



สนุกกับการ Coding ด้วย

SCRATCH 3

(Primary Level)



สามารถเขียนโปรแกรมสร้างเกม นิทาน เรื่องเล่าแบบโต้ตอบได้ สร้างแอนิเมชันด้วยตัวละคร ภาพ เสียง ถนนี่ และวิดีโอได้ครบ

ส่งเสริมให้เด็กๆ เรียนรู้ที่จะคิดแบบสร้างสรรค์ จินตนาการ จัดความคิดอย่างเป็นระบบ และการทำงานร่วมกัน

เหมาะกับผู้ที่ต้องการเริ่มต้นเขียนโปรแกรม สร้างโค้ด ทำได้ง่ายและสนุก ไม่มีพื้นฐานก็ทำตามได้

อธิบายแบบ Step by Step ภาพสี่สีสวยงาม พร้อมตัวอย่างมากมายให้สร้างเสริมครบทุกเรื่องที่ควรรู้



ดาวน์โหลด WS <https://serazu.com/> 9786164871847



มีเพียง “ความรู้” เท่านั้นที่มนุษย์ใช้พลิก “โลก”
และเปลี่ยนชีวิต เราจึงสร้างสรรค์ และส่งมอบ “ความรู้”
ในรูปแบบที่ดีกว่า เพื่อให้คนไทย “เรียนรู้” ได้ตลอดชีวิต

Only “Knowledge” can help human
change “The World” and “Their Lives”.
With this truth, it drives us to deliver
“Knowledge” for Thai being able to
“Learn” better everyday.



Think
Beyond



สนุกกับการ Coding ด้วย Scratch 3.0 (Primary Level)

Writer	ดวงพร เกียงคำ
Editor	พิชญ์ ประศิริ pitsanu_p@idcpremier.com
Graphic Designers	ชวนันท์ รัตนะ, สิริลักษณ์ วาระเลิศ
Page Layout	จตุรงค์ ศรีวิลาศ
Proofreaders	สุนทรี บรรลือศักดิ์, เกษรา พรวัฒนมงคล
Publishing Coordinators	วรพล ณิชกุล, สุพัตรา อาจปฐ, พิพัฒน์ ออสถิตย

MapReduce เป็นเครื่องหมายการค้าของบริษัท Google, Hadoop เป็นเครื่องหมายการค้าของบริษัท Yahoo, Apache Hive เป็นเครื่องหมายการค้าของบริษัท Facebook, Spark เป็นเครื่องหมายการค้าของมหาวิทยาลัย California และเครื่องหมายการค้าอื่นๆ ที่อ้างถึงเป็นของบริษัทนั้นๆ

บริษัท ไอทีซี พรีเมียร์ จำกัด จัดตั้งขึ้นเพื่อเผยแพร่ความรู้ที่มีคุณภาพสู่ผู้อ่านชาวไทย เรายินดีรับงานเขียนของนักวิชาการและนักเขียนทุกท่าน ท่านผู้สนใจกรุณาติดต่อผ่านทางอีเมลที่ infopress@idcpremier.com หรือทางโทรศัพท์หมายเลข 0-2962-1081 (อัตโนมัติ 10 คู่สาย) โทรสาร 0-2962-1084

สร้างสรรคโดย



พิมพ์ครั้งที่ 1 ธันวาคม 2563

ข้อมูลทางบรรณานุกรม

ดวงพร เกียงคำ
สนุกกับการ Coding ด้วย Scratch 3.0 (Primary Level)
นนทบุรี : ไอทีซีฯ, 2563
272 หน้า
1. การเขียนโปรแกรมโดยใช้ภาษาเฉพาะชนิด
I ชื่อเรื่อง
005.262
ISBN 978-616-487-184-7

ราคา 275 บาท

จัดพิมพ์และจัดจำหน่ายโดย



บริษัท ไอทีซี พรีเมียร์ จำกัด
200 หมู่ 4 ชั้น 19 ห้อง 1901
อาคารจัสมินอินเตอร์เนชั่นแนลทาวเวอร์
ถ.แจ้งวัฒนะ อ.ปากเกร็ด จ.นนทบุรี 11120
โทรศัพท์ 0-2962-1081 (อัตโนมัติ 10 คู่สาย)
โทรสาร 0-2962-1084
ลูกค้าสัมพันธ์
โทรศัพท์ 0-2962-1081-3 ต่อ 121
โทรสาร 0-2962-1084
ร้านค้าและตัวแทนจำหน่าย
โทรศัพท์ 0-2962-1081-3 ต่อ 112-114
โทรสาร 0-2962-1084

PREFACE

เด็ก ๆ ในยุคของ Gen Z ที่เติบโตมาพร้อมกับความสะดวกสบายมากมายรอบตัว มีเทคโนโลยีที่ทันสมัยสามารถเรียนรู้ได้เร็ว มีพฤติกรรมการใช้โทรศัพท์มือถือ แท็บเล็ต คอมพิวเตอร์ในตึกบู๊คที่มากพร้อมกับการใช้งานอินเทอร์เน็ต มีตัวตนอยู่ในโซเชียลเน็ตเวิร์ค เสพติดการเชื่อมต่อและออนไลน์ จนมาถึงยุคไวรัสโควิด-19 ระบาด ต้องอยู่ที่บ้านเรียนหนังสือแบบออนไลน์ ทำให้ชีวิตเข้าไปอยู่ในโลกออนไลน์กันมากยิ่งขึ้น เราจะเห็นว่าเด็ก 4 ขวบสามารถใช้มือถือ แท็บเล็ต ในการสร้างการเรียนรู้ เล่นเกม หรือหาความบันเทิงในด้านต่างๆ ได้อย่างคล่องแคล่ว

ด้วยความที่เทคโนโลยีมีการเปลี่ยนแปลงตลอดเวลา บุคลากรสายงาน Programmer & Developer จะเนื้อหอมเป็นที่ต้องการของบริษัทต่างๆ เพราะธุรกิจมีการเร่งพัฒนาระบบงานต่างๆ ไม่ว่าจะเป็นเว็บไซต์ และแอปพลิเคชันที่รองรับการใช้งานบนสมาร์ตโฟน ทุกแพลตฟอร์ม เด็กหลายคนสนใจอยากเข้าไปมีส่วนร่วมคือ การนำเทคโนโลยีเหล่านี้มาเป็นนักสร้างเกม สร้างแอป เป็นนักโปรแกรมเมอร์ นักเขียนโปรแกรม ก็ต้องเรียนรู้การเขียนโปรแกรมด้วยภาษาคอมพิวเตอร์ ซึ่งจะต้องเริ่มจากพื้นฐานการใช้งานง่ายๆ แล้วพัฒนาไปถึงการเขียนโค้ด (Coding) ภาษายากๆ

เด็ก ๆ ที่ต้องการเขียนโปรแกรมสามารถเริ่มต้นการเขียนโปรแกรมด้วย Scratch ที่วางพื้นฐานด้วยการใช้บล็อกสร้างชิ้นงานได้อย่างง่าย เช่น นิทานที่สามารถโต้ตอบกับผู้อ่านได้ ภาพเคลื่อนไหว เกม ดนตรี และศิลปะ มีการเรียนรู้หลักการและแนวคิด และการคิดอย่างสร้างสรรค์ไปตามตรรกะ (Logic) และเหตุผลตามหลักของการพัฒนาโปรแกรมมาตรฐานด้วยภาษาต่างๆ แต่ Scratch จะให้เด็ก ๆ เขียนโปรแกรมด้วยการนำบล็อก (Block) คำสั่งมาเรียงต่อกันเหมือนการต่อจิ๊กซอว์ เพื่อสร้างสรรค์ชุดคำสั่งให้ตัวละครทำงานตามคำสั่งเป็นไปตามลำดับขั้นตอนและเงื่อนไขที่กำหนด

