

21st Century

DIGITAL KIDS

หนังสือแบบเรียนวิชาเทคโนโลยี หลักสูตรแกนกลางปี 2560



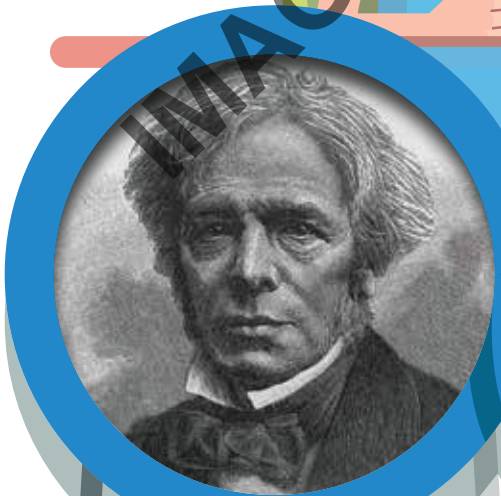
No.4

- ⚙️ ตรงตามมาตรฐานการเรียนรู้ 2560
- ⚙️ Problem Solving , Computational Thinking
- ⚙️ Coding for kids : Scratch 2.0 , Basic Robotics
- ⚙️ Windows 10 , Paint 3D
- ⚙️ Digital Citizenship



หน่วยที่ ๒ การออกแบบโดยใช้ storyboard หรือการออกแบบอัลกอริทึม

วิทยาการคอมพิวเตอร์(Computer Science) การแก้ปัญหาอย่างเป็นขั้นตอน และเป็นระบบการใช้แนวคิดเชิงคำนวณในการแก้ปัญหาในชีวิตประจำวัน บูรณาการกับวิชาอื่น การเขียนโปรแกรมการคาดการณ์ผลลัพธ์ การตรวจหาข้อผิดพลาดการพัฒนาแอปพลิเคชันหรือพัฒนาโครงงานอย่างสร้างสรรค์เพื่อแก้ปัญหาในชีวิตจริง

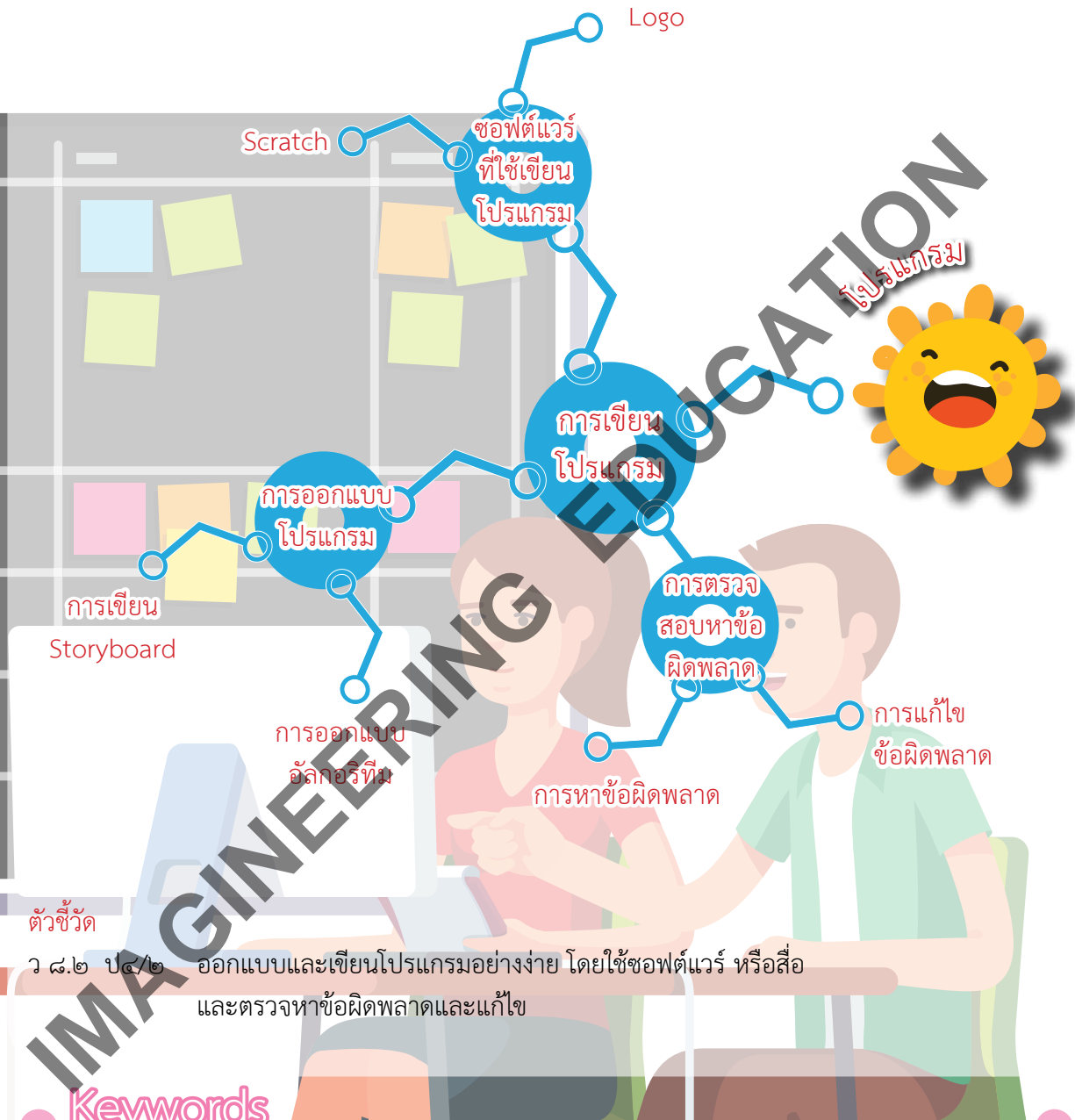


“The greatness of a nation and its moral progress can be judged by the way its animal are treated.”

... ความยิ่งใหญ่ของมนุษยชาติและความก้าวหน้าทาง
จริยธรรมสามารถตัดสินได้จากวิธีการปฏิบัติต่อสัตว์
ของพวกเขา...

– ไมเคิล ฟาราเดย์ (Michael Faraday)–

MindMap แผนผังสาระการเรียนรู้



ตัวชี้วัด

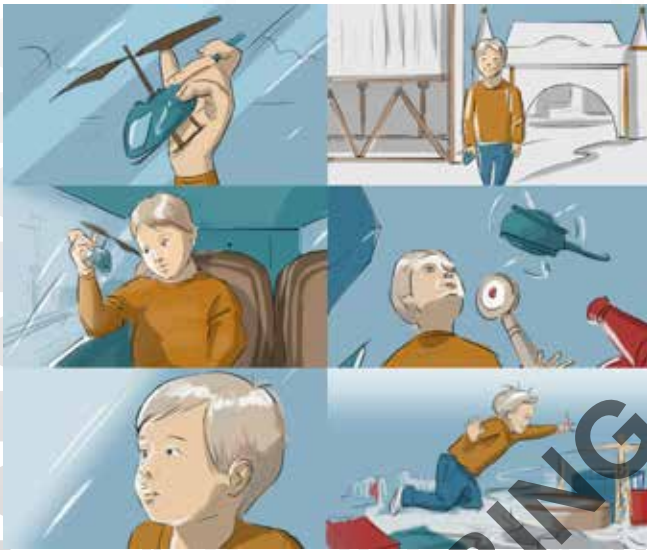
ว ๘.๒ ป๘/๒ ออกแบบและเขียนโปรแกรมอย่างง่าย โดยใช้ซอฟต์แวร์ หรือสื่อ และตรวจหาข้อผิดพลาดและแก้ไข

