

CDG Code Their Dreams X CED KMUTNB



Scratch Program Training

การเขียนโปรแกรมคอมพิวเตอร์เบื้องต้น ด้วยโปรแกรม Scratch

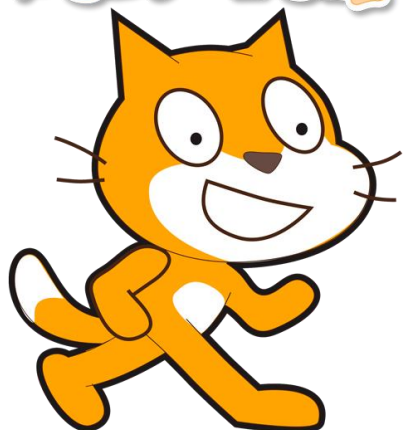


โครงการ Public Training หลักสูตรอบรมการเขียนโปรแกรมคอมพิวเตอร์เพื่อการศึกษา

โครงการความร่วมมือระหว่าง CDG Code Their Dreams กับ ภาควิชาคอมพิวเตอร์ศึกษา คณะครุศาสตร์อุตสาหกรรม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

โปรแกรม Scratch คืออะไร ?

Scratch



Scratch เป็นโปรแกรมสร้างสื่อมัลติมีเดียที่ใช้งานได้ทั้งแบบ ออฟไลน์ และออนไลน์ เหมาะสำหรับนักเรียน นักวิชาการ ครู และผู้ประกอบการ ที่มีวัตถุประสงค์เพื่อสร้างงานด้านการศึกษาหรือเพื่อความบันเทิง เช่น โครงงาน คณิตศาสตร์ โครงงานวิทยาศาสตร์ แบบจำลอง สถานการณ์จำลอง งานนำเสนอหรือสื่อที่ประกอบด้วยภาพเคลื่อนไหว เล่าเรื่องราว งานด้านศิลปะและดนตรีแบบมีปฏิสัมพันธ์ หรือเกมง่ายๆ ล้วนสร้างได้ด้วย Scratch

Scratch คืออะไร ?

Scratch ประกอบไปด้วย 2 ส่วน

- 1) โปรแกรมคอมพิวเตอร์
- 2) ภาษาคอมพิวเตอร์

ในส่วนของโปรแกรมคอมพิวเตอร์มีหน้าที่
ในการสร้าง จัดการภาพ เสียง และ Script ในชิ้นงาน



Scratch คืออะไร ?



ในส่วนของภาษาคอมพิวเตอร์ หรือ ภาษา Scratch จะทำหน้าที่ผสม ภาพ เสียง สร้างกระบวนการและปฏิสัมพันธ์ มีลักษณะการเขียนที่แตกต่างไปจากภาษาคอมพิวเตอร์อื่นคือ ลากบล็อกคำสั่งมาเชื่อมต่อกัน โดยไม่ต้องเขียนคำสั่งใด ๆ แล้วทำการ Run โดยการคลิกที่บล็อกคำสั่งหรือธงสีเขียว ก็จะแสดงผลลัพธ์ออกมาทันที

Scratch คืออะไร ?

การเขียนโปรแกรมในลักษณะนี้ทำให้เข้าใจง่าย มองเห็นภาพ และลดความผิดพลาดในการพิมพ์ คำสั่งลงบล็อกคำสั่งจะมีรูปแบบเฉพาะที่เชื่อมต่อกัน ไม่สามารถต่อผิดรูปแบบได้ รูปแบบภาษา Scratch ถูกนำไปใช้ในโปรแกรมมากมายเพื่อทำให้การสร้างผลงานหรือการเขียนโปรแกรมแบบเต็มที่มีความยุ่งยากซับซ้อนให้ง่ายขึ้นและสร้างชิ้นงานได้รวดเร็ว

Scratch

เครื่องมือที่ใช้ในการออกแบบและเขียนโปรแกรม Scratch

1. โปรแกรมเขียนผังงานระบบ (Flow chart) Microsoft visio, Microsoft word
2. โปรแกรมออกแบบหน้าจอ Microsoft PowerPoint
3. โปรแกรมพัฒนาระบบ Scratch Version 3.0



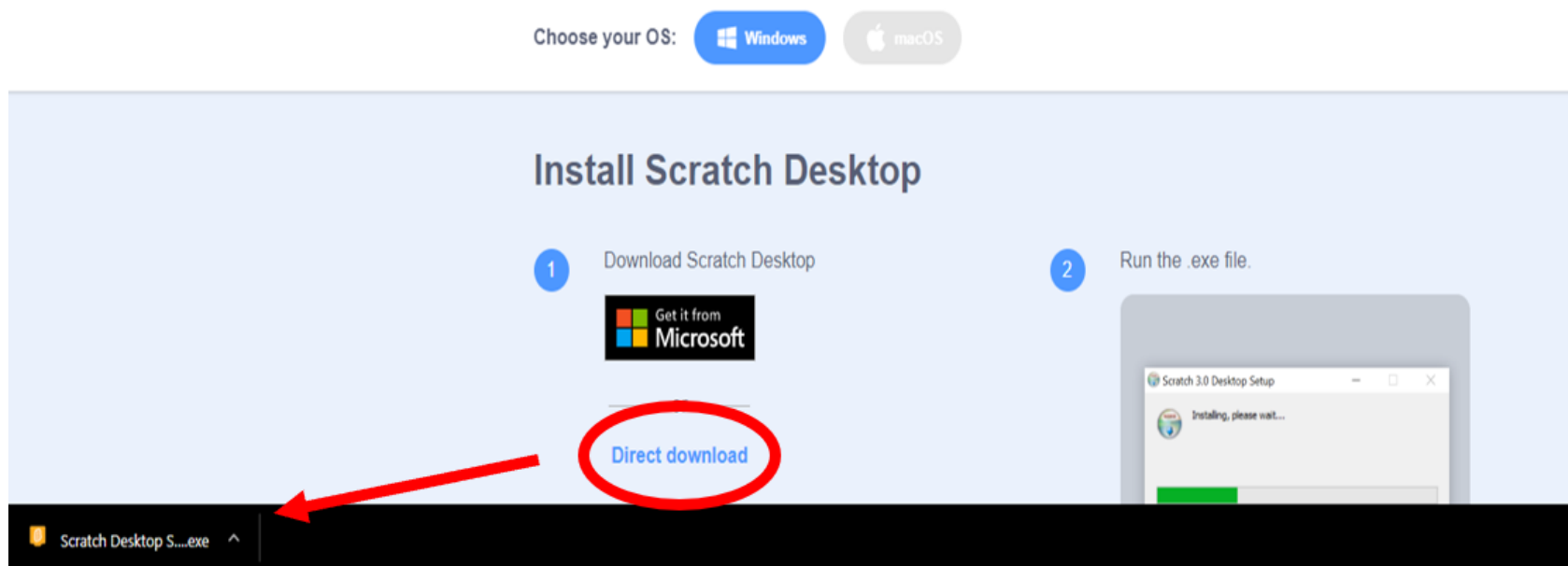
การติดตั้งโปรแกรม Scratch

ติดตั้งโปรแกรม scratch version 3.0

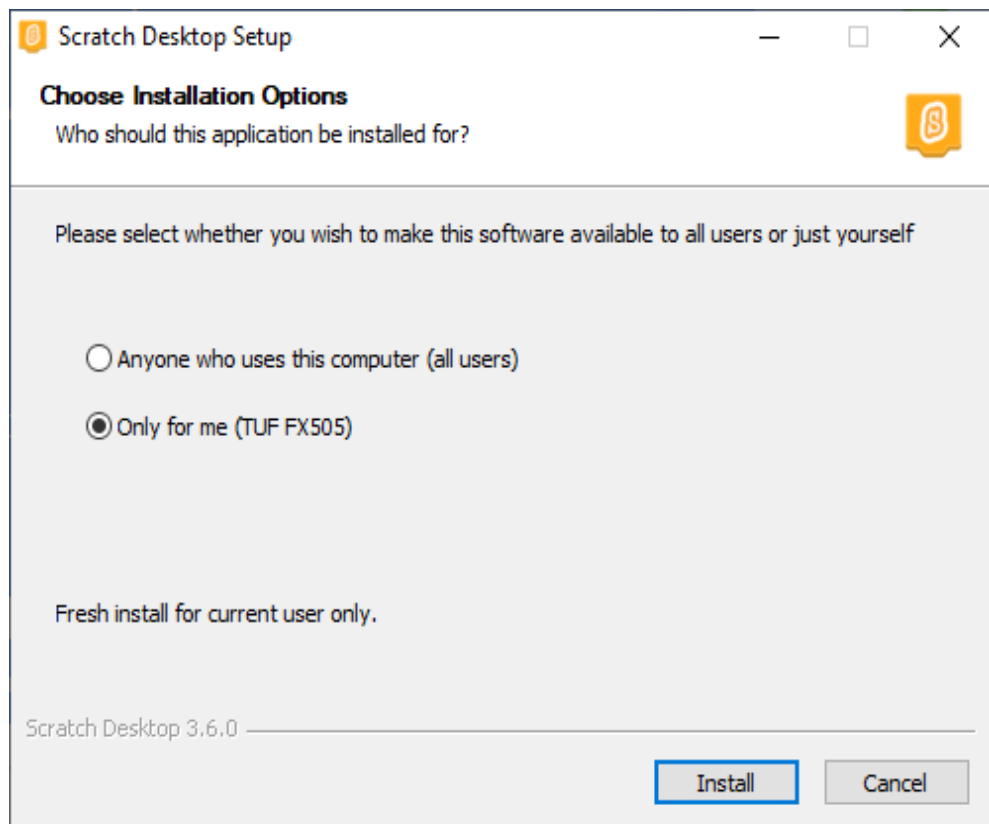
1. เข้าสู่เว็บไซต์ของ Scratch เพื่อ
เข้าไปดาวน์โหลดไฟล์ติดตั้ง
scratch.mit.edu/download

ติดตั้งโปรแกรม scratch version 3.0

2. กด Direct Download เพื่อดาวน์โหลดตัวติดตั้ง หลังจากดาวน์โหลดเสร็จ กด
เปิดตัวติดตั้ง



ติดตั้งโปรแกรม scratch version 3.0



3. เมื่อเปิดตัวติดตั้งขึ้นมาจะพบหน้าต่าง ให้เลือกระหว่าง Anyone กับ Only for me

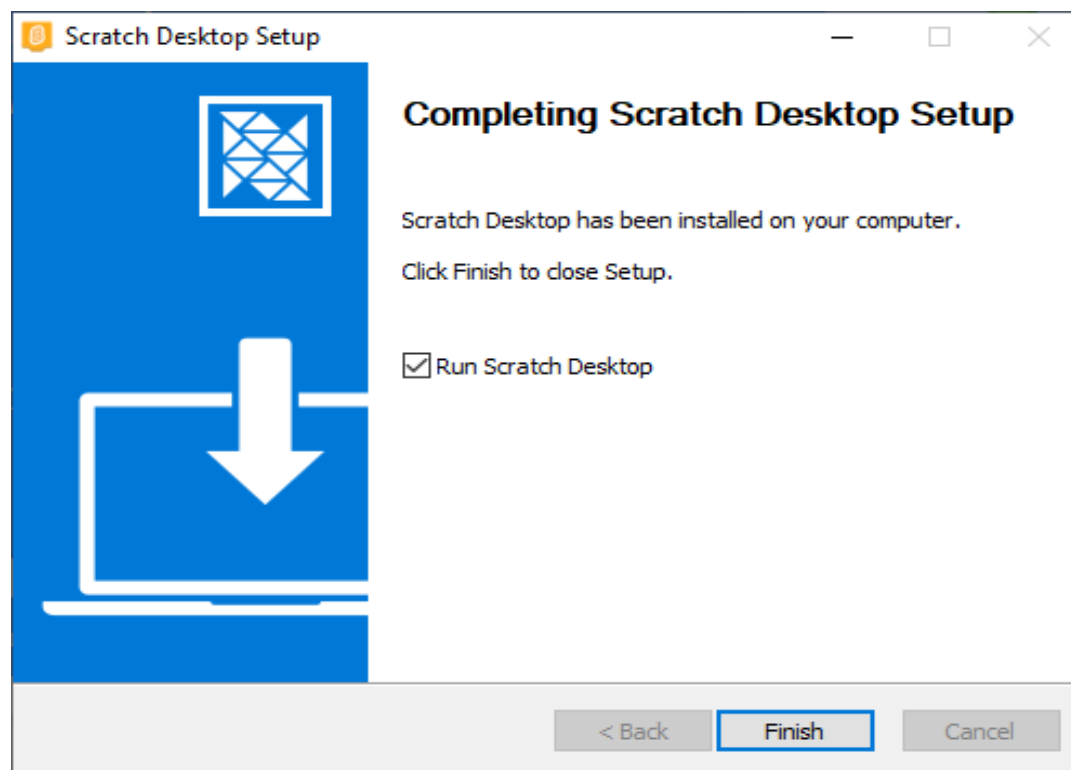
*Anyone เหมาะสำหรับเครื่องที่ต้องการใช้โปรแกรมกับทุก user

*Only for me เหมาะสำหรับเครื่องที่ต้องการใช้เพียง user เดียว

หลังจากเลือกแล้วกด Install แล้วรอจนเสร็จ

ติดตั้งโปรแกรม scratch version 3.0

4. เมื่อเสร็จจะปรากฏหน้าต่างบอก กด Finish เพื่อเริ่มต้นใช้งานโปรแกรม



ความรู้เบื้องต้นการเขียนโปรแกรม

ตัวแปร

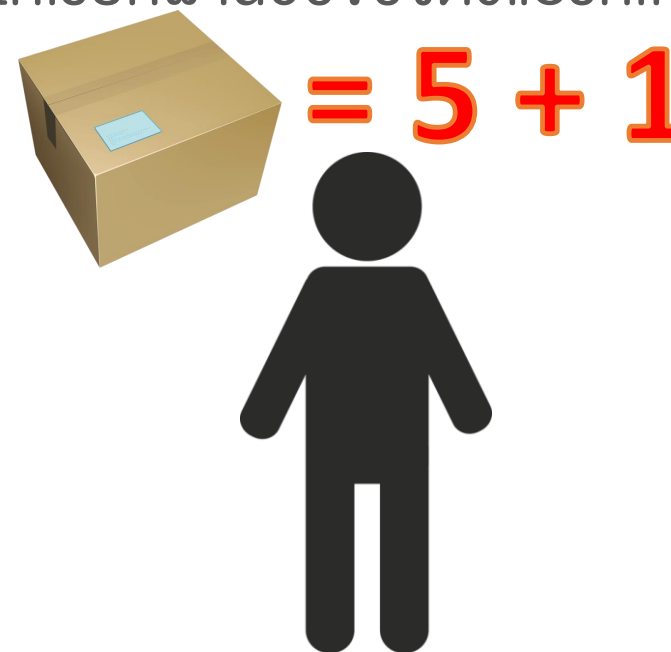
ตัวแปร (Variable) คือ การจองพื้นที่ในหน่วยความจำของคอมพิวเตอร์สำหรับเก็บข้อมูลที่ต้องใช้ในการทำงานของโปรแกรม โดยมีการตั้งชื่อเรียกหน่วยความจำในตำแหน่งนั้นด้วย เพื่อความสะดวกในการเรียกใช้ข้อมูล ถ้าจะใช้ข้อมูลใดก็ให้เรียกผ่านชื่อของตัวแปรที่เก็บเอาไว้

5 + 1



ตัวแปร

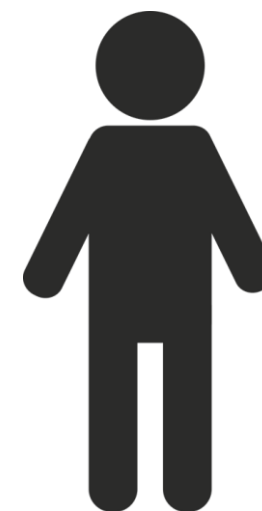
ตัวแปร (Variable) คือ การจองพื้นที่ในหน่วยความจำของคอมพิวเตอร์สำหรับเก็บข้อมูลที่ต้องใช้ในการทำงานของโปรแกรม โดยมีการตั้งชื่อเรียกหน่วยความจำในตำแหน่งนั้นด้วย เพื่อความสะดวกในการเรียกใช้ข้อมูล ถ้าจะใช้ข้อมูลใดก็ให้เรียกผ่านชื่อของตัวแปรที่เก็บเอาไว้



