

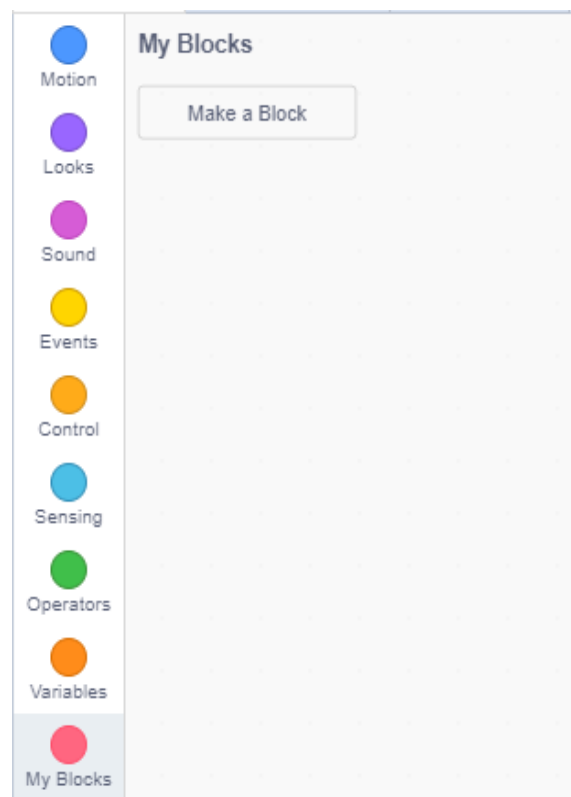
ใบความรู้ที่ 1 เรื่อง ฟังก์ชัน (function)
หน่วยการเรียนรู้ที่ 2 เรื่อง การโปรแกรม
แผนการจัดการเรียนรู้ที่ 7 เรื่อง การออกแบบและเขียนโปรแกรมที่มีฟังก์ชัน (2)
รายวิชา วิทยาการคำนวณ รหัสวิชา ว22104 ภาคเรียนที่ 1 ชั้นมัธยมศึกษาปีที่ 2

ฟังก์ชัน (function) หรือโปรแกรมน้อย (subroutine) เป็นกลุ่มของคำสั่งที่ทำงานอย่างใดอย่างหนึ่ง โดยเฉพาะผู้ใช้จะนิยามฟังก์ชัน โดยกำหนดชื่อฟังก์ชันและคำสั่งที่ต้องการภายในฟังก์ชันนั้น

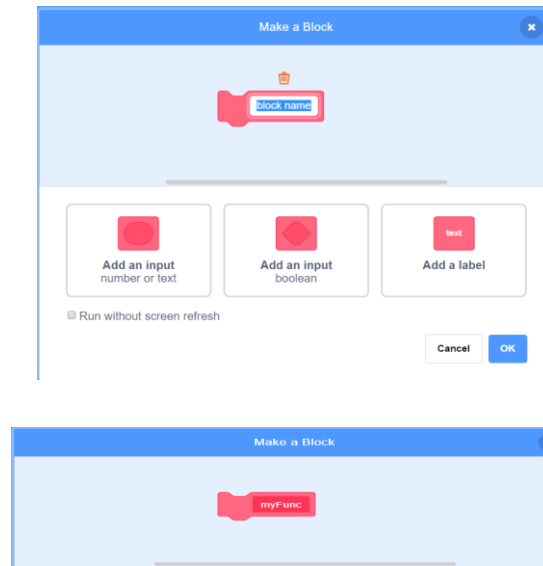
การตั้งชื่อฟังก์ชันควรตั้งให้เหมาะสมกับการทำงานเพื่อให้ง่ายต่อการทำความเข้าใจ ผู้เขียนโปรแกรมสามารถเรียกใช้ฟังก์ชันได้โดยไม่ต้องเขียนชุดคำสั่งซ้ำ ๆ กันอีก และสามารถนำไปใช้กับโปรแกรมอื่นที่มีลักษณะคล้ายกันได้