Keywords

- | | |
|--|---|
| 1. programming = การเขียนโปรแกรม | 5. scratch = โปรแกรมฝึกเขียนโค้ด |
| 2. coding = การเขียนโค้ด | 6. storyboard = การเขียนภาพลงในกรอบแสดงเรื่องราวเป็นลำดับขั้น |
| 3. computer Language = ภาษาในการเขียนโปรแกรม | 7. algorithm = การแสดงลำดับขั้นตอน |
| 4. programmer = นักเขียนโปรแกรม | 8. debugging = การแก้ไขข้อผิดพลาด |

การออกแบบโปรแกรมอย่างง่าย โดยใช้ Story Board

สตอรี่บอร์ด(story board) คือแสดงลำดับขั้นตอนการทำงานของโปรแกรมทีละลำดับขั้น โดยเขียนเป็นภาพ ข้อความ และภาพเคลื่อนไหว ซึ่งการเขียน storyboard มี 2 ส่วนดังนี้



Scene duration:

① ส่วนของภาพโดยจะวาดอยู่ใน กรอบสี่เหลี่ยม

② ส่วนของการบรรยายเป็น ข้อความหรือบทสนทนา

การเขียน Storyboard มีข้อดีอย่างไรบ้างครับ

ทำให้เราได้เห็น
เนื้อเรื่องที่จะสร้างตั้งแต่ต้นจนจบ
ทำให้ได้เนื้อเรื่องเหมาะสม
และหากมีข้อผิดพลาดจะได้
ทราบก่อนลงมือทำไงละ

ตัวอย่าง ให้ออกแบบ story board นิทานเรื่อง ราชสีห์กับหนู

ราชสีห์ตัวหนึ่งนอนหลับอยู่ในป่า มีเจ้าหนูมาวิ่งไต่ตามลำตัวราชสีห์โกรธมากแต่เห็นว่าเจ้าหนูเป็นสัตว์ตัวเล็กจึงไล่ให้ไปไกลๆเมื่อครั้งราชสีห์ออกไปล่าสัตว์ได้ติดกับบ่วงนายพรานออกไม่ได้ เจ้าหนูสำนึกในบุญคุณจึงกลับมาช่วยราชสีห์ให้รอดพ้นกับบ่วงนายพราน นิทานเรื่องนี้สอนให้รู้ว่าสัตว์ตัวเล็กก็มีประโยชน์ไม่แพ้สัตว์ใหญ่



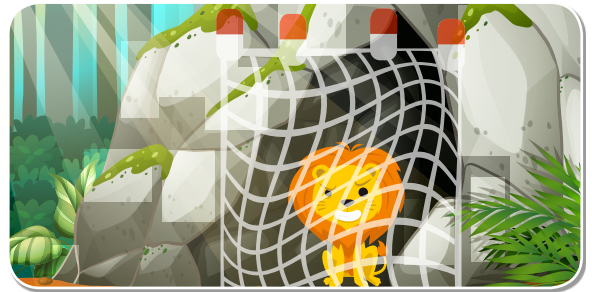
- 1 ราชสีห์นอนหลับอยู่ในป่า เจ้าหนูมาเดินรบกวนอยู่ใกล้ๆ



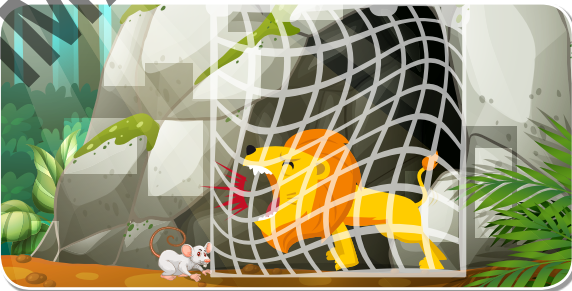
- 2 ราชสีห์โกรธและพูดว่าเห็นว่าเจ้าเป็น สัตว์ตัวเล็กๆ จึงปล่อยเจ้าไป



- 3 เมื่อราชสีห์ออกไปล่าสัตว์



- 4 กลับมาจึงติดกับบ่วงของ นายพรานราชสีห์จึงออกไม่ได้



- 5 เจ้าหนูสำนึกในบุญคุณจึงได้มาช่วย กัดบ่วงนายพรานราชสีห์จึงออกมาได้









- 6 นิทานเรื่องนี้สอนให้รู้ว่าสัตว์ตัวน้อย ก็มีประโยชน์ไม่แพ้สัตว์ใหญ่เช่นกัน

การออกแบบโปรแกรมอย่างง่าย โดยใช้อัลกอริทึม (Algorithm)

ก่อนจะลงมือเขียนโปรแกรม ต้องออกแบบโปรแกรมโดยการเขียนขั้นตอนการทำงานหรืออัลกอริทึมก่อน เพื่อจะได้เห็นภาพรวมการทำงานของโปรแกรม และหากเกิดข้อผิดพลาดจะสามารถตรวจสอบและแก้ไขได้ก่อนจะลงมือเขียนโปรแกรม

การออกแบบอัลกอริทึมนั้นสามารถทำได้โดยใช้การเขียนสัญลักษณ์ที่เรียกว่า **Flowchart** โดยมีรูปแบบของสัญลักษณ์ดังต่อไปนี้

สัญลักษณ์	ภาษาอังกฤษ	ความหมายของสัญลักษณ์
	Terminator	จุดเริ่มต้น/สิ้นสุด
	Input Output	ข้อมูล นำเข้า/ส่งออก
	Process	การประมวลผลข้อมูล
	Decision	การตัดสินใจ เพื่อพิจารณาเงื่อนไขและกำหนดทิศทางการทำงานต่อไป
	Direction Flow	ทิศทางการทำงาน ใช้เชื่อมต่อสัญลักษณ์ต่างๆ เพื่อแสดงทิศทางของโปรแกรม
	Connector	จุดเชื่อมต่อ ใช้รวมเส้นการทำงานให้ออกไปเหลือเพียงเส้นเดียว

ตัวอย่างการออกแบบโปรแกรมอย่างง่าย โดยใช้ Flowchart

วันนี้ลิปดาตื่นสาย จึงคิดเลือกวิธีการเดินทางไปโรงเรียนใหม่ อาจจะต้องใช้ รถยนต์หรือ รถมอเตอร์ไซด์ ซึ่งปกติลิปดาเดินทางด้วยรถประจำทาง ลิปดา ต้องประมวลผล เปรียบเทียบว่าระยะเวลาการเดินทางใดจะสามารถเดินทาง ได้เร็วที่สุด



แม่แล้วเราจะตัดสินใจ
อย่างไรดีนะเนี่ย



ออกแบบการแก้ไข
ปัญหาได้เยี่ยมมาก
ลิปดา



ลิปดาตื่นตั้งแต่ 6 โมงเช้า อาบน้ำ แปรงฟัน ทานข้าว แต่งตัวและออกเดินทางไปโรงเรียน เรียนหนังสือ และทำกิจกรรมต่างๆ จนเข้านอน ทุกคนก็มีกิจวัตรประจำวันเหมือนกันเรามาสร้าง storyboard หรือ ออกแบบอัลกอริทึมกิจวัตรประจำวันตนเองกัน



Missions :

1. จากโจทย์ให้ออกแบบโปรแกรมโดยใช้ Story Board หรือออกแบบอัลกอริทึมกิจวัตรประจำวันของตนเอง
2. ให้แสดง storyboard หรือ อัลกอริทึมในกระดานเทปขาว พร้อมนำเสนอหน้าชั้นเรียน