หนังสือ “สนุกกับการ Coding ด้วย Scratch 3.0 (Primary Level)” เล่มนี้ เด็กๆ อายุตั้งแต่ 6 ขวบขึ้นไปสามารถเรียนรู้และใช้งานได้ จะมีเมมูภาษาไทยและภาษาอังกฤษ พร้อมกับตัวอย่างงานและภาพประกอบที่สวยงาม อธิบายตามลำดับขั้นตอน ผู้ปกครองก็สามารถเรียนรู้และช่วยสอน หรือชี้แนะบุตรหลานเพิ่มเติม เพราะ Scratch เป็นโปรแกรมที่เยาวชนมากกว่า 140 ประเทศทั่วโลกนำมาสร้างพื้นฐานความรู้ เพื่อจุดประกายความรู้ให้เด็ก ๆ ได้เข้าไปสัมผัสกับการเขียนโปรแกรมคอมพิวเตอร์ว่า ทำได้ง่ายไม่ยากเลยคะ ลองนำหนังสือเล่มนี้มาศึกษาก่อนได้

หนังสือเล่มนี้เป็นการเล่นโปรแกรมพื้นฐานเริ่มต้นสำหรับเด็ก กลุ่มเป้าหมายหลักเป็นเด็ก ๆ ซึ่งถือว่าเป็นงานหินเหมือนกัน เพราะไม่ได้เขียนหนังสือเด็ก หรือสอนเด็ก ๆ มานานมาก จำได้เมื่อนานเกือบ 30 ปี เคยสอนเด็กเขียนโปรแกรมด้วยภาษา “Basic” โอ้ยยย... กว่าที่จะเขียนข้อความ 1 ประโยคแสดงในหน้าจอ ต้องเขียนโค้ดกันเป็น 10 บรรทัด คำว่า “Hello World” เป็นข้อความที่นักโปรแกรมเมอร์รุ่นเก๋ารู้จักกันดี เพราะเป็นข้อความฮิตที่ใช้แสดงตัวอย่างการแสดงผลข้อความ ตอนที่น้องหนึ่ง บ.ก. ส่งข้อความมาถามว่าทำไมต้องอยากเขียนหนังสือ Scratch มั้ย เขียนโปรแกรมสำหรับเด็ก ก็รีบตอบไปว่าไม่ได้มั้ง แต่พอได้มาดูคอนเซ็ปต์และลองเล่นดู เออน่ารักดีนะ ด้วยความที่ตัวเองเป็นคนชอบตัวการ์ตูน ทำอะไรที่เป็นการ์ตูนเหมือนเด็ก ๆ เลยตอบตกลงเขียน เขียนไปเขียนมาก็สนุกดี ต้องสร้างสรรค์จินตนาการไปแบบเด็ก ๆ ลดอายุไปได้หน่อย (^-^)

ขอขอบคุณทางสำนักพิมพ์ไอทีซี พรีเมียร์ ที่มีแนวคิดในการจัดทำหนังสือเล่มนี้ ขอขอบคุณน้องหนึ่ง (คุณพิษณุ ประศิริ) ที่ชักชวนให้ทำและเป็น บ.ก. ตรวจสอบหนังสือเล่มนี้ หวังว่าหนังสือเล่มนี้จะมีประโยชน์ต่อเด็ก ๆ เพื่อนำมาเป็นพื้นฐานการเริ่มต้นเข้าสู่การเป็นนักพัฒนาโปรแกรมในอนาคต อ่านหนังสือแล้วก็ติดตามคลิบสอน Scratch เพิ่มเติมได้ที่ YouTube Channel นะคะ

ดวงพร เกียงคำ (ต้อย)

ช่องทางติดต่อนักเขียน

Facebook : Duangporn Kiengkam

E-Mail : Duangporn.toy@gmail.com

YouTube Channel : Duangporn Toy





LESSON 1

เริ่มต้นกับ Scratch	1
ดาวน์โหลดและติดตั้งแอป Scratch	2
วิธีที่ 1 : ดาวน์โหลดจากเว็บไซต์	2
ดาวน์โหลด/ติดตั้งจากเว็บไซต์ Microsoft ผ่านบราวเซอร์	3
ดาวน์โหลด/ติดตั้งจากแอป Store บน Windows 10	3
ใช้งาน Scratch ผ่านบราวเซอร์ (Online Editor)	5
หน้าต่างแอป Scratch บนเดสก์ท็อป (Desktop)	7
แท็บการทำงานพื้นฐานของ Scratch	8
Stage : เวที	10
เพิ่มสไปรต์ (Add New Sprite)	11
ตรวจสอบตัวละครและคอสตุ่ม	13
ปรับแต่งคุณสมบัติของสไปรต์	14
Size : กำหนดขนาดสไปรต์	14
Direction : กำหนดทิศทางสไปรต์	15
Position : ตำแหน่งของสไปรต์บนเวที	15
การจัดการสไปรต์	16
Move : ย้ายตำแหน่งสไปรต์	16
Show/Hide : ซ่อน/แสดงสไปรต์	16
Rename : เปลี่ยนชื่อสไปรต์	17
Delete Sprite : ลบสไปรต์	17
เพิ่มฉากหลัง (Add Backdrop)	18
Upload Backdrop : อัปโหลดภาพฉากหลัง	19
สลับการใช้งาน Backdrop	20
วาดฉากหลังขึ้นใหม่	21
ตัวอย่างการใส่โค้ดเพื่อเปลี่ยนฉากหลัง	24
หน้าต่างการเพิ่ม Backdrop ภาษาไทย	25
ศึกษาเรียนรู้จากบทเรียน (Tutorials)	26

สำรวจตัวอย่างโปรเจกต์ (Project)	27
การใช้งานบทเรียน (Tutorials)	28
เข้าร่วมในชุมชนออนไลน์	29
ตัวอย่างการอัปเดตผลงานขึ้นสู่ Studio ตามชื่อบัญชี	32
Save to your computer : บันทึกลงในคอมฯ	33

LESSON 2

Sprite Costume ออกแบบตัวละคร	35
Sprite Costume (สไปรต์คอสตูม)	36
การเลือกคอสตูมเมนูภาษาไทย	37
คอสตูมการแต่งกายของตัวละคร	38
Duplicate Sprite : ทำซ้ำหรือก๊อปปี้สไปรต์	39
เพิ่มคอสตูม (Choose a costume)	40
ปรับแต่งดัดแปลงคอสตูมเดิม	41
Flip Horizontal : พลิกกลับด้านคอสตูมแนวนอน	41
Flip Vertical : พลิกกลับด้านคอสตูมแนวตั้ง	41
Copy/Paste : คัดลอกและวางคอสตูม	42
ปรับขนาด และหมุนคอสตูม	43
การเลือกพื้นที่ในคอสตูม (Select)	44
คลิกลากคลุมเลือกอบเจกต์	45
Group/Ungroup : จัดกลุ่มแยกกลุ่มรูปร่าง	46
เทคนิคการเปลี่ยนสีคอสตูม (Color)	48
Fill Color : เลือกสีพื้น	48
Outline Color : กำหนดสีเส้น	49
เปลี่ยนสีดำ (Black Color)	50
Delete : ลบคอสตูม	51
Undo : ยกเลิกการกระทำ หรือยกเลิกความผิดพลาด	51
เทคนิคการสร้างคอสตูมใหม่	52
การอัปเดตคอสตูมจากรูปภาพที่มี	56



ใส่ข้อความลงในคอสตุ่ม	58
การใช้ Effect (เอฟเฟกต์) กับตัวละคร	59
ตัวอย่างการเปลี่ยนสีตัวละคร	60

