technoteach

technocamps

UNDEB EWROPEAIDD
EUROPEAN UNION

Llywodraeth Cymru
Welsh Government

Cronfa Gymdeithasol Ewrop
European Social Fund

Prifysgol Abertawe
Swansea University

CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD

PRIFYSGOL BANGOR UNIVERSITY

Cardiff Metropolitan University
Prifysgol Metropolitan Caerdydd

i.t.wales

PRIFYSGOL ABERYSTWYTH UNIVERSITY

PRIFYSGOL glyndŵr Wrecsam
Wrexham glyndŵr UNIVERSITY

University of South Wales
Prifysgol De Cymru

Micro:Bits across the Curriculum For Wales

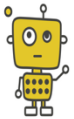
Coding Across the CFW

Coding can be implemented across all the Areas of Learning and Experience, reinforcing learning in the classroom and improving digital literacy in the process.

In today's world digital literacy is an essential skill for learners to develop. The technological requirements for jobs are ever increasing, and a strong start in digital skills will prepare learners and give them an advantage.

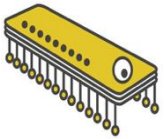
Expressive Arts

Health and Wellbeing



Humanities

Languages, Literacy and Communication



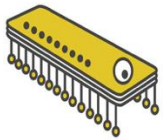
Mathematics and Numeracy

Science and Technology

Ideas for Micro:Bits Across the Curriculum

Health and Wellbeing

- Emotion Tracker
- Pedometer



Mathematics and Numeracy

- Maths Game - Salute
- Clicker Counter

Science and Technology

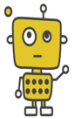
- Conductivity Tester
- Sea Turtle Lights

Languages, Literacy and Communication

- Morse Code
- 5G & Wireless Comms

Expressive Arts

- Animation
- Musical Micro:Bits

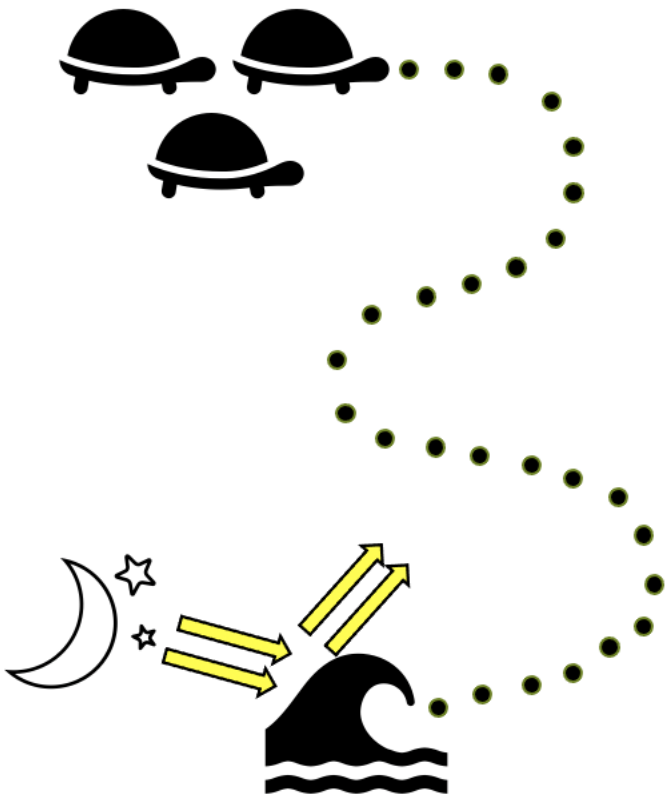


Humanities

- Climate Controller
- Compass

Science & Technology: Sea Turtle Lights

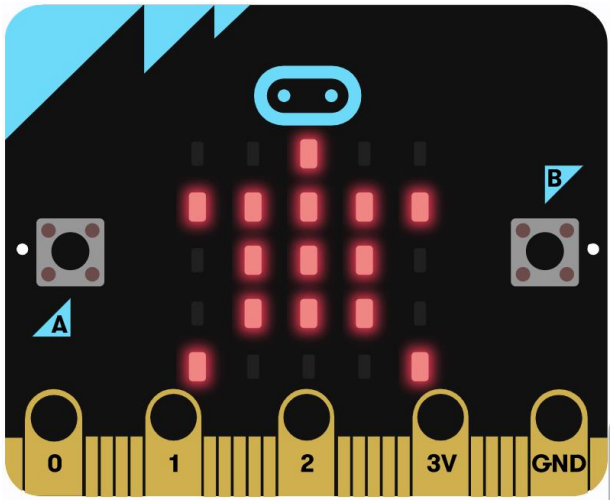
Sea Turtle Hatchlings



- When the eggs on the beach have hatched, the young turtles make their way to the sea in search of food and shelter.
- They use the moonlight and stars reflecting off the water to guide them to the ocean.
- Bright lights used by humans can confuse the turtles causing them to get lost.
- Can you think of a way to solve this problem?

Sea Turtle Lights

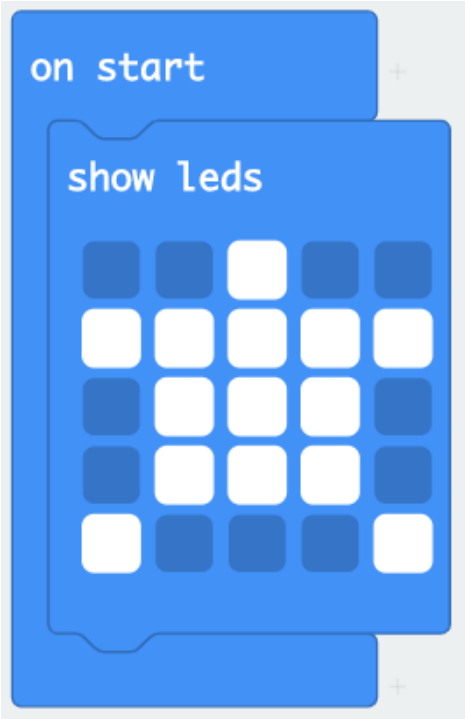
- Instead of strong lights like lamp posts and torches, we can use turtle-safe lights to guide humans around turtle nesting beaches.
- We can program the micro:bits to display red low wattage lights that can be placed along the ground. They are bright enough to guide humans without confusing turtles.



Activity: Turtle Light Sensors

Turtle Lights

- Start by creating a turtle icon that will light up on the micro:bit
- How can we make the micro:bit only light up during nighttime?
- Hint: Think about the sensors you learnt about earlier.



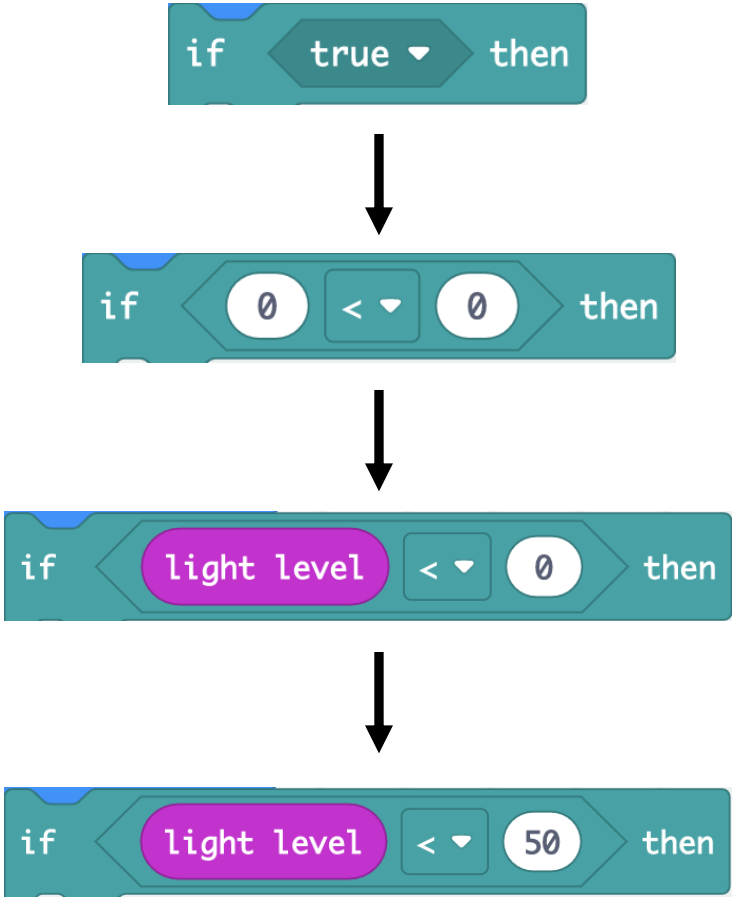
Turtle Lights

1. Start by dragging and dropping the if true then block into the forever block from the logic section.

Turtle Lights

2. We need to program the micro:bit to react to the light level. Add the 0 < 0 command.

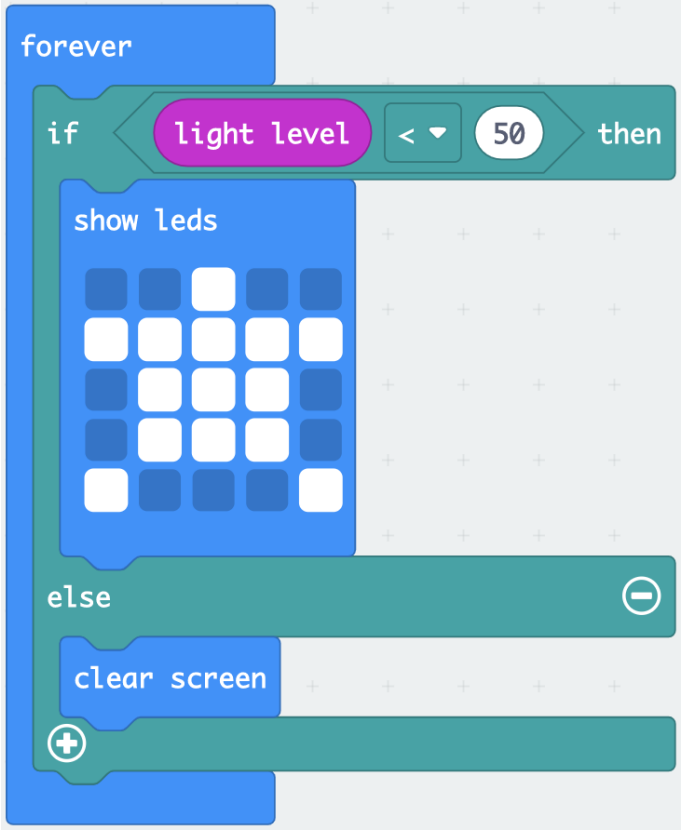
3. Add the light level command and change the number to 50.