Materials : 1. กระดานเทปขาว 2. อุปกรณ์เครื่องเขียนอื่นๆ

Ask : การออกแบบโปรแกรมโดยใช้ Story Board หรืออัลกอริทึมในการแก้ไขปัญหา คืออะไร

Plan & Design : แบ่งหน้าที่การทำงานวางแผนออกแบบโปรแกรมโดยใช้ Story Board หรืออัลกอริทึมในการแก้ไขปัญหา

ให้นักเรียนเขียนลำดับการทำงานของกิจกรรมนี้



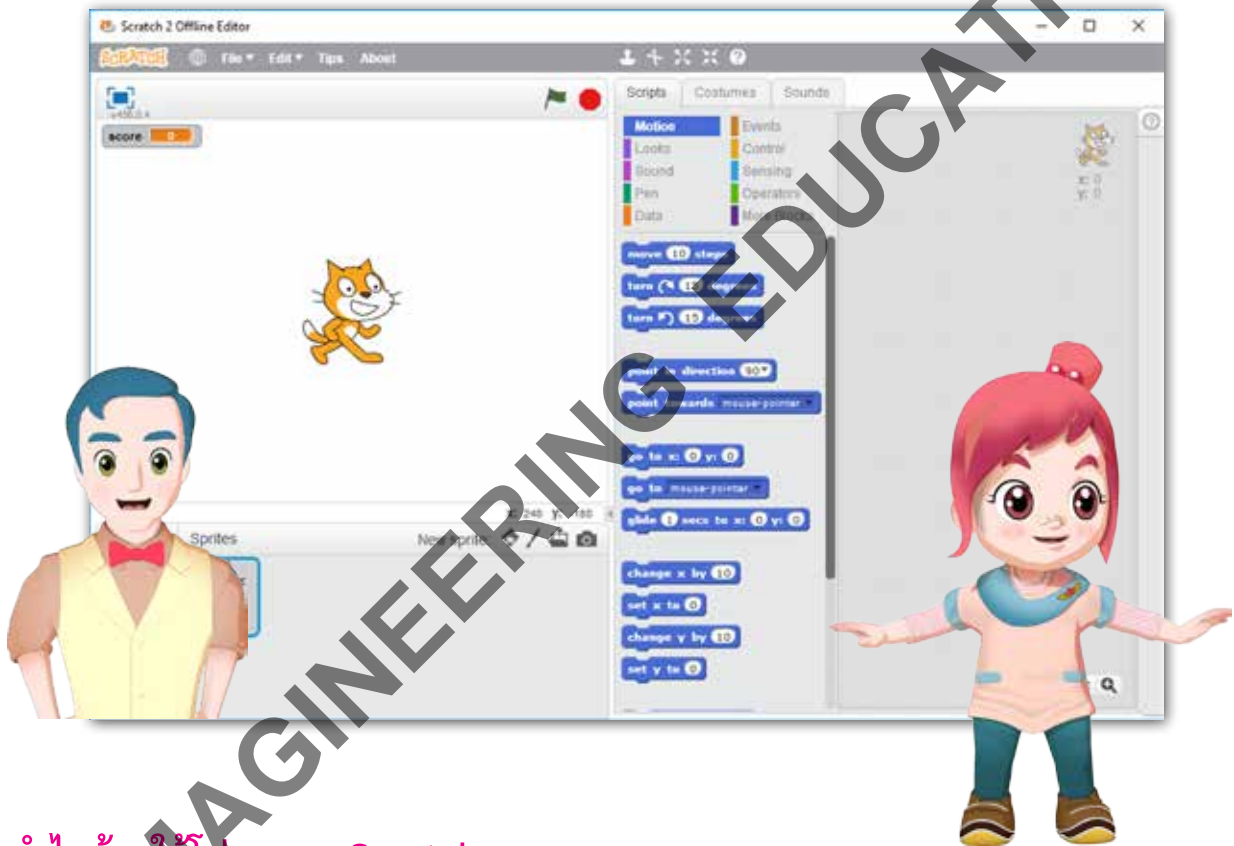
IMAGINEERING EDUCATION



- การออกแบบโปรแกรมก่อนจะลงมือเขียนโปรแกรมมีประโยชน์อย่างไร
- การออกแบบโปรแกรมโดยใช้ Story Board คืออะไร
- การออกแบบโปรแกรมโดยใช้ อัลกอริทึม คืออะไร

เด็กๆ จะได้อะไรจากการเขียนโปรแกรม Scratch

การฝึกเขียนโปรแกรมเป็นเครื่องมือที่ส่งเสริมทักษะกระบวนการคิดเชิงคำนวณในการแก้ปัญหาการคิดอย่างเป็นระบบ(Computational Thinking) และการคิดสร้างสรรค์ (Creative Thinking) ที่ช่วยให้ผู้เรียนพัฒนาทักษะการคิด ออกมาในรูปแบบการเขียนโปรแกรมสร้างโปรเจกต์หรือสร้างชิ้นงาน และนำเสนอ



ทำไมต้องใช้โปรแกรม Scratch

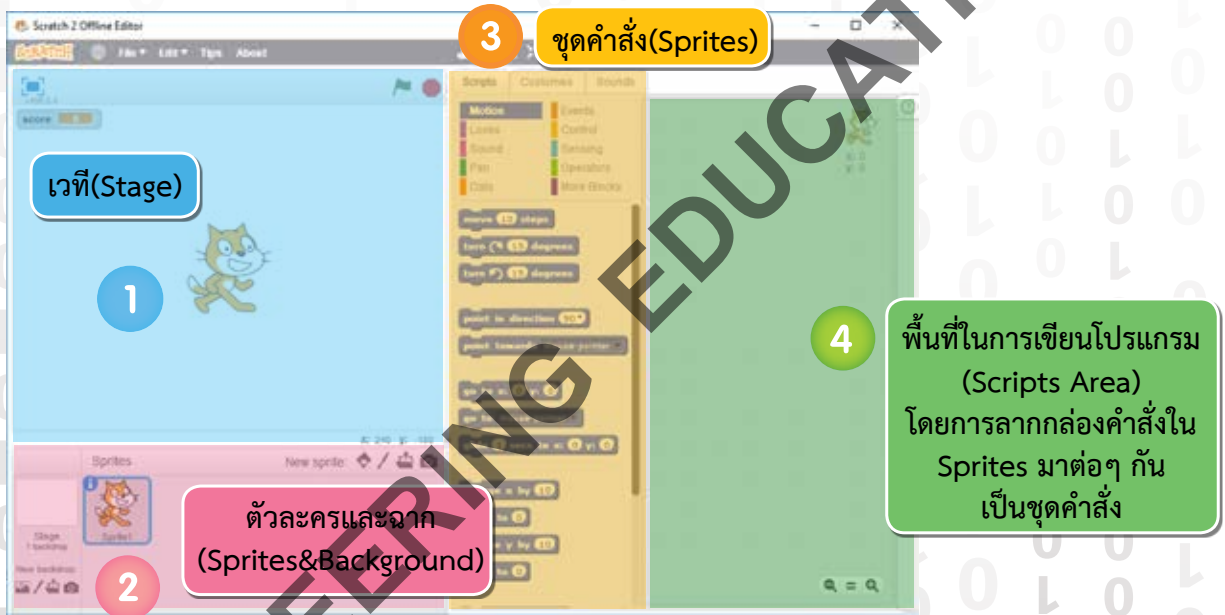
โปรแกรม Scratch เป็นทั้งเครื่องมือและโปรแกรมภาษา พัฒนาโดยทีมงาน Media Lab MIT มีกลุ่มเป้าหมายเป็นเด็กๆ หรือผู้เริ่มต้นหัดเขียนโปรแกรม วัตถุประสงค์ของทีมพัฒนา คือการส่งเสริมผู้เรียนให้ใช้ความคิดสร้างสรรค์ ผ่านการใช้สื่อต่างๆ ไม่ว่าจะเป็นข้อความ ภาพ เสียง หรือ ภาพเคลื่อนไหว ด้วยเครื่องมือที่ไม่ซับซ้อน และมาพร้อมกับตัวโปรแกรม Scratch เพื่อสร้างโปรเจกต์ต่างๆ ตามจินตนาการของผู้เรียน อาทิเช่น การเล่าเรื่อง การสร้างแอนิเมชั่น การจำลอง แม้กระทั่งการเขียนเกมก็ทำได้