LESSON 3

Scratch Blocks การวางบล็อกโค้ด	61
ภาษา Scratch ทำงานอย่างไร	62
หลักการเขียนโค้ด	63
แท็บ Code : แท็บโค้ดในการเขียนโปรแกรม	64
Motion : เคลื่อนไหว	65
Looks : รูปลักษณ์	65
Sound : เสียง	66
Events : เหตุการณ์	66
Control : ควบคุม	67
Sensing : ตรวจสอบ	68
Operators : ตัวดำเนินการ	69
Variables : ตัวแปร	70
My Blocks : บล็อกของฉัน	70
Extension : ส่วนขยายของโปรแกรม	71
Go/Stop : รันโค้ดและหยุดการรัน	75
วิธีการใส่โค้ด	76
เทคนิคการทดสอบโค้ดก่อนวางบล็อก	77
หาตำแหน่ง x, y บนเวที	77
เปิดดูตัวเลือกก่อนวางบล็อก	78
Blocks : บล็อกโค้ดกับการเขียนโปรแกรม	79
เริ่มต้นเขียนโปรแกรม	80
Control : ใส่บล็อกควบคุมการทำงาน	82
การจัดการบล็อกโค้ด	84
Delete : ลบบล็อกโค้ด	84

คลิกลากทิ้ง เพื่อยกเลิกบล็อกโค้ด	84
สลับทำแหน่งและยกเลิกบล็อก	85
ขยับ/ย้ายตำแหน่งบล็อก	85
บล็อกที่มีตัวเลือก (Options)	87
การใช้งานบล็อกที่ซ้อนกัน และเกี่ยวข้องกับสัมพันธ์กัน	87
กำหนดขนาดหน้าจอทำงาน	88

LESSON 4

Coding การเขียนโค้ด	89
หลักการทำให้ภาพเคลื่อนไหว	90
ประเภทของภาพเคลื่อนไหว	90
สร้างภาพเคลื่อนไหวอย่างง่าย	91
เพิ่มโค้ดให้เดินหลายๆ รอบ	92
การสลับล็อกของตัวละคร (Sprite)	93
การเคลื่อนที่ตามทิศทาง (Direction)	95
ใช้บล็อก point in direction	97
การหัน (turn) ไปตามองศาที่กำหนด	98
การเคลื่อนที่แบบสุ่ม (Random Position)	99
บล็อกที่ใช้งานตำแหน่ง x, y บนเวที	101
เลือกตัวแปรแสดงตำแหน่ง x, y และ direction บนเวที	101
การหาตำแหน่ง x, y แบบง่ายๆ	102
ตัวอย่างการสร้างภาพเคลื่อนไหวด้วย go to x, y	103
การเคลื่อนที่แบบ glide (เหิน)	104
ตัวอย่างการเคลื่อนไหวและเปลี่ยนสีตัวละคร	105
กำหนดให้ตัวละครเคลื่อนที่ไปรอบเวที	107
เพิ่มตัวควบคุม forever	107
การเคลื่อนไหวไปตามตัวชี้เมาส์	108
การใช้ go to mouse-pointer	108



สร้างการเคลื่อนไหวด้วย point towards mouse-pointer	110
if on edge, bounce : เมื่อชนขอบเวทีให้แดงกลับ	111
การเคลื่อนที่ด้วยการกดปุ่มบนแป้นพิมพ์	112
การตรวจสอบการกดปุ่มบนแป้นพิมพ์ (key pressed)	114
การเคลื่อนที่และปรับขนาด (Size)	116
การเปลี่ยนฉากหลัง (Backdrop)	117
เพิ่มตัวละครและฉากหลัง	117
การเปลี่ยนฉากหลังอัตโนมัติ	119
การเปลี่ยนฉากหลังที่ซับซ้อน	120
กำหนดโค้ดการเคลื่อนที่ และเปลี่ยนฉากหลัง	120
การกระจายข้อความ (broadcast)	122
กำหนดโค้ดการรับข้อความที่ส่งมาให้กับตัวละครที่ 2	123

LESSON 5

broadcast กระจายข้อความ	125
การส่งข้อความบรอดคาสต์	126
broadcast : การสร้างข้อความที่ต้องการแจ้งออกไป	126
when I receive : รับข้อความที่ broadcast มา	127
ตัวอย่างการส่งข้อความ และรอตอบกลับ (broadcast and wait)	128
เทคนิคการกระจายข้อความง่ายๆ	129
สร้างบทการโต้ตอบระหว่างตัวละคร	132
Variables : สร้างตัวแปรเก็บคำตอบ	133
ask : สร้างคำถาม และรับคำตอบจากผู้ใช้	133
join : เชื่อมข้อความจากตัวแปร และ say	133
ตรวจสอบโค้ดและผลลัพธ์	134
สร้างการหักทหายจากตัวละคร	136
เทคนิคการปรับคออสตัม	138

LESSON 6

Events & Control ควบคุมการทำงาน	139
Events : เหตุการณ์	140
บล็อกควบคุมการทำงานของโปรแกรม	143
wait : หน่วงเวลา หรือให้รอก่อนทำงานอื่นต่อ	144
wait until : ให้หยุดรอจนกระทั่งตรงกับเงื่อนไข	145
repeat : การทำงานแบบวนซ้ำ (Loop)	147
forever : วนซ้ำตลอด	148
repeat until : ทำซ้ำจนเจอเงื่อนไขที่เป็นจริง	149
if : ให้ทำงานตามเงื่อนไขที่กำหนด	150
if-then : ถ้าเงื่อนไขเป็นจริงแล้วทำอะไร	150
if-then-else : ตรวจสอบเงื่อนไขจริง-เท็จทำอะไร	152
if-touching color : ถ้าแตะโดนสีให้ทำอะไร	154
ตัวอย่างการตรวจจับการแตะสี	155
Cloning Sprite : การโคลนตัวละคร	158
การโคลนตัวละคร 1 ตัว	158
Multiple Cloning : โคลนตัวละครหลายๆ ตัว	159
Delete Clone : ลบตัวละครที่โคลน	159
ตัวอย่างการโคลนตัวละคร	160

LESSON 7

Variables ตัวแปรเก็บข้อมูล	161
Variables : ตัวแปร	162
Make a Variable : สร้างตัวแปรใหม่	162
เรียกใช้ตัวแปร และตั้งค่าตัวแปร	163
ใช้ตัวแปรเก็บคะแนน	163
การเก็บคะแนนจากการตอบคำถาม	166
ตัวอย่างโปรแกรมเกมคูณเลขหลายๆ ข้อ	169



แสดงค่าตัวแปร	170
ใช้ if ตรวจสอบค่าตัวแปร	171
การจัดการตัวแปร	172
Rename variable : เปลี่ยนชื่อตัวแปร	172
Delete variable : ลบตัวแปร	172
Make a List : สร้างตัวแปรแบบรายการ	173
ซ่อน-แสดงกรอบรายการตัวแปร	173
การเพิ่มค่าในตัวแปรแบบ List	174
การลบค่าตัวแปรออกจาก List	174
การใช้งานตัวแปรแบบ List	175
ซ่อน/แสดงกรอบรายการข้อมูล	176
เพิ่ม-ลบรายการในตัวแปรด้วยการพิมพ์ลงไปได้	177
ใช้ if (ถ้า) เอาค่าตัวแปรมากำหนดเงื่อนไขได้	177
การใช้ตัวแปรมาทำเกม	178
กำหนดโค้ดให้ตัวละครลิง	178
เพิ่มฉากหลัง และสร้างฉากเพิ่มใหม่	178
วางบล็อกโค้ดกลุ่มการทำคะแนน และเก็บคะแนน	179
ใส่บล็อกโค้ดให้ตัวละครกล้วย (Bananas)	179
โค้ดโปรแกรมภาษาอังกฤษ	180

