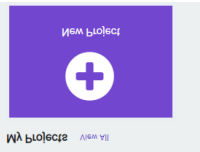


Micro:bit Throw the Dice Game

Click New Project



Give it a new e.g. Throw The dice

 A dialog box titled 'Create a Project' with a close button (X) in the top right corner. Below the title, there is a text input field containing the text 'Throw the Dice'. Below the input field, there is a link that says '> Code options'. At the bottom right of the dialog, there is a green button labeled 'Create' with a checkmark icon.

Click 'create'

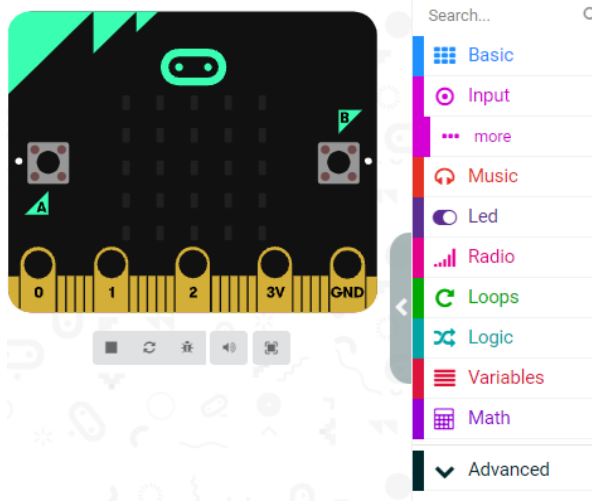
Describe the screen



The basic idea is to select code-blocks from the central toolbox, drag them to the programming canvas, run the programme and view the results on the simulator.

The simulator can sense shake, button A and button B

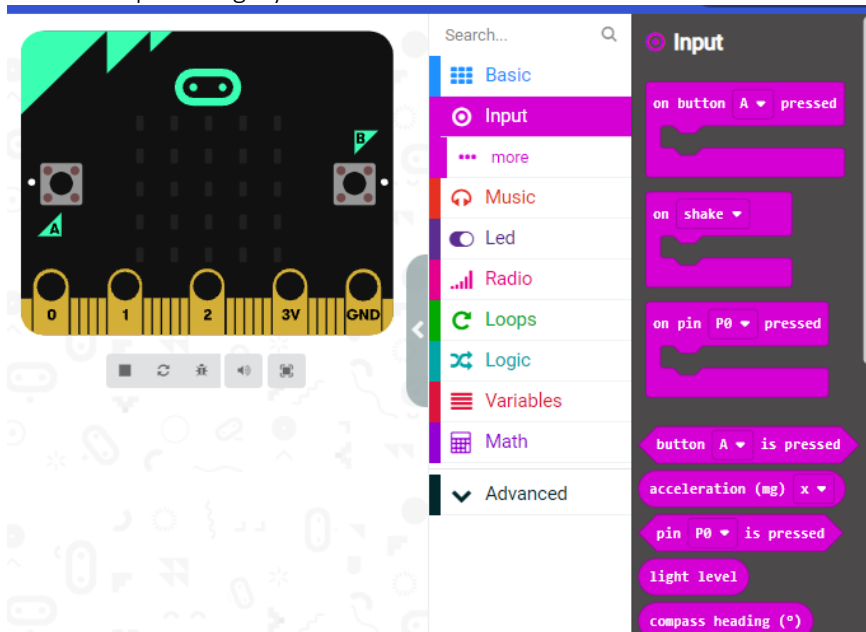
THROW THE DICE GAME



You can use the search (just above the toolbox categories) to find the correct block to use or use the colour codes, when you click on code block category it will show you all the blocks for that category.

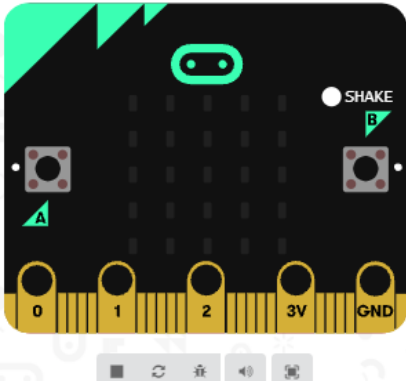
Lets start making our first programme today – Throw the dice

We want to detect a 'shake' event – this will be an input from the simulator. Click the input category in the toolbox.



