

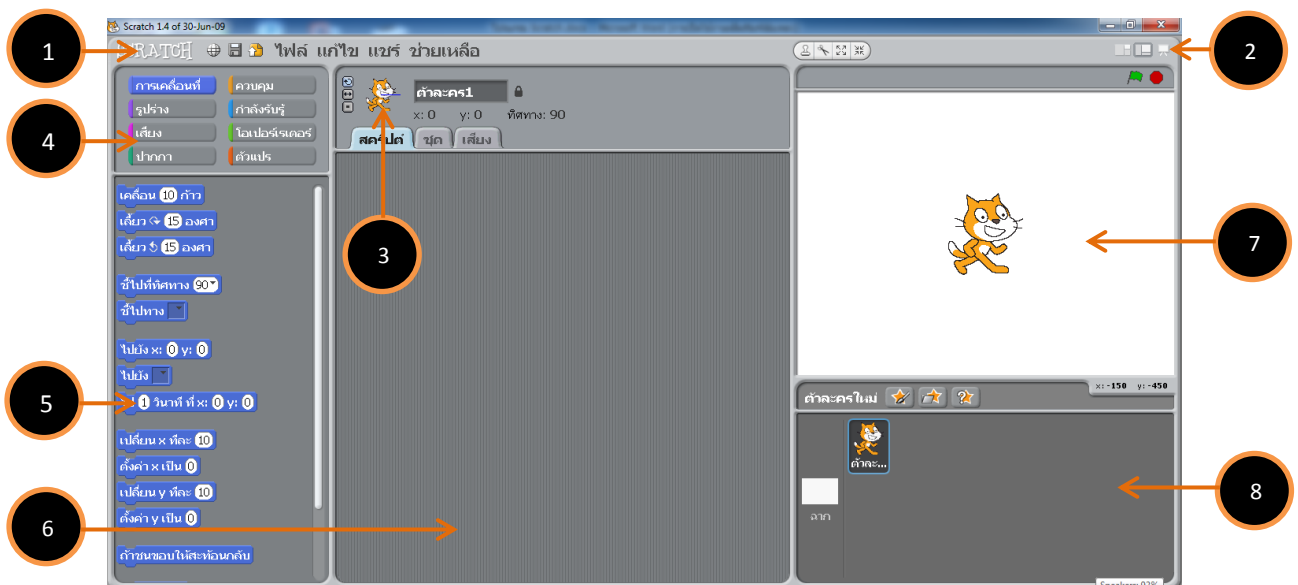


รู้จักโปรแกรม Scratch

โปรแกรม Scratch (อ่านว่า สะ - แครช) เป็นโปรแกรมภาษา ที่ผู้เรียนสามารถสร้างชิ้นงานได้อย่างง่าย เช่น นิทานที่สามารถโต้ตอบกับผู้อ่านได้ ภาพเคลื่อนไหว เกมดนตรี และศิลปะ และเมื่อสร้างเป็นชิ้นงานเสร็จแล้ว สามารถนำชิ้นงานที่สร้างสรรค์นี้ แสดง และแลกเปลี่ยนความคิดเห็นร่วมกับผู้อื่นบนเว็บไซต์ได้ ทำให้ผู้เรียนได้ เรียนรู้หลักการทางคณิตศาสตร์ และแนวคิดการโปรแกรมไปพร้อมๆ กับการคิดอย่างสร้างสรรค์ มีเหตุผล เป็นระบบ และเกิดการทำงานร่วมกัน

1. ส่วนประกอบหลักของโปรแกรม

หน้าต่างการทำงานของโปรแกรม Scratch มีส่วนประกอบหลักดังนี้



1 แถบเมนูเครื่องมือ

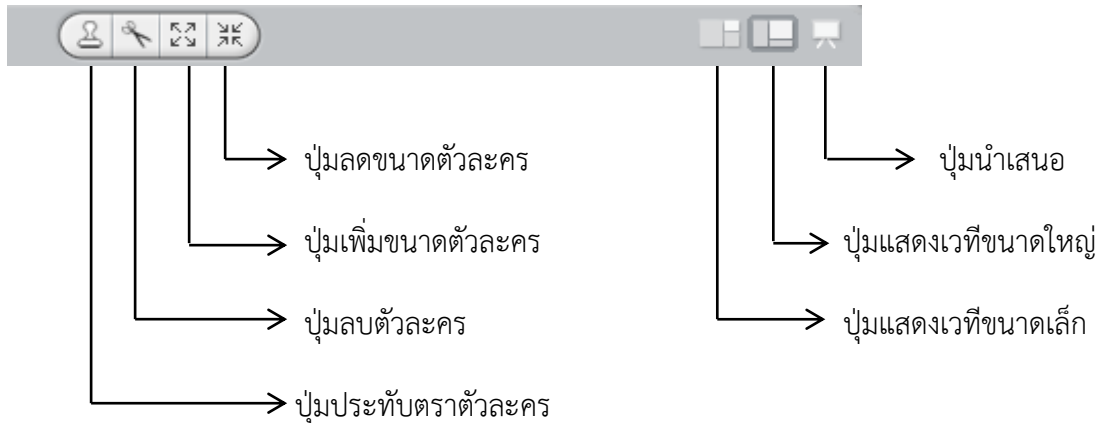
SCRATCH โฟล์ แก๊ว แบร์ ข่ายเหลือ

ปุ่มเปลี่ยนภาษา

ปุ่มบันทึกโปรเจกต์

ปุ่มเผยแพร่ผลงานทางเว็บไซต์ <http://scratch.mit.edu/>

2 เครื่องมือเวที (Stage Toolbar)



3 ข้อมูลของเวที หรือตัวละครที่ถูกเลือก (Sprite Header Pane)

4 กลุ่มบล็อก (Block Palette)

5 บล็อกในกลุ่มที่เลือก

6 พื้นที่ทำงาน (Script Area)

7 เวที (Stage)

8 รายการตัวละคร และเวทีที่ใช้ในโปรเจกต์ปัจจุบัน (Sprites Pane)

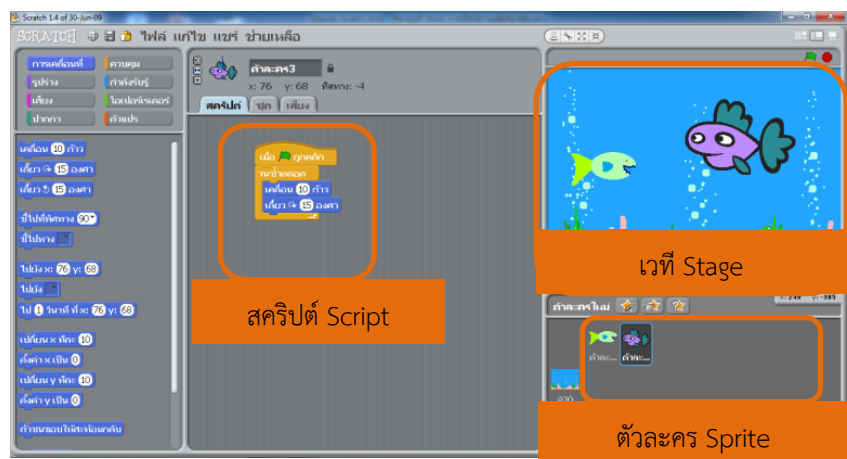
2. รู้จักโปรเจกต์

โปรเจกต์ใน Scratch มีโครงสร้าง 3 ส่วนประกอบด้วย

2.1 เวที (Stage)