ตัวแปร

ตัวแปร (Variable) คือ การจองพื้นที่ในหน่วยความจำของคอมพิวเตอร์สำหรับเก็บข้อมูลที่ต้องใช้ในการทำงานของโปรแกรม โดยมีการตั้งชื่อเรียกหน่วยความจำในตำแหน่งนั้นด้วย เพื่อความสะดวกในการเรียกใช้ข้อมูล ถ้าจะใช้ข้อมูลใดก็ให้เรียกผ่านชื่อของตัวแปรที่เก็บเอาไว้

num = 5 + 1



ตัวแปร

ตัวแปร (Variable) คือ การจองพื้นที่ในหน่วยความจำของคอมพิวเตอร์สำหรับเก็บข้อมูลที่ต้องใช้ในการทำงานของโปรแกรม โดยมีการตั้งชื่อเรียกหน่วยความจำในตำแหน่งนั้นด้วย เพื่อความสะดวกในการเรียกใช้ข้อมูล ถ้าจะใช้ข้อมูลใดก็ให้เรียกผ่านชื่อของตัวแปรที่เก็บเอาไว้

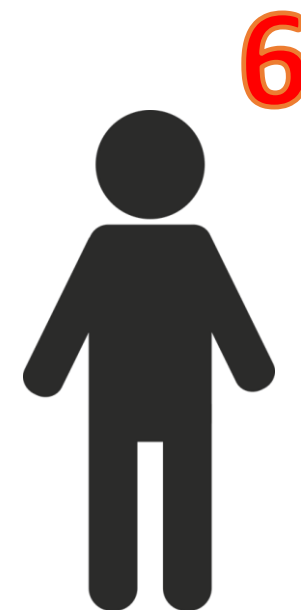
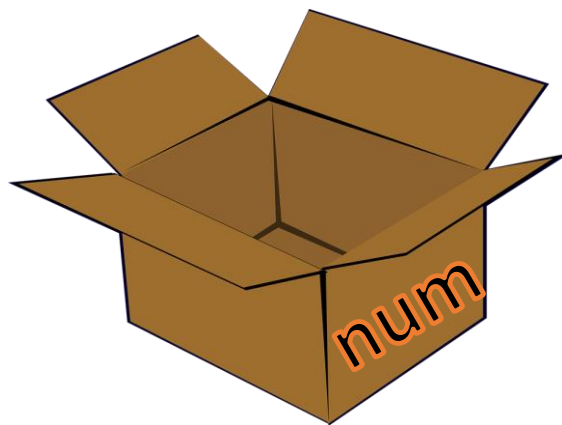


num = 6



ตัวแปร

ตัวแปร (Variable) คือ การจองพื้นที่ในหน่วยความจำของคอมพิวเตอร์สำหรับเก็บข้อมูลที่ต้องใช้ในการทำงานของโปรแกรม โดยมีการตั้งชื่อเรียกหน่วยความจำในตำแหน่งนั้นด้วย เพื่อความสะดวกในการเรียกใช้ข้อมูล ถ้าจะใช้ข้อมูลใดก็ให้เรียกผ่านชื่อของตัวแปรที่เก็บเอาไว้



ตัวแปร

ตัวแปร (Variable) คือ การจองพื้นที่ในหน่วยความจำของคอมพิวเตอร์สำหรับเก็บข้อมูลที่ต้องใช้ในการทำงานของโปรแกรม โดยมีการตั้งชื่อเรียกหน่วยความจำในตำแหน่งนั้นด้วย เพื่อความสะดวกในการเรียกใช้ข้อมูล ถ้าจะใช้ข้อมูลใดก็ให้เรียกผ่านชื่อของตัวแปรที่เก็บเอาไว้



ตัวดำเนินการ

ตัวดำเนินการจะถูกใช้กับตัวแปรและค่าคงที่ในการดำเนินการบางอย่าง เช่น การดำเนินการทางคณิตศาสตร์ มีตัวดำเนินการที่กำหนดที่แตกต่างกันไป แบ่งออกได้ 3 ประเภท

1. Arithmetic operators
2. Relational และ comparison operators
3. Logical operators



1. Arithmetic operators (+, -, *, /, %)

ตัวดำเนินการทางคณิตศาสตร์ คือตัวดำเนินการที่ใช้เพื่อกระทำการดำเนินการทางคณิตศาสตร์ระหว่างตัวแปรหรือค่าคงที่ เช่น การบวก การลบ การคูณ การหาร และการหารเอาเศษ

Symbol	Name	Example
+	Addition	$c = a + b$
-	Subtraction	$c = a - b$
*	Multiplication	$c = a * b$
/	Division	$c = a / b$
%	Modulo	$c = a \% b$

2. Relational และ comparison operators (==, !=, >, <, >=, <=)

ตัวดำเนินการความสัมพันธ์และเปรียบเทียบ คือ ตัวดำเนินการที่ถูกใช้เพื่อประเมินค่า true และ false ระหว่างสองค่าถูกดำเนินการ ซึ่งขึ้นกับเงื่อนไขและความสัมพันธ์ของข้อมูล

Operator	Example	Result
==	a == b	true if `a` equal to `b`, otherwise false
!=	a != b	true if `a` not equal to `b`, otherwise false
<	a < b	true if `a` less than `b`, otherwise false
>	a > b	true if `a` greater than `b`, otherwise false
<=	a <= b	true if `a` less than or equal to `b`, otherwise false
>=	a >= b	true if `a` greater than or equal to `b`, otherwise false

3. Logical operators (!, &&, ||)

ตัวดำเนินการทางตรรกศาสตร์ถูกใช้เพื่อประเมิน Expression (การแสดงผล) ย่อย
หลายๆ Expression ให้เหลือเพียงอันเดียว โดยผลลัพธ์สุดท้ายนั้นจะเป็นจริงหรือเท็จ

Name	Symbol	Example
not	!	!a
and	&&	a && b
or		a b

ผังงาน (Flowchart) คืออะไร

แผนภาพแสดงการทำงานของโปรแกรม โดยใช้สัญลักษณ์แสดงขั้นตอนและลักษณะการทำงานแบบต่างๆ สัญลักษณ์เหล่านี้จะถูกเชื่อมโยงด้วยลูกศรเพื่อแสดงลำดับการทำงาน ช่วยให้มองเห็นภาพการทำงานโดยรวมของโปรแกรม สะดวกต่อการตรวจสอบความถูกต้องของลำดับการทำงานและการไหลของข้อมูลในโปรแกรม


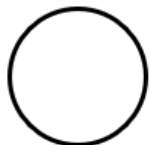
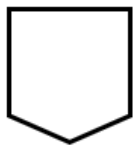
สัญลักษณ์ของผังงาน (Flowchart) มีอะไรบ้าง

สัญลักษณ์	ชื่อสัญลักษณ์	คำอธิบาย
	Terminal	แสดงจุดเริ่มต้น และจุดสิ้นสุดการทำงาน
	Manual Input	การรับข้อมูลเข้าทางแป้นพิมพ์ โดยการป้อนข้อมูลด้วยตนเอง
	Input / Output	แสดงผลการรับข้อมูลหรือการแสดงผลลัพธ์โดยไม่ระบุชนิดของอุปกรณ์ที่ใช้
	Display	แสดงผลลัพธ์ทางหน้าจอ

สัญลักษณ์ของผังงาน (Flowchart) มีอะไรบ้าง

สัญลักษณ์	ชื่อสัญลักษณ์	คำอธิบาย
	Process	การประมวลผล
	Decision	การตรวจสอบเงื่อนไข (การตัดสินใจ) เพื่อเลือกทำงานอย่างใดอย่างหนึ่ง
	Flow Line	ลูกศรแสดงทิศทางการทำงานของโปรแกรมและการไหลของข้อมูล

สัญลักษณ์ของผังงาน (Flowchart) มีอะไรบ้าง

สัญลักษณ์	ชื่อสัญลักษณ์	คำอธิบาย
	Document Output	แสดงผลทางเอกสารหรือเครื่องพิมพ์
	On page connection	จุดเชื่อมต่อของผังงานในหน้าเดียวกัน
	Off page connection	จุดเชื่อมต่อของผังงานไปหน้าอื่น

ประโยชน์ของการเขียนผังงาน

- เราสามารถเรียนรู้และเข้าใจผังงานได้ง่าย เพราะผังงานไม่ขึ้นอยู่กับภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง
- ผังงานเป็นการสื่อความหมายด้วยภาพ ทำให้ง่ายและสะดวกต่อการพิจารณาลำดับขั้นตอนในการทำงานดีกว่าการบรรยายเป็นตัวอักษร
- ช่วยให้การค้นหาความผิดพลาดของลำดับการทำงานได้รวดเร็ว และสามารถแก้ไขได้ง่าย

ประโยชน์ของการเขียนผังงาน

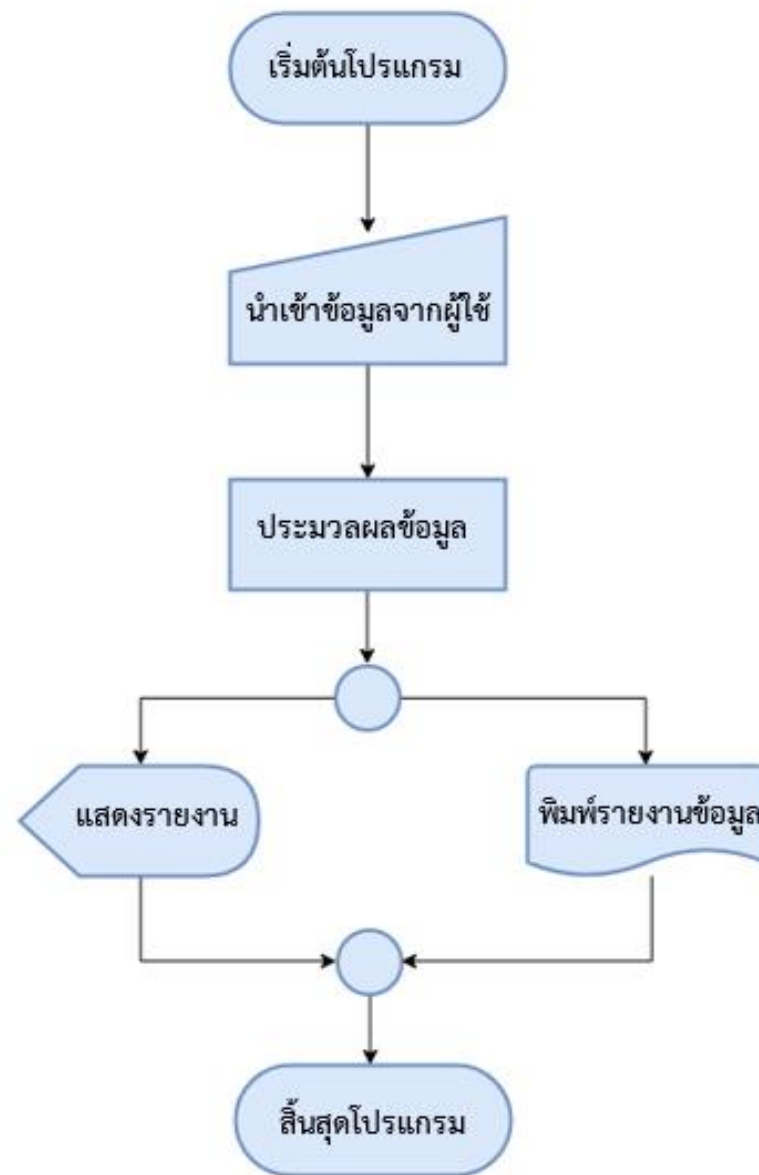
- การเขียนโปรแกรมโดยพิจารณาจากผังงาน สามารถทำได้รวดเร็วและง่ายขึ้น
- ในการบำรุงรักษาโปรแกรม หรือมีการเปลี่ยนแปลงแก้ไขโปรแกรมให้มีประสิทธิภาพ ถ้าดูจากผังงานจะช่วยให้สามารถทบทวนการทำงานในโปรแกรมได้ง่ายขึ้น

ผังงานทางคอมพิวเตอร์แบ่งออกเป็น 2 ประเภท ได้แก่

1. ผังงานระบบ (System flowchart)
2. ผังงานโปรแกรม (Program flowchart)

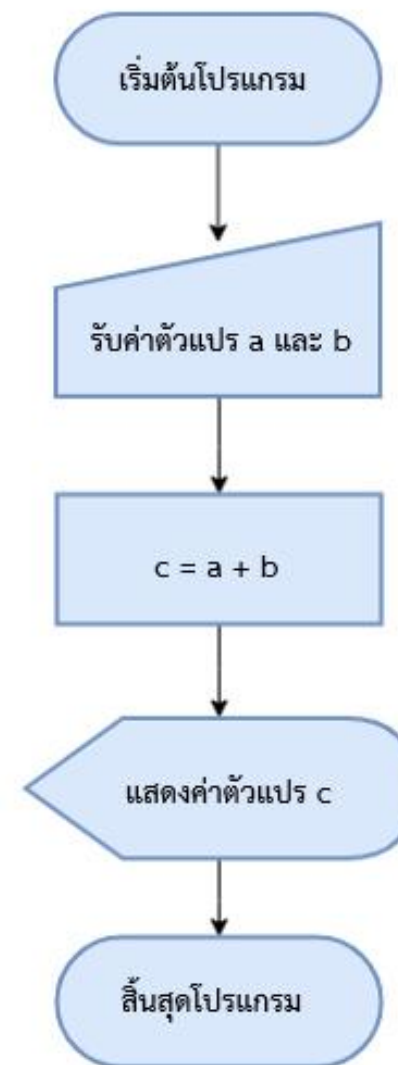
ผังงานระบบ (System flowchart)

เป็นผังงานที่ใช้แสดงขั้นตอนการทำงานของระบบงานอย่างกว้าง ๆ ทั้งระบบ โดยแสดงให้เห็นถึงความสัมพันธ์ของส่วนสำคัญต่าง ๆ ในระบบนั้น



ผังงานโปรแกรม (Program flowchart)