รู้จักโปรแกรม Scratch



เป็นโปรแกรมภาษาคอมพิวเตอร์สำหรับผู้เริ่มต้นหัดเขียนโปรแกรม โดยใช้สร้างสื่ออย่างง่าย เช่น ภาพ เสียง ข้อความ ดนตรี เพื่อสร้างเรื่องราวภาพเคลื่อนไหว นิทานที่มีการตอบโต้กับผู้ใช้ เล่ากวี จวัตรประจำวัน การตูนสั้น สร้างเกม เพื่อเป็นการฝึกทักษะการคิดวิเคราะห์ และความคิดสร้างสรรค์

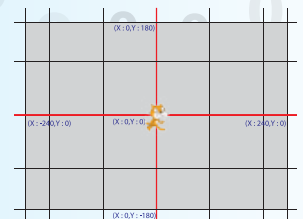
หน้าต่างของโปรแกรม



1. **เวที (Stage)** ซึ่งมีขนาดกว้าง 480 หน่วย สูง 360 หน่วย ใช้แสดงผลของชุดคำสั่งการทำงานของโปรแกรม

2. **ตัวละคร และ ฉาก (Sprites & Background)**

จะมีข้อมูลของ ภาพ เสียง ซึ่งสามารถคลิกเข้าไปดูในบริเวณ **ชุดคำสั่ง**₃ ซึ่งมีแถบ



3. **ชุดคำสั่ง (Scripts)** เป็นชุดคำสั่งควบคุมตัวละครและฉาก เช่น การเคลื่อนไหว (Motion), หน้าตาภาพ (Looks), เสียง (Sound), ปากกา (Pen), เหตุการณ์ (Events) เป็นต้น

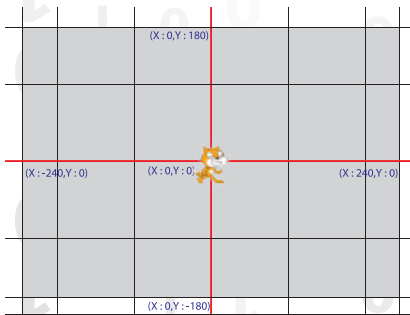


การเขียนโปรแกรมย้ายตำแหน่ง



เป็นการย้ายตำแหน่งของตัวละคร (Sprite) โดยมีคำสั่งที่เกี่ยวข้อง เช่น

1. **move 10 steps** คือ คำสั่งให้เคลื่อนที่ไปในแนวแกน X ก้าว
2. **go to x: 0 y: 0** คือ คำสั่งย้ายตัวละครไปยังตำแหน่ง
แกน X , แกน Y เท่าไร



เวทีมีขนาดกว้าง 480 หน่วย
สูง 360 หน่วย จุดกึ่งกลาง
จะเป็นตำแหน่ง X=0, Y=0



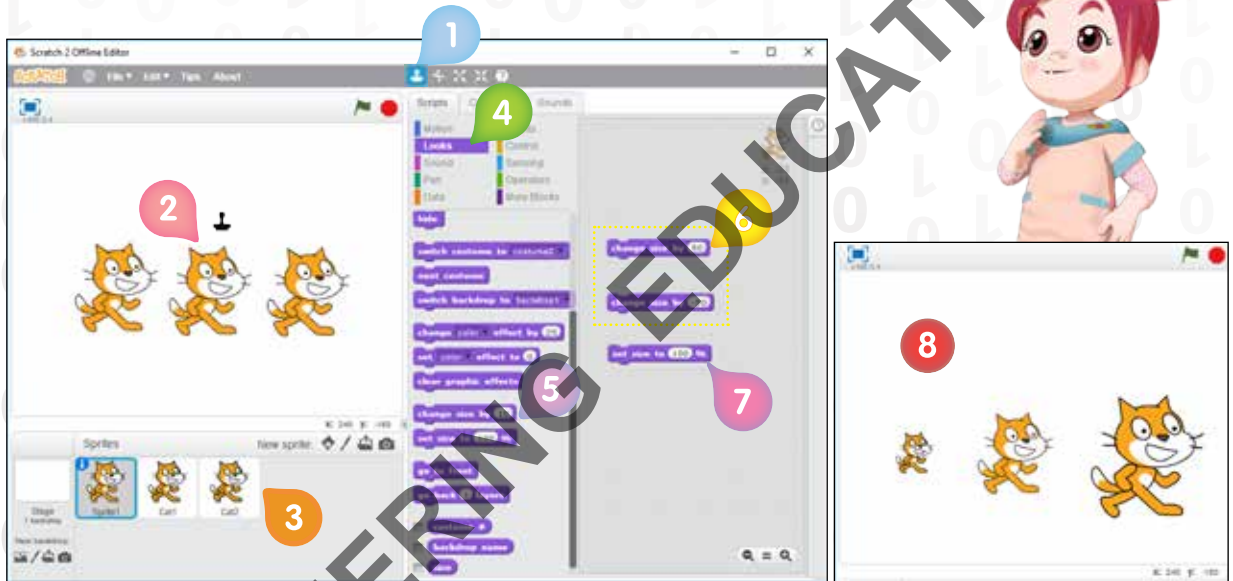
1. **คลิกเลือกตัวละคร (Sprites) ที่ต้องการ**
ก่อนจะทำให้ตัวละครใดเคลื่อนที่ที่ต้องสังเกต
และเลือกตัวละครที่ต้องการให้ถูกต้อง
2. **สังเกตตำแหน่งของตัวละคร**
จากตัวอย่าง ตัวละครแมว
อยู่ ณ ตำแหน่ง X=0, Y=0
3. **เลือกชุดคำสั่ง** Scripts Motion
ซึ่งเป็นชุดคำสั่งควบคุมการเคลื่อนที่ของตัวละคร
มีลักษณะเป็น กล่องคำสั่ง ต่อกันแบบตัวต่อ
4. **คลิกเลือกลากคำสั่ง** go to x: 0 y: 0
แล้วลากไปยังพื้นที่สั่งงานเขียนโปรแกรม
(Scripts Area) ลากคำสั่งออกมา 4 คำสั่งตามรูป
5. **พิมพ์ตำแหน่งที่ต้องการ**
ให้ทดลองเปลี่ยนตำแหน่งตามรูป
6. **เมื่อแก้ไขตำแหน่งแล้วให้คลิกคำสั่งนั้น**
เป็นการคลิกคำสั่งนั้นให้ทำงาน และสังเกตการ
เปลี่ยนแปลงบริเวณเวที (Stage)
7. **คลิกเลือกและลากคำสั่ง** move 10 steps
ซึ่งเป็นคำสั่งย้ายตำแหน่งไปข้างหน้าก็ก้าว
8. **ทดลองเปลี่ยนระยะตามที่ต้องการ**
เป็นการเขียนโปรแกรมสั่งให้ ตัวละคร
ย้ายตำแหน่งตามที่ต้องการ

