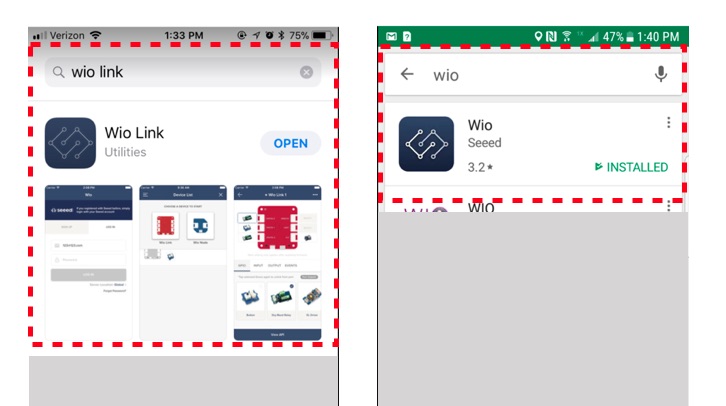
1. Preparation
   1. Get board and sensors

Host instructor will hand you a kit containing all the parts, check the list of contents. The Wio Node device box should have a "device name" on it, for example, dev-803. (Note: in the video, we used serial number D2D117, we changed it to dev-[Seq#] format, so it is easier to find device).

* 1. Download Wio App

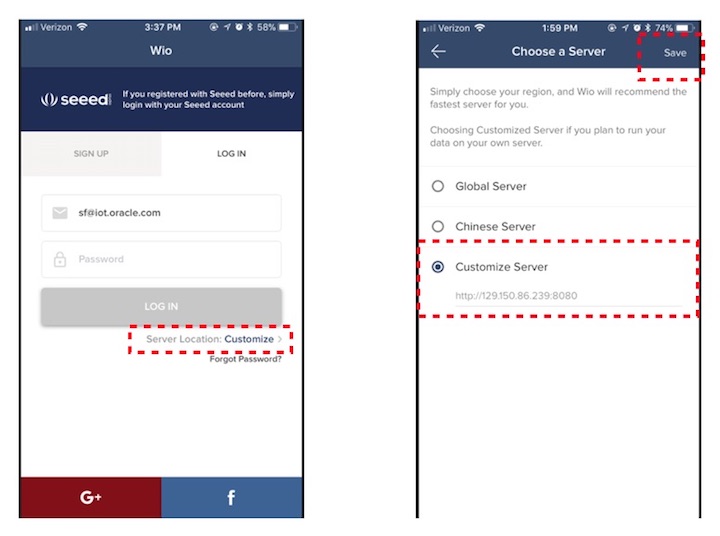
We use a Wio App to set up sensors on devices. You can download the app for iPhone, in Apple Store, search for "wio link". And for Android phone, in Google Play store, search for "wio".



* 1. Configure Wio App

We host a Wio virtualization server on Oracle Cloud, so you need point your Wio app to the server.

customize server: <http://129.150.86.239:8080>



* 1. Login to Wio App

We will use a common event account, where we register all Wio Node boards there. Host instructor will let you know the account and password to login.



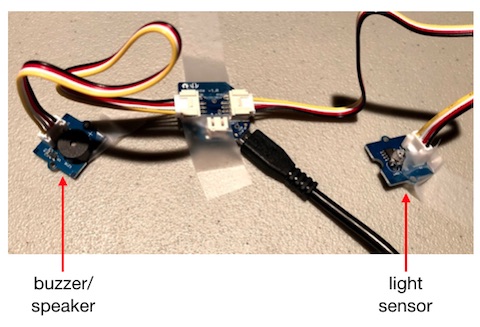
On your Wio Node device box, it should have a "device name" with sequence number, such as dev-803, and try to find your device with that name from the list in the Wio App. For example, if you have a box with dev-3103, then the first device on the list is yours.

