

Coding Fundamentals



Micro:bit Python Programming Programming with Displays

Overview

In this lesson, students learn to scroll text on the micro:bit. In this lesson, students learn to display an image on the micro:bit LEDs using Python.

Objectives

- Identify components on the Micro:bit
- Create code for Micro:bit using a Python editor
- Define the functions `display.set()` and `display.show()`
- Use x-y coordinates to reference grid of LEDs on Micro:bit

Materials

- micro:bit and micro-USB cord
- Computer with access to the internet

Approx. Time Required

1-2 hours

CYBER.ORG
THE ACADEMIC INITIATIVE OF THE CYBER INNOVATION CENTER

Cyber Connections

- **Programming** – Students will program in Python.
- **Hardware and Software** – Students will utilize small electronics and learn how a computer is programmed while using micro-controllers.

This content is based upon work supported by the US Department of Homeland Security's Cybersecurity & Infrastructure Security Agency under the Cybersecurity Education Training and Assistance Program (CETAP).

Programming with Displays

- Start the lesson by reviewing the parts of the Micro:bit with students or giving a short quiz on the vocabulary and parts of the Micro:bit from the previous lesson.
- Remind students of the important line of code that should be included at the top of every program they create for the Micro:bit.

```
from microbit import *
```

- Review `display.show()`. Demonstrate the display of the heart in lesson 01 and look at some of the other images that are built in to the micropython programming language. Specifically look at `Image.CHESSBOARD`, `Image.HOUSE`, and `Image.XMAS` as students will be recreating these images later in the lesson.
- This lesson focuses on a command called `display.set_pixel()`. This command is used to change the light levels of individual pixels. When using `display.set_pixel(x, y, i)`, students must specify three arguments: the x-coordinate, the y-coordinate, and the intensity level. The x and y coordinates range from 0 to 4, starting with 0,0 in the top left corner of the Micro:bit display. The intensity levels range from 0 to 9. The standard for each LED is 0 which means that it is off. The code to turn the top left LED to full light intensity would look like this:

```
display.set_pixel(0,0,9)
```

- After students become familiar with the command and the use of coordinates, give them the task of recreating the following images with their own code: `Image.CHESSBOARD`, `Image.HOUSE`, and `Image.XMAS`. The code for each of the images is included below.
- Challenge students to create their own images using varying light levels. Make sure they keep track of the coordinates of their images to use in the next lesson.

Chessboard

```
from microbit import *

display.set_pixel(1,0,9)
display.set_pixel(3,0,9)
display.set_pixel(0,1,9)
display.set_pixel(2,1,9)
display.set_pixel(4,1,9)
display.set_pixel(1,2,9)
display.set_pixel(3,2,9)
display.set_pixel(0,3,9)
display.set_pixel(2,3,9)
display.set_pixel(4,3,9)
display.set_pixel(1,4,9)
display.set_pixel(3,4,9)
```

House

```
from microbit import *

display.set_pixel(2,0,9)
display.set_pixel(1,1,9)
display.set_pixel(2,1,9)
display.set_pixel(3,1,9)
display.set_pixel(0,2,9)
display.set_pixel(1,2,9)
display.set_pixel(2,2,9)
display.set_pixel(3,2,9)
display.set_pixel(4,2,9)
display.set_pixel(1,3,9)
display.set_pixel(2,3,9)
display.set_pixel(3,3,9)
display.set_pixel(1,4,9)
display.set_pixel(3,4,9)
```

Xmas

```
from microbit import *

display.set_pixel(2,0,9)
display.set_pixel(1,1,9)
display.set_pixel(2,1,9)
display.set_pixel(3,1,9)
display.set_pixel(2,2,9)
display.set_pixel(1,3,9)
display.set_pixel(2,3,9)
display.set_pixel(3,3,9)
display.set_pixel(0,4,9)
display.set_pixel(1,4,9)
display.set_pixel(2,4,9)
display.set_pixel(3,4,9)
display.set_pixel(4,4,9)
```