

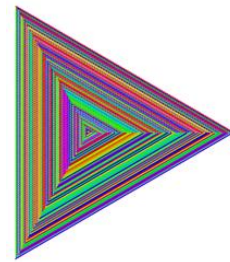
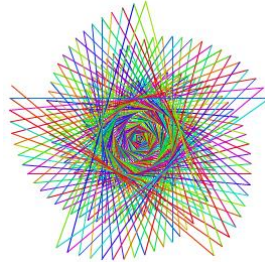
# Scratch ระดับปานกลาง



## การใช้ปากกา (Pen)

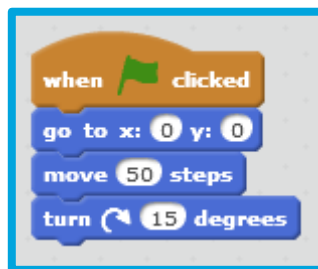
การ์ดแผ่นที่ 1 จาก 5  
ฉันทำล้งเรียน: Scratch

- 1 คุณคงได้เรียนรู้การใช้ Scratch เบื้องต้น และได้ลองสร้างเกมขึ้นมาแล้ว (แต่หากยัง ก็ให้ไปดู ชูชิการ์ด “Scratch เบื้องต้น”) สำหรับในชูชิการ์ดชุดนี้ เราจะมาเรียนรู้เคล็ดลับดีๆ เพิ่มอีกหน่อย และทดลองสร้างโปรเจกต์ใหม่ ที่ผมคิดว่าคุณก็คงจะชอบด้วย คือ การให้ Scratch วาดลวดลายที่มีสีสัน และรูปทรงต่างๆ ซึ่งถ้าคุณทำได้ถูกต้อง ก็จะได้เห็นภาพที่น่าตื่นตาทีเดียว

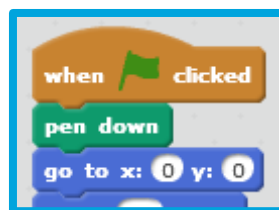


ในการวาดรูป เราจะต้องใช้ ปากกา หรือ Pen ซึ่งควบคุมด้วยบล็อกคำสั่งในกลุ่ม Pen โดยบล็อกเหล่านี้จะวาดเส้นที่จุดศูนย์กลางของสไปรท์ ในขณะที่มันเคลื่อนที่ ซึ่งเราจะได้มาเรียนรู้วิธีใช้กัน!

เริ่มด้วยการสร้างไฟล์ Scratch ขึ้นมา เลือกสไปรท์ที่เป็นแมว Scratch แล้วลากบล็อกต่างๆ มาวางตามตัวอย่างข้างล่างนี้ (หวังว่า คุณจะยังจำบล็อกเหล่านี้จากการ์ดชุด “Scratch เบื้องต้น” ได้นะ)



- 2 เยี่ยม! ทีนี้ ก็มาทดลองใช้ปากกากัน โดยให้ไปที่กลุ่ม Pen เลือกบล็อก pen down นำมาเพิ่มไว้ที่ตอนต้นของโปรแกรม ดังนี้



เสร็จแล้ว ก็ลองคลิกธงสีเขียวหลายๆ ครั้ง และดูว่าเกิดไรขึ้น!



This work is licensed under the Creative Commons Attribution 4.0 International License.

แปลและเรียบเรียงจากเอกสารของ CoderDojo โดย จิตรภณ และณรงค์พร เหล่าศรีสิน (fb.me/narongpom.lao)

# Scratch ระดับปานกลาง



## การใช้ปากกา (Pen)

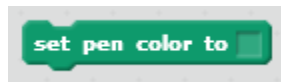
การ์ดแผ่นที่ 1 จาก 5  
ฉันทำล้งเรียน: Scratch

3 ถ้าคุณเห็นเส้นที่อยู่ด้านหลังของแมว ก็แสดงว่าปากกาใช้งานได้แล้ว ทีนี้ คุณก็สามารถจะวาดรูปหรือ ลวดลายสวยๆ ได้แล้ว!