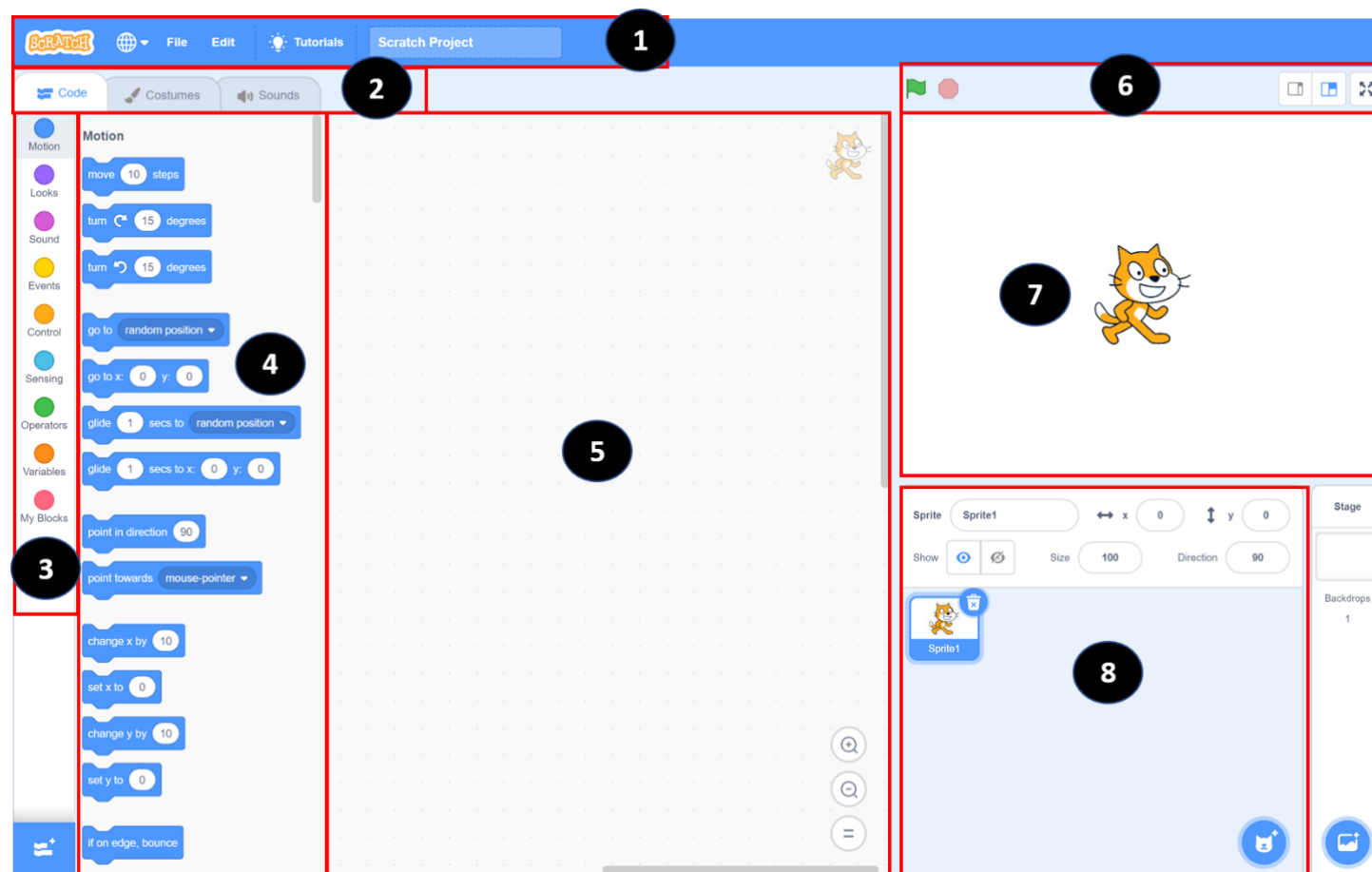
เป็นผังงานที่แสดงขั้นตอนของคำสั่งที่ใช้ในโปรแกรม ผังงานนี้แยกย่อยมาจากผังงานระบบ โดยผู้เขียนโปรแกรมจะดึงเอาแต่ละจุดที่เกี่ยวข้องกับการประมวลผลในคอมพิวเตอร์ที่ปรากฏในผังงานระบบ มาเขียนแสดงรายละเอียดในการประมวลผลอย่างเป็นลำดับขั้นตอน เพื่อนำไปประกอบการเขียนโปรแกรมต่อไป



ใช้ Scratch

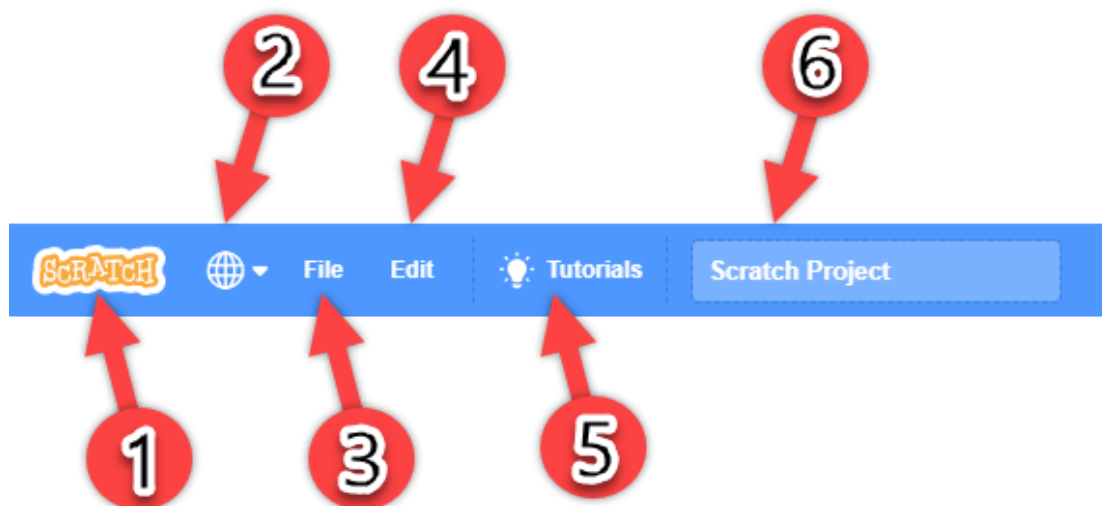
ส่วนประกอบและเครื่องมือโปรแกรม Scratch

หน้าต่างของ Scratch เป็นพื้นที่ที่จะต้องใช้ในการสร้างชิ้นงาน แบ่งออกเป็น 8 ส่วน ดังนี้



ส่วนประกอบและเครื่องมือโปรแกรม Scratch

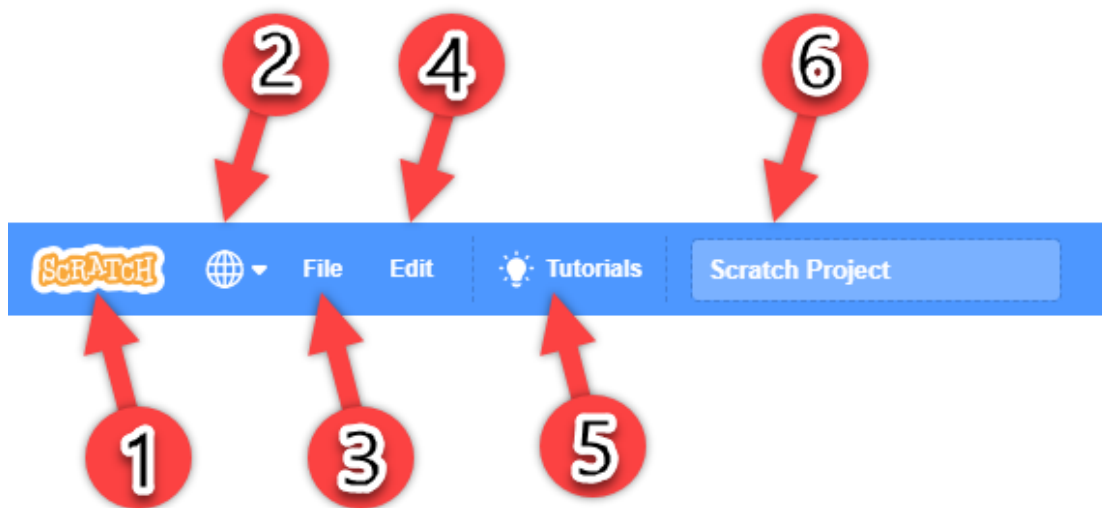
1. Tools bar เป็นส่วนที่อยู่ด้านบนของ Scratch ประกอบด้วยเมนู และปุ่มเครื่องมือ



1. เข้าสู่เว็บไซต์ของ Scratch
2. เปลี่ยนภาษา
3. จัดการไฟล์ต่าง ๆ (สร้างไฟล์ใหม่, โหลดและบันทึกไฟล์จากคอมพิวเตอร์)

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

1. Tools bar เป็นส่วนที่อยู่ด้านบนของ Scratch ประกอบด้วยเมนู และปุ่มเครื่องมือ



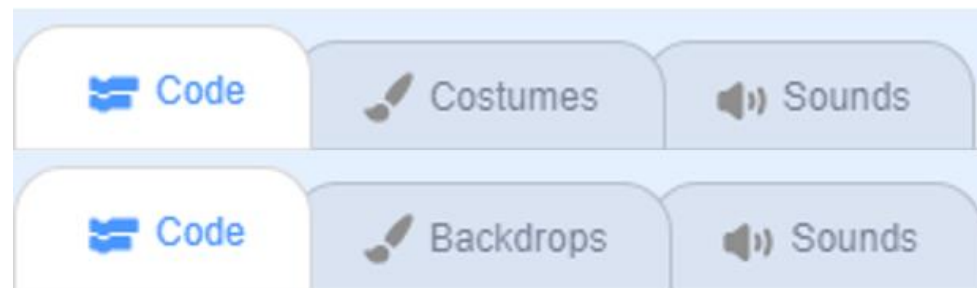
4. ใช้ในการ Restore ไฟล์ในกรณีที่เราเรียกไฟล์จากเครื่องหรือบันทึก แล้วเปลี่ยนโหมดของการเล่น Animation ของงาน

5. เป็นฟังก์ชันในการสอนทริปต่างๆ

6. ใช้ในการค้นหา Public Project

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

2. Code, Costumes, Backdrops, Sounds



1. เมนู Code - เป็นส่วนประกอบที่สำคัญเพราะในเมนูนี้จะเป็นส่วนในการนำชุด Code มาใส่ในพื้นที่เขียน Scripts เพื่อกำกับการทำงานของวัตถุ
2. เมนู Cosstumes – เป็นส่วนในการปรับแต่งวัตถุต่างๆ
3. เมนู Backdrops - เป็นส่วนในการปรับแต่งพื้นหลังของฉาก
4. เมนู Sounds – เป็นส่วนในการจัดการเสียงต่างๆ

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

3. หมวดหมู่ของ Scripts



1. Motion – จะเป็นส่วนของการสั่งเคลื่อนไหววัตถุ (ใช้ไม่ได้กับพื้นหลัง)
2. Looks – จะเป็นส่วนของการสั่งปรับเปลี่ยน เช่น ชุด, ขนาด, แสดง, ซ่อน ฯลฯ
3. Sound – จะเป็นส่วนของการใส่เสียงต่าง ๆ
4. Event – จะเป็นส่วนของการใส่เงื่อนไขเพื่อทำสิ่งต่าง ๆ

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

3. หมวดหมู่ของ Scripts



Motion



Looks



Sound



Events



Control



Sensing



Operators

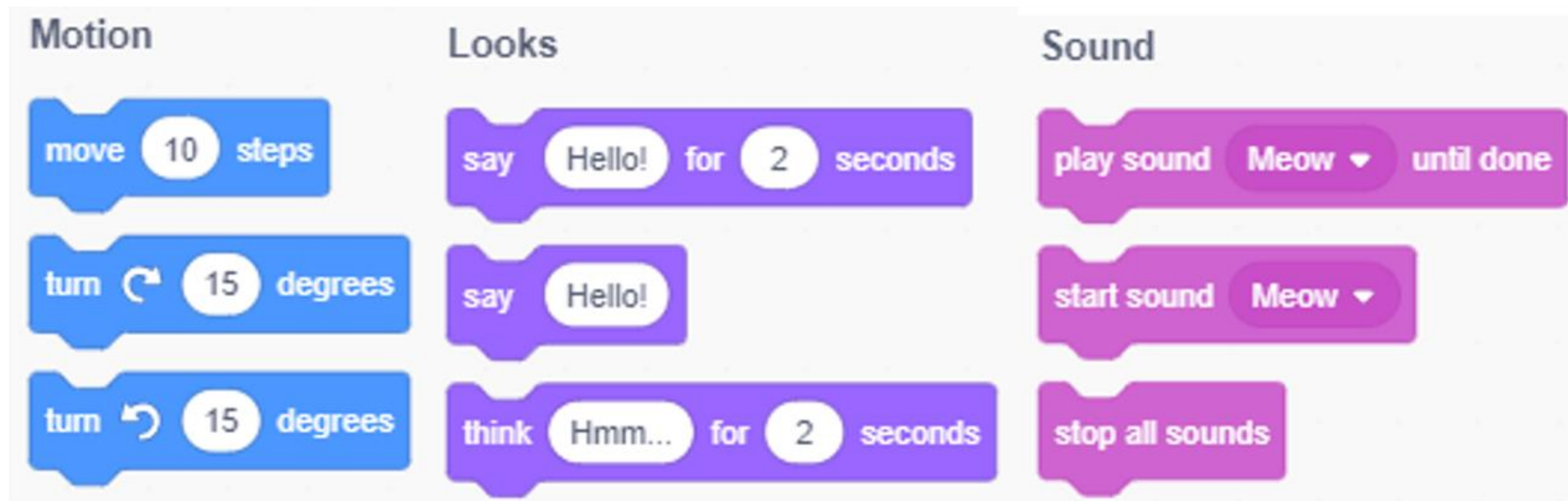


Variables

5. Control – จะเป็นส่วนของการควบคุมของชุดคำสั่ง
6. Sensing – จะเป็นส่วนของการรับค่าจากผู้ใช้ เพื่อสั่งงานในลำดับต่อไป
7. Operators – จะเป็นส่วนของการสร้างเงื่อนไขในรูปแบบลอจิก
8. Variables – จะเป็นส่วนของการสร้างตัวแปรเพื่อเก็บค่าต่าง ๆ

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

4. พื้นที่แสดงบล็อกคำสั่งต่าง ๆ

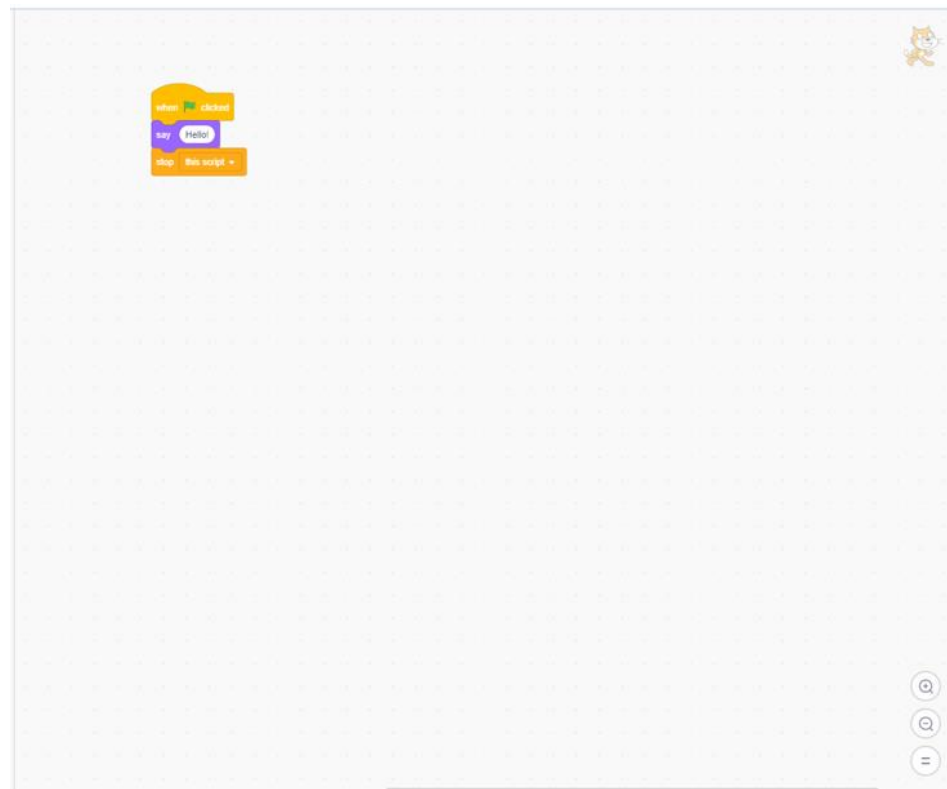


The image displays three columns of Scratch code blocks:

- Motion:**
 - move 10 steps
 - turn 15 degrees (clockwise)
 - turn 15 degrees (counter-clockwise)
- Looks:**
 - say Hello! for 2 seconds
 - say Hello!
 - think Hmm... for 2 seconds
- Sound:**
 - play sound Meow until done
 - start sound Meow
 - stop all sounds