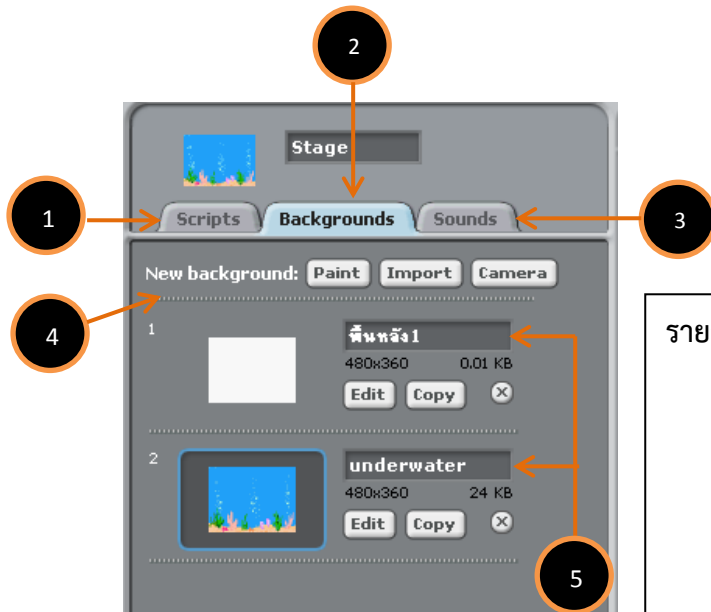
2.2 ตัวละคร (Sprite)

2.3 สคริปต์ (Script)



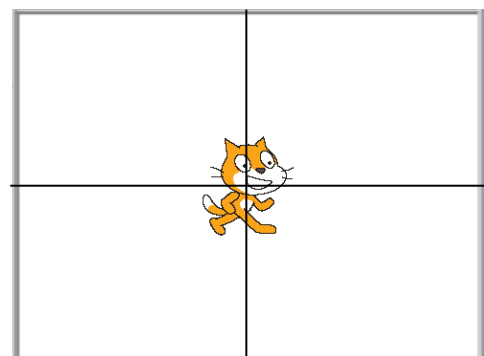
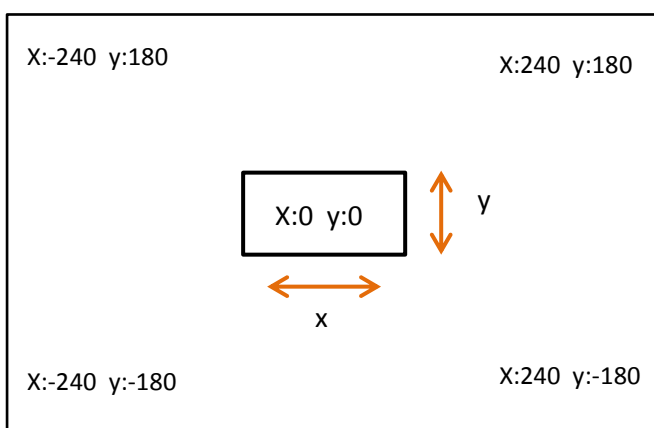
2.1 เวที (Stage)

เวทีมีความกว้าง 480 หน่วย สูง 360 หน่วย ในแต่ละโปรเจกต์มีเวทีเดียว จึงมีชื่อเดียวและไม่สามารถเปลี่ยนชื่อได้ เวทีใช้แสดงผลการทำงานของสคริปต์ (Script) เสียง (Sound) หรือพื้นหลัง (Background) ได้ และพื้นหลังที่จะแสดงบนเวทีจะต้องมีขนาดไม่เกินขนาดของเวที (480 X 360) ถ้าพื้นหลังที่ใช้มีขนาดใหญ่กว่า โปรแกรม Scratch จะลดขนาดพื้นหลังนั้นอัตโนมัติเพื่อให้พอดีกับขนาดของเวที



- รายละเอียดของเวที
1. แท็บ Scripts
 2. แท็บ Backgrounds
 3. แท็บ Sounds
 4. สร้างพื้นหลังใหม่ (New Background)
 5. พื้นหลังลำดับที่ 1 และ 2

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



2.2 ตัวละคร

ตัวละครแต่ละตัวจะมีข้อมูลแตกต่างกัน โดยสามารถคลิกที่ภาพตัวละครในพื้นที่แสดงรายการตัวละครเพื่อดูข้อมูลตัวละครนั้น เช่น ตัวละครปลา



หมายเลข	ข้อมูล	รายละเอียดข้อมูลตัวละคร
1	ชื่อตัวละคร	Sprite1
2	ตำแหน่งบนเวที	X: -110 y: 53
3	ทิศทางการเคลื่อนที่	90 องศา
4	รูปแบบการหมุนของตัวละคร มี 3 ลักษณะ	หันได้เฉพาะซ้ายหรือขวา
5	การลากตัวละครในโหมดนำเสนอหรือบนเว็บไซต์	ลากตัวละครไม่ได้

2.2.1 ชื่อตัวละคร

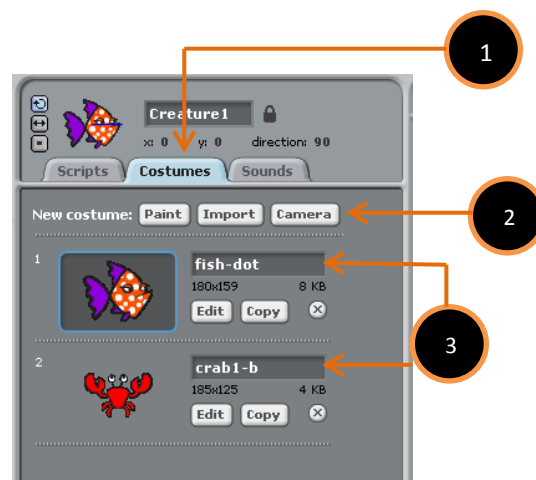
โปรแกรมจะตั้งชื่อตัวละครให้เป็น Sprite1, Sprite2, Sprite3... ตามลำดับที่สร้างขึ้นโดยอัตโนมัติ ถ้าต้องการเปลี่ยนชื่อตัวละครให้พิมพ์ชื่อใหม่บนแถบชื่อหมายเลข 1 ตามภาพด้านบน

2.2.2 ชุดตัวละคร

ชุดตัวละคร (Costumes) เป็นภาพของตัวละคร ซึ่งสามารถเปลี่ยนแปลงภาพเดิม หรือเพิ่มภาพใหม่ และอาจเขียนสคริปต์เพิ่มให้กับตัวละครเปลี่ยนชุด หรือให้มองเห็นเป็นการเคลื่อนไหวในรูปแบบต่างๆ ตามต้องการ


รายละเอียดชุดตัวละคร

1. แทะ็บ Costumes
2. สร้างชุดตัวละครใหม่
3. ชุดตัวละคร ในตัวอย่างเป็นชุดของตัวละครชื่อ Creature1
ชื่อชุด fish-dot เป็นปลาลายจุด และ crab1-b เป็นปู