Turtle Lights

4. Add the show leds and clear screen commands.

Try running the code. Can you get the micro:bit to light up?



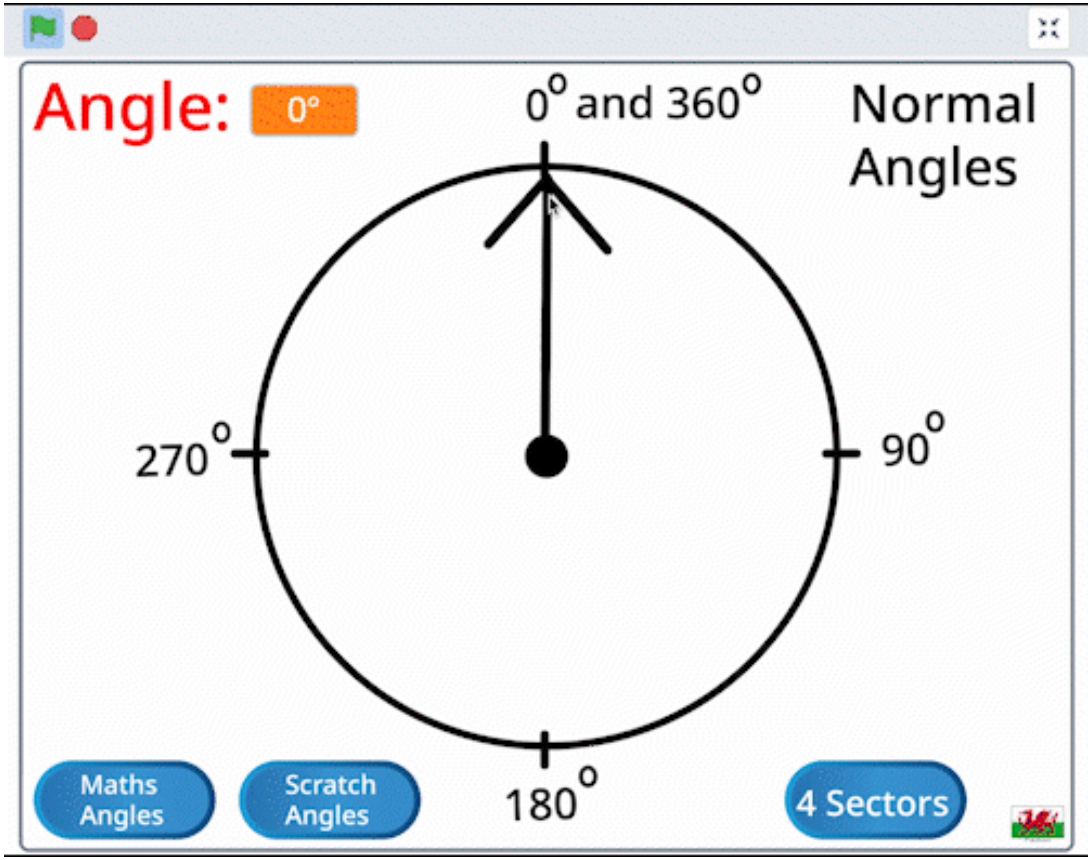
Humanities: Compass

Compass

A compass is like a full circle and Makecode (and your Micro:Bit) store the compass heading in degrees.

It is much like like the Scratch programme here.

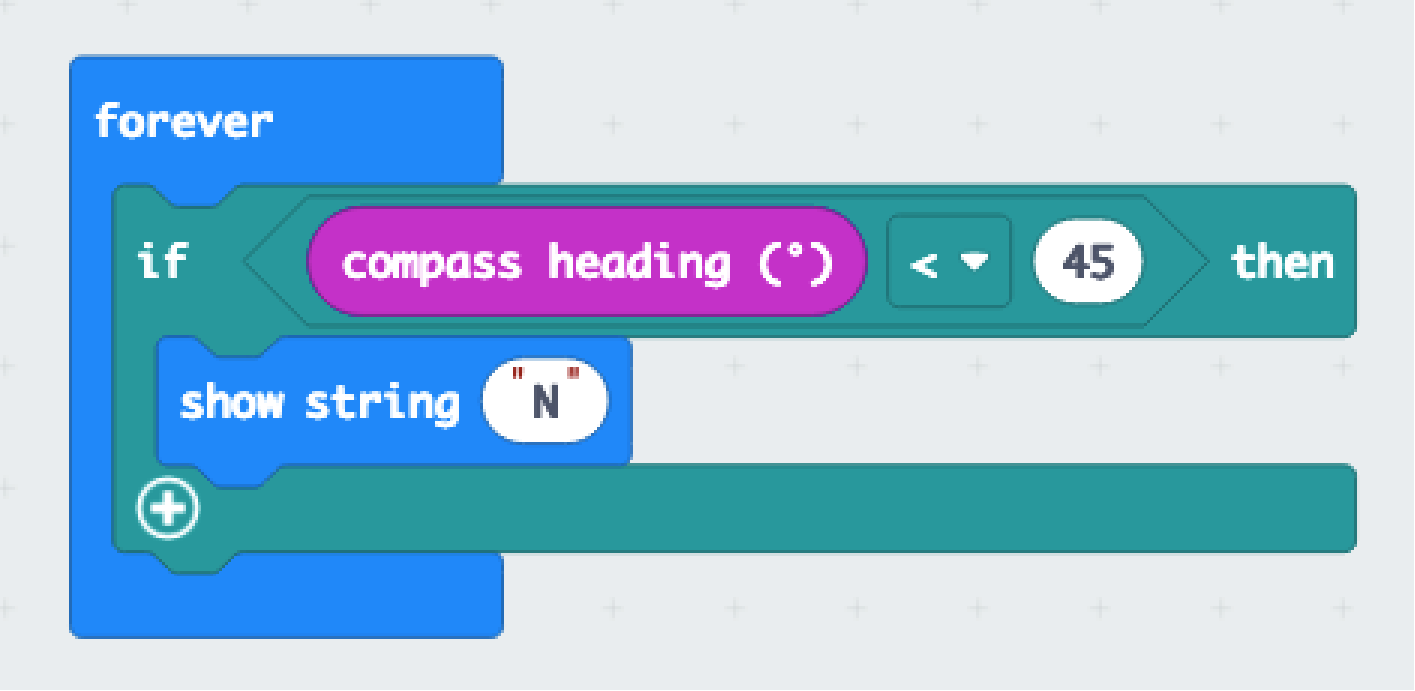
Using this we can turn our Micro:Bit into a compass



Compass

We can set our Micro:Bit to display "N" when the compass heading is less than a certain value