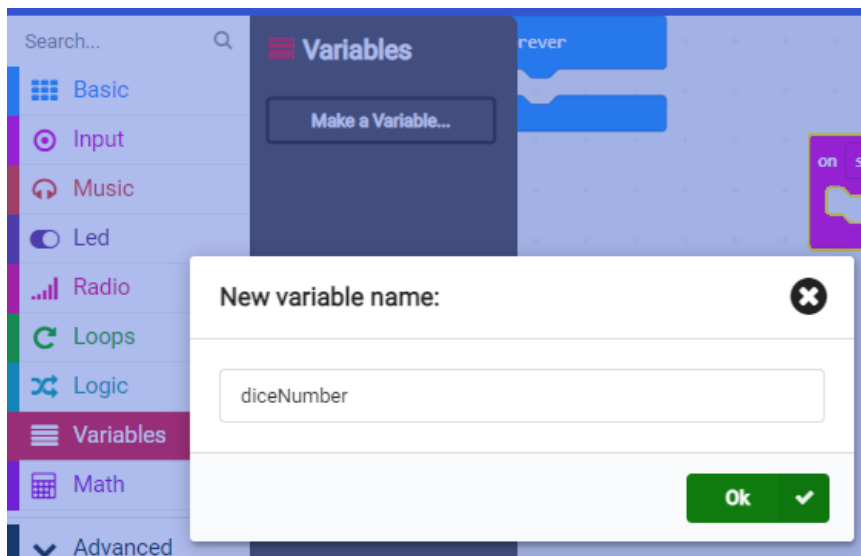
Now drag the on shake code block on to the canvas.

NOTE: the micro:bit on the left will go grey (this is while the makecode is 'compiling or making the programme) and when it returns to colour you there is a shake button on it!!



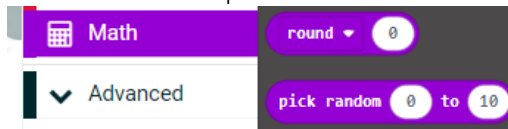
Note: the beginning of a programme is used to initialise or set-up the requirements for the programme, in this case we need to make a variable for the number to be displayed on the dice (LED)- I called it diceNumber –

Note: There is no space between the word dice and Number, also I used a capital or upper case letter for Number so that it makes it easier to read – called camelised text – do you know why? Think of a camel!



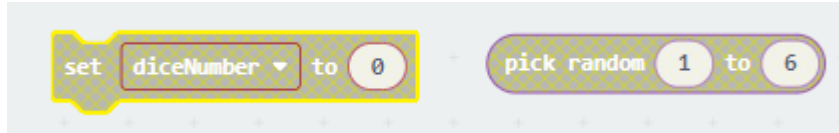
Random Number:

Now in the Math toolbox there is a block called pick random –



Drag this onto the canvas for now, change the value 0 to 1 and the value 10 to 6 – this will create a random number from 1 to 6.

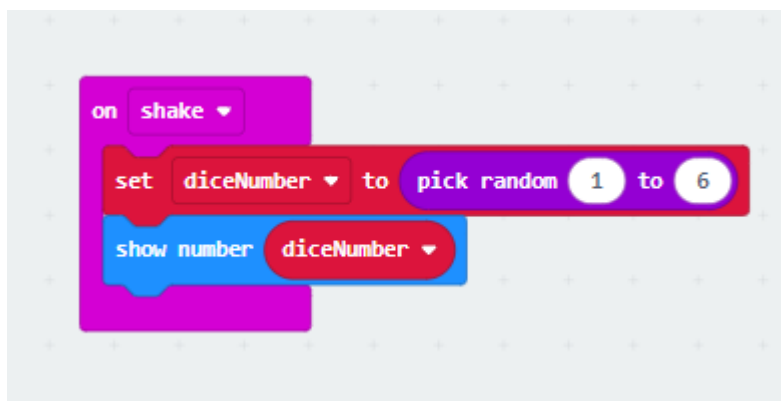
Now we need to set the value of diceNumber to the pick random value, go to the Variables toolbox then using the 'set diceNumber to' drag that to the canvas.



