Summer 2019 Engineering Notebook

Jacob Smith

[Jsmith2021@brandeis.edu](mailto:Jsmith2021@brandeis.edu)

This engineering notebook contains my work during the summer of 2019 in the Automation Lab at Brandeis University. Project contained include

Arduino Class Generator:

A tool integrated to the Arduino IDE to create class body, header, keywords, and example sketch files

[https://github.com/jsmith2021Brandeis/Arduino](https://github.com/jsmith2021Brandeis/Arduino-Class-Maker)

[Class-Maker](https://github.com/jsmith2021Brandeis/Arduino-Class-Maker)

Arduino Education:

A set of workshops, tutorials, and background programs to get students started with Arduino

[https://github.com/jsmith2021Brandeis/ArduinoEd](https://github.com/jsmith2021Brandeis/ArduinoEducation)

[cation](https://github.com/jsmith2021Brandeis/ArduinoEducation)

(includes my helping two high school students refine their project [https://github.com/AidenKunkler](https://github.com/AidenKunkler-Peck/Tactile-Necklace)

[Peck/Tactile-Necklace](https://github.com/AidenKunkler-Peck/Tactile-Necklace))

Rasberry Pi Digital Window Project

Programs and setup for a series of digital windows that will allow the makerlab to communicate internally with itself

Table of Contents

|  |  |  |
| --- | --- | --- |
| Page # | Project | Date |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Jacob Smith 5/16/2019 5:06 AM: I looked up how much information there is on Object oriented programming In Arduino, and <https://forum.arduino.cc/index.php?topic=6127.0>

Post 3 says that writing libraries is difficult, which helps to justify my automatic library creator

Jacob Smith 6:53 PM: I wrote a series of Java classes to allow for the user friendly creation of new Arduino Libraries. The ScriptEditor class allows for the simple reading and writing of files, the Arduino Class allows for the parsing and editing of Arduino Classes, and the GUI class creates a simple user interface.

**Graphical User Interface of Library Creator🡪**

I made these libraries because I found myself repeating a series of tasks every time I made a library like replacing the name and changing the header comment. The user interface accomplishes both of those actions by reading user input in textboxes and using it to modify a simply example class.

**Template Class for Arduino Code generation🡪**

While this isn’t that helpful on its own, these classes show the possibilities for making it easier to create arduino classes, in the future the user could add a global variable and have both the header and cpp files automatically updated, currently only the cpp file is changed.

Jacob Smith 5/13/2019 5:19 AM: I am rewriting the Java ArduinoClass to build a cpp file from a template, and I am researching arduino libraries some more. In addition to my current scheme, auomatic conversion of an arduino sketch (.ino) to an arduino library (.cpp) would be very useful

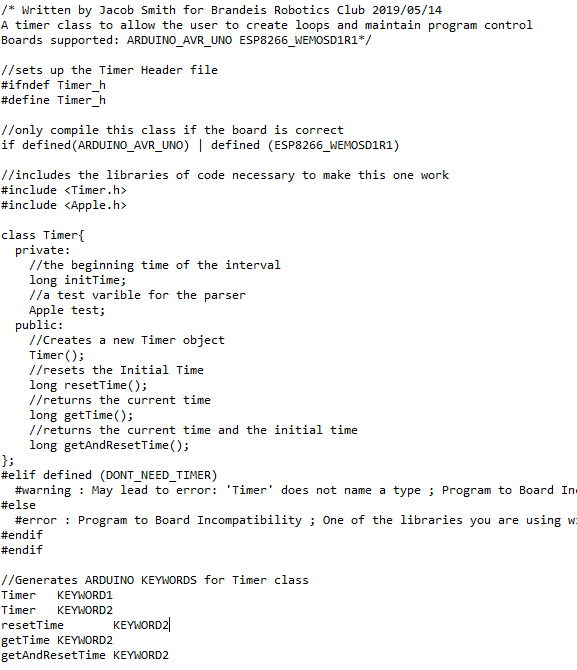
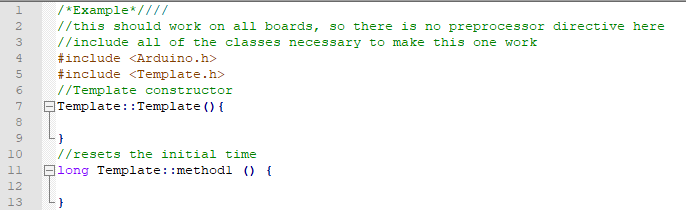
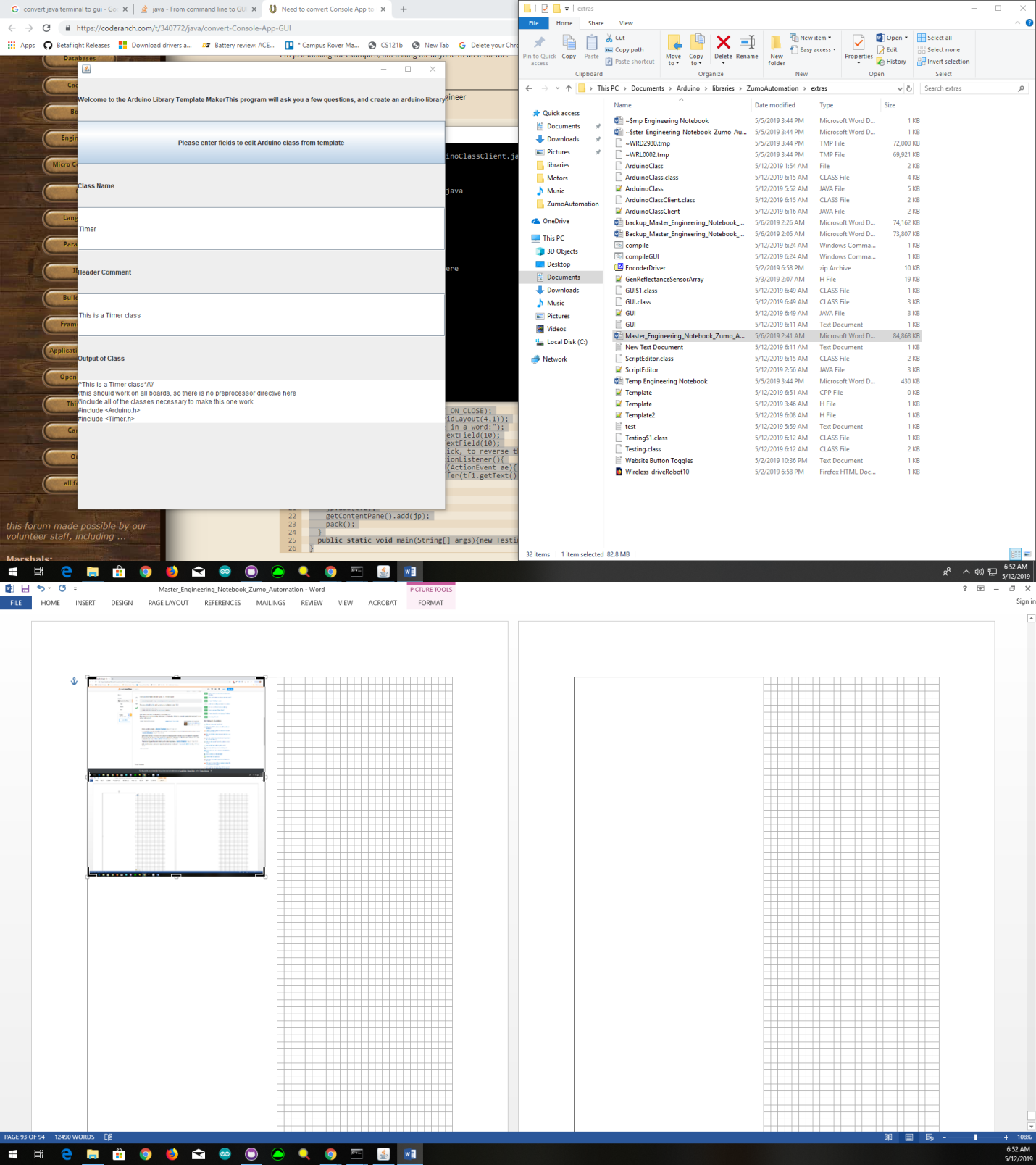
[1] <https://arduino.stackexchange.com/questions/32998/how-to-convert-arduino-example-sketch-to-a-complete-c-project>

[2] <https://community.platformio.org/t/tutorial-for-creating-multi-cpp-file-arduino-project/5830/13>

5/14/2019 Jacob Smith: I wrote an inheritance hierarchy of common methods in ArduinoClassMaster, body methods in ArduinoClassCpp, and header methods in ArduinoClassH. Left is Autogenerated header and keywords file from ArduinoClassH

**Automatically Generated Header file🡪**

Jaocb Smith5/15/2019



Jacob Smith 5/15/2019 1:23 AM: I restructured the github folders, here are some toggle button schemes that Deveroah and I thought of a while ago: **Wireless Button Driving Schemes🡪**

I have been working on the user interface for Arudino class generator, and I am fidnign hat putting the required varibales names, prompts, and example formatting in an enum makes my client code easy to read and write. 🡪

**Motion-Stop**

Dive F -stoppped

Drive B -stopped

Turn L-stepped

Turn R=stopped

**Direction-Turn**

Drive F-DriveB

Turn L-Turn R

Move- Stop

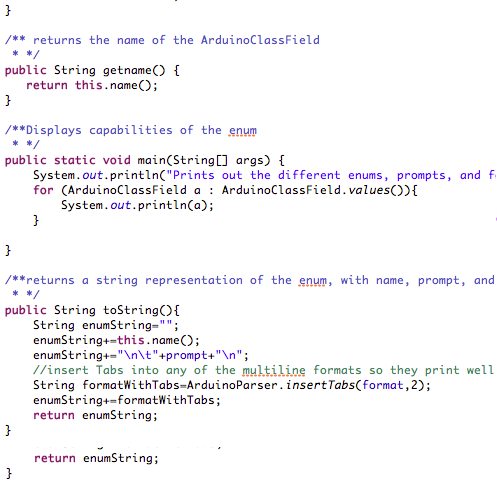
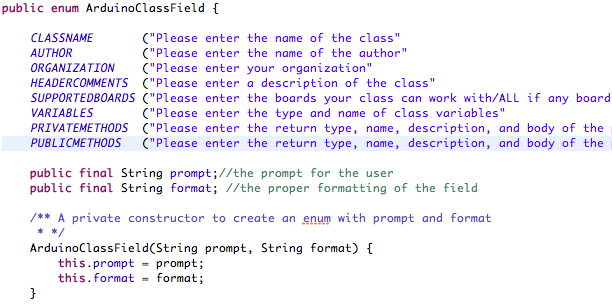
**Pin Directed**

Drive--Stop

Forward--Backward

Turn Right--Stop

Turn Left--Stop



**ArduinoFields Enum for Code Readability and to allow all field related prompts to happen In one place**

My Journal Notes on the Project

Unknown Date:

void|driveLeft|tells robot to Drive Left robot.driveLeft()

void|driveRight|tells robot to Drive Right robot.driveRight()

had to reset project tested prompt method in prompt enum, going to display DONE prompt more cleanly and also, I have to break up internally parsed prompts better I am using recursion to handle the complex formatting🡪

is on the innermost level tokens of |, then tokens of newlines, then tokens of double newlines while this format is complicated, it is both computer and human readable

**Example of Method Formatting**

I am working on a field by field prompter, handling initial conditions before loop does simplify code

Sunday: I am only allowing allowed characters, to hide which characters I am using for parsing and also exclude strange characters I haven't thought of

I am replacing all instances of scanner read next line with my validating looping method

It seems like using an enum for prompting is too complicated, I will just use methods

I am handling special cases manually, simplifies my methods and makes code more readable

I am using interface to save code enums can implement interfaces

Manually typing in user responses for testing is tedious, so with print streams and enums, I can now just run a test instead of manually typing in test input for ArduinoClassPrompts.

I am now at the point where I can prompt field by field and try to generate the Arduino class, but I found methods are missing | character set breakpoints as you work with debugger, maybe directly from runtime exception found a parsing error in variables, explains why only header file had it, I’ve got an extra newline in variables I fixed that error which was missing |, but methods had extra newline, I removed it with substrings Now test runs correctly, but public methods aren't showing up TO DO: not do substring by flipping with fencepost solution, fix public methods, unit test error corrections public methods was newline issue,

Now that I added constructor, the output essentially looks like an Arduino class. I switched order of private and public methods in body file, which puts constructor at top and puts most visible methods at top too line by line unit test passed.  
​  
Created Arduino class maker GitHub repository so I can actually delete unused code and know it is backed up

May 23 I am working on the program that converts an existing Arduino sketch to a library, which required a lot of background code work. I modified the MiniScanner class to display an error with the word that was looked for, and the Arduino Class generators to allow method parameters of null if blank. The methodParser class handles the work of converting a method into the format the code generator uses, and I want the program to automatically generate an example sketch. The sketch parser should also be able to tell which methods are public vs private by looking for which methods are in the setup and loop methods The parsedMethod class has fields even though I could have the whole thing be one big toString. The SketchParser class currently can read a sketch into the code representing the header, variables, public and private methods, loop and setup, but they aren't formatted into the format the Arduino class generator needs

Now that the SketchParser method can convert my sketch into a fairly well formatted list of header comment, variables and methods,the next step is separating private and public methods and parsing variables. The variables parsing will become fairly interesting/complicated, because they should all go to a constructor. This means that the sketch to library program is forcing me to add some features like adding constructor and adding method parameters.

The method correctly decided that wifiConnect and run server where public methods I am using replaceAll String method a lot, very useful in reformatting sketch. I am writing ParseVariable method with test cases to convert code of variable declaration to parsable format. This is complicated because data types and values can be more than one word, and arrays are in form type name [], and variables don't have to be initialized .

5/24 10:51 PM: Working on Arduino parser and unit tests, I got test to pass when variable is declared not initialized, still failing Linked List switch nodes (I wrote linked list to iterate over variables which can have multiple words like const char) and eliminating extra vertical bars. passed extra vertical bars test using String. It seems that using String.replace works when String.replaceAll does not, I don’t know why.

5/25 10:48: FUTURE WORK: I want to integrate this project into the wider Arduino Environment by reformatting and compiling files before they are parsed. This would allow me to standardize text before I have to parse it, and return an error in advance if program doesn't compile. Also, I want to eventually make a pull request to have the automatic class generator be part of the Arduino ide. Relevant links:

[1] Arduino Command Line Interface <https://github.com/arduino/Arduino/blob/master/build/shared/manpage.adoc>

[2] Jar files already part of Arduino <https://github.com/arduino/Arduino/tree/master/arduino-core/lib> [3] How to create a jar file in eclipse <http://www.skylit.com/javamethods/faqs/createjar.html>

May 27, 2019 10:34 AM: I looked up who else is working on an Arduino Class Generator on the Arduino developers’ group, no one there is. I'm working on passing more of the VariableParser unit tests, so I wrote a MiniScanner getRest method to help.

CODE STYLE: Else statements are useful in avoiding something right before exiting a loop. In that scheme, use Boolean loop control variable as condition. Set variable in loop, and use else statement to do whatever should be done while looping is chugging, but not when it is about to exit. This is a more subtle version of a break statement.

12:58 PM: Working on LinkedList switch method for variable parser, I wrote a state method that handles null pointer exceptions, and made setter methods private to force its use

1:13 PM: Arduino IDE Auto format inserts spaces but doesn't remove them, I'll have to do that on my own, which should be using the MiniScanner class internally.

3:07 PM May 28, 2019: I got the MiniScanner to allow it to ignore multiple tokens in input, but now a lot of other unit tests fail

6:58 PM: I spent a lot of time today trying to get rid of scanner errors, and put the project on GitHub so I wouldn't have to redo my work again

11:00 PM I got all unit tests to pass, and having project on GitHub with commits on passing tests makes it much easier to keep project moving forward with unit tests passing, and creating branches is useful. In the SketchParser class, I am working on an autogenerated constructor based on variables. Before that, I am preparing the program to handle the <https://www.arduino.cc/en/Hacking/LibraryTutorial> Morse code sketch as is so I can post it to the Arduino developers forum <https://groups.google.com/a/arduino.cc/forum/#!forum/developers>. The challenges so far with that have been to reformat the newline brackets style of coding into the same line and to create to do comments when comments are missing, which is what I'm working on now.

May 29, 2019 10:37 AM: last night, I got the SketchParser class to produce keyword, example file, body, and header for the ESPServer and Morse classes. The unreliable parts are automatic constructor generation and how the sketch file doesn't call based on object. In addition, I posted my project idea on the GitHub developers forum, and I got a response from someone to share my GitHub link, although I haven’t decided whether to keep my code private yet.