LESSON 8

ใช้ Pen ใส่วาดรูปวาดภาพ	181
เพิ่มส่วนขยาย Pen (ปากกา)	182
บล็อกของปากกา	183
เริ่มต้นวาดเส้นแบบง่ายๆ	183
วาดรูปทรงสี่เหลี่ยม	184
การวาดรูปวงกลม	184
วาดวงกลมสีรุ้ง ด้วยการเปลี่ยนสีขณะวาด	185
วาดเส้นไปรอบๆ เวกี	186
วาดเส้นวงกลมซ้อนๆ กัน	187

Half Rainbow : วาดเส้นวงกลมสีรุ้งครึ่งวง	189
ซ่อนตัวละครขณะที่วาดรูป	191
ตัวอย่างการวาดรูปทรงแบบต่างๆ	192
stamp : ปุ่มตัวละครเพิ่มด้วยเมาส์	193
ปุ่มตัวละคร และเปลี่ยนขนาด (Stamp & Resize)	194

LESSON ๑

Sound & Music เสียงประกอบและดนตรี	195
ใส่เสียงประกอบการแสดงตัวละคร	196
ตั้งระดับเสียงสูงต่ำ และแพนเสียงซ้าย/ขวา	197
การเพิ่มเสียง และลดเสียง	197
สร้างเสียงดนตรีจากตัวละคร	198
ปรับแต่งเสียงจากแท็บ Sounds	199
การปรับแต่งเสียง	200
Record : บันทึกเสียง	202
เพิ่มเสียงดนตรี	204
อัปเดตเสียงจากเครื่องคอมพิวเตอร์	205
เพิ่มเสียงดนตรี	206
ใส่เสียงแบ็คกราวนด์	207
สร้างเสียงดนตรี (Music)	208
ประเภทเครื่องดนตรี (Instrument)	209
การเลือกเครื่องดนตรีที่จะเล่น	209
บล็อก และตั้งค่าเครื่องดนตรี	210
play note : เล่นดนตรีตามตัวโน้ต	211
กำหนดค่าตัวโน้ต	212
สร้างเสียงเพลง “Happy Birthday”	213
สร้างเสียงเพลง “หนูมาลี” ตามตัวโน้ต	214



LESSON 10

Extension การทำงานอื่นๆ	215
Text to Speech : อ่านออกเสียงข้อความ	216
ตั้งค่าการอ่านตามข้อความที่เขียน	216
การใช้ Text to Speech : อ่านออกเสียงข้อความ	217
การอ่านข้อความ answer จากผู้ใช้	217
translate : แปลข้อความ	218
การเพิ่มบล็อก Translate	218
แสดงข้อความจากการแปลภาษา	219
สร้างการสนทนา และการแปลภาษาระหว่างตัวละคร	219
Video Sensing	220
การตรวจจับวิดีโอตามเงื่อนไข if-then-else	221
ปิดและเปิดกล้องวิดีโอ	221
ส่วนขยายอื่นๆ	222

LESSON 11

My Blocks บล็อกของฉัน	223
สร้าง My Blocks เก็บการทำงาน	224
การสร้าง My Blocks	224
การใช้งาน My Blocks แบบหมวด	225
การแก้ไขโค้ด My Blocks	226
สร้างบล็อก My Blocks แบบรับค่า	227
การใช้งานบล็อกรับค่าที่สร้างใหม่	228
นำบล็อกรับค่าอินพุตตัวเลขมาใช้งาน	229
เพิ่มบล็อกวาดรูปทรงเรขาคณิตแบบอื่นๆ	229
ตัวอย่างการสร้าง My Blocks หลายๆ ตัว	231
รายละเอียดการทำงานของบล็อกแต่ละชุด	231
เปลี่ยนแปลงรูปทรงและสีได้ง่ายๆ	232
Edit Blocks การแก้ไข My Blocks	233
Delete Blocks การลบ My Blocks	234

LESSON 12

ตัวอย่างงาน และสร้างไฟล์ .exe	235
สร้างเกมแมวบินเก็บรูปหัวใจ	236
เพิ่มตัวแปรเก็บคะแนน	238
แสดงคะแนนแบบคิด (think)	239
นับคะแนนแล้วเปลี่ยนฉากหลัง	239
เกมจับคู่เหมือน	240
เตรียมตัวละครและฉากหลัง	240
สร้างตัวแปรเก็บคะแนน	240
ใส่โค้ดการจับเวลาเพื่อจบเกม (Game Over)	241
กำหนดโค้ดบล็อกตัวละครผีเสื้อ 1 (Butterfly 1)	241
กำหนดโค้ดบล็อกให้ตัวละครผีเสื้อ 2 (Butterfly 2)	242
กำหนดโค้ดบล็อกตัวละครนก (Dove)	242
แสดงค่าตัวแปร และตัวจับเวลา (timer)	243
รันโค้ดทดสอบเกมจับคู่	243
สร้างเกมเก็บคะแนนแบบง่ายๆ	244
โค้ดบล็อกของฉากหลัง (Backdrop)	244
โค้ดบล็อกของตัวละครค้างคาว	244
โค้ดบล็อกของตัวละครโดนนัท	244
โค้ดบล็อกของตัวละครแมลงเต่าทอง	245
รันโค้ดเล่นเกมกินโดนนัท	245
สร้างแมวบิน และเพิ่มขนาด	246
กำหนดบล็อกโค้ด	247
แปลงไฟล์ Scratch เป็นไฟล์ .exe	248
แปลงไฟล์ Scratch (.sb3) เป็นไฟล์ .html ก่อน	248
เปิดไฟล์เดสก์ทอปเก็บไฟล์ html	249
โหลดไฟล์จาวาเว็บแอป nw.js	250
สร้างไฟล์ package.json	251
การนำไฟล์ .exe ไปแชร์ให้กับคนอื่น	253
เพิ่มไอคอนให้กับไฟล์	254
กรณีรันไฟล์ .exe ไม่ผ่าน	254



บันทึก



Lesson



เริ่มต้นกับ scratch

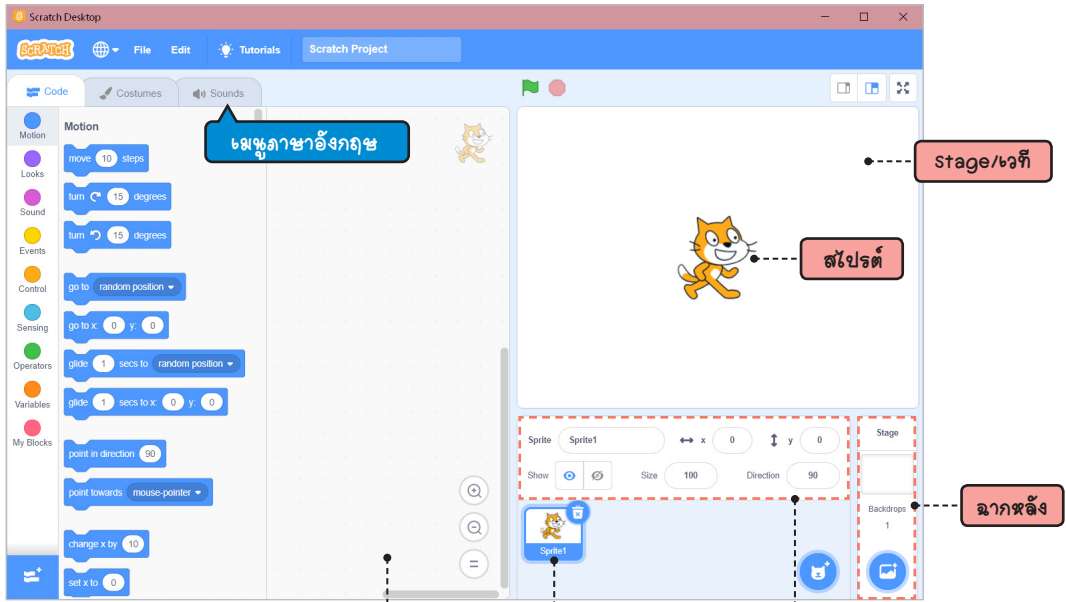
scratch เป็นภาษาการเขียนโปรแกรมภาพแบบบล็อก (Block) เป็น Visual Programming Language คุณสามารถเขียนโปรแกรมเรื่องราวแบบโต้ตอบ เกม และแอนิเมชันได้ และยังมีสิ่งที่คุณสร้างกับคนอื่น ๆ บนชุมชนออนไลน์ scratch ส่งเสริมให้เด็ก ๆ เรียนรู้ที่จะคิดแบบสร้างสรรค์ คิดอย่างมีเหตุผลเป็นระบบ และทำงานร่วมกัน ซึ่งเป็นทักษะจำเป็นสำหรับชีวิตในศตวรรษที่ 21 scratch เป็นโครงการของ Lifelong Kindergarten ณ MIT Media Lab ให้ใช้โดยไม่เสียค่าใช้จ่าย