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

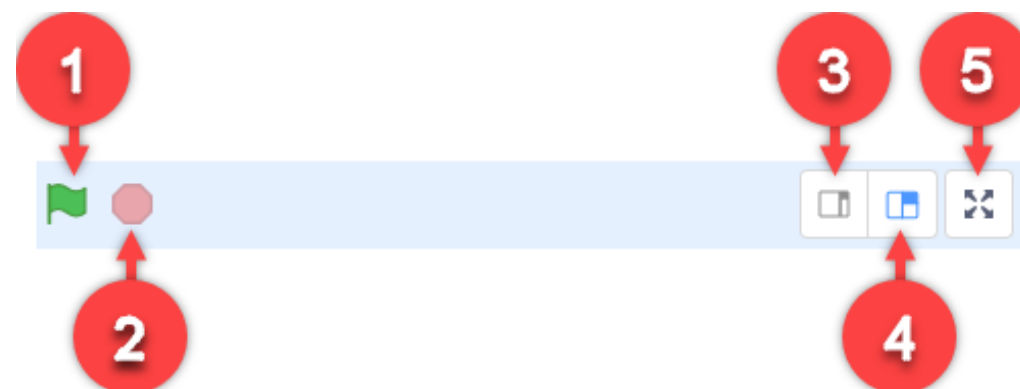
5. พื้นที่สำหรับเขียน Scripts



ส่วนประกอบและเครื่องมือโปรแกรม Scratch

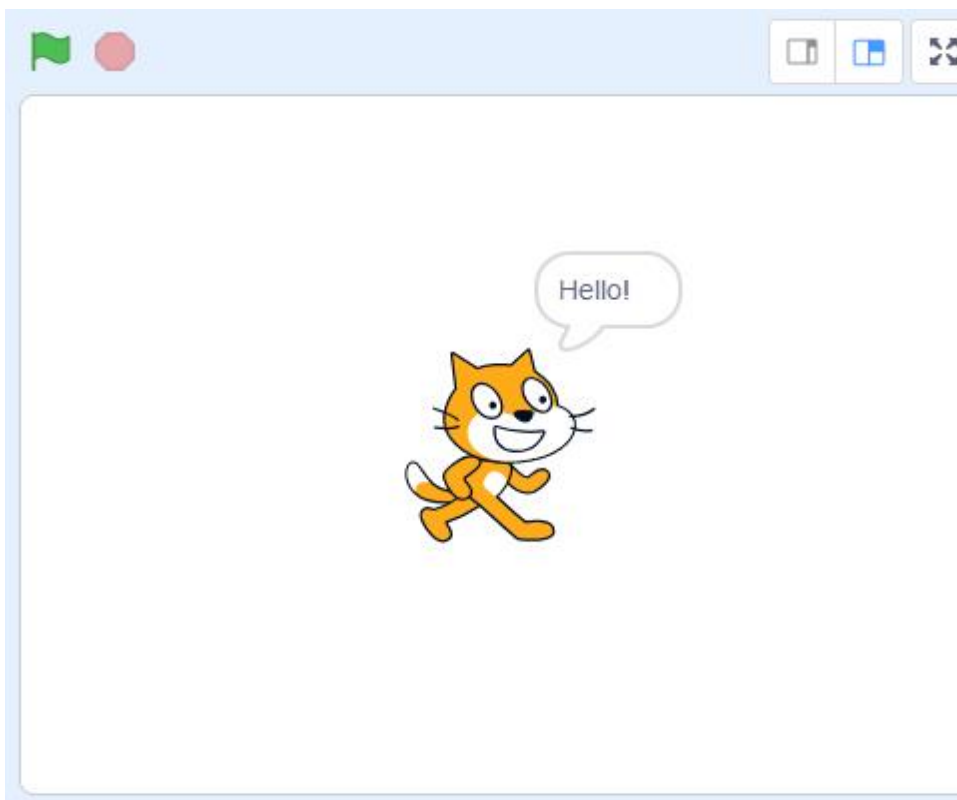
6. แถบเครื่องมือด้านบน Stage

1. เริ่มการทำงานของโปรแกรมที่เขียน
2. หยุดการทำงานของโปรแกรม
3. ปรับหน้าต่างโปรแกรม โฟกัสที่ Scripts
4. ปรับหน้าต่างโปรแกรม โฟกัสที่ Balance
5. ปรับหน้าต่างโปรแกรม ขยายจอ โฟกัสที่ Stage



ส่วนประกอบและเครื่องมือโปรแกรม Scratch

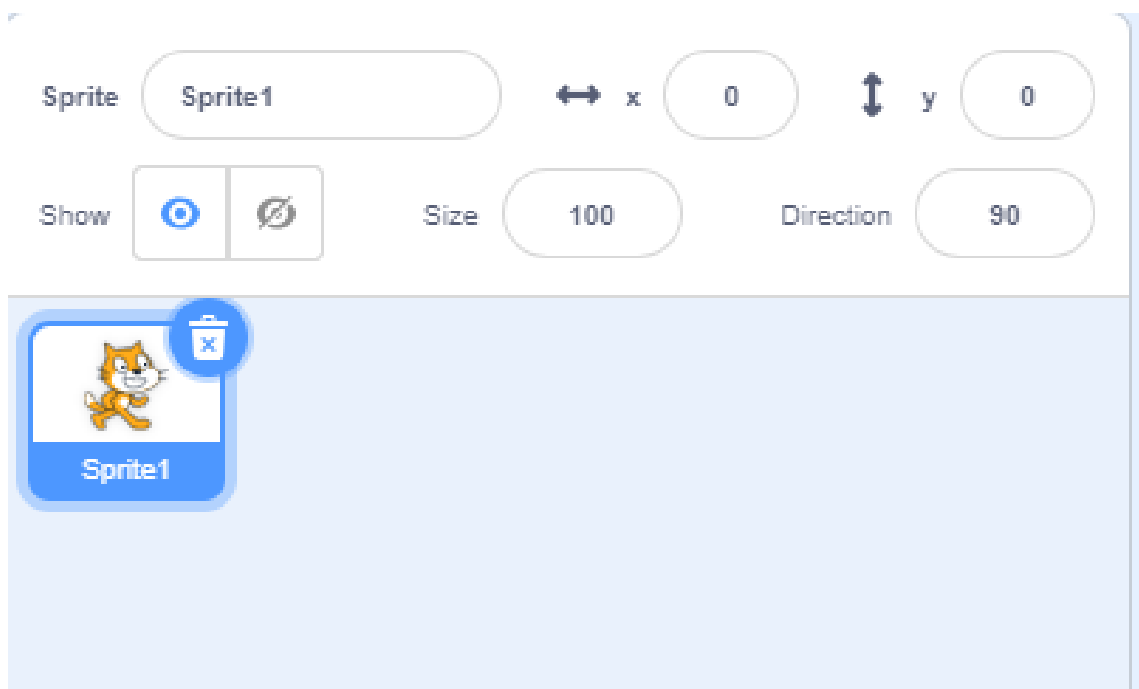
7. Stage (เวที)



เป็นพื้นที่แสดงผลพร้งของชิ้นงานที่สร้างขึ้น มีขนาดกว้าง 480 และสูง 360 โดยที่จุดกึ่งกลางเป็นจุดอ้างอิงที่ $x=0$ และ $y=0$ โดยค่า x เป็นตัวกำหนดตำแหน่งในแนวนอน มีค่าตั้งแต่ -240 ถึง 240 และ y เป็นตัวกำหนดตำแหน่งในแนวตั้ง มีค่าตั้งแต่ -180 ถึง 180

ส่วนประกอบและเครื่องมือโปรแกรม Scratch

8. Sprites List (ตัวละคร)



แสดงรายการของตัวละครและพื้นหลัง โดยชิ้นงานใหม่จะเริ่มด้วยตัวละครแมวสีส้ม และพื้นหลังสีขาว การคลิกที่ Sprite ใดจะเป็นการทำงานกับ Sprite นั้นทั้งการเขียนคำสั่งและแก้ไขตัวละคร โดย Sprite ที่ถูกเลือกจะปรากฏกรอบสีน้ำเงินล้อมรอบ ส่วนการคลิกที่ Backdrop หรือพื้นหลังจะเป็นการสลับไปทำงานกับพื้นหลังแทน

Block Programming (บล็อกคำสั่งต่าง ๆ)

เป็นส่วนของคำสั่งที่เราจะนำไปประกอบกันเป็นโปรแกรมแบ่งออกเป็นหมวดหมู่ต่าง ๆ ตามประเภทการใช้งานและส่วนต่อบล็อกคำสั่ง



Motion



Looks



Sound



Events



Control



Sensing



Operators

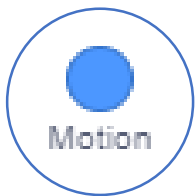


Variables



กลุ่มบล็อก Motions

กลุ่มบล็อก Motion ประกอบด้วยบล็อกที่ใช้เพื่อกำหนดรูปแบบการเคลื่อนที่ให้ตัวละคร เช่น เคลื่อนที่ไปข้างหน้า หมุน ไปที่ตำแหน่งต่าง ๆ บนเวที โดยสามารถกำหนดค่าของการเคลื่อนที่ได้ บล็อกในกลุ่มบล็อก Motion มีดังนี้

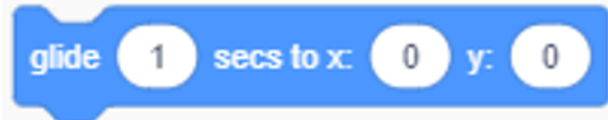


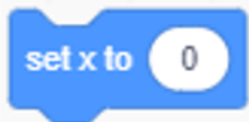

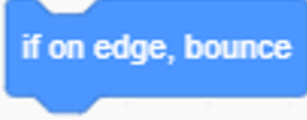


กลุ่มบล็อก Motions

บล็อก	ความหมาย
	ไปข้างหน้าหรือถอยหลัง
	หมุนตามเข็มนาฬิกา
	หมุนทวนเข็มนาฬิกา
	ระบุทิศทางที่ต้องการ
	ระบุทิศทางตามที่กำหนด
	ไปที่ตำแหน่ง (x, y) บนเวที
	ไปตำแหน่งที่กำหนดแบบสุ่ม บนเวที



กลุ่มบล็อก Motions

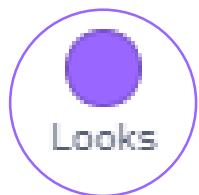
บล็อก	ความหมาย
	เคลื่อนไปที่ตำแหน่ง (x, y) บนเวทีในเวลาที่กำหนด ตัวเลขยิ่งมากยิ่ง ช้า
	เปลี่ยนตำแหน่งของตัวละครไปด้านซ้ายหรือขวา
	เปลี่ยนตำแหน่งของตัวละครไปด้านบนหรือ
	ตั้งค่าตำแหน่งตัวละครไปด้านซ้ายหรือขวาตามแนวแกน x
	ตั้งค่าตำแหน่งตัวละครไปด้านบนหรือด้านล่างตามแนวแกน Y
	ตัวละครชนขอบให้สะท้อนกลับ



กลุ่มบล็อก Motions



บล็อก	ความหมาย
<input type="checkbox"/> x position	ตำแหน่งตัวละครตามแนวแกน x ถ้าต้องการให้แสดงค่าตำแหน่งของตัวละครตามแนวแกน x ให้คลิกที่ ช่องสี่เหลี่ยม จะปรากฏเครื่องหมาย ✓ ซึ่งแสดงตำแหน่งบนเวที
<input type="checkbox"/> y position	ตำแหน่งตัวละครตามแนวแกน y ถ้าต้องการให้แสดงค่าตำแหน่งของตัวละครตามแนวแกน y ให้คลิกที่ ช่องสี่เหลี่ยม จะปรากฏเครื่องหมาย ✓ ซึ่งแสดงตำแหน่งบนเวที
<input type="checkbox"/> direction	ทิศทางตัวละคร ถ้าต้องการให้แสดงค่าทิศทางตัวละครให้คลิกที่ช่องสี่เหลี่ยมจะ ปรากฏเครื่องหมาย ✓ ซึ่งแสดงทิศทางบนเวที



กลุ่มบล็อก Looks

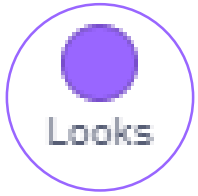
กลุ่มบล็อก Looks ใช้สำหรับการเขียนสคริปต์เพื่อสั่งให้ตัวละครหรือเวทีแสดงคุณสมบัติต่าง ๆ เช่น ตัวละครพูด เปลี่ยนชุดตัวละคร เปลี่ยนสีตัวละคร เปลี่ยนขนาดตัวละคร เปลี่ยนพื้นหลัง บล็อกในกลุ่มบล็อก Looks ของตัวละคร มีดังนี้



กลุ่มบล็อก Looks

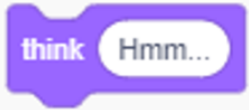
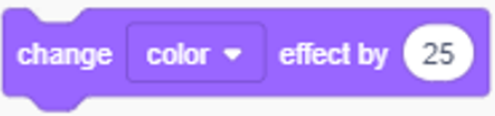
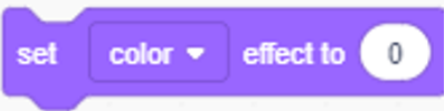
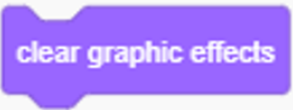
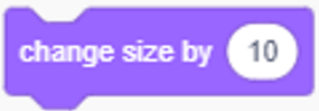
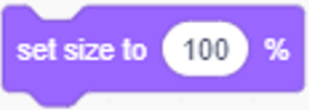


บล็อก	ความหมาย
	สลับชุดตัวละคร ตัวอย่าง สลับชุดตัวละครเป็นชุด costume2
	เปลี่ยนชุดตัวละครเป็นชุดถัดไปที่มีอยู่ในรายการ
	ชุดตัวละครชุดที่... ถ้าต้องการให้แสดงชุดตัวละครให้คลิกที่ช่องสีเหลี่ยมจะปรากฏ เครื่องหมายถูก ซึ่งแสดงเลขชุดตัวละครบนเวที
	แสดงคำพูดตามเวลาที่กำหนด
	แสดงคำพูด
	แสดงความคิดตามเวลาที่กำหนด



กลุ่มบล็อก Looks



บล็อก	ความหมาย
	แสดงความคิด
	เปลี่ยนเทคนิคการแสดงผลตัวละคร โดยเลือกเทคนิคพิเศษ จากเครื่องหมาย
	ตั้งค่าเทคนิคการแสดงผลตัวละครตามที่กำหนด
	ล้างค่าเทคนิคการแสดงผลตัวละครที่กำหนด
	เปลี่ยนขนาดตัวละครตามที่กำหนด ถ้าค่า + จะเป็นการเพิ่ม ขนาด ค่า - เป็นการลดขนาด
	ขนาดตัวละครเป็น %



กลุ่มบล็อก Looks



บล็อก	ความหมาย
	ขนาดของตัวละครถ้าต้องการให้แสดงค่าขนาดตัวละครบนเวทีให้คลิกที่ช่องสี่เหลี่ยมจะปรากฏเครื่องหมายถูก
	แสดงตัวละคร
	ซ่อนตัวละคร
	นำตัวละครไปไว้ด้านหน้าสุด
	ย้ายตัวละครไปไว้ด้านหน้าตามจำนวนเลเยอร์ (ชั้น) ที่กำหนด
	สลับพื้นหลัง
	สลับพื้นหลังเป็นพื้นหลังถัดไปที่มีอยู่ในรายการ
	พื้นหลังที่... ถ้าต้องการให้แสดงหมายเลขพื้นหลังบนเวทีให้คลิกที่ช่องสี่เหลี่ยมจะ ปรากฏเครื่องหมายถูก



กลุ่มบล็อก Sound



กลุ่มบล็อก Sound ทำงานเกี่ยวกับการแสดงเสียงต่าง ๆ ซึ่งอาจเป็นเสียงจากไฟล์ที่มีอยู่ หรือไฟล์เสียงที่ บันทึกโดยเครื่องมือบันทึกเสียง (Sound Recorder) ซึ่งอยู่ในโปรแกรม Scratch ไฟล์เสียงจะมีส่วนขยายเป็น .wav หรือ .mp3 นอกจากนี้ ยังมีบล็อกที่สามารถกำหนดเสียงของเครื่องดนตรีชนิดต่าง ๆ ได้ มีดังนี้