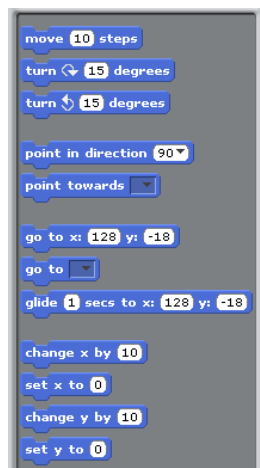
2.3 สคริปต์

สคริปต์คือชุดคำสั่งสำหรับตัวละครหรือเวที เพื่อสั่งให้ตัวละครหรือเวทีทำงานตามวัตถุประสงค์ที่ต้องการ โดยการเลือกสคริปต์จากกลุ่มบล็อก ซึ่งแบ่งเป็น 8 กลุ่มดังนี้

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 เล่นเสียงสัตว์ กลอง โน้ต
 Operators	ตัวดำเนินการ เช่น บวก และ หรือ
 Pen	ปากกา เช่น ยกปากกาขึ้น ตั้งค่าสีปากกา
 Variables	ตัวแปร เช่น สร้างค่าตัวแปร

เมื่อคลิกที่กลุ่มบล็อกใด จะปรากฏบล็อกในกลุ่มนั้น บล็อกสำหรับตัวละครและเวทีอาจมีความแตกต่างกัน เช่น กลุ่มบล็อก Motion ของตัวละครจะมีบล็อกตั้งรูปด้านซ้าย ส่วนรูปด้านขวาเป็นของเวทีซึ่งไม่มีบล็อก Motion เนื่องจากเวทีเคลื่อนที่ไม่ได้

บล็อก Motion สำหรับตัวละคร



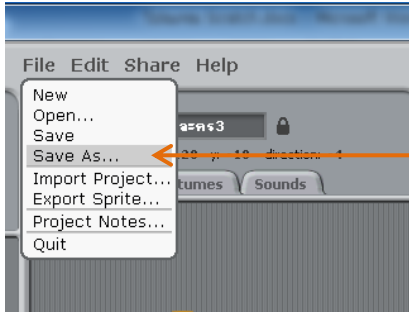
ไม่มีบล็อก Motion สำหรับเวที



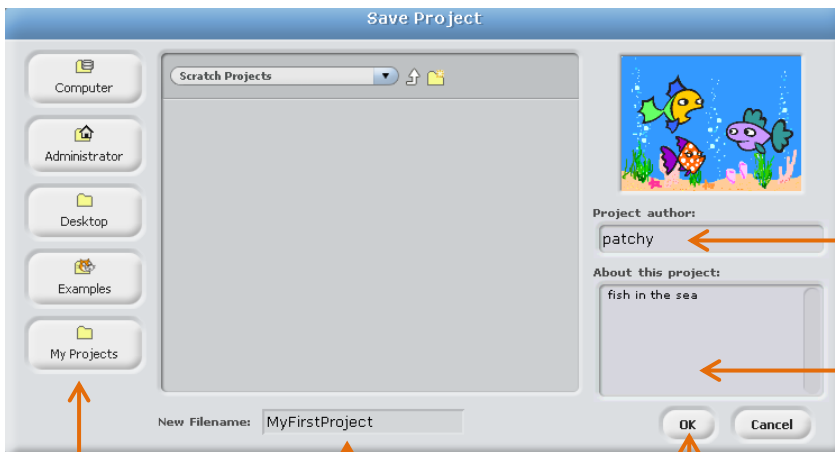
3. การสั่งให้โปรเจกต์เริ่มทำงานและหยุดทำงาน

การสั่งให้โปรเจกต์เริ่มทำงาน ทำได้โดยคลิก  ซึ่งอยู่มุมขวาบนขวาของเวที โดยทุกสคริปต์ของทุกตัวละครและเวที ที่เริ่มต้นสคริปต์ด้วยบล็อก  จะเริ่มทำงานพร้อมกัน และถ้าต้องการหยุดการทำงานทั้งโปรเจกต์ ให้คลิก 

3. การบันทึกโปรเจกต์



1 คลิกเมนู File -> Save หรือ Save As จะปรากฏกรอบโต้ตอบ Save Project



2 พิมพ์ชื่อผู้ทำโปรเจกต์

3 พิมพ์รายละเอียดเกี่ยวกับโปรเจกต์

4

พิมพ์ชื่อโปรเจกต์

5

คลิกปุ่ม Ok

6

โปรเจกต์จะถูกบันทึกไว้ที่ My Documents\Scratch Projects\ จะได้ไฟล์ข้อมูลที่มีส่วนขยายเป็น .sb ในโฟลเดอร์ที่ใช้บันทึกงาน เช่น MyFistProject.sb





กลุ่มบล็อกต่างๆ

1. บล็อก Motion

Motion

กลุ่มบล็อก Motion ประกอบด้วยบล็อกที่ใช้เพื่อกำหนดรูปแบบการเคลื่อนที่ให้ตัวละคร เช่น เคลื่อนที่ไปข้างหน้า หมุน ไปที่ตำแหน่งต่างๆ บนเวที โดยสามารถกำหนดค่าของการเคลื่อนที่ได้ บล็อกในกลุ่มบล็อก Motion ดังตาราง

บล็อก	ความหมาย
	ไปข้างหน้า หรือถอยหลัง ตัวอย่างไปข้างหน้า 10 หน่วย
	หมุนตามเข็มนาฬิกา ตัวอย่าง หมุนตามเข็มนาฬิกา 15 องศา
	หมุนทวนเข็มนาฬิกา ตัวอย่าง หมุนทวนเข็มนาฬิกา 15 องศา
	ระบุทิศทางที่ต้องการ ตัวอย่าง ระบุทิศทาง 90 องศา
	ระบุทิศทางตามที่กำหนด
	ไปที่ตำแหน่ง (x, y) บนเวที ตัวอย่าง ไปตรงกลางเวที (x=0, y=0)
	ไปที่ตำแหน่งที่กำหนด บนเวที ตัวอย่าง ไปที่ตำแหน่งตัวเมาส์ชี้ ไปที่ตำแหน่ง Sprite1
	เคลื่อนไปที่ตำแหน่ง (x, y) บนเวทีในเวลาที่กำหนด ตัวเลขยิ่งมากยิ่งช้า ตัวอย่าง ไปตำแหน่งกลางเวที ในเวลา 1 วินาที
	เปลี่ยนตำแหน่งของตัวละครไปด้านซ้ายหรือขวา ตัวอย่าง เปลี่ยนตำแหน่งตัวละครไปด้านขวา 10 หน่วย
	ตั้งค่าตำแหน่งตัวละครไปด้านซ้ายหรือขวา ตัวอย่าง ตั้งค่าตัวละครไปที่ตำแหน่ง x = 0 คือ กลางเวที X = 240 คือ ขอบขวาสุด X = -240 คือ ขอบซ้ายสุด


