BBC micro:bit
Dice Rolling

Introduction

Here is a short example of how to make a simple dice rolling animation.

Programming

A fairly short program is needed to make this possible.

```
from microbit import *
import random

faces = [Image('00000:00000:00900:00000:00000:'),
        Image('00009:00000:00000:00000:90000:'),
        Image('00009:00000:00900:00000:90000:'),
        Image('90009:00000:00000:00000:90009:'),
        Image('90009:00000:00900:00000:90009:'),
        Image('90009:00000:90009:00000:90009:')]

def RandomImages(n, delay):
    for i in range(0,n):
            display.show(random.choice(faces))
            sleep(delay)
            display.clear()
            sleep(delay)
while True:
    if button_a.was_pressed():
        RandomImages(20, 75)
        rolled = random.choice(faces)
        display.show(rolled)
```

The faces list is a list of images of the possible die rolls. Dots are used in the classic way.

The RandomImages function is not strictly necessary. Before choosing the final roll, a sequence of randomly selected images is shown blinking on the matrix.

This code never works with the numbers. If you needed to use the die roll in part of a game, generate a random integer from 0 to one less than the number of images. When you need to work with the image (to display it), you write image_list[random_integer], where these words are replaced by your own variables which store those things.

Challenges

1. Replace the images here with your own selection. Perhaps use that as part of a game.
2. Instead of choosing randomly from the list, you could use it as a series of images in sequence, as an animation. You can make lists of lists. If you make several lists of images that each make up a short sequence of movements, you can make a random animation. Get the right mix of sequences and you can create something quite effective.

BBC Microbit

Collapse All | Expand All

+ Block Editor - The Basics
+ Block Editor - Components
+ Kodu - micro:bit Worlds
+ JavaScript Blocks
+ JavaScript Blocks - Exercises
+ Blocks - Bit:Bot
+ Blocks - Bit:Commander
+ MicroPython - Starting Off
- MicroPython - Examples
  - Dice Rolling
  - Shut The Matrix
  - Encoding Morse Code
  - Encoding Ciphers
  - Drawing A Maze
  - Scrolling Race Track
  - Vertical Scroller
  - Concentration Game
  - Text Entry - Accelerometer
  - Charlie's Python Game
  - Lights Out Game
  - Sam's French Number Game
  - Bryn's Concentrated Clocks
  - Lattice Paths
  - Knight Moves
+ MicroPython - Components
+ MicroPython - Breakout Boards
+ MicroPython - Exercises
+ MicroPython - Pi Accessories
+ MicroPython - Bit:Bot
+ MicroPython - Bit:Commander
+ MicroPython - Projects
+ MicroPython - Visual Basic
+ Other - Odds & Ends