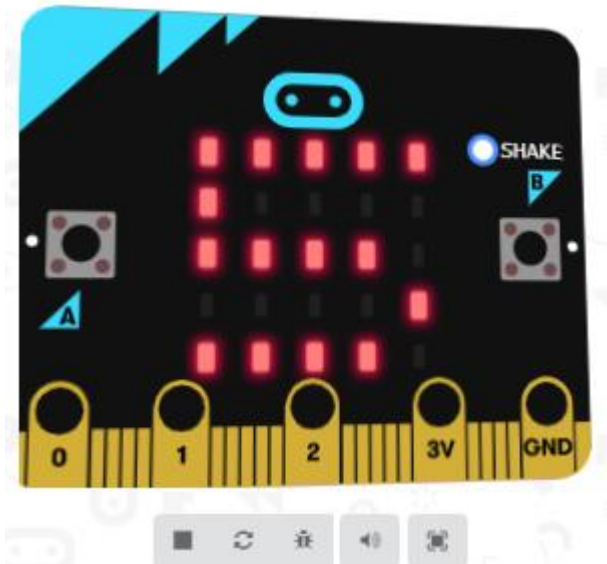
Now we just drag the pick random to the '0' in set (you will see two red dots when you get close.) Move the set block into the middle of the on shake block.



Now the last part is to display the number – in the Basic toolbox there is a 'show number' block, replace the '0' with the variable diceNumber and we are all set



To 'Throw the Dice' all you need to do is to click the white circle beside the word 'SHAKE' on the micro:bit.



Well done.

So, you should see that every time you click the white button the number displayed changes – of course it is possible every now and then that the same number will be picked one after the other – its random!!!!

JavaScript Version.

So we used block code for that implementation of the solution – there are loads of different programming languages that can run on a micro:bit including JavaScript and python. On the top of the screen there are two buttons



Or



Try changing the button to JavaScript

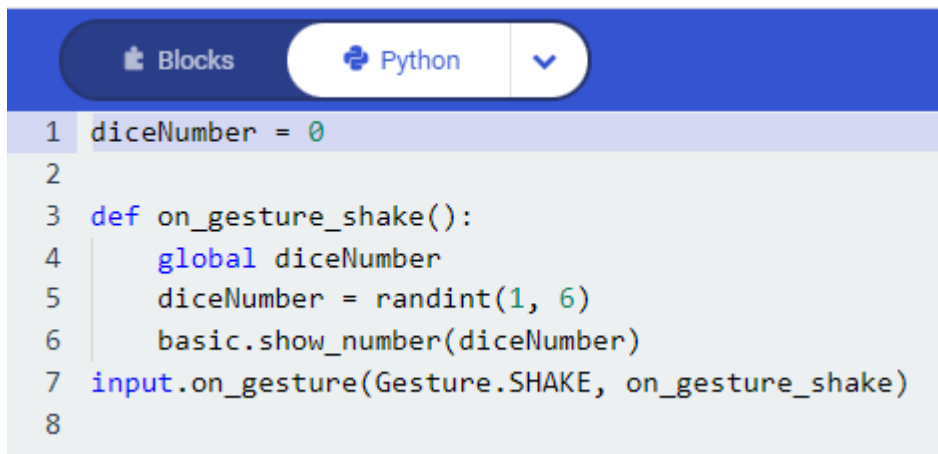
The Programming canvas now looks like this;

```

1 let diceNumber = 0
2 input.onGesture(Gesture.Shake, function () {
3     diceNumber = randint(1, 6)
4     basic.showNumber(diceNumber)
5 })
6

```

If you change it to Python (use the down arrow)

A screenshot of a code editor interface. At the top, there are two buttons: 'Blocks' and 'Python'. The 'Python' button is selected and highlighted. Below the buttons, there is a code editor with the following Python code:

```
1 diceNumber = 0
2
3 def on_gesture_shake():
4     global diceNumber
5     diceNumber = randint(1, 6)
6     basic.show_number(diceNumber)
7 input.on_gesture(Gesture.SHAKE, on_gesture_shake)
8
```

These two languages are different yet similar in a way. FYI randint stands for random integer – an integer is a whole number i.e. no decimal places.