การเขียนโปรแกรมย่อและขยายขนาด




เป็นการเปลี่ยนขนาดของตัวละคร โดยมีคำสั่งที่เกี่ยวข้อง เช่น

1. **change size by 50** **change size by -50** คือการเปลี่ยนขนาดเพิ่มขึ้นหรือลดลง
2. **set size to 100 %** คือการกำหนดขนาดเพิ่มขึ้น-ลดลงตามเปอร์เซ็นต์






1 **คลิกคำสั่งคัดลอกตัวละคร (Stamp)** 
ซึ่งเป็นคำสั่งคัดลอกตัวละคร โดยในตัวอย่าง
ต้องการคัดลอกเพื่อให้เห็นการเปลี่ยนแปลง


2 **ให้คลิกตัวละครที่ต้องการคัดลอก**
โดยเมาส์พอยเตอร์จะเปลี่ยนเป็น 
แต่ตัวละครที่คัดลอกจะซ้อนกันต้องลากออกมา

3 **สังเกตตัวละครและเลือกตัวละครที่ต้องการ**
จากตัวอย่างจะมีตัวละคร 3 ตัว ที่คัดลอกไว้ซึ่ง
ตัวละคร 1 ตัวจะมีคำสั่งแต่ละตัวไม่เหมือนกัน

4 **คลิกเลือกลากคำสั่ง**  
ซึ่งเป็นชุดคำสั่งเกี่ยวกับหน้าตารูปร่าง

5 **ลากชุดคำสั่ง**  
ไปยัง **พื้นที่สั่งงานเขียนโปรแกรม**
(Scripts area)

6  **คำสั่งเปลี่ยนขนาด**
เป็นคำสั่งเปลี่ยนขนาดโดยสามารถ
ใส่ขนาดเพิ่ม (+10) และ ลด (-10)

7  **คำสั่งกำหนดขนาด**
เป็นคำสั่งกำหนดขนาดคิดเป็นเปอร์เซ็นต์ของภาพ

8 **เมื่อเปลี่ยนขนาดตามต้องการแล้ว**
จะเห็นการเปลี่ยนแปลงในบริเวณเวทีดังภาพ

การเขียนโปรแกรมเปลี่ยนรูปร่าง

การเปลี่ยนรูปร่างนิยมใช้เมื่อ ตัวละครที่เคลื่อนที่หรือพบกับเหตุการณ์ แล้วเปลี่ยนลักษณะท่าทางการแสดงของตัวละคร โดยมีตัวอย่างคำสั่งเช่น

switch costume to costume1

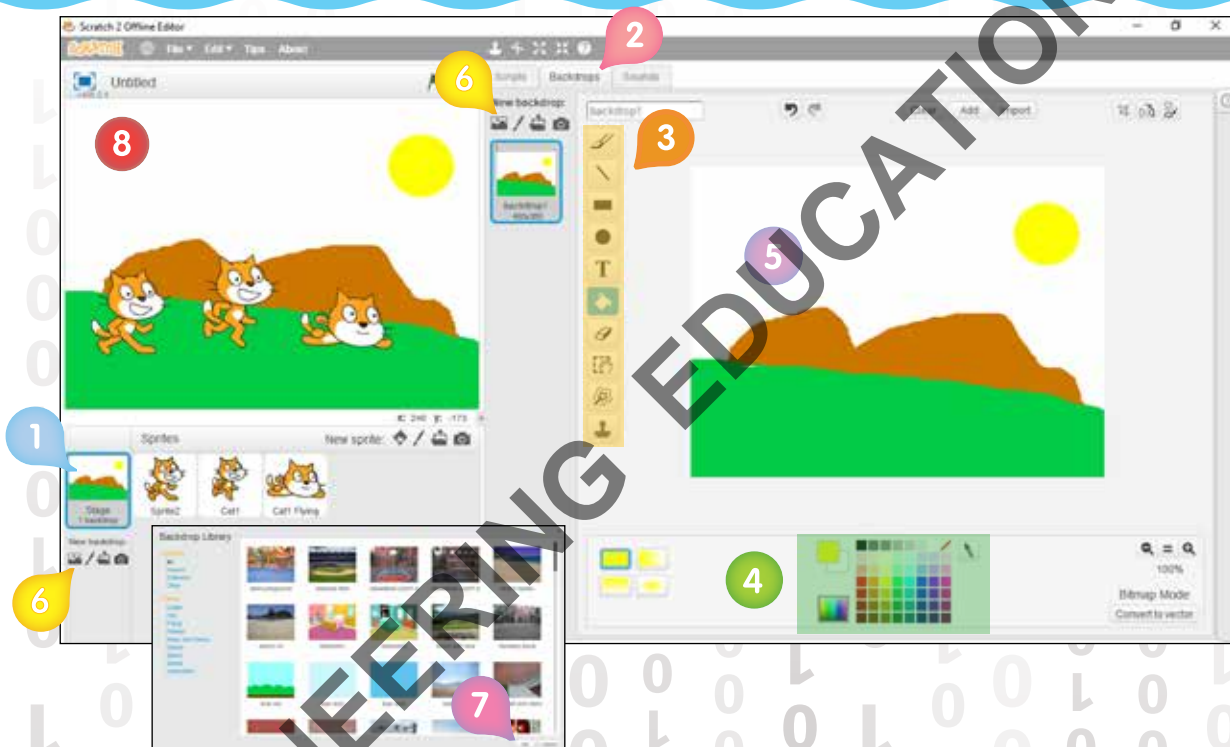
คือ คำสั่งเปลี่ยนชุดตัวละครไปยังชุดที่.....



1. คลิกเลือกตัวละครที่ต้องการเขียนโปรแกรมเปลี่ยนรูปร่าง
2. คลิกเลือกคำสั่ง `Costumes` ซึ่งเป็นคำสั่งจัดการชุดหรือรูปร่างของตัวละครแต่ละตัว
3. จะมีหน้าต่าง `Costumes` ขึ้นมา ให้สังเกตชุดของตัวละครด้านซ้ายซึ่งโดยปกติจะมี 2 ชุด
4. สามารถเลือกชุดหรือรูปร่างตัวละครเพิ่มได้โดยการกดที่
5. จะมีหน้าต่าง `Customize` คือหน้าต่างให้สามารถเลือกชุดหรือรูปร่างตัวละครเพิ่มเติมลงไป วาดหรือเปลี่ยนสีเองได้
6. เมื่อเลือกแล้วสังเกตชุดหรือรูปร่างที่เพิ่มขึ้น โดยในตัวอย่างจะเป็นแมวอน
7. คลิกเลือก `Scripts` `Looks` แล้วลากคำสั่ง `switch costume to costume1` ไปยัง พื้นที่สั่งงานเขียนโปรแกรม (Scripts area)
8. ลองใช้คำสั่งเปลี่ยนชุดหรือรูปร่างตัวละคร สังเกตการเปลี่ยนแปลงบริเวณเวที

การสร้างพื้นหลัง (Background)

ในโปรแกรม scartch สามารถวาดรูปเพื่อสร้างพื้นหลัง สามารถนำภาพที่มีอยู่ในโปรแกรม หรือนำรูปถ่ายจากภายนอกเข้ามาเป็นฉากพื้นหลังได้ เพื่อนำมาสร้างเป็นฉากเรื่องราวที่เหมาะสมกับตัวละคร




1 คลิกเลือก stage backdrop เป็นการเลือกเข้าสู่การแก้ไขพื้นหลัง

2 คลิกเลือกโหมด Backdrops จะมี Scripts Backdrops Sounds คำสั่งควบคุม ทำพื้นหลัง และเรื่องเสียง

3 ชุดคำสั่งการวาด ได้แก่ พู่กัน วาดเส้น สีเหลี่ยม วงกลม ตัวอักษร เทสี ยางลบ คัดลอก