กลุ่มบล็อก Sound



บล็อก	ความหมาย
	เล่นเสียง
	เล่นเสียงจนจบ แล้วทำคำสั่งถัดไป
	หยุดเสียงทั้งหมด
	เปลี่ยนเสียง Effect เสียงสูงต่ำ ทีละ 10
	ตั้งค่าเสียง Effect เสียงสูงต่ำ เป็น 100
	ล้าง Effect
	เพิ่มหรือลดระดับความดังของเสียง
	กำหนดความดังของเสียงเป็นเปอร์เซ็นต์



กลุ่มบล็อก Events



มีบล็อกคำสั่งที่ใช้จัดการเหตุการณ์ต่าง ๆ ที่เกิดขึ้นขณะที่สคริปต์กำลังทำงานรวมถึง
การเริ่มทำ และการหยุดทำงานของสคริปต์ด้วย ดังนี้



กลุ่มบล็อก Events

บล็อก	ความหมาย
	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้เมื่อคลิกที่ 
	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้เมื่อแป้นที่ระบุถูกกด
	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้เมื่อตัวละครนี้ถูกคลิก
	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้เมื่อพื้นหลังของเวทีที่ระบุถูกคลิก
	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้ เมื่อค่าที่ระบุเป็นไปตามเงื่อนไขที่กำหนด โดยสามารถเลือกค่าได้จาก loudness, time หรือ video motion



กลุ่มบล็อก Events



บล็อก	ความหมาย
	เริ่มต้นทำงานตามบล็อกคำสั่งที่อยู่ต่อจากบล็อกนี้ เมื่อได้รับสารที่ระบุ
	กระจายสาร (ตามชื่อสารที่เลือก) ไปยังทุกตัวละครและเวที
	กระจายสารออกไป ผู้ส่งจะหยุดและทำคำสั่งถัดไปก็ต่อเมื่อ ผู้รับสาร (ตัวละครหรือเวที) ทุกตัวทำงาน ตามบล็อกคำสั่งของตนเองจนเสร็จสิ้น



กลุ่มบล็อก Control

มีบล็อกคำสั่งที่ใช้ควบคุมทิศทางการทำงานของสคริปต์ การวนทำบล็อกคำสั่งซ้ำ ๆ การเลือกทำบล็อกคำสั่งตามเงื่อนไข การสั่งงานแบบวนซ้ำ เป็นการสั่งให้ตัวละครทำงานเหมือนเดิมหลายครั้ง ทำให้การเขียนสคริปต์สั้นลงแต่ทำงานเหมือนเดิม โดยใช้คำสั่ง repeat และ forever ซึ่งอยู่ในกลุ่มบล็อก Control มีดังนี้



กลุ่มบล็อก Control



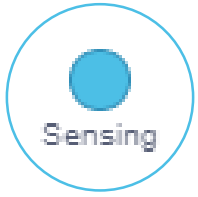
บล็อก	คำอธิบาย
	การทำงานซ้ำไม่รู้จบ
	การทำงานซ้ำตามจำนวนรอบที่กำหนด
	ใช้ตรวจสอบเงื่อนไข ถ้าเงื่อนไขหลัง if  เป็นจริง จึงจะทำคำสั่งภายในบล็อก if
	ใช้ตรวจสอบเงื่อนไข ถ้าเงื่อนไขหลัง if  เป็นจริง จะทำคำสั่งภายในบล็อก if แต่ถ้าเงื่อนไขเป็นเท็จ จะทำคำสั่งภายในบล็อก else



กลุ่มบล็อก Control



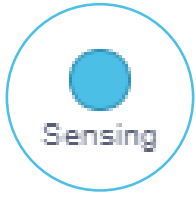
บล็อก	คำอธิบาย
	หยุดรอเวลา จนกระทั่งเงื่อนไขหลัง until  เป็นจริง จึงจะทำคำสั่งต่อไป
	หยุดคำสั่งทั้งหมด
	เริ่มการทำงานเมื่อมีการโคลน
	ทำการโคลน
	ลบการโคลนทิ้ง
	ทำซ้ำจนกว่าเงื่อนไขจะเป็นจริง



กลุ่มบล็อก Sensing



กลุ่มบล็อก Sensing ใช้ตรวจสอบการรับรู้ต่าง ๆ เช่น การสัมผัส เสียง ระยะเวลา ซึ่งอยู่ในกลุ่มบล็อก Sensing มีดังนี้

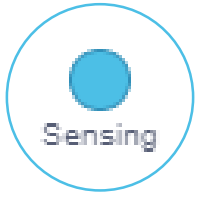


กลุ่มบล็อก Sensing



Sensing







บล็อก	ความหมาย
	ตรวจสอบว่าตัวละครนั้นสัมผัสกับ ตัวชี้เมาส์ ขอบเวที หรือตัวละคร อื่นหรือไม่
	ตรวจสอบว่าตัวละครนั้นสัมผัสกับสีที่กำหนดหรือไม่
	ตรวจสอบว่าสีตัวละครที่กำหนด สัมผัสกับสีที่ต้องการหรือไม่
	ถามคำถามที่กำหนด แล้วรอคำตอบ
	คำตอบที่ผู้ใช้กรอก จากคำถาม ask (สามารถคลิกที่ช่องเพื่อแสดงค่า คำตอบบนเวที)
	ตำแหน่งตัวชี้เมาส์ตามแนวแกน x
	ตำแหน่งตัวชี้เมาส์ตามแนวแกน y



กลุ่มบล็อก Sensing



Sensing

บล็อก	ความหมาย
	ตรวจสอบเมาส์ว่าถูกคลิกหรือไม่
	ตรวจสอบการกดแป้นพิมพ์บนคีย์บอร์ด
	เริ่มเวลาใหม่ เป็น 0
	เวลา
	ข้อมูลตัวละคร เช่น ตำแหน่งตามแนวแกน x, ตำแหน่งตามแนวแกน y, มุม, ลำดับชุดตัวละคร, ขนาด, ระดับความดังของเสียง
	ระดับเสียง (สามารถคลิกที่ช่องเพื่อแสดงค่าระดับของเสียงบนเวที)



กลุ่มบล็อก Operators



ภายในบล็อก Operators จำนวนพื้นฐาน มีตัวดำเนินการต่างๆ ที่ใช้ในการคำนวณ เช่น บวก ลบ คูณหาร มอดดูเลก ซึ่งอยู่ในกลุ่มบล็อก Operators มีดังนี้



กลุ่มบล็อก Operators



บล็อก	ความหมาย
	บวก
	ลบ
	คูณ
	หาร
	สุ่มตัวเลข
	มากกว่า
	น้อยกว่า
	เท่ากัน



กลุ่มบล็อก Operators



บล็อก	ความหมาย
	เชื่อมข้อความในช่องกำหนดค่าที่ 1 และ 2
	เลือกอักขระในลำดับที่ระบุจากค่าที่กำหนด
	นับจำนวนอักขระ
	ตรวจตัวอักษรในประโยค
	มอดดูเลข
	ปัดเศษ
	ฟังก์ชันทางคณิตศาสตร์





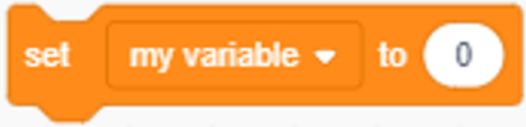
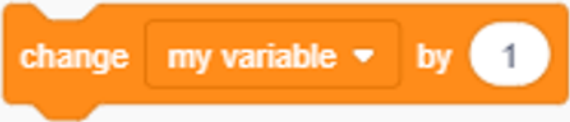


กลุ่มบล็อก Variables

ภายในบล็อก Variables คือส่วนที่ใช้ในการจัดการตัวแปรต่างๆ มีดังนี้



กลุ่มบล็อก Variables

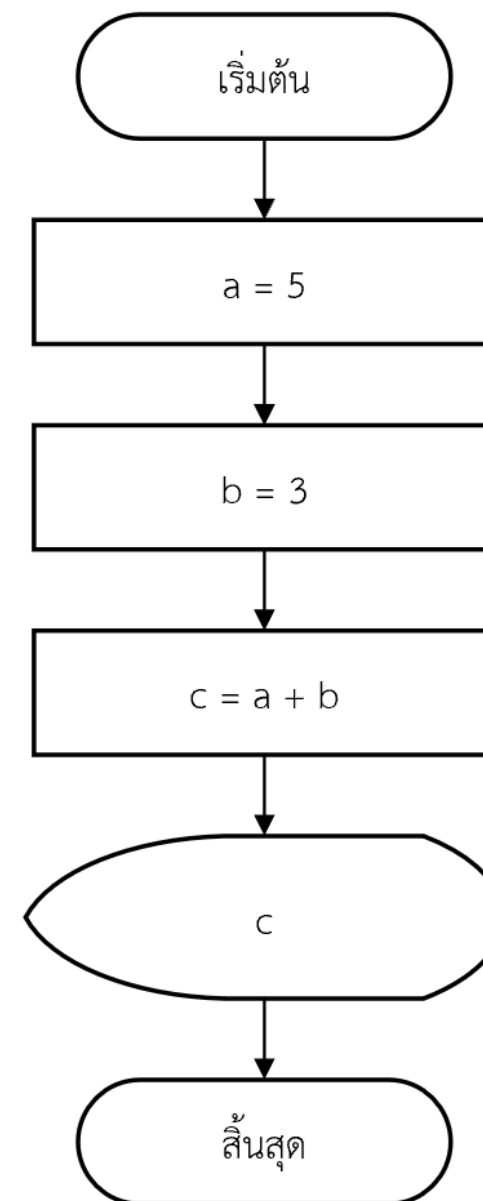


บล็อก	ความหมาย
	สร้างตัวแปร
	ตัวแปร
	ตั้งค่าตัวแปร
	เปลี่ยนค่าตัวแปร
	แสดงตัวแปร
	ซ่อนตัวแปร

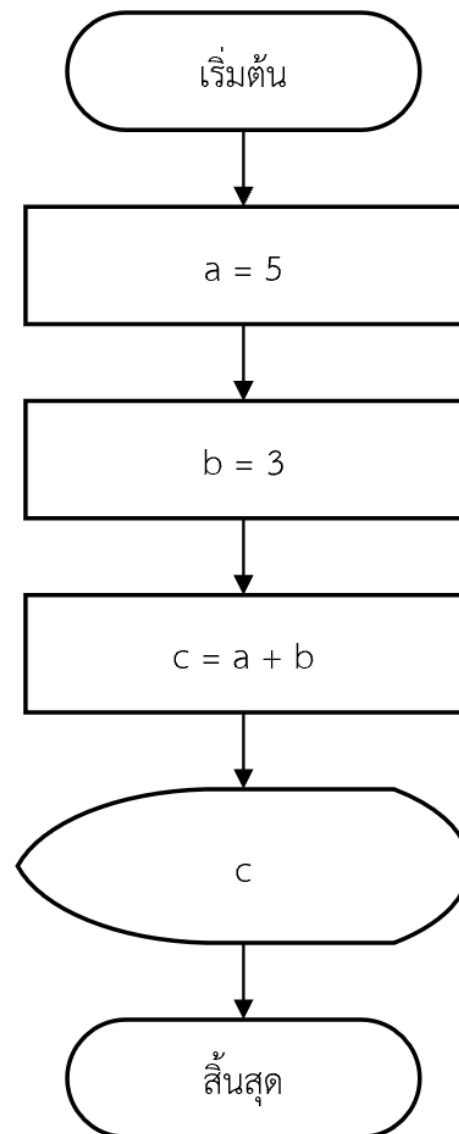
การเขียนโปรแกรมแบบลำดับ

โครงสร้างแบบลำดับ (Sequential Structure)

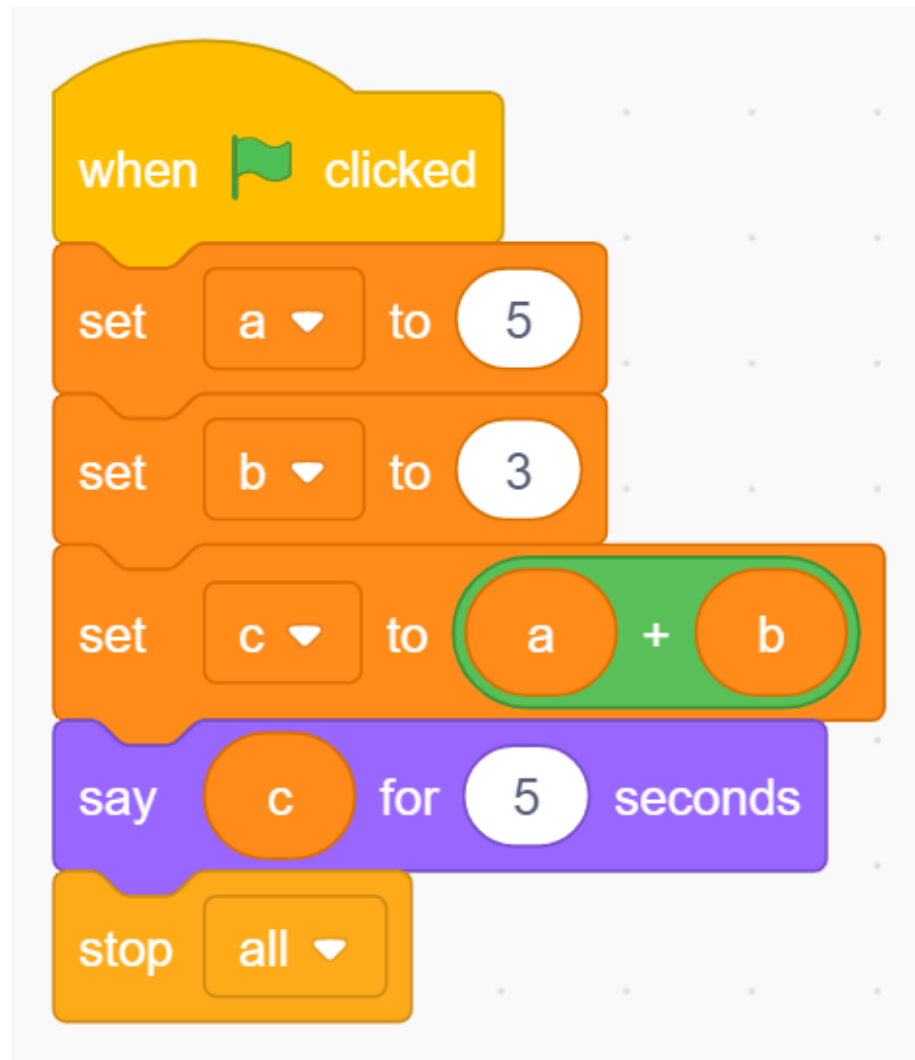
เป็นลักษณะโครงสร้างพื้นฐานของผังงานทุกผังงาน
ลักษณะการทำงานของโครงสร้างผังงานแบบลำดับจะ
ทำงานทีละขั้นตอน ตั้งแต่ขั้นตอนแรกจนถึงขั้นตอน
สุดท้าย ลำดับการทำงานของผังงานจะทำงานตาม
ทิศทางของลูกศร โครงสร้างของผังงานแบบลำดับ มี
ลักษณะโครงสร้างดังนี้