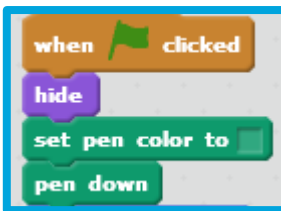
แต่ก่อนอื่น คุณควรจะเอาแมวออกไปก่อน! เพื่อไม่ให้มันมาเกะกะขณะกำลังวาดรูป โดยนำบล็อก **hide** จากกลุ่ม **Looks** มาวางไว้ที่ตอนต้นของโปรแกรม เท่านั้น แมวก็จะหายไปแล้ว



4 คราวนี้ ให้คุณลองเปลี่ยนสีปากกา โดยใช้บล็อกจากกลุ่ม Pen แต่บล็อกนี้จะแตกต่างกับบล็อกอื่นๆ ที่คุณเคยเจอ มันคือบล็อก **set pen color to** ซึ่งมีหน้าต่างแบบนี้



ให้คุณลากบล็อกนี้มาใส่ในแถบสปรอท โดยวางไว้ก่อนหน้าบล็อก **pen down**



ต่อไป ให้คุณคลิกที่ช่องสีในบล็อก (จากตัวอย่างข้างต้น ในช่องจะเป็น **สีเขียว**) เมื่อคลิกแล้ว คุณจะเห็นว่าเมาส์เปลี่ยนเป็นรูปมือ ให้คุณลองเลื่อนเมาส์ไปรอบๆ คุณจะเห็นว่า สีในช่องจะเปลี่ยนไปตามสีที่เมาส์ ชี้อยู่ และเมื่อคุณคลิกอีกครั้ง Scratch ก็จะบันทึกสีนั้น ให้เป็นสีของปากกาทันที

5 ถ้าคุณคลิกธงสีเขียวเพื่อทดสอบโค้ด คุณก็จะเห็นว่าภาพที่วาดด้วยปากกายังไม่หายไปไหน ดังนั้น ให้เพิ่มบล็อก **clear** จากกลุ่ม **Pen** ที่ตอนต้นของโปรแกรม เพื่อจัดการเรื่องนี้ให้!



ทีนี้ ให้คลิกธงสีเขียว แล้วรอดูผลลัพธ์



This work is licensed under the Creative Commons Attribution 4.0 International License.

แปลและเรียบเรียงจากเอกสารของ CoderDojo โดย จิตรภณ และณรงค์พร เหล่าศรีสิน (fb.me/narongpom.lao)

# Scratch ระดับปานกลาง



วาดรูป วนๆ ไป

การ์ดแผ่นที่ 2 จาก 5  
ฉันท้าลองเรียน: Scratch

1 แม้ว่าตอนนี้ คุณจะได้โปรแกรมที่สามารถวาดเส้นได้แล้ว แต่มันก็วาดให้เพียงเส้นเดียว ซึ่งนั่นดูธรรมดาเกินไป! คุณอาจใช้บล็อก loop อย่างเช่น **forever** เพื่อให้โปรแกรมวาดเส้นไปเรื่อยๆ แต่การใช้บล็อก **forever** นั้น จะทำให้เส้นที่วาดหลุดออกนอกเวทีได้!

ดังนั้นคุณจะต้องใช้รูปแบบอื่น ซึ่งคุณจะพบได้ในกลุ่ม **Control** เรียกว่าบล็อก **repeat until** ซึ่งจะทำงานวนไปเรื่อยๆ ซ้ำแล้วซ้ำเล่า จนกว่าเงื่อนไขจะเป็นจริง โดยให้ลากบล็อกนี้มาคลุมบล็อก **move** และบล็อก **turn** ดังภาพ



2 ตอนนี้ให้ลองคลิกธงสีเขียวเพื่อให้โปรแกรมทำงาน และดูว่าเกิดอะไรขึ้น! คุณจะสังเกตเห็นสองอย่าง คือ ตอนที่เริ่มวาดเส้นจะลากไปหาจุดกึ่งกลางของเวทีเสมอ และไม่ยอมหยุดวาดแม้จะไปถึงขอบเวทีก็ตาม

