของจุดบนแกนเลื่อนให้เป็นสัญญาณทางไฟฟ้า ดังภาพ



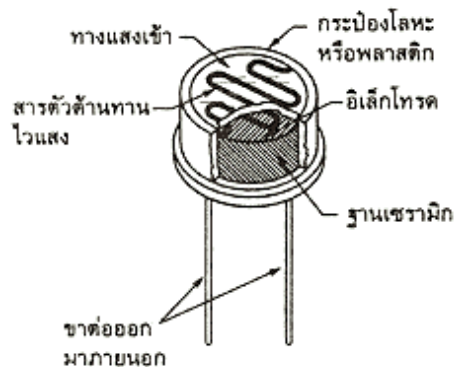
วงจรที่ใช้อ่านสถานะจากตัวต้านทานปรับค่าได้บนแผงวงจร IPST LINK เป็นดังภาพ



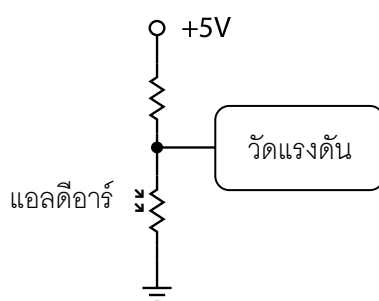
ความต่างศักย์ที่ตกคร่อมตัวต้านทานปรับค่าได้ คือ 5 โวลต์ ส่วนค่าที่วัดได้จากจุดเลื่อนจะมีศักย์ไฟฟ้าที่แปรผันตรงกับตำแหน่งบนแกนเลื่อน ทำให้ไมโครคอนโทรลเลอร์บนแผงวงจร IPST LINK ตรวจสอบสัญญาณและส่งข้อมูลตำแหน่งให้กับโปรแกรม Scratch ได้ โดยจะมีค่าตั้งแต่ 0 ถึง 100

3. ตัวตรวจจับแสง

ตัวตรวจจับแสงที่ใช้กันเป็นที่แพร่หลาย ได้แก่ แอลดีอาร์ (Light Dependent Resistor: LDR) ทำหน้าที่เปลี่ยนระดับความเข้มของแสงให้กลายเป็นค่าความต้านทานทางไฟฟ้า เมื่อความเข้มของแสงมาก จะให้ค่าความต้านทานน้อย แอลดีอาร์ยังมีชื่อเรียกอื่น เช่น โฟโตคอนดักทีฟเซลล์ (photoconductive cell) หรือตัวต้านทานไวแสง (Light Sensitive Resistor: LSR) ส่วนใหญ่จะทำมาจากสารกึ่งตัวนำ ดังภาพ



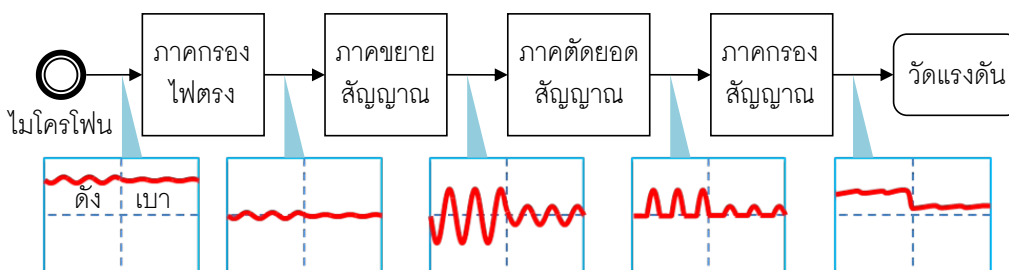
อุปกรณ์แอลดีอาร์บนแผงวงจร IPST LINK มีการเชื่อมต่อเข้ากับวงจร ดังภาพ



เมื่อแสงตกกระทบบนมาก แอลดีอาร์จะมีค่าความต้านทานต่ำลง มีผลให้ความต่างศักย์ที่ตกคร่อมแอลดีอาร์ต่ำลงด้วย แต่ถ้าแสงตกกระทบน้อย แอลดีอาร์จะมีค่าความต้านทานสูงขึ้น ทำให้ความต่างศักย์ที่ตกคร่อมแอลดีอาร์สูงขึ้นตาม อย่างไรก็ตาม ไมโครคอนโทรลเลอร์บนแผงวงจร IPST LINK จะประมวลผลค่าแล้วส่งข้อมูลไปยังโปรแกรม Scratch โดยได้ค่า 0 เมื่อมีแสงน้อยที่สุด และค่า 100 เมื่อมีแสงสว่างที่สุด

4. ตัวตรวจจับเสียง

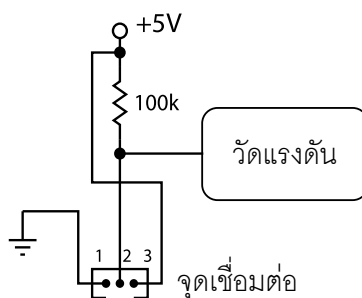
ตัวตรวจจับเสียงที่นิยมใช้ ได้แก่ คอนเดนเซอร์ไมโครโฟน (condenser microphone) ทำหน้าที่เปลี่ยนคลื่นเสียง (sound wave) จากแหล่งกำเนิดเสียง เช่น เสียงพูด เสียงเพลง เสียงดนตรี ให้เป็นสัญญาณไฟฟ้า มีหลักการคือ เมื่อคลื่นเสียงกระทบแผ่นลั่น หรือ แผ่นไดอะแฟรมจะทำให้เกิดการเปลี่ยนแปลงของกระแสไฟฟ้าที่ถูกส่งผ่านสายนำสัญญาณไปยังเครื่องขยายเสียง ไมโครโฟนที่ประกอบด้วยขดลวดและแม่เหล็ก เมื่อเสียงกระทบตัวรับในไมโครโฟนจะทำให้ขดลวดลั่นสะเทือนตัดกับสนามแม่เหล็ก เกิดเป็นสัญญาณไฟฟ้าขึ้น วงจรวัดระดับเสียงบนแผงวงจร IPST LINK มีการทำงาน ดังแผนภาพ



เสียงที่ถูกส่งเข้าไปยังไมโครโฟนจะก่อให้เกิดการเปลี่ยนแปลงของกระแสไฟฟ้าที่ไหลผ่านตัวไมโครโฟน โดยมีอัตราการเปลี่ยนแปลงตามความถี่ของเสียง และปริมาณการเปลี่ยนแปลงตามความดังของเสียง อย่างไรก็ตามกระแสไฟฟ้าที่เปลี่ยนแปลงนี้อยู่ในปริมาณที่ต่ำมากจึงต้องมีการกรองสัญญาณไฟตรงแล้วส่งสัญญาณกระแสสลับเข้าไปยังภาคขยายสัญญาณ สัญญาณผลลัพธ์จะถูกส่งเข้าสู่วงจรตัดยอดสัญญาณและถูกกรองให้เรียบเป็นลำดับสุดท้าย ก่อนที่สัญญาณจะถูกวัดแรงดันโดยไมโครคอนโทรลเลอร์ สัญญาณที่วัดได้จึงไม่ใช่ตัวแทนของสัญญาณเสียงโดยตรง แต่แทนระดับความดังของเสียง โดยค่าที่อ่านได้จาก Scratch จะให้ค่า 0 เมื่อไม่มีเสียงจนถึง 100 เมื่อเสียงดังมาก

5. จุดเชื่อมต่อตัวตรวจจับอื่นๆ

แผงวงจร IPST LINK มีจุดที่สามารถนำตัวตรวจจับชนิดอื่น ๆ มาเชื่อมต่อได้ 4 จุด โดยมีวงจรภายในดังภาพ



จากภาพ จุดวัดแรงดันเป็นการวัดศักย์ไฟฟ้าที่ตกคร่อมขา 1 และ 2 ที่จุดเชื่อมต่อ ซึ่งขึ้นอยู่กับความต้านทานของอุปกรณ์ที่นำมาต่อคร่อมที่ขาคุณนี้ ศักย์ไฟฟ้าที่วัดได้จะถูกคำนวณเป็นปริมาณที่แทนค่าความต้านทาน โดยค่า 0 หมายถึง ไม่มีความต้านทาน (ขาคุณนี้เชื่อมต่อกัน) และ 100 หมายถึง ความต้านทานเป็นอนันต์ (ขาคุณนี้ไม่เชื่อมต่อกัน)

กิจกรรมที่ 2 ประยุกต์ใช้งานตัวตรวจจับ

ระดับชั้น

มัธยมศึกษาตอนปลาย

แนวคิด

ตัวตรวจจับ (Sensor) เป็นอุปกรณ์ที่ใช้รับข้อมูลจากภายนอกแล้วส่งให้โปรแกรมประมวลผลเพื่อทำงาน ตัวตรวจจับจึงสามารถนำมาประยุกต์ใช้รับค่าแทนการป้อนข้อมูลผ่านเมาส์และแป้นพิมพ์ โดยเฉพาะอย่างยิ่งในสถานการณ์ที่ต้องการให้โปรแกรมคอมพิวเตอร์ตอบสนองต่อการเปลี่ยนแปลงของสภาพแวดล้อม การเลือกใช้ตัวตรวจจับนั้นจึงขึ้นอยู่กับข้อมูลภายนอกที่ต้องการรับเข้าสู่โปรแกรม นอกจากนี้ยังสามารถนำตัวตรวจจับหลายตัวมาใช้งานร่วมกันเพื่อสร้างชิ้นงานที่ได้

ตัวชี้วัด

- แก้ปัญหาด้วยเทคโนโลยีสารสนเทศ
- สร้างหรือพัฒนาผลิตภัณฑ์หรือวิธีการอย่างมีความคิดสร้างสรรค์ตามกระบวนการเทคโนโลยี โดยการถ่ายทอดความคิดเกี่ยวกับผลิตภัณฑ์เป็นภาพร่างอย่างง่าย หรือ ภาพร่าง 3 มิติ และภาพฉาย อาจสร้างแบบจำลองก่อนสร้างชิ้นงาน หรือถ่ายทอดความคิดเกี่ยวกับวิธีการเป็นผังงาน
- บอกค่าความจริงของนิเสธของประพจน์และประพจน์ที่มีตัวเชื่อม “และ” “หรือ” “ถ้า...แล้ว” “ก็ต่อเมื่อ”

สาระการเรียนรู้

- S:** การแก้ปัญหาเป็นกระบวนการที่ต้องใช้ความรู้ ทักษะ ความเข้าใจ ซึ่งบางปัญหาอาจมีแนวทางในการแก้ปัญหาคือได้หลายวิธี
- T:** อุปกรณ์ตรวจจับมีเทคโนโลยีที่หลากหลาย จำเป็นต้องทราบถึงหลักการทำงานเพื่อเลือกเทคโนโลยีที่เหมาะสมกับสภาพแวดล้อมที่เราต้องการตรวจจับ
- E:** การสร้างแบบจำลองมีประโยชน์เพื่อสื่อสารความคิดให้ผู้อื่นเข้าใจหรือใช้ทดสอบการทำงานของชิ้นงาน
- M:** ค่าความจริงของนิเสธของประพจน์และประพจน์ที่มีตัวเชื่อม “และ” “หรือ” “ถ้า...แล้ว” และใช้สมการทางคณิตศาสตร์เพื่อแปลผลค่าที่ได้รับจากตัวตรวจจับให้เป็นสถานะของสภาพแวดล้อมได้

จุดประสงค์

ประยุกต์ใช้ตัวตรวจจับที่หลากหลายมาทำงานประสานกันในโปรแกรม Scratch

แนวการจัดกิจกรรมการเรียนรู้

1. แบ่งกลุ่มผู้เรียนกลุ่มละ 2 คน
2. ผู้สอนกระตุ้นผู้เรียนด้วยคำถามเกี่ยวกับสิ่งของในชีวิตประจำวันที่มีการใช้ตัวตรวจจับ
3. ผู้เรียนเชื่อมต่อแผงวงจร IPST LINK กับคอมพิวเตอร์พร้อมทั้งตรวจสอบการทำงานให้สามารถเชื่อมต่อกับคอมพิวเตอร์ได้
4. ผู้สอนอธิบายแนวคิดของใบงานที่ 2.1 เรื่อง E-Librarian จากนั้นผู้เรียนลงมือปฏิบัติ โดยผู้สอนคอยให้คำแนะนำ กระตุ้นให้ปรับปรุงและพัฒนาชิ้นงานตามแนวคิดของตนเอง
5. ผู้สอนสุ่มกลุ่มของผู้เรียนออกมานำเสนอชิ้นงาน
6. ผู้สอนอธิบายแนวคิดของใบงานที่ 2.2 เรื่อง Smart Home จากนั้นผู้เรียนลงมือปฏิบัติ โดยผู้สอนคอยให้คำแนะนำ กระตุ้นให้ปรับปรุงและพัฒนาชิ้นงานตามแนวคิดของตนเอง
7. ผู้สอนสุ่มกลุ่มของผู้เรียนออกมานำเสนอชิ้นงาน
8. ผู้เรียนและผู้สอนสรุปกิจกรรมการประยุกต์ใช้งานตัวตรวจจับ

การวัดและประเมินผล

1. ตรวจชิ้นงาน
2. การตอบคำถามในใบงาน

วัสดุอุปกรณ์/สื่อ/แหล่งเรียนรู้

1. โปรแกรม Scratch 2.0 Offline Edition
2. ใบงานที่ 2.1 และ 2.2
3. ชุดแผงวงจร IPST LINK

หมายเหตุ

1. ความรู้ที่จำเป็น : การเขียนโปรแกรม Scratch โดยใช้ชุดแผงวงจร IPST LINK
2. ตัวอย่างไฟล์ใบงานที่ 2.1 อยู่ในโฟลเดอร์ MediaScratch\ActStem2\solution_act2 ชื่อ elibrarian.sb2
3. ตัวอย่างไฟล์ใบงานที่ 2.2 อยู่ในโฟลเดอร์ MediaScratch\ActStem2\solution_act2 ชื่อ smarhome.sb2

ใบงานที่ 2.1 E-Librarian

ชื่อ-สกุล เลขที่

- 1) พิจารณาปัญหาที่พบในห้องสมุดเช่น ผู้ใช้มักจะรับประทานอาหาร ส่งเสียงดัง และใช้ห้องสมุดเกินเวลาที่กำหนด ให้ใช้ IPST LINK ร่วมกับโปรแกรม Scratch เพื่อแก้ปัญหาโดยการจำลองเป็นบรรณารักษ์อิเล็กทรอนิกส์ในห้องสมุดจำลอง ให้วิเคราะห์ความต้องการ จากนั้นระบุเครื่องมือที่ใช้ (เช่น ตัวตรวจจับ บล็อกคำสั่ง) และแนวคิดการทำงาน ดังต่อไปนี้

ที่	ความต้องการ	เครื่องมือที่ใช้	แนวคิดในการออกแบบ
1	มีการตรวจสอบว่าในห้องสมุดเสียงดังเกินไปไหม เพื่อให้บรรณารักษ์อิเล็กทรอนิกส์พูดเตือนให้เงียบเสียง	
2	ผู้ใช้สามารถปรับย่อขยายขนาดป้ายคำเตือน “ห้ามรับประทานอาหาร” ด้วยตัวตรวจจับบนแผงจร IPST LINK	
3	เมื่อพลบค่ำให้บรรณารักษ์อิเล็กทรอนิกส์แจ้งบอกผู้ใช้บริการว่าจะปิดการให้บริการ	

- ② ให้นำแนวคิดจากข้อที่ 1 มาเขียนเป็นผังความคิดหรือแผนที่ความคิด

ห้องสมุดจำลอง

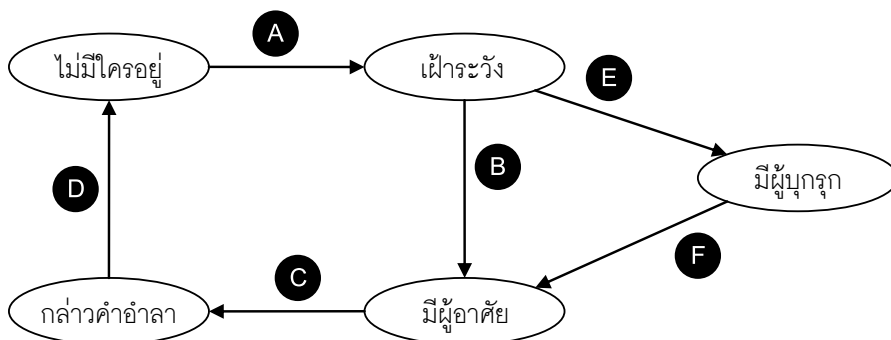
- ③ ลงมือพัฒนาชิ้นงานตามที่ได้เขียนแนวคิดไว้ โดยสร้างตัวละครแทนบรรณารักษ์และเวทีเป็นบรรยากาศห้องสมุด
- ④ ปรับปรุงชิ้นงานให้มีประสิทธิภาพหรือมีความน่าสนใจมากยิ่งขึ้น



ใบงานที่ 2.2 Smart Home

ชื่อ-สกุล เลขที่

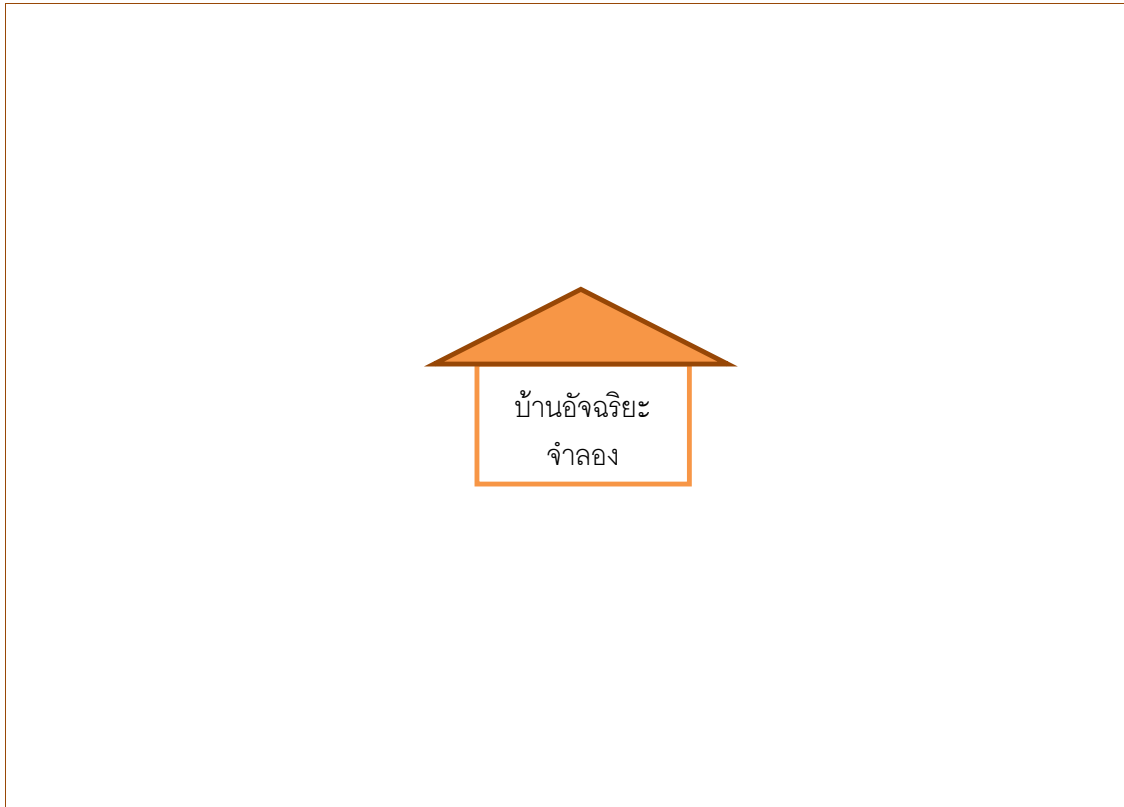
- 1) ต้องการสร้างบ้านจำลองอัจฉริยะที่มีความต้องการการทำงานแบบอัตโนมัติ ดังนี้
 - 1) เมื่อเริ่มต้นระบบ ให้บ้านอยู่ในสถานะ “ไม่มีใครอยู่”
 - 2) ระหว่างที่อยู่ในสถานะ “ไม่มีใครอยู่” หากมีการเปิดประตูบ้านจะทำให้บ้านเข้าสู่สถานะ “เฝ้าระวัง”
 - 3) หากเปิดประตูแล้วไม่เปิดไฟภายใน 10 วินาที ให้บ้านเข้าสู่สถานะ “มีผู้บุกรุก”
 - 4) ตรวจเท่าที่บ้านอยู่ในสถานะ “มีผู้บุกรุก” ให้โปรแกรมกล่าวคำว่า “มีผู้บุกรุก” ซ้ำๆ กันทุก 5 วินาที จนกว่าบ้านจะเปลี่ยนไปเป็นสถานะอื่น
 - 5) หากเปิดประตูและเปิดไฟภายใน 10 วินาที ให้บ้านเปลี่ยนสถานะไปเป็น “มีผู้อาศัย”
 - 6) บ้านที่อยู่ในสถานะ “มีผู้บุกรุก” จะออกจากสถานะนี้ได้ก็ต่อเมื่อเปิดไฟหรือกดปุ่มปลดล็อกที่ซ่อนเอาไว้ หากพบเงื่อนไขนี้บ้านจะเปลี่ยนสถานะไปเป็น “มีผู้อาศัย”
 - 7) เมื่อบ้านเปลี่ยนจากสถานะใดก็ตามมาเป็นสถานะ “มีผู้อาศัย” ให้โปรแกรมกล่าวคำทักทายว่า “ยินดีต้อนรับ” หนึ่งครั้ง
 - 8) ระหว่างที่อยู่ในสถานะ “มีผู้อาศัย” หากมีการปิดไฟและมีการเปิดประตู ให้บ้านเปลี่ยนสถานะเป็น “กล่าวคำอำลา”
 - 9) เมื่อบ้านเปลี่ยนจากสถานะใดก็ตามมาเป็นสถานะ “กล่าวคำอำลา” ให้โปรแกรมกล่าวคำว่า “ขอให้โชคดี” หนึ่งครั้ง
 - 10) ระหว่างที่อยู่ในสถานะ “กล่าวคำอำลา” และประตูปิดลง ให้บ้านเปลี่ยนสถานะเป็น “ไม่มีผู้อาศัย”
- 2) นำความต้องการข้างต้นมาร่างเป็นแผนภาพสถานะของบ้านดังนี้