0 is North, so we can work out that 45° is the limit

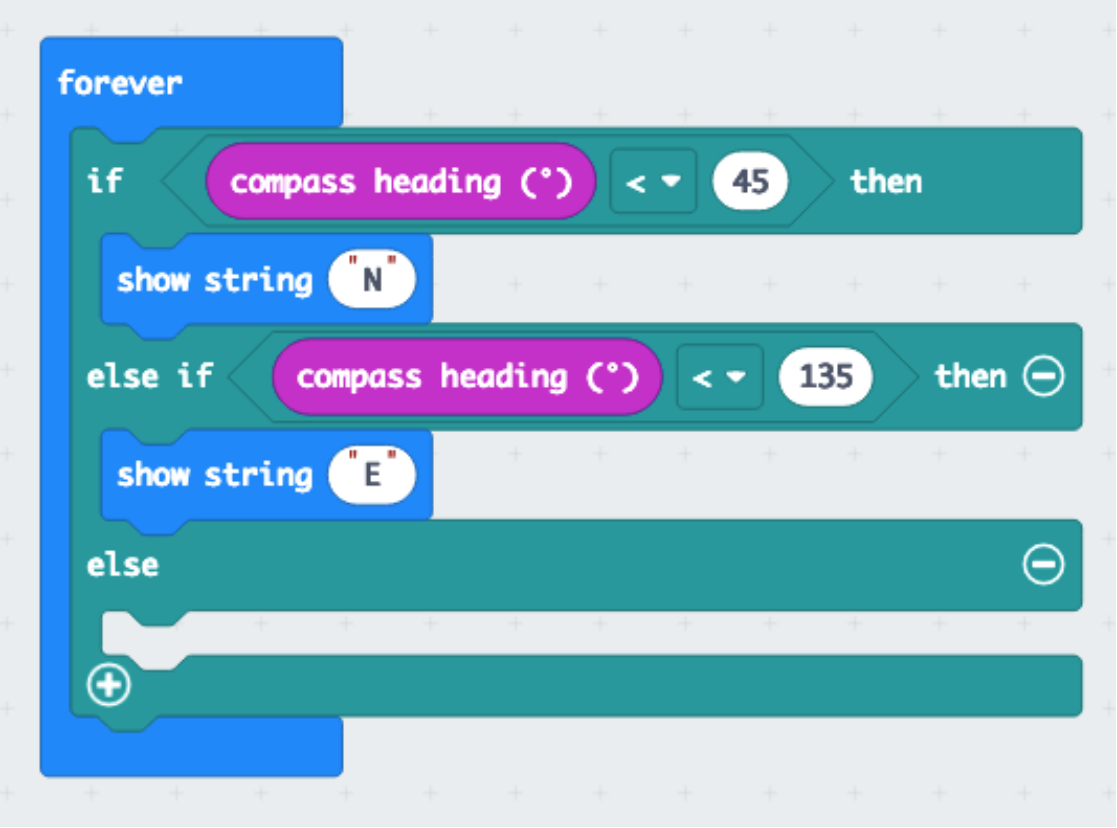


Compass

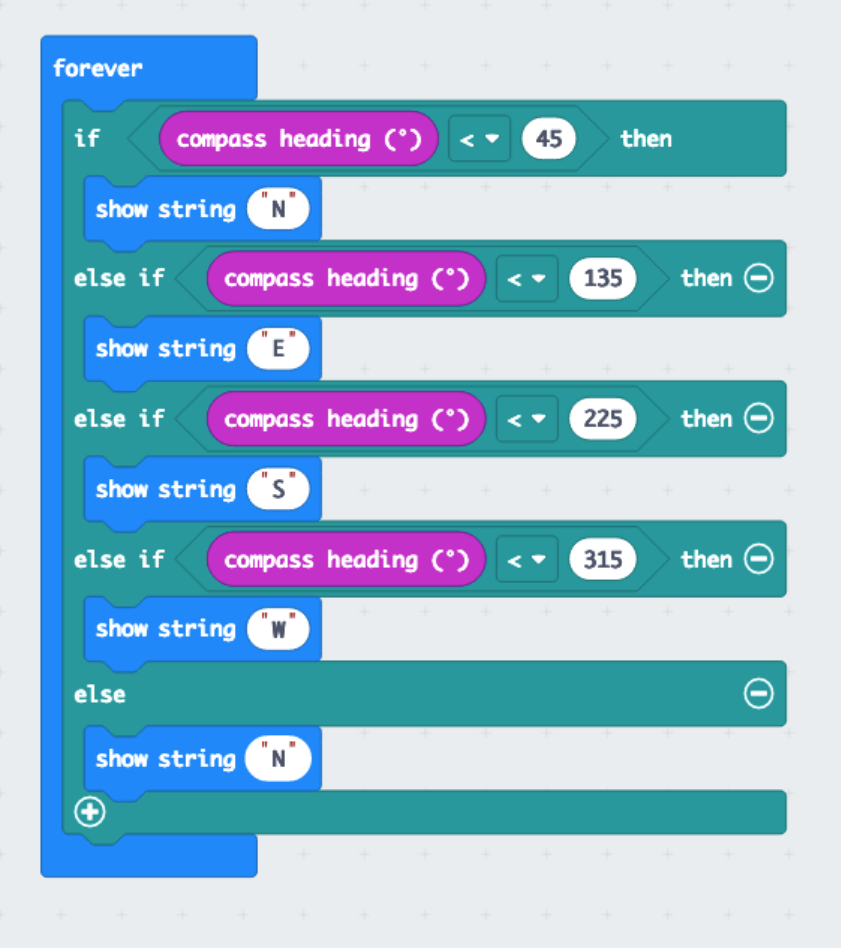
Now we have North, we can start adding the other cardinal directions.

Click the "+" at the bottom right of the block to add an 'else if'

This will allow us to add East



Compass



Now we can continue adding the rest of the directions

This is quite a long sequence of ifs but we should be able to follow it with our finger to understand what is going on

We can test our Micro:Bits to see if they work

Compass Extension

It should be possible to add in NE, NW, SE, SW

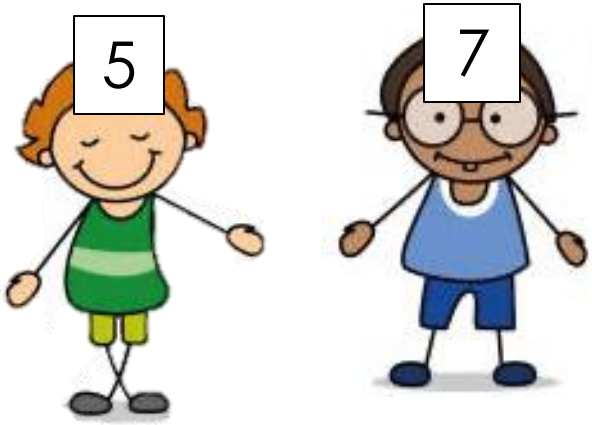
Can you work out how to do it?

technoteach
technocamps

Maths & Numeracy: Salute!

How to Play Salute

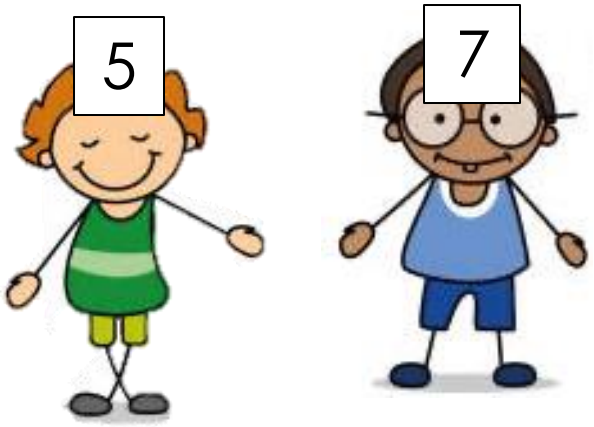
- Salute is a fun math game for 3 players.
- Two players hold up a number to their foreheads— they can see the other person's number, but not their own!
- A third player looks at the two numbers and decides either to multiply them or add them. They say the result out loud.
- Based on that number and the other person's number, the two other players try to guess their own number.
- The fastest player to guess their own number wins the round!



How to Play Salute

For example:

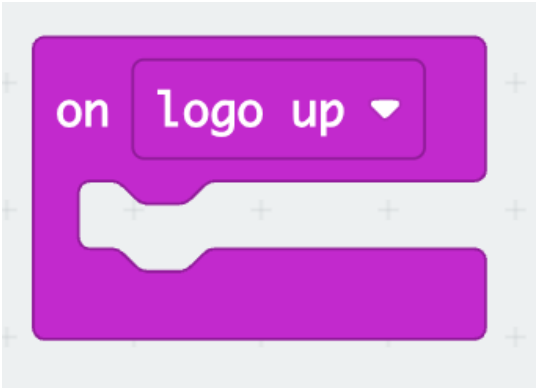
- The green shirt player sees the number 7 on the blue player
- After hearing the answer is 12, the player knows that no number can multiply with 7 to equal 12.
- Therefor he knows the numbers must have been added.
- What adds with 7 to equal 12?
- 5!



Playing Salute with micro:bits

We can play Salute with micro:bits. micro:bits have a way to generate random numbers, and there is even an event to detect when the micro:bit is held up.

Let's try it now!



1. Clear your code
2. Drag and drop the '**on shake**' block from **Input**
3. Change the '**shake**' event to '**logo up**'

Playing Salute with micro:bits

4. Drag and drop the '**show string**' block from **Basic** into the '**on logo up**' block.
5. Choose the '**pick random …**' block from **Math** and drag it into the '**show string**' block.
6. Set the numbers to **1** to **9** in the '**pick random …**'. For an added challenge, add bigger numbers.

Try holding the micro:bit up to your head. A random number will appear. Time to play!

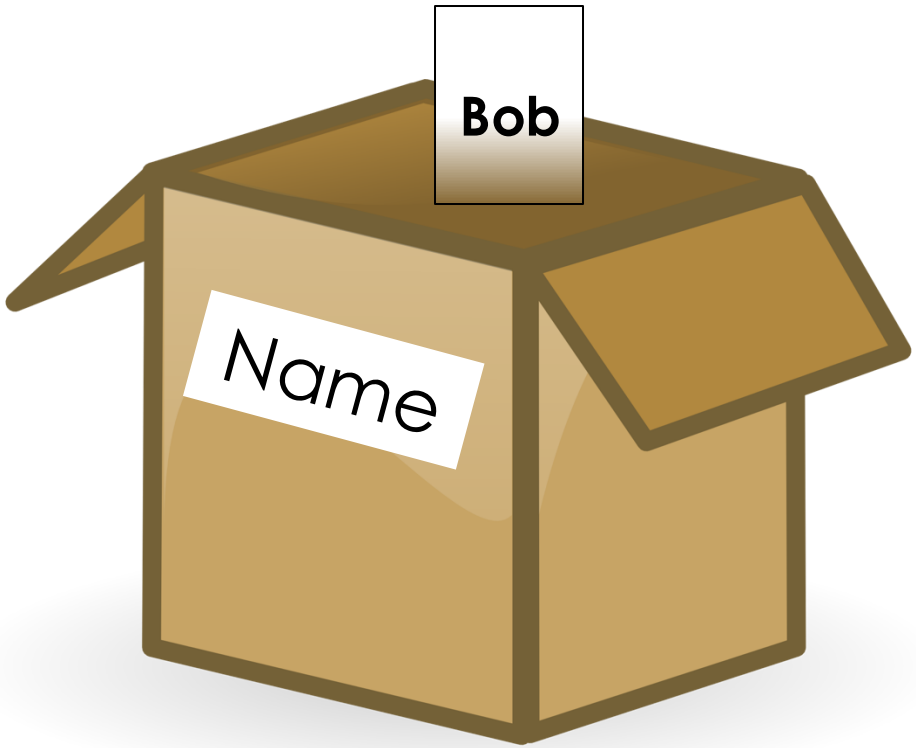
Variables

What Is a Variable?

A **variable** is something that stores data in our program. It is like a box with a label on it.

I can store different things in the box, but the label stays the same.

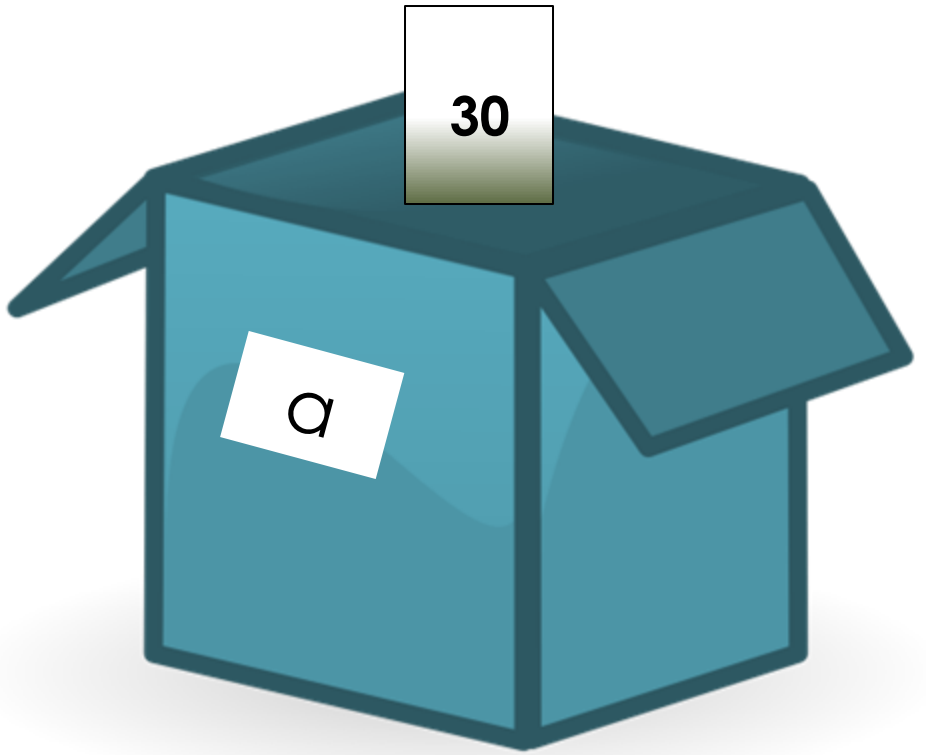
For example, I have stored the word "Bob" in my **variable** which is labelled "Name".