ที่เป็นอย่างนี้ ก็เพราะ บล็อกที่ทำงานต่อบล็อก **pen down** คือบล็อก **go to x: 0 y: 0** ดังนั้นให้แก้ไขโดยย้ายบล็อก **go to x: 0 y: 0** ไปอยู่ก่อนบล็อก **pen down** และเพิ่มบล็อก **pen up** ที่ตอนต้นของโค้ดด้วย

ส่วนเรื่องที่สอง เป็นเพราะคุณยังไม่ได้กำหนดเงื่อนไขใดๆ ลงในบล็อก **repeat until** เลย ดังนั้นเงื่อนไขจึงไม่มีทางเป็น “จริง” ได้ การทำงานจึงวนซ้ำไม่รู้จบ เหมือนการใช้บล็อก **forever**

3 เพื่อแก้ไขปัญหา **repeat until** คุณจะต้องหาวิธีตรวจสอบว่าสไปรท์ที่คุณซ่อนไว้นั้น ได้เคลื่อนที่ไปถึงขอบเวทีหรือยัง โดยคุณจะต้องใช้บล็อก **เซ็นซิ่ง** ซึ่งในกรณีนี้ คือ บล็อก **touching ?** ให้นำมาต่อเข้ากับบล็อก **repeat until** และกำหนดให้ในช่องเป็น **edge**



# Scratch ระดับปานกลาง



วาดรูป วนๆ ไป

การ์ดแผ่นที่ 2 จาก 5  
ฉันกำลังเรียน: Scratch

4

ลองเปลี่ยนจำนวน steps เป็น 5 และตรวจดูว่าโค้ดของคุณเป็นไปตามนี้

```
when clicked
pen up
hide
clear
go to x: 0 y: 0
set pen color to
pen down
repeat until touching edge ?
  move 5 steps
  turn 15 degrees
```

5

ที่นี่ เมื่อคุณรันโปรแกรม คุณก็จะเห็นว่า มันวาดเป็นวงกลม แต่อย่างน้อย ตอนนี้ก็ไม่ต้องตกขอบเวทีแล้ว! ปัญหาก็คือ ค่ามุมที่หมุน 15 องศา เมื่อหมุนไปเรื่อยๆ ก็จะกลายเป็น 360 องศา หรือหมุนครบรอบเป็นวงกลมนั่นเอง สิ่งที่คุณจะต้องทำก็คือ เพิ่มจำนวน steps ให้มากขึ้นในแต่ละรอบ เพื่อให้วงรอบขยายออกไป ซึ่งในกรณีแบบนี้ คุณจะใช้ ตัวแปร มาช่วย

คุณเคยเรียนเรื่องตัวแปรมาแล้วในชุด “Scratch เบื้องต้น” โดยพื้นฐานแล้ว ตัวแปรก็คือ ที่เก็บข้อมูล ซึ่งคุณสามารถสร้างตัวแปรใหม่ โดยไปที่กลุ่ม Data และเลือกใช้บล็อกอื่นๆ ที่เกี่ยวข้องได้จากที่นั่นด้วย

ให้คุณสร้างตัวแปรชื่อ steps และนำค่าตัวแปรไปแทนที่เลข 5 ในบล็อก move 5 steps จากนั้น ให้เพิ่มบล็อก set steps to 0 ที่ตอนต้นของโปรแกรม และใส่บล็อก change steps by 1 ไว้ในลูป repeat until (โดยไม่จำกัดว่าจะใส่ไว้ตรงไหนก็ได้ ขอเพียงให้อยู่ในลูป)

```
when clicked
set steps to 0
pen up
```

```
repeat until touching edge ?
  move steps steps
  turn 75 degrees
  change steps by 1
```

เสร็จแล้ว ให้รันโปรแกรม และลองเปลี่ยนค่ามุมดู เช่น 76 และ 120!



This work is licensed under the Creative Commons Attribution 4.0 International License.

