MicroPython and Microcontrollers

NetApp YWIT

May 1, 2020

Contents

Chapter 1

Introduction

This workshop will introduce the student to Python coding, electronics, and project design. We will building several projects ranging from simple to complicated. These projects are based on the ESP8266 microcontroller which is running MicroPython and they depend on some other electronics components such as LEDs, buttons, and more.

Chapter 2

Project 1: Blink

Chapter 3

Project 2: Button

Chapter 4

Project 3: LED Party

Chapter 5

Project 4: Sensor

Chapter 6

Project 5: Game

Chapter 7

Project 6: Chat

Chapter 8

Project 7: Sound

Appendix A

Electronics Essentials

Appendix B

Python Primer

If you're new to Python, this section will give you a few things you should know in order to better understand the projects in this guide. This is by no means a complete or comprehensive look at the Python language. For that, we recommend looking at the official Python site and reading through the tutorial there.

> NOte: for the projects being used here, we are using an implementation of Python known as MicroPython. This version is meant to run on microcontrollers with limited resources. It also has built into it libraries for dealing with hardware devices that are not part of the standard CPython distribution. Therefore, not all Python examples you find online will run on your microcontroller and not all projects for a microcontroller can be run on your computer. But a lot of the code can be shared so the lessons you learn here can apply to other Python projects.

Here is a sample of a small Python script. We will disect and explain what each section does below:

Listing B.1: An example Python script

```
1  def show(message, repeat=1):
2      """This function prints the given message to the
3      console as smany times as specified in the
4      srepeat parameter.
5      """
6
7      for iteration in range(0, repeat):
8          print(iteration, message)
9
10 name = input("What is your name: ")
11 show(name)
12 show(name, repeat=2)
```

On line 1, we are defining a function named show. This function accepts two parameters, message and repeat. The message parameter is required and the repeat parameter is optional with a default value of 1.

Lines 2 through 5 comprise the docstring for the function. This information is meant for programmers to read and explains what the function does. It does not affect how the function works.

Line 7 starts a loop. The loop will repeat the statements in the loop body until a condition is met. In this case, it will loop until it has performed the operation for each repeat.

Line 8 is the body of the loop. This statement will print the message that the user passed in to the console along with the iteration number of the loop.

Line 10 prompts the user for their name and saves the result in a variable called name.

Line 11 calls our show function which will print the user's name once (the default).

Line 12 calls our show function again, this time saying that we want to repeat the loop of printing the name twice.

Running the program, we will see output like this:

```
$ program program.py
What is your name: Emily
0 Emily
0 Emily
1 Emily
$
```