การสร้างฟังก์ชัน

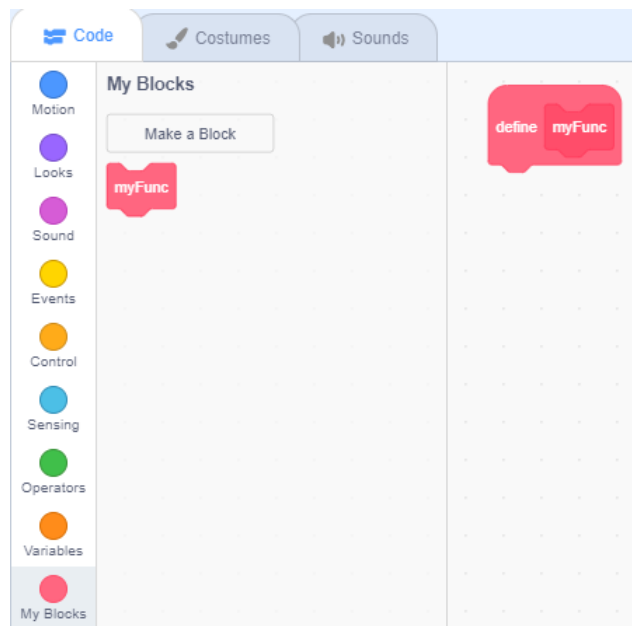
1. เปิดโปรแกรม scratch
2. คลิกกลุ่มบล็อก My Blocks
3. คลิก Make a Block เพื่อสร้างฟังก์ชัน จะปรากฏหน้าต่าง Make a Block



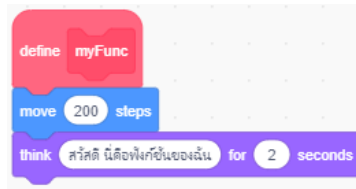
4. พิมพ์ชื่อฟังก์ชันที่ต้องการ โดยตั้งชื่อให้สอดคล้องกับงานที่กำหนด ในที่นี้ทดลองพิมพ์ชื่อ myFunc แล้วคลิกปุ่ม OK



5. จะปรากฏบล็อกชื่อฟังก์ชันที่สร้างขึ้น และบล็อกคำสั่งที่นิยามส่วนหัวของฟังก์ชันในพื้นที่เขียนโปรแกรม