scratch ได้ถูกใช้ในประเภทต่าง ๆ กว่า 150 ประเภท และมีภาษาให้เลือกถึง 40 ภาษา ทั้งเขียนสามารถเรียนรู้ด้วย scratch ในทุกระดับ (ตั้งแต่ประถมศึกษาจนถึงระดับมหาวิทยาลัย) และหลากหลายสาขา (เช่น คณิตศาสตร์ วิทยาการคอมพิวเตอร์ ศิลปะ ภาษา สังคมศึกษา)

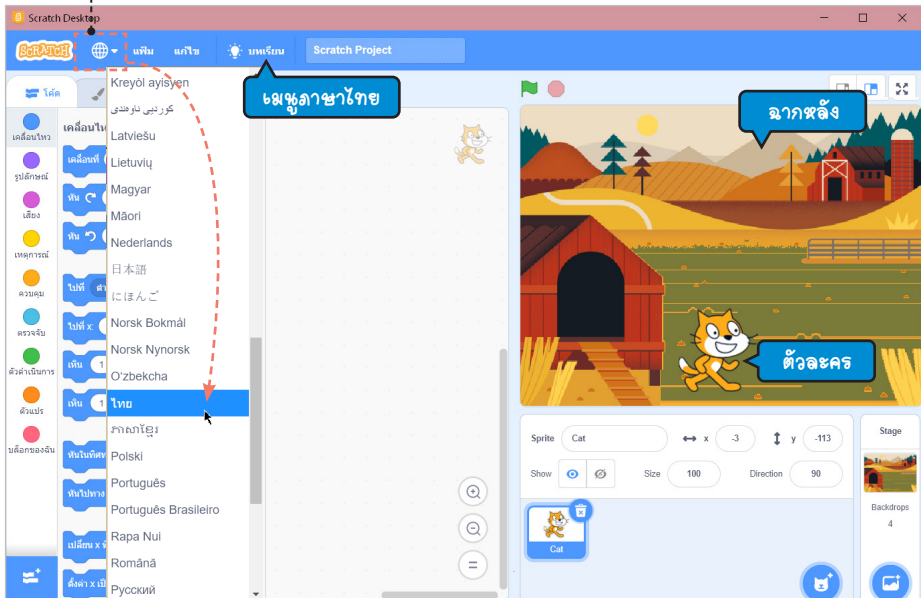
scratch มีชุมชนผู้ใช้ที่เติบโต สามารถเขียนโปรแกรม และยังมีสิ่งใหม่รูปแบบต่าง ๆ สามารถผลิต Digital Media เช่น นิทาน, เกม และแอนิเมชัน Interactive stories, Animations and games ได้คล้ายกับการใช้โปรแกรม Flash หรือ swift แต่ใช้ง่ายกว่า มีตัวละครและฉากที่ใช้สร้างเรื่องราวต่าง ๆ

🗨️ หน้าตาจอของ scratch บนเดสก์ทอป (Desktop)

เป็นการใช้งานแอปบน Windows หรือ iOS ที่ทำงานบนเดสก์ทอป เมื่อเรียกเปิดแอป Scratch ขึ้นมา เราจะเจอกับพื้นที่การทำงาน (Interface) ซึ่งจะแบ่งเป็น 3 ส่วนหลักๆ คือ พื้นที่ออกแบบเกม และภาพเคลื่อนไหว ซึ่งจะใช้ในการวางตัวละคร หรือจะเรียกว่า Sprite (สไปรต์) และกำหนดโค้ดเพื่อให้ตัวละครมีการเคลื่อนไหว หรือทำอะไรตอนไหนบ้าง ส่วนด้านบนสุดที่แถบไดเทลบาร์คือ คำว่า Scratch Desktop

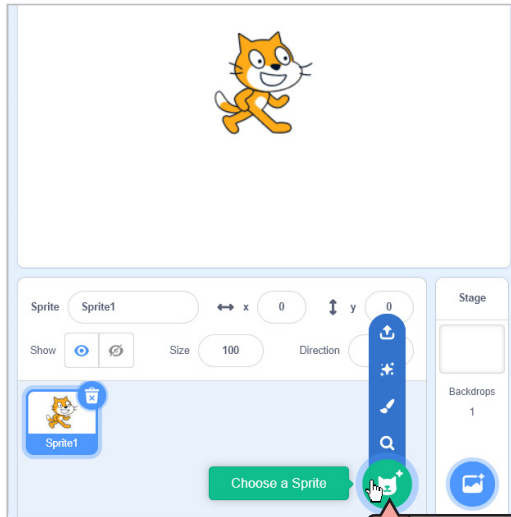


คลิกเปลี่ยนเมนู ไทย/English พื้นที่วางโค้ด (Code) Sprite : สไปรต์ (ตัวละคร) คุณสมบัติของสไปรต์



เพิ่มสไปรต์ (Add New Sprite)

การสร้างโปรเจกต์ใหม่จะมีตัวละคร หรือ Sprite (สไปรต์) 1 ตัว หรือก็ตัวก็ได้ ขึ้นอยู่กับเนื้อเรื่องที่เราสร้างว่าต้องมีตัวละครไบบ้าง การเพิ่มสไปรต์ทำได้ง่ายๆ โดยคลิกปุ่ม Choose a Sprite ซึ่งแพอจะมีสไปรต์ให้เลือกมากมายโดยแบ่งออกเป็นหมวดๆ ตามประเภท เช่น สัตว์, กีฬา, ต้นไม้, ดนตรี และอื่นๆ หรือจะเลือก Paint วาดภาพตัวละครขึ้นมาใช้งานเองก็ได้เช่นเดียวกัน