แปลและเรียบเรียงจากเอกสารของ CoderDojo โดย จิตรภณ และณรงค์พร เหล่าศรีสิน (fb.me/narongporm.lao)

# Scratch ระดับปานกลาง

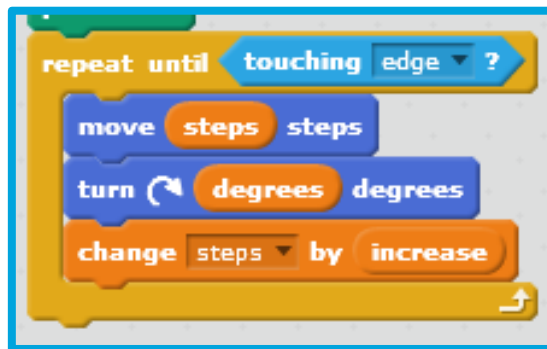


การขออนุญาต

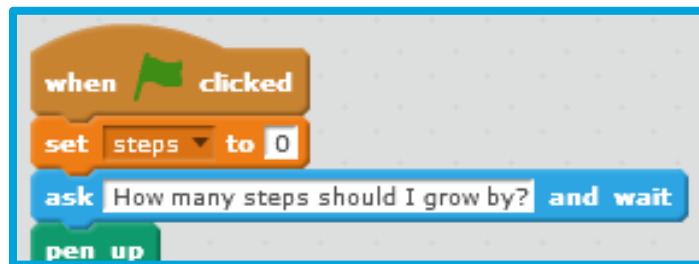
การ์ดแผ่นที่ 3 จาก 5  
ฉันกำลังเรียน: Scratch

1 ทุกอย่างเริ่มดูดีขึ้นมาแล้ว แต่มันคงเป็นเรื่องที่ยุ่งยากนิดๆ ที่ต้องมาคอยแก้โค้ด เพื่อวาดลวดลายที่แปลกออกไป คงจะดี หากโปรแกรมสามารถถามคุณได้ว่าต้องการแบบไหน? ซึ่งเราทำแบบนั้นได้!

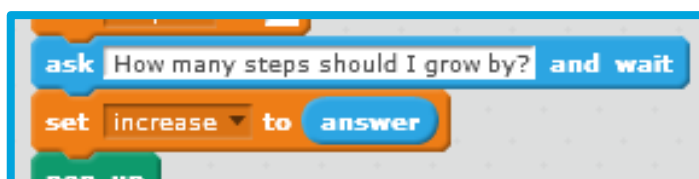
ขั้นแรก ให้ไปที่ Data สร้างตัวแปรชื่อ degrees และ increase แล้วนำไปใส่ในโค้ดของคุณ ตามนี้



2 เราจะต้องขอให้ผู้ใช้กรอกข้อมูลเพื่อกำหนดค่าให้กับตัวแปรทั้งสองตัวนี้ ซึ่งสามารถทำได้ โดยใช้บล็อก เช่นสิ่งที่ชื่อว่า Ask and wait ให้คุณไปที่กลุ่ม Sensing นำบล็อกนี้มาวางแทรกในโค้ดเดิมของคุณ ต่อจากบล็อก set steps to 0 ดังตัวอย่าง แล้วกำหนดคำถามเป็น “How many steps should I grow by?” (ความหมาย คือ จะให้เพิ่มจำนวน steps ที่ละเท่าไร?)



3 เมื่อคุณทำให้โปรแกรมถามคำถามได้แล้ว ก็ต้องจำคำตอบไว้ด้วย! โชคดีที่ใน Scratch มีตัวแปรพิเศษที่ชื่อว่า answer ซึ่งจะเก็บคำตอบที่ฟังได้รับมา โดยจะอยู่ในกลุ่ม Sensing จากนั้น ให้ใช้บล็อก set to จากกลุ่ม Data เพื่อนำค่าจาก answer ไปใส่ไว้ในตัวแปร increase ดังภาพ



# Scratch ระดับปานกลาง



## การขออนุญาต

การ์ดแผ่นที่ 3 จาก 5  
ฉันกำลังเรียน: Scratch

- 4 ให้ทำแบบเดียวกันกับตัวแปร **degrees** โดยให้ถามว่า “How many degrees should I turn?” ซึ่งหมายความว่า จะให้หมุนกี่องศา? แล้วเก็บคำตอบไว้ในตัวแปร **degrees**

```
set increase to answer
ask 'How many degrees should I turn?' and wait
set degrees to answer
```