1. Connect light sensor and buzzer
   1. Power Wio Node using micro-USB cable

After a while, refresh the device list in Wio App, and you should see your device in Wio App to show "online" state.

* 1. Connect sensors to board

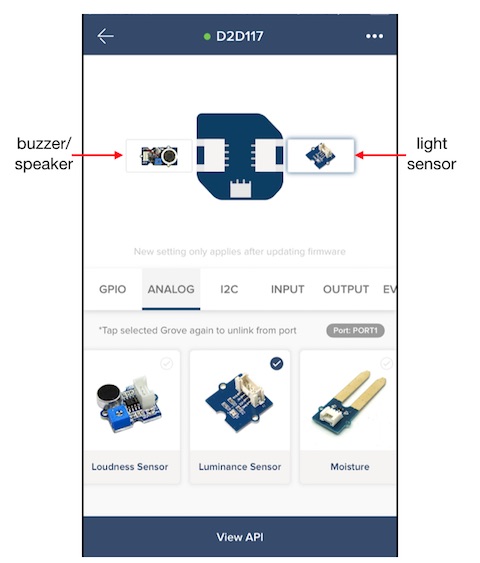
Please connect Light sensor and buzzer to Wio board, as shown in this diagram:



Note that we connect Light Sensor (aka Luminance Sensor) to right-hand size of port which is an analog port, and Buzzer (aka Speaker) to left-hand size digital port.

* 1. Connect sensor in Wio App

In the Wio App, we need to connect Light Sensor (aka Luminance Sensor) and Buzzer (aka Speaker) to reflect exactly how you connected them physically at step 2.2.



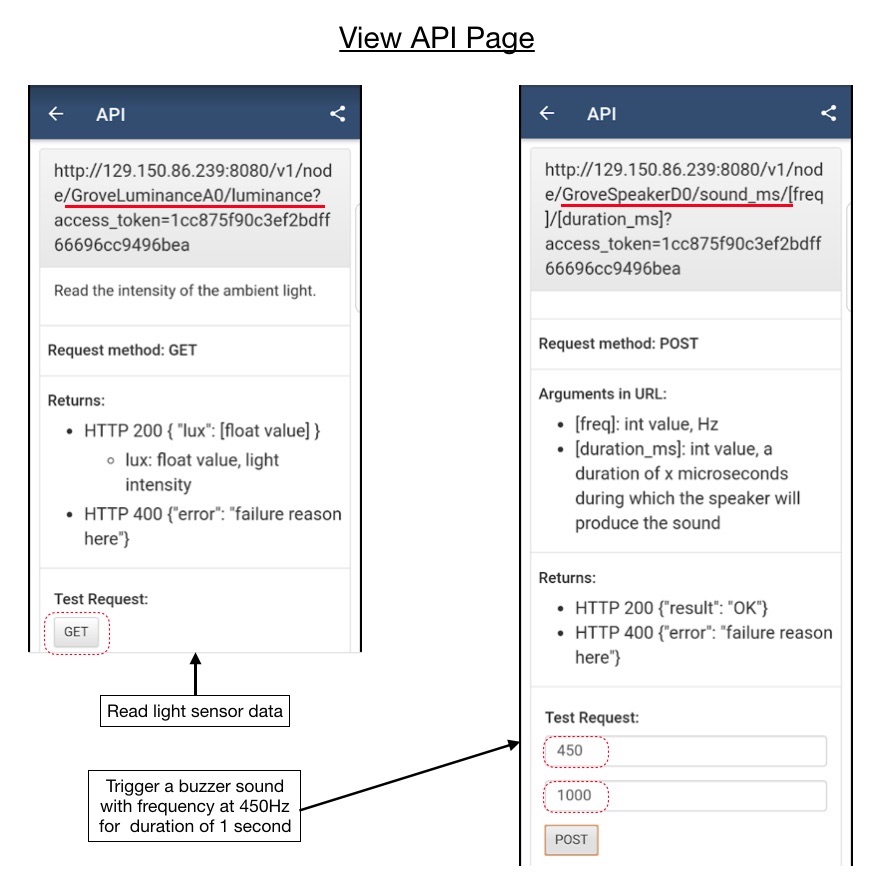
This tells Wio server how the device and sensors are connected, so that Wio server can produce corresponding firmware for your Wio Node.