กิจกรรมที่ 1 จงเขียนโปรแกรมตามผังงานแบบลำดับ



กิจกรรมที่ 1 รูปแบบ Block โปรแกรมแบบลำดับ



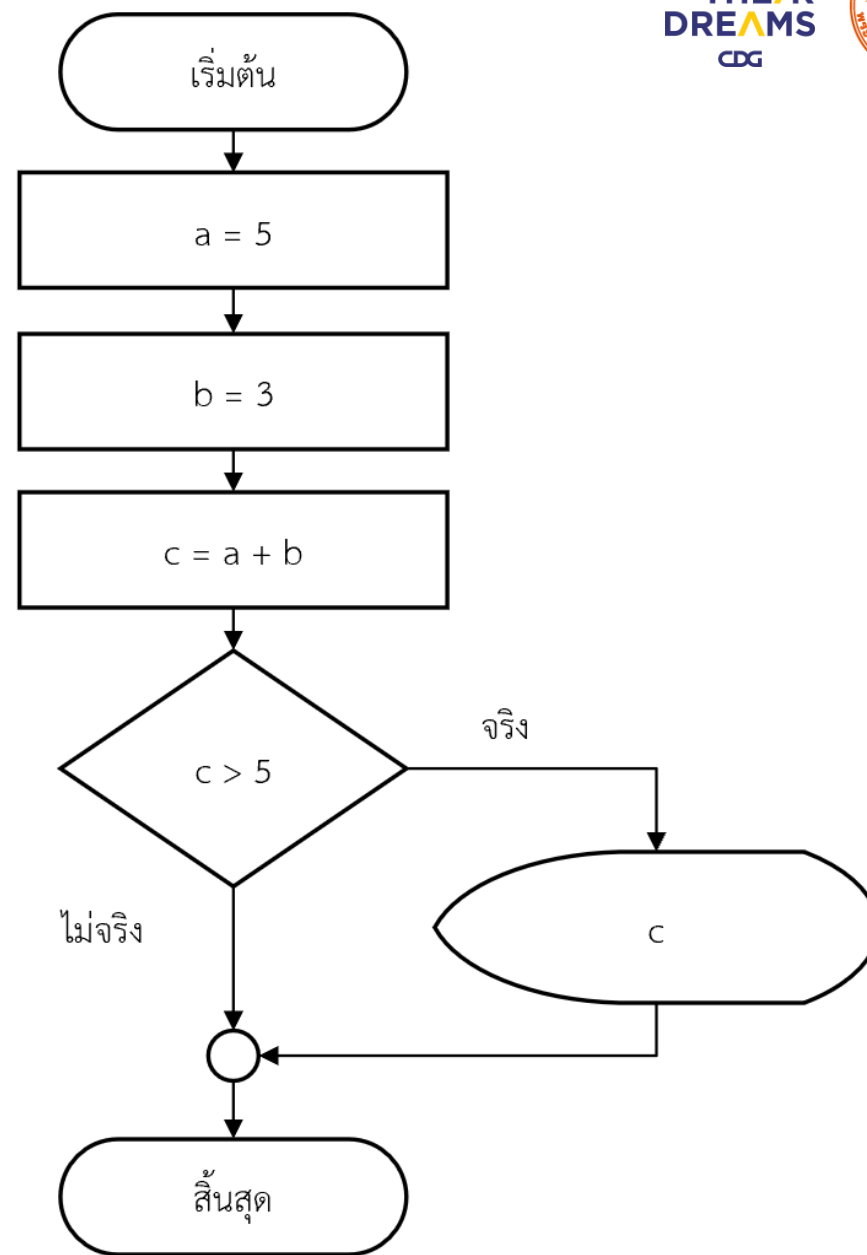
การเขียนโปรแกรมแบบทางเลือก

โครงสร้างแบบทางเลือก (Selection Structure) หมายถึง โครงสร้างที่มีเงื่อนไข ขั้นตอนการทำงานบางขั้นตอน ต้องมีการตัดสินใจ โครงสร้างผังงานการทำงานแบบเลือกทำ ใช้สำหรับกรณีที่ต้องการตัดสินใจเพื่อเลือกขั้นตอนการทำงานที่เหมาะสมกับข้อมูลที่ทำการประมวลผลในขณะนั้น โครงสร้างผังงานการเลือกทำประกอบด้วย สัญลักษณ์ของการตัดสินใจ 1 สัญลักษณ์ เพื่อใช้สำหรับการตัดสินใจสำหรับเลือกการทำงานที่ต้องทำเป็นลำดับถัดไป

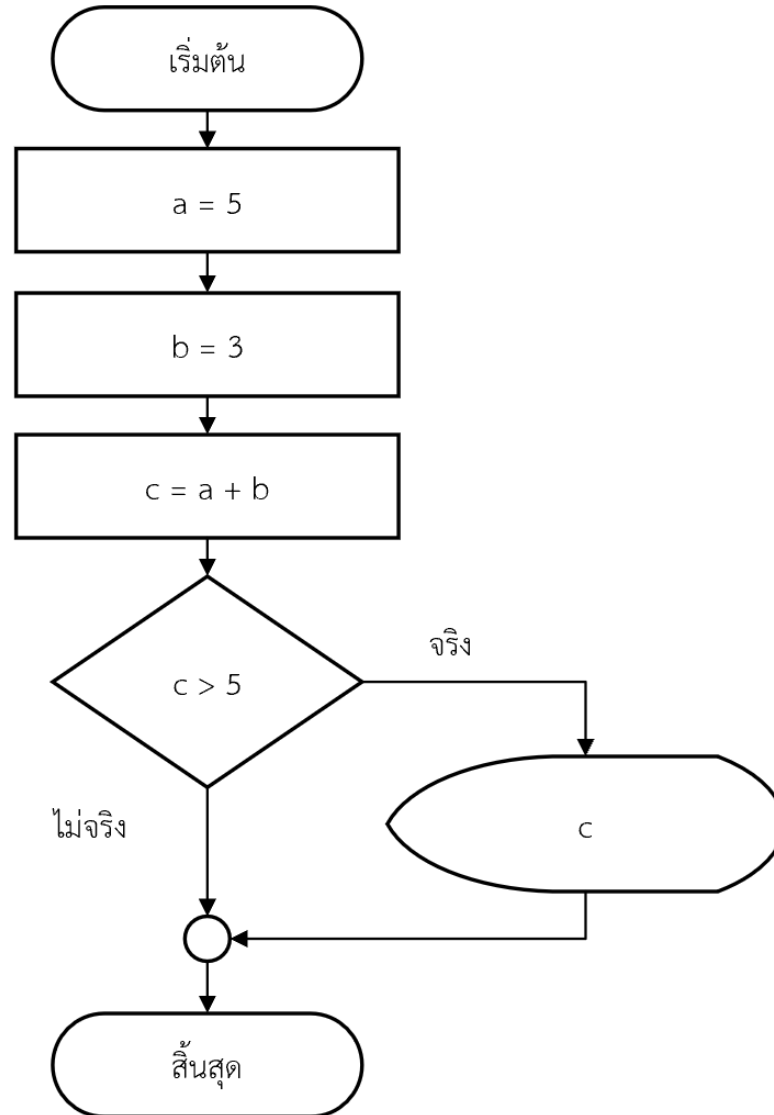
การทำงานหลังจากการตรวจสอบเงื่อนไข จะมีการทำงานอยู่ 3 กรณี

การเขียนโปรแกรมแบบทางเลือกเดียว

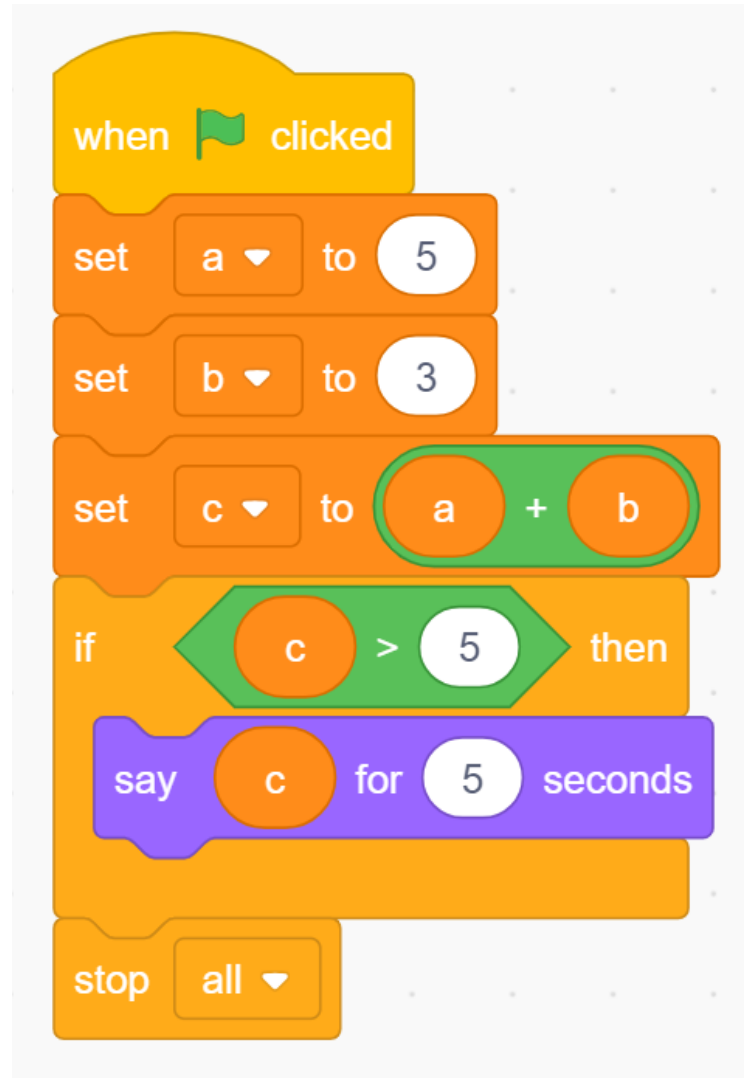
1. กรณีที่มีการทำงานเพียงขั้นตอนเดียว ในกรณีที่ผลการตรวจสอบเงื่อนไขเป็นจริง หรือเป็นเท็จ เพียงอย่างเดียวอย่างหนึ่งเท่านั้น



กิจกรรมที่ 2 จงเขียนโปรแกรมตามผังงานแบบทางเลือกเดียว

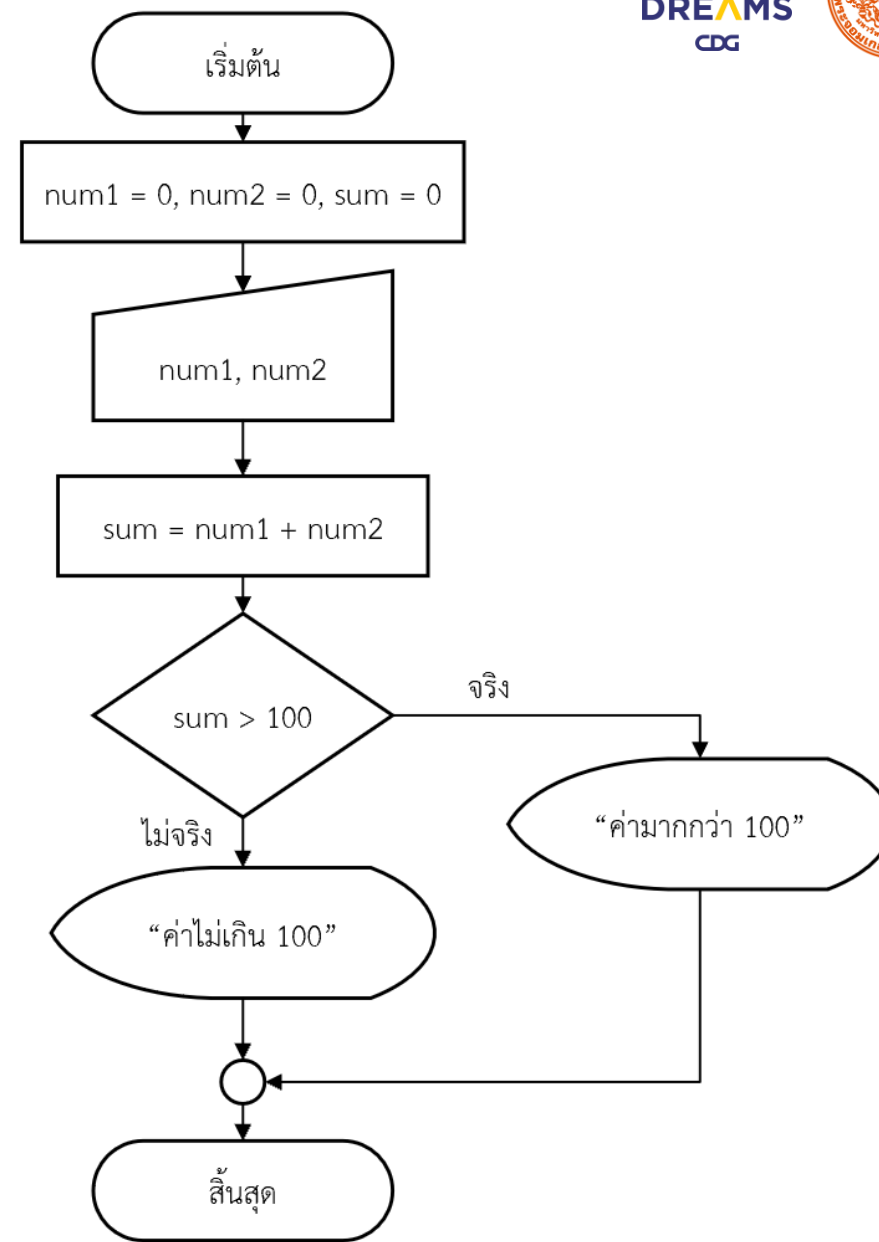


กิจกรรมที่ 2 รูปแบบ Block โปรแกรมแบบทางเลือกเดียว

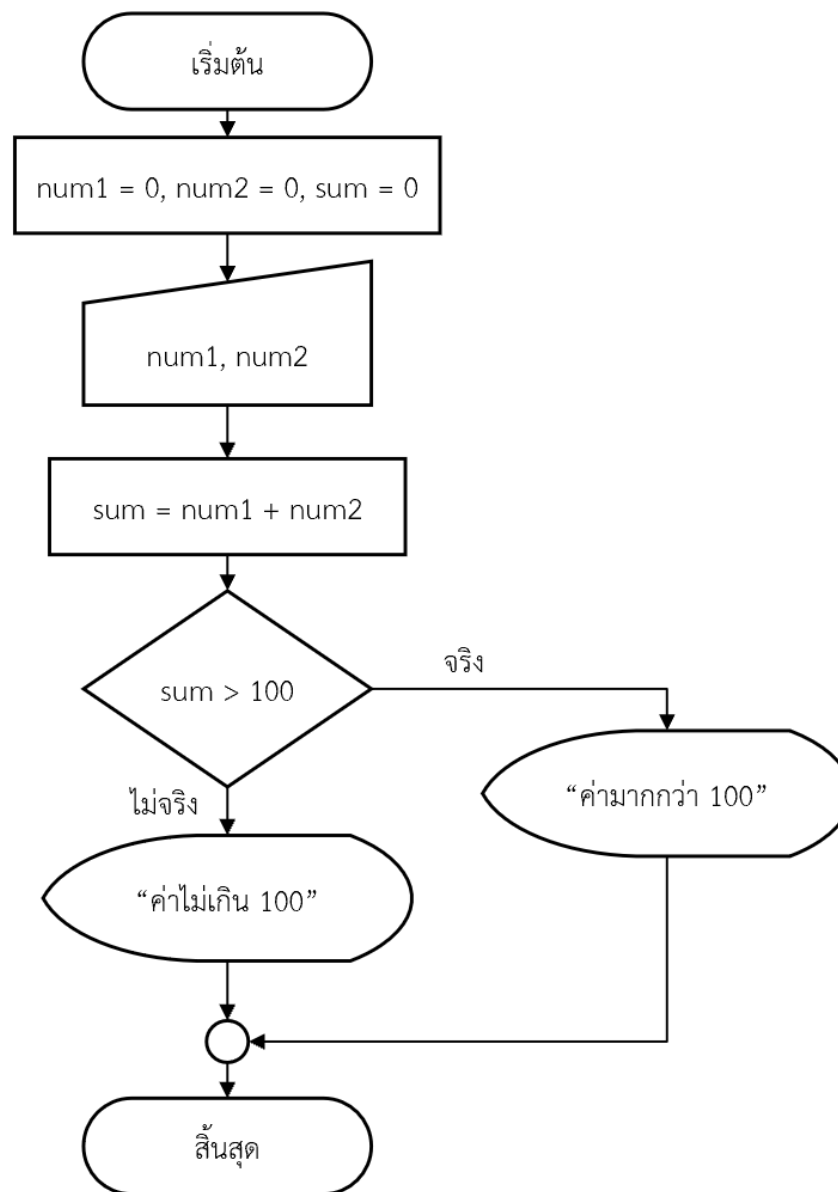


การเขียนโปรแกรมแบบสองทางเลือก

2. กรณีที่มีการทำงาน 2 ขั้นตอน คือ มีขั้นตอนการทำงานให้ ไม่ว่าจะผลการตรวจสอบเงื่อนไขจะเป็นจริงหรือเท็จ