7:18 PM: I got the generator to change method calls in example sketches to be an object method, like converting dash(); to morse.dash(). Also, I got the GUI to be useful, now the user can click on

RUNme.jar, and this window pops up🡪

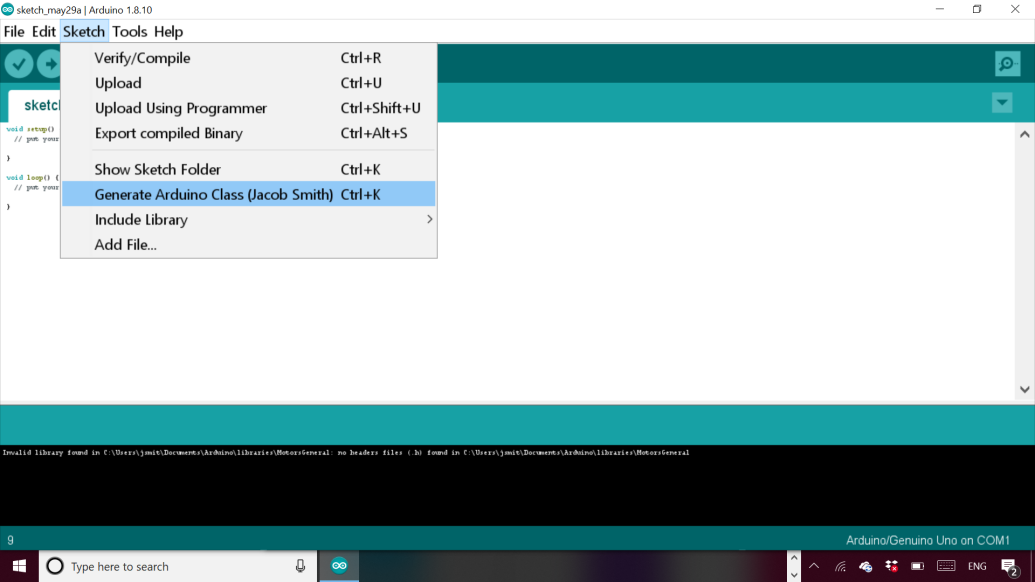
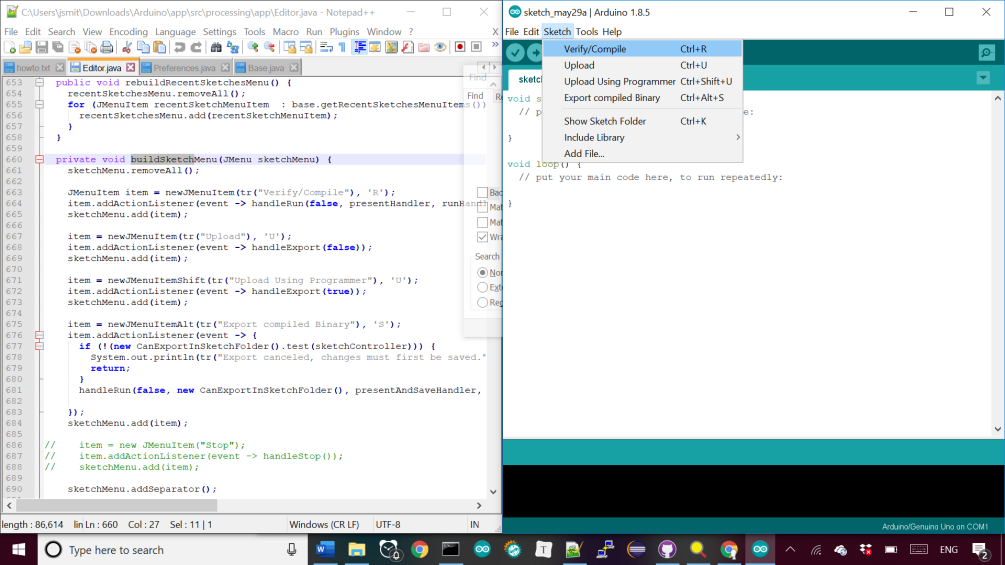
The user can enter the name of the class and location of the correct Arduino sketch.

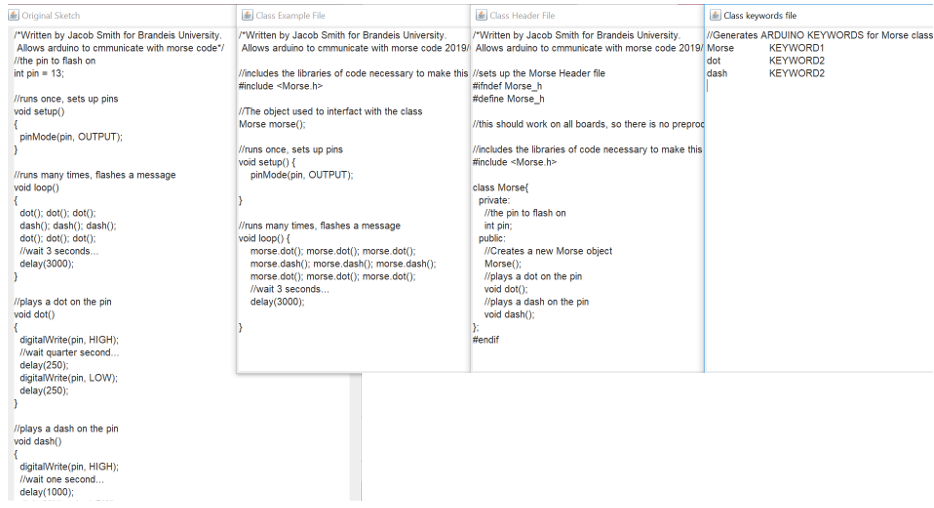
Then, when the button is pressed, the following windows pop up (next page).

Future work: Integrate this into the Arduino IDE: <https://github.com/arduino/Arduino/wiki/Building-Arduino>, research whether I should publish this program.



**User input interface, runMe.jar**





(TOP) Generated Example, Header, keywords, and original sketch. From runme.jar

(BOTTOM) Generated JDK error message because FJD must be 32 bit

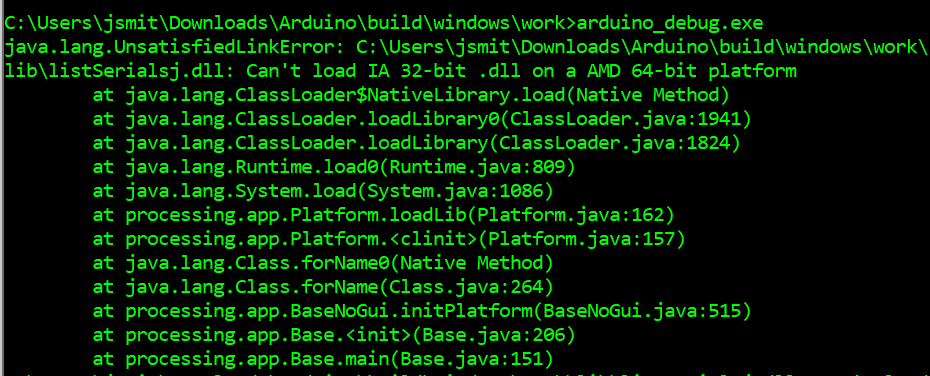
How to build Arduino IDE<https://www.mkyong.com/ant/how-to-install-apache-ant-on-windows/>

Why I need 32 bit JDK

<https://github.com/arduino/Arduino/issues/3276>

May 30 1:00 AM: I can now compile and run the Arduino application; everything is in the Editor class. I can now add a menu option called generate Library, and I’m trying to use the tab creation feature to generate all the correct files at once. Then, I need to add the relevant classes into the Arduino folders.

May 30 2:52 PM: I am trying to automatically create files in Arduino ide, relevant functions: Addfile Sketch.java line 311, SketchFile.java line 94, EditorTab.java createTextArea line 146, EditorTab.



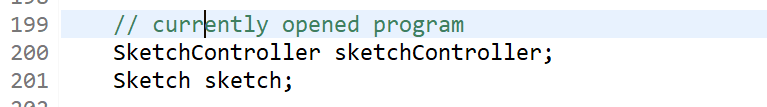
(Middle)🡪 The menu option to generate Arduino Class.

(Bottom)🡪 The area of the Editor class in the Arduino IDE that I will be modifying for the GUI.

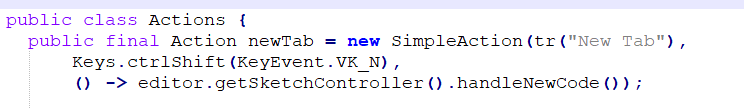
SketchControler.java 136, nameCode

Sketch Controller Line 598 import Library

Sketch Controller upload line 598 (some boards have a password)



**Editor.java line 199 fields, sketchController allows new tabs to be created**



**Editor Header.Java Line 88 , shows how new tabs are generated**

Relevant background:

<https://github.com/sudar/Arduino-Makefile> More control of Arduino Compiler

Arduino Tabs are just a way of breaking up a sketch <https://forum.arduino.cc/index.php?topic=206078.0>,

Nice tutorial for library creation <http://arduino.land/FAQ/content/7/43/en/breaking-a-sketch-into-multiple-files.html>

Using sketchContoller code avoids java exceptions and lets you create new tab.

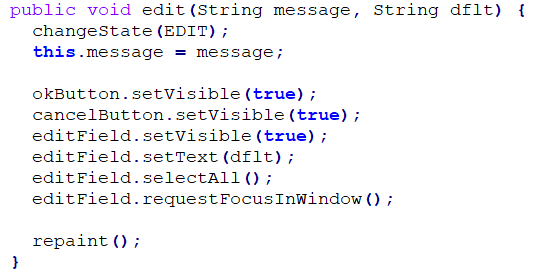
11:00 PM: I am writing a script to compile the Arduino ide, so I don’t have to repeat a manual task by going to a certain directory and deleting the dll file. The dll file gets a wired error access is denied, when the real error is that the file is already running

12:50 AM: I can now generate tabs for the .cpp,.h, keywords.txt, and example file.ino files. I am now getting the string of the file name and file contents to pass to my Arduino class generator.

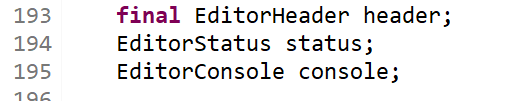
2:41 AM: I can now set the text of the files and save automatically, now I will abstract the code into a method that takes the four strings of the body, text, keywords, and example files, the only unknown inputs

Useful method:

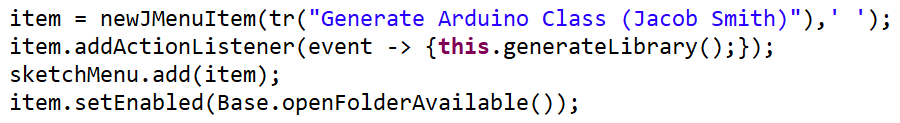
Build method sketch controller.java.



EditorStatus.java line 147, the method used to get the file name, which I want to set automatically



Editor.java line 193 status field which is used to display messages

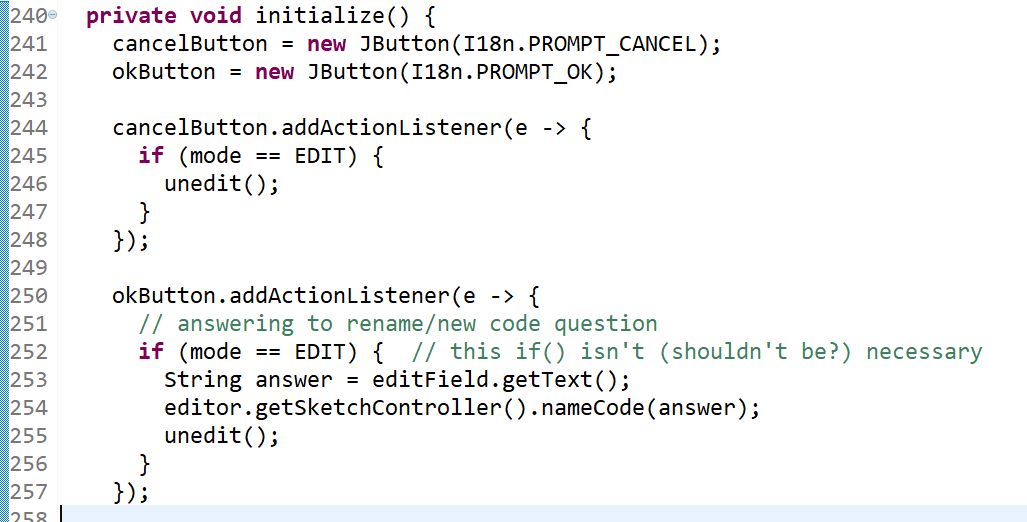


**Arduino IDE showing the new menu option, file name dialogue, and console printout🡪**

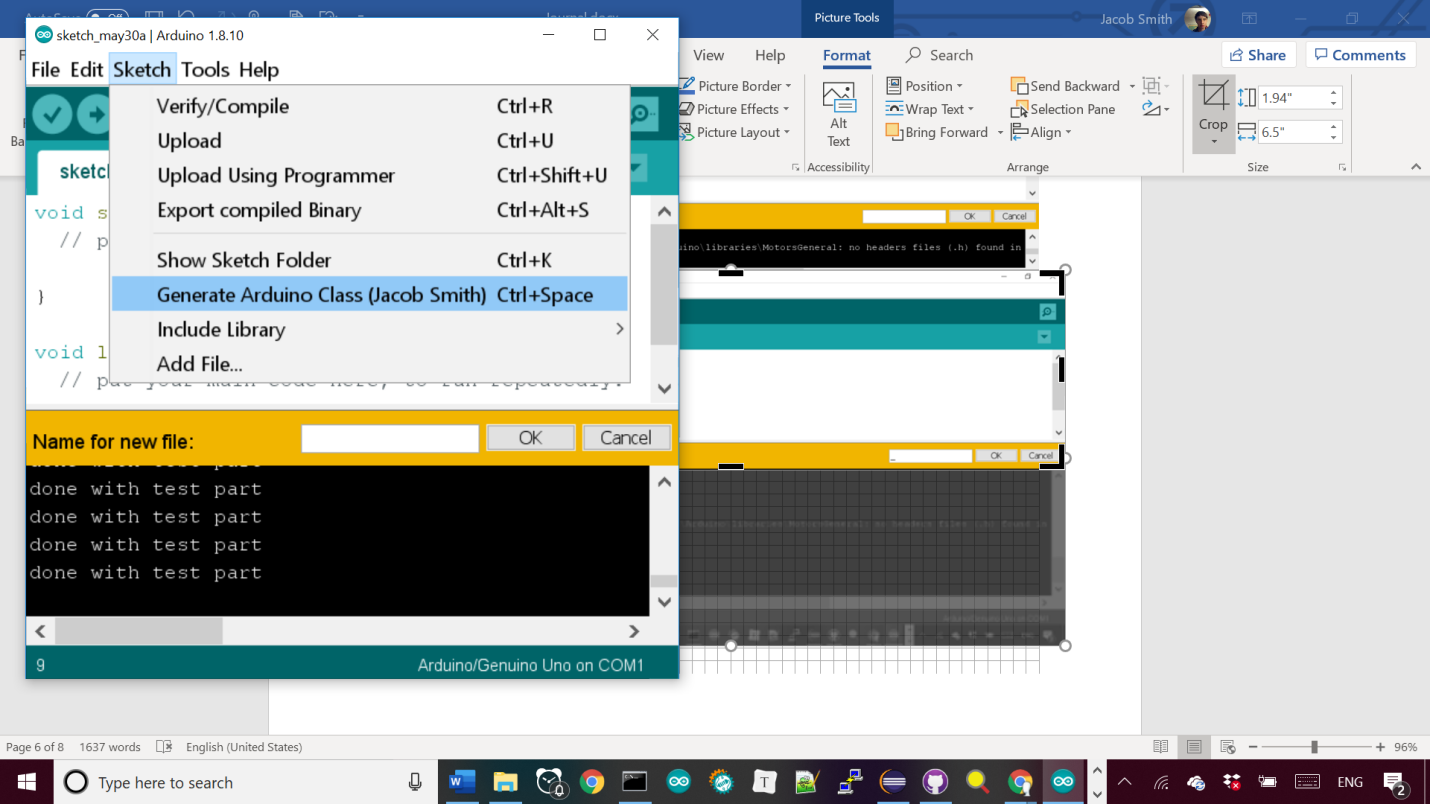
**🡨 Editor Java Line 697, showing how keyboard shortcut finally works**

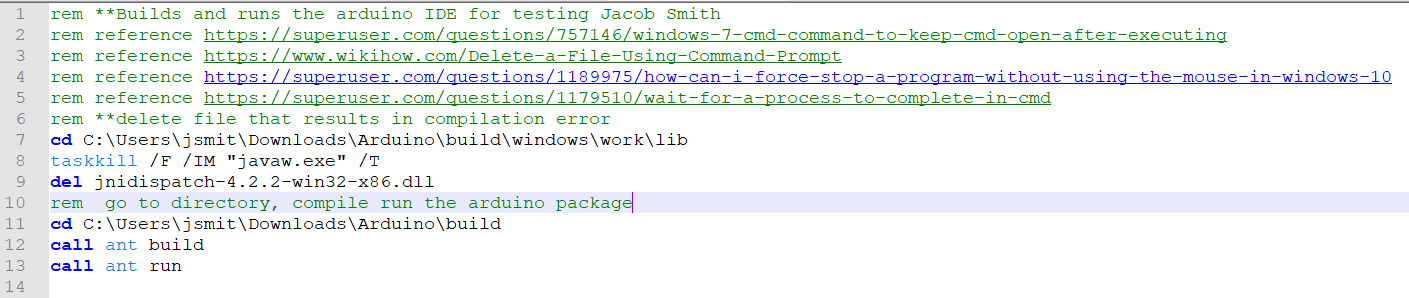
Clockwise from left. These pictures show: how the menu option is shown, how new tabs are created in example IDE code, the menu option being used to display the generate class option which now works with keyboard shortcut, and the batch file that compiles the Arduino application.