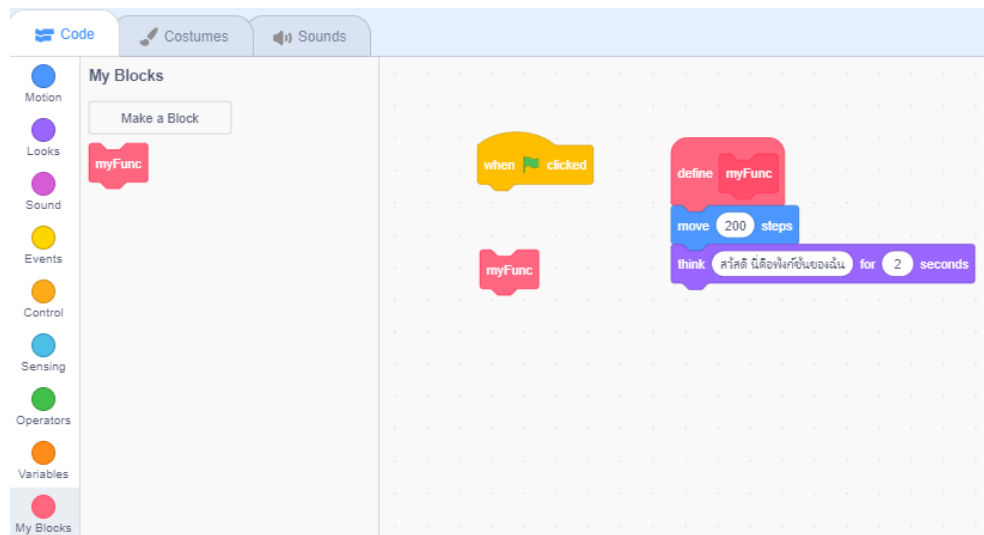


6. สามารถลากบล็อกคำสั่งที่ต้องการมาต่อส่วนหัวของฟังก์ชันได้ดังรูป



การเรียกใช้งานฟังก์ชัน

1. คลิกบล็อกฟังก์ชันที่ต้องการ
2. ลากไปวางในพื้นที่เขียนโปรแกรม

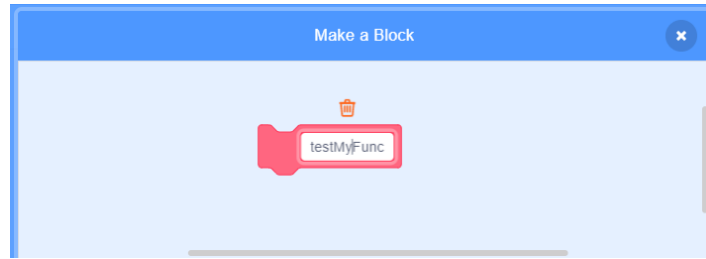


การแก้ไขชื่อฟังก์ชัน

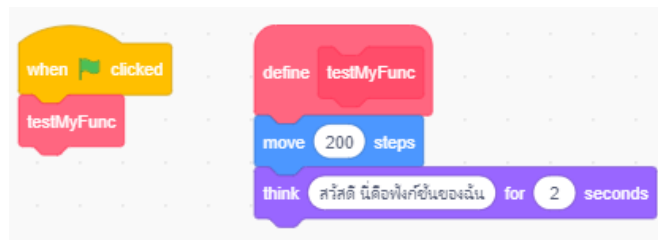
1. คลิกขวาที่บล็อกคำสั่งฟังก์ชันที่สร้างขึ้น แล้วเลือก edit



2. เมื่อปรากฏหน้าต่าง Make a Block แก้ไขเป็นชื่อฟังก์ชันที่ต้องการ แล้วคลิกปุ่ม OK

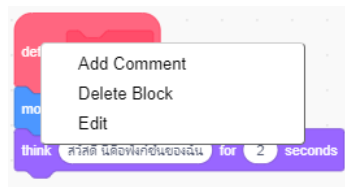


3. เมื่อเปลี่ยนชื่อฟังก์ชัน โปรแกรม Scratch จะเปลี่ยนชื่อฟังก์ชัน ในส่วนที่มีการอ้างอิงให้โดยอัตโนมัติ