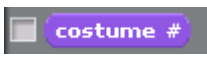



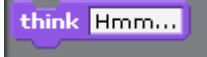


บล็อก	ความหมาย
	เปลี่ยนตำแหน่งของตัวละครไปด้านบนหรือล่าง ตัวอย่าง เปลี่ยนตำแหน่งตัวละครไปด้านบน 10 หน่วย
	ตั้งค่าตำแหน่งตัวละครไปด้านบนหรือด้านล่าง ตัวอย่าง ตั้งค่าตัวละครไปที่ตำแหน่ง $y = 0$ กลางเวที $y = 180$ คือ ขอบบนสุด $y = -180$ คือ ขอบล่างสุด
	ตัวละครชนขอบให้สะท้อนกลับ
	ตำแหน่งตัวละครตามแนวแกน x ถ้าต้องการให้แสดงค่าตำแหน่งของตัวละครตามแนวแกน x ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมาย ✓ ซึ่งแสดงตำแหน่งบนเวที
	ตำแหน่งตัวละครตามแนวแกน y ถ้าต้องการให้แสดงค่าตำแหน่งของตัวละครตามแนวแกน y ให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมาย ✓ ซึ่งแสดงตำแหน่งบนเวที
	ทิศทางตัวละคร ถ้าต้องการให้แสดงค่าทิศทางตัวละครให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมาย ✓ ซึ่งแสดงทิศทางบนเวที



2. บล็อก Looks

Looks

กลุ่มบล็อก Looks ใช้สำหรับการเขียนสคริปต์เพื่อสั่งให้ตัวละครหรือเวทีแสดงคุณสมบัติต่างๆ เช่น ตัวละครพูด เปลี่ยนชุดตัวละคร เปลี่ยนสีตัวละคร เปลี่ยนขนาดตัวละคร เปลี่ยนพื้นหลัง บล็อกในกลุ่มบล็อก Looks ของตัวละคร ดังตารางที่ 2.1 และของเวที ดังตารางที่ 2.2

ตารางที่ 2.1 บล็อกในกลุ่มบล็อก Looks ของตัวละคร

บล็อก	ความหมาย
	สลับชุดตัวละคร ตัวอย่าง สลับชุดตัวละครเป็นชุด costume2
	เปลี่ยนชุดตัวละครเป็นชุดถัดไปที่มีอยู่ในรายการ
	ชุดตัวละครชุดที่... ถ้าต้องการให้แสดงชุดตัวละครให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก   ซึ่งแสดงเลขชุดตัวละครบนเวที
	แสดงคำพูดตามเวลาที่กำหนด ตัวอย่าง แสดงคำพูด Hello! เป็นเวลา 2 วินาที
	แสดงคำพูด ตัวอย่าง แสดงคำพูด Hello!
	แสดงความคิดตามเวลาที่กำหนด ตัวอย่าง แสดงความคิด Hmm... เป็นเวลา 2 วินาที
	แสดงความคิด ตัวอย่าง แสดงความคิด Hmm..
	เปลี่ยนเทคนิคการแสดงผลภาพตัวละคร โดยเลือกเทคนิคพิเศษจากเครื่องหมาย▼ 
	ตั้งค่าเทคนิคการแสดงผลภาพตัวละครตามที่กำหนด ตัวอย่าง ตั้งค่าเทคนิค color ให้เป็น 0 (สีเดิม)
	ล้างค่าเทคนิคการแสดงผลภาพตัวละครที่กำหนด
	เปลี่ยนขนาดตัวละครตามที่กำหนด ถ้าค่า + จะเป็นการเพิ่มขนาด ค่า - เป็นการลดขนาด ตัวอย่าง เพิ่มขนาดตัวละคร 10%
	ขนาดตัวละครเป็น % ตัวอย่าง กำหนดขนาดตัวละครเป็น 100% (ขนาดจริง)

บล็อก	ความหมาย
	ขนาดของตัวละคร ถ้าต้องการให้แสดงค่าขนาดตัวละครบนเวทีให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก 
	แสดงตัวละคร
	ซ่อนตัวละคร
	นำตัวละครไปไว้ด้านหน้าสุด
	ย้ายตัวละครไปไว้ด้านหลังตามจำนวนเลเยอร์ (ชั้น) ที่กำหนด ตัวอย่าง ย้ายตัวละครไปไว้ด้านหลัง 1 เลเยอร์


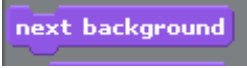
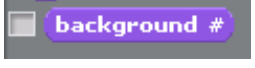
เลเยอร์ คือ ชั้นการวางซ้อนกันของตัวละคร เลเยอร์จะใช้เพื่อระบุว่าตัวละครนั้นอยู่ที่ชั้นใด
เช่น ถ้าต้องการสร้างภาพให้ตึกถูกต้นไม้บัง



ย้ายตึกซึ่งเป็นตัวละครที่ 1 ไปอยู่หลัง
ต้นไม้ซึ่งเป็นตัวละครที่ 2,3 โดยใช้คำสั่ง




ตารางที่ 2.2 บล็อกในกลุ่มบล็อก Looks ของเวที

บล็อก	ความหมาย
	สลับพื้นหลัง ตัวอย่าง สลับพื้นหลังเป็นพื้นหลัง background
	สลับพื้นหลังเป็นพื้นหลังถัดไปที่มีอยู่ในรายการ
	พื้นหลังที่... ถ้าต้องการให้แสดงหมายเลขพื้นหลังบนเวทีให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก

3. บล็อก Sound

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

บล็อก	ความหมาย
	เล่นเสียง
	เล่นเสียงจนจบ แล้วทำคำสั่งถัดไป
	หยุดเสียงทั้งหมด
	เล่นเสียงเครื่องตี โดยมีประเภทของเครื่องตีชนิดต่างๆ ให้เลือกได้ จำนวน 47 ชนิด (ระบุค่า 35-81) และกำหนดจังหวะในการตีแต่ละครั้งได้
	หยุดพักเสียง ตามเวลาที่กำหนด
	เล่นเสียงโน้ตดนตรี (โด เร มี..) มีค่าให้เลือกได้ 25 เสียง (ระบุค่า 48-72) และกำหนดค่าเวลาในการเล่นเสียงได้
	กำหนดชนิดของเครื่องดนตรีชนิดต่างๆ โดยสามารถเลือกชนิดของเครื่องดนตรีได้ 128 ชนิด (ระบุค่า 1-128)
	เพิ่มหรือลดระดับความดังของเสียง
	กำหนดความดังของเสียงเป็นเปอร์เซ็นต์
	เพิ่มหรือลดจังหวะของเสียง
	กำหนดจังหวะของเสียง

4. บล็อก Pen



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

บล็อก	ความหมาย
	ล้างภาพวาด
	วางปากกา
	ยกปากกา
	กำหนดสีปากกา สามารถคลิกเลือกสีได้
	กำหนดสีปากกา โดยระบุค่าสีเป็นตัวเลข
	เพิ่มหรือลดค่าสีปากกา
	เพิ่มหรือลดเฉดสีปากกา
	กำหนดเฉดสีของปากกา โดยระบุค่าเฉดสีเป็นตัวเลข
	เพิ่มหรือลดขนาดของปากกา
	กำหนดขนาดปากกา โดยระบุขนาดเป็นตัวเลข
	ประทับตราตัวละคร