- 5 ตรวจสอบว่าโปรแกรมของคุณเป็นไปตามรูปข้างล่างนี้หรือไม่ และลองรันโปรแกรมดู ลองเปลี่ยนตัวเลขแล้วบันทึกค่าที่ทำให้ได้ภาพสวยๆ เพราะคุณจะต้องใช้ค่าเหล่านั้นในการ์ดต่อไป!

```
when clicked
set steps to 0
ask 'How many steps should I grow by?' and wait
set increase to answer
ask 'How many degrees should I turn?' and wait
set degrees to answer
pen up
hide
clear
go to x: 0 y: 0
set pen color to blue
pen down
repeat until touching edge:
  move steps steps
  turn degrees degrees
  change steps by increase
```



# Scratch ระดับปานกลาง



## วาดลายเส้นหลากสี

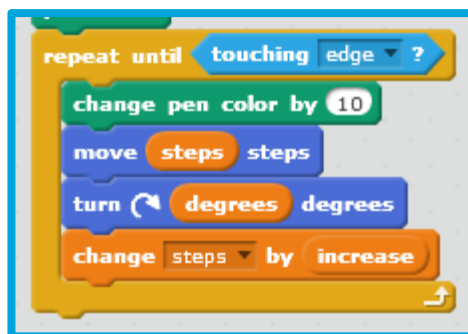
การ์ดแผ่นที่ 4 จาก 5  
ฉันกำลังเรียน: Scratch

1 ถึงเวลาเพิ่มสีสັນกันแล้ว! เพราะตอนนี้ลายเส้นยังมีแค่สีเดียว แต่ในกลุ่ม Pen มีบล็อกที่สามารถเปลี่ยนสีปากกาได้ และเมื่อใช้ร่วมกับบล็อกในกลุ่ม Operators คุณจะสามารเปลี่ยนสีแบบสุ่มได้เลย

บล็อกสำหรับเปลี่ยนสีปากกานั้น มีชื่อว่า change pen color by



เลือกบล็อกนี้จากกลุ่ม Pen และนำมาใส่ในลูป repeat until ดังนี้



2 ถึงแม้จะดูมีสีสັນมากขึ้น แต่ก็ยังเป็นรูปแบบตายตัวไปหน่อย คุณสามารถทำให้ดูแปลกตามากขึ้นได้ ด้วยการสุ่มตัวเลขเพื่อใช้ในการเปลี่ยนสี โดยใส่บล็อก pick random ลงในบล็อก change pen color by ซึ่งคุณอาจลองใช้ค่า 1 และ 100 ไปก่อนก็ได้



ลองรันโปรแกรมอีกครั้ง คราวนี้จะได้เห็นรู้งสีสັນแปลกตากัน!



# Scratch ระดับปานกลาง



## วาดลายเส้นหลากสี

การ์ดแผ่นที่ 4 จาก 5  
ฉันทำล้งเรียน: Scratch

3 คุณสามารถใช้ตัวเลขสุ่ม เพื่อให้โปรแกรมทั้งหมดทำงานด้วยตัวมันเองไปเรื่อยๆ และเปลี่ยนลวดลายไปทุกครั้ง เหมือนกับสกรีนเซฟเวอร์ที่ใช้กันในยุค 90 ถ้าใครที่ไม่รู้จัก ก็ลองถาม Dojo Mentors ดู!