สรุปเงื่อนไขการตรวจสอบการเปลี่ยนแปลงสถานะจากคำอธิบายข้างต้นลงในตาราง พร้อมทั้งระบุแนวคิดการออกแบบโปรแกรมว่าสามารถตรวจสอบเงื่อนไขหรือสถานะอย่างไร สามารถนำอุปกรณ์ตรวจจับบน IPST LINK และบล็อกคำสั่งมาช่วยในการตรวจสอบได้อย่างไร

จุดเปลี่ยนสถานะ	เงื่อนไขการเปลี่ยนสถานะ	แนวคิดในการตรวจสอบเงื่อนไข	อุปกรณ์และบล็อกที่คาดว่าจะนำมาใช้ในการตรวจสอบ
A
B
C
D
E
F

- 3) ให้นำแนวคิดจากข้อที่ 1 และ 2 มาเขียนเป็นผังความคิดหรือแผนที่ความคิด



- 4) ลงมือพัฒนาชิ้นงานตามที่ได้เขียนแนวคิดไว้ โดยสร้างแบบจำลองบ้านอย่างง่ายขึ้นประกอบการใช้งาน
- 5) เพิ่มเติมหรือปรับปรุงชิ้นงานให้มีประสิทธิภาพ หรือมีความน่าสนใจมากยิ่งขึ้น



กิจกรรมที่ 3 สร้างชิ้นงาน STEM

ระดับชั้น

มัธยมศึกษาตอนปลาย

แนวคิด

ชุดแผนวงจร IPST LINK เป็นชุดอุปกรณ์ที่ใช้ประกอบกับการเขียนโปรแกรม Scratch ซึ่งสามารถนำไปสร้างชิ้นงานที่บูรณาการกับสาระวิชาวิทยาศาสตร์ เทคโนโลยี วิศวกรรม และคณิตศาสตร์ได้ ซึ่งผู้เรียนต้องมีความรู้พื้นฐานในการเขียนโปรแกรมด้วย Scratch มีทักษะในการแก้ปัญหาและมีความคิดสร้างสรรค์ ในการนำชุดแผนวงจร IPST LINK ไปออกแบบและสร้างชิ้นงานเพื่อใช้ประโยชน์ตามความต้องการ

ตัวชี้วัด

- แก้ปัญหาด้วยเทคโนโลยีสารสนเทศ
- สร้างหรือพัฒนาผลิตภัณฑ์หรือวิธีการอย่างมีความคิดสร้างสรรค์ตามกระบวนการเทคโนโลยี โดยถ่ายทอดความคิดเกี่ยวกับผลิตภัณฑ์เป็นภาพร่างอย่างง่ายหรือภาพร่างสามมิติ และภาพถ่าย อาจสร้างแบบจำลองก่อนสร้างชิ้นงานหรือถ่ายทอดความคิดเกี่ยวกับวิธีการเป็นผังงาน
- บอกค่าความจริงของประพจน์ที่มีตัวเชื่อม “และ” “หรือ” “ถ้า...แล้ว” “ก็ต่อเมื่อ”

สาระการเรียนรู้

S: -

- T:**
1. การแก้ปัญหาเป็นกระบวนการที่ต้องใช้ความรู้ ทักษะ ความเข้าใจ ซึ่งบางปัญหาอาจมีแนวทางในการแก้ปัญหาได้หลายวิธี
 2. ขั้นตอนการแก้ปัญหา ประกอบด้วย การวิเคราะห์และกำหนดรายละเอียด การวางแผน และดำเนินการแก้ปัญหา การตรวจสอบและปรับปรุง

E: การสร้างหรือพัฒนาผลิตภัณฑ์ให้เกิดการเคลื่อนที่หรือเคลื่อนไหวตามความต้องการ ต้องอาศัยหน้าที่ของกลไก หรือมีการควบคุมหรือทำงานแบบอัตโนมัติหรือเพิ่มประสิทธิภาพการทำงานของผลิตภัณฑ์โดยนำความรู้ด้านอิเล็กทรอนิกส์มาใช้ประโยชน์ต่างๆ เช่น การตรวจจับ การวัด การสื่อสารหรือความปลอดภัย

M: ค่าความจริงของนิเสธของประพจน์และประพจน์ที่มีตัวเชื่อม “และ” “หรือ” “ถ้า...แล้ว”

จุดประสงค์

สร้างชิ้นงานที่บูรณาการวิทยาศาสตร์ เทคโนโลยี วิศวกรรม และคณิตศาสตร์ โดยใช้แพลตฟอร์ม IPST LINK และโปรแกรม Scratch

แนวการจัดกิจกรรมการเรียนรู้

1. แบ่งกลุ่มผู้เรียน กลุ่มละ 4 คน
2. ผู้สอนยกตัวอย่างชิ้นงานที่ประยุกต์ใช้ชุดแพลตฟอร์ม IPST LINK จากไฟล์เดอร์ MediaScratch\AcStem3\example_act3 หรือเปิดแบบออนไลน์ซึ่งอยู่ในรายการตัวอย่างโครงงาน IPST LINK
3. ผู้เรียนและผู้สอนร่วมกันอภิปรายเกี่ยวกับการสร้างชิ้นงานโดยใช้ความรู้ที่เรียนมาเกี่ยวกับ Scratch และชุดแพลตฟอร์ม IPST LINK ซึ่งเป็นการต่อยอดความรู้เดิมและเป็นการเชื่อมโยงสาระการเรียนรู้ทั้ง 4 สาระ คือ วิทยาศาสตร์ เทคโนโลยี วิศวกรรม และคณิตศาสตร์
4. ผู้สอนชี้แจงการทำชิ้นงานโดยให้ผู้เรียนทำชิ้นงานที่เป็นการต่อยอดจากความรู้เดิมที่เรียนมาหรือคิดขึ้นมาใหม่ และมีเวลาดลงมือทำให้เสร็จภายในเวลาที่กำหนด
5. ผู้เรียนทำใบงานที่ 3.1 เรื่อง ชิ้นงาน STEM โดยผู้สอนคอยให้คำแนะนำอย่างใกล้ชิด
6. ผู้เรียนสร้างชิ้นงานตามทีออกแบบไว้โดยให้สอดคล้องใบงานที่ 3.1
7. ผู้เรียนนำเสนอชิ้นงาน และผู้เรียนกลุ่มอื่นประเมินผลงานของเพื่อนโดยใช้แบบประเมินผลงานและการนำเสนอ
8. ผู้เรียนส่งผลการประเมินและผู้สอนสรุปผลงานของกลุ่มที่ได้คะแนนสูงสุด
9. ผู้เรียนและผู้สอนร่วมกัน อภิปรายถามตอบและสรุปเกี่ยวกับการสร้างชิ้นงาน STEM ด้วยชุดแพลตฟอร์ม IPST LINK

การวัดและประเมินผล

1. ประเมินจากชิ้นงาน
2. การตอบคำถามในใบงาน

วัสดุอุปกรณ์/สื่อ/แหล่งเรียนรู้

1. ชุดแพลตฟอร์ม IPST LINK
2. คอมพิวเตอร์ที่ติดตั้งโปรแกรม Scratch 2.0 offline edition
3. ใบงานที่ 3.1

หมายเหตุ

1. ความรู้ที่จำเป็น : ขั้นตอนการพัฒนาโปรแกรมคอมพิวเตอร์ การเขียนโปรแกรมด้วย Scratch การใช้งานชุดแพลตฟอร์ม IPST LINK และความรู้อื่นๆ ที่ใช้ทำชิ้นงาน
2. ตัวอย่างไฟล์ชิ้นงานอยู่ในไฟล์เดอร์ Act_3\example_act3

ใบงานที่ 3.1 สร้างชิ้นงาน STEM

ชื่อ-สกุล เลขที่

ให้ผู้เรียนออกแบบชิ้นงานที่ใช้ความรู้ทั้ง 4 สาขาคือ วิทยาศาสตร์ เทคโนโลยี วิศวกรรมและคณิตศาสตร์ โดยประยุกต์ใช้ชุดแผงวงจร IPST LINK

- 1 ชื่อชิ้นงาน
- 2 ประโยชน์ที่ได้รับจากชิ้นงาน
- 3 แนวคิดของชิ้นงาน (อธิบายหลักการหรือเป้าหมายของชิ้นงาน)
- 4 ขอบเขตของชิ้นงาน (อธิบายว่า ชิ้นงานทำอะไรได้บ้าง ทำอะไรไม่ได้บ้าง) สิ่งที่ชิ้นงานนี้ทำได้
- 5 สิ่งที่ชิ้นงานนี้ไม่ครอบคลุมถึงหรือทำไม่ได้

6 ให้ระบุสาระสำคัญของวิชาต่างๆ ที่ใช้ในการสร้างชิ้นงานลงในผังความคิดต่อไปนี้

T: เทคโนโลยี
สาระสำคัญ :

M: คณิตศาสตร์
สาระสำคัญ :

ชื่อชิ้นงาน
.....
.....
.....
.....

S: วิทยาศาสตร์
สาระสำคัญ :

E: วิศวกรรม
สาระสำคัญ :

แบบประเมินผลงานและการนำเสนอ

ชื่อ

ชื่อ

ร.ร.	กลุ่ม	เกณฑ์การประเมิน										
		เนื้อหาเสนอ ครบถ้วน	ตรงประเด็น	วิธีการนำเสนอ น่าสนใจ	มีความมั่นใจ ในการนำเสนอ	ใช้เวลาในการ นำเสนอได้เหมาะสม	ผลงานมีความ สอดคล้องกับเนื้อหาที่กำหนด	โปรแกรมทำงาน ถูกต้อง ตามขั้นตอนที่กำหนด	ผลงานมีความ แปลกใหม่ สร้างสรรค์	ผลงานมีความ สมบูรณ์	ผลงานมีความ สวยงาม น่าสนใจ	ผลงานสามารถ นำไปใช้ให้เกิดประโยชน์
1		10	10	10	10	10	10	10	10	10	10	100
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												

ลงชื่อ

วันที่



รายการตัวอย่างโครงการ IPST LINK

ที่	ชื่อชิ้นงาน	คำอธิบาย	ชื่อแฟ้ม	URL <หมายเลขโครงการ>
1	แผลงศร	เกมที่ใช้ IPST LINK ในการควบคุมลูกศร ให้ไปชนลูกบอลเพื่อเก็บคะแนน	Arrow.sb2	100125179
2	ลูกโป่ง	เกมที่ควบคุมลูกโป่งให้ลอยขึ้นด้วย เซนเซอร์เสียง	Balloon.sb2	100126767
3	เก็บของขวัญ	เกมพาน้องหมีเก็บของขวัญสีน้ำเงิน หากเก็บ ของขวัญสีแดงจะทำให้พลังชีวิตลดลง	BearAndGift.sb2	100125841
4	ปาลูกดอก	เกมเรียนรู้การเคลื่อนที่แบบโพรเจกไทล์โดย การควบคุมการปาลูกดอกให้ตกในเป้าหมาย	Dart.sb2	99964077
5	ลิงกินกล้วย	เกมควบคุมลิงให้ขึ้นลงเพื่อรับกล้วยที่ลอย เข้ามา	Flappy-monkey.sb2	100125413
6	ขลุ่ย	จำลอง IPST LINK ให้เป็นขลุ่ยอิเล็กทรอนิกส์ เป่าที่ตัวตรวจจับเสียงและเลื่อนสไลด์เป็น ตัวโน้ต	Flute.sb2	99964004
7	ไดโนเสาร์ของฉัน	เกมให้อาหารไดโนเสาร์โดยการโยนผลไม้ ให้แบบโพรเจกไทล์	myDino.sb2	100125672
8	ป้ายคะแนน	จำลองป้ายคะแนนที่มีทั้งตัวจับเวลาและ ป้ายบอกคะแนนของทั้ง 2 ทีมในการแข่ง กีฬา	Scoreboard.sb2	99964208
9	เซนเซอร์ปอง	เกมปิงปองโดยใช้ IPST LINK ควบคุม เพื่อตีลูก ความเร็วของลูกปิงปอง	Sensorpong.sb2	99964108
10	บ้านอัจฉริยะ	จำลองบ้านอัจฉริยะที่สามารถควบคุมการ ทำงานอุปกรณ์ภายในบ้านด้วยเสียงและ ท่าทาง	Smarthome2.sb2	100125468
11	สัญญาณเสียง	จำลองอุปกรณ์ตรวจจับคลื่นเสียงแล้วนำมา แสดงในรูปแบบกราฟแท่งแสดงระดับเสียง	Sound Graph.sb2	99958092
12	สัญญาณเสียง v.3	จำลองอุปกรณ์ตรวจจับคลื่นเสียงแล้วนำมา แสดงในรูปแบบกราฟแท่งแสดงระดับเสียง (ปรับปรุงการแสดงผล)	Sound Graph3.sb2	99966163
13	ยิงระเบิด	เกมเรียนรู้การเคลื่อนที่แบบโพรเจกไทล์ โดยการควบคุมการยิงลูกระเบิดให้ตกบน เป้าหมาย	Target.sb2	99963081

หมายเหตุ การเข้า URL เพื่อดูตัวอย่างโครงการแบบออนไลน์ สามารถทำได้โดยการใช้ชื่อเว็บ ดังนี้ <https://scratch.mit.edu/projects/<หมายเลขโครงการ>>
ตัวอย่าง ต้องการดูโครงการเรื่อง ยิงระเบิด ใช้ <https://scratch.mit.edu/projects/99963081>

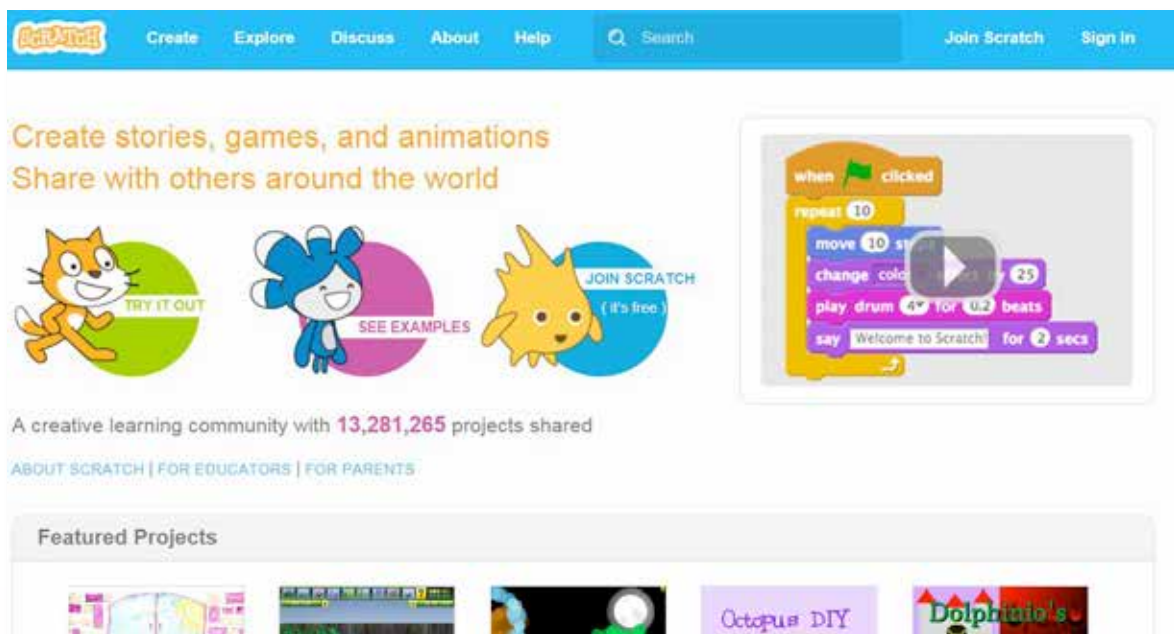


ภาคผนวก ก โปรแกรม Scratch

โปรแกรม Scratch เป็นภาษาคอมพิวเตอร์สำหรับสร้างผลงานต่างๆ เพื่อพัฒนาความคิด ทักษะ การเขียนโปรแกรม โดยมีวิธีการโปรแกรมที่ไม่ต้องมีการพิมพ์คำสั่งที่ยุ่งยากซับซ้อน เหมาะกับการสอน หลักการโปรแกรมที่ส่งเสริมความคิดริเริ่มสร้างสรรค์ การคิดอย่างมีระบบ และการทำงานร่วมกันโดยการ เผยแพร่และแลกเปลี่ยนผลงานผ่านทางเว็บไซต์ <http://scratch.mit.edu/> ในเอกสารฉบับนี้ใช้โปรแกรม Scratch เวอร์ชัน 2.0

ชิ้นงานในโปรแกรม Scratch เรียกว่าโปรเจกต์ ซึ่งประกอบด้วยโครงสร้างที่สำคัญอยู่ 3 ส่วน ได้แก่ เวที (Stage) ตัวละคร (Sprite) และสคริปต์ (Script) โดยแต่ละส่วนมีความสัมพันธ์กันและทำงานร่วมกัน

การสร้างโปรเจกต์เกิดจากการเขียนสคริปต์ (ชุดคำสั่ง) ซึ่งโปรแกรม Scratch ใช้การวางบล็อก เรียงต่อกันแทนการเขียนคำสั่งในโปรแกรมภาษาทั่วไป เพื่อสั่งให้ตัวละครทำงาน ณ ตำแหน่งต่างๆ บนเวที คำสั่งพื้นฐานเบื้องต้น เพื่อสั่งงานให้ตัวละครเคลื่อนที่ เปลี่ยนแปลงท่าทาง หรือการควบคุมการทำงานของ ตัวละคร



เว็บไซต์ Scratch (<http://scratch.mit.edu>) เป็นแหล่งชุมชนออนไลน์ของผู้ใช้งานโปรแกรม Scratch มีคุณสมบัติเด่นดังนี้


- เป็นแหล่งเรียนรู้เกี่ยวกับ Scratch สำหรับนักเรียนและผู้ที่ต้องการฝึกเขียนโปรแกรม Scratch
- เป็นแหล่งข้อมูลการเรียนการสอนสำหรับครู มีเทคนิคและวิธีการเรียนการสอน จากครูที่สอน โปรแกรม Scratch ทั่วโลก รวมทั้งที่ Media Lab MIT ด้วย
- สามารถสร้างงานจาก Scratch แบบออนไลน์หรือดาวน์โหลดโปรแกรม Scratch รุ่น offline มาใช้ที่เครื่องคอมพิวเตอร์ได้

- สามารถสร้างงานที่มีการเชื่อมต่อกับอุปกรณ์ภายนอก เช่น บอร์ดที่มีตัวตรวจจับต่างๆ อาทิ ปุ่มกด ตัวต้านทานปรับค่าได้แบบแกนเลื่อน ตัวตรวจจับแสง ตัวตรวจจับเสียง
- สามารถดาวน์โหลดโปรเจกต์ที่ผู้อื่นแบ่งปันไว้มากมาย
- เมื่อลงทะเบียนแล้ว ผู้เรียนสามารถสร้างโปรเจกต์ที่เครื่องของตน แล้วแชร์มายังเว็บไซต์นี้ได้โดยตรง
- เป็นชุมชนแลกเปลี่ยนข่าวสาร ความรู้ ประสบการณ์ ความคิดเห็น เกี่ยวกับ Scratch

สามารถค้นหาความรู้เกี่ยวกับโปรแกรม Scratch นอกจากเว็บไซต์ <http://scratch.mit.edu> ได้ เช่น เว็บไซต์ครูสมชาย <http://kidsangsan.com>, เว็บไซต์สาขาคอมพิวเตอร์ สสวท. <http://oho.ipst.ac.th> หรือค้นหาโดยใช้ <http://www.google.com>

การสร้างงานด้วยโปรแกรม Scratch

Scratch ใช้งานง่าย สามารถเขียนโปรแกรมหรือสร้างงานได้หลากหลายวิธี ในที่นี้ขอแนะนำ 3 วิธี ดังนี้

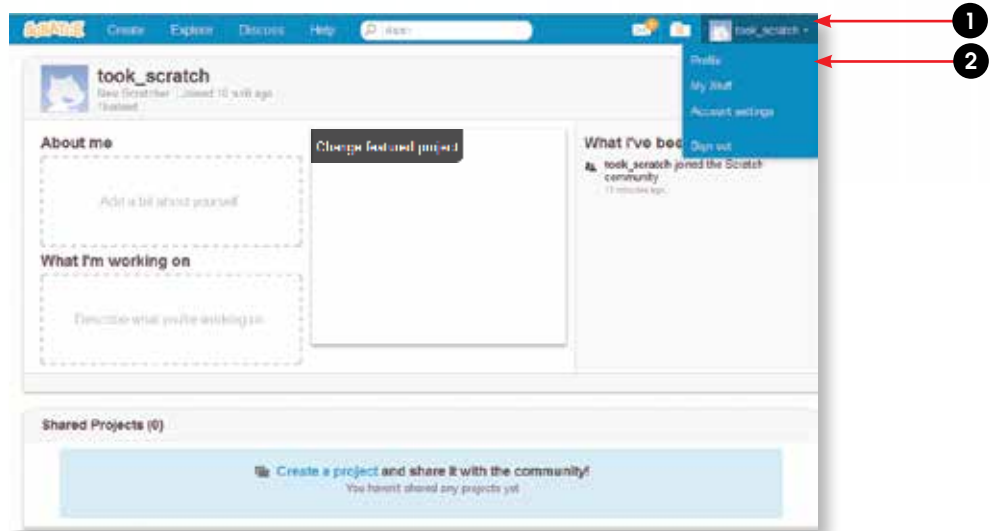
1. สร้างงานออนไลน์แบบสมัครสมาชิก โดยต้องเชื่อมต่ออินเทอร์เน็ต แล้วเข้าเว็บไซต์ <http://scratch.mit.edu> หลังจากนั้นคลิกที่ไอคอน  เพื่อสมัครสมาชิกระบบจะให้เติมข้อมูลชื่อผู้ใช้ รหัสผ่าน เดือนและปีเกิด เพศ ประเทศ อีเมล (ถ้าเป็นเยาวชนอายุไม่เกิน 13 ปี ระบบจะให้กรอกอีเมลของผู้ปกครอง) เมื่อสมัครสมาชิกเรียบร้อยแล้ว ให้เข้าสู่ระบบโดยใช้ชื่อและรหัสผ่านที่ได้กำหนดไว้ในการสมัครสมาชิก หลังจากนั้นสร้างงานโดยคลิกที่ปุ่ม Create
2. สร้างงานออนไลน์แบบไม่ต้องสมัครสมาชิก แต่ต้องเชื่อมต่ออินเทอร์เน็ต ให้เข้าเว็บไซต์ <http://scratch.mit.edu> แล้วคลิกปุ่ม Create เพื่อสร้างงาน
3. สร้างงานออฟไลน์ โดยดาวน์โหลดโปรแกรมมาติดตั้งที่เครื่อง ได้แก่ Adobe AIR และ Scratch offline Editor