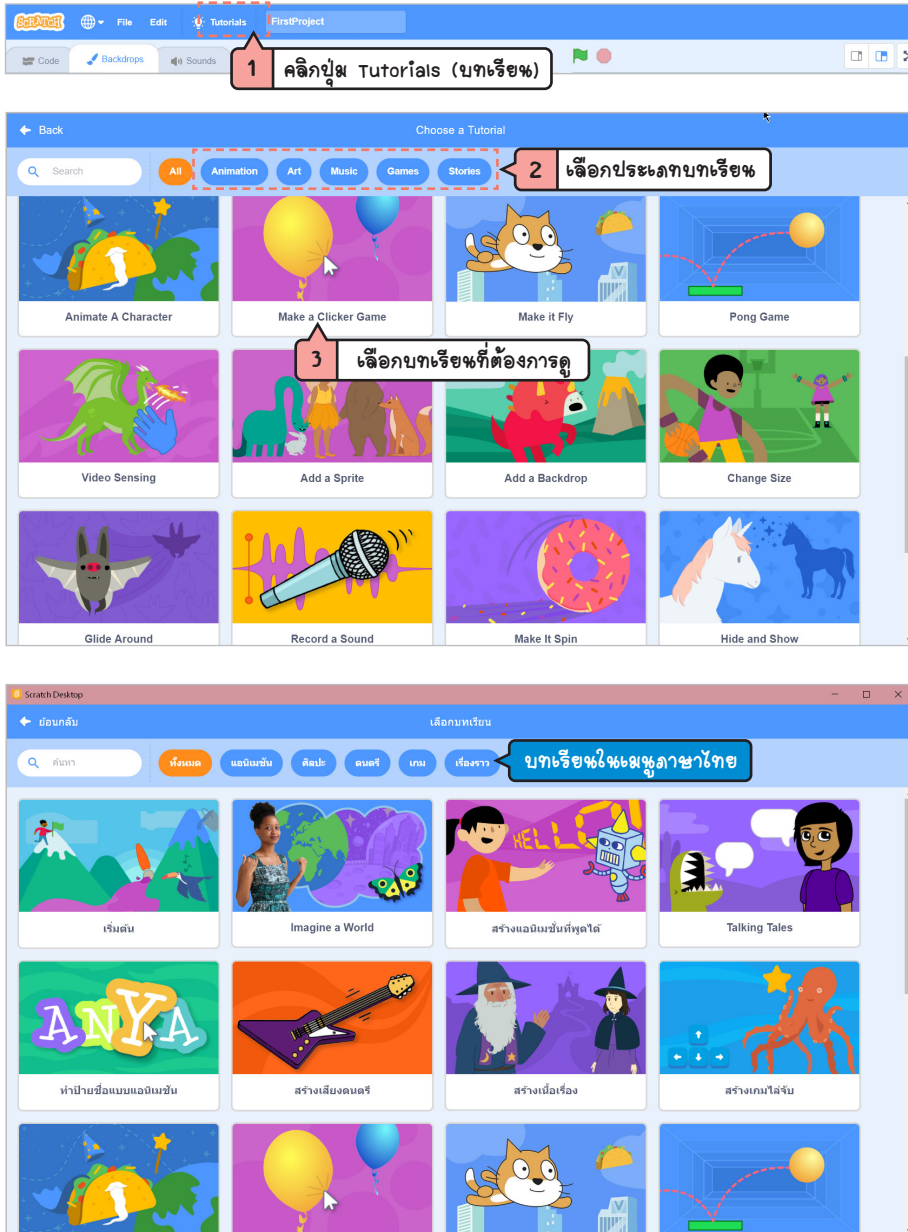
1 คลิกปุ่ม Choose a sprite

- เมื่อเราเปิดโปรแกรม Scratch เข้ามาครั้งแรก หรือใช้คำสั่ง New สร้างโปรเจกต์ใหม่ ตัวละครแรกที่ปรากฏจะเป็นแมว ในชื่อ Sprite 1 แต่สไปรต์ตัวนี้ถ้าเราเพิ่มเข้ามาใหม่มันจะชื่อ Cat คือ ตัวละครแมวนั่นเอง



📖 ศึกษาเรียนรู้จากบทเรียน (Tutorials)

สำหรับมือใหม่หัดใช้สามารถเลือกบทเรียนช่วยสอนที่แอปเตรียมไว้ให้ มาเปิดดูวิธีการทำงานและการใส่โค้ดในแบบต่างๆ ได้ โดยบทเรียนจะแสดงเป็นวิดีโอภาพเคลื่อนไหว ที่มีภาพและเสียงบรรยายแนะนำการสร้างเกม หรือภาพเคลื่อนไหวที่คุณสามารถนำมาเป็นไอเดียประยุกต์สร้างสรรค์งานได้อีก



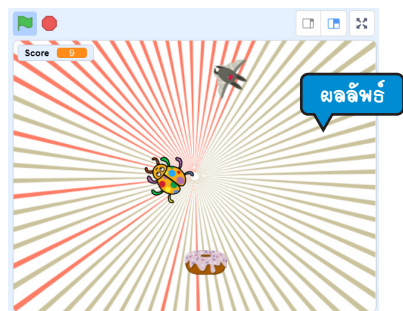
📌 ภาษา scratch ทำงานอย่างไร

ภาษาคอมพิวเตอร์ (Programming Language) ที่ใช้สำหรับเขียนโปรแกรม (Computer Programming) หรือ การเขียน Coding นั้นจะมีมากมายหลายภาษา ภาษาที่ผู้คนใช้งานก็มีหลักการและแนวคิดคล้ายกัน เช่น เริ่มจากภาษา Basic, ภาษา C, Visual Basic, VBA, ASP รวมถึงภาษาประเภท Script อย่าง JavaScript เป็นต้น แต่ละภาษาอาจจะมีความยากง่ายหรือมีอัลกอริทึมที่แตกต่างกันบ้าง

สำหรับภาษา Scratch ก็จะมีโครงสร้างภาษามาตรฐานที่ใช้งานไม่ยาก เพราะเน้นการใช้งานสำหรับเด็ก (อายุ 8-16 ปี) เพื่อปูพื้นฐานให้เด็กฯ เริ่มต้นหัดเขียนโปรแกรม เพื่อให้คุ้นเคยและนำไปต่อยอดความคิดสร้างสรรค์ประยุกต์ใช้งาน และศึกษาต่อในการเรียนภาษาคอมพิวเตอร์ภาษาอื่นๆ ต่อไป แต่ใน Scratch จะเน้นคำสั่งที่ใช้งานง่าย ไม่ซับซ้อน จะมีการตัดเอาคำสั่งหรือการทำงานยากๆ ออกไป เพราะอาจจะยากเกินไปสำหรับเด็ก เพียงแค่ทำได้ตามงานเหมือนการต่อจิ๊กซอว์ และกำหนดการทำงานโดยไม่ต้องมานั่งคิดยึกยักง่ายต่อการเรียนรู้ การฝึกกลยุทธ์สำคัญในการแก้ปัญหา การออกแบบโครงงาน และการสื่อสารแนวคิด แต่เพียงพอต่อการทำโปรเจกต์ได้ตั้งแต่ง่ายๆ จนถึงระดับซับซ้อนที่ไม่ยากเกินไปเพื่อจุดประกายความคิดให้เด็กฯ รักการเขียนโปรแกรม การทำงานของ Scratch จะมีหลักการดังนี้

Sequence	<ul style="list-style-type: none"> การทำงานแบบเรียงลำดับ คือลำดับขั้นตอนการทำงาน เพื่อใช้ในการสร้างโค้ดโดยการนำบล็อกคำสั่งมาเรียงต่อกันทำงานจากบนลงล่าง
Control flow	<ul style="list-style-type: none"> การควบคุมการทำงาน คือแบบวนซ้ำ (Loop) และแบบมีเงื่อนไข (Condition) โดยในบล็อกทำซ้ำและบล็อกเงื่อนไขในการตัดสินใจทำงาน เช่น เงื่อนไขจริงทำอะไรต่อ ไม่จริงทำอะไร หรือตั้งเวลาให้ทำอะไร เป็นต้น
Parallelism	<ul style="list-style-type: none"> การทำงานแบบขนาน ทำงานพร้อมๆ กันได้ ณ เวลาหนึ่ง เช่น มีตัวละครหลายๆ ตัวบนเวทีทำงานพร้อมๆ กันได้ เช่น เปลี่ยนภาพ วิ่ง เดิน มิน หรือแสดงข้อความ ได้พร้อมกัน
Operation:	<ul style="list-style-type: none"> การประมวลผลด้วยระหังข้อความ (String) หรือการคำนวณทางคณิตศาสตร์ (Logic, Mathematical functions) การเปรียบเทียบทางตรรกะศาสตร์ มากกว่า เพื่อนำไปตัดสินใจในการทำงานอย่างอื่นต่อ
Input	<ul style="list-style-type: none"> การรับข้อมูลจากผู้ใช้กับโปรแกรมเพื่อนำไปใช้งาน เช่น การคลิกเมาส์, กดแป้นพิมพ์, สัมผัส, รับเสียง เป็นต้น เพื่อให้มีการทำงานแบบ Interactive เป็นการโต้ตอบกับผู้ใช้
Message	<ul style="list-style-type: none"> การส่งสารระหว่างตัวละครต่างๆ และเวที แสดงข้อความโต้ตอบ จากการตรวจรับการทำงาน หรือการกระทำที่กำหนดเมื่อถึงเวลา
Event and event handling	<ul style="list-style-type: none"> เหตุการณ์และการตอบสนองต่อเหตุการณ์นั้นๆ แล้วให้ทำอะไร เช่น เมื่อตัวละครถูกคลิก, ฉากหลังเปลี่ยน หรือได้รับข้อความอะไร รวมถึงการรับข้อมูลจากตัวแปร เช่น คะแนนจากการเล่น เป็นต้น
Synchronization	<ul style="list-style-type: none"> การประสานและสั่งต่อการทำงาน เป็นการทำงานร่วมกัน มีการรับส่งข้อมูลหรือส่งสารต่อ จากตัวละครตัวหนึ่ง ไปยังตัวละครตัวอื่น เพื่อทำงานร่วมกัน ตัวละครทำงานประสานกัน โดยสร้างเหตุการณ์ให้ตอบสนองต่อกัน