คุณจะต้องปรับเปลี่ยนบางอย่าง เพื่อให้เป็นไปตามที่คิดไว้ อันดับแรก คุณต้องกำหนดค่าตัวแปร **increase** และ **degrees** แบบสุ่ม แทนที่จะถามจากผู้ใช้ ดังนั้น ให้เปลี่ยนโค้ดบล็อกบางส่วน ดังนี้

```
when clicked
  set steps to 0
  ask How many steps should I grow by? and wait
  set increase to answer
  ask How many degrees should I turn? and wait
  set degrees to answer
  pen up
```

เปลี่ยนเป็น

```
when clicked
  set steps to 0
  set increase to pick random 1 to 10
  set degrees to pick random 1 to 180
  pen up
```

4 เมื่อลองรันโปรแกรม คุณจะพบว่าโปรแกรมวาดเส้นแบบสุ่ม แต่วาดเพียงครั้งเดียว ทำไมถึงเป็นแบบนี้? ทั้งนี้ก็เพราะลูปจะทำงานไปเรื่อยๆ แต่หยุดเมื่อชนขอบเวที คุณจึงต้องใช้อีกลูปหนึ่ง เพื่อให้โปรแกรมเริ่มวาดใหม่หลังจากที่ชนขอบ โดยนำบล็อก **forever** มาครอบโค้ดทั้งหมด ดังภาพ

```
when clicked
  forever
    set steps to 0
    set increase to pick random 1 to 10
    set degrees to pick random 1 to 180
    pen up
    hide
    clear
    go to x: 0 y: 0
    set pen color to
    pen down
    repeat until touching edge?
      change pen color by pick random 1 to 100
      move steps steps
      turn degrees degrees
      change steps by increase
```



This work is licensed under the Creative Commons Attribution 4.0 International License.

แปลและเรียบเรียงจากเอกสารของ CoderDojo โดย จิตรภณ และณรงค์พร เหล่าศรีสิน (fb.me/narongporm.lao)



# Scratch ระดับปานกลาง



## วาดแต่ภาพสวยๆ

การ์ดแผ่นที่ 5 จาก 5  
ฉันทำลังเรียน: Scratch

1

ถึงตรงนี้ คุณก็ได้ภาพสวยๆ ไว้ดูแล้ว! อย่างไรก็ตาม คุณอาจสังเกตเห็นว่า ไม่ใช่ทุกครั้งที่ภาพออกมาดี บางรูปก็ดูค่อนข้างแย่นั่นก็เพราะ ค่าตัวเลขของตัวแปรบางตัว อาจเป็นค่าที่ไม่ค่อยเหมาะสมสักเท่าใด หรือบางค่าเมื่อนำมาใช้ร่วมกัน ก็อาจเป็นตัวเลือกที่ไม่ค่อยดีเช่นกัน

คุณยังจำได้ไหมว่า ในการดก่อนหน้า เราได้บอกให้คุณบันทึกค่าสำหรับตัวแปร **increase** และ **degrees** ซึ่งทำให้ได้รูปวาดที่ดูดีที่สุด ถ้าจำไม่ได้ ก็ไม่ต้องกังวล ให้คุณนั่งดูโปรแกรมทำงานแบบสุมไปสักพักก่อน แล้วค่อยบันทึกค่าตัวแปรทั้งสองที่ทำให้ได้ภาพที่สวยงามออกมา

สิ่งที่เรากำลังจะทำต่อไป ก็คือ เราจะต้องสอน Scratch ให้ใช้ค่าตัวแปรที่เหมาะสมเท่านั้น เพื่อที่จะได้วาดแต่รูปสวยๆ ออกมาให้คุณ!

การจะทำแบบนั้นได้ คุณต้องใช้ **ลิสต์ (list)** ซึ่งมันจะอยู่ที่เดียวกับ **ตัวแปร** ต่างๆ คือ อยู่ในกลุ่ม **Data** แต่ก่อนที่คุณจะใช้งานลิสต์ คุณจะต้องสร้างลิสต์ขึ้นมาก่อน

คลิก Make a List

ตั้งชื่อว่า Degrees List

คุณก็จะได้ List ไว้ใช้ในโปรแกรม

2

สร้างลิสต์อีกชุด ตั้งชื่อว่า **Increase List** จากนั้นก็คลิกเครื่องหมาย **+** เล็กๆ ตรงด้านล่างของแต่ละลิสต์ แล้วใส่ตัวเลขลงไป ในทั้งสองลิสต์เป็นคู่ๆ ไป

ตรวจสอบอีกครั้งให้แน่ใจว่า ค่ามุมมองใน **Degrees List** กับค่าตัวเลขใน **Increase List** ที่เป็นคู่กันนั้น อยู่ในลำดับเดียวกัน



This work is licensed under the Creative Commons Attribution 4.0 International License.