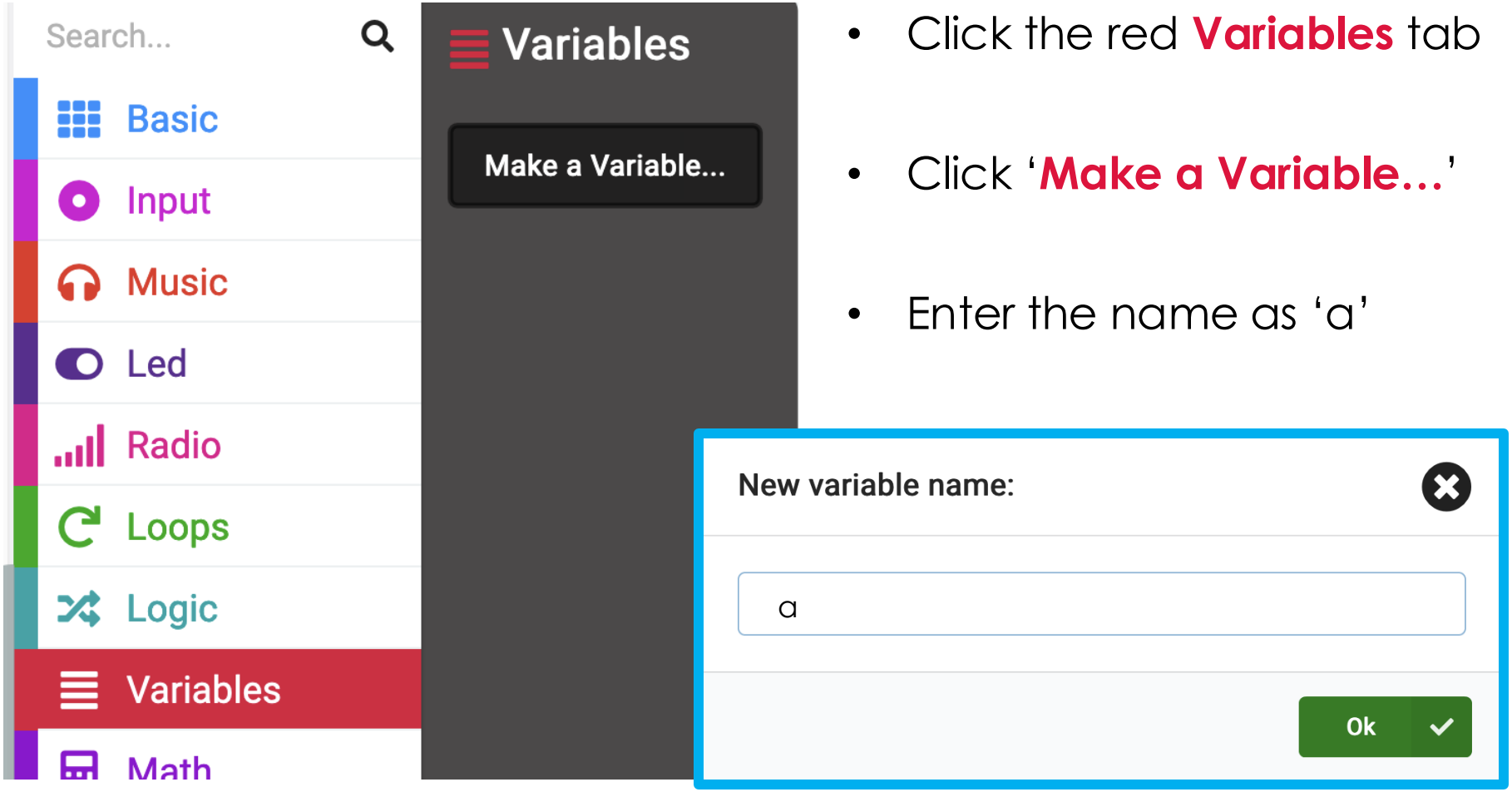
Humidity Variable

We will make a **variable** named '**a**' which will remember the most recent reading from the sensor.

If we program our micro:bits correctly, this will help our program to work properly, even if the sensor gives an error message.



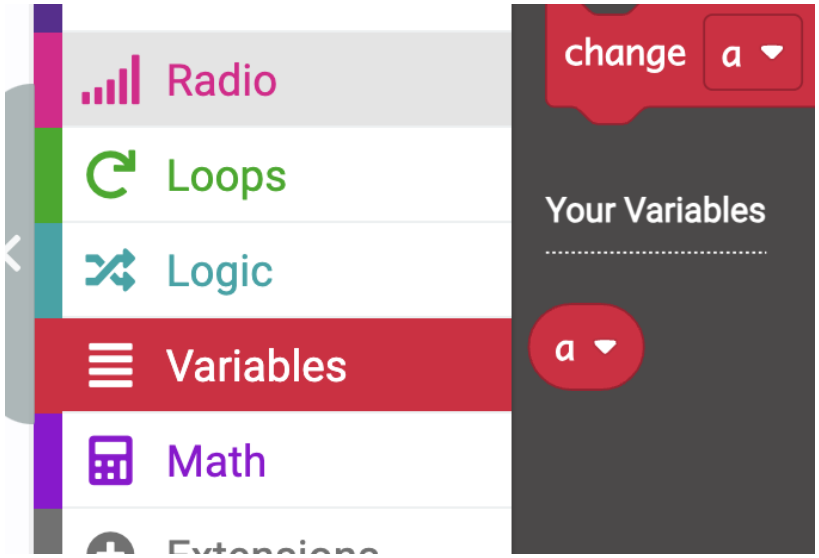
Fixing the Sensor



- Click the red **Variables** tab
- Click '**Make a Variable…**'
- Enter the name as 'a'

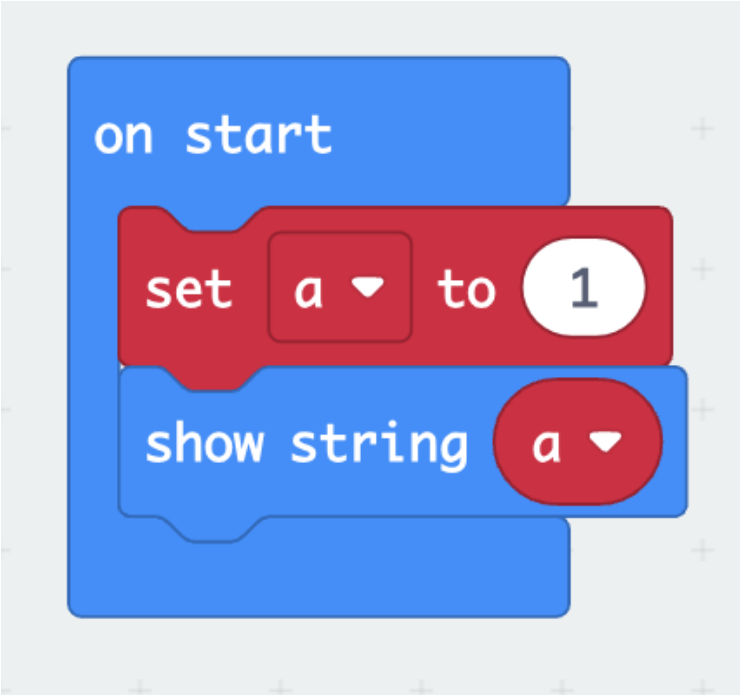
Variables

- In micro:bit variables are found under the **Variables** section.
- We can use variables to do all sorts of things. Remember the counter we made earlier? We can make one that we can control ourselves using variables.



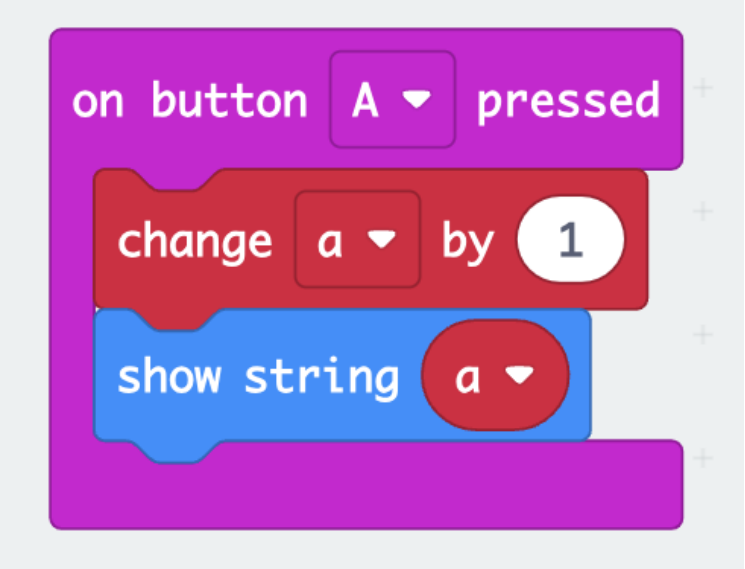
Making a Counter

1. Add an '**on start**' block
2. Go into the **Variables** section and choose the '**set a to 1**' block
3. Add a '**show string**' block and go back to **Variables**
4. Drag and drop the '**a**' from 'Variables' into the '**show string**' block



Making a Counter

5. Add an '**on button A pressed**' event.
6. Go back to **Variables** and choose the '**change a by 1**' block.
7. Copy the '**show string a**' block from before and add that too.