This stage of the project was about reverse engineering the Arduino ide to connect it to the class generator API. To do this, I used GREP WIN to search for where the text of a menu option occurred in code to see how it is used and what source functions are important. This allowed me to write my own functions to create tabs for the class information.



**↑Initialize Method of Editor Status.java line 240, shows how ok button is wired to create a new tab**





**RunArduino.bat file to compile and run the Arduino package automatically.**

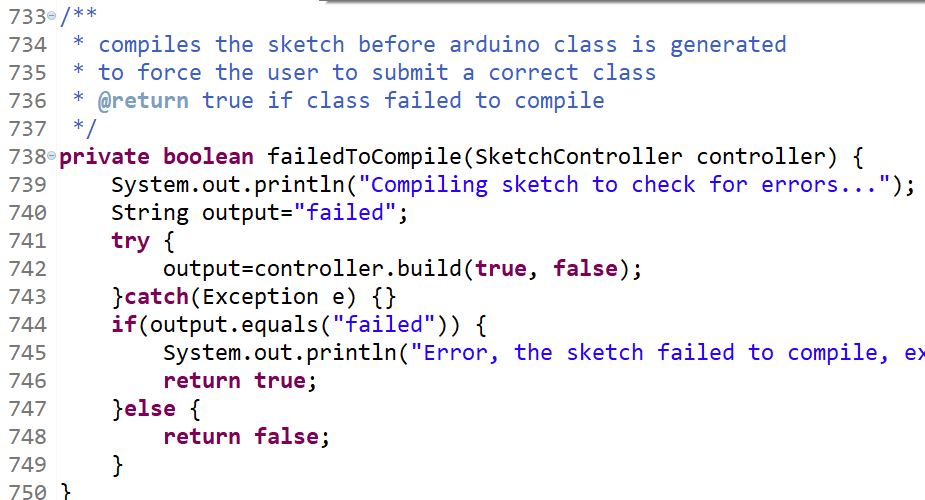
4:46 AM: I wrote this method🡪 to stop library generation if sketch doesn’t compile, it is based on the build method of Sketch Controller class which returns null if the build failed.

Currently, the messages aren’t displayed after the couple seconds it takes to compile the program, probably because I’m not calling a repaint method.

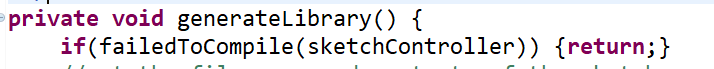
I am working on abstracting the inputs and outputs of the sketch generator interface in the same manner as this method, notice how it is one method that can be used easily and doesn’t rely on global variables.

I’m trying to do this for setting the texts of the tabs, which was complicated because they are automatically alphabetized, but I’m creating them in alphabetical order now.

CODE STYLE: I’m making methods static before I put them in their own class. This will prove that they don’t rely on instance variables, making them easy to put into their own class.



(TOP) My FailedToCompile method in Editor.java to stop generation of library if sketch doesn’t compile. (BOTTOM) usage of method in generate library client code to stop method if compilation fails with screenshots of ide in both use cases.



WORKFLOW: The Arduino IDE lets me input my data directly from the sketch contents and file name and compile the sketch beforehand. This means that I deleted all the classes that prompt the user for input and validates the response. I also separated tests and source files from each other, which will make it easier to put the source files into the Arduino ide, possibly with a jar file to encapsulate and hide them. Potential bug: Some fields can be read to the Arduino class generators as null or "null", I should allow both.

June 3 2019 10:15 PM: After trying out working environments, I settled on this structure**🡪,** where all of the classes to generate an Arduino class from a string are in the ArduinoClassGenerator package, and the interface is in the processing.app class.

The interface needs to be in its own class because it calls the nameCode method of the SketchController class to create tabs, which is a protected variable. **🡪**

The other members of the processing.app class are dummy classes so the referenced classes in the ClassGeneratorInterface don’t result in errors.

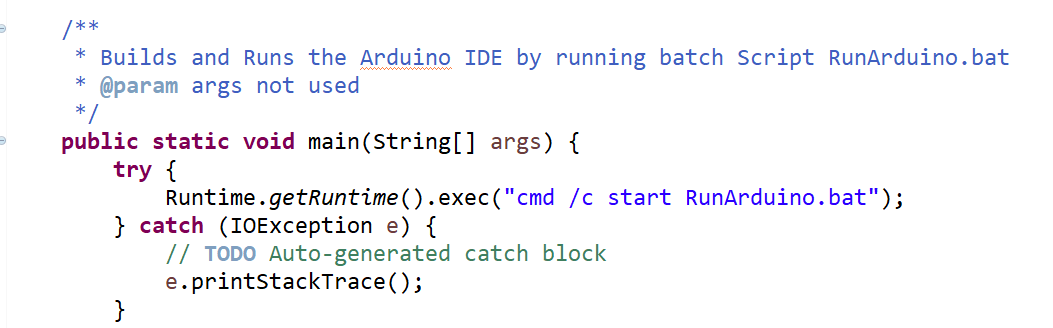
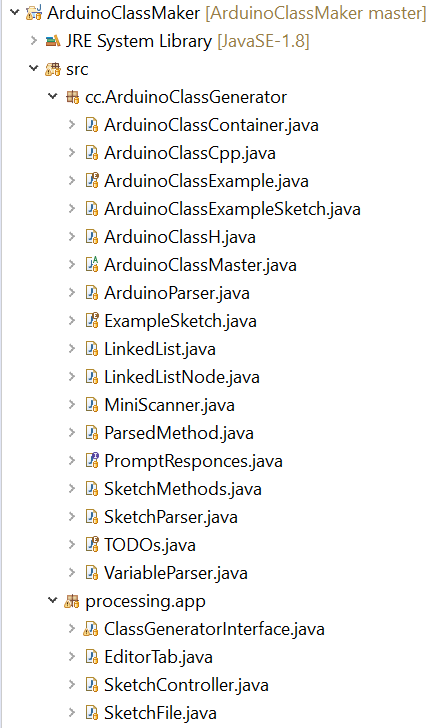
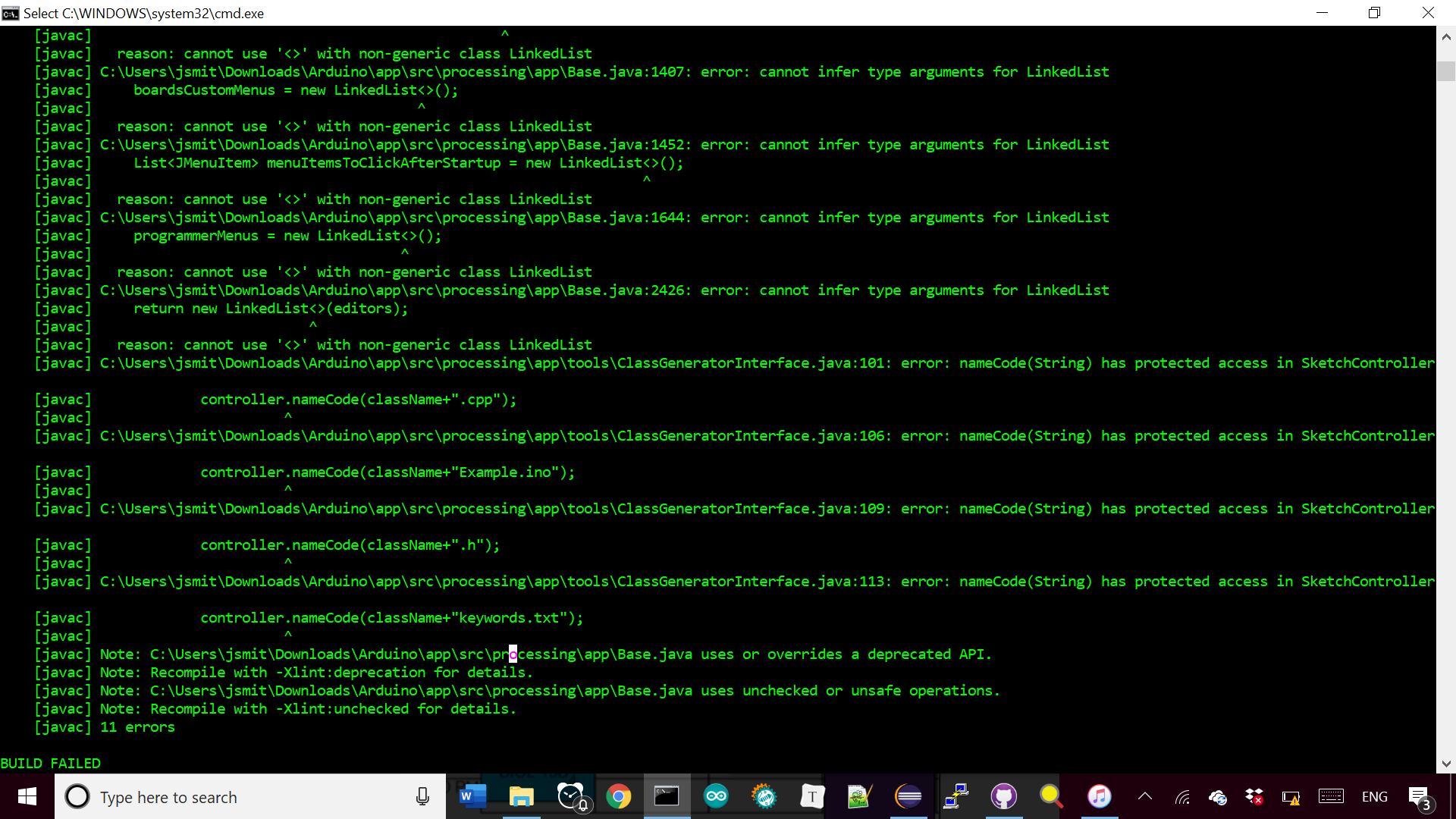
Finally, I can build the arduino IDE with the main method of the ArduinoClassGenerator class, which calls my batch file. **🡪**

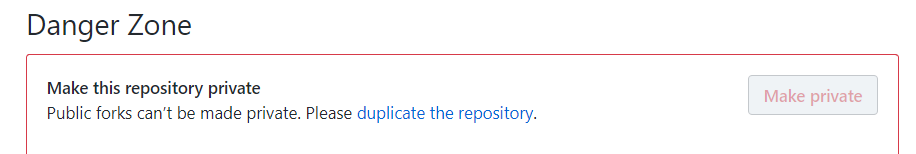
This setup means that all of the required files are in the ArudinoClassGenerator repository (as opposed to having to juggle classes in Arduino build and the batch file being in my desktop to run)

TOP: new Arduino File structure

Below: Error building arduino, sketchController var

Bottom: main method of ArduinoClassGenerator class calling batch file





The batch script is shown at left🡪

Line 15-18 automatically copy files from the ArduinoClassMaker workspace into the Arduino IDE.

Also note the use of function declaration line 33 to reduce redundancy.

The reason I didn’t make my project a fork of the Arduino github project is because I couldn’t import it as an eclipse project and because I can’t make the fork private**🡪**

I looked up using .jar files to package the class generator, but I couldn’t find an example of one being called from java code. Finally, I am including the only other additions to the Arduino IDE necessary to use the class generation feature.

The left shows the Editor class with the code necessary to add class generation as a menu option, which calls the generate library method of the ClassGenerator Interface class.

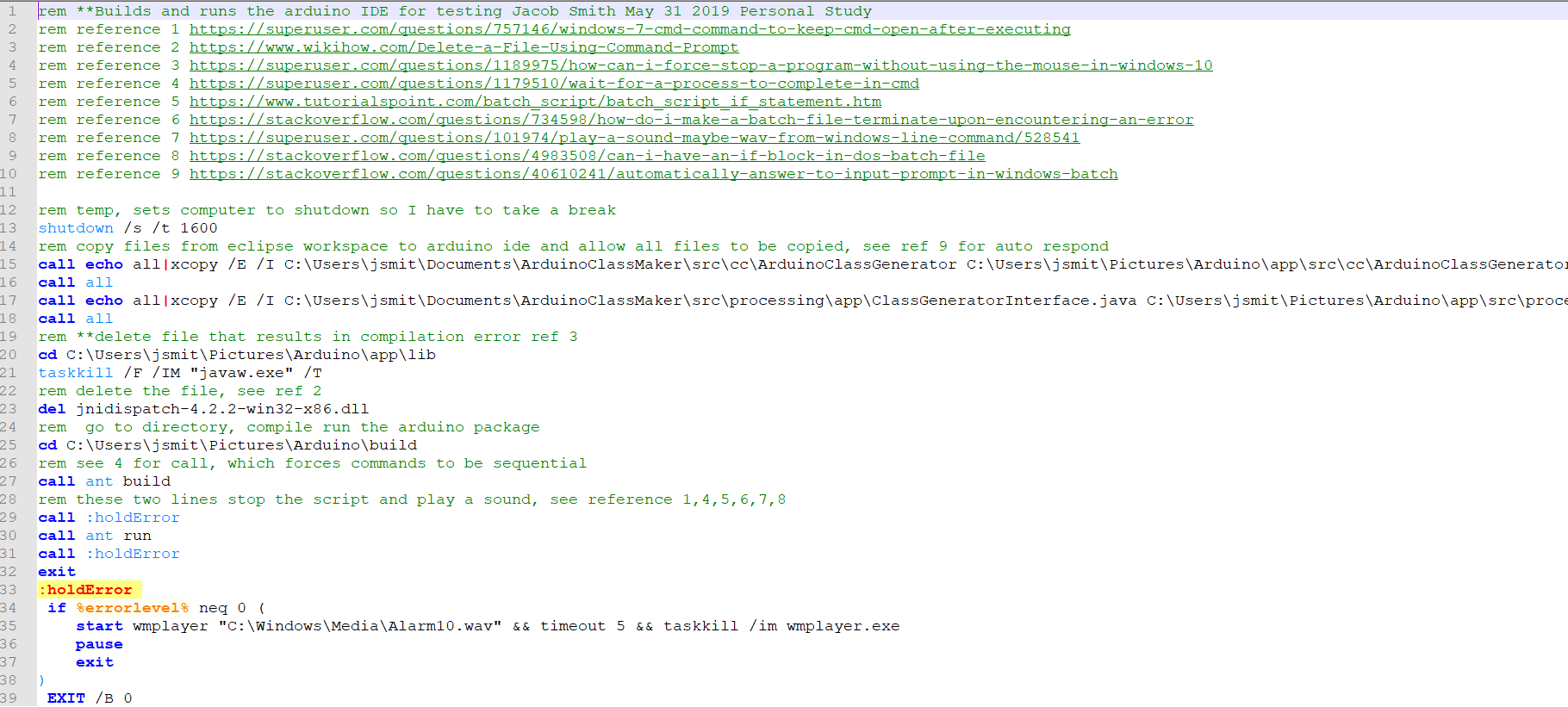
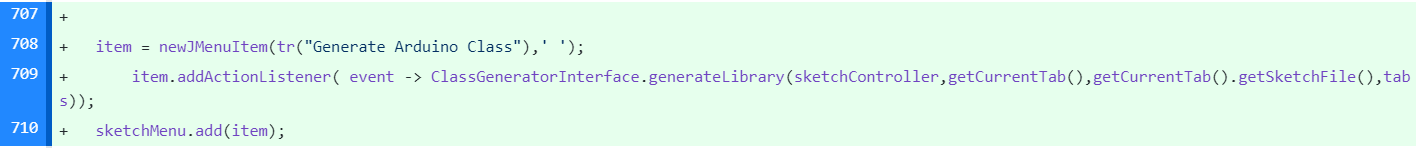
The last modification to the IDE was to allow .txt files🡪, so I can generate the keywords.txt file. This change is unlikely to be accepted by the developers of Arduino, but I could generate the keywords file without opening it.