กิจกรรมที่ 3 จงเขียนโปรแกรมตามผังงานแบบสองทางเลือก



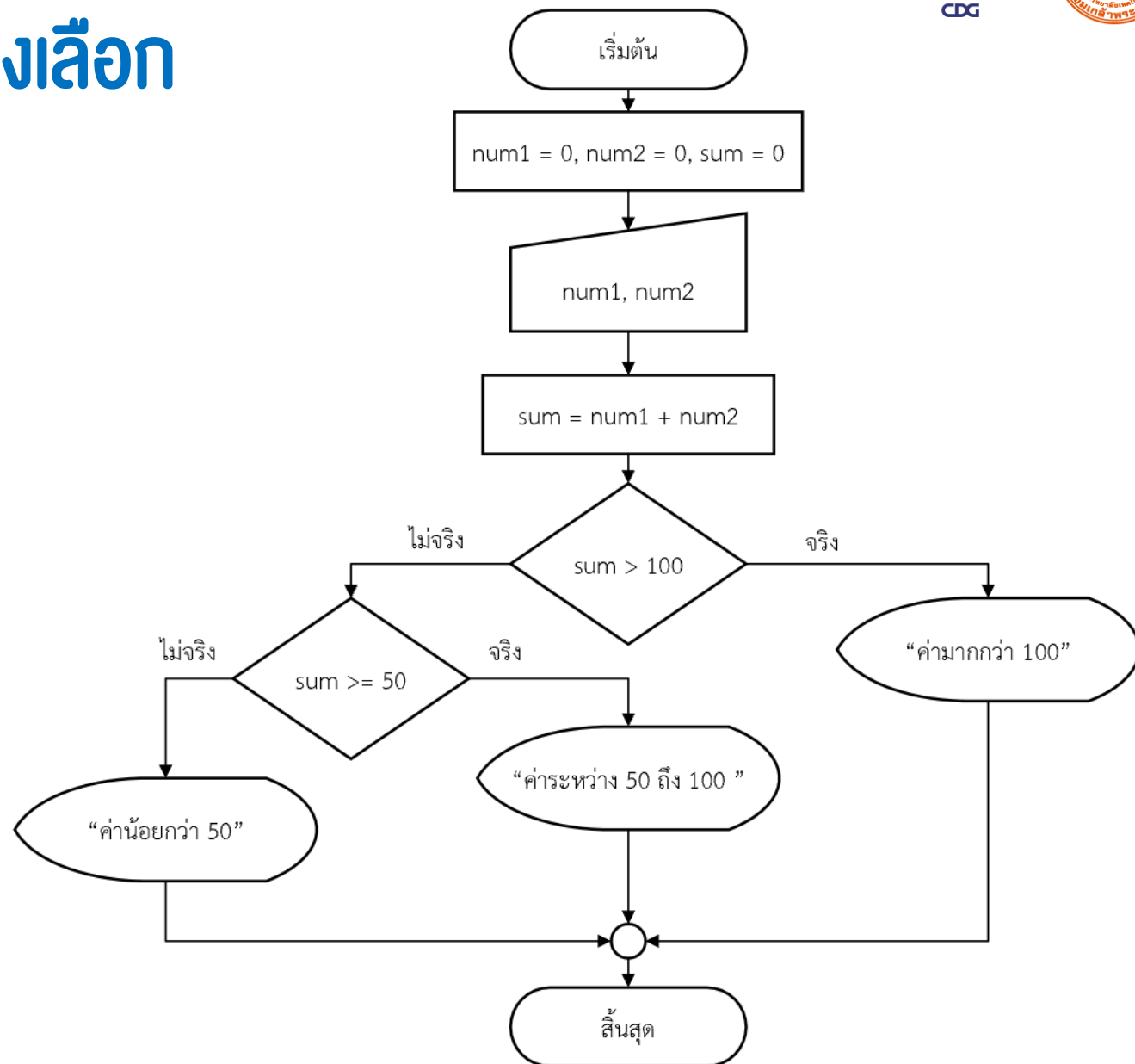
กิจกรรมที่ 3 รูปแบบ Block โปรแกรมแบบสองทางเลือก

```

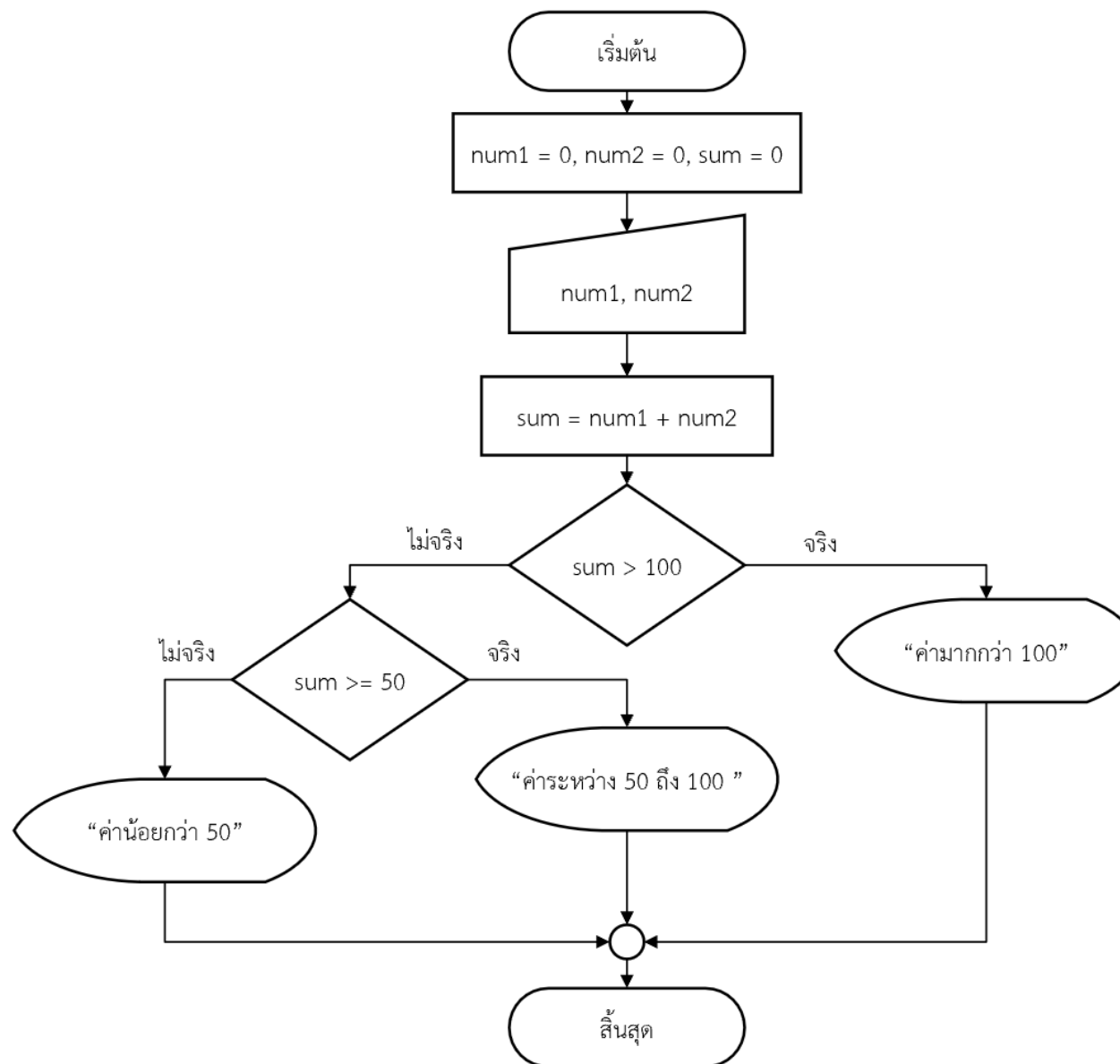
when clicked
  set num1 to 0
  set num2 to 0
  set sum to 0
  ask "กรอกค่าที่ 1" and wait
  set num1 to answer
  ask "กรอกค่าที่ 2" and wait
  set num2 to answer
  set sum to num1 + num2
  if sum > 50 then
    say "ค่ามากกว่า 100" for 5 seconds
  else
    say "ค่าไม่เกิน 100" for 5 seconds
  stop all
  
```

การเขียนโปรแกรมแบบหลายทางเลือก

3. กรณีที่มีการทำงาน 2 ขั้นตอนแต่มีมากกว่า 2 เงื่อนไข คือ มีขั้นตอนการทำงานให้ ไม่ว่าจะผลการตรวจสอบเงื่อนไขจะเป็นจริงหรือเท็จหากเป็นเท็จจะไปตรวจสอบเงื่อนไขอื่นถัดไปก่อนจะเข้าขั้นตอนการทำงานถัดไป



กิจกรรมที่ 4 จงเขียนโปรแกรมตามผังงานแบบหลายทางเลือก



กิจกรรมที่ 4 รูปแบบ Block โปรแกรมแบบหลายทางเลือก

```

when clicked
  set num1 to 0
  set num2 to 0
  set sum to 0
  ask "กรอกค่าที่ 1" and wait
  set num1 to answer
  ask "กรอกค่าที่ 2" and wait
  set num2 to answer
  set sum to num1 + num2
  if sum > 100 then
    say "ค่ามากกว่า 100" for 5 seconds
  else
    if sum > 50 or sum = 50 then
      say "ค่าระหว่าง 50 ถึง 100" for 5 seconds
    else
      say "ค่าน้อยกว่า 50" for 5 seconds
  stop all
  
```

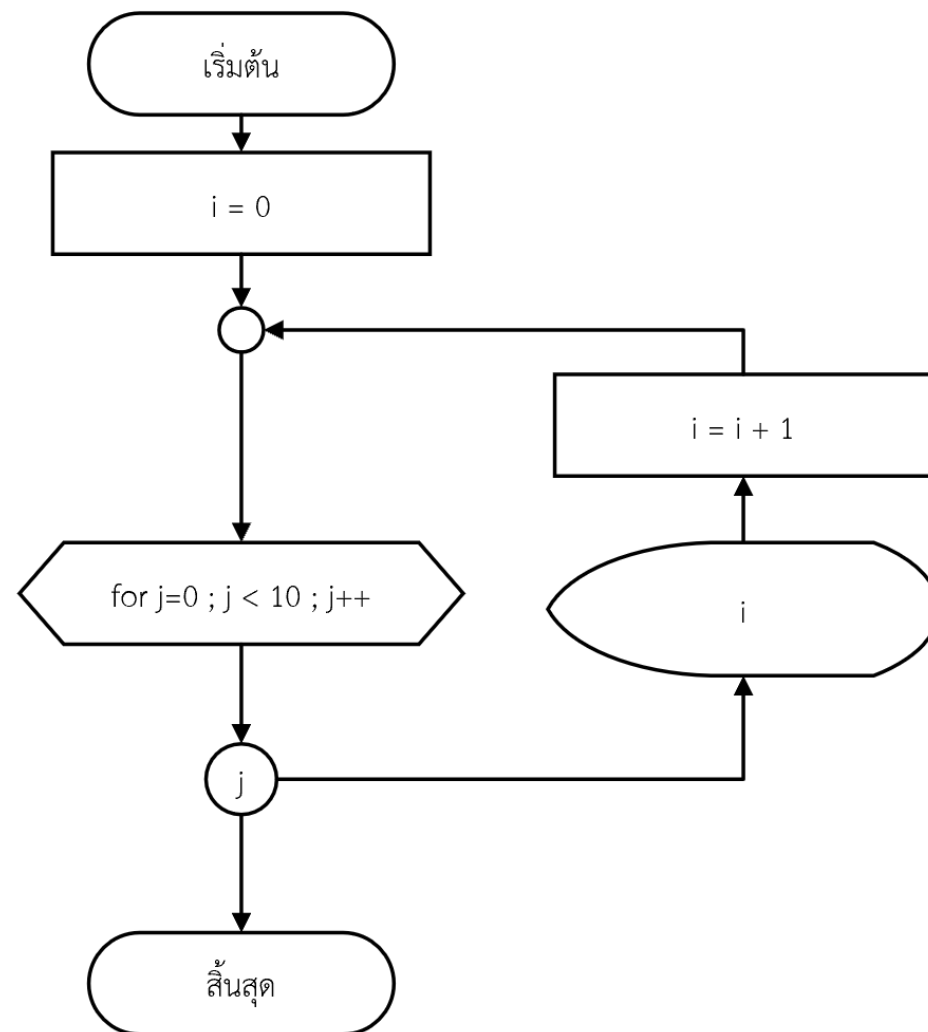
การเขียนโปรแกรมแบบวนซ้ำ

โครงสร้างแบบทำซ้ำ (Repetition Structure) ลักษณะโครงสร้างผังงานการทำงานแบบทำซ้ำ ใช้ประโยชน์ในกรณีที่ต้องการทำงานอย่างใดอย่างหนึ่งซ้ำกันหลายครั้ง โครงสร้างผังงานแบบทำซ้ำจะประกอบไปด้วยสัญลักษณ์

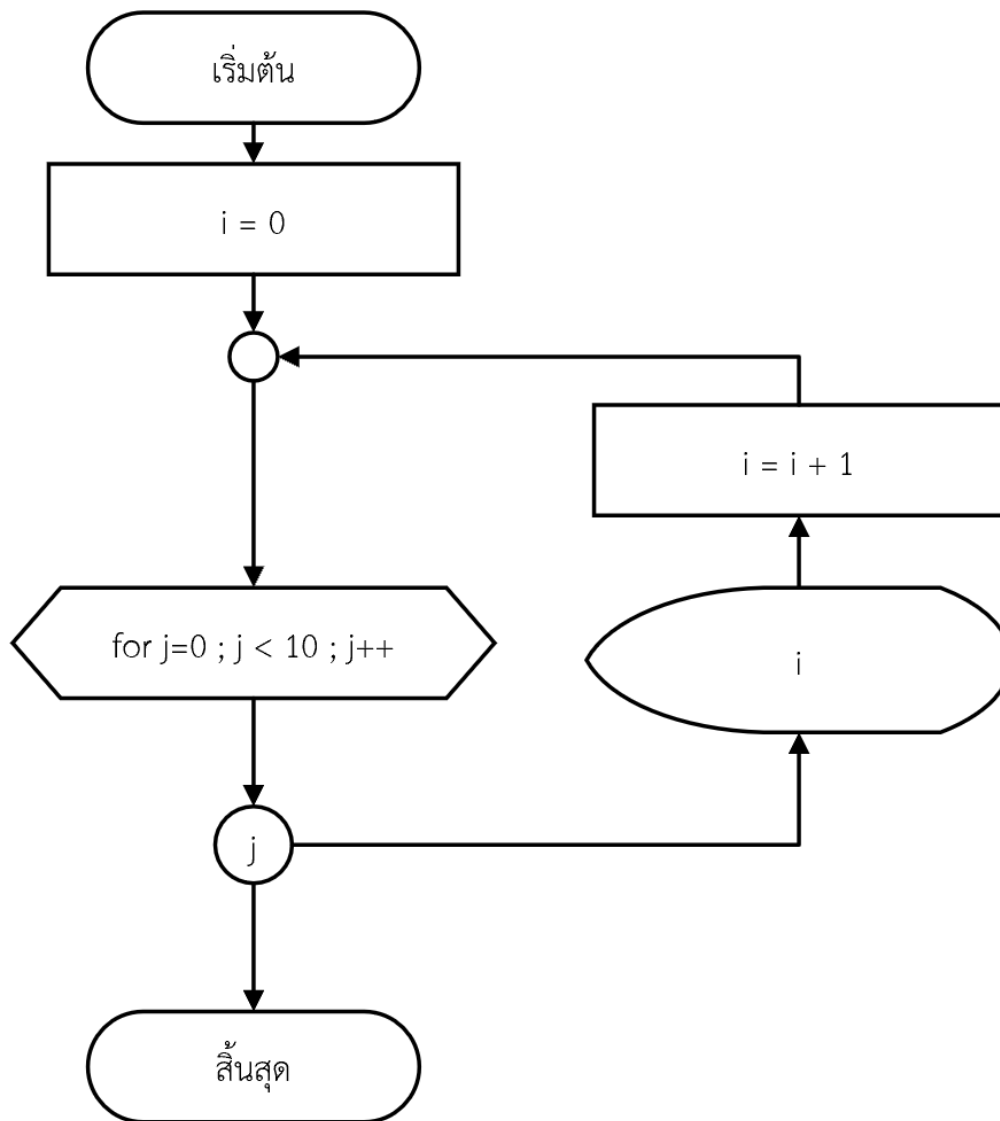
การตัดสินใจ ใช้สำหรับตรวจสอบเงื่อนไขเพื่อตัดสินใจว่า จะเข้าสู่ขั้นตอนการทำซ้ำหรือไม่ ลักษณะการทำซ้ำ สามารถแบ่งได้เป็น 2 ลักษณะคือ