Making a Counter

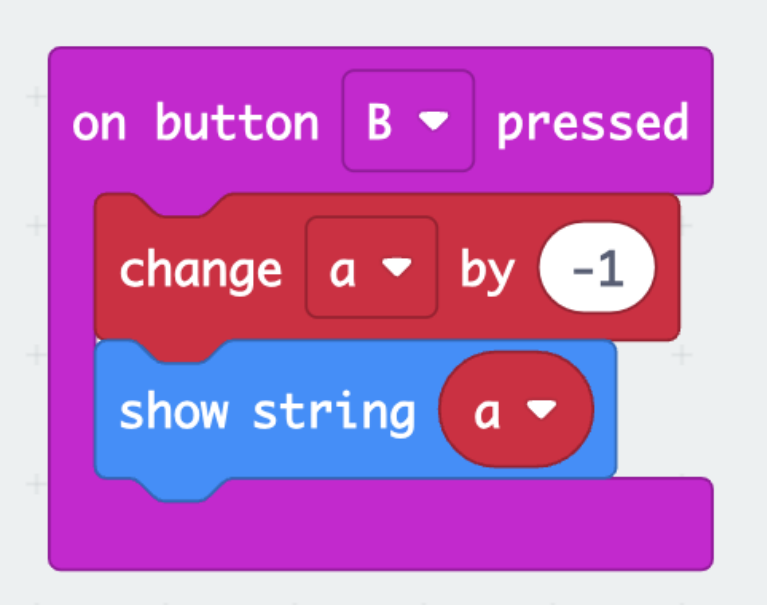
It's even possible to decrease the counter using **Button B**. Can you figure out how to do it?

Hint: you can use negative numbers in the '**change a by …**' block.

Making a Counter

It's even possible to decrease the counter using **Button B**. Can you figure out how to do it?

Hint: you can use negative numbers in the '**change a by …**' block.



technoteach
technocamps

Activity: Scoreboard

Making a Scoreboard

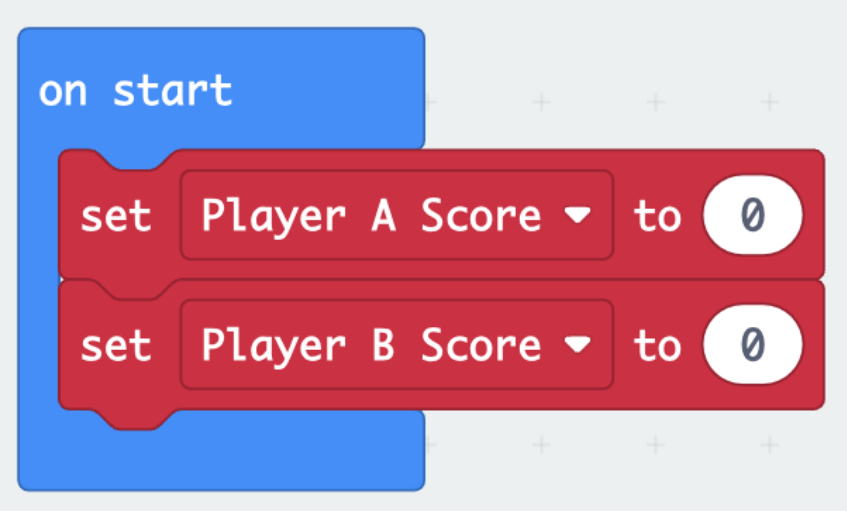
Now we'll make a scoreboard for our game of Salute! The third player will keep track of the score.

1. You will need to create two **variables** - one for player A and one for player B
2. Set Player A Score and Player B Score to 0 using the '**on start**' block.

Making a Scoreboard

Now we'll make a scoreboard for our game of Salute! The third player will keep track of the score.

1. You will need to create two **variables** - one for player A and one for player B
2. Set Player A Score and Player B Score to 0 using the '**on start**' block.

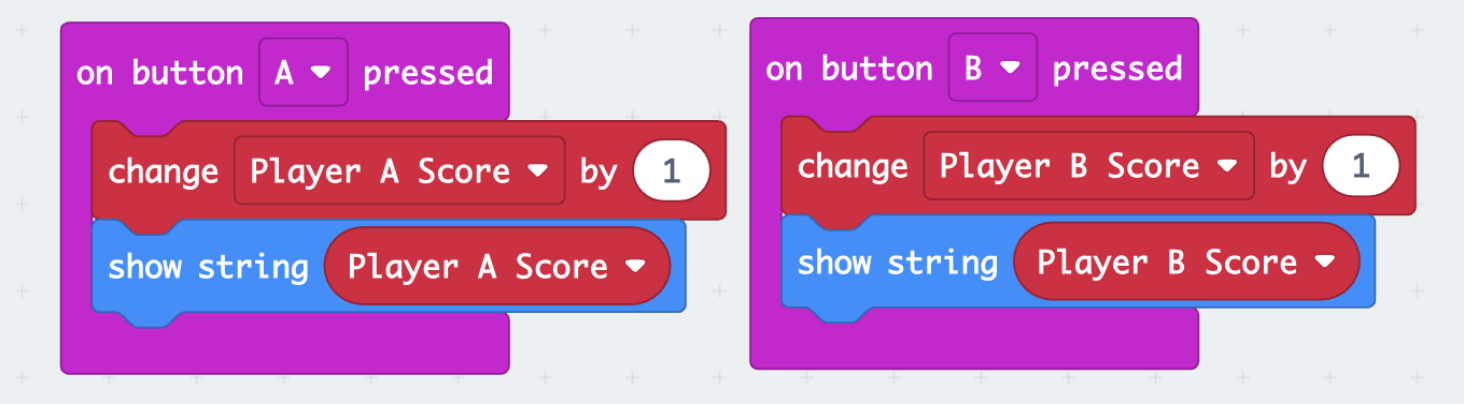


Making a Scoreboard

3. Set **button A** to increase **Player A Score** by 1 and display it.
4. Set **button B** to increase **Player B Score** by 1 and display it.

Making a Scoreboard

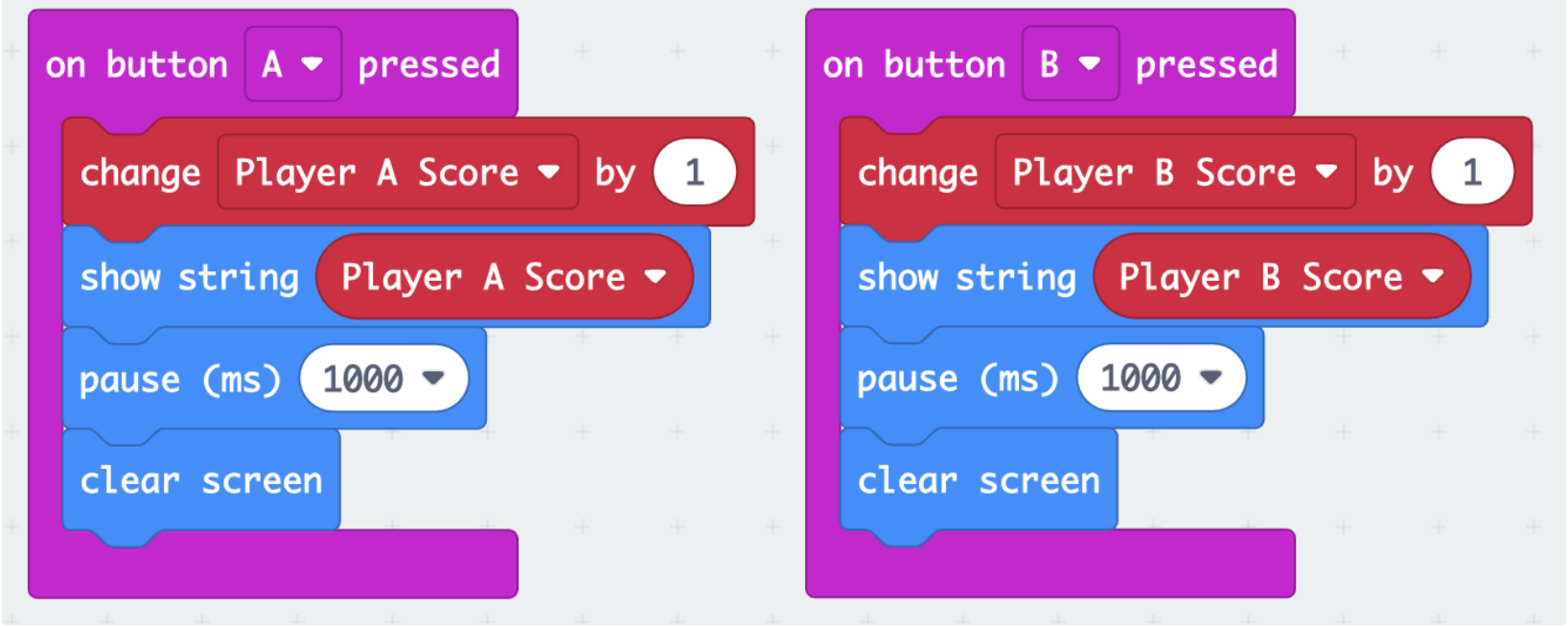
3. Set **button A** to increase **Player A Score** by 1 and display it.
4. Set **button B** to increase **Player B Score** by 1 and display it.



Making a Scoreboard

Once we increase the score, we want to clear the screen.