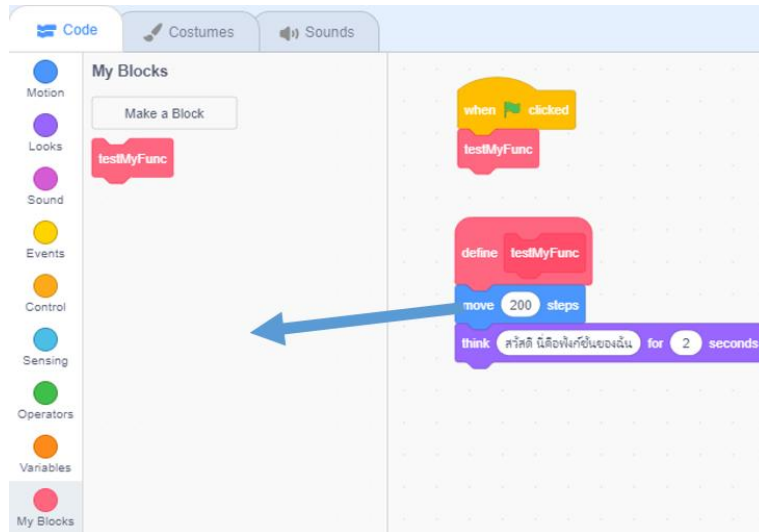


การลบฟังก์ชัน

วิธีที่ 1 คลิกขวาที่ฟังก์ชัน แล้วเลือก Delete Block



วิธีที่ 2 ลากฟังก์ชันออกจากพื้นที่เขียนโปรแกรมมาวางในพื้นที่แสดงบล็อกคำสั่ง



ตัวอย่าง โปรแกรม ATM

แนวคิด

โปรแกรมการจ่ายเงินของตู้ ATM โดยให้ผู้ไชรอกจำนวนเงิน แล้วโปรแกรมจะตรวจสอบว่า ถูกต้องหรือไม่ โดยมีเงื่อนไขว่า จำนวนเงินต้องเป็นจำนวนที่สามารถจ่ายเป็นธนบัตรที่มีมูลค่าต่างๆ แล้วมีผลรวมเท่ากับจำนวนเงินที่ระบุ ในที่นี้กำหนดให้เครื่อง ATM มีธนบัตรมูลค่าต่าง ๆ คือ ธนบัตรใบละ 1000 ธนบัตรใบละ 500 และธนบัตรใบละ 100 โดยฟังก์ชันจะมีความทำงานดังนี้

ฟังก์ชัน input

1. รับจำนวนเงินจากผู้ใส่เก็บในตัวแปร money
2. ถ้า (money หาร 100) มีเศษเท่ากับ 0 (สามารถจ่ายธนบัตรได้) แล้ว
 - 2.1 เรียกใช้ฟังก์ชัน ATM

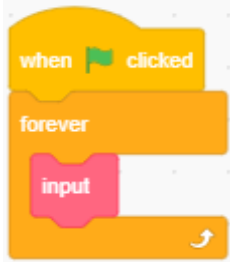
ไม่เช่นนั้น

- 3.1 แสดงข้อความแจ้งผู้ใส่ว่าใส่จำนวนเงินไม่ถูกต้อง

ฟังก์ชัน ATM

1. นับธนบัตรใบละ 1000 ที่ตรงกับจำนวนเงิน ($\text{money} / 1000$)
2. หักมูลค่าธนบัตรหลัก 1000 ออก (เศษจาก $\text{money} / 1000$)
3. นับธนบัตรใบละ 500 ที่ตรงกับจำนวนเงิน ($\text{money} / 500$)
4. หักมูลค่าธนบัตรหลัก 500 ออก จะได้ธนบัตร 100 (เศษจาก $\text{money} / 500$)
5. แสดงจำนวนธนบัตรแต่ละชนิด

โปรแกรมมีดังนี้

เริ่มต้น 	เรียกใช้ฟังก์ชัน input
input 	1. ให้ผู้ใช้กรอกจำนวนเงิน เก็บในตัวแปร money 2. ถ้า $(\text{money} \bmod 100) = 0$ จริง 2.1 เรียกใช้ฟังก์ชัน ATM 3. ไม่จริง 3.1 แสดงข้อความ “คุณใส่จำนวนเงินไม่ถูกต้อง” เป็นเวลา 3 วินาที
ฟังก์ชัน ATM  1. นับธนบัตรใบละ 1,000 บาท ใช้สูตร $b1000 = \text{money} / 1000$ (ฟังก์ชัน floor คือฟังก์ชันปัดเศษ ทำให้ผลลัพธ์เป็นจำนวนเต็ม) 2. แยกจำนวนเงินที่มีมูลค่าหลัก 1,000 ออกจากจำนวนเงิน โดยใช้สูตร $\text{money} = \text{money} \bmod 1000$ 3. นับธนบัตรใบละ 500 บาท ใช้สูตร $b500 = \text{money} / 500$ 4. นับธนบัตรใบละ 100 บาท ใช้สูตร $b100 = \text{money} \bmod 500$ 5. แสดงจำนวนธนบัตรแต่ละชนิด เป็นเวลา 3 วินาที	

ตัวอย่างการทำงานของโปรแกรม ATM

กรณีที่ใส่จำนวนเงินถูกต้อง



กรณีที่ใส่จำนวนเงินไม่ถูกต้อง