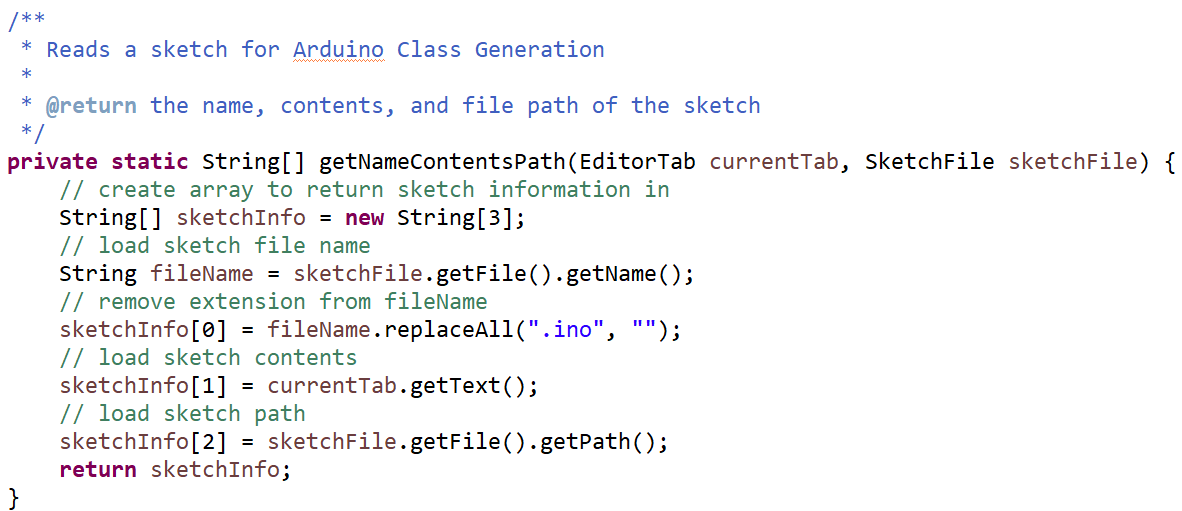
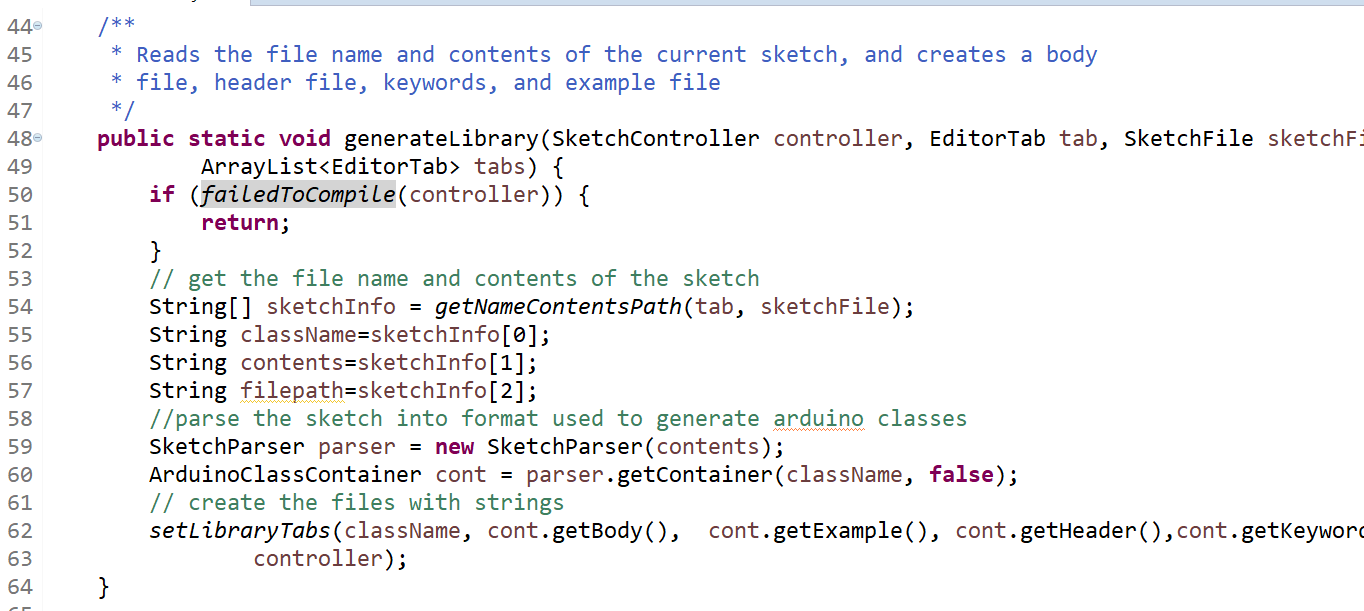


TOP: Github website privacy message

LEFT: Editor.java line 708 showing addition of Line 22 of Sketch.java allowing .txt file for keywords

BOTTOM: Batch file to build Arduino IDE





This is the ClassGenerator Interface class, which is all of the logic necessary to: read the file name and contents of a sketch, generate the class body, header, keywords, and example file, and display them to the IDE.

The failed to Compile method is 3 pages ago.

The getNameContentsPath method gets the filename and contents of the sketch without requiring the user’s input.

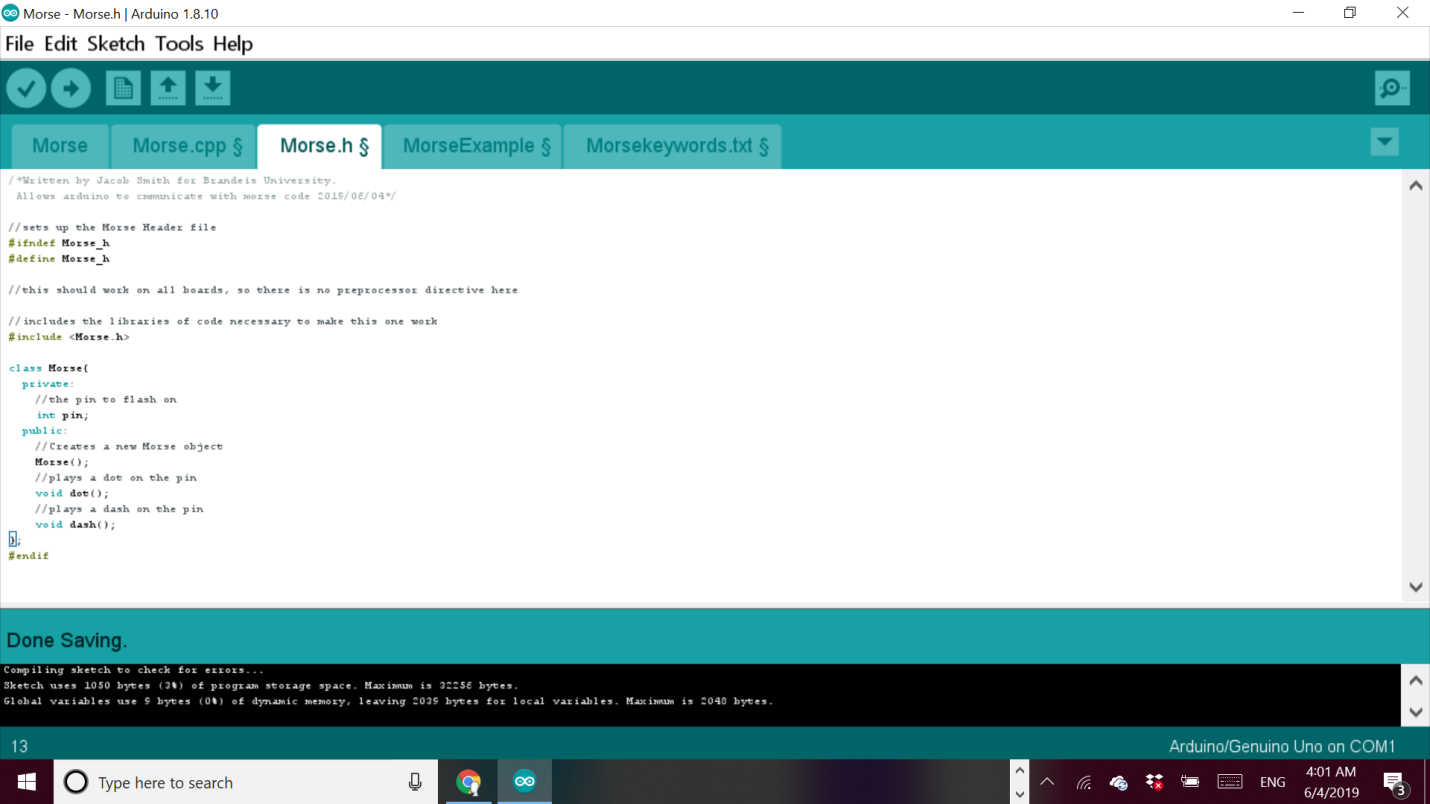
The SetLibraryTabs method actually creates the tabs in the IDE to display the generated class files.

The IDE with current tabs is shown at lower right corner. They are currently not saved automatically so the user can look over them before saving.

FUTURE WORK: A) writing unit tests that cover the generator’s behavior under different types of sketches, B) create interactive user prompts to get the author name, required boards, and organization, display a loading dialogue when class is being generated. C) research whether this project should become a public fork and pull request to Arduino, or whether I should keep it private.

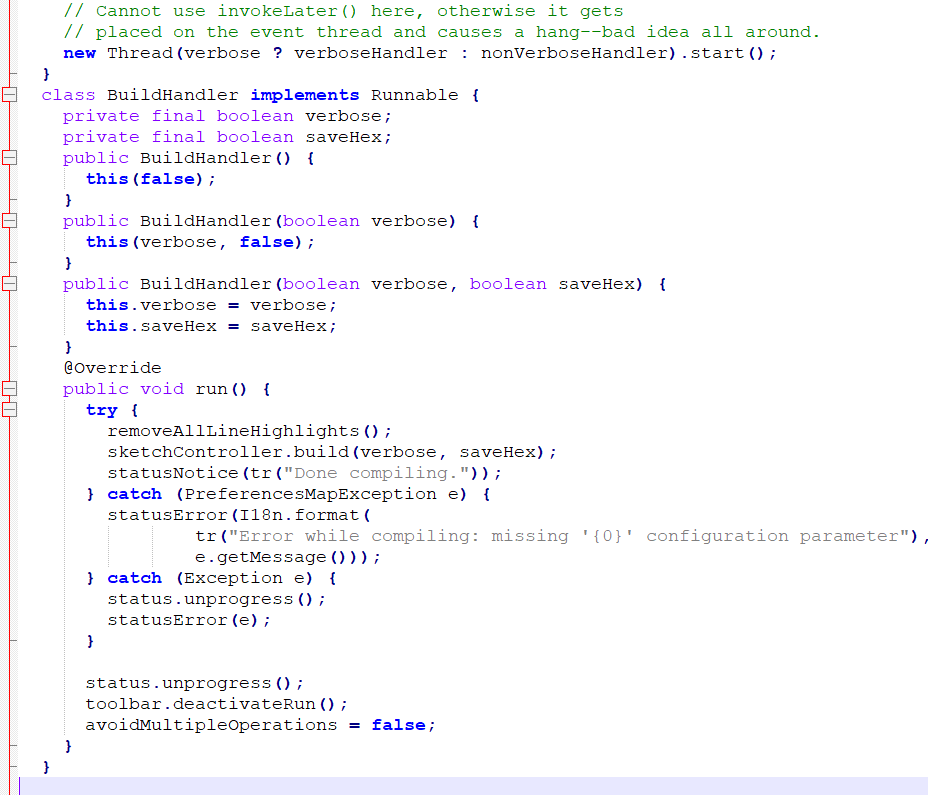
6/10/2019 1:14 PM: Pito Salas documentation, gif creator, gui multithreading, prompt messages.

Because the Class Generator takes a couple seconds to complete, I wanted progress messages to be displayed to show the user something is happening. I can use the progressNotice method of the



TOP: ClassGeneratorInterface Class

LEFT: Arduino IDE with generated class information tabs



Editor status method, but that only displays the messages at the end of execution. The reason is because graphical user interfaces are multithreaded. To find a solution, I found this source from codeRanch:

1. <https://coderanch.com/wiki/660058/JProgressBar-Doesnt-Update>

And this on Worker Threads:

1. <https://docs.oracle.com/javase/tutorial/uiswing/concurrency/worker.html>
2. [https://docs.oracle.com/javase/tutorial/displayCode.html?code=https://docs.oracle.com/javase/tutorial/uiswing/examples/concurrency/FlipperProject/src/concurrency/Flipper.java](https://docs.oracle.com/javase/tutorial/displayCode.html?code=https:/docs.oracle.com/javase/tutorial/uiswing/examples/concurrency/FlipperProject/src/concurrency/Flipper.java)

I simplified the code from sources 1 and 3, but in the end I looked up how displaying is handles by the Arduino IDE (TOP)--> , and I saw how the Editor class is able to display interm results while compiling.

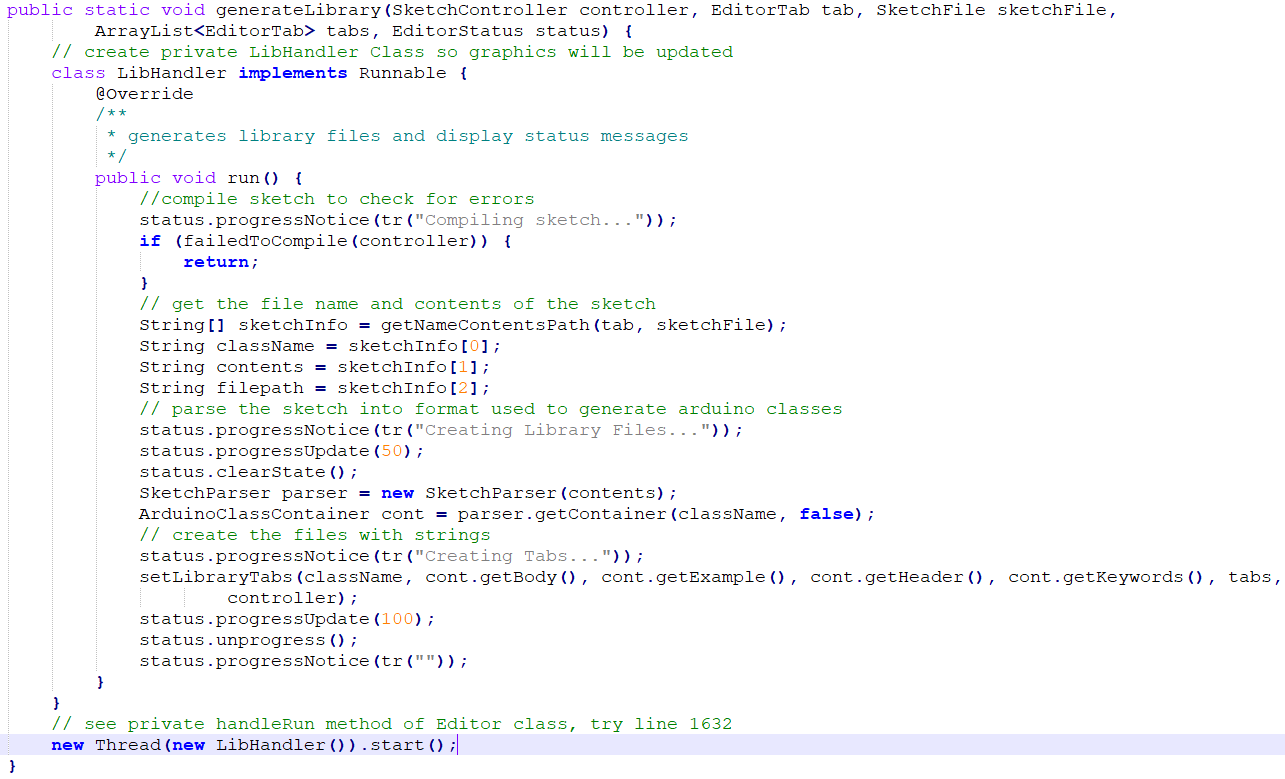
This method is to create an internal class that implements the Runnable interface, with a run method which performs computations and displays prompts, and then create a new Thread which calls the start method.