* 1. Update firmware

Whenever you change the sensors in Wio App at step 2.3, it will show the button Update Firmware in red color. You need to click that button, to trigger Wio server to update device firmware over the air.

* 1. Go to API page

Using either View API button or menu item to get to API page. Look at the API page details, and you will find various APIs to interact light sensor and buzzer. You can interact with sensors on the board, by using the API calls.



* 1. Read light sensor data

Refer the picture diagram at Step 2.5 above, look for luminance API in your Wio App View API page and perform a read on light sensor by clicking "GET" button.

(We found that this didn’t work on the new update to the app on iOS 12, so we took the API and tested it in Postman or something similar – try the same to ensure the sensors are working)

* 1. Play a sound on buzzer

Refer the picture diagram at Step 2.5 above, look for sound\_ms API, and produce a buzzer sound by giving sound frequency and duration and clicking "POST" button. (We found that this didn’t work on the new update to the app on iOS 12, so we took the API and tested it in Postman or something similar – try the same to ensure the sensors are working)

* 1. Summary of what you just did for this exercise

You just used Wio App and its API page to interact with sensors and buzzer on the Wio Node board. So far, no programming needed. In Exercise #3 below, we will write Java program invoking those API through library calls.

1. Complete the Java program to read light sensor and make buzzer sound
   1. From USB – take relevant collateral
      1. JDeveloper
         1. Linux
         2. Windows
      2. JDK 1.8
         1. Linux
         2. Windows
         3. Mac
      3. OracleWioIoT
         1. All contents

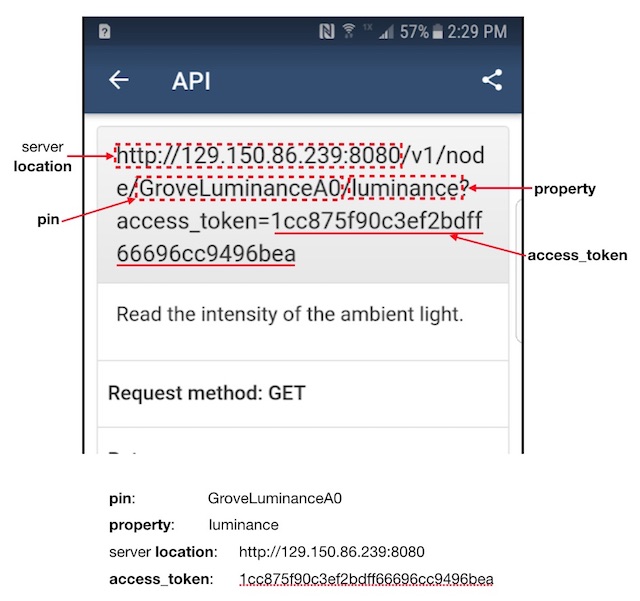
This contains the source code for the whole project.