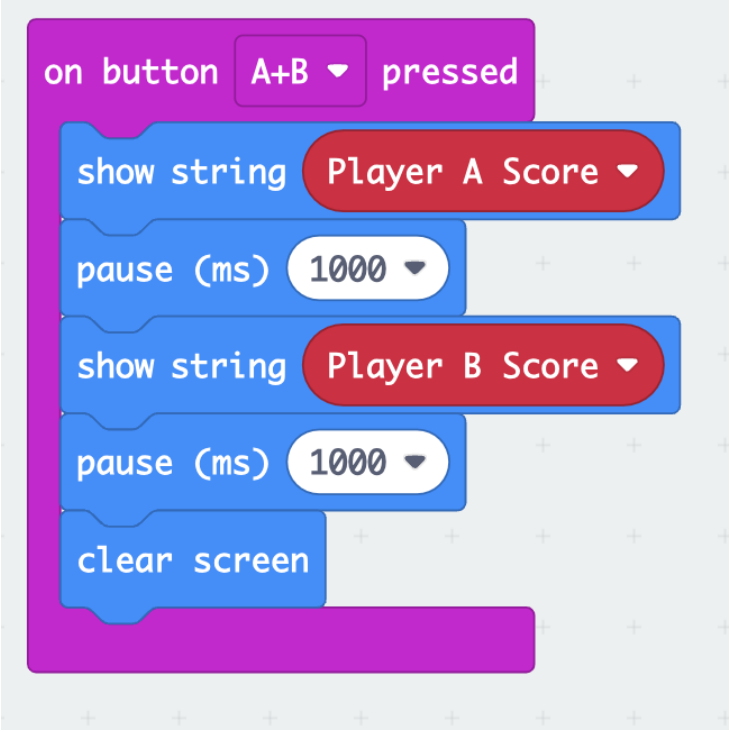
5. Under each '**show string**' block, add a '**pause (ms) 1000**' block followed by a '**clear screen**' block from **Basic**.



Making a Scoreboard

Now we need a way to check the current scores. There is an event for when both Button A and B are pressed at the same time.

1. Add an '**on button A+B pressed**' block
2. Now, add a '**show string**' block to show '**Player A Score**'.
3. Add a '**pause (ms)**' block and set it to **1 second**.
4. Add another '**show string**' to show '**Player B score**'.
5. Add another '**pause**' block and clear the screen.



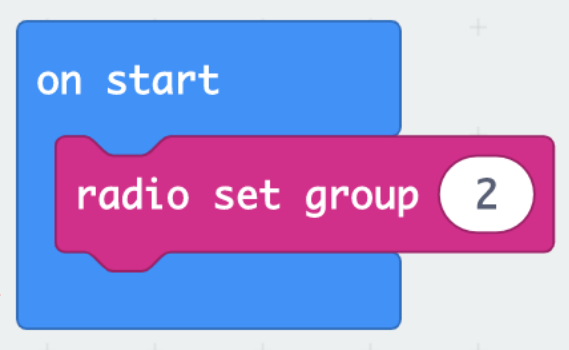
technoteach
technocamps

Languages, Literacy & Communication: Morse Code

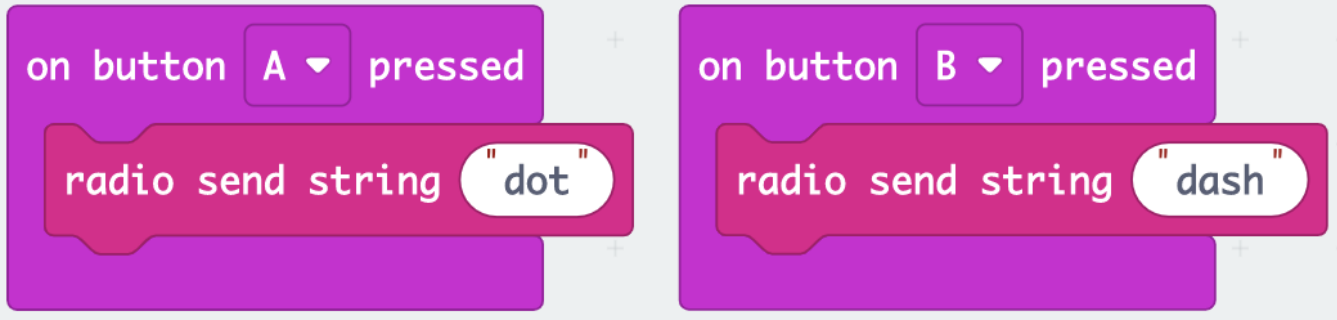
Micro:bit Morse Code

Using everything you have learnt today we are going to program the micro:bits to communicate using Morse code.

1. Start by setting both micro:bits to the same radio group.



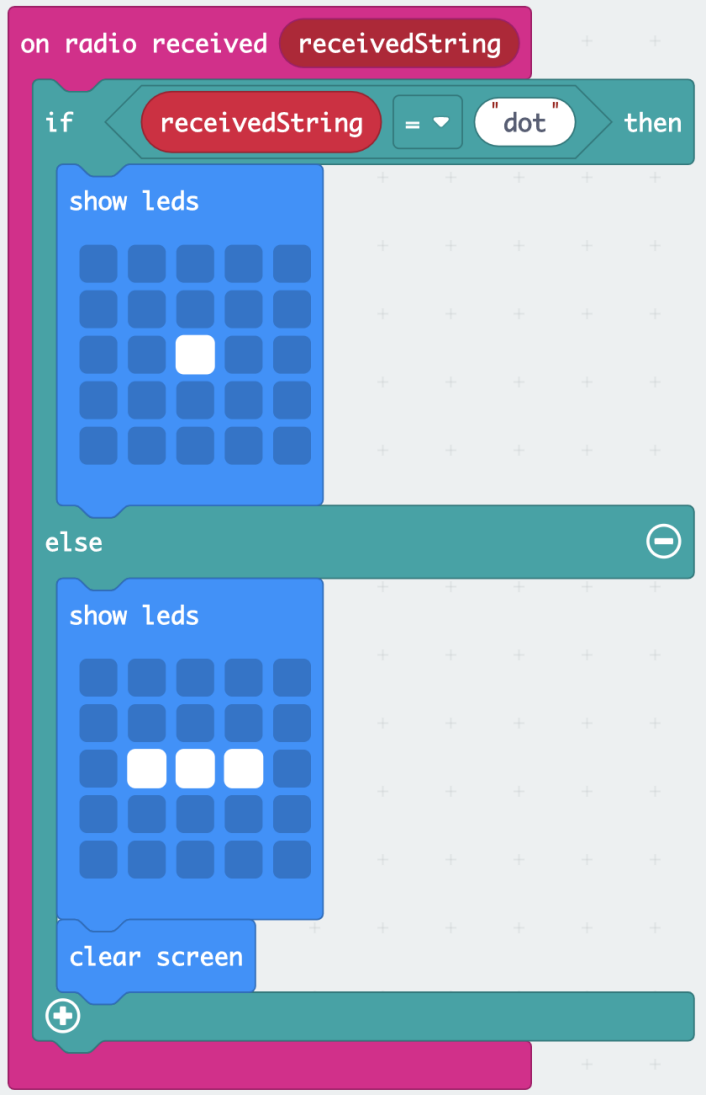
2. Use events to send a different messages for each button



Micro:bit Morse Code

3. Use an if-else statement and the show leds block so that the receiving micro:bit will show the correct output.
4. Try sending some messages to each other. Use the Morse code key and see if you can work out what your friend is trying to say.

Remember to leave a time gap between letters so that your partner knows when to start a new letter.



Extension Activity: Word Breaks

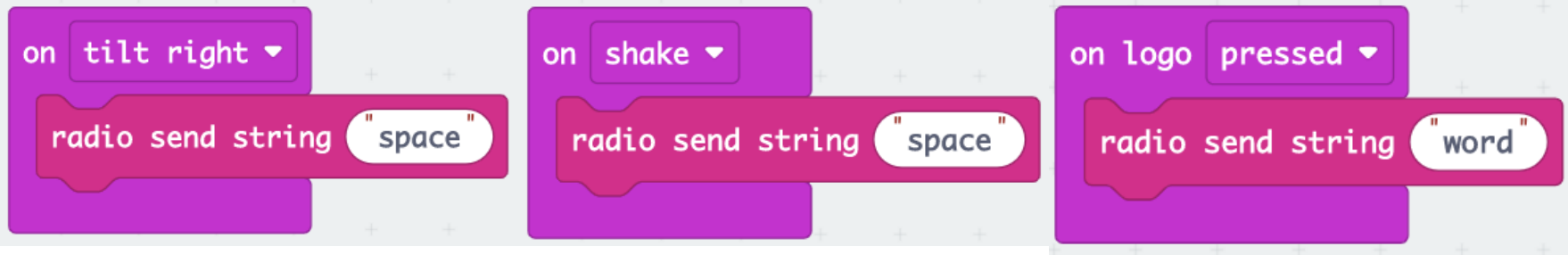
Word Breaks

- Can you think of some problems with using our Morse code radio? What if we want to send entire sentences?
- How does the receiver know when a new word is starting?
- If they don't know when to add a space, they could get a message like "iamlearningtocodewithmicrobit" which can be difficult to understand

- We can add a symbol to represent a word ending.
- We've already used buttons A and B – can you find another input block we can use to send the space symbol signal?

Word Breaks

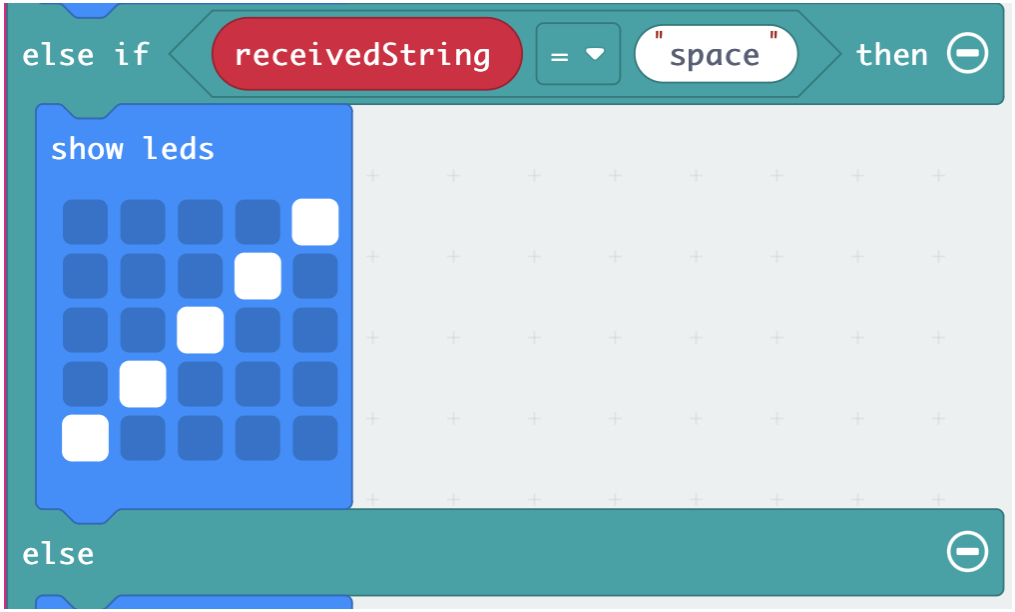
Input block examples:



There are many more options – try a few!