การเขียนโปรแกรมวนซ้ำแบบกำหนดรอบไว้ล่วงหน้า

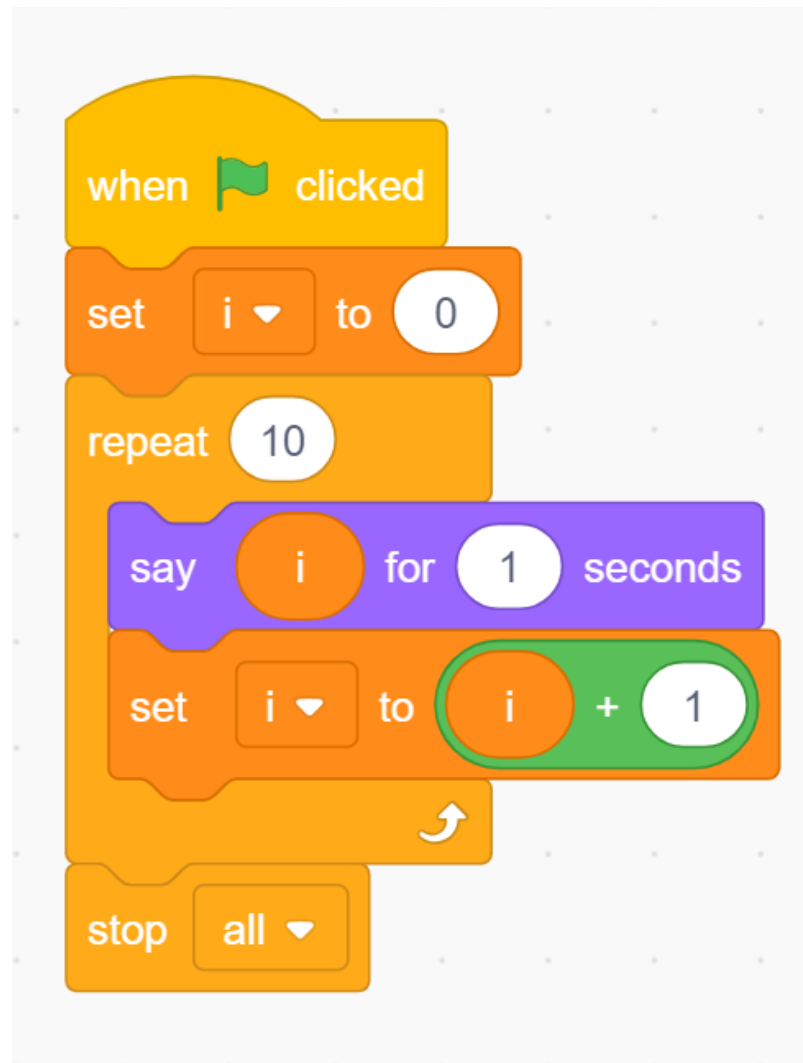
การทำงานของโครงสร้างผังงานการทำซ้ำ (for loop) ขั้นตอนแรกของการทำงาน คือ การตรวจสอบเงื่อนไขการทำซ้ำ ถ้าผลที่ได้จากการตรวจสอบเงื่อนไขเป็นจริง จึงเข้าสู่ขั้นตอนของการทำงานในส่วนของการทำซ้ำ หากเป็นเท็จจะข้ามไปทำงานในลำดับถัดไป



กิจกรรมที่ 5 จงเขียนโปรแกรมวนซ้ำตามผังงานแบบ กำหนดรอบไว้ล่วงหน้า



กิจกรรมที่ 5 รูปแบบ Block โปรแกรมทำซ้ำแบบ กำหนดรอบไว้ล่วงหน้า

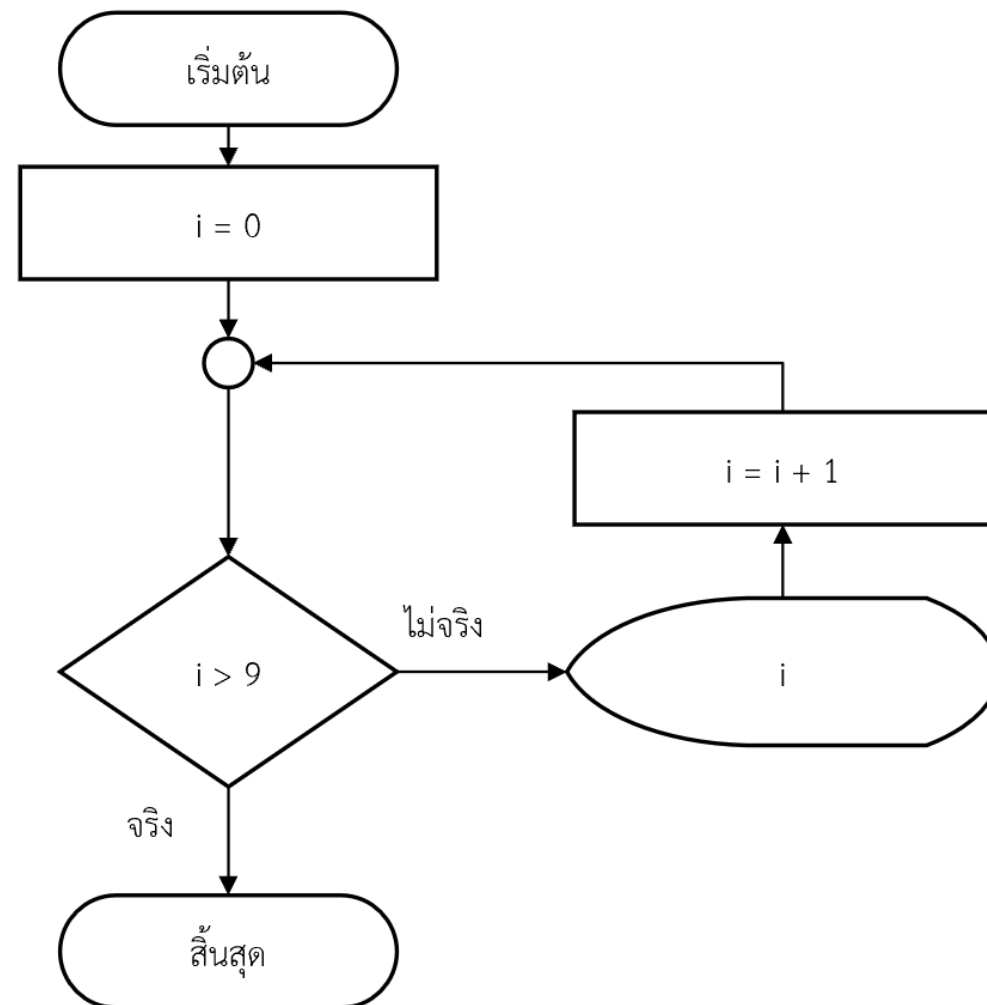


```
when clicked
set i to 0
repeat 10
  say i for 1 seconds
  set i to i + 1
stop all
```

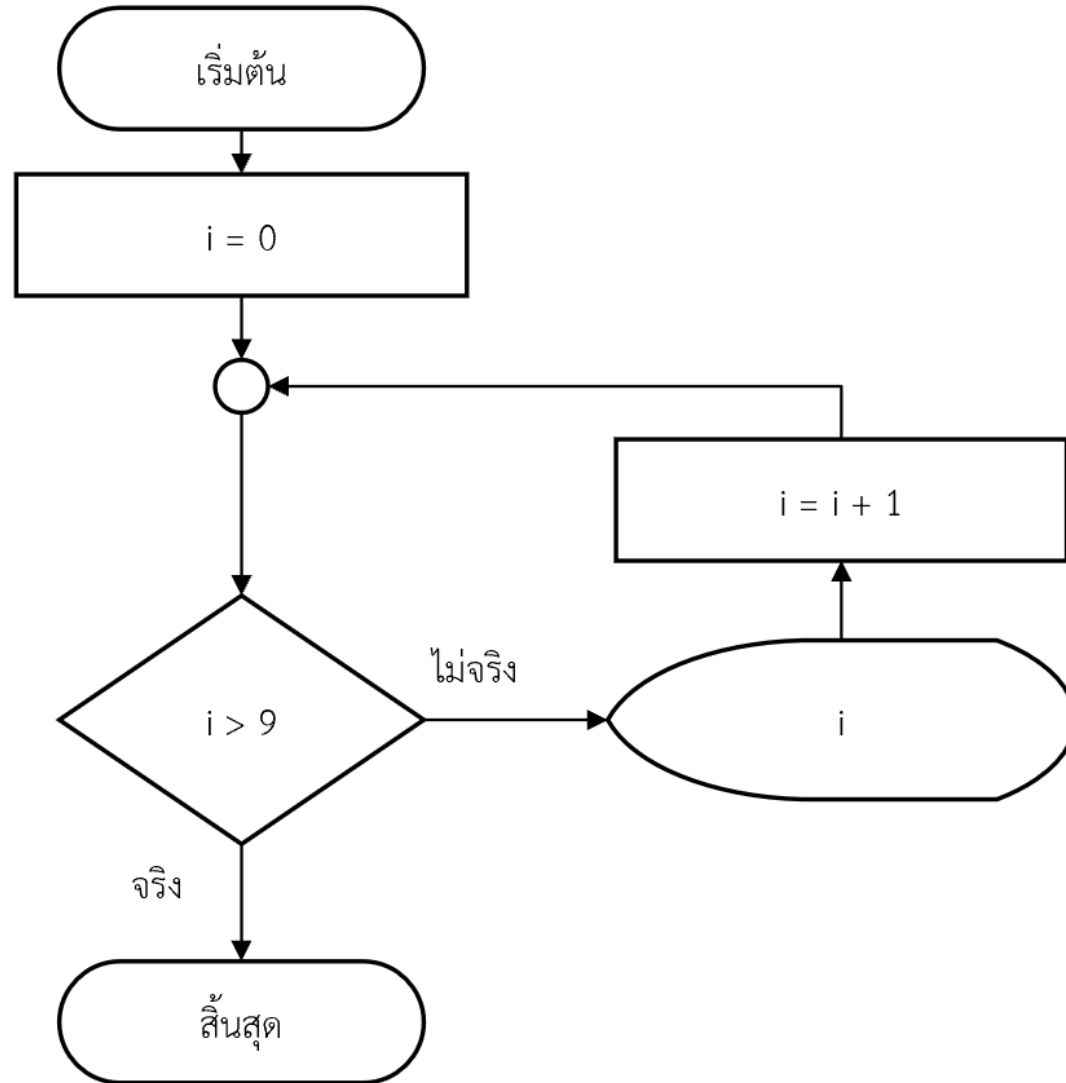
The image shows a Scratch script on a grid background. It starts with a yellow 'when clicked' block. This is followed by an orange 'set i to 0' block. Then, a large orange 'repeat 10' block contains two sub-blocks: a purple 'say i for 1 seconds' block and an orange 'set i to i + 1' block. The 'set i to i + 1' block has a green circle around the '+' sign. The 'repeat' block has a white arrow icon at the bottom right. Finally, there is an orange 'stop all' block at the bottom.

การเขียนโปรแกรมวนซ้ำแบบตรวจสอบเงื่อนไข

ลักษณะของการทำงานของผังงานการทำซ้ำลักษณะทำงานกระทั่ง (Do - Until) ขั้นตอนแรกของการทำงาน คือ การตรวจสอบเงื่อนไขการทำซ้ำ ถ้าผลที่ได้จากการตรวจสอบเงื่อนไขเป็นเท็จ จึงเข้าสู่ขั้นตอนของการทำงานในส่วนของการทำซ้ำ หากเป็นจริงจะข้ามไปทำงานในลำดับถัดไป



กิจกรรมที่ 6 จงเขียนโปรแกรมวนซ้ำตามผังงานแบบตรวจสอบเงื่อนไข



กิจกรรมที่ 6 รูปแบบ Block โปรแกรมทำซ้ำแบบตรวจสอบเงื่อนไข



```
when green flag clicked
  set i to 0
  repeat until i > 9
    say i for 1 seconds
    set i to i + 1
  stop all
```

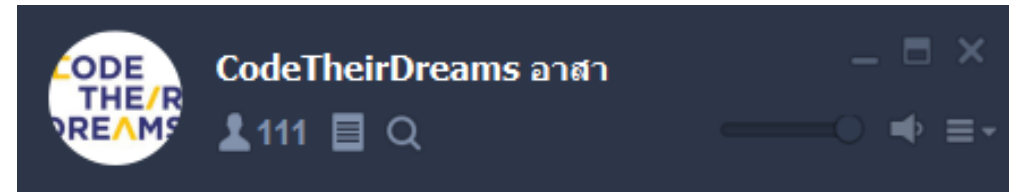
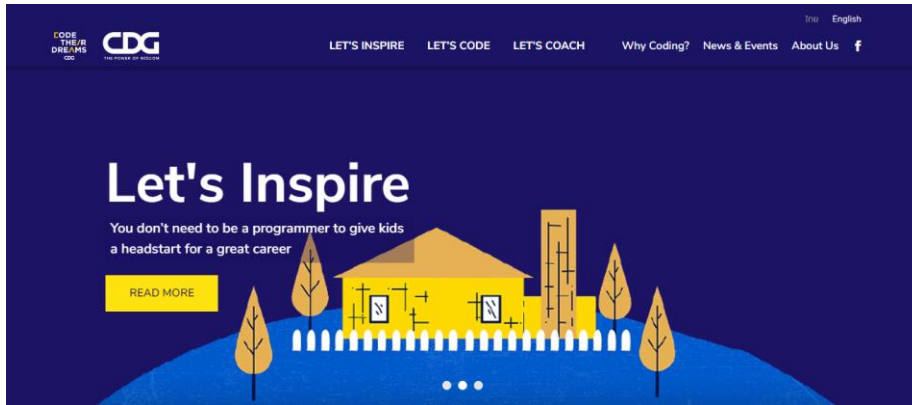
The image shows a Scratch script with the following blocks:

- when green flag clicked** (yellow)
- set i to 0** (orange)
- repeat until i > 9** (orange)
- say i for 1 seconds** (purple)
- set i to i + 1** (orange)
- stop all** (orange)

Code Their Dreams: Communication Channel

 www.codetheirdreams.com

 Line Group: CodeTheirDreams ภาษา



 Facebook Fanpage: CodeTheirDreams

 Youtube: Code Their Dreams