การจัดการข้อมูลสมาชิกในเว็บไซต์ Scratch

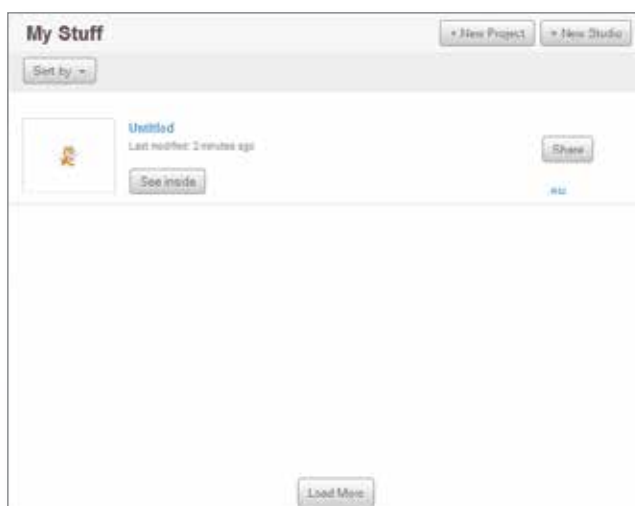
เมื่อสมัครสมาชิกในเว็บไซต์แล้ว ผู้ใช้สามารถเข้าไปจัดการข้อมูลสมาชิกได้ในเมนูต่างๆ ดังนี้

1. Profile หน้าแสดงรายละเอียดเกี่ยวกับประวัติของตนเอง การเข้าสู่หน้า Profile ทำได้ดังนี้
 1. คลิกที่ ▼ จะปรากฏเมนูจัดการสมาชิกแบบต่างๆ
 2. คลิกที่ Profile จะปรากฏหน้าแสดงรายละเอียดของผู้ใช้
 - About me รายละเอียดประวัติผู้ใช้
 - What I'm working on รายละเอียดประวัติการทำงาน
 - Change featured project การเปลี่ยนแปลงรายละเอียดของโปรเจกต์ที่สร้างในเว็บไซต์

- Share Projects จำนวนของโปรเจกต์ที่ได้แบ่งปันในเว็บไซต์ โดยสามารถสร้างโปรเจกต์ใหม่โดยการคลิกที่ Create a project

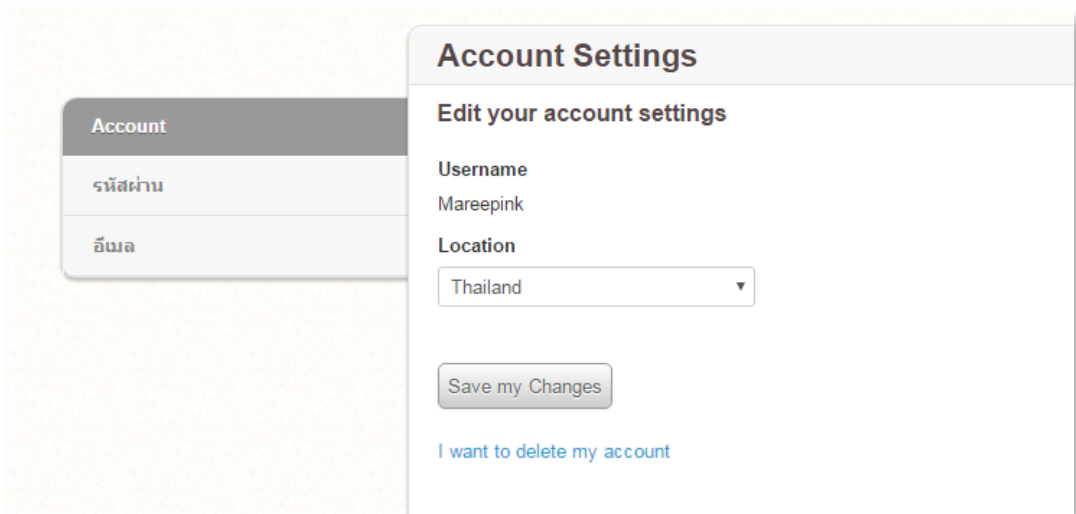


- My Stuff หน้าแสดงคลังโปรเจกต์ (งาน) ที่สร้างขึ้นในเว็บไซต์ของผู้ใช้ มีปุ่มต่างๆ ที่ใช้ในการจัดการดังนี้



1. New Project การสร้างโปรเจกต์ใหม่
2. New Studio การสร้างคลังโปรเจกต์ใหม่
3. Sort by การคัดกรองโปรเจกต์
4. See inside รายละเอียดในโปรเจกต์
5. Share การแบ่งปันโปรเจกต์
6. Load More การนำเข้าไปรเจกต์ในคลัง

- Accounting Setting หน้าแสดงการจัดการข้อมูลบัญชีผู้ใช้
 - การตั้งค่ารหัสผ่านใหม่
 - การเปลี่ยนแปลงอีเมล



เมื่อตั้งค่าต่างๆ ตามต้องการแล้วคลิกปุ่ม Save my changes เพื่อบันทึกการเปลี่ยนแปลง

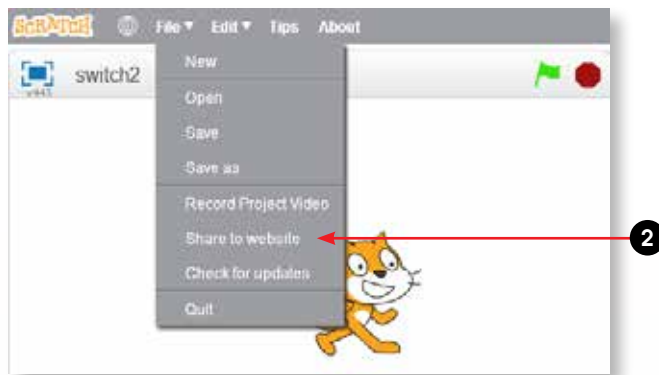
4. Sign out การออกจากระบบ

ขั้นตอนการอัปโหลดผลงาน

การอัปโหลดผลงานที่สร้างจากโปรแกรม Scratch เวอร์ชัน 2 เพื่อแบ่งปัน (share) และเผยแพร่บนเว็บไซต์ Scratch ทำได้ 2 วิธีดังนี้

วิธีที่ 1 อัปโหลดจากโปรแกรม Scratch รุ่นออฟไลน์ โดยทำตามขั้นตอนดังนี้

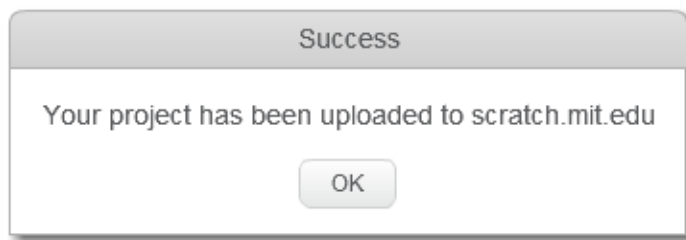
1. เปิดโปรเจกต์ที่ต้องการแล้วคลิกเมนู File เลือก Share to website



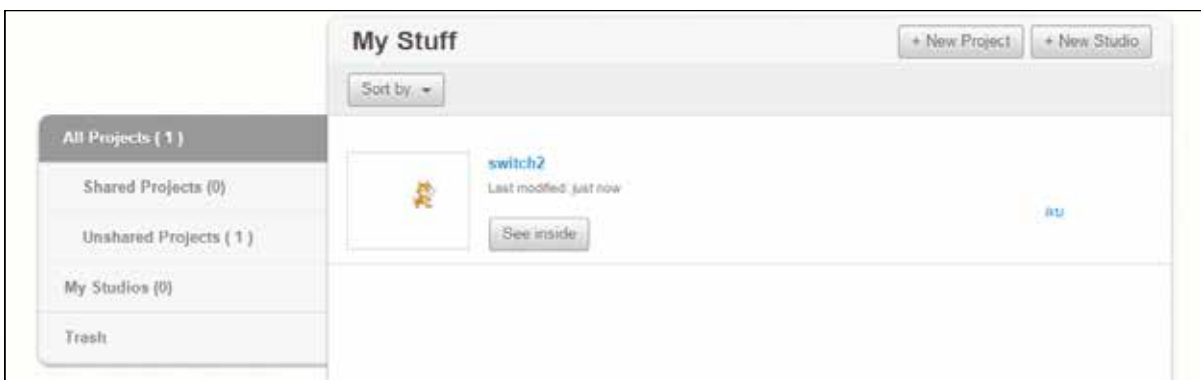
2. จะปรากฏหน้าต่าง Shared to website แล้วกรอกรายละเอียดดังนี้

Project name: ชื่อของโปรเจกต์
 Your Scratch name: ชื่อผู้ใช้
 Password: รหัสผ่าน } สำหรับ Login เข้าสู่เว็บไซต์ Scratch

- 3 เมื่อปรากฏหน้าต่าง Success แสดงข้อความ “Your project has been uploaded to scratch.mit.edu” ให้คลิกปุ่ม **OK**

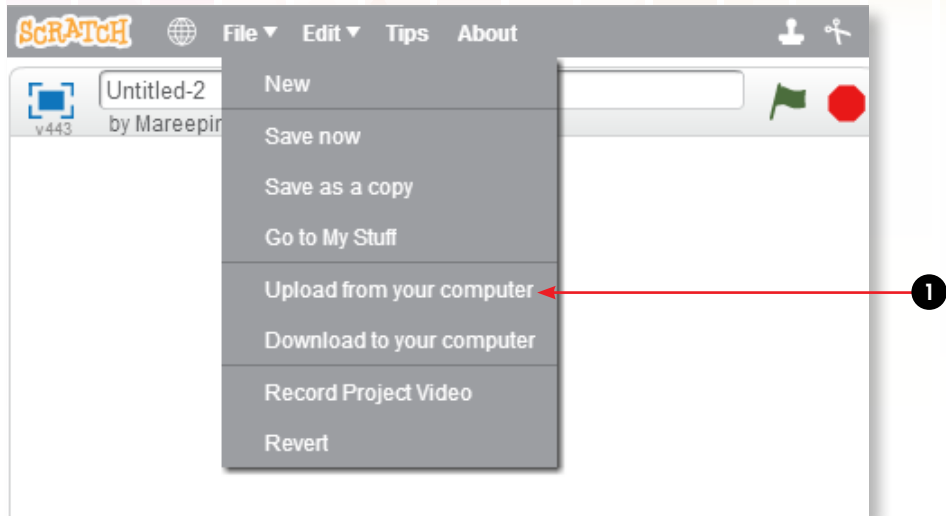


ผลงานที่อัปโหลดจะปรากฏอยู่ใน My Stuff ของผู้ใช้ในเว็บไซต์ Scratch ดังรูป

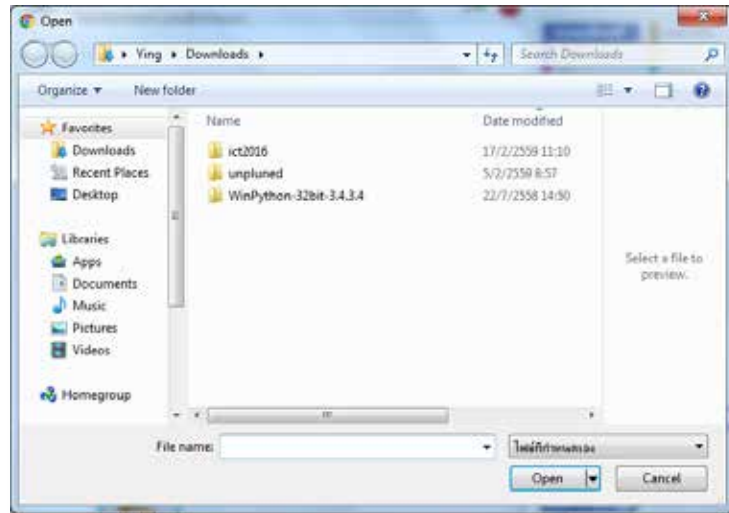


วิธีที่ 2 อัปโหลดผ่านทางเว็บไซต์ Scratch ทำได้ดังนี้

- 1 คลิกปุ่ม Create บนเว็บไซต์ Scratch.mit.edu แล้วคลิกเมนู File เลือก New หลังจากนั้นเลือก Upload from your computer (หากไม่เลือกคำสั่ง New ก่อนการอัปโหลด ไฟล์ที่อัปโหลดจะทับไฟล์ที่เปิดอยู่)

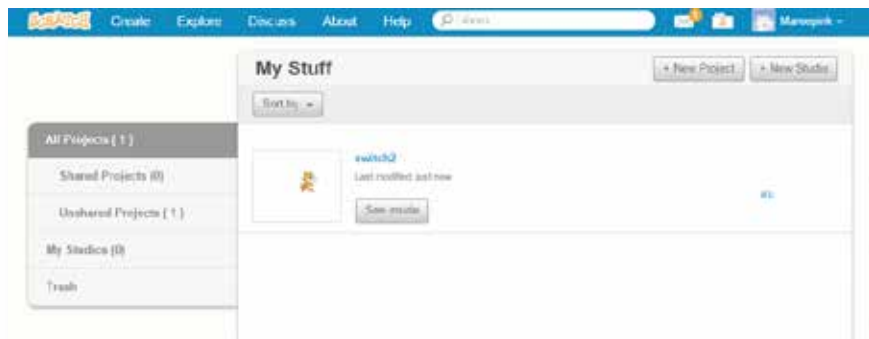


2 จะปรากฏหน้าต่าง Open แล้วเลือกไฟล์ที่ต้องการ แล้วคลิกปุ่ม Open



ภาคผนวก

ผลงานที่อัปโหลดจะปรากฏอยู่ใน My stuff ของผู้ใช้ในเว็บไซต์ Scratch ดังรูป



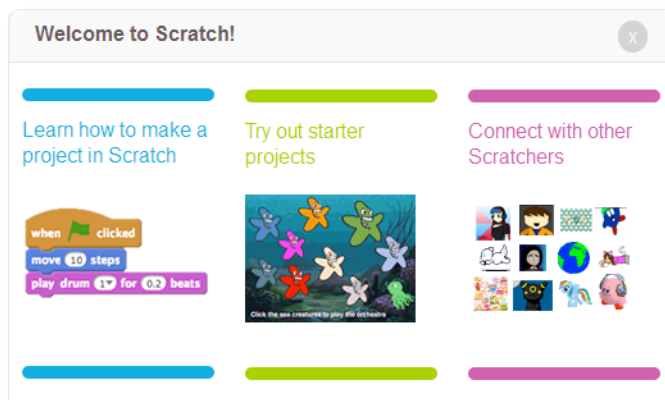
หมายเหตุ หากต้องการแบ่งปันผลงานให้คลิกปุ่ม **Share** หลังจากนั้นผลงานจะไปปรากฏใน My Stuff ส่วนของ Shared Projects



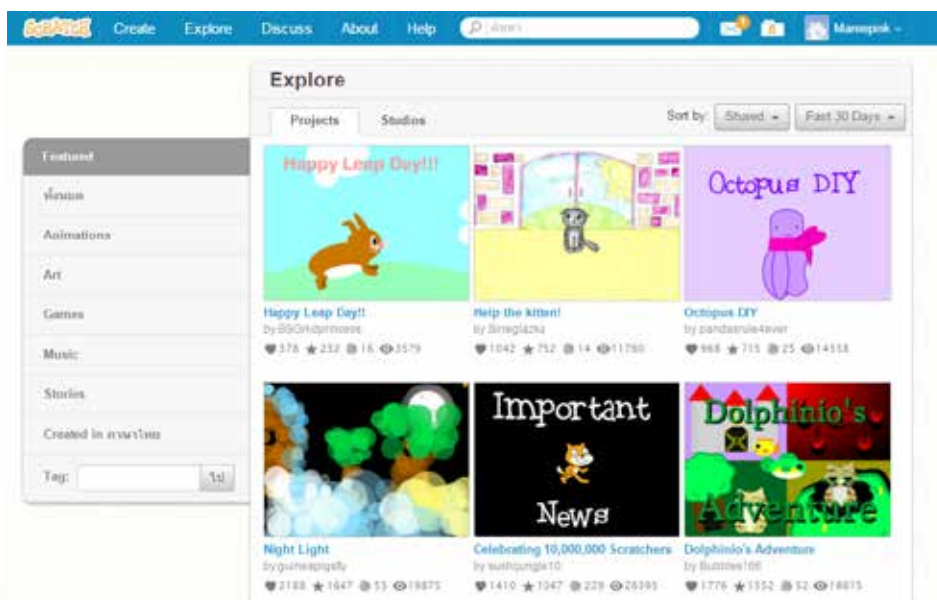
การศึกษาโปรเจกต์จากเว็บไซต์ Scratch

เว็บไซต์ Scratch เป็นแหล่งรวมความรู้ที่มีตัวอย่างโปรเจกต์ และวิธีการใช้งานโปรแกรม ให้ผู้ใช้ได้ศึกษาเพื่อเป็นแนวทางในการสร้างโปรเจกต์ โดยส่วนของเว็บไซต์ที่ให้ศึกษาค้นคว้าหาความรู้ เช่น

1. Learn how to make a project in Scratch วิธีการสร้างโปรเจกต์ด้วย Scratch ในส่วนนี้เว็บไซต์จะแสดงวิดีโอสาธิตการสร้างโปรเจกต์ ซึ่งผู้ใช้สามารถศึกษาด้วยตนเอง



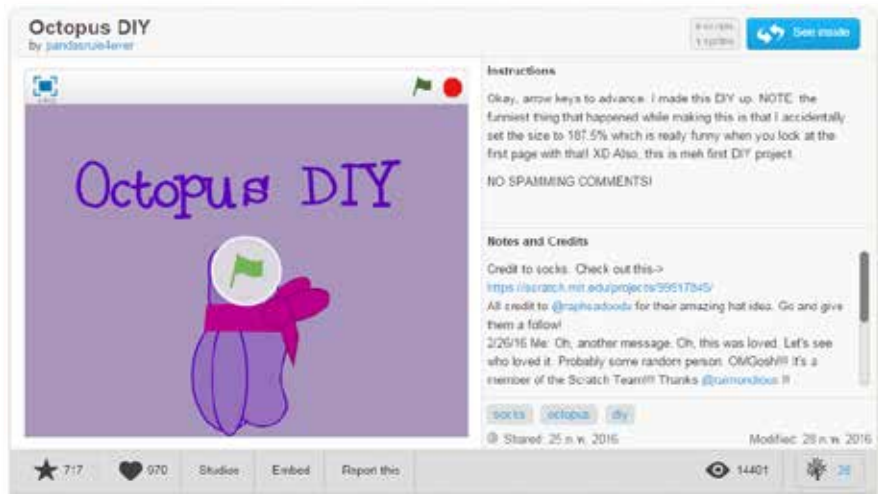
2. Explore สํารวจโปรเจกต์ ผู้ใช้สามารถเข้าไปดูโปรเจกต์ที่สมาชิกแบ่งปันในเว็บไซต์ โดยสามารถเลือกดูตามประเภทต่างๆ ที่เว็บไซต์ได้แบ่งไว้เป็นหมวดหมู่ และใช้ตัวกรองเพื่อดูโปรเจกต์ยอดนิยมหรือโปรเจกต์ที่มีผู้ชื่นชอบมากที่สุด



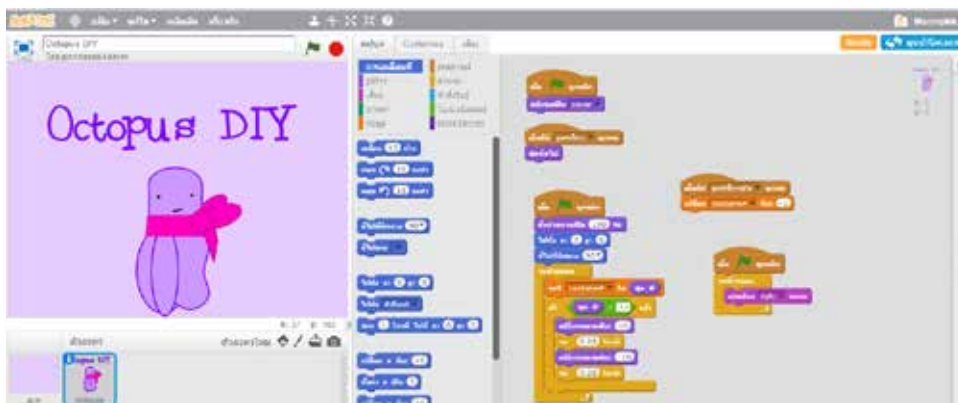
การพัฒนาโปรเจกต์จากผลงานที่ผู้อื่นพัฒนาและแบ่งปันไว้

เราสามารถพัฒนาโปรเจกต์จากผลงานที่ผู้อื่นพัฒนาและแบ่งปันไว้ ผ่านการ Remix โดยเมื่อเลือกโปรเจกต์ที่ต้องการระบบจะแสดงสคริปต์เดิม หลังจากนั้นผู้ใช้สามารถเปลี่ยนชุดตัวละคร ปรับปรุงสคริปต์ สร้างตัวละครใหม่ ขึ้นตอนการ remix ทำได้ดังนี้

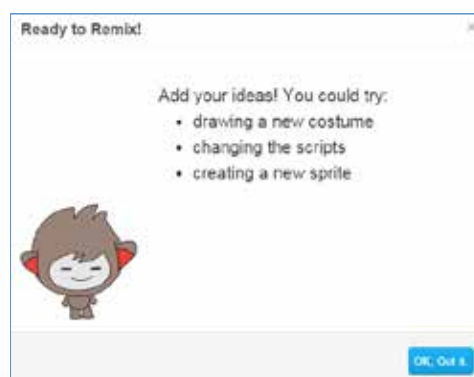
1. เข้าเว็บไซต์ Scratch แล้วคลิกโปรเจกต์ที่ต้องการ



2. คลิกปุ่ม See inside เพื่อเข้าส่วนของโปรแกรม scratch
3. กดปุ่ม Remix

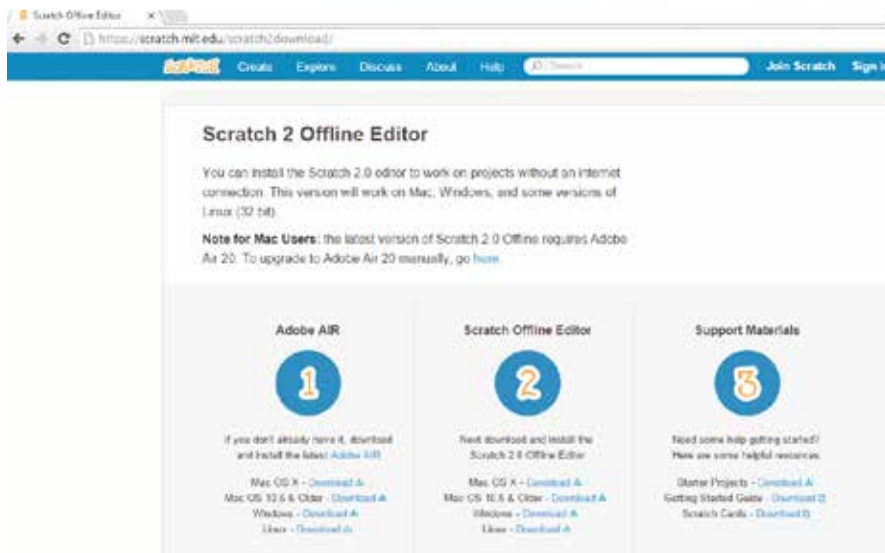


4. จะปรากฏหน้าต่าง Ready to remix! ให้คลิกปุ่ม OK, Got it.