Word Breaks

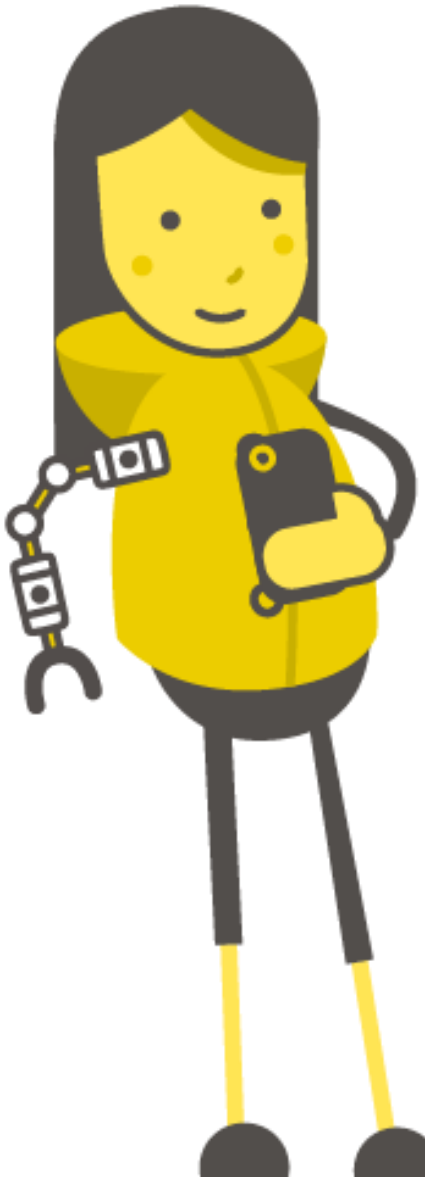
- You will also have to add another condition in the if block.
- Choose an icon to display when a word ends



Letter breaks

- Instead of leaving a time gap between letters, you can create yet another symbol to represent a letter ending.
- This will make it even easier for the receiver to understand the message, and quicker for the sender to send the message.

technocamps



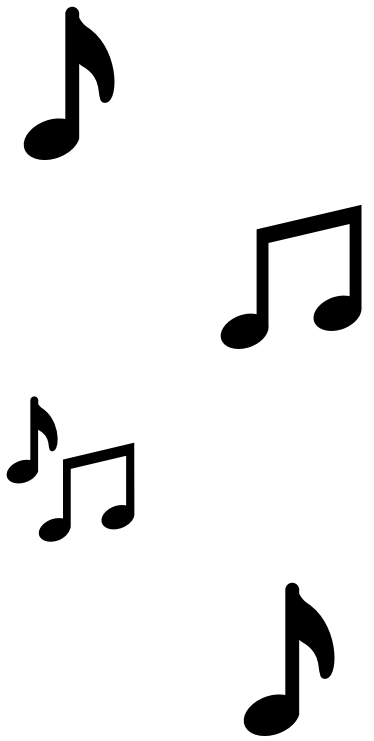
Expressive Arts: Musical Microbits

Jukebox

- A jukebox is a music playing device that gives you the choice of multiple songs to play.

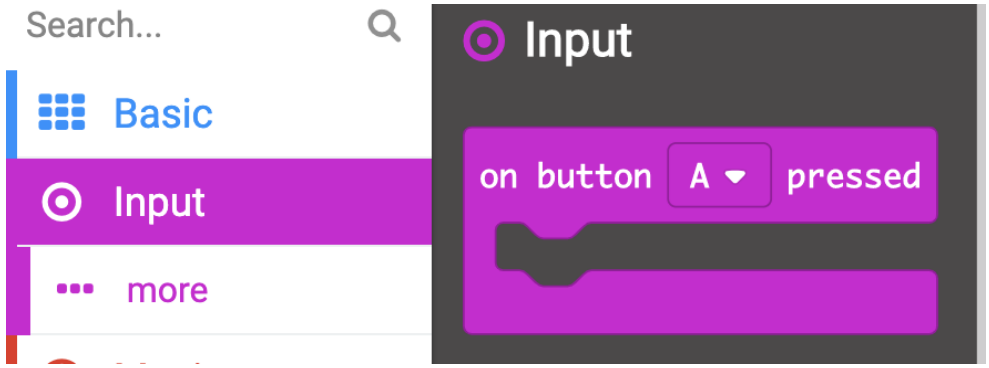
- So far, our micro:bits can only play one song. How can we program it to play several different songs? And how can we allow the user to choose which one to play?

- We need to use events.



Events

- Each micro:bit has two buttons, A and B.
- These buttons help us choose which action to take without reprogramming the micro:bit each time.
- For example, we can play one song when we press button A, and a different song when we press button B.
- The commands we need are found in the input section.



Events

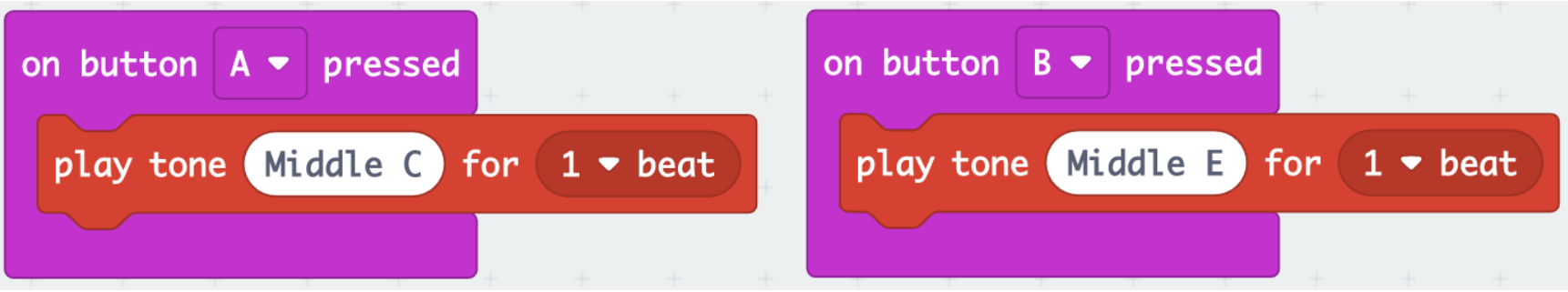
Let's try using an event for button A:

1. Click on Input
2. Drag and drop the on button A pressed block into your code
3. Now click on basic and drag and drop the play tone command into the event block
4. Choose you're a note to play and download the code.

What happens?

Multiple Events

- We can use multiple input commands – one for each button.
- Try adding an input for button B, the same way you did for button A.
- Try running the code.



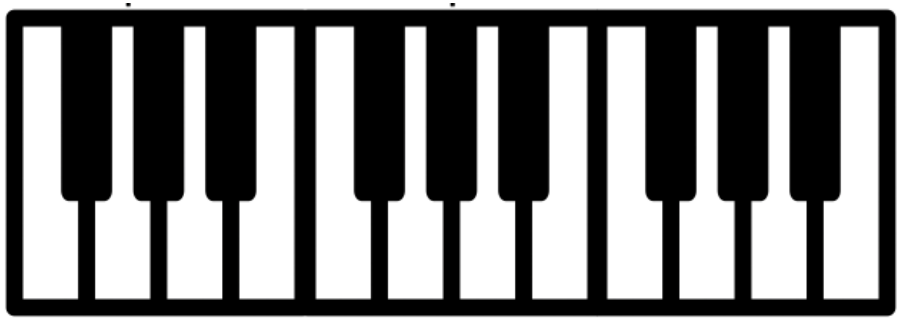
Jukebox

- Create your own jukebox by adding a different song to each input block.

- You can make your own melody using the play note commands or use the existing melodies in the Music section.

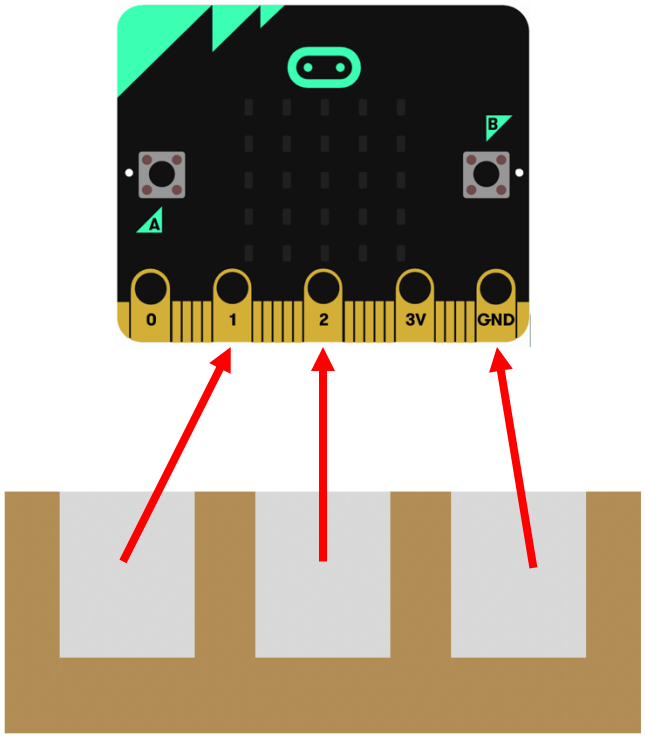
Micro:Bit Piano

- Instead of using the buttons to play notes or a melody, we can connect our own input device.
- To do this, we use the pins at the bottom of the micro:bit, just like you did to connect the headphones.
- We can make our own instrument. When we play it, an input signal will be sent to the micro:bit. The micro:bit will then produce output.



Connecting the Piano

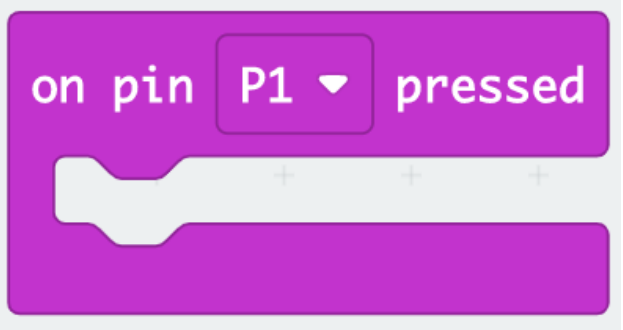
- Attach three crocodile clips to the cardboard piano.
- Connect the other ends of the crocodile clips to pin 1, pin 2, and the GND pin.



