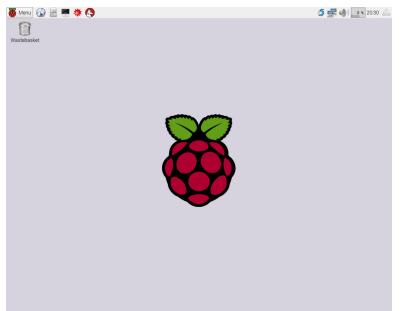
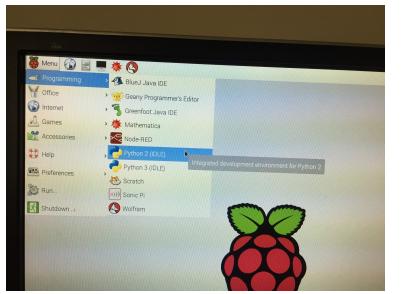
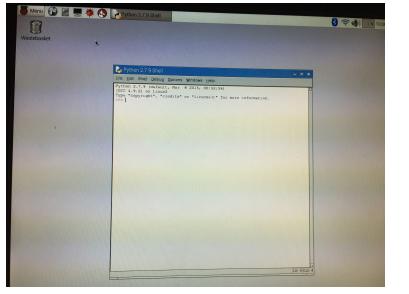


Instructions for Coding (1 LED strip)

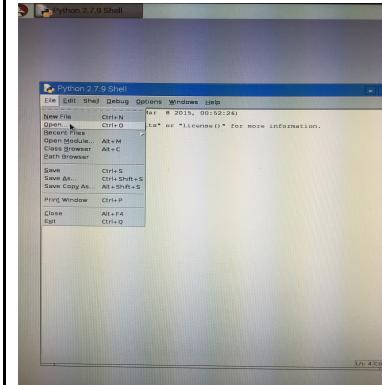
1. Running the test program
2. Exercise A
3. Using the Python Shell
4. Exercise B
5. Programming Cheat Sheet
6. Additional Examples

1. Running the test program

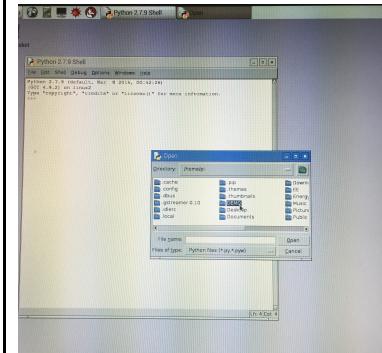
Raspberry Pi Screen	
<p>Step 1: Click on the menu icon (the smaller raspberry symbol on the top left corner. Then select, programming.</p>	
<p>Step 2: This gives a list programs available. We will be using Python2 (IDLE).</p>	
<p>Step 3: Click on Python2 (IDLE). This will give us the IDLE interface, where we can start programming!!</p> <p>IDLE stands for Integrated Development and Learning Environment. It is an easy way to write, save and run programs in ‘Python’ programming language.</p>	

Step 4: Let us see how to run a test program to light up the LED lights!

Click on File and pick Open..



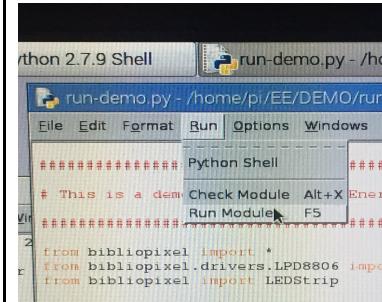
Step 5: Open the DEMO folder and open the file run-test.py.



Step 6: Once the file is open, you can run the program by click on Run to get a drop-down menu. Pick Run Module (F5) from the drop-down.

To run the program multiple times, you can simply click F5 instead of using the drop-down!!

Note: Don't worry if the program seems totally foreign at this point. We don't need to understand every line to start using the programs. (Just like you don't have to know how smartphones work to send texts!)



On running the program

You should notice the following:

- > The LED lights will turn on one at a time.
- > As the lights turn on, their number will appear on the IDLE screen. So as the first LED turns on, the IDLE screen should have the letters LED 1. So on till LED 10, when all LEDs have turned on.

2. Exercise A

Exercise A

- 1) What color were your LED lights?

Answer:

After the program finishes, click back on the run-demo.py screen and click Run → Run Module (or press F5) to run the program again.

- 2) What color are your LED lights now? Is it the same or different?

Answer:

Now, run the program again. This time start a timer when you hit enter. Stop the timer when the screen says ‘All LEDS!!!’. (If you don’t have a timer, simply look a clock and count the number of seconds before ‘All LEDS!!’ appears on the screen.)

- 3) How long did it take for the program to run fully?

Answer:

3. Using the python shell

Here we will be using the Python Shell to write simple commands to do specific LED tasks! We can think of the Shell as a program testing space, where we can try out various commands and see what each one does directly.

When the screen has the ‘>>>’ symbol, it is ready to accept commands from us. On hitting enter, it runs the command. Here it can display different things (such as LED 1, LED 2 etc from the previous example) or have no display at all, depending on the command.