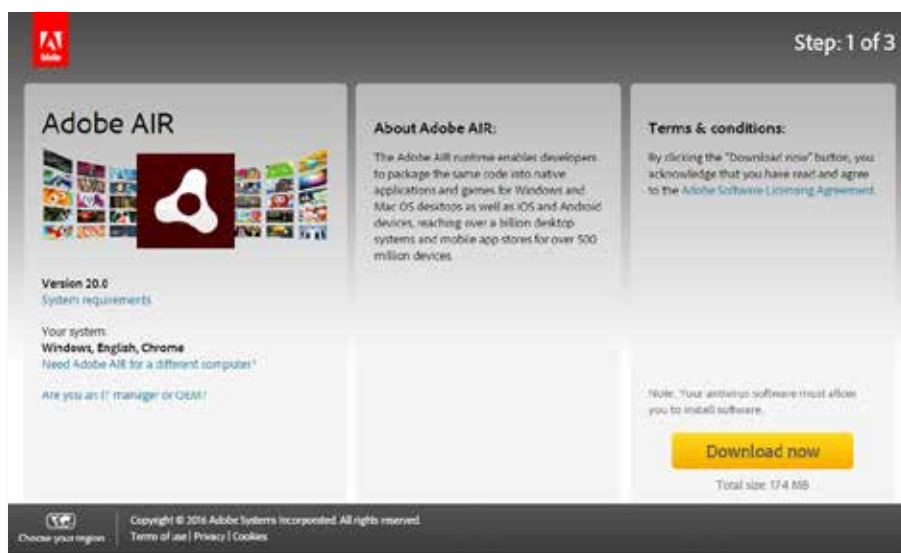
ภาคผนวก ข การติดตั้งโปรแกรม Scratch

โปรแกรม Scratch สามารถใช้งานได้ทั้งแบบออนไลน์ และแบบออฟไลน์ ในภาคผนวกนี้จะแนะนำการติดตั้งโปรแกรม Scratch 2.0 สำหรับใช้งานแบบออฟไลน์ โดยสามารถดาวน์โหลดโปรแกรมได้ที่เว็บไซต์ <https://scratch.mit.edu/scratch2download/> ไฟล์ที่ใช้ในการติดตั้งมี 2 ไฟล์ คือ Adobe AIR และ Scratch Offline Editor ขั้นตอนในการติดตั้งโปรแกรม มีดังนี้



ขั้นตอนที่ 1 : ติดตั้ง Adobe AIR

- 1) ดาวน์โหลดโปรแกรม Adobe AIR โดยเลือกระบบปฏิบัติการให้ตรงกับเครื่องที่ใช้



2) คลิกปุ่ม Download now เมื่อดาว์โหลดเสร็จแล้ว จะได้ไฟล์ชื่อ AdobeAIRInstaller.exe

AdobeAIRInstaller	29/2/2559 9:03	Application	17,917 KB
-------------------	----------------	-------------	-----------

3) ดับเบิลคลิกที่ไฟล์ AdobeAIRInstaller.exe เพื่อติดตั้งโปรแกรม Adobe AIR จะปรากฏหน้าต่าง Adobe AIR Setup ให้คลิกปุ่ม I Agree

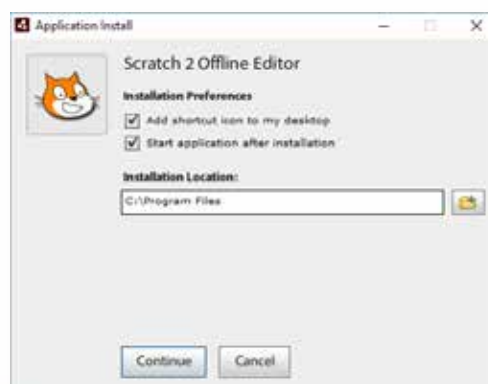


ขั้นตอนที่ 2 : ติดตั้ง Scratch Offline Editor

1) ดาว์โหลดโปรแกรม Scratch Offline Editor โดยเลือกระบบปฏิบัติการให้ตรงกับเครื่องที่ใช้ เมื่อดาว์โหลดเสร็จแล้ว จะได้ไฟล์ชื่อ Scratch-443.exe

Scratch-443	29/2/2559 9:05	Application	65,008 KB
-------------	----------------	-------------	-----------

2) ดับเบิลคลิกไฟล์ Scratch-443.exe เพื่อติดตั้งโปรแกรม Scratch 2 Offline Editor จะปรากฏหน้าต่าง Application Install ให้คลิกปุ่ม Continue เพื่อติดตั้งโปรแกรม





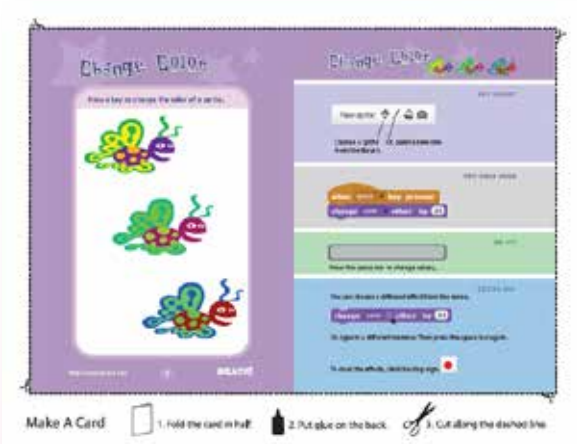
ขั้นตอนที่ 3 : สื่อสำหรับโปรแกรม Scratch

ผู้ใช้สามารถดาวน์โหลดสื่อที่อยู่ใน Support Materials มาใช้งานได้ ซึ่งประกอบด้วย 3 ส่วน ดังนี้

- 1) Starter Projects เป็นตัวอย่างโปรแกรม Scratch ที่ใช้ Video Sensing จากกล้องเว็บแคม เมื่อดาวน์โหลดจะได้ไฟล์ชื่อ Scratch2StarterProjects.zip



- 2) Getting Started Guide เป็นคู่มือการอธิบายการใช้งานโปรแกรม Scratch เมื่อดาวน์โหลดจะได้ไฟล์ชื่อ Getting-Started-Guide-Scratch2.pdf



- 3) Scratch Cards เป็นสื่อการสอนโปรแกรม Scratch ในรูปของบัตรคำ



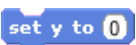











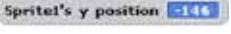


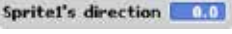
ภาคผนวก ค กลุ่มบล็อกคำสั่งในโปรแกรม Scratch

บล็อกคำสั่งทั้งหมดในโปรแกรม Scratch ถูกจัดออกเป็น 10 กลุ่มตามลักษณะหน้าที่ของแต่ละบล็อกคำสั่ง โดยบล็อกคำสั่งที่ใช้งานได้สำหรับตัวละครและเวทีจะแตกต่างกัน ในที่นี่จะได้อธิบายความหมายของบล็อกคำสั่งในแต่ละกลุ่มบล็อกพอสังเขป

กลุ่มบล็อก Motion







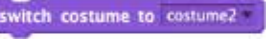






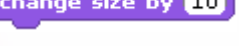
กลุ่มบล็อก Motion ประกอบด้วยบล็อกที่ใช้กำหนดรูปแบบการเคลื่อนที่ให้ตัวละคร เช่น เคลื่อนที่ไปข้างหน้า, หมุนทิศทาง, ไปที่ตำแหน่งต่าง ๆ บนเวที โดยสามารถกำหนดค่าของการเคลื่อนที่ได้ บล็อกในกลุ่มบล็อก Motion มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓		ไปข้างหน้า หรือถอยหลัง ตัวอย่าง ไปข้างหน้า 10 หน่วย
	✓		หมุนตามเข็มนาฬิกา ตัวอย่าง หมุนตามเข็มนาฬิกา 15 องศา
	✓		หมุนทวนเข็มนาฬิกา ตัวอย่าง หมุนทวนเข็มนาฬิกา 15 องศา
	✓		ระบุทิศทางที่ต้องการ ตัวอย่าง ระบุทิศทาง 90 องศา
	✓		ระบุให้ตัวละครหันไปหาเป้าหมายที่กำหนด เช่น
	✓		ไปที่ตำแหน่ง (x, y) บนเวที ตัวอย่าง ไปตรงกลางเวที (x=0, y=0)
	✓		ไปที่ตำแหน่งของเป้าหมายที่ระบุ ตัวอย่าง ไปที่ตำแหน่งตัวชี้เมาส์ ไปที่ตำแหน่งเดียวกับตัวละคร Sprite2
	✓		เคลื่อนไปที่ตำแหน่ง (x, y) บนเวทีในเวลาที่กำหนด ตัวเลขยิ่งมากยิ่งช้า ตัวอย่าง ไปตำแหน่งกลางเวที (0, 0) โดยใช้เวลาเคลื่อนไป 1 วินาที
	✓		เปลี่ยนตำแหน่งของตัวละครไปด้านซ้ายหรือขวา ตัวอย่าง เปลี่ยนตำแหน่งตัวละครไปด้านขวา 10 หน่วย

บล็อก	ตัวละคร	เวที	ความหมาย
	✓		ตั้งค่าตำแหน่งตัวละครไปด้านซ้ายหรือขวา ตัวอย่าง ตั้งค่าตัวละครไปที่ตำแหน่ง $x = 0$ คือ กลางเวที $x = 240$ คือ ขอบขวาสุด $x = -240$ คือ ขอบซ้ายสุด
	✓		เปลี่ยนตำแหน่งของตัวละครไปด้านบนหรือล่าง ตัวอย่าง เปลี่ยนตำแหน่งตัวละครไปด้านบน 10 หน่วย
	✓		ตั้งค่าตำแหน่งตัวละครไปด้านบนหรือล่าง ตัวอย่าง ตั้งค่าตัวละครไปที่ตำแหน่ง $y = 0$ คือ กลางเวที $y = 180$ คือ ขอบบนสุด $y = -180$ คือ ขอบล่างสุด
	✓		ถ้าตัวละครชนขอบให้สะท้อนกลับ 
	✓		กำหนดลักษณะการหมุนของตัวละคร ถ้าเป็น  ตัวละครจะหันไปทางซ้ายหรือขวาเท่านั้น โดยจะหันไปทางซ้ายเมื่อทิศเป็นค่าลบ (อยู่ในช่วง -1 ถึง -179 องศา) และหันไปทางขวาเมื่อทิศเป็นค่าบวก (อยู่ในช่วง 0 ถึง 180 องศา) ถ้าเป็น  ตัวละครจะไม่เปลี่ยนการหันหน้าไปตามทิศเลย ถ้าเป็น  ตัวละครจะหันไปตามทิศตามที่กำหนด
	✓		แสดงตำแหน่งตัวละครตามแนวแกน x ถ้าต้องการให้แสดงค่าตำแหน่งของตัวละครตามแนวแกน x ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก  ซึ่งแสดงตำแหน่งบนเวที ตัวอย่าง 
	✓		แสดงตำแหน่งตัวละครตามแนวแกน y ถ้าต้องการให้แสดงค่าตำแหน่งของตัวละครตามแนวแกน y ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก  ซึ่งแสดงตำแหน่งบนเวที ตัวอย่าง 
	✓		แสดงทิศทางตัวละคร ถ้าต้องการให้แสดงค่าทิศทางตัวละคร ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก  ซึ่งแสดงทิศทางบนเวที ตัวอย่าง 

กลุ่มบล็อก Looks

กลุ่มบล็อก Looks ใช้สำหรับการเขียนสคริปต์เพื่อสั่งให้ตัวละครหรือเวทีแสดงคุณสมบัติต่างๆ เช่น ตัวละครพูด เปลี่ยนชุดตัวละคร เปลี่ยนสีตัวละคร เปลี่ยนขนาดตัวละคร เปลี่ยนพื้นหลัง บล็อกในกลุ่มบล็อก Looks มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓		แสดงคำพูดตามเวลาที่กำหนด ตัวอย่าง แสดงคำพูด Hello! เป็นเวลา 2 วินาที
	✓		แสดงคำพูดค้างไว้จนกว่าจะกำหนดให้แสดงคำพูดอื่น ตัวอย่าง แสดงคำพูด Hello! ค้างไว้
	✓		แสดงความคิดตามเวลาที่กำหนด ตัวอย่าง แสดงความคิด Hmm... เป็นเวลา 2 วินาที
	✓		แสดงความคิดค้างไว้จนกว่าจะกำหนดให้แสดงความคิดอื่น ตัวอย่าง แสดงความคิด Hmm... ค้างไว้
	✓		แสดงตัวละคร
	✓		ซ่อนตัวละคร
	✓		สลับชุดตัวละคร ตัวอย่าง สลับชุดตัวละครเป็นชุด costume2
	✓		เปลี่ยนชุดตัวละครเป็นชุดถัดไปที่มีอยู่ในรายการ
		✓	สลับภาพพื้นหลังของเวที ตัวอย่าง สลับภาพพื้นหลังของเวทีเป็นภาพ backdrop1
	✓	✓	เปลี่ยนเทคนิคการแสดงผลภาพตัวละคร โดยเลือกเทคนิคพิเศษจาก เครื่องหมาย ▼  ตัวอย่าง เทคนิค color และตั้งค่าปริมาณการใช้เป็น 25
	✓	✓	ตั้งค่าเทคนิคการแสดงผลภาพตัวละครตามที่กำหนด ตัวอย่าง ตั้งค่าเทคนิค color ให้เป็น 0 (สีเดิม)
	✓	✓	ล้างค่าเทคนิคการแสดงผลภาพตัวละครที่กำหนด
	✓		เปลี่ยนขนาดตัวละครตามที่กำหนด ถ้าค่า + จะเป็นการเพิ่มขนาด ค่า - เป็นการลดขนาด ตัวอย่าง เพิ่มขนาดตัวละคร 10%

บล็อก	ตัวละคร	เวที	ความหมาย
	✓		กำหนดขนาดตัวละครเป็น % ตัวอย่าง กำหนดขนาดตัวละครเป็น 100% (ขนาดจริง)
	✓		นำตัวละครไปไว้ด้านหน้าสุด
	✓		ย้ายตัวละครไปไว้ด้านหลังตามจำนวนเลเยอร์ (ชั้น) ที่กำหนด ตัวอย่าง ย้ายตัวละครไปไว้ด้านหลัง 1 เลเยอร์
	✓		แสดงเลขชุดตัวละคร ถ้าต้องการให้แสดงเลขชุดตัวละคร ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก <input checked="" type="checkbox"/> ซึ่งแสดงเลขชุดตัวละครบนเวที
		✓	แสดงชื่อภาพพื้นหลังเวที ถ้าต้องการให้แสดงชื่อภาพพื้นหลังเวที ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก <input checked="" type="checkbox"/> ซึ่งจะแสดงชื่อภาพพื้นหลังบนเวที
	✓		แสดงขนาดตัวละคร ถ้าต้องการให้แสดงค่าขนาดตัวละครบนเวที ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก <input checked="" type="checkbox"/>
	✓		เปลี่ยนภาพพื้นหลังเวทีเป็นภาพถัดไปที่มีอยู่ในรายการ














กลุ่มบล็อก Sound

กลุ่มบล็อก Sound ทำงานเกี่ยวกับการแสดงเสียงต่างๆ ซึ่งอาจเป็นเสียงจากไฟล์ที่มีอยู่ หรือไฟล์เสียงที่บันทึกโดยเครื่องมือบันทึกเสียง (Sound Recorder) ซึ่งอยู่ในโปรแกรม Scratch ไฟล์เสียงจะมีส่วนขยายเป็น .wav หรือ .mp3 นอกจากนี้ ยังมีบล็อกที่สามารถกำหนดเสียงของเครื่องดนตรี ชนิดต่างๆ ได้ บล็อกในกลุ่มบล็อก Sound มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	เริ่มเล่นเสียงที่ระบุ แล้วทำคำสั่งถัดไปทันที
	✓	✓	เริ่มเล่นเสียงที่ระบุ จนกระทั่งจบแล้วจึงทำคำสั่งถัดไป
	✓	✓	หยุดเสียงทั้งหมดที่กำลังเล่นอยู่
	✓	✓	เล่นเสียงเครื่องตี โดยมีประเภทของเครื่องตีชนิดต่างๆ ให้เลือกได้จำนวน 18 ชนิด รวมทั้งกำหนดจังหวะในการเล่นเสียงแต่ละครั้งได้
	✓	✓	หยุดพักเสียง ตามจังหวะที่กำหนด
	✓	✓	เล่นเสียงโน้ตดนตรี (โด เร มี ...) มีค่าให้เลือกได้ 25 เสียง (48-72) และกำหนดจังหวะในการเล่นเสียงได้
	✓	✓	กำหนดชนิดของเครื่องดนตรีชนิดต่างๆ โดยสามารถเลือกชนิดของเครื่องดนตรีได้ 21 ชนิด
	✓	✓	เพิ่มหรือลดระดับความดังของเสียง
	✓	✓	กำหนดความดังของเสียงเป็นเปอร์เซ็นต์
	✓	✓	แสดงความดังของเสียง ถ้าต้องการให้แสดงค่าความดังของเสียง ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก <input checked="" type="checkbox"/> volume
	✓	✓	เพิ่มหรือลดจังหวะของเสียง
	✓	✓	กำหนดจังหวะความเร็วของเสียงต่อนาที
	✓	✓	แสดงจังหวะความเร็วของเสียง ถ้าต้องการให้แสดงค่าจังหวะความเร็วของเสียง ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก <input checked="" type="checkbox"/> tempo

กลุ่มบล็อก Pen

กลุ่มบล็อก Pen มีบล็อกคำสั่งที่ทำงานเหมือนกับตัวละครมีปากกาที่ช่วยในการวาดรูปติดอยู่ด้วย โดยสามารถกำหนดลักษณะปากกาได้ เช่น กำหนดสี ขนาดเส้น การยกปากกา การวางปากกา บล็อกในกลุ่มบล็อก Pen มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	ล้างภาพวาดที่เกิดจากคำสั่งในกลุ่มบล็อก Pen
	✓		ประทับตราตัวละคร ทำให้เกิดรูปของตัวละครถูกประทับติดอยู่บนเวทีอย่างถาวร
	✓		วางปากกาลงบนพื้นเวที มีผลให้เกิดเส้นที่ถูกวาดตามการเคลื่อนที่ของตัวละคร
	✓		ยกปากกาขึ้น มีผลให้ไม่มีการวาดเส้นตามการเคลื่อนที่ของตัวละคร
	✓		กำหนดสีปากกา สามารถคลิกเลือกสีได้
	✓		เพิ่มหรือลดค่าสีปากกา
	✓		กำหนดสีปากกา โดยระบุค่าสีเป็นตัวเลข
	✓		เพิ่มหรือลดเฉดสีปากกา
	✓		กำหนดเฉดสีของปากกา โดยระบุค่าเฉดสีเป็นตัวเลข
	✓		เพิ่มหรือลดขนาดของปากกา
	✓		กำหนดขนาดปากกา โดยระบุขนาดเป็นตัวเลข

กลุ่มบล็อก Data

กลุ่มบล็อก Data มีบล็อกคำสั่งที่ใช้ในการดำเนินการต่างๆ กับตัวแปร ทั้งตัวแปรเดี่ยว (variable) และรายการ (list) เมื่อเริ่มต้นจะมีเพียงแค่ปุ่มสำหรับให้สร้างตัวแปร หรือรายการ เท่านั้น หลังจากสร้างตัวแปร หรือรายการแล้ว บล็อกคำสั่งที่เกี่ยวข้องจึงจะปรากฏให้ใช้งานได้ บล็อกในกลุ่มบล็อก Data มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	สร้างตัวแปรใหม่
	✓	✓	แสดงชื่อและค่าของตัวแปรบนเวที ตัวอย่าง ตัวแปรชื่อ
	✓	✓	กำหนดค่าให้กับตัวแปร
	✓	✓	เพิ่มค่าของตัวแปรที่กำหนดขึ้นจากค่าเดิมอีก 1
	✓	✓	สั่งให้แสดงชื่อและค่าของตัวแปรบนเวที
	✓	✓	สั่งให้ซ่อนชื่อและค่าของตัวแปรบนเวที
	✓	✓	เปลี่ยนชื่อตัวแปร และลบตัวแปร สามารถทำได้โดยคลิกเมาส์ขวาที่บล็อกชื่อตัวแปร แล้วเลือก rename variable หรือ delete variable ตามลำดับ
	✓	✓	สร้างรายการใหม่
	✓	✓	แสดงชื่อและค่าของรายการบนเวที ตัวอย่าง รายการชื่อ
	✓	✓	เพิ่มข้อมูล (item) เข้าไปต่อท้ายข้อมูลตัวสุดท้ายในรายการ
	✓	✓	ลบข้อมูลลำดับที่ระบุออกจากรายการ โดยข้อมูลทุกตัวตั้งแต่ลำดับที่ถูกลบ จะถูกเลื่อนไปหนึ่งตำแหน่ง
	✓	✓	แทรกข้อมูลเข้าไปที่ตำแหน่งที่ระบุในรายการ โดยข้อมูลทุกตัวตั้งแต่ลำดับที่ถูกลบ จะถูกเลื่อนไปหนึ่งตำแหน่ง
	✓	✓	นำข้อมูลใหม่ ไปแทนที่ข้อมูลที่อยู่ในตำแหน่งที่ระบุในรายการ โดยข้อมูลเดิมจะสูญหายไป
	✓	✓	ให้ค่าข้อมูลในตำแหน่งที่ระบุของรายการ
	✓	✓	แสดงจำนวนข้อมูลที่มีในรายการ