4 ชุดคำสั่งการเลือกสี จะมีทั้งในตารางสีและสามารถเลือกเพิ่มเติมได้

5 พื้นที่วาดภาพ มีขนาดกว้าง 480 หน่วย สูง 360 หน่วย

6 หากต้องการเลือกภาพจากสมุดภาพในโปรแกรมสามารถเลือกได้โดยการคลิก  คำสั่ง Choose backdrop from library

7 ในหน้าต่าง Backdrops Library จะมีหมวดให้เลือก เช่น Indoor outdoor เป็นต้น เมื่อเลือกเรียบร้อยแล้วให้กด Ok

8 ให้สังเกตที่บริเวณเวที (Stage) เมื่อมีการเปลี่ยนแปลงพื้นหลัง (Backdrop)

การสร้างภาพเคลื่อนไหว (Animation)



สามารถทำได้โดยการใช้คำสั่งเปลี่ยนแปลงชุดหรือรูปร่าง(Costmue) และคำสั่งหยุดรอ wait สลับไปมาเพื่อให้ตาของเรามองเห็นภาพเป็นลักษณะของภาพเคลื่อนไหว

ชุดที่ 1 ชุดที่ 2 ชุดที่ 1 ชุดที่ 2 ชุดที่ 1

หยุดแสดง ภาพ 1 วินาที หยุดแสดง ภาพ 1 วินาที หยุดแสดง ภาพ 1 วินาที หยุดแสดง ภาพ 1 วินาที

Scripts Costumes Sounds Scripts Costumes Sounds

when green flag clicked

move 10 steps

switch costume to costume2

wait 1 secs

move 10 steps

switch costume to costume1

wait 1 secs

move 10 steps

switch costume to costume2

wait 1 secs

move 10 steps

switch costume to costume1

wait 1 secs

1 คลิกเลือกตัวละครที่ต้องการ ทำให้เป็นภาพเคลื่อนไหว



2 คลิกเลือกและลากคำสั่ง เริ่มต้นเมื่อซักรง ไปที่ Event แล้วลากคำสั่ง when flag click ไปยังพื้นที่ทำงาน

3 คลิกเลือกและลากคำสั่ง move จะอยู่ในชุดคำสั่ง Motion

4 คลิกเลือกและลากคำสั่ง เปลี่ยนรูปร่าง ซึ่งจะอยู่ใน Looks

5 ให้เลือกชุดรูปร่างที่ 2 ของแมว โดยการกดที่ลูกศรลงด้านขวามือ และทำการเลือก costume 2

6 คลิกเลือกและลากชุดคำสั่ง wait

7 เลือกคำสั่งซ้ำตั้งแต่ข้อ 3-6 แต่ให้เปลี่ยนชุดกลับมาเป็นชุดที่ 1 หรือ costume 1

8 เลือกคำสั่งซ้ำตั้งแต่ข้อ 3-6 อีกครั้งหนึ่ง และเลือกคำสั่งซ้ำอีกครั้ง ในข้อ 7 แล้วลองกดปุ่มให้เริ่มคำสั่งหรือกดที่ ธง

การวนซ้ำและวนซ้ำไม่สิ้นสุด (Loop Repeat , Loop Forever)

จากชุดคำสั่งการสร้างภาพเคลื่อนไหว จะเห็นได้ว่าการใช้คำสั่งที่ซ้ำกัน มีลักษณะเป็นชุดคำสั่ง คือ move , switch costume และ wait ซึ่งเราสามารถใส่คำสั่ง Repeat เพื่อให้ไม่ต้องเขียนคำสั่งที่ยาวต่อเนื่อง



1. คลิกเลือกตัวละคร แมว 
2. คลิกเลือกและลากคำสั่ง เริ่มต้นเมื่อซักรง ไปที่ Event แล้วลากคำสั่ง when flag click ไปยังพื้นที่ทำงาน
3. คลิกเลือกและลากคำสั่ง Loop Repeat โดยกดที่คำสั่ง Control และ ลากคำสั่ง Repeat ไปยังพื้นที่ทำงาน
4. คลิกเลือกและลากคำสั่ง 3-7 ของ หน้าการทำภาพเคลื่อนไหว แล้วลองกดปุ่มให้เริ่มคำสั่งหรือกดที่ ธง 
5. ในกรณีนี้สามารถใช้คำสั่ง next costume เพื่อลดชุดคำสั่งลงมาได้อีก แต่ต้องปรับ จำนวนครั้งที่ทำซ้ำเพิ่มจาก 10 เป็น 20
6. สามารถทำให้เดินวนซ้ำไม่มีที่สิ้นสุดได้ โดยใช้คำสั่ง Forever แทน Repeat
7. สามารถทำให้ตัวละครเดินเร็วขึ้นได้ด้วยการ ลดเวลาการหยุดรอ จาก 1 วินาที เป็น 0.5 วินาที 
8. สามารถทำให้ตัวละครเมื่อเดินจนสุดขอบ พื้นที่การทำงานให้ดังกลับได้โดยใช้คำสั่ง if on edge , bounce 

การกำหนดเหตุการณ์เพื่อตอบโต้กับผู้ใช้งาน (Events)

การสร้างเหตุการณ์(Events) เพื่อตอบโต้กับผู้ใช้งาน ซึ่งในโปรแกรม scratch สามารถกำหนดเหตุการณ์ต่างๆ ได้โดยคลิกเลือกที่คำสั่ง **Events**

Events

ซึ่งมีเหตุการณ์ที่สามารถตอบโต้กับผู้ใช้งานดังต่อไปนี้



1 When Flag Clicked

กำหนดเหตุการณ์เมื่อธงถูกกด

2 When Key pressed

กำหนดเหตุการณ์เมื่อกดปุ่มต่างๆ บน Keyboard เช่น spacebar, a b c เป็นต้น

3 When this sprite Clicked

กำหนดเหตุการณ์เมื่อตัวละครถูกคลิก

4 When backdrop switches to

กำหนดเหตุการณ์เมื่อ ฉากหลัง ถูกเปลี่ยนไปเป็น ฉากที่กำหนด

5 When >

กำหนดเหตุการณ์เมื่อความดังเสียง(Loudness) ช่วงเวลา(Timer), วิดีโอเคลื่อนที่ (VDO Motion)

6 When I receive (Message1)

กำหนดเหตุการณ์เมื่อ โปรแกรม ได้รับข้อความที่กำหนดไว้

7 Broadcast (Message1)

กำหนดให้ส่งข้อความเมื่อถึงช่วงเวลาที่ต้องการ

8 Broadcast (Message1) and wait

กำหนดให้ส่งข้อความเมื่อถึงช่วงเวลาที่ต้องการ และรอเหตุการณ์อีกเหตุการณ์

ตัวอย่างการสร้างโปรแกรมการตอบโต้กับผู้ใช้งาน

โจทย์ : ให้สร้างเกมเกี่ยวกับตรรกะ การเรียงลำดับรูปแบบ โดยผู้ใช้งานสามารถเลือกคำตอบได้ว่าอันไหนถูกและผิด พร้อมทั้งมีคะแนนขึ้นให้ทราบ โดยถ้าเลือกคำตอบถูก ได้ 10 คะแนน และ ผิด ลบ 10 คะแนน



1 สร้างตัวละครและฉากตามโจทย์
ทำตามโจทย์หรือกำหนดเอง

2 กำหนดค่าตัวแปรใหม่ คะแนน(score)
โดยกดที่ data --> make a variable
จะมีหน้าต่างแล้วพิมพ์ คำว่า score

3 คลิกเลือกตัวละคร รหัสสีเขียว
เพื่อกำหนด event

4 คลิกเลือกคำสั่งกำหนดเหตุการณ์
เมื่อตัวละครตัวนี้ถูกกด ที่ event
When this sprite clicked

5 คลิกเลือกคำสั่ง Change score by
เป็นการกำหนดการเปลี่ยนแปลงค่าคะแนน
ให้เพิ่มหรือลดลงตามที่ต้องการ

6 คลิกเลือกคำสั่งถ้าถูกต้องให้เปลี่ยนเป็นข้อ
ต่อไป โดยเลือกคำสั่ง Look-->
switch backdrop to next backdrop

7 สำหรับคำตอบที่ผิดให้คลิกที่ตัวลูกบาศ
ให้คลิกและเลือกเหมือนข้อ 4-5 แต่ให้ลดคะแนน
ลงเป็น -10 คะแนน ถ้าตอบข้อนี้

8 อาจจะทำหน้าที่ผิดเพื่อแสดงเพิ่มเติม
หรืออาจจะเพิ่มเสียงผิดก็ได้

การกำหนดเงื่อนไข ถ้า(If...then.....)