This approach makes sense, because it creates a thread other than the display thread to perform computations on, which was the cause of my problem.

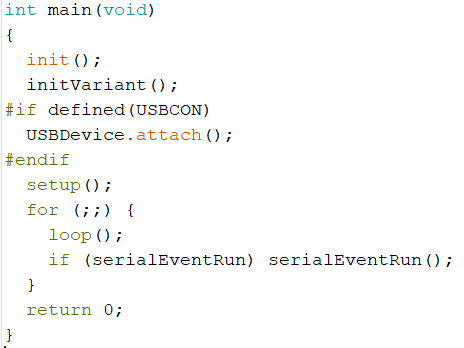
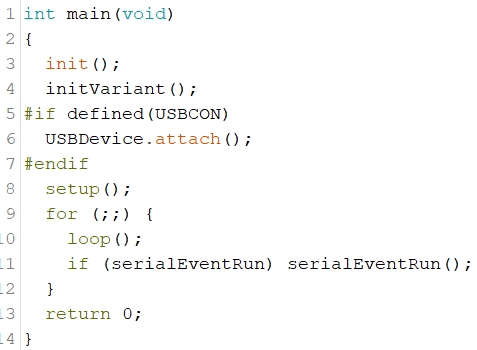
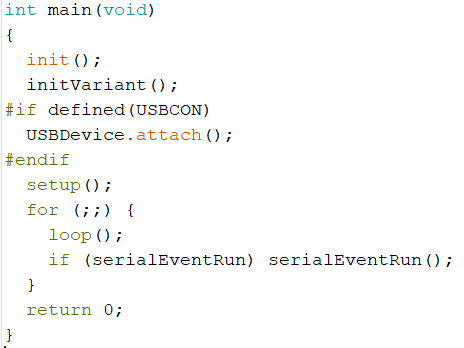
I created the internal libHandler class, where the run method displays prompts just like at top of page, and the class generator now displays the prompts properly.

(TOP) Editor Line 1630, Showing how compilation is able to display messages while thread is being compiled.

(Bottom) GenerateLibrary method of ClassGeneratorInterface, showing additions to display interm prompts to the user.



GenerateLibrary method of ArduinoClassGenerator. The LibHandler Class to show how to display interm messages in Arduino IDE, the new Thread line executes the run method.



June 14 2019 4:13 PM: I worked with Daniel to find a useful feature for the class generator. He created a class that set pin modes In the constructor, but they needed to be called in an initialization method at setup, because Arduino does background work before the setup method is called. References:

1. Problem with using Arduino in constructor<https://forum.arduino.cc/index.php?topic=243680.0>

background of setup and loop methods[2]<https://arduino.stackexchange.com/questions/4039/is-setup-and-loop-provided-for-convenience>

[3] Use setup method <https://stackoverflow.com/questions/18806141/move-object-creation-to-setup-function-of-arduino>

[4] Background on Arduino main method<http://arduino.land/FAQ/content/2/2/en/can-i-use-int-main-with-arduino.html>

[5] The Arduino Main method, files called are in same folder [https://github.com/arduino/ArduinoCoreavr/blob/master/cores/arduino/main.cpp](https://github.com/arduino/ArduinoCore-avr/blob/master/cores/arduino/main.cpp)

The related methods called in Arduino main method is:

Attach in USBCore

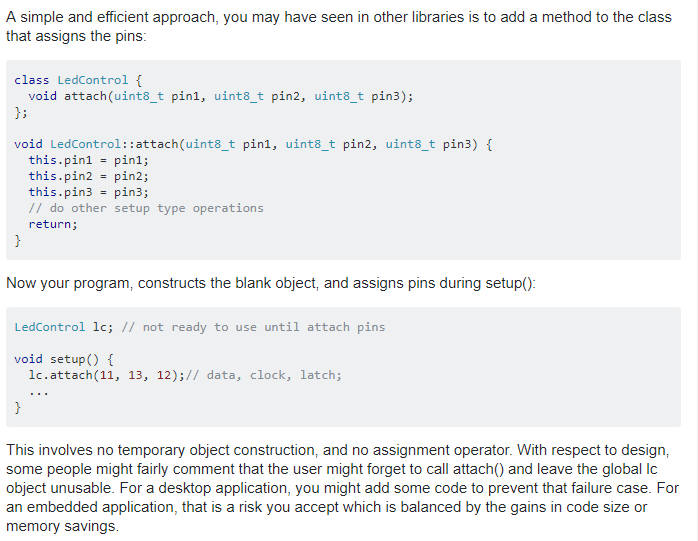
Init in wiring

initVariant line 28

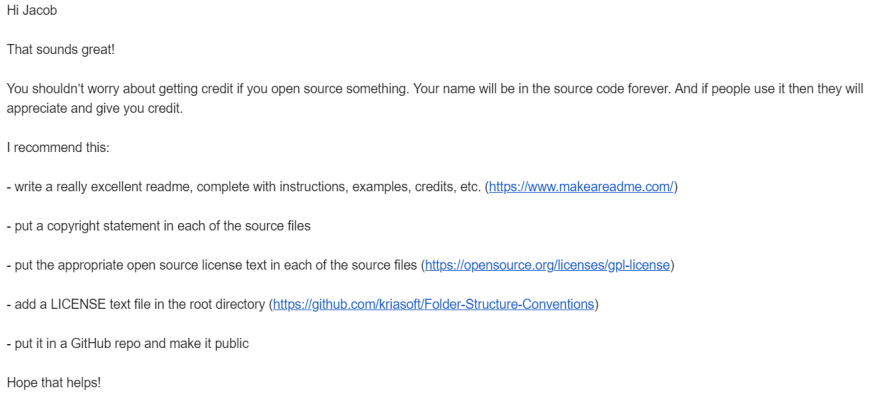
While looking through the Arudino source code, I found a new compiler directive, which may be useful to enforce scope (Consult with Tim).

[5] C Processor directive with #undef <https://users.cs.cf.ac.uk/Dave.Marshall/C/node14.html>

With this problem, I need to write the class generator to automatically create a begin method to be called in setup if anything complicated with the Arduino is being done (Consult with Tim).



(TOP) StackOverflow link 2 showing how an initialization method (called attach here) needs to be used inside setup method (BOTOM) Arduino Main method link 5



Useful commands for internet

Hostname-I to get IP address: <https://www.raspberrypi.org/documentation/remote-access/ip-address.md>

Ifconfig to get mac address: <https://www.raspberrypi-spy.co.uk/2012/06/finding-the-mac-address-of-a-raspberry-pi/>

These links show how to connect a webcam and run a server

Overall Tutorial

<https://www.instructables.com/id/How-to-Make-Raspberry-Pi-Webcam-Server-and-Stream-/>

Webcam usage

<https://www.raspberrypi.org/documentation/usage/webcams/>

Register rasberry pi brandeis open wifi network<https://netreg.brandeis.edu/index.php>

Download noobs rasberry pi

<https://www.raspberrypi.org/documentation/installation/noobs.md>

(TOP) Documentation from Professor Salas

DOCUMENTATION: I got an email from Professor Salas on how to better document the github page,--> and one useful feature was that I can add gifs in my readMe

Screen Capture to make readme[1]:<https://www.laptopmag.com/articles/how-to-video-screen-capture-windows-10>

Convert Video to GIF: [2]<https://giphy.com/>

Add Gifs to github readme: [3]<https://medium.com/@josephcardillo/how-to-add-gifs-to-your-github-readme-89c74da2ce47>

Rights information

[4]<https://softwareengineering.stackexchange.com/questions/19649/copyright-notices-disclaimers-in-source-files>

June 17 2019 Jacob Smith 4:07 PM: I am using a rasberry pi to run a live viewing server, which can help with publicity for the Arduino Outreach program

These links are to use the rasberry Pi

Getting Started

<https://projects.raspberrypi.org/en/projects/raspberry-pi-getting-started/5>

FAQ <https://www.raspberrypi.org/documentation/faqs/#networking>

A related project would be to use the rasberry pi to tweet a picture of the room every week:

Tweeting and taking picture tutorial

<https://projects.raspberrypi.org/en/projects/the-all-seeing-pi>

Just Tweeting tutorial

<https://projects.raspberrypi.org/en/projects/getting-started-with-the-twitter-api/8>

Scheduling a task in python:

<https://stackoverflow.com/questions/373335/how-do-i-get-a-cron-like-scheduler-in-python>

I talked to H about these ideas, and she suggested a doorbell feature. This means that by the end of this project, there will be monitors displaying live feeds of the makerlab rooms with some additional displays in the lobby, and the user can go to a website to view the streams or ring a doorbell in person.

This could then be connected to existing PIR sensors in used rooms to display when there is a person there (Daniel Lays idea).

While setting up the rasberry pi, I got an input/output error, which showed me that the Rasberry Pi uses its sd card as its storage so it must have an sd card in at all times. <https://www.raspberrypi.org/forums/viewtopic.php?t=26715>

6:21 PM: The secret to download noobs on previous page is to copy contents of noobs folder to sd card.

6/18/2019 11:18 Am Jacob Smith: I set time on Rasberry Pi using startup menu, and went to Brandeis ITS help desk to handle the MAC address registration on Brandeis Open (see page 102 of Zumo notebook)

4:49 PM: I spent today looking through functions on the Rasberry Pi.   
I can now tweet, read button presses, and take pictures from the rasberry Pi.

Some Errors I got along the way where

A folder not found error when the webcam was not plugged in

<https://forum.arduino.cc/index.php?topic=424233.0>

A parsing error with Python because methods need to be declared before they are called

<https://stackoverflow.com/questions/3754240/declare-function-at-end-of-file-in-python/3754284>

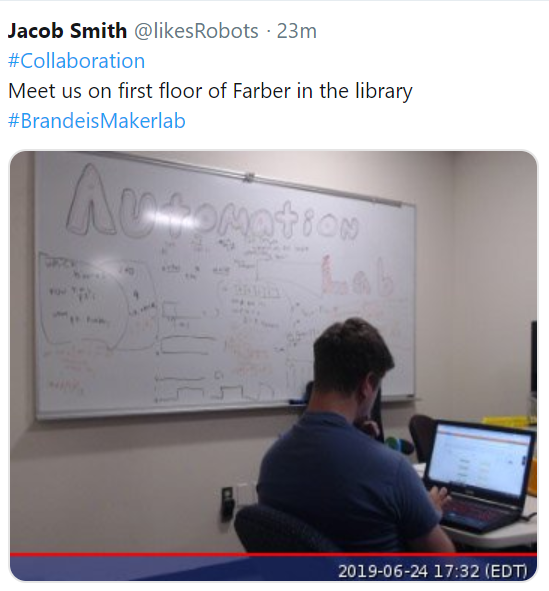
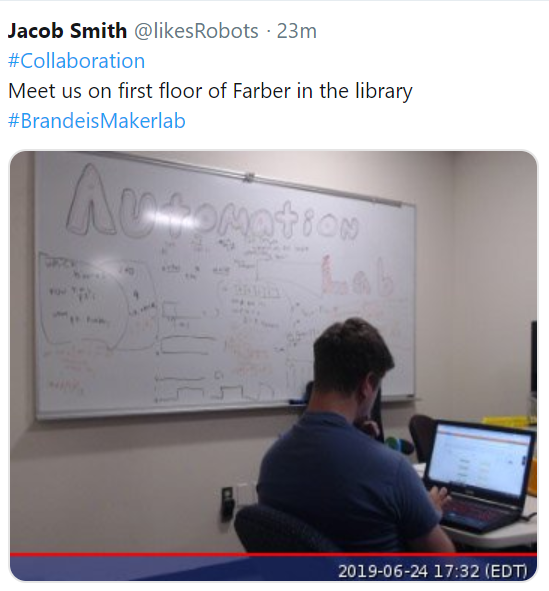
A server rejected error when remotely accessing IP address, Brandeis ITS gave me a secure file transfer protocol datasheet.

<https://www.ssh.com/ssh/putty/putty-manuals/0.68/Chapter10.html#errors-connrefused>

Python blocks need one element in body<https://stackoverflow.com/questions/43189302/syntaxerror-unexpected-eof-while-parsing?rq=1>

Defining a python function <https://www.tutorialspoint.com/python/python_functions.htm>

<https://www.codementor.io/kaushikpal/user-defined-functions-in-python-8s7wyc8k2>



Jacob Smith 6//21/2019 (Backlog of documentation)

<-**-This is the Twitter.py python program I wrote to: load my twitter account information, and tweet a newly taken picture when a button is pressed.**

The tweet also includes a randomly selected message such as #collaboration.

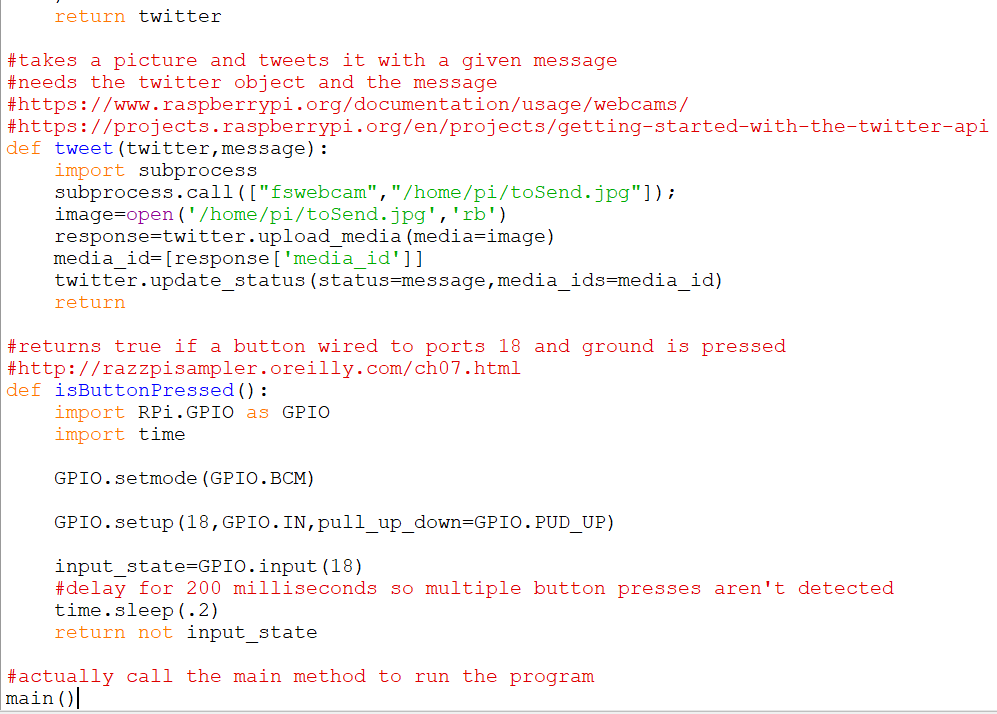
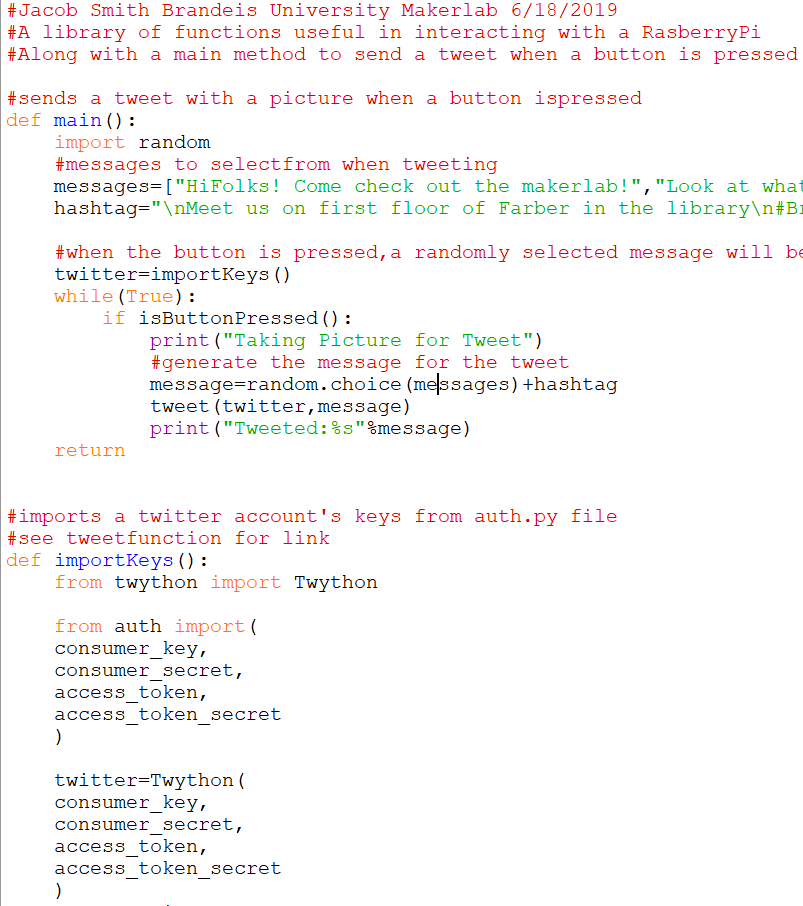
The programs are modified from existing tutorials.