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	เงื่อนไขในการตรวจสอบว่ามีข้อมูลที่ระบุ ถูกเก็บอยู่ในรายการหรือไม่ ถ้ามีจะให้ค่าเป็นจริง ถ้าไม่มีจะให้ค่าเป็นเท็จ
	✓	✓	สั่งให้แสดงชื่อและค่าของรายการบนเวที
	✓	✓	สั่งให้ซ่อนชื่อและค่าของรายการบนเวที
	✓	✓	ลบรายการ และลบบล็อกคำสั่งต่างๆ ที่เกี่ยวกับรายการบนเวทีโดยคลิกเมาส์ขวาที่บล็อกชื่อรายการ แล้วเลือก delete list

กลุ่มบล็อก Events

กลุ่มบล็อก Events มีบล็อกคำสั่งที่ใช้จัดการเหตุการณ์ต่างๆ ที่เกิดขึ้นขณะที่สคริปต์กำลังทำงาน รวมถึงการเริ่มทำ และการหยุดทำงานของสคริปต์ด้วย บล็อกในกลุ่มบล็อก Events มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อบล็อกนี้ เมื่อคลิกที่
	✓	✓	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อบล็อกนี้ เมื่อแป้นที่ระบุถูกกด
	✓		เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อบล็อกนี้ เมื่อตัวละครนี้ถูกคลิก
	✓	✓	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อบล็อกนี้ เมื่อพื้นหลังของเวทีที่ระบุถูกคลิก
	✓	✓	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อบล็อกนี้ เมื่อค่าที่ระบุเป็นไปตามเงื่อนไขที่กำหนด โดยสามารถเลือกค่าได้จาก loudness, timer หรือ video motion
	✓	✓	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อบล็อกนี้ เมื่อได้รับสารที่ระบุ
	✓	✓	กระจายสาร (ตามชื่อสารที่เลือก) ไปยังทุกตัวละครและเวที
	✓	✓	กระจายสารออกไป ผู้ส่งจะหยุดและทำคำสั่งถัดไปก็ต่อเมื่อผู้รับสาร (ตัวละครหรือเวที) ทุกตัวทำงานตามบล็อกคำสั่งของตนเองจนเสร็จสิ้น

กลุ่มบล็อก Control

กลุ่มบล็อก Control มีบล็อกคำสั่งที่ใช้ควบคุมทิศทางการทำงานของสคริปต์ การวนทำบล็อกคำสั่งซ้ำๆ การเลือกทำบล็อกคำสั่งตามเงื่อนไข บล็อกในกลุ่มบล็อก Control มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	หยุดรอเป็นระยะเวลา 1 วินาที ก่อนที่จะทำบล็อกคำสั่งถัดไป
	✓	✓	ทำคำสั่งภายในบล็อกนี้ซ้ำเป็นจำนวนครั้งที่ระบุ ตัวอย่างทำซ้ำ 10 ครั้ง แล้วจึงจะทำงานในบล็อกคำสั่งถัดไป
	✓	✓	ทำคำสั่งภายในบล็อกนี้ซ้ำไปเรื่อยๆ
	✓	✓	ทำคำสั่งภายในบล็อก if ถ้าเงื่อนไขที่กำหนดหลัง if เป็นจริง แต่ถ้าเงื่อนไขเป็นเท็จ ก็จะข้ามคำสั่งทั้งหมดที่อยู่ในบล็อกไป
	✓	✓	ทำคำสั่งภายในบล็อก if ถ้าเงื่อนไขที่กำหนดหลัง if เป็นจริง แต่ถ้าเงื่อนไขเป็นเท็จ จะทำคำสั่งภายในบล็อก else
	✓	✓	หยุดรอเวลา จนกระทั่งเงื่อนไขหลัง until เป็นจริง จึงจะทำคำสั่งต่อไป
	✓	✓	ทำคำสั่งภายในบล็อกซ้ำไปเรื่อยๆ ถ้าเงื่อนไขที่กำหนดหลัง until เป็นเท็จ และข้ามไปทำคำสั่งถัดไปถ้าเงื่อนไขที่กำหนดหลัง until เป็นจริง
			หยุดการทำงานของสคริปต์ โดยสามารถเลือก <ul style="list-style-type: none"> • all เพื่อให้หยุดการทำงานของสคริปต์ทั้งหมด • this script เพื่อให้หยุดการทำงานเฉพาะสคริปต์ที่มีบล็อก stop นี้ • other scripts in sprite เพื่อให้หยุดการทำงานเฉพาะสคริปต์อื่นๆ ของตัวละครตัวนี้
			เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้ เมื่อตัวละครนี้ถูกสร้างขึ้นด้วยการโคลน
			สร้างตัวละครโคลนเพิ่มขึ้น 1 ตัว จากตัวละครนี้
			ลบตัวละครโคลน

กลุ่มบล็อก Sensing

กลุ่มบล็อก Sensing มีบล็อกคำสั่งที่ใช้ตรวจสอบการรับรู้ต่างๆ เช่น การสัมผัส เสียง ระยะเวลา บล็อกในกลุ่มบล็อก Sensing มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓		ตรวจสอบว่าตัวละครนั้นสัมผัสกับ ตัวชี้เมาส์ ขอบเวที หรือตัวละครอื่นหรือไม่
	✓		ตรวจสอบว่าตัวละครนั้นสัมผัสกับสีที่กำหนดหรือไม่
	✓		ตรวจสอบว่าสีที่กำหนดของตัวละคร สัมผัสกับสีที่ต้องการหรือไม่
	✓		ระยะทางระหว่างตัวชี้เมาส์ถึงตัวละคร
	✓	✓	ถามคำถามที่กำหนด แล้วรอคำตอบ
	✓	✓	คำตอบที่ผู้ใช้กรอก จากคำถาม ask (สามารถคลิกที่ช่อง เพื่อแสดงค่าคำตอบบนเวที)
	✓	✓	ตรวจสอบการกดแป้นพิมพ์บนคีย์บอร์ด
	✓	✓	ตรวจสอบเมาส์ว่าถูกคลิกหรือไม่
	✓	✓	ตำแหน่งตัวชี้เมาส์ตามแนวแกน x
	✓	✓	ตำแหน่งตัวชี้เมาส์ตามแนวแกน y
	✓	✓	ระดับความดังของเสียงที่รับได้จากไมโครโฟน มีค่าในช่วง 1 ถึง 100 (สามารถคลิกที่ช่องเพื่อให้เห็นค่าระดับของเสียงบนเวที)
	✓	✓	กำหนดให้ตรวจจับ ความเคลื่อนไหว (motion) หรือทิศทาง (direction) ของภาพที่จับได้จากกล้องวิดีโอ โดยสามารถเลือกขนาดของพื้นที่ตรวจจับว่าจะให้เป็นพื้นที่ที่มีขอบเขตเท่ากับ ขอบเขตของตัวละคร (this sprite) ที่สคริปต์นี้รันอยู่ หรือเป็นพื้นที่ของทั้งเวที (Stage)
	✓	✓	ควบคุมการทำงานของกล้องวิดีโอ โดย <ul style="list-style-type: none"> on เป็นการเปิดการทำงาน off เป็นการปิดการทำงาน on-flipped เป็นการเปิดการทำงานจับภาพแบบกลับด้าน
	✓	✓	กำหนดความโปร่งใสของภาพที่กล้องวิดีโอจับได้ โดย 0% เป็นภาพทึบ และ 100% เป็นภาพโปร่งใส (ไม่เห็นภาพเลย)

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	แสดงเวลาเป็นวินาทีบนเวที
	✓	✓	กำหนดเวลาของตัวนับให้กลับไปเริ่มที่ 0
	✓	✓	ข้อมูลตัวละคร เช่น ตำแหน่งตามแนวแกน x, ตำแหน่งตามแนวแกน y, มุม, ลำดับชุดตัวละคร, ขนาด, ระดับความดังของเสียง
	✓	✓	แสดงเวลาปัจจุบันบนเวทีในหน่วยต่างๆ ดังนี้ <ul style="list-style-type: none"> • year บอกปี • month บอกเดือน • date บอกวันที่ • day of week บอกวันในอาทิตย์ (1=อาทิตย์, 2=จันทร์, ...) • hour บอกชั่วโมง • minute บอกนาที • second บอกวินาที
	✓	✓	ค่าของจำนวนวันนับตั้งแต่วันที่ 1 มกราคม ปี ค.ศ. 2000
	✓	✓	ชื่อผู้ใช้ที่กำลังใช้งานสคริปต์นี้อยู่ (กรณีใช้ Scratch online เท่านั้น)
	✓	✓	ตรวจสอบว่าระดับเสียงที่ไมโครโฟนจับได้สูงกว่า 30 หรือไม่ ในรูปค่าความจริง (True) หรือ เท็จ (False)
	✓	✓	ค่าที่ได้จากอุปกรณ์ตรวจจับต่างๆ ที่ระบุ ซึ่งจะต้องใช้งานร่วมกับอุปกรณ์เสริม
	✓	✓	ตรวจสอบการกดปุ่ม หรือการเชื่อมต่ออุปกรณ์ตรวจจับ

กลุ่มบล็อก Operators

กลุ่มบล็อก Operators มีบล็อกคำสั่งที่เป็นตัวดำเนินการต่างๆ ทั้งด้านคำนวณ เปรียบเทียบ และ ตรรกะ สำหรับใช้ร่วมกับบล็อกคำสั่งที่มีการกำหนดเงื่อนไข บล็อกในกลุ่มบล็อก Operators มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
	✓	✓	ตัวดำเนินการบวก
	✓	✓	ตัวดำเนินการลบ
	✓	✓	ตัวดำเนินการคูณ
	✓	✓	ตัวดำเนินการหาร
	✓	✓	สุ่มเลขจำนวนเต็มในช่วงที่ระบุ
	✓	✓	ตัวดำเนินการเปรียบเทียบ ให้ค่าเป็นจริง ถ้าค่าแรก น้อยกว่า ค่าหลัง
	✓	✓	ตัวดำเนินการเปรียบเทียบ ให้ค่าเป็นจริง ถ้าค่าทั้งสองเท่ากัน
	✓	✓	ตัวดำเนินการเปรียบเทียบ ให้ค่าเป็นจริง ถ้าค่าแรก มากกว่าค่าหลัง
	✓	✓	ตัวดำเนินการทางตรรกะ “และ” (and)
	✓	✓	ตัวดำเนินการทางตรรกะ “หรือ” (or)
	✓	✓	ตัวดำเนินการทางตรรกะ “นิเสธ” (not)
	✓	✓	ให้ค่าของการเชื่อมข้อความสองข้อความเข้าด้วยกัน โดยข้อความที่กำหนดอาจอยู่ในรูปของตัวแปรก็ได้
	✓	✓	ให้ค่าอักขระจากข้อความ ณ ตำแหน่งที่ระบุ
	✓	✓	ให้ค่าจำนวนอักขระของข้อความที่กำหนด
	✓	✓	ตัวดำเนินการหารเอาเศษ (modulo)
	✓	✓	ปัดเศษค่าตัวเลข ให้เป็นจำนวนเต็ม
	✓	✓	หาค่าฟังก์ชันทางคณิตศาสตร์ของตัวเลขที่กำหนด โดยฟังก์ชันที่มีให้เลือก ได้แก่ abs, sqrt, sin, cos, tan, asin, acos, atan, ln, log, e [^] และ 10 [^]

กลุ่มบล็อก More Blocks

กลุ่มบล็อก More Blocks เป็นบล็อกคำสั่งที่ใช้สร้างบล็อกเพิ่มเติมในรูปของฟังก์ชัน หรือเพิ่มบอร์ดขยายความสามารถในการตรวจสอบการรับรู้ บล็อกในกลุ่มบล็อก More Blocks มีดังนี้

บล็อก	ตัวละคร	เวที	ความหมาย
Make a Block	✓	✓	สร้างเพิ่มเติมในรูปของฟังก์ชัน
Add an Extension	✓	✓	เพิ่มบอร์ดขยายความสามารถในการตรวจสอบการรับรู้

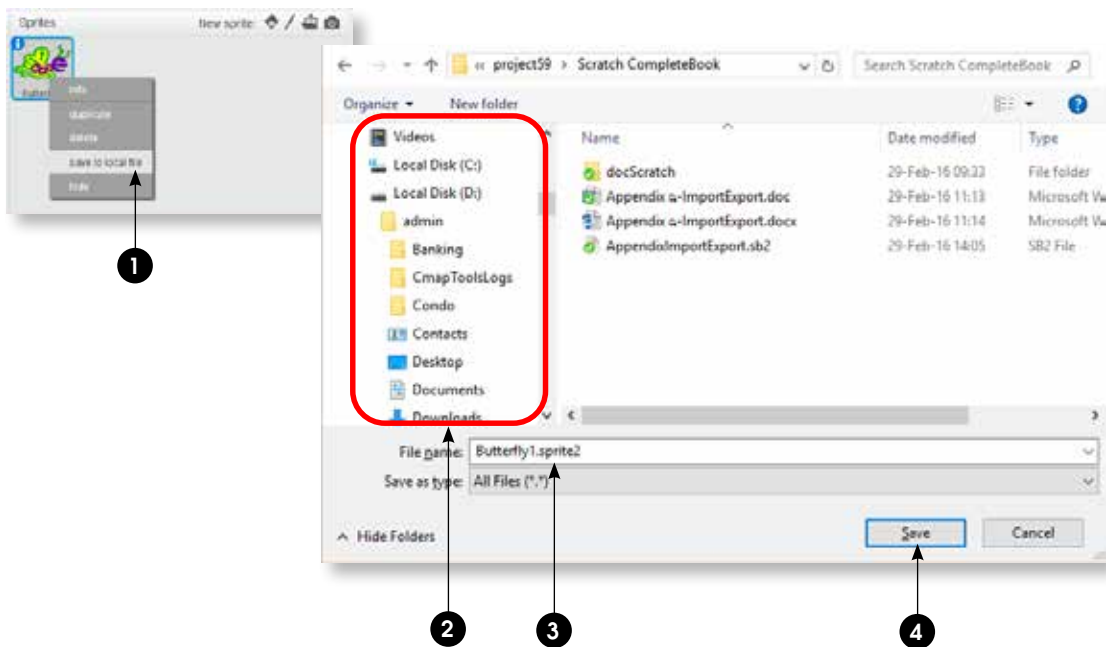


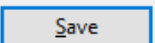
ภาคผนวก ง การส่งออกและนำเข้าตัวละคร

โปรแกรม Scratch อนุญาตให้ผู้ใช้ส่งออกและนำเข้า (export /import) ตัวละคร เสียง และเวทีได้ ผู้ใช้สามารถแบ่งปันสิ่งเหล่านี้ระหว่างโปรเจกต์ของ Scratch ได้สะดวก ในที่นี้จะนำเสนอการส่งออกและนำเข้าตัวละครที่มีอยู่แล้วมารวมเข้ากับโปรเจกต์ปัจจุบัน

การส่งออกตัวละคร


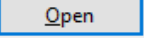
การส่งออกตัวละครเป็นการเก็บข้อมูลของตัวละครไว้ในรูปของไฟล์ที่มีนามสกุลเป็น .sprite2 ถ้าตัวละครมีการกำหนดชุดหรือเสียง รวมถึงสคริปต์และตัวแปรไว้ ก็จะมีการเก็บรวมไว้กับตัวละครที่ส่งออกนั้นด้วย การส่งออกตัวละคร ทำได้ดังนี้

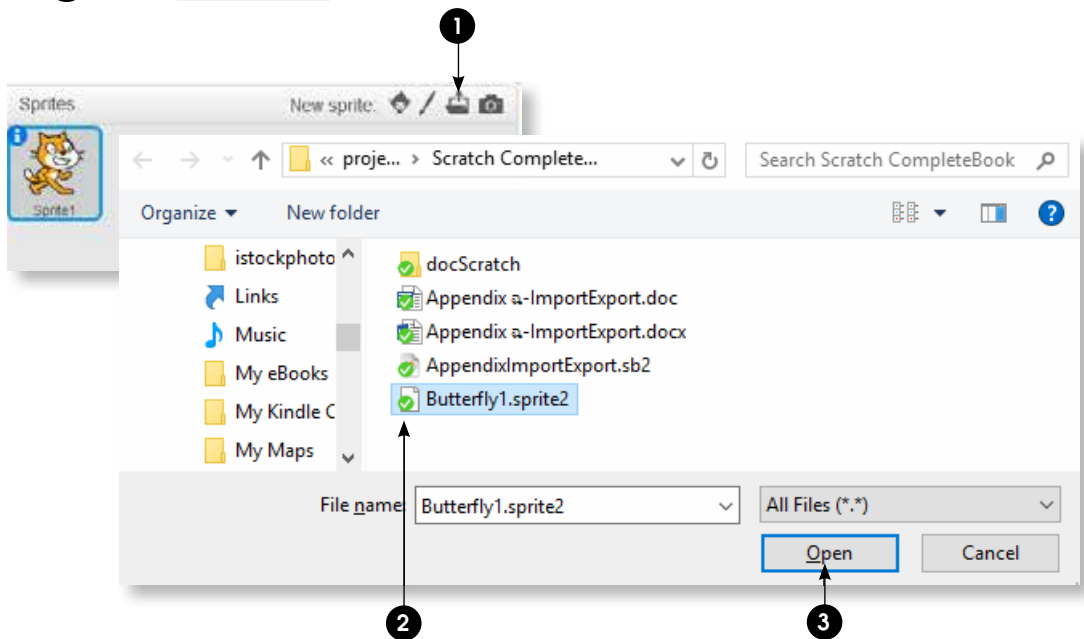


- 1 คลิกขวาที่ตัวละครที่ต้องการส่งออก เลือก "save to local file"
- 2 เลือกโฟลเดอร์ที่ต้องการจัดเก็บ
- 3 ตั้งชื่อไฟล์ที่ต้องการบันทึกให้กับตัวละครที่ส่งออกไป ซึ่งโปรแกรม Scratch จะเติม .sprite2 ให้เองโดยอัตโนมัติ
- 4 คลิก 

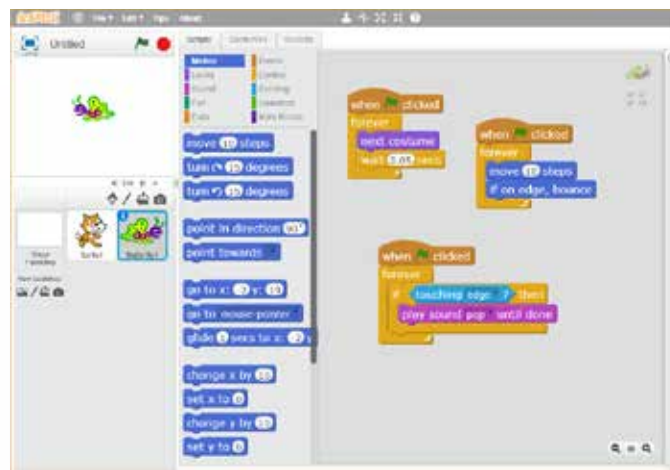
การนำเข้าตัวละคร

การนำเข้าตัวละครเป็นการนำไฟล์ตัวละครที่มีส่วนขยายเป็น .sprite2 เข้ามาในโปรเจกต์ที่กำลังใช้งานอยู่ ซึ่งประกอบด้วย ละคร และองค์ประกอบอื่น เช่น เสียง ชูดตัวละคร สคริปต์ และตัวแปร ส่งผลให้สามารถนำตัวละครที่เคยสร้างไว้แล้วกลับมาใช้ซ้ำได้อีก การนำเข้าตัวละครทำได้ดังนี้

- 1 คลิกที่ไอคอน 
- 2 เลือกไฟล์ตัวละครที่ต้องการ (หากต้องการนำเข้าตัวละครหลายตัวสามารถกดแป้น shift หรือ control ค้างไว้แล้วเลือกตัวละครที่ต้องการได้)
- 3 คลิก 



ตัวละครใหม่จะถูกนำเข้ามาพร้อมกับตัวละครที่มีอยู่เดิม รวมไปถึงองค์ประกอบต่างๆ ของตัวละครใหม่ด้วย ดังรูป




ข้อสังเกต ผู้ใช้ควรตรวจสอบผลของการนำเข้าว่าเป็นไปตามต้องการ หรือเพื่อลดความผิดพลาดที่อาจเกิดขึ้นได้

ภาคผนวก ๑ การแปลงไฟล์ Scratch ให้เป็น .exe

โดยปกติแล้ว เมื่อเราสร้างชิ้นงานจากโปรแกรม Scratch โปรแกรมจะทำการบันทึกไฟล์ที่สร้างเป็น .sb2 ซึ่งสามารถเปิดใช้งานในเครื่องที่มีโปรแกรม Scratch ติดตั้งอยู่แล้วเท่านั้น หรือผู้ใช้สามารถแปลงชิ้นงานที่สร้างผ่านทางเว็บไซต์ <http://scratch.mit.edu/> ซึ่งได้กล่าวถึงไปแล้วในหัวข้อเว็บไซต์ Scratch แต่ถ้าเครื่องคอมพิวเตอร์ไม่มีโปรแกรม Scratch ในที่นี้ขอแนะนำโปรแกรม Scratch2Exe ในการแปลงไฟล์จาก .sb2 เป็น .exe เพื่อความสะดวกในการเปิดชิ้นงานที่สร้างด้วยโปรแกรม Scratch ซึ่งโปรแกรมนี้สามารถดาวน์โหลดได้จากเว็บไซต์ <http://oho.ipst.ac.th/download/program/scratch/setup2exe-ch-se-2014.3.27.zip>