Activity: Piano Code

Pin Input

- Instead of using the buttons as input, we use the pins. The "keys" on our piano are connected to the pins.
- When we press on one of our keys and the ground pin, we complete an electrical circuit.
- Try adding input for pin P1 and pin P2. Give the code some music to play, run the code, and see what happens when you play your piano.



Pin Input

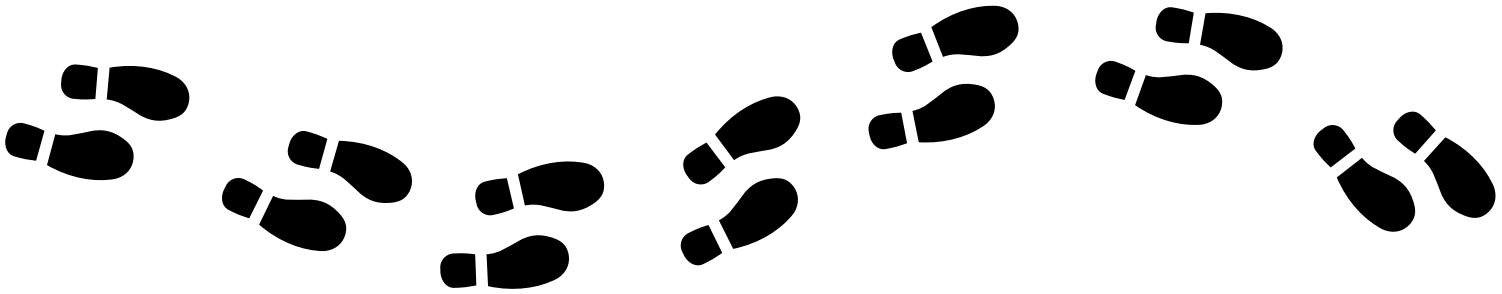
This is what your code should look like:

Experiment with some different musical outputs and see which songs you can play.

Health and Wellness: Pedometer

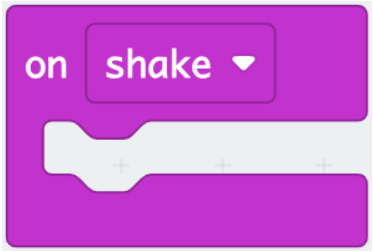
Step Counter

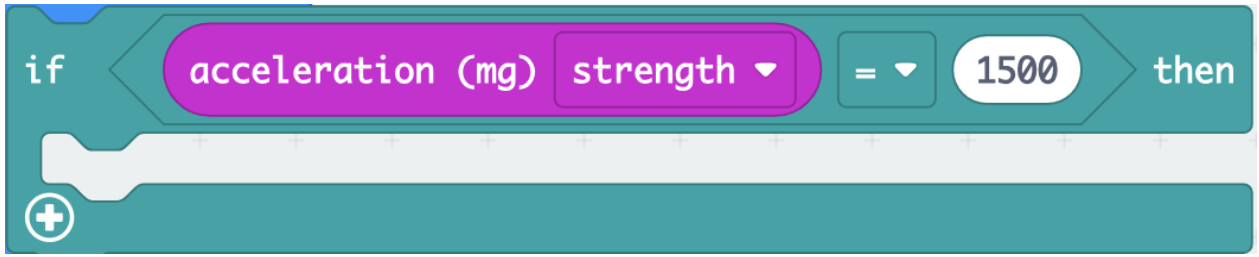
- A step counter (or pedometer) will work in the a very similar way to the exercise counter, but we want the micro:bit to react to movement instead of pressing a button.
- How do you think we can use movement as an input?



Movement Sensing

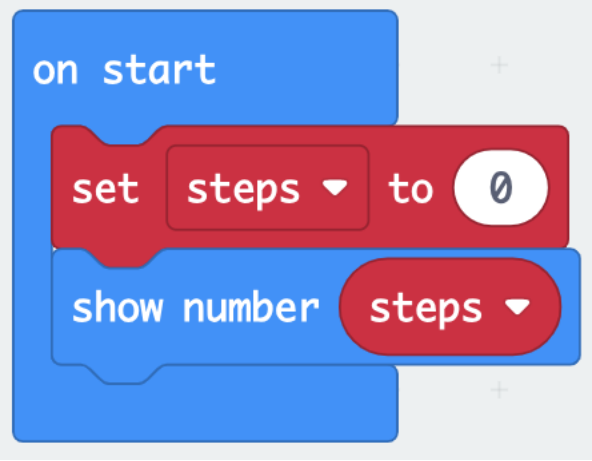
- One option is to use the on shake input. The micro:bit will carry out an action when it is shaken. However, this can be unreliable.
- Another option is to make the micro:bit react to a certain amount of acceleration. This value can be changed so can be customized to each person's step.
- Try making a step counter using this if statement.

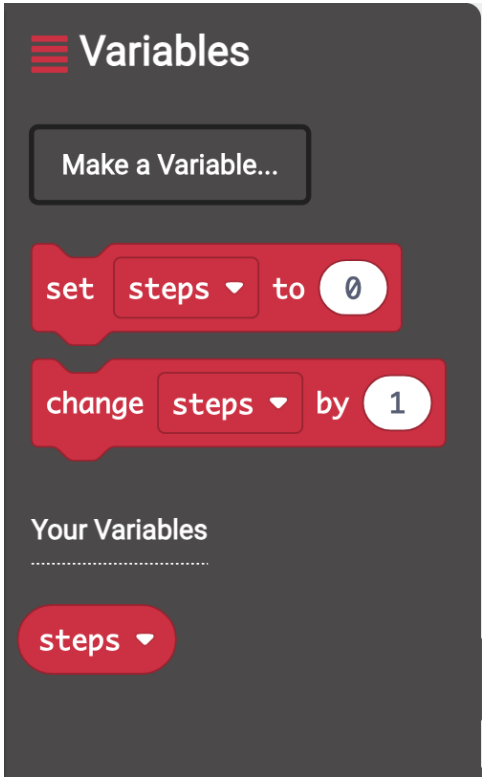




Step Variable

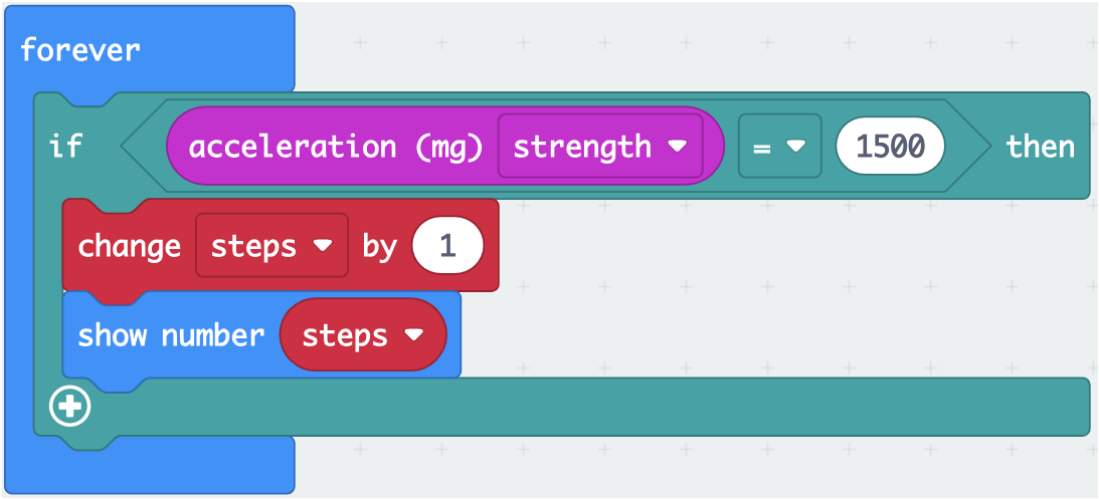
- Once again, we need to start by making a variable that we can modify. Let's call this "steps".
- Set the step count to 0.





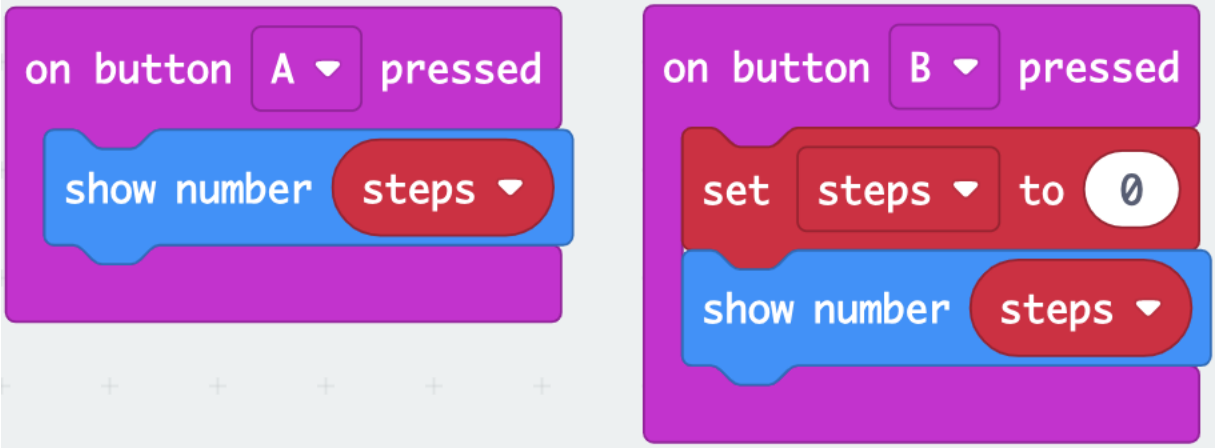
Step Counter

- Add the if statement to the forever loop and add the change steps command.



Step Counter

- You can add some commands to the button A and button B input blocks.
- This will allow you to reset the counter.



Step Counter

- Try running the code and see if it works!
- You can attach the micro:bit to your shoe using an elastic band.
- Does it accurately count your steps? You may need to change the acceleration value.