การกำหนดเงื่อนไขเป็นการสร้างรูปแบบหรือตรรกะวิธีการตัดสินใจในการดำเนินการใดๆกับเหตุการณ์นั้น โดยถ้าเกิดเหตุการณ์หรือเงื่อนไขที่กำหนดไว้เป็นจริง แล้วตัวละครหรือฉากจะดำเนินการตามที่ถูกกำหนด

โจทย์ปัญหา : หากตัวละครแมวเดินชนผีเสื้อเดินแล้วให้แมวร้องเหมียวและเปลี่ยนทิศทางการเคลื่อนที่



1 สร้างตัวละครและฉากตั้งรูป
โดยให้มีแมว และ ผีเสื้อ

2 กำหนดเงื่อนไขให้ แมว และ ผีเสื้อ
เคลื่อนที่ และหาชนของให้ดังกลับ
ดังตัวอย่าง

3 กำหนดฟังก์ชัน If
โดยคลิกเลือกคำสั่ง
ใน Control



4 คลิกเลือกและลากคำสั่งการกำหนดเงื่อนไข
touching color ?

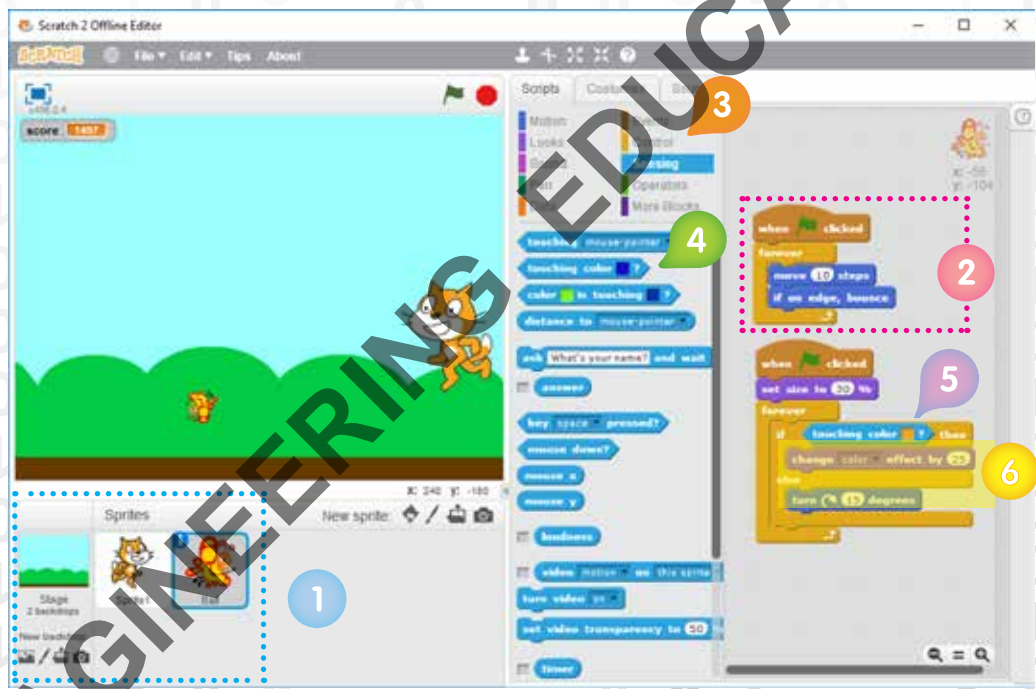
5 กำหนดเงื่อนไข
ถ้าสัมผัสสีส้ม
(โดยการกดที่สีในช่องคำสั่งแล้วคลิกสีที่ผีเสื้อ)

6 กำหนดเหตุการณ์ตามต้องการ
โดยถ้าเหตุการณ์แมวสัมผัสกับสีส้ม
ของผีเสื้อแล้วจะเกิดเหตุการณ์ใดขึ้น

การกำหนดเงื่อนไขถ้า...เป็นจริงแล้ว.....ไม่จริงแล้ว.....(If...else...)

เป็นการกำหนดเงื่อนไขให้กับตัวละครหรือฉากว่าถ้าหากเกิดเหตุการณ์หรือเงื่อนไขใดๆแล้วเป็นจริง จะดำเนินการอย่างไร หรือหากไม่จริงแล้วจะดำเนินการอย่างไรต่อไป

โจทย์ปัญหา : หากตัวละครผีเสื้อชนกับแมวแล้วให้ผีเสื้อเปลี่ยนสี หากไม่ชนกับผีเสื้อให้เปลี่ยนทิศทางการเคลื่อนที่ไป 15 องศา



1 สร้างตัวละครและฉากตั้งรูป โดยให้มีแมว และ ผีเสื้อ

2 กำหนดเงื่อนไขให้ แมว และ ผีเสื้อ เคลื่อนที่ และหาชนของให้ตั้งกลับ ดังตัวอย่าง

3 กำหนดฟังก์ชัน If else โดยคลิกเลือกคำสั่ง ใน Control

4 คลิกเลือกและลากคำสั่งการกำหนดเงื่อนไข **touching color ?**

5 กำหนดเงื่อนไข ถ้าสัมผัสสีส้มของแมว (โดยการกดที่สีในช่องคำสั่งแล้วคลิกสีที่ผีเสื้อ)

6 กำหนดเหตุการณ์ถ้าเจอสีส้มของแมวให้เปลี่ยนสี และถ้าหากไม่เจอสีส้มให้เปลี่ยนทิศทางการหมุนไป 15 องศา

ให้นักเรียนสร้างนิทานหรือเรื่องราว ในชีวิตประจำวันของตนเอง

Investigate Idea

ลิปดาได้ฟังเรื่องราวของเพื่อนที่เดินทางไปเที่ยวได้
พบเห็นสิ่งต่างๆมากมาย ลิปดาคิดว่าน่าจะนำเรื่อง
ราวนั้นมาทำการ์ตูนสั้น หรือเรื่องราวให้กับ
เพื่อนๆ คนอื่นได้รับทราบจะได้ไม่ต้องเล่าใหม่



- Mission**
1. ให้นักเรียนสร้างนิทานหรือเรื่องราวของตนเอง ใน
โปรแกรม scratch โดยให้มีการตอบโต้กับผู้ใช้ มีภาพเคลื่อนไหว
 2. ให้ออกแบบโปรแกรมก่อนลงมือทำ (storyboard, อัลกอริทึม)
 3. นำเสนอผลงานพร้อมจัดการประกวดผลงาน