Block Code (บล็อกโค้ด)



หลักการเขียนโค้ด

การเขียนโปรแกรมของ Scratch จะทำแบบ OOP (Object Oriented Programming) โค้ดชุดหนึ่งๆ ประกอบไปด้วยการนำบล็อกมาเรียงต่อกันเหมือนการต่อจิ๊กซอว์ โดยบางบล็อกสามารถซ้อนอยู่บนบล็อกอื่นได้ เพื่อสร้างการทำงานต่อเนื่อง ตัวอย่างการเขียนโค้ดคำสั่งให้เจ้าสไปรต หรือตัวละครแมวเดินเคลื่อนที่ไปข้างหน้า แล้วส่งเสียงร้อง เมื่อหยุดให้แสดงข้อความทักทาย เราก็ต้องกำหนดโค้ดให้กับสไปรตแมว 3 บล็อก สำหรับเด็กๆ แรกๆ อาจกำหนดสตอรี่บอร์ดให้กับตัวละครก่อน เพื่อเป็นการเรียบเรียงแนวคิดก่อนวางบล็อก

1 สร้างสตอรี่บอร์ดภาพรวม

เพิ่มสไปรต → เดินไป 10 หน่วย → ส่งเสียงร้อง → ส่งข้อความทักทาย

2 ลองเปรียบเทียบกับการใช้โค้ด

เพิ่มสไปรต
เดินไป 10 หน่วย
ส่งเสียงร้อง
ส่งข้อความทักทาย

3 เลือกประเภทโค้ด และลากไปวางทางขวา

บางบล็อกคลิกทดสอบก่อนลากไปวางใช้งาน

4 ลากบล็อกไปวางต่อกัน

- การวางบล็อกอาจจะลากมาวางต่อกันได้เลย หรือนำมาวางแยกเป็นกลุ่มงานแล้วค่อยคลิกลากมาต่อกันภายหลังได้
- บางบล็อกคลิกทดสอบจากกรอบทางซ้ายก่อนคลิกลากมาวาง

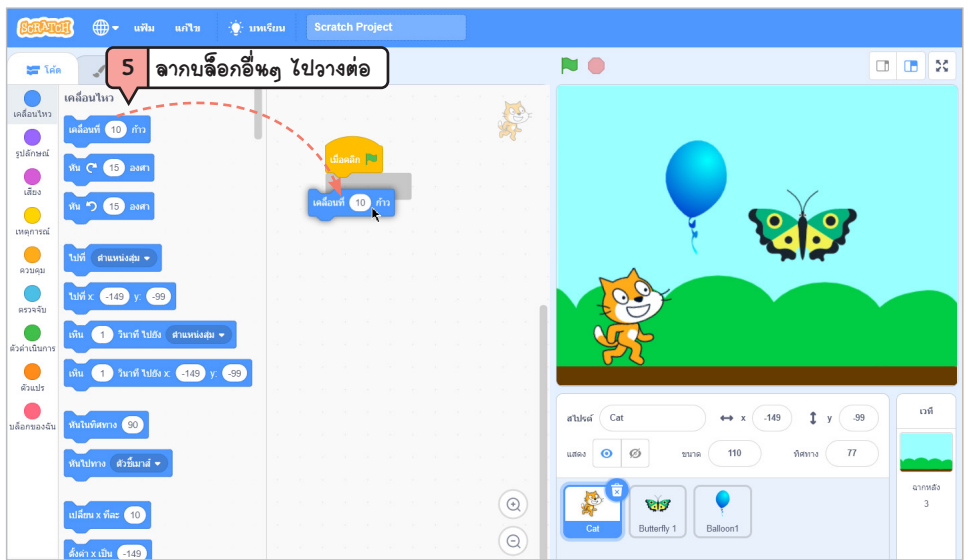
- ทดสอบการทำงานของโค้ด หรือเราจะเรียกว่าการ “รันโค้ด” จะคลิกที่บล็อกบนสุดเพื่อทดสอบทั้งหมดในกลุ่ม หรือจะทดสอบทีละบล็อกก็คลิกที่บล็อกนั้น แต่ถ้าคลิกปุ่ม Go (Flag ธงสีเขียว) จะรันโค้ดทั้งหมดของทุกกลุ่มทุกโค้ดที่วางในสไปรตทุกตัว

5 คลิกปุ่ม Go ทดสอบ

หรือคลิกที่บล็อกโค้ด

เริ่มตั้งเบี่ยงโปรแกรม

หลังจากที่เราทำความรู้จักกับบล็อกคำสั่งต่างๆ ของ Scratch ไปแล้ว ต่อไปก็จะเริ่มเขียนโปรแกรมสร้างเรื่องราวให้กับตัวละครแต่ละตัว โดยจะทดลองวางบล็อกรวมๆ ทั้งหมดก่อน และจะเจาะรายละเอียดการใช้งานแต่ละบล็อกในหัวข้อย่อยต่างๆ แยกออกไป เพื่อจะได้ใช้งานได้ครบถ้วน



📌 หลักการทำภาพเคลื่อนไหว

ภาพเคลื่อนไหว หมายถึง ภาพกราฟิกที่มีการเคลื่อนไหวเพื่อแสดงขั้นตอนหรือปรากฏการณ์ต่างๆ ที่เกิดขึ้นอย่างต่อเนื่อง โดยการนำภาพนิ่งมาเรียงกันเป็นชุดๆ เพื่อแสดงบนจอที่ละภาพด้วยความเร็วสูง ในการฉายภาพแต่ละภาพจะต่อเนื่องกันทำให้ดูเหมือนว่าเคลื่อนไหวจริง ซึ่งอาจเป็นภาพที่ได้จากภาพวาด ภาพถ่าย หรือภาพคน สัตว์ สิ่งของก็ได้ ไม่จำเป็นต้องเฉพาะเจาะจงแต่ภาพการ์ตูน Animation (แอนิเมชัน) เท่านั้น หรือภาพเคลื่อนไหว หมายถึง ภาพเคลื่อนไหวที่สร้างขึ้นโดยการนำภาพนิ่งหลายๆ ภาพมาฉายต่อเนื่องกันด้วยความเร็วสูง ทำให้เกิดภาพลวงตาของการเคลื่อนไหว โดยปกติความเร็วของภาพเคลื่อนไหวจะฉายด้วยความเร็วที่ต่างกัน ขึ้นอยู่กับชนิดของการแสดงผล (Output) โดยถ้าฉายเป็นภาพยนตร์จะฉายด้วยความเร็ว 24 เฟรมต่อวินาที ถ้าถ่ายทอดในระบบ PAL จะฉายด้วยความเร็ว 25 เฟรมต่อวินาที แต่ในระบบ NTSC จะฉายด้วยความเร็ว 29.97 หรือ 30 เฟรมต่อวินาที