1. We have stripped out some of the contents of the java leaving you with a skeleton so this is where we want you to complete some tasks to ensure communication between the devices and our IoTCS.
   1. WioLuminanceMonitorSkeleton.java
      1. You need to get the current luminance value, test it and sound the buzzer if it's too low. See the WioConnectivityTest for examples of how to do this
   2. WioLuminanceToLEDSkeleton.java
      1. Setup the WioStream, start it and use a Lambda to call the monitor.onWioEvent method when a "onWioEvent" is received. If you want call the logger to report the progress
      2. You need to get the current luminance value, and send it to the LED Bar. Remember to replace the buzzer on the Wio Node with the LED bas \*AND\* update the firmware! Note - the luminance value is a float, but the LED bad takes an int (0 - 1023) You may want to have a delay if the updates are to frequent
   3. WioOracleIoTIntegrationSkeleton.java
      1. Register for the Led Bar command that calls the processLedBarCmd command. A lambda is easiest, but you could use an anonymous inner class if you liked. (WioLabConstants has a constant of the appropriate name to use)
   4. Buzzer.java
      1. Implement methods
   5. Humidity.java
      1. Implement methods
   6. LedBar.java
      1. Implement methods
   7. Luminance.java
      1. Implement methods
   8. Temperature.java
      1. Implement methods
2. Check configuration
   1. You need to have an access\_token for running program to access the board and connected sensors. Go to folder ‘OracleWioIoT-tgfordublin’, open wioConfig.json file.

"location" : "https://us.wio.seeed.io/",

"access\_token" : "19aac46bae65216dc63aff0c01f65da9"

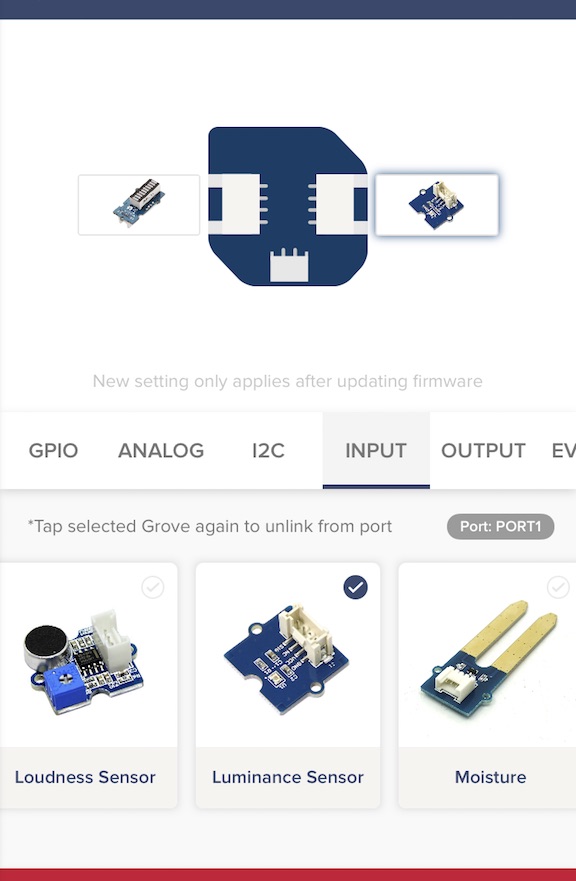
Please verify location is set to proper Wio server, and change access\_token to match with the token in your Wio App. Refer the API page in your Wio App, for example:



1. Device communicating with Oracle IoTCS
   1. Config file?
   2. Security provision?
   3. Run IoTCS client program?
   4. Check sensor reading in IoTCS?
2. Try other Sensors
   1. By now, you have completed all exercises for the workshop session. If you like, you can continue to try the Temperature & Humidity sensor and LED bar. Just follow with Exercise #2, replacing Light sensor with Temperature & Humidity sensor, replacing buzzer with LED bar.

Here are some examples of connecting various sensors to the Wio Board, for your reference:

Wio-node with light sensor and LED bar



Wio-node with temperature sensor and a buzzer



1. Go home and try more
   1. When you get home, the Wio Node device no longer has access to the WiFi network you used at IoT Workshop, so you need to follow this setup instruction - Getting Started, to set your home WiFi network SSID and password to Wio Node device, so it can connect to Internet. Also you can use the Wio Global server, instead of the Customize server used in the IoT Workshop. You need sign up an account on the Global server, add your device to your account. Please just follow Getting Started guide, from step 1 to step 5. You can replace the Grove LED used in the guide, with LED bar we supplied.