แปลและเรียบเรียงจากเอกสารของ CoderDojo โดย จิตรภณ และณรงค์พร เหล่าศรีสิน (fb.me/narongpom.lao)

# Scratch ระดับปานกลาง



## วาดแต่ภาพสวยๆ

การ์ดแผ่นที่ 5 จาก 5  
ฉันทำล้งเรียน: Scratch

3 ตอนนี้ ลิสต์ของคุณก็พร้อมแล้ว สิ่งที่คุณต้องทำต่อ ก็คือ ให้โค้ดของคุณอ่านข้อมูลจากลิสต์ แล้วนำไปวาดที่ละภาพ โดยคุณจะต้องสร้างตัวแปรใหม่ ชื่อ **counter** ขึ้นมา จากนั้นก็เพิ่มค่าตัวแปรนี้ และใช้บล็อก **if then** ในการอ่านข้อมูลจากลิสต์ทีละคู่จนครบ ซึ่งโค้ดที่แก้ไขแล้ว จะเป็นดังภาพข้างล่างนี้

กำหนด counter เป็น 0 อยู่ข้างนอกของทุกลูป

เพิ่มค่า counter ทีละ 1

```
when clicked
  set counter to 0
  forever
    if counter = length of Increase List then
      set counter to 0
    change counter by 1
    set steps to 0
    set increase to item counter of Increase List
    set degrees to item counter of Degrees List
    pen up
    hide
    clear
    go to x: 0 y: 0
    set pen color to
```

หากค่า counter เท่ากับจำนวนข้อมูลในลิสต์ ก็จะทำให้ counter เป็น 0 ซึ่งจะทำให้ค่า counter ไม่เกินกว่าจำนวนข้อมูลในลิสต์

ใช้ค่า counter เพื่อกำหนดลำดับของข้อมูลในลิสต์ที่จะหยิบขึ้นมาใส่ในตัวแปร increase และ degrees

### ถ้าคุณอยากรู้ว่าเกิดอะไรขึ้น

ให้ลองจินตนาการว่าลิสต์ของคุณมีข้อมูลเพียงสองคู่ และนี่คือสิ่งที่เกิดขึ้น:

1. กำหนดค่า counter เป็น 0
2. เริ่มเข้าสู่ลูป forever
3. บล็อก if จะตรวจสอบว่า counter (0) เท่ากับจำนวนข้อมูลในลิสต์ Increase List (2) หรือไม่
4. ถ้าไม่เท่า ให้เพิ่มค่า counter อีก 1 ตอนนี้ counter ก็จะมีค่าเป็น 1
5. กำหนดค่า steps เป็น 0
6. นำค่าแรกจากลิสต์ Increase List มาใส่ในตัวแปร increase
7. นำค่าแรกจากลิสต์ Degrees List มาใส่ในตัวแปร degrees
8. จากนั้น ก็เริ่มวาดภาพตามที่เขียนโปรแกรมไว้
9. กลับไปเริ่มลูป forever อีกครั้ง
10. บล็อก if จะตรวจสอบว่า counter (1) เท่ากับจำนวนข้อมูลในลิสต์ Increase List (2) หรือไม่
11. เมื่อไม่เท่า จึงเพิ่มค่า counter อีก 1 เป็น 2
12. กำหนดค่า steps เป็น 0
13. นำค่าที่ 2 จากลิสต์ Increase List มาใส่ในตัวแปร increase
14. นำค่าที่ 2 จากลิสต์ Degrees List มาใส่ในตัวแปร degrees
15. วาดภาพเสร็จ ก็กลับไปเริ่มลูป forever อีกครั้ง
16. คราวนี้ บล็อก if พบว่า counter (2) เท่ากับจำนวนข้อมูลในลิสต์ Increase List (2) แล้ว!
17. รีเซตค่าในตัวแปร counter เป็น 0
18. ทำต่อจากขั้นที่ 4 วนอยู่ในลูปที่ไม่รู้จบ!