5. บล็อก Sensing



กลุ่มบล็อก Sensing ใช้ตรวจสอบการรับรู้ต่างๆ เช่น การสัมผัส เสียง ระยะเวลา มีดังนี้

บล็อก	ความหมาย
	ตรวจสอบว่าตัวละครนั้นสัมผัสกับ ตัวชี้เมาส์ ขอบเวที หรือตัวละครอื่นหรือไม่
	ตรวจสอบว่าตัวละครนั้นสัมผัสกับสีที่กำหนดหรือไม่
	ตรวจสอบว่าสีตัวละครที่กำหนด สัมผัสกับสีที่ต้องการหรือไม่
	ถามคำถามที่กำหนด แล้วรอคำตอบ
	คำตอบที่ผู้ใช้กรอก จากคำถาม ask (สามารถคลิกที่ช่องเพื่อแสดงคำตอบบนเวที)
	ตำแหน่งตัวชี้เมาส์ตามแนวแกน x
	ตำแหน่งตัวชี้เมาส์ตามแนวแกน y
	ตรวจสอบเมาส์ว่าถูกคลิกหรือไม่
	ตรวจสอบการกดแป้นพิมพ์บนคีย์บอร์ด
	ระยะทางของตัวชี้เมาส์หรือตัวละคร
	เริ่มเวลาใหม่ เป็น 0
	เวลา
	ข้อมูลตัวละคร เช่น ตำแหน่งตามแนวแกน x, ตำแหน่งตามแนวแกน y, มุม, ลำดับชุดตัวละคร, ขนาด, ระดับความดังของเสียง
	ระดับเสียง (สามารถคลิกที่ช่องเพื่อแสดงค่าระดับของเสียงบนเวที)
	ตรวจสอบว่ามีเสียงหรือไม่
	ค่าที่ได้จากอุปกรณ์ตรวจจับต่างๆ
	ตรวจสอบการกดปุ่ม หรือการเชื่อมต่ออุปกรณ์ตรวจจับ



การสั่งงานแบบวนซ้ำ

การสั่งงานแบบวนซ้ำใช้สำหรับเขียนสคริปต์ที่สั่งให้การทำงานซ้ำกันไปมาหลายๆ รอบ ให้พิจารณาสคริปต์ที่สั่งให้ตัวละครเดินบนเวทีเป็นรูปสี่เหลี่ยม โดยเคลื่อนที่ไปข้างหน้า 50 ก้าว จากนั้นหมุนทิศทวนเข็มนาฬิกาไป 90 องศา ทำแบบนี้ซ้ำกัน 4 รอบดังนี้



สคริปต์ดังกล่าวมีบล็อกที่ใช้คำสั่งซ้ำกันจำนวนมาก ทำให้เสียเวลาในการแก้ไข เช่น ถ้าต้องการให้ตัวละครเดินเป็นรูปสี่เหลี่ยมที่กว้างขึ้น ก็ต้องมีการแก้ไขสคริปต์ถึง 4 จุด ถ้าใช้การทำงานแบบวนซ้ำ จะได้สคริปต์ที่สั้นลง และทำความเข้าใจได้ง่ายขึ้นดังนี้



การควบคุมลำดับการทำงานแบบวนซ้ำ มีหลายลักษณะ ตามแต่ละชนิดของบล็อกควบคุม เช่น

บล็อก	คำอธิบาย	ตัวอย่าง	
	การทำงานซ้ำไม่รู้จบ	เดินไปข้างหน้าครึ่งละ 10 หน่วย เมื่อชนขอบเวทีให้เดินกลับ ทำแบบนี้เรื่อยๆ ไม่รู้จบ	
	การทำงานซ้ำตามจำนวนรอบที่กำหนด	เดินไปข้างหน้า 50 หน่วย หันขวา 36 องศา ทำแบบนี้ซ้ำ 10 รอบ	



การทำงานแบบเงื่อนไข

การทำงานแบบมีเงื่อนไข (Condition)

เงื่อนไขใช้การควบคุมทิศทางการทำงาน เพื่อตัดสินใจในการทำงานอย่างใดอย่างหนึ่ง บล็อกเงื่อนไขมี 2 แบบ ดังนี้

บล็อก	คำอธิบาย	ตัวอย่าง
	ใช้ตรวจสอบเงื่อนไข ถ้าเงื่อนไขหลัง if เป็นจริง จึงจะทำคำสั่งภายในบล็อก if	 ถ้า ตำแหน่งตัวละครบนเวที อยู่แนวแกน x = 0 แล้ว จะเล่นเสียงแมว
	ใช้ตรวจสอบเงื่อนไข ถ้าเงื่อนไขหลัง if เป็นจริง จะทำคำสั่งภายในบล็อก if แต่ถ้าเงื่อนไขเป็นเท็จ จะทำคำสั่งภายในบล็อก else	 ถ้า ความดังของเสียงเท่ากับ 100% แล้ว จะเล่นเสียงแมว มิฉะนั้น จะเล่นเสียงนก

การวนซ้ำแบบมีเงื่อนไข (Loop)

การทำงานแบบวนซ้ำโดยมีเงื่อนไข ใช้ในการควบคุมการทำงานซ้ำ ตามเงื่อนไขที่กำหนด โดยจะมีการตรวจสอบเงื่อนไขก่อนการทำงานทุกครั้ง แบ่งเป็น 2 ลักษณะ คือ การทำซ้ำเมื่อเงื่อนไขเป็นจริง (forever if) และการทำซ้ำจนกว่าเงื่อนไขจะเป็นจริง (repeat until)

บล็อก	คำอธิบาย	ตัวอย่าง
	ทำคำสั่งภายในบล็อกซ้ำไปเรื่อยๆ ถ้าเงื่อนไขหลัง if เป็นจริง	 เดินไปข้างหน้าทีละ 10 หน่วย ตรวจจับที่มีการกดแป้น →
	ทำคำสั่งภายในบล็อกซ้ำไปเรื่อยๆ จนกระทั่งเงื่อนไขหลัง until เป็นจริง	 เดินไปข้างหน้าทีละ 10 หน่วย จนกว่าจะมีการกดแป้น space bar




นอกจากนี้ ยังมีบล็อกที่เป็นคำสั่งใช้งานเฉพาะ โดยมีการตรวจสอบเงื่อนไข เช่น บล็อก wait until

บล็อก	คำอธิบาย	ตัวอย่าง
	หยุดรอเวลา จนกระทั่งเงื่อนไขหลัง until เป็นจริง จึงจะทำคำสั่งต่อไป	 หยุดรอเวลา จนกว่าขนาดของตัวละครจะเท่ากับ 50% แล้วทำคำสั่งต่อไป



การใช้ตัวดำเนินการ

ในการคำนวณใดๆ จะต้องมีการระบุตัวดำเนินการ (Operator) เพื่อสั่งให้คำนวณ เปรียบเทียบ ในโปรแกรม Scratch มีกลุ่มบล็อก Operators ให้ใช้งาน ซึ่งประกอบด้วยบล็อกที่ใช้ในการคำนวณพื้นฐาน การเปรียบเทียบ ตรรกะ และฟังก์ชันในการทำงานต่างๆ ในบล็อก Operators จะมีโครงสร้างที่ประกอบไปด้วย ช่องกำหนดค่า ตัวดำเนินการหรือคำสั่ง ซึ่งช่องกำหนดค่า สามารถใส่ตัวเลข อักขระ หรือลากบล็อกอื่นๆ มาวางบนบล็อกก็ได้ โดยกำหนดค่าดังนี้