Materials 1. คอมพิวเตอร์และโปรแกรม Scratch 2. อุปกรณ์เครื่องเขียน

นักเรียนเข้าใจการออกแบบโปรแกรมโดยใช้ Story Board
และอัลกอริทึม การเขียนโปรแกรม Scratch และวิธีการหา
ข้อผิดพลาดหรือไม่

Plan & Design

ร่วมกันวางแผนและออกแบบโปรแกรมโดยใช้โปรแกรม scratch
ข้อผิดพลาดหรือไม่



KEYWORD

การออกแบบ
โปรแกรมและหา
ข้อผิดพลาด

STANDARD

ว.ศ.๒ ป๔/๒

Building & Testing

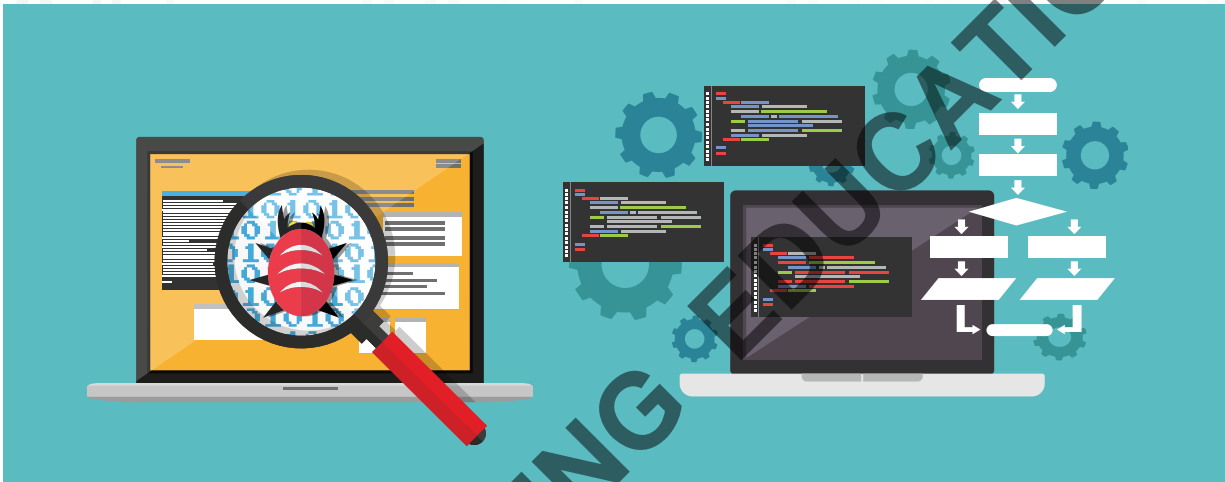
ให้นักเรียนลงมือทำแผนที่กำหนดไว้

Evaluation & Redesign

1. นักเรียนสามารถทำภารกิจนี้สำเร็จหรือไม่
2. ส่วนที่ยากที่สุดของภารกิจนี้คืออะไร
3. แนวความคิดที่ได้จากการทำกิจกรรมคืออะไร

การตรวจสอบข้อผิดพลาดของโปรแกรม (Testing and Debugging)

หลังจากที่ทำการเขียนโปรแกรมเสร็จสิ้นแล้วโปรแกรมนั้นจะต้องได้รับการตรวจสอบก่อนว่า มีข้อผิดพลาด (error) ในโปรแกรมหรือไม่ หากพบข้อผิดพลาดให้ตรวจสอบ การทำงานทีละคำสั่ง เมื่อพบจุดที่ทำให้ผลลัพธ์ไม่ถูกต้อง ให้ทำการแก้ไขจนกว่าจะได้ผลลัพธ์ที่ถูกต้อง



เราออกแบบโปรแกรมมาดีมากแล้วทำไมถึงใช้งานไม่ได้ละคะ

แสดงว่าโปรแกรมที่ทำไว้มีข้อผิดพลาดไงละครับ เราต้องตรวจสอบหาข้อผิดพลาดทีละคำสั่ง เมื่อพบจุดที่ทำให้ผลลัพธ์ไม่ถูกต้องจึงลงมือแก้ไข



ตัวอย่างการหาข้อผิดพลาดในโปรแกรม (Debuggin)

โจทย์ : จากการออกแบบโปรแกรมสั่งให้ตัวละคร เดินมาพบกันแล้วให้ตัวละครตัวที่ 1 แมว ไปกับตัวละครตัวที่ 2 ค้างคาวแล้ว พูด "Hello"



```
when clicked
  forever loop
    move 10 steps
    wait 0.1 secs
    on edge, bounce
    if touching color ? then
      say Hello! for 1 secs
      turn 180 degrees
```

ปัญหาเกิดขึ้นเมื่อตัวละครสองตัวพบกันแต่ไม่พูดคำว่า Hello

แบ่งแยกปัญหาออกเป็นส่วนๆ :

- แสดงว่าปัญหาเกิดในช่วง
- การแสดงผลคำพูดหรือเงื่อนไข

หาจุดสำคัญของปัญหา :

ให้ตรวจสอบเงื่อนไขว่าถูกต้องหรือไม่

```
if touching color ? then
  say Hello! for 1 secs
  turn 180 degrees
```

ผลการตรวจสอบข้อผิดพลาด

พบว่าเงื่อนไขผิดพลาดเนื่องจากเมื่อตัวละครสัมผัสสีฟ้าถึงจะพูด

ทำการแก้ไขเพื่อให้ได้ผลลัพธ์ตามที่ต้องการ

เปลี่ยนเงื่อนไขให้พบสีน้ำตาลของค้างคาว

```
touching color ?
```

```
touching color ?
```

การฝึกตรวจหาข้อผิดพลาด จากโปรแกรมของผู้อื่น

ดร.วี กำลังทำงานอยู่หน้าคอมพิวเตอร์ ลิปตามาเห็นจึงสงสัย ดร.วี จึงตอบว่า กำลังหาข้อผิดพลาดของโปรแกรมอยู่นะครับ ถ้าหากหาข้อผิดพลาดพบจะทำให้โปรแกรมทำงานได้อย่างสมบูรณ์แบบครับ และไม่เพียงแต่นักเขียนโปรแกรมเท่านั้นนะที่ควรจะต้องฝึกฝนทักษะนี้ ทุกคนควรจะต้องฝึกฝนเนื่องจากจะทำให้ เราสามารถนำไปหาสาเหตุของปัญหาในชีวิตประจำวันของเราได้เช่นกัน

Missions :

1. ให้นักเรียนแบ่งกลุ่ม กลุ่มละ 4 คน ออกแบบโปรแกรม Scratch ที่มีข้อผิดพลาด
2. แลกเปลี่ยนให้เพื่อนๆ แต่ละกลุ่มหาข้อผิดพลาด
3. กลุ่มใดทำได้เร็วที่สุดและถูกต้องที่สุดเป็นผู้ชนะ

Materials :

1. คอมพิวเตอร์โปรแกรม scratch
2. อุปกรณ์เครื่องเขียน

Ask :

การฝึกตรวจหาข้อผิดพลาดของโปรแกรมสำคัญอย่างไร

Plan & Design

ให้ออกแบบและวางแผนการเขียนโปรแกรมที่มีข้อผิดพลาด



ให้นักเรียนออกแบบโปรแกรมที่มีข้อผิดพลาดเพื่อให้เพื่อนๆหา

IMAGINEERING EDUCATION



- การหาสาเหตุของข้อผิดพลาดของโปรแกรมควรทำอย่างไร
- การแบ่งแยกปัญหาและหารูปแบบจะช่วยให้หาสาเหตุของปัญหาได้อย่างไร
- การหาสาระสำคัญของปัญหาจะช่วยให้หาสาเหตุของปัญหาได้อย่างไร