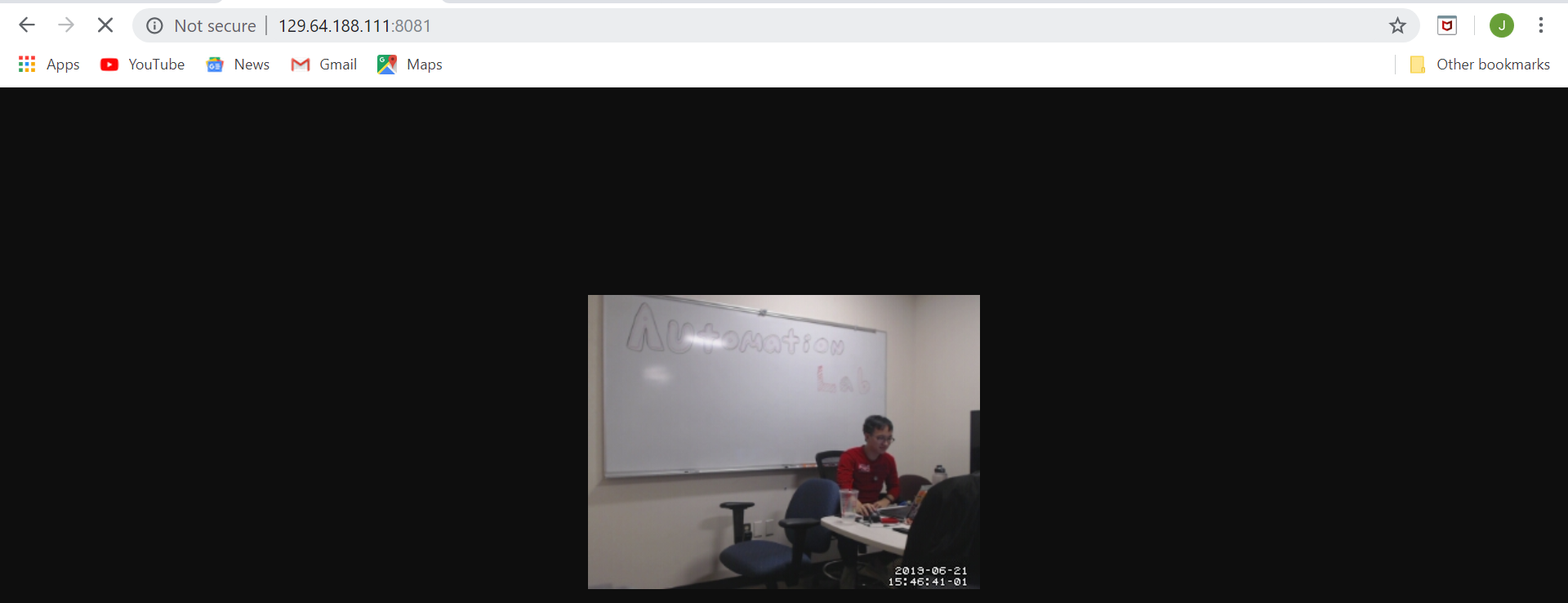
In this version, I am importing code in a method instead of at the top as I would in java, which is bad practice.

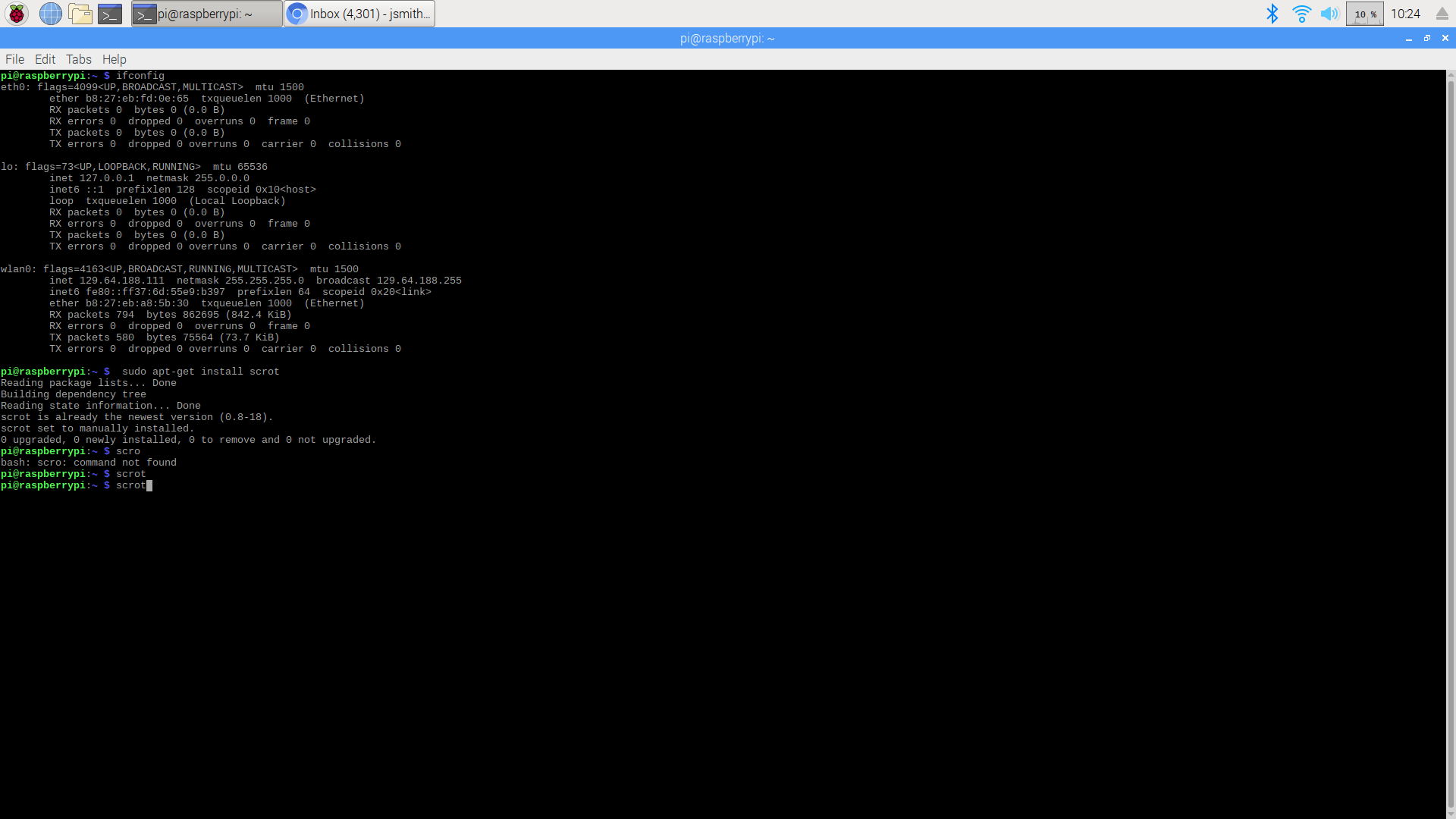
Error messages I encountered are presented here and on the next page

**Tweet Generated fromTwitter.py Python program**

While loop in python <https://www.tutorialspoint.com/python/python_while_loop.htm>







Result of ifConfig command in rasberry Pi, with correct IP adress for remote access highlighted

6/21/2019 1:02 PM Jacob Smith

I got fswebcam to display the server, using Tims help with step 6. That step involved using the Rasberry Pi Terminal to find the configuration file and editing it.

I needed to learn more about Linux commands<http://linuxcommand.org/lc3_lts0020.php>

Another command that was necessary was how to stop the motion stream:<https://ubuntuforums.org/showthread.php?t=2145787>

**Output of fswebcam server**

Run a bash script in python

<https://unix.stackexchange.com/questions/190495/how-to-execute-a-bash-command-in-a-python-script>

File Paths for imaging <https://stackoverflow.com/questions/11727598/pil-image-open-working-for-some-images-but-not-others>

Sleeping in Python

<https://stackoverflow.com/questions/510348/how-can-i-make-a-time-delay-in-python>

Randomly Select element from a list python

<https://stackoverflow.com/questions/306400/how-to-randomly-select-an-item-from-a-list>

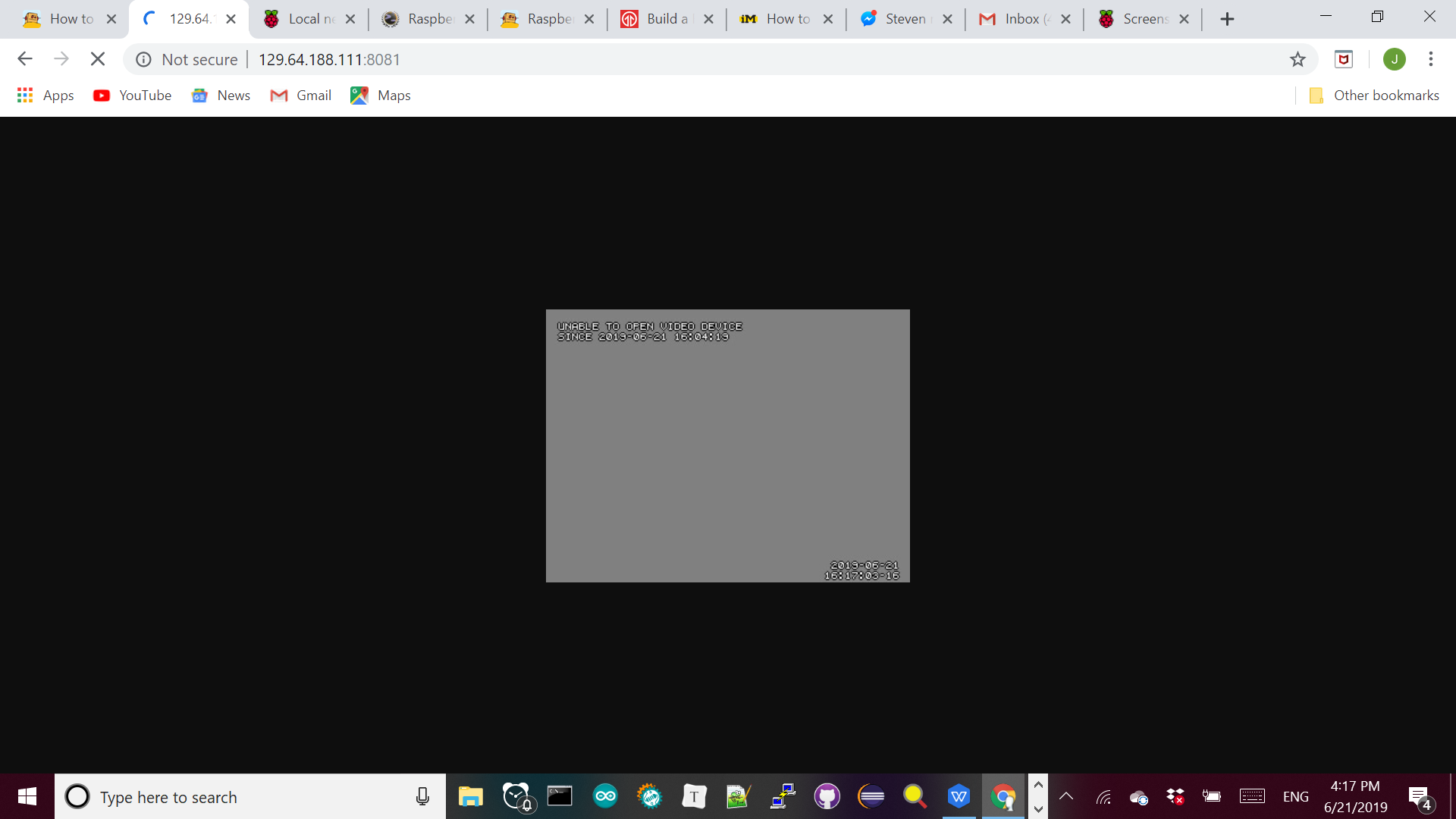
Python whitespace error <https://stackoverflow.com/questions/26720841/python-indentationerror-unindent-does-not-match-any-outer-indentation-level/34100174>

Jacob Smith 10:20 AM 6/19/2019: I am continuing on the Rasberry Pi Webserver and tweeting project, I connected to the ip address now, it was the broadcast ip address, I was using the inet ip address-->

<https://www.raspberrypi.org/forums/viewtopic.php?t=172891>

LEFT: Error with one of web server commands shows blank picture

RIGHT: Error when using streamer.sh



Unfortunately, the video is about one frame per second, which is the same as this post : <https://raspberrypi.stackexchange.com/questions/29543/motion-filming-cant-get-on-real-speed>, which recommends MJPEG

I tried using MJPEG

<https://blog.miguelgrinberg.com/post/how-to-build-and-run-mjpg-streamer-on-the-raspberry-pi>

MJPG Streamer download: [https://github.com/jacksonliam/mjpg-streamer](https://github.com/jacksonliam/mjpg-streamer#inbox/_blank)

This results in a address already in use message, which seems that the other webserver I started isnt closing

In addition, this forum claims to have a high enough frame rate to control a robot from the feed:

[https://www.raspberrypi.org/forums/viewtopic.php?p=751735](https://www.raspberrypi.org/forums/viewtopic.php?p=751735#inbox/_blank)

The tutorial resulted in a error opening V4L interface error,

**See Right-->**

[https://raspberrypi.stackexchange.com/questions/77768/error-opening-v4l-interface-no-such-file-or-directory](https://raspberrypi.stackexchange.com/questions/77768/error-opening-v4l-interface-no-such-file-or-directory#inbox/_blank)

However, this stream is also very slow

I looked for other streaming platforms than motion, and tried this tutorial:

<https://chriscarey.com/blog/2017/04/30/achieving-high-frame-rate-with-a-raspberry-pi-camera-system/>