-  ช่องสี่เหลี่ยมมุมมนรับค่าเป็นตัวเลขและรับบล็อกสี่เหลี่ยมมุมมน
-  ช่องหกเหลี่ยมรับได้เฉพาะข้อมูลที่เป็นจริงหรือเท็จจากบล็อกหกเหลี่ยมเท่านั้น
-  ช่องสี่เหลี่ยมรับข้อมูลเป็นอักขระ ซึ่งถ้าใส่ตัวเลขในช่องนี้จะถือว่าตัวเลขนั้นเป็นอักขระ นอกจากนี้ยังรับบล็อกสี่เหลี่ยมมุมมนและบล็อกหกเหลี่ยมได้ด้วย

1. บล็อก Operators คำนวณพื้นฐาน

ภายในบล็อก Operators คำนวณพื้นฐาน มีตัวดำเนินการต่างๆ ที่ใช้ในการคำนวณ เช่น บวก ลบ คูณหาร มอดุลัส ตัวอย่างการใช้งานเช่น

บล็อก	ความหมาย	ตัวอย่างการใช้งาน	ผลลัพธ์
	บวก		9
	ลบ		5
	คูณ		14

บล็อก	ความหมาย	ตัวอย่างการใช้งาน	ผลลัพธ์
	หาร		3.5
	มอดุล		1

หมายเหตุ มอดุลัส คือ การหาเศษที่เหลือที่น้อยที่สุดจากการหารและไม่ใช่จำนวนลบ เช่น $5 \text{ mod } 2$
 ($5 \text{ หาร } 2 = 2$ เหลือเศษ 1) ผลลัพธ์ที่ได้คือ 1

ในกรณีที่ต้องการใช้ตัวดำเนินการมากกว่าหนึ่งตัว สามารถทำได้โดยนำบล็อกที่ต้องการมาวางบนบล็อก Operators ดังตัวอย่างต่อไปนี้

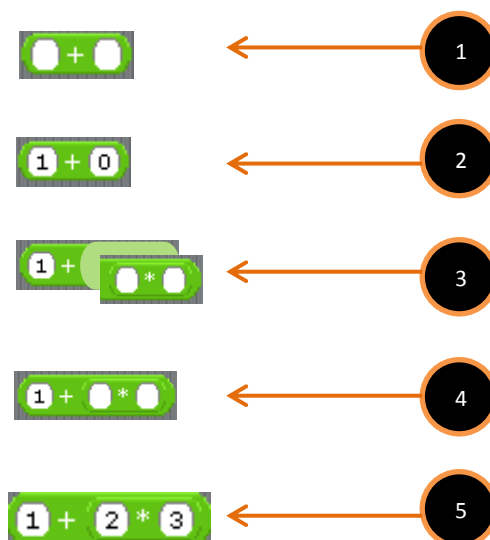
ตัวอย่างที่ 1 $1 + (2 * 3)$

ลำดับการคำนวณ คือ

- 1) คำนวณค่าในวงเล็บก่อน โดยนำ 2 มาคูณกับ 3 ซึ่งมีค่าเท่ากับ 6
- 2) นำผลลัพธ์ที่ได้ข้อ 1 มาบวกกับ 1 จะได้ $6 + 1 = 7$

ในโปรแกรม Scratch สามารถทำได้ดังนี้

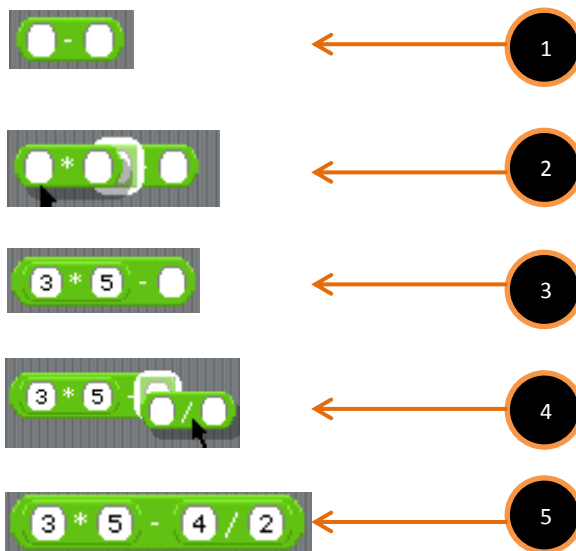
- 1) ลากบล็อกบวกมาวาง
- 2) พิมพ์เลข 1 ลงในช่องกำหนดค่าด้านซ้าย
- 3) ลากบล็อกคูณมาวางลงในช่องกำหนดค่าด้านขวา
- 4) จะได้บล็อกคูณซ้อนอยู่ในบล็อกบวก
- 5) พิมพ์เลข 2 และ 3 ลงในช่องกำหนดค่าของบล็อกคูณ



หมายเหตุ การคำนวณค่าในบล็อกจะคำนวณในบล็อกนั้นๆ ก่อน เหมือนกับการคำนวณในวงเล็บก่อน แล้วจึงคำนวณส่วนที่เหลือภายหลัง เช่น $(3*5)-(4/2) = 15-2 = 13$

ตัวอย่างที่ 2 $(3*5)-(4/2)$ สามารถกำหนดในโปรแกรม scratch ได้ดังนี้

- 1) ลากบล็อกลงมาวาง
- 2) ลากบล็อกคูณ มาวางในช่องกำหนดค่าด้านซ้ายของบล็อกลบ จะได้บล็อกคูณซ้อนอยู่ในบล็อกลบ
- 3) พิมพ์เลข 3 และ 5 ลงในช่องกำหนดค่า ตามลำดับ
- 4) ลากบล็อกหาร มาวางในช่องกำหนดค่าด้านขวาของบล็อกลบ จะได้บล็อกหารซ้อนอยู่ในบล็อกลบ
- 5) พิมพ์เลข 4 และ 2 ลงในช่องกำหนดค่า ตามลำดับ



2. บล็อก Operators เปรียบเทียบ

การใช้งานบล็อก Operators เปรียบเทียบ สามารถใส่ค่าหรือลากบล็อกที่ต้องการเปรียบเทียบลงในช่องกำหนดค่า ผลลัพธ์จากการใช้ตัวดำเนินการเปรียบเทียบจะได้ค่าเป็นจริงหรือเท็จ ตัวอย่างเช่น

บล็อก	ความหมาย	ตัวอย่างการใช้งาน	ผลลัพธ์
	น้อยกว่า		เท็จ
	มากกว่า		จริง
	เท่ากัน		เท็จ