การติดตั้งโปรแกรม Scratch2Exe

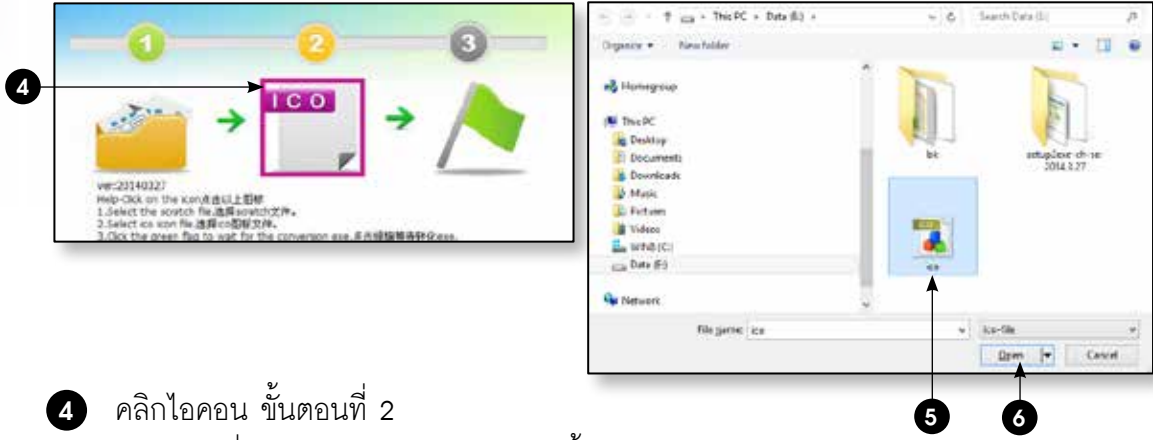
- 1 ดับเบิลคลิกที่ไฟล์ setup2exe-ch-se.exe
- 2 คลิก **下一步(N) >** (Next) เพื่อดำเนินการต่อไป
- 3 คลิก **安装(I)** (Install) เพื่อทำการติดตั้งโปรแกรม
- 4 คลิก **完成(F)** (Finish) เสร็จสิ้นการลงโปรแกรม

จะปรากฏไอคอน  ที่เดสก์ทอป และโปรแกรมจะเปิดอัตโนมัติ

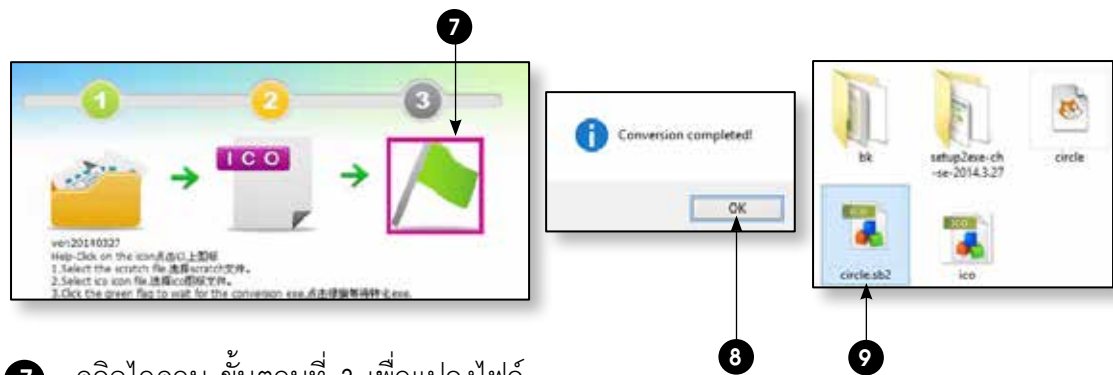
วิธีการแปลงชิ้นงานจาก Scratch ให้เป็นไฟล์ exe

- 1 คลิกไอคอน ขั้นตอนที่ 1
- 2 เลือกไฟล์โปรแกรมที่เขียนขึ้นด้วย Scratch

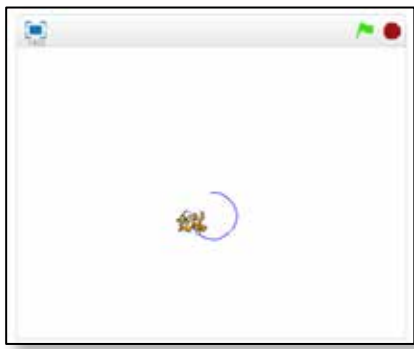
3 คลิก



- 4 คลิกไอคอน ขั้นตอนที่ 2
- 5 เลือกไฟล์ที่จะใช้เป็นไอคอน (.ico) ของชิ้นงาน
- 6 คลิก




- 7 คลิกไอคอน ขั้นตอนที่ 3 เพื่อแปลงไฟล์
- 8 คลิก
- โปรแกรมจะแปลงไฟล์และบันทึกไว้ที่โฟลเดอร์เดียวกับไฟล์ต้นฉบับ
- 9 ทดลองการทำงานของโปรแกรม โดยเปิดไฟล์ .exe ที่ได้

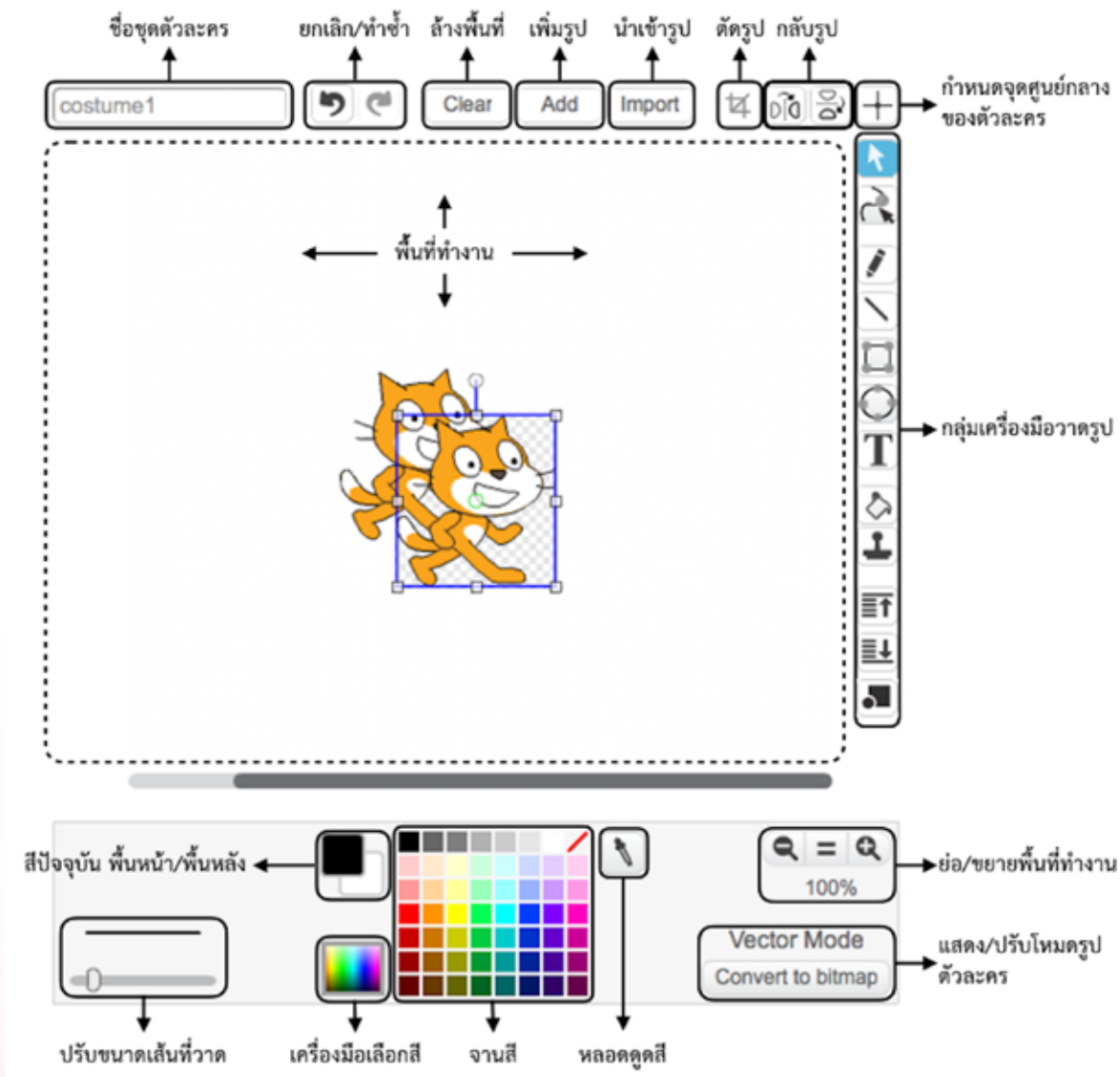


ภาคผนวก ง การใช้งานคำสั่งอื่นๆ










ในบทนี้จะกล่าวถึงคำสั่งเพิ่มเติมนอกเหนือจากกิจกรรมที่ 1-9 ซึ่งสามารถนำไปพัฒนาโปรเจกต์ให้มีความซับซ้อนมากขึ้น มีความน่าสนใจ และมีการประยุกต์ใช้งานที่หลากหลาย โดยจะกล่าวถึง เครื่องมือใน Paint Editor การใช้เลเยอร์ การใช้งานตัวแปรและรายการอย่างง่าย การโคลนร่างตัวละคร การโต้ตอบกับผู้ใช้ผ่านเว็บแคม โดยมีรายละเอียดการใช้งานดังนี้

เครื่องมือใน Paint Editor

การสร้างชุดตัวละครขึ้นมาใหม่ใน Scratch ทำได้โดยคลิกปุ่ม  หรือคลิกที่ชุดตัวละครที่มีอยู่แล้ว หรือพื้นหลัง ก็จะสามารถสร้างหรือแก้ไขชุดตัวละครหรือพื้นหลังได้จากหน้าต่างโปรแกรมวาดรูป (Paint Editor) ดังนี้



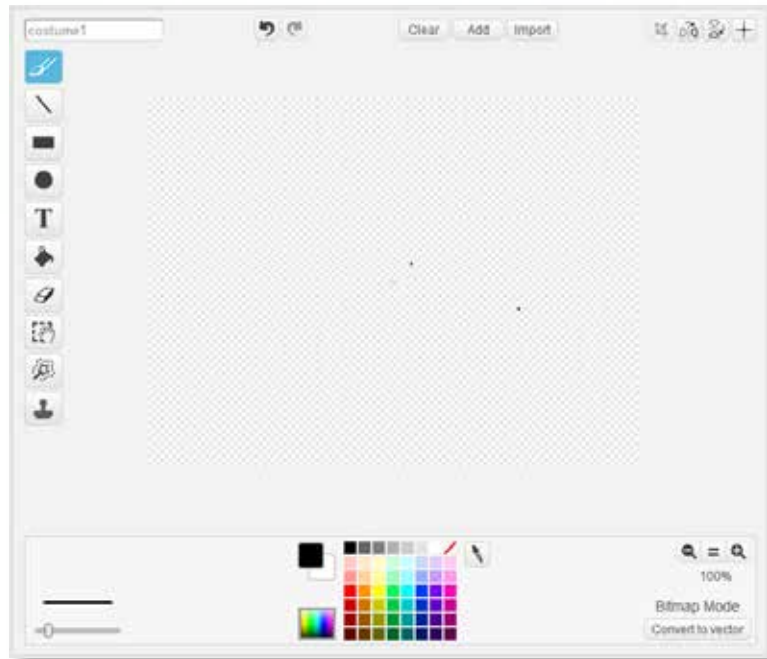
เครื่องมือในโหมดรูปเวกเตอร์

เครื่องมือ	ชื่อ	หน้าที่
	ตัวเลือก	เลือกวัตถุ
	เลือกขอบเขต	เลือกขอบเขตที่จะใส่สี
	ดินสอ	วาดภาพ
	เส้นตรง	วาดเส้นตรง
	สี่เหลี่ยม	วาดภาพสี่เหลี่ยม
	วงกลม/วงรี	วาดภาพวงกลมหรือวงรี
	ข้อความ	เขียนข้อความ
	ถังเทสี	เทสี
	ตราประทับ	คัดลอกและทำซ้ำบริเวณที่เลือก










สิ่งที่สำคัญของการสร้างตัวละครด้วยวิธีนี้คือ การกำหนดจุดศูนย์กลางของชุดตัวละคร ซึ่งนอกจากใช้บอกตำแหน่งพิกัดของตัวละครแล้ว ยังเป็นจุดอ้างอิงสำหรับการเคลื่อนที่ การหมุน และการสลับทิศ

สำหรับใน Scratch version 2 นี้ ยังสามารถปรับเปลี่ยนชนิดของรูปชุดตัวละครหรือพื้นหลัง ระหว่างโหมดภาพแบบเวกเตอร์ (Vector Mode) หรือแบบบิตแมพได้ด้วย (Bitmap Mode) โดยแบบเวกเตอร์ จะทำให้การขยายภาพใหญ่ขึ้น ได้ภาพที่ไม่แตก แต่สีอาจไม่ละเอียดเท่าภาพแบบบิตแมพ อย่างไรก็ตาม การเปลี่ยนโหมดของภาพจากเดิมที่เคยสร้างไว้ในโหมดหนึ่ง ไปเป็นอีกโหมดหนึ่ง อาจทำให้ได้คุณภาพของภาพในโหมดใหม่เสียไปได้ จึงควรกำหนดโหมดของภาพตามที่ต้องการตั้งแต่เริ่มสร้างภาพใหม่

ในโหมดบิตแมพจะมีส่วนประกอบคล้ายโหมดเวกเตอร์ โดยมีหน้าต่างดังนี้



เครื่องมือวาดรูปในโปรแกรมมีดังนี้

เครื่องมือ	ชื่อ	หน้าที่
	ฟู่กัน	วาดหรือระบาย
	เส้นตรง	วาดเส้นตรง
	สี่เหลี่ยม	วาดภาพสี่เหลี่ยม
	วงกลม/วงรี	วาดภาพวงกลมหรือวงรี
	ข้อความ	เขียนข้อความ
	ถังเทสี	เทสี
	ยางลบ	ลบ
	เลือกพื้นที่	เลือกพื้นที่เพื่อใช้ในการปรับแต่งเคลื่อนย้าย ลบ ย่อ/ขยายภาพ
	เลือกวัตถุ	เลือกวัตถุที่จะใส่สีหรือเปลี่ยนสี
	ตราประทับ	ตัดลอกและทำซ้ำบริเวณที่เลือก

การใช้เลเยอร์ (layers)

เลเยอร์ คือ ลำดับชั้นการวางซ้อนกันของตัวละคร ลักษณะการทำงานแบบเลเยอร์ เป็นเหมือนการวางแผ่นใส ซ้อนทับกันเป็นลำดับชั้น โดยแต่ละแผ่นใส เปรียบเสมือนเป็นแต่ละเลเยอร์ บริเวณของเลเยอร์ที่ไม่มีรูป จะเห็นทะลุถึงเลเยอร์ที่อยู่ข้างล่าง โดยกระบวนการเช่นนี้ จะทำให้เกิดเป็นภาพที่สมบูรณ์ และทำให้เราสามารถจัดวางตำแหน่งของตัวละครได้ง่าย ผู้ใช้สามารถกำหนดชั้นของเลเยอร์เพื่อใช้วางตัวละครได้

คำสั่ง **go to front** เลื่อนชั้นเลเยอร์มาข้างหน้า 1 เลเยอร์

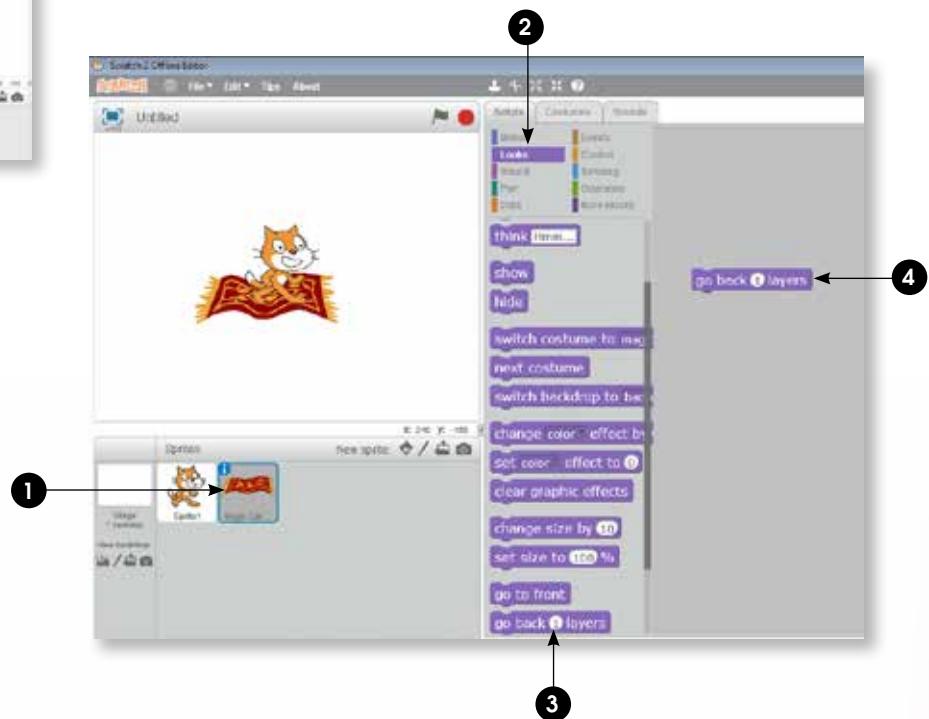
คำสั่ง **go back 1 layers** เลื่อนชั้นเลเยอร์มาข้างหลัง 1 เลเยอร์

ตัวอย่างการใช้งาน

โดยปกติแล้วเลเยอร์ของตัวละคร จะเรียงตามลำดับการนำตัวละครเข้ามาสู่เวที



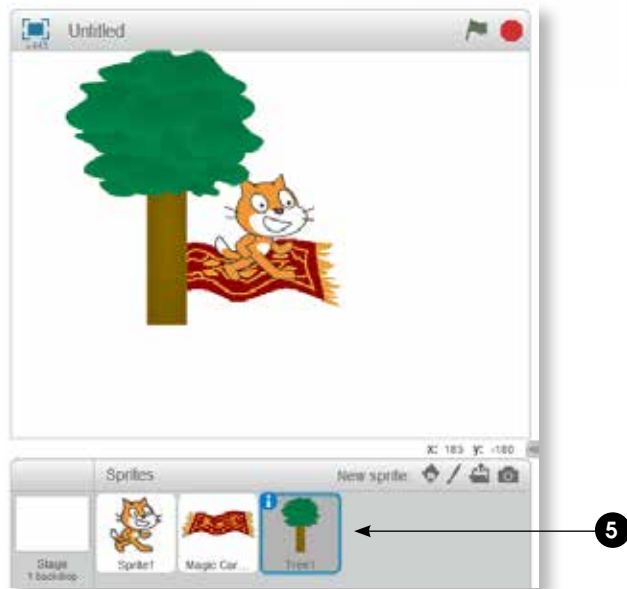
จากภาพ ผู้ใช้นำตัวละคร Spite1 เข้ามาก่อนตัวละคร Spite2 ดังนั้นในการแสดงผลบนเวที ตัวละคร Spite2 จะวางทับตัวละคร Spite1 การแก้ไขสามารถทำได้โดยใช้บล็อก **go back 1 layers** ดังนี้



- 1 คลิกที่ตัวละคร spite2
- 2 คลิกกลุ่มบล็อก Looks
- 3 เลือกบล็อก **go back 1 layers**

4 วางบล็อก **go back 1 layers** กำหนดค่า ให้เป็น 1 เนื่องจาก ต้องการให้ตัวละคร sprite2 ไปอยู่ด้านหลัง 1 เลเยอร์

5 เพิ่มตัวละคร sprite3 (ต้นไม้)



6 คลิกที่ตัวละคร sprite3 ใช้คำสั่ง **go back 2 layers** เพื่อให้ตัวละคร sprite3 ไปอยู่ด้านหลัง 2 เลเยอร์ ผลลัพธ์ที่ได้เป็นดังรูป
ถ้าต้องการให้ตัวละครมาอยู่ด้านหน้า 1 เลเยอร์ สามารถใช้คำสั่ง **go to front**

การใช้งานตัวแปรและรายการอย่างง่าย

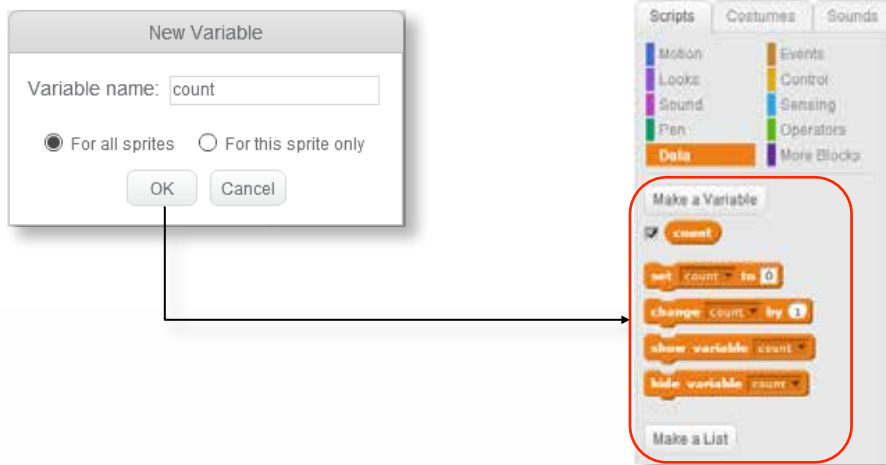
ตัวแปร (Variables)

ตัวแปรเปรียบเสมือนที่เก็บข้อมูลชั่วคราว ใช้เก็บข้อมูลได้เพียงค่าเดียว ซึ่งค่าที่เก็บไว้สามารถนำไปใช้ได้ หรือเปลี่ยนแปลงให้เป็นค่าใหม่ได้ ตัวอย่างการนำตัวแปรไปใช้ เช่น นำไปเก็บคะแนนในการเล่นเกมนับตำแหน่งในแกน X ของตัวละคร หรือเก็บเวลาที่เกมดำเนินไป เป็นต้น

