

Python Computing: Building a Sensor System



CSCI 250

Lecture 6: Chapter 2.1 – 2.4

Data Structures and Looping



COLORADO SCHOOL OF MINES
EARTH • ENERGY • ENVIRONMENT

Previously in CSCI 250

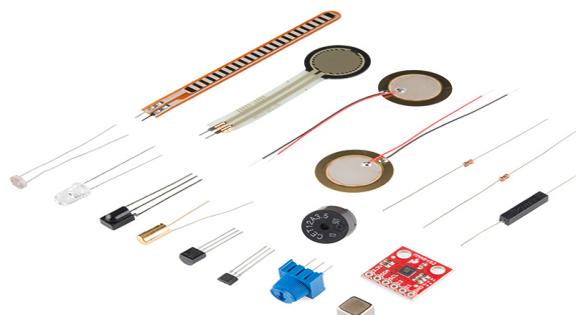
- + Assignments:
 - + Lab 1: Blink with Extensions – Due 1/31 11:59pm
 - + Read Chapter 2.1-2.4 from textbook, pp. 17-33.
- + Discussion from last week:
 - + Coding basics with Python - basic syntax, libraries, and variables.

Learning Outcomes

- + Course Objective: Develop and run basic Python functions and programs in the Linux environment to collect data from sensors using the Raspberry Pi Hardware (e.g., light, sound, motion, magnetism).
- + During this lesson, we will learn how to:
 - + Identify the difference between different data structures.
 - + Create, assign values, and manipulate data in arrays.
 - + Use control structures for repetitive tasks (e.g., for and while loops).

Today's Plan

- + Return the What's in the Box and GPIO Worksheets
- + The grouping of data or *data structures*
 - + Objects and their methods
 - + Built in data structures
- + The grouping of code into repeated blocks or *looping*
- + Next Lab: Light sensor



Objects and their methods

- + Objects are a combination of
 - + Attributes: data values and Methods: functions
- + Example:
 - + `myVal = 5.0` `powerVal.is_integer()`
 - + `powerVal = 2` `pow(myVal, powerVal)`

What are some other methods or functions we would want to do with numerical values?

Built-in Data Structures

- + In addition to the array, list additional data structures that can be used for the grouping of data?

Built-in Data Structures

- + In addition to the array, list additional data structures that can be used for the grouping of data?
 - + Lists
 - + Arrays
 - + Tuples
 - + Strings
 - + Dictionary
- + Discuss functions you would use with a set of data

Built-in Data Structures

- + In addition to the array, list additional data structures that can be used for the grouping of data?
 - + Lists
 - + Arrays
 - + Tuples
 - + Strings
 - + Dictionary
- + Discuss functions you would use with a set of data
 - + Length, insert, sort, find, adding, etc.

Loops

+ What are a few types and advantages of grouping of code into repeated blocks or *looping*?

+ For loop

```
for x in range(0, 3):  
    print x
```

+ While loop

```
x = 1  
while True:  
    print ("My Number is: ", x)  
    x += 1
```

Loops

+ What are a few types and advantages of grouping of code into repeated blocks or *looping*?

+ For loop

```
for x in range(0, 3):  
    print x
```

0
1
2

+ While loop

```
x = 1  
while True:  
    print ("My Number is: ", x)  
    x += 1
```

Loops

+ What are a few types and advantages of grouping of code into repeated blocks or *looping*?

+ For loop

```
for x in range(0, 3):  
    print x
```

0
1
2

+ While loop

```
x = 1  
while True:  
    print ("My Number is: ", x)  
    x += 1
```

My Number is: 1
My Number is: 2
My Number is: 3

...

Wrap Up

+ Discussion

- + The grouping of data or *data structures*.
- + The grouping of code into repeated blocks or *looping*.

+ Assignments

- + Lab 1 - Due tomorrow 1/31 11:59pm.
- + Python Practice: L6 – submit today's .py file – Due 1/30 11:59pm
- + Thursday - Lab 2: Light and working with more sensors.
- + Read Chapter 2.5-2.8 from textbook, pp. 33-45.

Python Practice : L6

- + This simple exercise is broken down in pieces to practice importing libraries, creating, storing, and accessing elements in an array**.
- + Create a new program, which creates a random value and prints the value to the screen.
- + Next, add code to create a numpy array initialized with 100 zeros and print that entire array to the screen.
- + Store 100 random values into an array using a **for** or a **while** loop.
- + Round the values to 2 decimal places.
- + Then print every other value to the screen one at a time using a **for** loop; yes, another loop is just fine for this practice.
- + Now, do the same things as above, using an empty list and the append command

** you can always combine steps if you wish