3. บล็อก Operators ฟังก์ชัน

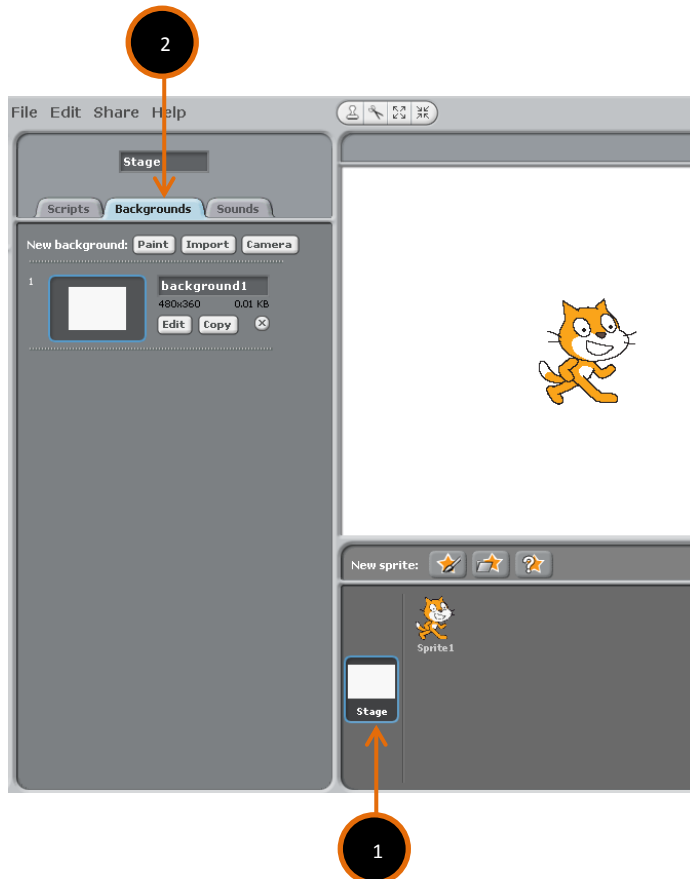
บล็อก Operators ฟังก์ชัน มีบล็อกที่จัดการตัวอักษรและข้อความ และบล็อกที่คำนวณทางคณิตศาสตร์ ตัวอย่างเช่น

บล็อก	ความหมาย	ตัวอย่างการใช้งาน	ผลลัพธ์
	เชื่อมข้อความในช่อง กำหนดค่าที่ 1 และ 2		Hello Joy
	เลือกอักขระในลำดับที่ ระบุจากค่าที่กำหนด		m
	นับจำนวนอักขระ		8
	ปัดเศษ		6
			4
	สุ่มตัวเลข		ตัวเลขที่สุ่มได้มีค่า ตั้งแต่ 1 ถึง 10
	ฟังก์ชันทางคณิตศาสตร์		รากที่สองของ 25 คือ 5

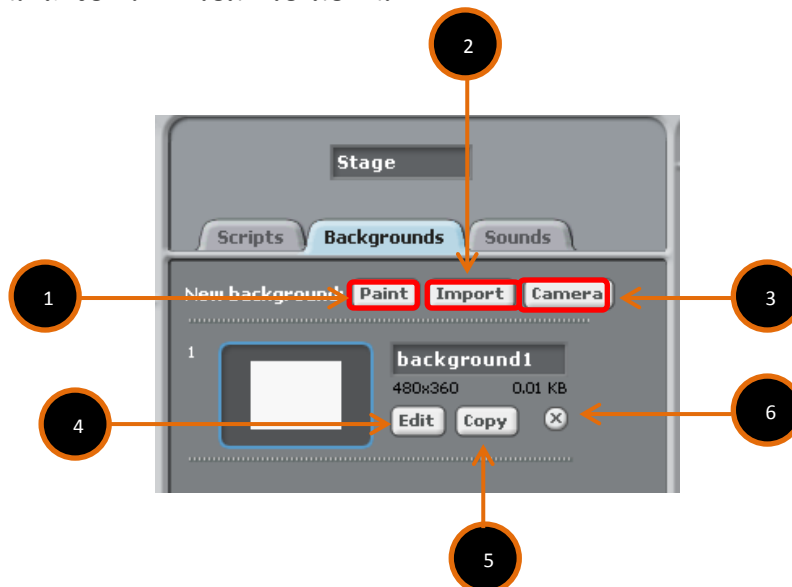


เดินเล่นชมวิวยสวย

1. สร้างโปรเจกต์ใหม่
2. เลือกพื้นหลังของเวทีดังนี้
 - ❶ คลิกที่ Stage
 - ❷ คลิกแท็บ Backgrounds



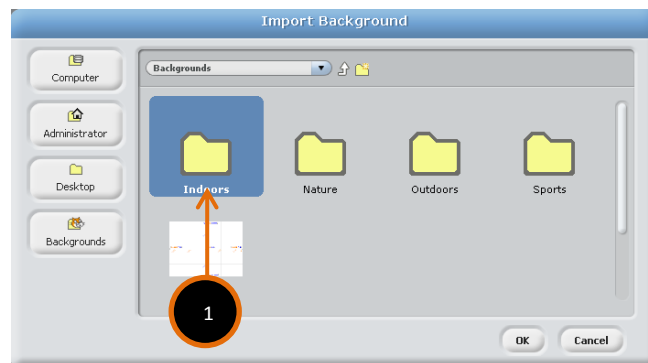
จะปรากฏพื้นหลังของเวที พร้อมเครื่องมือดังนี้



- ❶ ปุ่ม **Paint** วาดพื้นหลังขึ้นมาใหม่
- ❷ ปุ่ม **Import** เลือกพื้นหลังจากไฟล์ภาพ
- ❸ ปุ่ม **Camera** ถ่ายภาพพื้นหลังขึ้นมาใหม่ (ต้องใช้อุปกรณ์จับภาพ)
- ❹ ปุ่ม **Edit** แก้ไขพื้นหลังโดย Paint Editor
- ❺ ปุ่ม **Copy** คัดลอกพื้นหลัง
- ❻ ปุ่ม **X** ลบพื้นหลัง

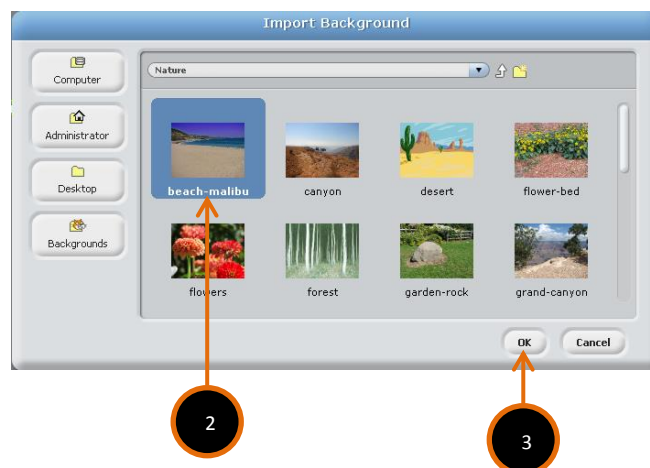
3. เลือกพื้นหลังจากไฟล์เตอร์ภาพ โดยคลิกปุ่ม **Import** จะปรากฏหน้าต่าง Import Background จะมีไฟล์เตอร์ภาพต่างๆ ในที่นี้ให้เลือกภาพ beach-malibu จากไฟล์เตอร์ Nature ดังนี้