โปรแกรม Scratch มีทั้งตัวแปรที่สร้างมาให้แล้ว และตัวแปรที่ผู้ใช้สร้างขึ้นเอง สำหรับตัวแปรที่มีมาให้และนำไปใช้งานได้เลย เช่น **timer** ใช้เก็บเวลาที่ดำเนินไปของโปรแกรม หรือ **answer** ใช้เก็บคำตอบที่ผู้ใช้ป้อนเข้ามาจากการทำงานของบล็อก **ask and wait**

สำหรับตัวแปรที่ผู้ใช้สร้างขึ้นเอง ก่อนที่จะใช้งานตัวแปรได้ ผู้ใช้ต้องสร้างและตั้งชื่อให้กับตัวแปรก่อน จากนั้นจะสามารถกำหนดค่าที่ต้องการเก็บให้กับตัวแปรได้ การทำงานกับตัวแปรที่ผู้ใช้สร้างขึ้นเองสามารถทำได้ดังนี้

- การสร้างตัวแปรใหม่ ให้คลิกที่ปุ่ม **Make a Variable** แล้วกำหนดชื่อให้กับตัวแปร โดยผู้ใช้สามารถเลือกได้ว่าจะให้ตัวแปรนี้ใช้งานได้สำหรับทุกตัวละคร (คลิกเลือกที่ For all sprites) หรือใช้งานได้เฉพาะสำหรับตัวละครนี้ (คลิกเลือกที่ For this sprite only) จากนั้นจะปรากฏบล็อกคำสั่งที่สามารถใช้งานได้กับตัวแปรนี้ ดังรูปได้



- การกำหนดให้ซ่อนหรือแสดงค่าของตัวแปรบนเวที สามารถทำได้โดย การคลิกให้ปรากฏเครื่องหมาย ✓ ที่หน้าชื่อตัวแปร ดังหมายเลข ❶ จะปรากฏค่าของ ตัวแปรบนเวทีดังหมายเลข ❷ โดยค่าเริ่มต้นของตัวแปรจะเป็นค่าจำนวนเต็ม 0

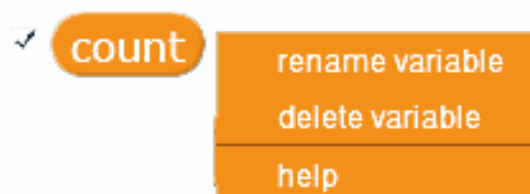


แต่ถ้าหากต้องการกำหนดให้ตัวแปรที่แสดงบนเวที หรือซ่อนจากเวทีในขณะที่สคริปต์กำลังทำงาน อยู่ สามารถทำได้โดยใช้บล็อก `show variable count` หรือ `hide variable count` เป็นส่วนประกอบของสคริปต์ได้

- การนำตัวแปรไปใช้งานร่วมกับบล็อกคำสั่งอื่น ตัวอย่างเช่น

บล็อก	คำอธิบาย
	บล็อกที่กำหนดให้ตัวแปร คะแนนรวม มีค่าเป็น 5
	บล็อกที่กำหนดให้ตัวแปร คะแนนรวม มีค่าเพิ่มขึ้นจากค่าเดิมอีก 5 โดยใช้งานร่วมกับกลุ่มบล็อก Operators
	กำหนดค่าใหม่ให้กับตัวแปร โดยมีค่าเพิ่มขึ้นจากเดิมอีก 1
	นำค่าในตัวแปรไปใช้เปรียบเทียบในเงื่อนไข โดยเปรียบเทียบว่าตัวแปร คะแนนรวม มีค่ามากกว่า 2000 หรือไม่

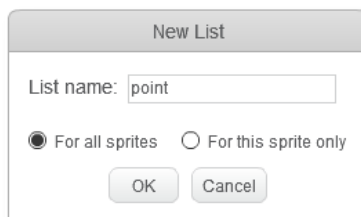
- การลบตัวแปรที่ไม่ต้องการใช้แล้ว ให้คลิกขวาตัวแปรที่ต้องการลบแล้วเลือกคำสั่ง `delete variable` ซึ่งมีข้อควรระวัง คือ เมื่อมีการลบตัวแปร จะต้องตรวจสอบสคริปต์ที่มีการนำตัวแปร ดังกล่าวไปใช้ด้วย เพราะผลลัพธ์ที่ได้อาจมีความผิดพลาด



รายการ (Lists)

รายการเปรียบเสมือนตัวแปรที่เก็บข้อมูลได้หลายค่าในขณะเดียวกัน โดยข้อมูลแต่ละตัวในรายการจะมีหมายเลขกำกับ และเก็บเรียงกันไปตั้งแต่ตัวแรกเป็นลำดับที่ 1 และตัวต่อไปเป็นลำดับที่ 2, 3 ไปเรื่อยๆ รายการสามารถนำไปใช้เก็บประวัติคะแนนสูงสุดของการเล่นเกม เก็บรายชื่อและหมายเลขโทรศัพท์ของเพื่อน เก็บชื่อนักเรียนในชั้นเรียน เป็นต้น โปรแกรม Scratch มีบล็อกคำสั่งต่างๆ ที่ช่วยให้ผู้ใช้จัดการกับข้อมูลในรายการได้ การทำงานกับรายการ สามารถทำได้ดังต่อไปนี้









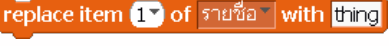


- การสร้างรายการใหม่ ทำได้โดยคลิกปุ่ม **Make a List** แล้วกำหนดชื่อให้กับรายการ โดยผู้ใช้สามารถเลือกได้ว่าจะให้รายการนี้ใช้งานได้สำหรับทุกตัวละคร (คลิกเลือกที่ For all sprites) หรือใช้งานได้เฉพาะสำหรับตัวละครนี้ (คลิกเลือกที่ For this sprite only)




จากนั้นจะปรากฏบล็อกคำสั่งที่สามารถใช้งานได้กับรายการ ดังนี้








- การเพิ่มข้อมูลเข้าไปในรายการ สามารถทำได้โดยใช้บล็อกคำสั่งดังนี้

บล็อก	คำอธิบาย
	<p>เป็นการเพิ่มข้อมูลเข้าไปต่อท้ายสุดของรายการ ข้อมูลที่เพิ่มจะมีหมายเลขกำกับ โดยลำดับถัดจากข้อมูลตัวสุดท้ายที่มีอยู่ เช่น เมื่อเพิ่มข้อมูล "อุษณีย์" เข้าไปในรายการ "รายชื่อ" ที่ยังว่างอยู่ ด้วยคำสั่ง  ข้อมูลที่มีค่าเป็น "อุษณีย์" จะถูกเก็บเข้าไปในรายการที่ตำแหน่งหมายเลข 1 ดังรูปที่ 1 ถ้าใช้บล็อกคำสั่ง  รายการจะมีข้อมูลเก็บอยู่ทั้งหมดเป็นดังรูปที่ 2</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p>รูปที่ 1</p> </div> <div style="text-align: center;">  <p>รูปที่ 2</p> </div> </div>
	<p>เป็นการแทรกข้อมูลเข้าไปในรายการที่มีข้อมูลอยู่แล้ว เช่น ถ้าต้องการแทรกข้อมูล "จามรรัตน์" เข้าไปที่ตำแหน่งที่ 2 ทำได้โดยใช้บล็อกคำสั่ง  ซึ่งจะได้ผลลัพธ์ดังรูป ให้สังเกตว่าข้อมูลเดิมในรายการลำดับที่ 2 จะถูกเลื่อนลงไปเป็นลำดับที่ 3 และข้อมูลใหม่ถูกจัดเก็บในลำดับที่ 2</p> <div style="text-align: center;">  </div>
	<p>เป็นการนำข้อมูลใหม่เข้าไปเก็บในรายการ โดยที่ข้อมูลใหม่จะถูกนำไปแทนที่ข้อมูลที่มีอยู่แล้ว และข้อมูลที่ถูกแทนที่จะสูญหายไป พร้อมกับกำหนดลำดับของข้อมูลเดิมในรายการจะถูกแทนที่ไปด้วยข้อมูลใหม่ เช่น ถ้าต้องการเก็บข้อมูล "อลงกต" ในลำดับที่ 1 แทนค่าข้อมูล "อุษณีย์" จะทำได้โดยใช้บล็อกคำสั่ง  ทำให้ได้ผลลัพธ์ดังรูป</p> <div style="text-align: center;">  </div>

- การหาจำนวนของข้อมูลที่มีเก็บอยู่ในรายการ ทำได้โดยใช้บล็อกคำสั่ง 


- การนำค่าข้อมูลที่เก็บในรายการไปใช้ แบ่งเป็น 2 กรณี คือ

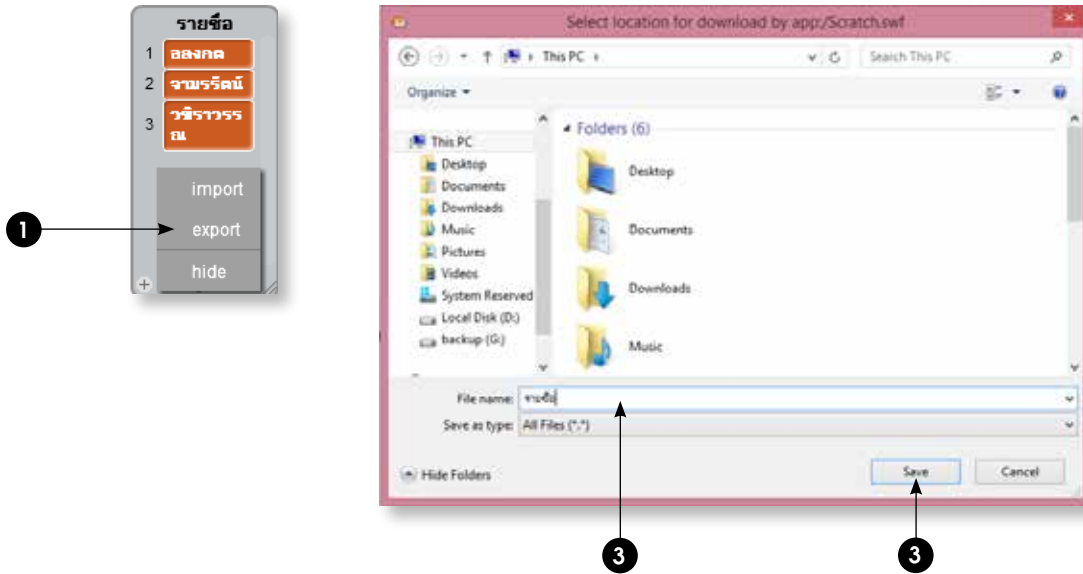
บล็อก	คำอธิบาย
	การหาค่าข้อมูลในลำดับที่ต้องการ โดยกำหนดลำดับที่ของข้อมูลที่ต้องการ พร้อมกับชื่อรายการ เช่น  จะหมายถึงข้อมูล “อลงกต”
	การตรวจสอบว่ามีข้อมูลที่กำหนดในรายการหรือไม่ เช่น ถ้าต้องการหาว่าในรายการมีข้อมูล “วชิรารวรรณ” อยู่หรือไม่ ให้ใช้บล็อกคำสั่ง  ซึ่งจะได้ค่าเป็นจริง แต่ถ้าต้องการหาว่าในรายการมีข้อมูล “อุษณีย์” อยู่หรือไม่ ต้องใช้บล็อกคำสั่ง  ซึ่งจะได้ค่าเป็นเท็จ เพราะในรายการไม่มีข้อมูล “อุษณีย์” เก็บไว้แล้ว และสามารถใช้การตรวจสอบนี้ร่วมกับบล็อกคำสั่งตรวจสอบเงื่อนไขได้

ตัวอย่างหนึ่งของการหาค่าข้อมูลที่เก็บในรายการ เช่น ถ้าต้องการหาว่าข้อมูล “วชิรารวรรณ” ถูกเก็บอยู่ในลำดับที่เท่าใดของรายการ สามารถใช้สคริปต์ต่อไปนี้ในการค้นหา ถ้าหาพบ สคริปต์จะให้ตัวละครพูดหมายเลขลำดับออกมาเป็นเวลา 10 วินาที

```

when clicked
  set name to วชิรารวรรณ
  set count to 0
  repeat until count > length of รายชื่อ
    if item count of รายชื่อ = name then
      say join I found you at number count for 10 secs
      change count by 1
  
```

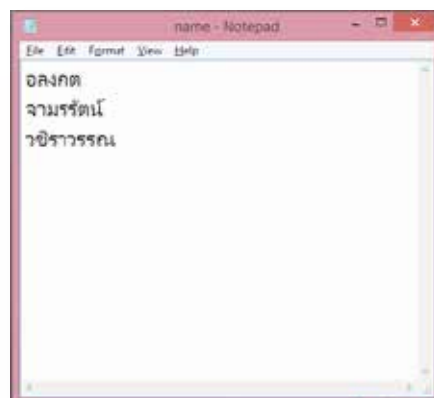
- การลบค่าข้อมูลในรายการ ต้องระบุหมายเลขลำดับของข้อมูลที่ต้องการลบในรายการ เช่น ถ้าต้องการลบข้อมูลในลำดับที่ 3 (“วชิรารวรรณ”) ให้ใช้บล็อกคำสั่ง 
- การลบรายการที่ไม่ต้องการใช้แล้ว ให้คลิกขวรายการที่ต้องการลบแล้วเลือกคำสั่ง delete ซึ่งมีข้อควรระวังเช่นเดียวกับการลบตัวแปร คือ ถ้ามีการนำรายการที่ลบไปใช้ในบล็อกคำสั่งใด จะต้องตามไปลบบล็อกคำสั่งเหล่านั้นด้วย เพราะโปรแกรม Scratch ไม่ได้ลบบล็อกคำสั่งต่างๆ เหล่านั้นให้โดยอัตโนมัติ
- การส่งออกรายการ รายการที่มีอยู่แล้วใน Scratch ยังสามารถถูกส่งออก (export) ไปเก็บไว้ในรูปของแฟ้มข้อมูล เพื่อให้เก็บข้อมูลของรายการไว้ใช้ต่อไปได้ ขั้นตอนการส่งออกมีดังนี้
 1. คลิกขวาที่รายการ เลือก “export”
 2. เลือกไฟล์เดอร์ และตั้งชื่อแฟ้มที่ต้องการให้บันทึกเก็บไว้ โดย Scratch จะตั้งชื่อมาให้ เป็นชื่อของรายการที่กำลังส่งออก ซึ่งสามารถเปลี่ยนแปลงได้ตามต้องการ
 3. คลิกปุ่ม Save



ซึ่งเมื่อเปิดดูในแฟ้ม รายการ .txt ดังกล่าว จะมีข้อมูลแต่ละลำดับของรายการ ถูกเก็บเรียงต่อกัน ตั้งแต่ข้อมูลลำดับแรกจนถึงลำดับสุดท้าย ดังนี้

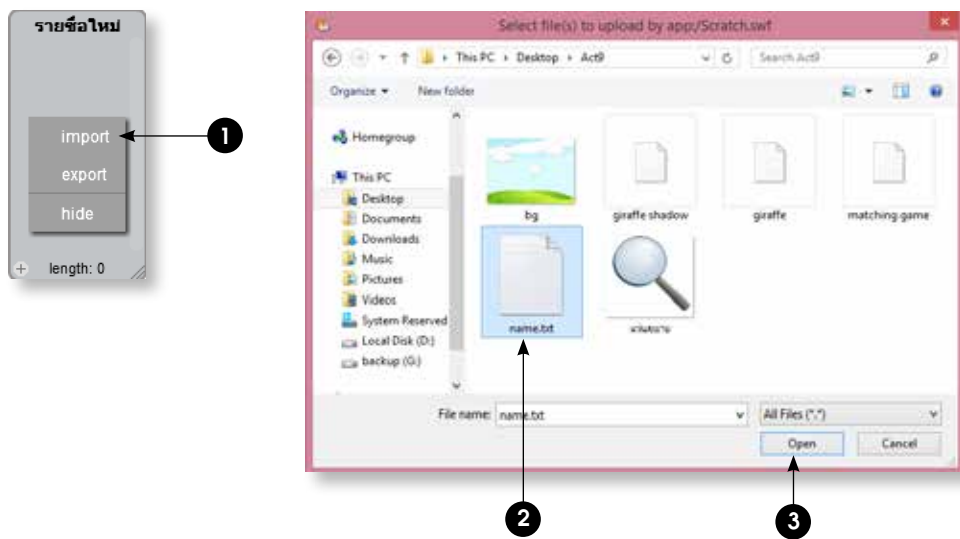
อลงกตจา มรรัตน์วชิราวรรณ

ในกรณีที่ต้องการนำเข้าข้อมูลในไฟล์ เข้ามาในโปรแกรม Scratch โดยไม่ต้องป้อนข้อมูลรายการใหม่ทีละค่าจะต้องจัดเตรียมไฟล์โดยแบ่งข้อมูลแต่ละชุดโดยการขึ้นบรรทัดใหม่ ดังตัวอย่างไฟล์นำเข้าต่อไปนี้



ขั้นตอนการนำเข้าไฟล์มีดังนี้

- 1 คลิกขวาที่รายการ เลือก "import"
- 2 เลือกไฟล์เดอร์ และเลือกชื่อแฟ้มที่เก็บรายการที่ต้องการนำเข้า ในที่นี้จะนำเข้าข้อมูลรายการจากไฟล์ชื่อ name.txt เข้าไปในรายการที่ชื่อ รายชื่อใหม่
- 3 คลิกปุ่ม Open



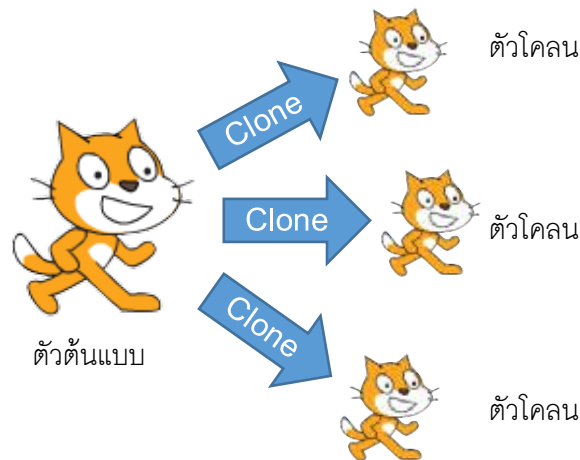
ผลที่ได้ คือ ข้อมูลที่มีอยู่ในแฟ้ม name.txt จะถูกนำเข้ามาเก็บในรายการที่ชื่อว่า "รายชื่อใหม่"

ดังรูป



การโคลนร่างตัวละคร

Scratch ตั้งแต่เวอร์ชัน 2.0 มีรองรับคำสั่งพิเศษที่อนุญาตให้ตัวละครสร้าง “โคลน” ของตัวเองขึ้นมา โดยตัวละครใหม่จะคัดลอกพฤติกรรมของตัวละครต้นแบบมาทั้งหมด รวมถึงตัวแปรที่สร้างขึ้นมาใช้สำหรับตัวละครนั้นๆ และค่าปัจจุบัน



ตัวอย่างที่ 1 กองทัพลูกโป่ง

การสร้างกองทัพลูกโป่ง สามารถทำได้ดังขั้นตอนต่อไปนี้

- 1) เริ่มต้นจากการสร้างโปรเจกต์ใหม่ และสร้างตัวละครที่เป็นรูปลูกโป่งดังภาพ



- 2) สร้างสคริปต์เพื่อกำหนดพฤติกรรมให้ตัวละครต้นแบบเมื่อเริ่มต้นทำงาน โดยให้ต้นแบบซ่อนตัวเองไว้และวางตำแหน่งไว้ที่ขอบด้านล่างของเวที และใช้บล็อก `create clone of myself` โคลนตัวเองขึ้นมาใหม่ 1 ตัวทุกๆ หนึ่งวินาที ดังแสดง

```

when clicked
hide
set y to -240
forever
  create clone of myself
  wait 1 secs


```

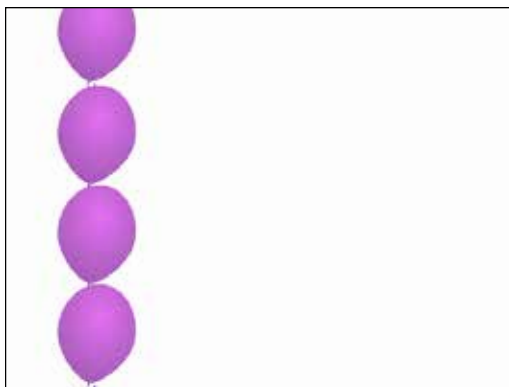
- 3) พฤติกรรมสำหรับตัวโคลนสามารถนำไประบุไว้ภายใต้บล็อก **when I start as a clone** เพื่อให้ตัวโคลนเริ่มทำงานทันทีที่เกิดขึ้นมา สคริปต์ตัวอย่างนี้ทำให้โคลนปรากฏตัวขึ้น และเคลื่อนที่ขึ้นไปด้านบนของเวทีครั้งละ 3 หน่วย จนกระทั่งเมื่อถึงขอบด้านบนของเวทีตัวโคลนจะลบตัวเองทิ้งด้วยบล็อก **delete this clone** ดังนี้

```

when I start as a clone
show
repeat until y position > 200
  change y by 3
delete this clone

```

- 4) คลิก  เพื่อเริ่มต้นทำงาน จะเห็นว่าทุก ๆ หนึ่งวินาทีจะมีลูกโป่งปรากฏขึ้นจากด้านล่างของหน้าต่างแสดงผล และเคลื่อนที่ขึ้นด้านบนจนกระทั่งสุดขอบหน้าต่างจึงหายไป



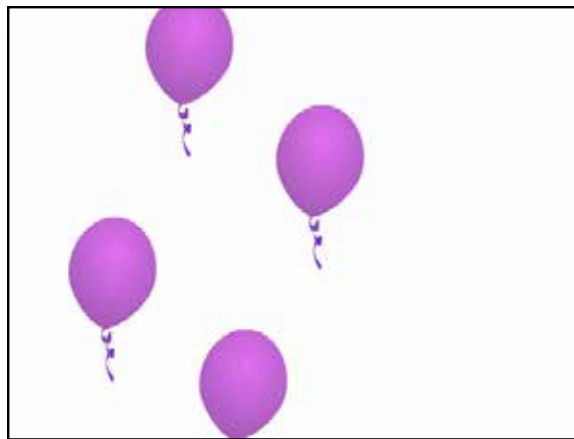
ตัวอย่างที่ 2 กองทัพลูกโป่งหลายสีหลากความเร็ว