But it resulted in a failed to create component vc.ril.camera message

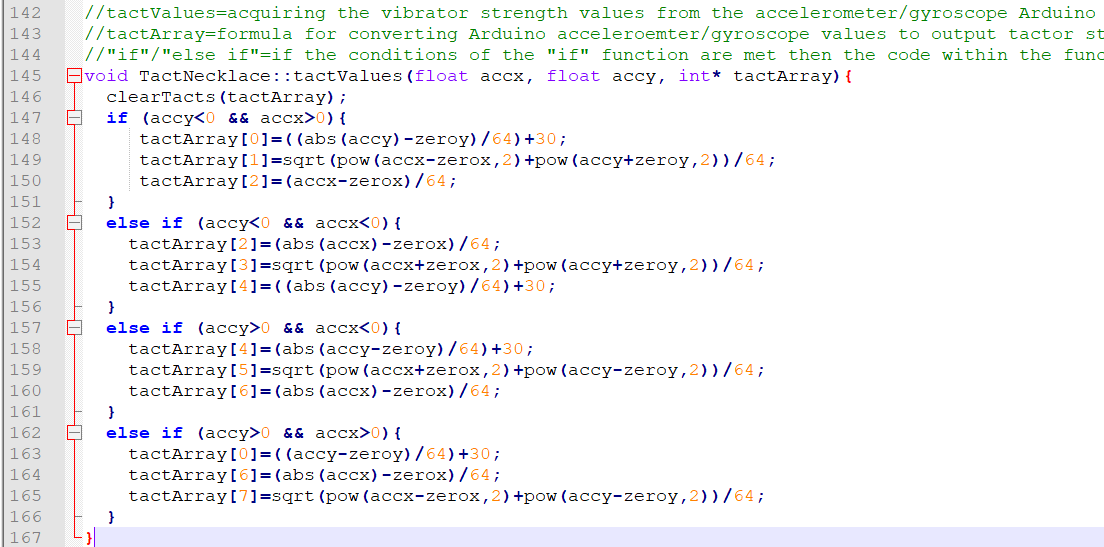
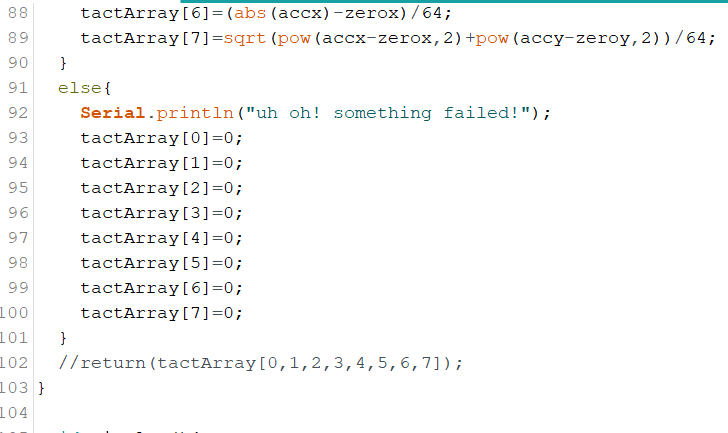
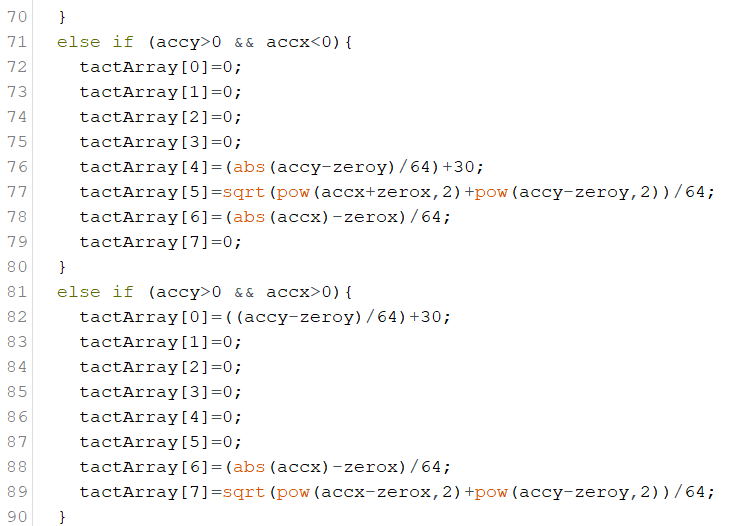
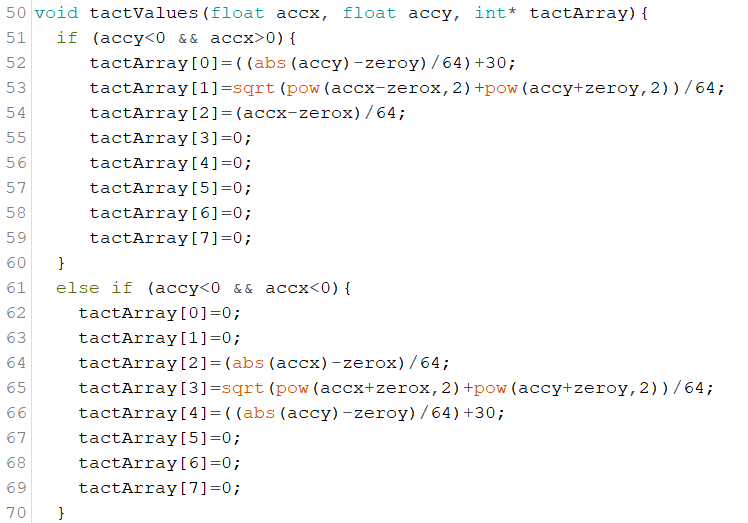
**Below ↓** is an example of the feed not working

6/24/2019 Jacob Smith: also today and last week, I helped Aiden and Johnny, two rising college freshman engineering students who wanted to turn their large Arduino sketch into an Arduino library.

This has implications for my class generator project, as I want to show that it can handle different coding styles.

Tactile Necklace Project: <https://github.com/AidenKunkler-Peck/Tactile-Necklace>.

The sketch at right shows their condition checking to start, and the below image shows how I got them to sue methods and control flow to clean it up right. **RIGHT, high school tactarray method, BOTTOM same method after refactoring**



Also, the process of writing a library and documenting it made the students think more critically about their program, which I created this repository to hold example sketches for:<https://github.com/jsmith2021Brandeis/ArduinoEducation>

The skills I taught them would serve as a good second part of my Arduino outreach program. The topic we covered where:

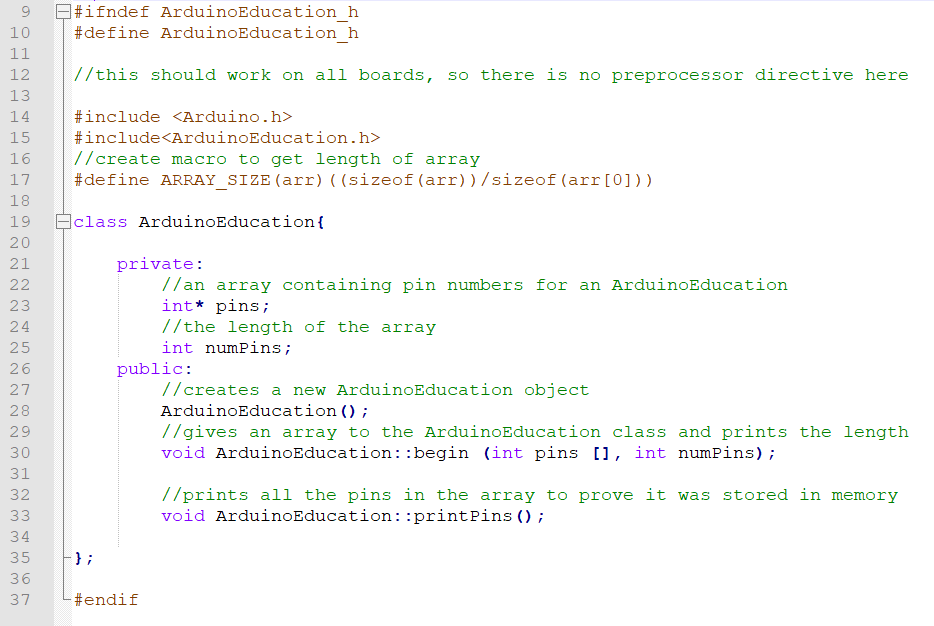
* Using github desktop
* Creating a github repository with a readme
* Using loops to avoid repition and methods to keep code clear
* Creating a library out of a sketch
* Main ideas of object oriented programming, along with what in the new library to make private vs public.
* Class constructors

The areas we spent the most time on where for loops, the mindset of using private class methods to increase usability, and debugging the library.

While Aiden and Johnny where quick studies, they already had a summer of programming experience, and I need to make sure my tutorials work for someone who is completely new to programming.

Today, I worked with Aiden to make his project work for any number of pins and any board. This lead to working with arrays in Arduino.

**Arduino Education Class-->**



(TOP) ArduinoEducation Header File (see next page)

(BOTTOM) ARDUINO EDUCATION EXAMPLE FILE

Shows how arrays can be used with classes in Arduino as cleanly as possible, array size macro highlighted

Jacob Smith 12:45 AM: I have been reading more about arrays in Arduino, and I’ve concluded that the closest I can get to how they work in java is to define an array size macro and pass it to the class.

I accomplish this with the **ArduinoEducation class, shown on this and previous page-->**

The nice part is the macro can be defined in the header file, away from the

User in the sketch file, see previous page.

The reason I can’t find the array size inside the class is because of array decay, as arrays turn into pointers when they are used by a function in C.

The only way to avoid the decay seems to require knowing the length in the first place, so it is not preferable:

Sources:

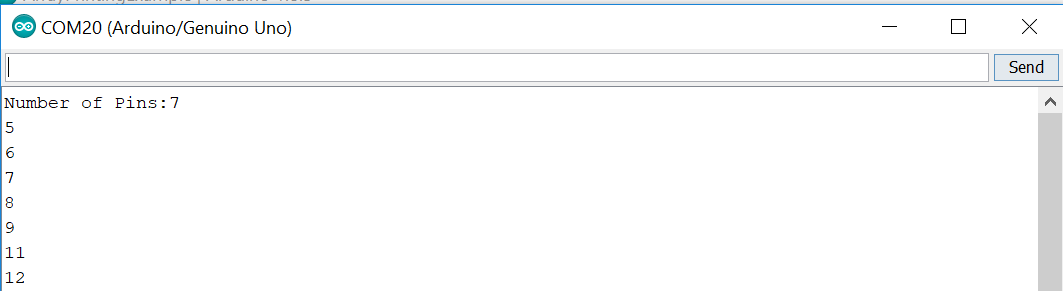
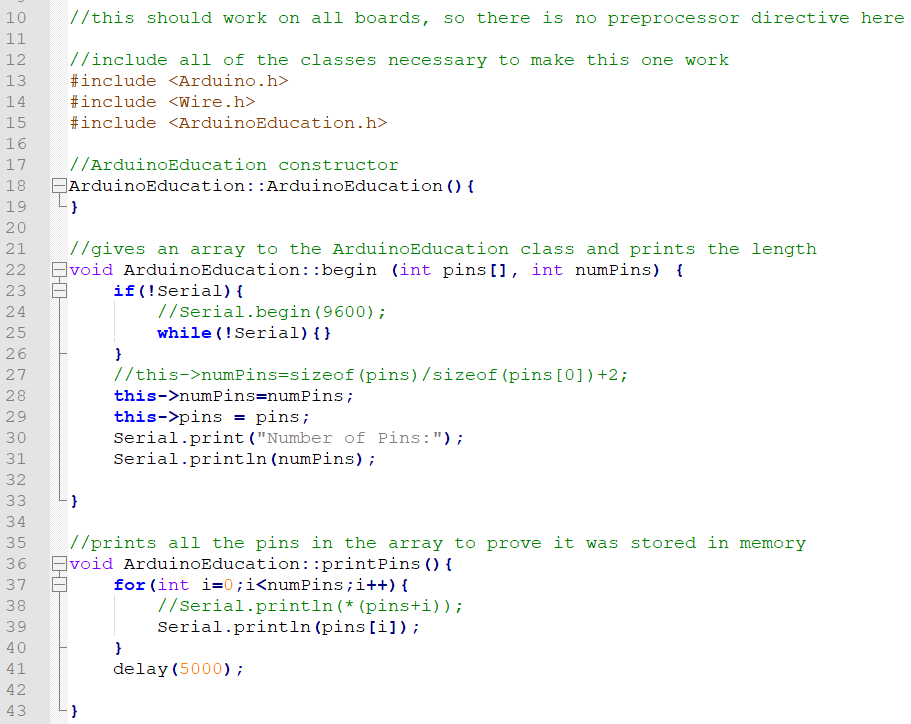
Array class Variable:[0]<https://forum.arduino.cc/index.php?topic=508126.0>

Background on Arrays in C [1]<https://blog.feabhas.com/2013/11/shock-horror-i-learned-something-about-arrays-in-c/>

FUTURE WORK: Tomorrow I will show Johnny this way of handling arrays to get dynamic length and high usability. In addition, I will show him how to display a warning if a board other than an uno or nano is used, saying that the necklace has not been tested with that board.

This will allow advanced users to ignore the warning, but still let new users follow the tutorial (Tim’s advice).

After that, they just need a tutorial explaining how to use their necklace with an uno and they should be finished.



(TOP) ArduinoEducation Example File Output

(BOTTOM) ARDUINO EDUCATION Body File

Shows how arrays can be used with classes in Arduino as cleanly as possible

Jacob Smith 6/25/2019 5:41 PM: I think the previous high frame rate tutorial is for rasberry pi camera

2012/5/7 post shows camera needs yuyv format:<https://sourceforge.net/p/mjpg-streamer/discussion/739917/thread/dfc46d26/>

6/26/2019 12:02 PM Jacob Smith: yesterday, I got the web stream to work at 60 frames per second!

First, I followed this tutorial to get the stream to work at all <https://blog.cudmore.io/post/2015/03/15/Installing-mjpg-streamer-on-a-raspberry-pi/>