ประเภทของภาพเคลื่อนไหว

ภาพเคลื่อนไหวแบ่งได้เป็น 2 ประเภทคือ

- 2D Animation ภาพเคลื่อนไหวแบบ 2 มิติ มองเห็นได้ทั้งความสูงและความกว้าง การสร้างจะไม่สลับซับซ้อนมากนัก โดยนำตัวละครหลักที่มีกิริยาท่าทางแตกต่างกัน นำภาพมาแสดงเรียงต่อกันแบบเฟรมต่อเฟรม ทำให้มองเป็นภาพเคลื่อนไหว ซึ่งเราจะเห็นจากการ์ตูนหรือภาพเคลื่อนไหวแบบ GIF Animation ในโลกอินเทอร์เน็ต หรือภาพสติ๊กเกอร์ที่ใช้กันใน LINE, Facebook หรือโปรแกรมแชททั่วไป
- 3D Animation คือ ภาพเคลื่อนไหวแบบ 3 มิติ ตัวละครจะมีมิติที่เห็นได้แบบ 360 องศา มีความลึก ความกว้าง และความสูง ภาพที่เห็นจะมีความสมจริงของตัวละครมากที่สุด เหมือนเราดูภาพยนตร์หรือการ์ตูนทั่วไป เป็นต้น

ตัวอย่างภาพเคลื่อนไหว 1



- ภาพเคลื่อนไหวใน Scratch เราจะใช้ Sprite (ตัวละคร) กำหนด Costumes (คอสตูม) ที่แสดงกิริยาท่าทางแตกต่างกันมาแสดงสลับกัน โดยการกำหนด Code (โค้ด) ให้สลับตัวละครในแต่ละช่วง ตัวอย่างการทำให้เจ้าแมววิ้งอาจใช้คอสตูม 4 ตัวให้แสดงสลับกัน เป็นต้น

ตัวอย่างภาพเคลื่อนไหว 2

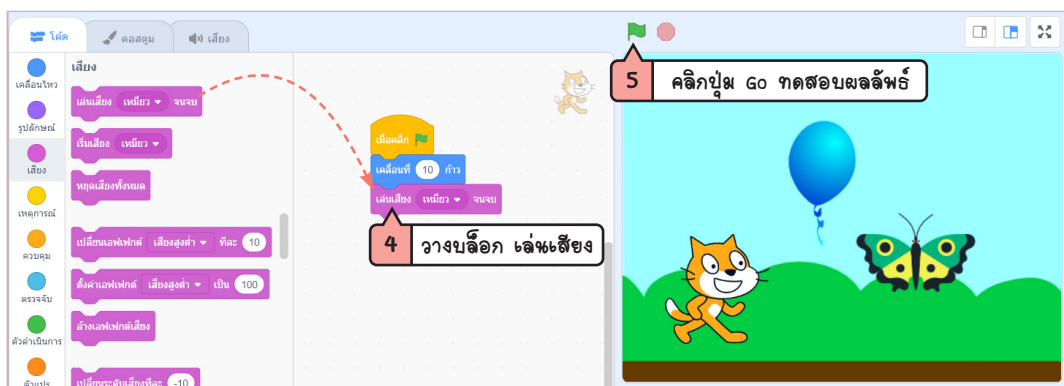
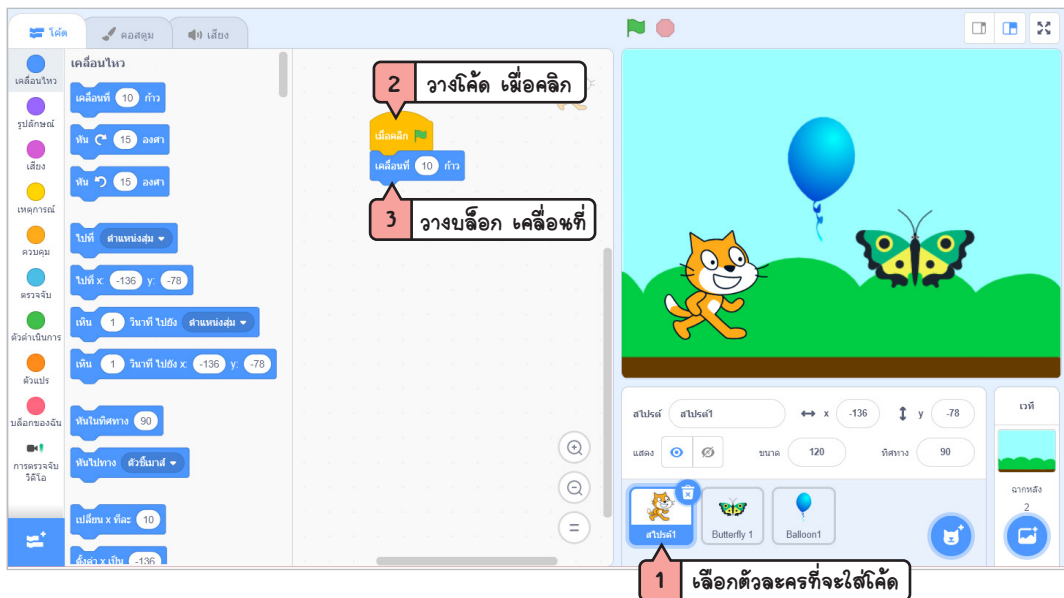


- ในตัวอย่างนี้เราจะทำให้ผีเสื้อบิน ให้สร้างหรือดัดแปลงคอสตูมของตัวละครผีเสื้อเอาไว้ 3 แบบ แล้วนำมาใส่โค้ดให้แสดงกันตามเวลา หรือทิศทางที่กำหนดลงไปได้

📌 สร้างภาพเคลื่อนไหวอย่างง่าย

ในตัวอย่างนี้เรามาลองสร้างภาพเคลื่อนไหวอย่างง่าย ๆ โดยแทรกตัวละครแมวแล้วให้เดินตรงไปทางข้างหน้า และส่งเสียงร้อง “เหมียว” ก็จะใส่โค้ดลงไปตามลำดับดังนี้

ตัวละคร : Sprite	บล็อก : Block	เหตุการณ์ : Events
Cat (แมว)		เมื่อคลิกปุ่ม Go หรือคลิกที่บล็อก
	 	ให้เคลื่อนที่ (move) ไป 10 ก้าว ให้ส่งเสียง (play sound) แมวร้องจนจบ



การใช้ตัวแปรมาทำเกม

ตัวอย่างการเขียนโปรแกรม และการใช้ตัวแปรอีกแบบหนึ่งด้วยการสร้างเกมลิงกินกล้วย แล้วเก็บคะแนนเอาไว้ โดยให้นำเอาการทำงานต่างๆ มาใช้งานร่วมกัน ตั้งแต่การกดลูกศรบนแป้นพิมพ์ เพื่อให้เจ้าลิงเดินไปซ้าย-ขวา, บน-ล่าง เพื่อเก็บกล้วย และใช้คำสั่งกลุ่ม Control บล็อก if มากำหนดเงื่อนไขการเก็บกล้วยและเก็บคะแนนด้วยการสร้างตัวแปร ขึ้นมาเก็บค่ากำลังและโดนกล้วย ซึ่งจะใช้บล็อกโค้ดหลายชุด ดังนี้

กำหนดโค้ดให้ตัวละครลิง

เพิ่มฉากหลัง และสร้างฉากใหม่