- ❶ ดับเบิลคลิกที่ไฟล์เตอร์ Nature จะปรากฏภาพต่างๆ ที่เป็นภาพธรรมชาติ



- ❷ คลิกที่ภาพ beach-malibu

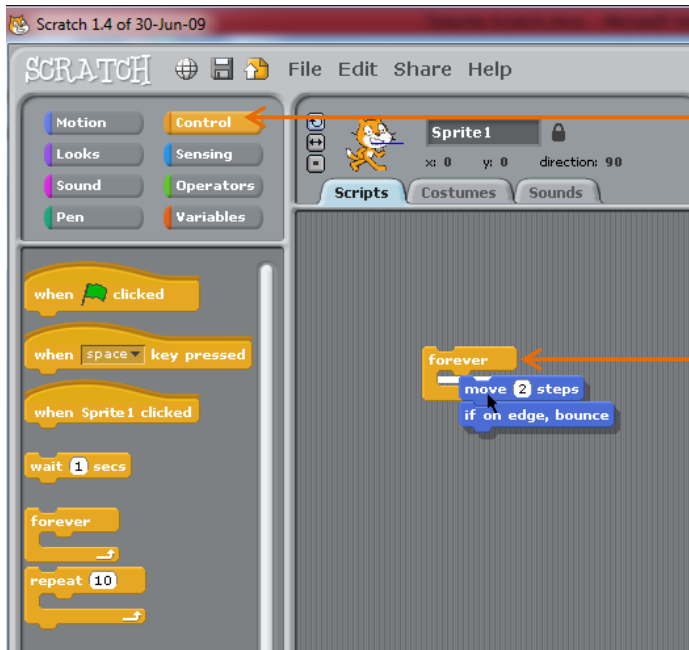
- ❸ คลิกปุ่ม **OK** ภาพพื้นหลังที่เลือกจะปรากฏบนเวที และรายการพื้นหลัง



4. ลบพื้นหลัง background1 โดยคลิกปุ่ม  สิ่งที่น่าประหลาดคือ.....
5. คลิกที่ตัวละคร Sprite1
6. เขียนสคริปต์ให้ตัวละคร Sprite1 ดังนี้

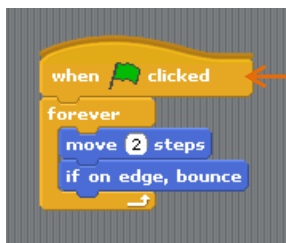


1 เขียนสคริปต์



2 คลิกปุ่มบล็อก Control

3 ลากบล็อก forever คลุม



4 ลากบล็อก when  clicked ต่อด้านบน



คลิก  ผลลัพธ์คือ

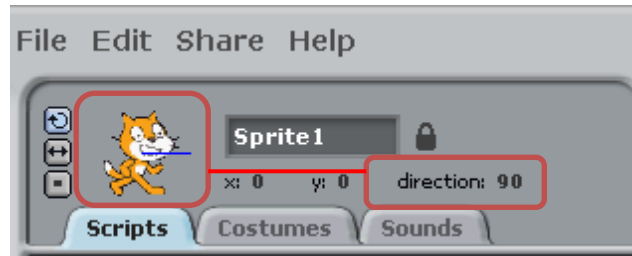
.....

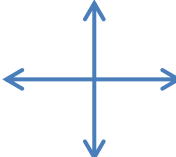
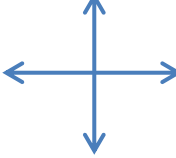
.....

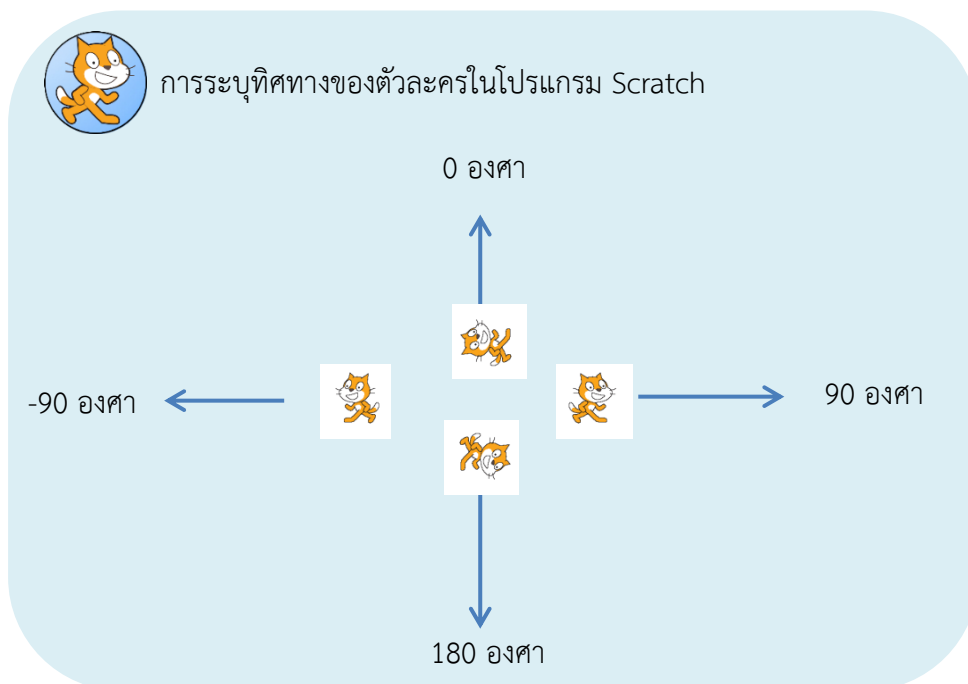
.....

7. ทดลองคลิกปุ่มรูปแบบการหมุนทั้ง 3 ลักษณะ    แล้วสังเกตผลลัพธ์ จากนั้นวงกลมล้อมรอบรูปตัวละครที่เป็นผลลัพธ์หลังชนขอบเวที

8. คลิกปุ่ม  เพื่อหยุดการเคลื่อนที่ของตัวละคร และเลือกปุ่ม  ให้ตัวละครหมุนได้รอบทิศทาง
9. หมุนเส้นสีน้ำเงินที่ปรากฏที่ข้อมูลของตัวละครที่ถูกเลือก (Sprite1) เพื่อกำหนดทิศทางการเคลื่อนที่ของตัวละคร (direction) ดังทิศทางที่ระบุในข้อ 9.1 และ 9.2 จากนั้นให้วาดเส้นทิศทางของตัวละครตามที่ปรากฏผลลัพธ์



ทิศทาง	ผลลัพธ์
9.1 -45 องศา	
9.2 75 องศา	



10. บันทึกโปรเจกต์ชื่อ MyFirstProject ดังนี้

- ❶ คลิกเมนู File -> Save หรือ Save As จะปรากฏกรอบโต้ตอบ Save Project
- ❷ พิมพ์ชื่อผู้ทำโปรเจกต์
- ❸ พิมพ์รายละเอียดเกี่ยวกับโปรเจกต์
- ❹ พิมพ์ชื่อโปรเจกต์
- ❺ คลิกปุ่ม ok
- ❻ โปรเจกต์จะถูกบันทึกไว้ที่ C:\Documents and Setting\Administrator\
My Documents\Scratch Projects (โฟลเดอร์ My Projects)