Here are some tasks to try on the LED strip

Task 1: Turn off all LEDs

Type the following and press enter

```
led.fill(Off)
```

Note: O is capitalized in Off.

```
led.update()
```

Note: The LEDs turn off only after you enter led.update(). This is because, it is the update command, that actually sends the signal to LED lights. The advantage of this system is that, the program can gather ALL the changes we do to the LEDs and update the lights in a single step, rather than sending a signal for every small change. We can see the benefit of this when we write more complex programs.

Task 2: Turn on all LEDs

Type the following and press enter

```
led.fill(On)
```

Note: O is capitalized in On.

```
led.update()
```

Task 3: Pick a Color

Here is a list of colors we can use with the LEDs.

Note: You can generate many more colors than the ones on this list, and we will learn how to do that later. But here is a handy list to begin with.

Blue	
Pink	
Purple	
Fuchsia	
Crimson	
White	
Brown	
Yellow	
Orange	
Red	
Gray	
Green	
Violet	
Beige	
Black	
Turquoise	
Indigo	
Lime	
Khaki	
Maroon	
Gold	

Pick a color of your choice and enter it as follows. Say, we pick ‘Green’

```
led.fill(Green)
```

Note: When you type the name of a color, make sure that you spell it EXACTLY the same way. All color names start with a Capital letter.

```
led.update()
```

Task 4: Pick one LED

Now, we will learn how to turn on only one of the many LEDs.

To start off let's turn off everything else.

```
led.fill(Off)
```

Note: Nothing happens. But that's ok, we haven't run the update command yet!

Let's try to turn on just the first LED.

```
led.set(1, On)
```

```
led.update()
```

Did the first light turn on or the second??

IMPORTANT NOTE:
Computer counting starts at 0!!!!

*So we have,
First LED is LED 0
Second is LED 1
Third is LED 2
....
Tenth is LED 9*

So, let's start over!

Turn off everything else.

```
led.fill(Off)
```

Turn on just the first LED and update.

```
led.set(0, On)
```

```
led.update()
```

Task 5: Pick LED and Color

We can now do pretty versatile things like setting different LEDs to different colors. Here is one example: this will set the second LED (LED 1) to Green, fifth LED (LED 4) to Blue and ninth LED (LED 8) to orange.

```
led.set(1, Green)
```

```
led.set(4, Blue)
```

```
led.set(8, Orange)
```

```
led.update()
```

Task 6: Pick a range of LEDs

Instead of choosing one LED at a time, we can also choose to set a few consecutive LEDs the same color. The following example will set all LEDs from the 1st (LED 0) to 5th (LED 4) to blue and everything else red.

```
led.set(0, 4, Blue)
```

```
led.set(5, 9, Red)
```

```
led.update()
```

4. Exercise B

Exercise B

1) What is different from the way we count and the way a computer counts?

Answer:

2) What does the led.update() command do?

Answer:

3) Here are some challenges for you to try on the LED strip. Use the coding cheat-sheet provided below.

Can you...

- a) turn all the LEDs off?
- b) turn all the LEDs into Red color?
- c) turn off all LEDs except the 10th one?
- d) turn the first 3 blue, next 3 red and the last 4 green?

5. Programming Cheat Sheet

- 1) To turn on all LEDs

```
led.fill(On)
```

Note: O in On is capitalized.

- 2) To turn off all LEDs

```
led.fill(Off)
```

Note: O in Off is capitalized.

- 3) To turn on ‘n’th LED

```
led.set(n-1, On)
```

Note: Computer counting starts at 0.

Note 2: We can set the ‘n’th LED to any color from the list, by replacing On by the color name.

- 4) To set a range of LEDs from n to m.

Note: n has to be smaller than m.

```
led.set(n-1,m-1, On)
```

Similarly, the range of LEDs can be any color. This can be set replacing On by the color name.

- 5) To update the LED.

Note: None of the other commands will show their effect on the LED strip till this command is executed.

```
led.update()
```

6. Additional Examples

Here are some example programs written by the high school students in our summer program and pilot program.

- 1) Use the Python Shell to open these files. (File → Open : and select the files)
- 2) On the files, click Run → Run Module to see what they do!

*Note: Programs marked with a * run continuously. To stop them, press cntl + c together.*

Program name	What it does	CS Concepts learned
EvenOrOdd.py	<i>Asks the user for a number. If it is even, all lights turn Green. If it is odd, all lights turn Blue</i>	- Getting input from user - If-else logic statements
*Chasing1LEDLoop.py	<i>Lights up one LED after another in a continuous loop.</i>	- Using 'while' loop
*ChasingRainbowLoop.py	<i>Lights up many LEDs at a time in Rainbow colors in a continuous loop.</i>	- Using 'for' loop - Using nested loops
*DimToBrightLoop.py	<i>Changes the intensity of the light color in a continuous way</i>	- Learning to generate more colors and intensities
*Twitter-1.py	<i>This is an example written by the teaching team flashes every time a user-defined hashtag (#) gets tweeted. The students learned to modify the base function to add more functionality.</i>	- Understanding complex programs written by others. - Incorporating others' programs into the students' work.