การสร้างกองทัพลูกโป่งหลายสีหลากความเร็ว สามารถทำได้ดังขั้นตอนต่อไปนี้

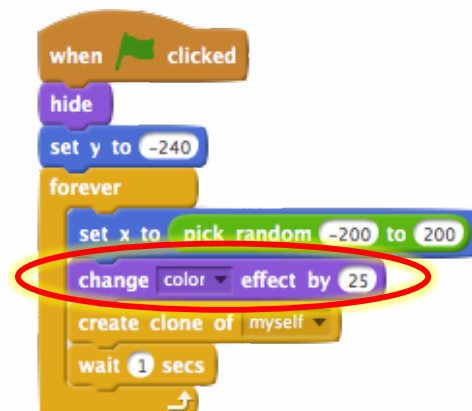
- 1) เพื่อเพิ่มความตื่นตา สคริปต์สามารถเรียกใช้บล็อก pick random มาใช้เพื่อทำให้ลูกโป่งต้นแบบย้ายไปยังตำแหน่งในแกน x แบบสุ่ม เพื่อให้โคลนปรากฏขึ้นในตำแหน่งแกน x ที่แตกต่างกันหลังจากถูกสร้างขึ้นมา ดังแสดง



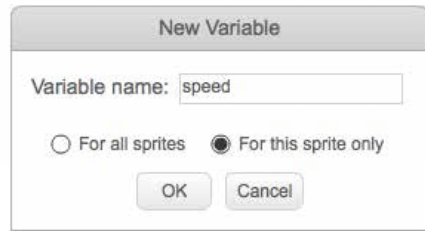
ผลลัพธ์ที่ได้คือ



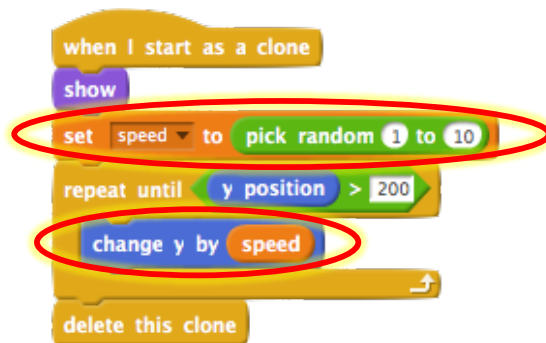
- 2) จากนั้นเพิ่มบล็อก change color เข้าไปเพื่อกำหนดสีใหม่ให้ลูกโป่งต้นแบบก่อนโคลนตัวเอง มีผลทำให้ลูกโป่งโคลนปรากฏขึ้นมาด้วยสีที่แตกต่างกัน



- 3) เพื่อให้ลูกโป่งที่โคลนขึ้นมาลอยขึ้นด้วยความเร็วที่แตกต่างกัน ให้สร้างตัวแปรเฉพาะที่ชื่อ speed ขึ้นมาให้ตัวละครลูกโป่ง ดังนี้



- 4) จากนั้นนำมากำหนดเป็นค่าความเร็วของลูกโป่งแต่ละลูกด้วยบล็อก random และนำค่าที่ได้ไปใช้เป็นระยะทางการลอยขึ้นของลูกโป่งในแต่ละรอบในบล็อก repeat..until ของตัวโคลนดังนี้



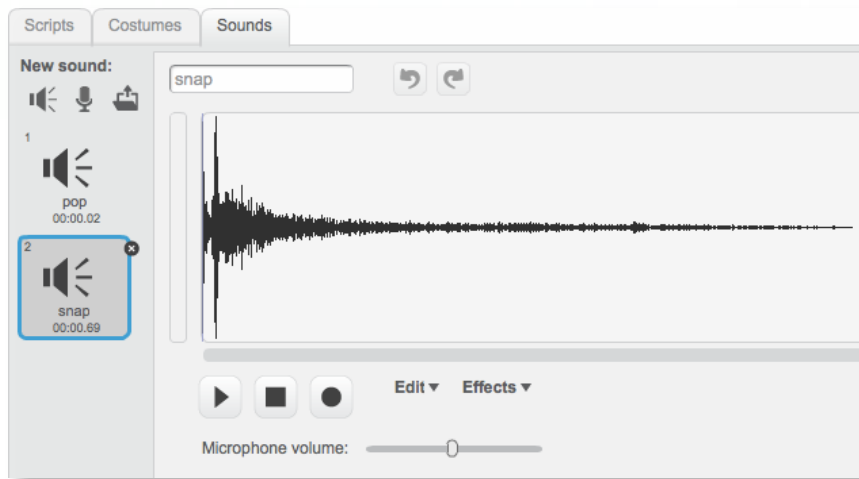
ผลลัพธ์ที่ได้คือ



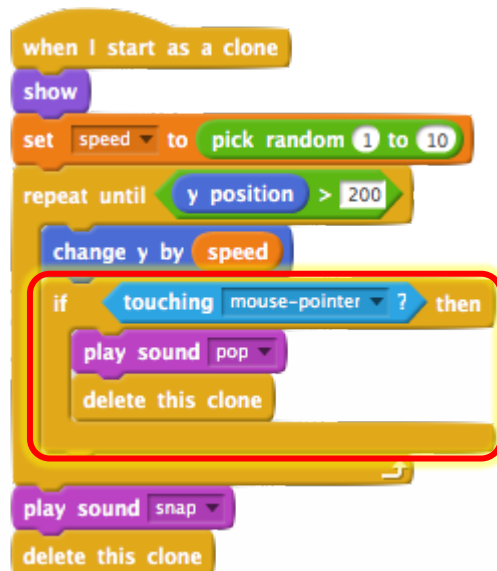
ตัวอย่างที่ 3 เกมเจาะลูกโป่งให้แตกด้วยเมาส์

ดัดแปลงตัวอย่างที่แล้วเพื่อสร้างเป็นเกมที่ใช้เมาส์เจาะลูกโป่งให้แตก สามารถทำได้ดังขั้นตอนต่อไปนี

- 1) เพิ่มคลิปเสียง snap ให้กับตัวละครลูกโป่งดังนี้



- 2) ดัดแปลงสคริปต์สำหรับตัวโคลนเพื่อตรวจสอบว่าตัวโคลนสัมผัสกับตัวชี้เมาส์หรือไม่ หากสัมผัสให้เล่นคลิปเสียง pop และลบตัวเองทิ้ง แต่หากลูกโป่งลอยขึ้นไปถึงขอบหน้าต่างด้านบนแล้วให้เล่นเสียง snap แล้วจึงค่อยลบตัวเอง ดังนี้



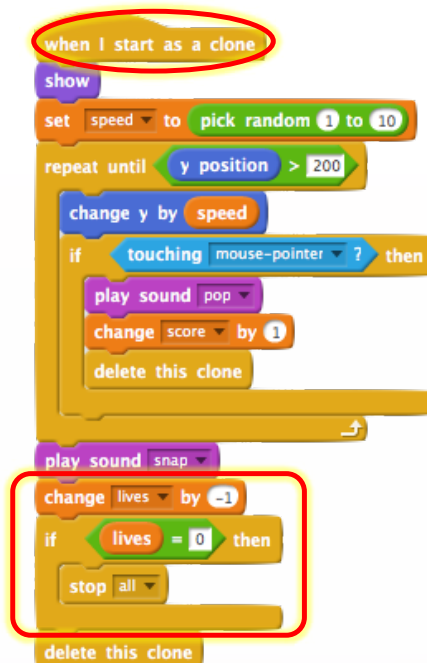
- 3) สร้างตัวแปรเพื่อบันทึกจำนวนลูกโป่งที่แตกและจำนวนชีวิตคงเหลือ โดยตัวแปรทั้งคู่ต้องถูกสร้างให้เรียกใช้ได้จากทุกตัวละคร



- 4) กำหนดค่าเริ่มต้นของตัวแปร lives และ score จากสคริปต์ของต้นแบบ



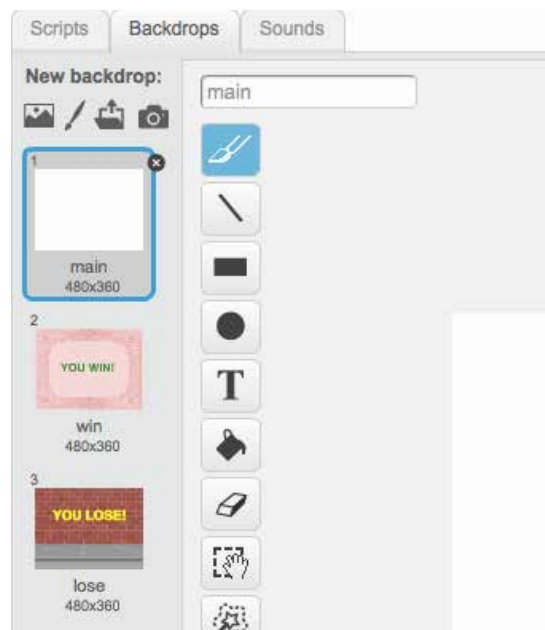
- 5) ให้สคริปต์ของตัวโคลนเพิ่มค่าตัวแปร score เมื่อลูกโป่งสัมผัสตัวซีเมีส และลดค่า lives เมื่อลูกโป่งลอยขึ้นไปสุดด้านบนของจอ หาก lives ลดค่าลงจนเป็นศูนย์ให้จบการทำงานทั้งหมด



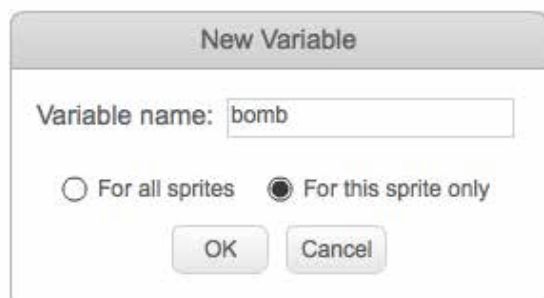
ตัวอย่างที่ 4 หลบลูกโป่งพิษและฉากจบเกม

เกมข้างต้นอาจเล่นง่ายเกินไปด้วยการกวาดเม้าส์ไปมาทั่วทั้งจอ ตัวอย่างนี้เพิ่มความยากและความท้าทายด้วยการสร้างลูกโป่งสองชนิด เป็นลูกโป่งธรรมดา (สีเขียว) และลูกโป่งบรรจุแก๊สพิษ (สีแดง) จำนวนชีวิตจะลดลงหากพลาดทำให้ลูกโป่งสีแดงแตกหรือปล่อยให้ลูกโป่งสีเขียวลอยขึ้นไปจนสุดขอบหน้าต่างแสดงผล และได้คะแนนหากทำให้ลูกโป่งสีเขียวแตกได้ เกมจบลงเมื่อจำนวนชีวิตลดลงเหลือศูนย์และปรากฏเป็นหน้าจอ You Lose หากทำแต้มได้ถึง 10 แต้มจะปรากฏหน้าจอ You Win และจบเกมเช่นกัน สามารถทำได้ดังขั้นตอนต่อไปนี้

- 1) สร้างพื้นหลังชื่อ win และ lose เพิ่มให้กับเวที (ตกแต่งฉากตามต้องการ) พร้อมทั้งเปลี่ยนชื่อพื้นหลังเดิมให้เป็นชื่อ main ดังนี้



- 2) สร้างตัวแปรเฉพาะที่ชื่อ bomb ขึ้นมาเพื่อใช้บ่งบอกว่าลูกโป่งที่โคลนขึ้นเป็นลูกโป่งพิษ (bomb = 1) หรือลูกโป่งธรรมดา (bomb = 0)



- 3) แก้ไขสคริปต์ของตัวต้นแบบในตัวอย่างที่ผ่านมาเพื่อกำหนดพื้นหลังให้เริ่มต้นที่ main และใช้บล็อก pick random ในการสุ่มค่าให้กับตัวแปร bomb ก่อนโคลนตัวเอง ดังนี้

```

when clicked
hide
switch backdrop to main
set y to -240
set score to 0
set lives to 5
forever
set x to pick random -200 to 200
set bomb to pick random 0 to 1
create clone of myself
wait 1 secs

```

- 4) ค่าของตัวแปร bomb ที่สุ่มมาได้ให้นำมาใช้ในตอนต้นของสคริปต์สำหรับตัวโคลน โดยหาก bomb = 1 ให้กำหนดสีเป็นหมายเลข 40 (สีแดง) ไม่เช่นนั้นให้เป็นหมายเลข 100 (สีเขียว)

```

when I start as a clone
show
if bomb = 1 then
set color effect to 40
else
set color effect to 100
set speed to pick random 1 to 10
repeat until y position > 200
change y by speed
if touching mouse-pointer then
play sound pop
change score by 1
delete this clone
play sound snap
change lives by -1
if lives = 0 then
stop all

```

- 5) เพื่อความสะดวกและลดความซ้ำซ้อน ให้สร้างฟังก์ชัน receive point และ lose live เพื่อใช้เพิ่มคะแนน และลดจำนวนชีวิตตามลำดับ โดยเมื่อได้รับคะแนน 10 คะแนนจะสลับพื้นหลังไปเป็นฉาก win และจบเกม ในขณะที่หากจำนวนชีวิตลดลงเหลือ 0 จะสลับพื้นหลังไปเป็นฉาก lose และจบเกม

```

define receive point
  play sound pop
  change score by 1
  if score = 10 then
    switch backdrop to win
  stop all

define lose live
  play sound snap
  change lives by -1
  if lives = 0 then
    switch backdrop to lose
  stop all
  
```

- 6) เรียกใช้ฟังก์ชันทั้งคู่ในสคริปต์ของตัวโคลน สังเกตว่าการวนซ้ำมีการเปลี่ยนจากบล็อก repeat.. until เป็น forever นอกจากนี้ยังมีการเพิ่มบล็อก wait 0.01 sec ลงไปในตอนท้ายเพื่อควบคุมความเร็วการลอยของลูกโป่งในภาพรวม

```

when I start as a clone
  show
  if bomb = 1 then
    set color effect to 40
  else
    set color effect to 100
  set speed to pick random 1 to 10

forever
  change y by speed
  if y position > 200 then
    if bomb = 0 then
      lose live
    delete this clone
  if touching mouse-pointer ? then
    if bomb = 1 then
      lose live
    else
      receive point
    delete this clone
  wait 0.01 secs
  
```

การโต้ตอบกับผู้ใช้ผ่านเว็บแคม

สำหรับเครื่องคอมพิวเตอร์ที่มีเว็บแคมติดตั้งไว้ Scratch เวอร์ชัน 2.0 สามารถดึงภาพจากกล้องมาแสดงผลเป็นพื้นหลังของฉากได้ นอกจากนี้ยังมีบล็อกคำสั่งที่รองรับการประมวลผลภาพที่นำมาสร้างเป็นสคริปต์โต้ตอบกับผู้ใช้ได้

บล็อกควบคุมวิดีโอ

Scratch 2.0 มีบล็อกสำหรับโต้ตอบกับวิดีโอหลัก 3 บล็อก อยู่ในกลุ่มบล็อก Sensing ได้แก่

- **turn video on** ใช้เปิดและปิดการทำงานของกล้อง โดยเมื่อสั่งเปิดภาพจากกล้องจะถูกนำมาแสดงเป็นพื้นหลังของฉาก
- **set video transparency to 50 %** ปรับความโปร่งใสของวิดีโอในระดับ 0% (มองเห็นภาพวิดีโอชัดเต็มที่) จนถึง 100% (มองไม่เห็นภาพวิดีโอ)



วิดีโอโปร่งใส 10%



วิดีโอโปร่งใส 50%



วิดีโอโปร่งใส 90%

- **video motion on this sprite** ให้ค่าปริมาณการเปลี่ยนแปลงของวิดีโอ ณ บริเวณตัวละคร ค่าที่ได้จะอยู่ระหว่าง 0 (ไม่มีการเปลี่ยนแปลง) ถึง 100 (มีการเปลี่ยนแปลงสูงสุด)

ตัวอย่างที่ 1 ตีลูกโป่งให้แตก

ดัดแปลงตัวอย่างสุดท้ายจากภาคผนวก การโคลนร่างตัวละคร โดยเพิ่มบล็อกลงไปสองบล็อกในสคริปต์ของตัวละครต้นแบบ ซึ่งจะเป็นการนำภาพจากกล้องมาแสดงผลเป็นพื้นหลังของเวที และหน่วงเวลารอบประมาณ 3 วินาที เพื่อให้กล้องปรับสภาพแสง ดังนี้



```
when green flag clicked
hide
switch backdrop to main
turn video on
wait 3 secs
set y to -240
set score to 0
set lives to 5
forever
  set x to pick random -200 to 200
  set bomb to pick random 0 to 1
  create clone of myself
  wait 2 secs
```

ในส่วนของสคริปต์ตัวโคลน ให้เปลี่ยนเงื่อนไขจากการตรวจสอบตำแหน่งเมาส์ไปเป็นการตรวจสอบการเคลื่อนไหวของวิดีโอภายใต้ตัวละครดังนี้



```
when I start as a clone
show
if bomb = 1 then
  set color effect to 40
else
  set color effect to 100
set speed to pick random 1 to 10
forever
  change y by speed
  if y position > 200 then
    if bomb = 0 then
      lose live
    delete this clone
  if video motion on this sprite > 30 then
    if bomb = 1 then
      lose live
    else
      receive point
    delete this clone
```

สุดท้ายคือสั่งให้ปิดกล้องวิดีโอเมื่อจบเกมในฟังก์ชัน receive point และ lose live ก่อนที่จะสั่งหยุดการทำงานทั้งหมดด้วย stop all

```
define lose live
  play sound snap
  change lives by -1
  if lives = 0 then
    switch backdrop to lose
    turn video off
  stop all

define receive point
  play sound pop
  change score by 1
  if score = 10 then
    switch backdrop to win
    turn video off
  stop all
```

เมื่อเริ่มเล่นเกมจะได้ผลลัพธ์ดังตัวอย่าง ทำให้วิดีโอมีการเปลี่ยนแปลงบริเวณลูกโป่งสีเขียว เช่น สบัดมือ ปรบมือ หรือสายหัว เพื่อให้ได้แต้ม แต่หลีกเลี่ยงการขยับบริเวณลูกโป่งสีแดงเพื่อไม่ให้เสียจำนวนชีวิต และควรเล่นในบริเวณที่ไม่มีผู้อื่น เพื่อเลี่ยงการเคลื่อนไหวที่ควบคุมไม่ได้

ภาคผนวก



คณะผู้พัฒนาเอกสารการจัดการเรียนรู้สะเต็มศึกษา ด้วยโปรแกรม Scratch ครั้งที่ 1

1. นายชัยพร ใจแก้ว	มหาวิทยาลัยเกษตรศาสตร์	กรุงเทพฯ
2. นายวรปรัชญ์ ลาวันย์วิไลวงศ์	โรงเรียนนราสิกขาลัย	จ.นราธิวาส
3. นายเฉลิมพล มีดวง	โรงเรียนกระทุ่มวิทยา	จ.ภูเก็ต
4. นางสาวสิริชชา วงศ์คม	โรงเรียนสวนบุญโญปถัมภ์	จ.ลำพูน
5. นายอรธนนพ แต่งอ่อน	โรงเรียนบ้านคลองไทร	จ.นครสวรรค์
6. นางณัฐกานต์ ฉิมงาม	โรงเรียนบ้านคลองไทร	จ.นครสวรรค์
7. นางสุปราณี ศรีตระกุลวงศ์	โรงเรียนป่าซาง	จ.ลำพูน
8. นายวิฑูรย์ สุธรรม	โรงเรียนป่าซาง	จ.ลำพูน
9. นายอลงกต หาญชนะ	โรงเรียนราชสีมาวิทยาลัย	จ.นครราชสีมา
10. นางสาวกิตติมา มุ่งวัฒนา	โรงเรียนสภาราชนิ 2	จ.ตรัง
11. นางสาวศรีอำพร ชูรี	โรงเรียนวัดราชโกษา	กรุงเทพฯ
12. นางสาวกมลนารี ลายคราม	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
13. นายพรพจน์ พุดวันเพ็ญ	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
14. นายนิพนธ์ ศุภศรี	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
15. นางสาวจินดาพร หมวกหมื่นไวย	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
16. นางสาวทัศนีย์ กรองทอง	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
17. นายนิรมิษ เพียรประเสริฐ	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
18. นายพนมยงค์ แก้วประชุม	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
19. นางสาวชिरพรรณ ทองวิจิตร	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	

คณะผู้พัฒนาเอกสารการจัดการเรียนรู้สะเต็มศึกษาด้วยโปรแกรม Scratch ครั้งที่ 2

1. นายขวลิต ศรีสถาพรพัฒน์	มหาวิทยาลัยเกษตรศาสตร์	กรุงเทพฯ
2. นายชัยพร ใจแก้ว	มหาวิทยาลัยเกษตรศาสตร์	กรุงเทพฯ
3. นายวรปรัชญ์ ลาวันย์วิไลวงศ์	โรงเรียนนราสิกขาลัย	จ.นราธิวาส
4. นายเฉลิมพล มีดวง	โรงเรียนกระทุ่มวิทยา	จ.ภูเก็ต
5. นางสาวสิริชชา มีดวง	โรงเรียนสวนบุญโญปถัมภ์	จ.ลำพูน
6. นายอรธนนพ แต่งอ่อน	โรงเรียนบ้านคลองไทร	จ.นครสวรรค์
7. นางสาวไศภิชฎ์ สวนปลีก	โรงเรียนกำแพงเพชรพิทยาคม	จ. กำแพงเพชร
8. นายอลงกต หาญชนะ	โรงเรียนราชสีมาวิทยาลัย	จ.นครราชสีมา
9. นางสาวชिरวารรณ เทียมทัน	โรงเรียนเมืองเลย	จ.เลย
10. นางสาวศวดี เขตต์เมืองมูล	โรงเรียนห้วยข้าวลิบ	จ.เชียงใหม่
11. นายประภาส สาระศาลิน	โรงเรียนสมเด็จพระปิยมหาราชรมณีย์	จ.กาญจนบุรี
12. นายสุธนัย สุธาประดิษฐ์	โรงเรียนศรีทธาสุมท จ.สมุทรสงคราม	จ.สมุทรสงคราม
13. นายเรวัต ะบุรงค์	โรงเรียนหนองเสือวิทยาคม	จ.ปทุมธานี
14. นางสาวทัศนีย์ กรองทอง	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
15. นางสาวพรพิมล ตั้งชัยสิน	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	
16. นายพนมยงค์ แก้วประชุม	สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี	