# INTEL EDISON

1. Following getting started guides such as Get Started with the Intel® Edison Board or Edison Getting Started Guide, set up developing environment for Intel Edison (installing driver, IDE, etc.). Note that, there is probably no need to flash Edison with the latest firmware. The currently used firmware version is Release 2.1 Yocto* complete image.
2. As far as IDE is considered, Microsoft Visual Studio 2015 and Visual Micro (an Arduino plugin for Visual Studio) are preferred than the original Arduino IDE.
3. Set up Bluetooth SPP on the Edison side, by following this article, COMMUNICATE TO ARDUINO CODE WITH YOUR ANDROID* PHONE BY BLUETOOTH SERIAL PORT PROFILE (SPP) ON INTEL® EDISON.
4. This version of image does not run Arduino program on boot. To solve this, follow the instructions in the post, Run Arduino sketch on boot Workaround. Please read and find useful information and files in this post.

# ARDUINO PROGRAM

1. Download codes from GitHub repository SentireArduino (https://github.com/songer1993/SentireArduino ), the file structure is shown below. Note that those normal and unmodified library stuffs are not here, it is assumed that you will have them if you do the above steps correctly.
   - Main program
     - SentireArduino.ino (main program file)
     - myutils (contains two functions handling String variables)
     - Pattern (a normal class file for Pattern objects)
   - Intel's own Arduino library (small changes added to the original library files (version 1.6.2 + 1.0). Overwrite those folders in the firmware library directores, which are like "C:\Users\songe\AppData\Local\Arduino15\packages\Intel\hardware\i686\1.6.2+1.0\libraries\" and "…\1.6.2+1.0\cores\arduino":
     - SD (in libraries folder)
     - Cores (WString.h and WString.cpp) (in cores\arduino folder)

- Third Party Arduino Libraries, usually installed in the directory "C:\Users\songe\Documents\Arduino\libraries\". Copy following library folders to this address:
    - Intel-Edison-BT-SPP-Library
    - DRV2605L
    - OriginalSSD1331
    - ArduinoJson-master
    - ADXL345

2. Program the device to see if any error happens.
3. Understand the main program by reading the thesis and comments in the program. Note Azure Mobile Service url and application key constants are hard-coded as constants in SentireArduino.ino.

## ANDROID APP

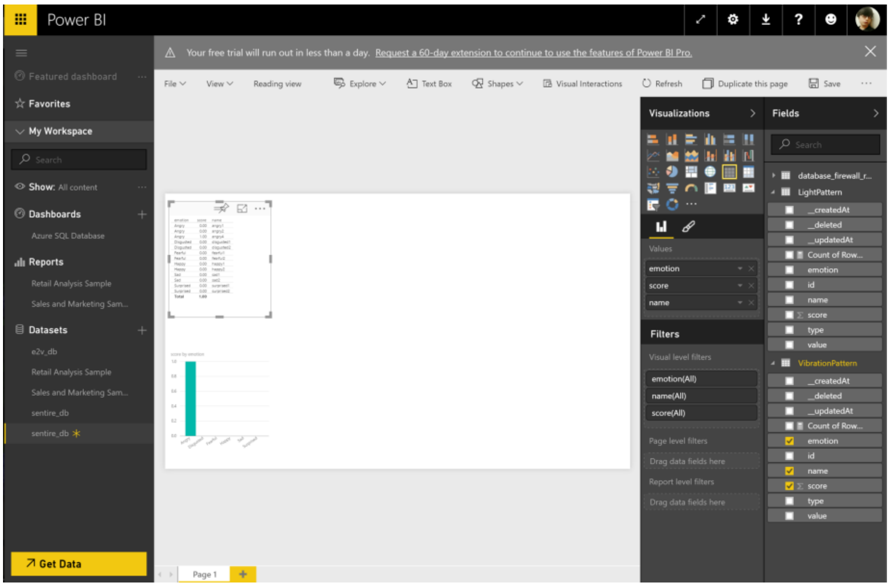
1. Download the code from GitHub repository Sentire (https://github.com/songer1993/Sentire), open or import it in Android Studio 2.1 (latest version). Detailed installation guide please read the official websites.
2. Program codes are stored in the "\Sentire\app\src\main\java\com\project\songer1993\sentire" folder. Several java classes are there, but some of them are just supporting others. The main files are shown below:
    - ConnectBT (starting page)
    - MainActivity (dashboard page)
        - DesignLibraryVibrationPatterns
        - DesignRealtimeVibrationPatterns
        - DesignLightPatterns
        - SeeSavedPatterns
        - PlayDemo
        - SetupWifi
    - Classes
        - Pattern, Vibration Pattern, Light Pattern
        - Constants (storing Azure Mobile Service url and application key constants)

3. Other resources (images, colours, layouts) are stored in the "res" folder. Layout files (.xml files) in the "res\layout" are also important, since they define the layouts of the app, i.e., the UIs.

## CLOUD

Unfortunately, Microsoft is trying to shut down Azure Mobile Services shown in the thesis. It is expected that those already created services can still be used until September 2016. However, no new same mobile services can be created now. This means, you can use the service (tables) I have created for the following two months, but I cannot give out my azure account so you cannot manipulate those tables directly (such as deleting records or tables). It is still possible (but a bit inconvenient) for you to view pattern data on a PC via a data visualisation software called Microsoft Power BI. Please follow steps below after signing into Power BI (use your Uni mailbox account):

1. Select "Get Data" and "Microsoft Azure SQL Database"
2. Enter the following:
   - Server: o4b4aflpnw.database.windows.net
   - Database: sentire_db
   - ID: sentire
   - Password: SensoryAssistiveDevice2016
3. Then you are able to view data as you want:



Your free trial will run out in less than a day. Request a 60-day extension to continue to use the features of Power BI Pro.

What actually is happening is that Microsoft is trying to merge several cloud services including Mobile Services to a new service called App Services. It is still possible to build connection between the android app and the cloud but I haven't been successful in connecting the device with the cloud.

So to solve the problem fundamentally, we can:

1. Figure out how to connect the device to the cloud using the new App Service. Not sure about this, maybe it is just impossible, but should try and could seek help online, for example, Intel Edison community forum.
2. Or give Microsoft Azure up, go for another cloud service provider or build your own remote server (which requires extra hardware). This may incur unexpectedly large amount of work.

On the other hand, a workaround solution can be:

1. Instead of updating pattern data directly from the cloud, we can use the mobile app to update pattern data stored in the device via BT. This requires adding more codes to both app and device's program, but certainly doable.
2. Or just keep using the existing Azure Mobile Service for a short period to collect some data.

## OTHER ADVICE

It is usual to face various kinds of problems when programming with Intel Edison. Don't panic, try to google first or search similar threads in its community.

If you have any questions, please contact me:

Email: qisongwang@outlook.com / songer1993@gmail.com

Phone: +4474059322 / +8618815275242

Skype / Wechat: qisong.wang / 472153761