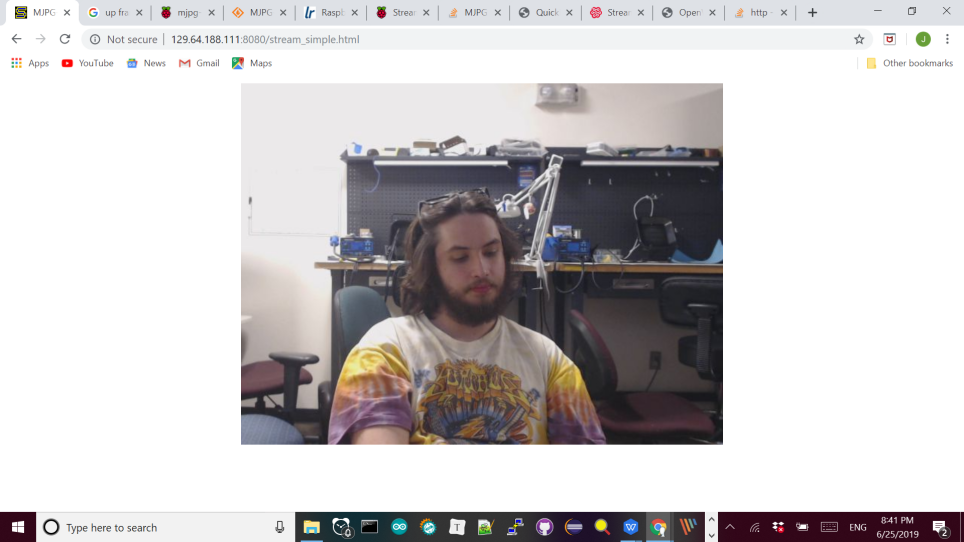
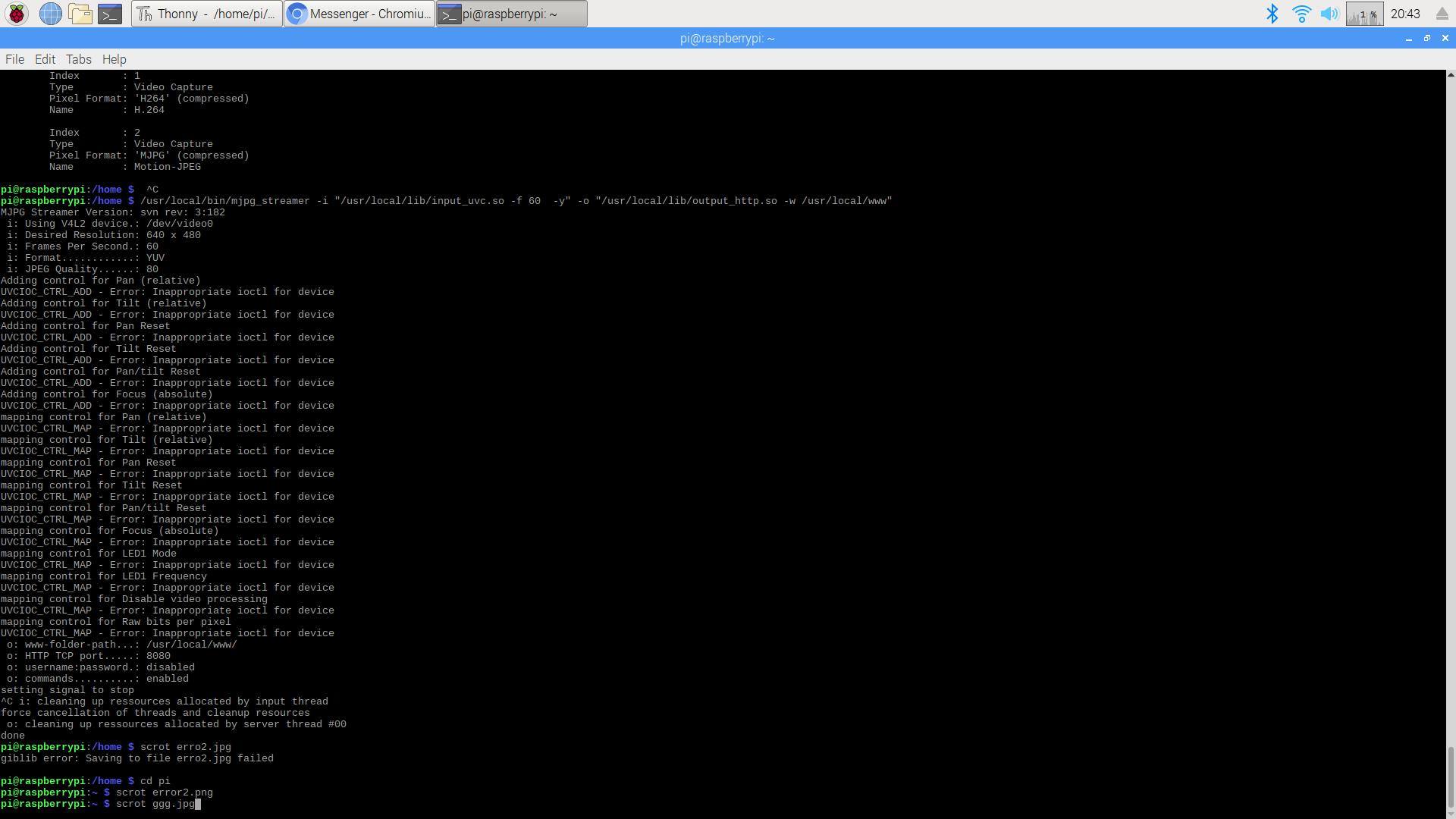
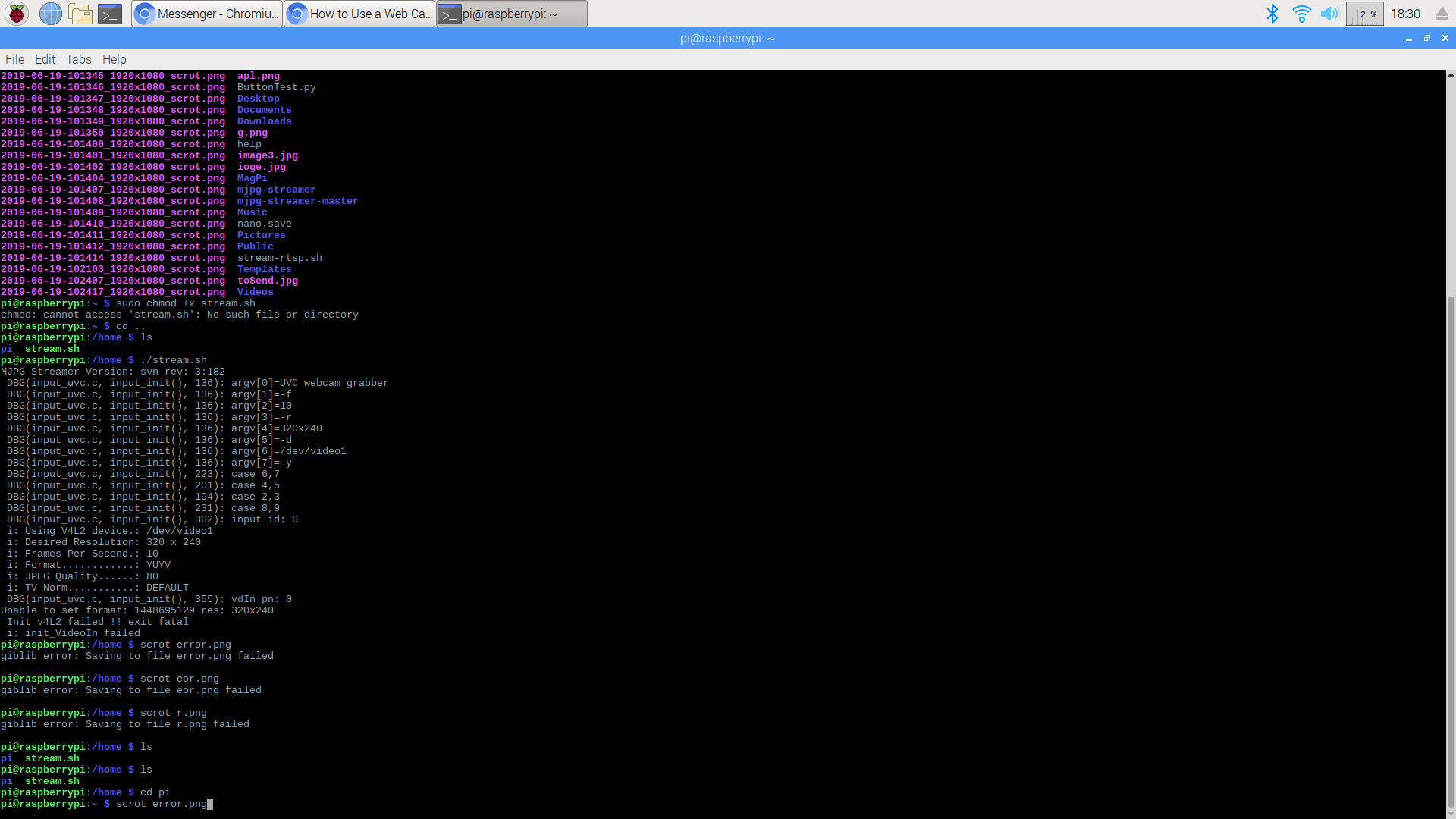
Then, I used this: 2010/4/8 :<https://sourceforge.net/p/mjpg-streamer/discussion/739917/thread/541a9b8c/> to change the frame rate from 6 fps to 60 fps.

**The stream and command line results are shown right and bottom (note highlight box showing framerate)-->**

The next goal is to allow streaming with audio and video, and YUYV can’t do that.

The Logitech c290 camera I am using supports the h264 video format, which includes audio

<https://raspberrypi.stackexchange.com/questions/4412/streaming-h264-with-logitech-c920>



(TOP)yuyv format spec

(LEFT)working mjpg stream at 60 fps 129.64.188.111:8080/stream\_simple.htm

(BOTTOM)message showing success



Empty Server Stack which will house the rasberry Pis.

Jacob Smith 6/27/2019 8:13 PM

TACTOR PROJECT CODE CLEANUP:Yesterday, I helped Johnny remove incorrect for loop conditions (which explains why the tactors where not working with incorrect array size two days ago).

Today, I helped Johnny print out detailed sensor data and write error messages in his code, while Tim gave us the idea to explicitly allocate array memory. This is the first time I’ve gotten an array in a class to work without the size of the array being specified in the class.

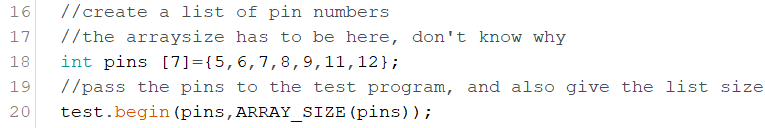
**See modification of Arduino Education Sketch at left-->**

Also, Tim’s help got us to see that a for loop in SendVibration was causing a memory error, which was stopping the program. Whenever a program stops running, the main reasons are power and memory (two issues not encountered in a computer science course).   
TACTOR LAYOUTS: In addition, Johnny’s code now works for either 4 pins on an uno, or 8 on a nano (the uno necklace is simpler and will be used for the workshop).

FUTURE WORK:Tomorrow, I will have Johnny make a [dozuki guide](https://brandeismakerlab.dozuki.com/) for the uno setup.

Jacob Smith 7/3/2019 12:46 PM: PISERVER:This week, I have been helping Aiden set up the Rasberry Pi Cluster Computer. He found and registered the MAC dresser of the raspberry Pis and formatted their sd cards according to the specified node.

**This is where the rasberry Pis will be mounted and connected to the internet with ethernet-->**



ArduinoEducation Example Sketch showing explicit array size. (replaces line 16 of bottom code 3 pages previous)

TackNecklace ClassFlow Diagram showing enhanced modularity.

TackNecklace Power Tactors based on sensor values

Temperature Read values

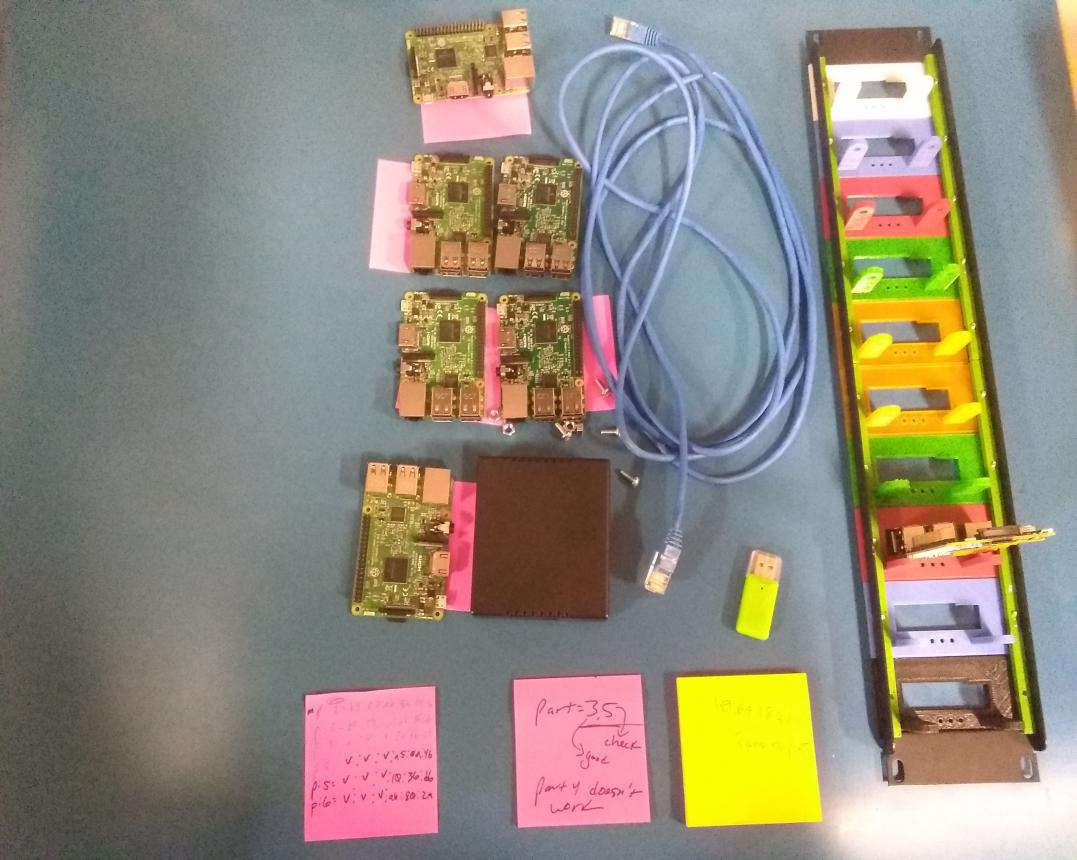
Temperature Convert sensor data to tactors

Ultrasonic Read values

Ultrasonic Convert sensor data to tactors

Acceleromter Read values

Accelereometer Convert sensor data to tactors



Rasberry Pis and Mounting for Rasberry Pi Linux Server Project

This is the equipment that Aiden is using for the project, note the **colored rasberry pi mounting material at right**

**-->**

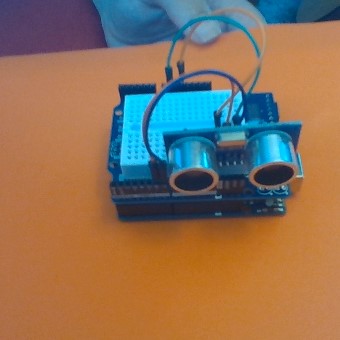
Daniel and I inspected the server stack today, and if we want to use the Dell Poweredge Rackmount Console, we would need to connect it to a computer that uses analog display. If we want to only use rasberry Pi computers, we could either use a converter <https://www.freetv.ie/vga-to-hdmi/> or replace the display with an hdmi display.

TACKNECKLACE: Also this week, I worked with Johnny to make the Tactile Necklace Program more modular so it can work with multiple sensors. Currently, the TackNecklace class just has all the sensor data, but to really follow good practice, I will have Johnny create classes for each of the sensors to be used, designed in a specified format for interfacing with the vibrators.

**Class Diagram for new TackNecklace-->**

It’s hard to explain to the students why a multi-class approach is better than putting sensors and motors into one large class, but the reasons are organization, modularity, and reducing the number of lines of programs.

Also, the level of code clarity required for this project is higher than usual, both because these programs will be used by novice programmers and because they all need to send an output to the TackNecklace class. For true encapsulation, I would want the sensor classes to return a magnitude and direction of sensed data, which is closer to a universal format.



(TOP) Ultrasonic Sensor wired to Arduino for demonstration

(BOTTOM) ArLCD showing analogMeter example file for demonstration.

DIGITAL WINDOW: While demonstrating the digital window tweeting to precollege students, I saw that the digital window wasn’t able to take pictures because the camera was busy running the stream. Solutions for this would include automatically pausing the stream to take a tweet or making the tweet be a screenshot of the stream. I prefer the latter approach because it would allow the user to see themselves while taking the picture.

ARDUINO OUTREACH: Now that I finished my trainings and am officially hired by the makerlab, I’m getting ready to contact laboratories to gauge interest for the electronic course. I drafted an introductory email which was reviewed by Tim and will be reviewed by Ian next week.

FUTURE WORK: That gives me this week to lay out the introductory materials of the course and prepare a publicity campaign.

July 6 2019: This weekend, worked on preparing content examples for the Arduino electronics course.

First, I used these tutorials to get an **ultrasonic sensor running for demonstration-->**

Wiring diagram: <https://www.theengineeringprojects.com/2017/08/ultrasonic-sensor-arduino-interfacing.html>

Program:<https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>

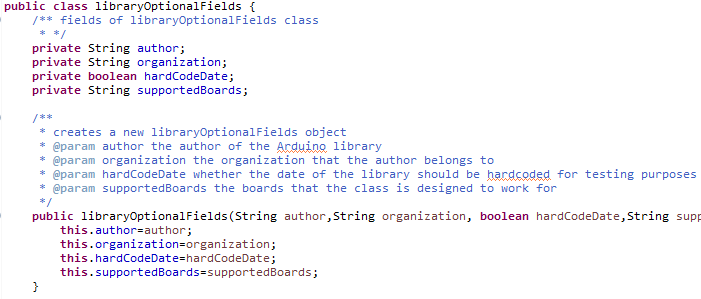
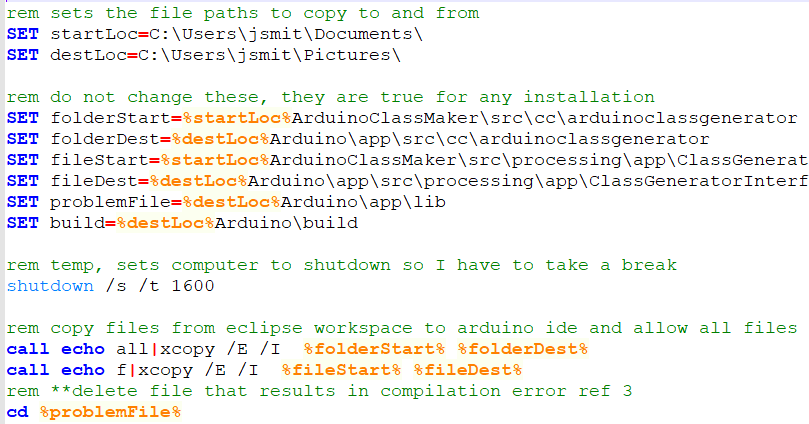
**ARLCD Running Example File AnalogMeter (see next page)-->**

|  |  |
| --- | --- |
| **Example File/Category** | **Purpose**  I also started up the ARLCD <https://earthlcd.com/products/ezlcd-intelligent-lcds/arlcd/>  so it can be used as an example output.  The library is posted here:  <https://github.com/earthmake/arLCD/issues/2>  The LCD worked just by running the **example files, which I have categorized at <-- right**  FUTURE WORK: Now, I will create wrapper classes for the Ultrasonic sensor and arLCd that allow for an Arduino demonstration, like printing distance with an analogMeter |
| **Communication** |  |
| SerialMonitorTest | shows ability to display well formatted text |
| **Complex Interactive Displays and Demos** | Allow the user to use the lcd as a touchscreen in some goal, or a more advanced demonstration |
| arlCDTest | has touch screen |
| choice | displays a menu |
| slider | shows ability to display a slider |
| static2 | Uses a button to flash virtual light |
| tictactoe | plays a tic tac toe game with lcd |
| touchzone | displays a keypad |
| towers1 | displays the towers of hanoi |
| **Data Display** | Demonstrate simple ways to display an output to the lcd |
| analogMeter | makes a nice dial |
| digitalMeter | shows a box withtext field |
| gauge | shows a bar that can be filled |
| **Files** | Allows lcd to interact with files |
| files | shows file reading, don't understand |
| picture | allows screen to display a picture |
| **Graphics** | Shows drawing capabilities of lcd |
| circle | displays a nice drawing |
| cls | stes background color |
| Colorid | shows three example colors, don't understand |
| fonts | shows fonts available |
| linetype | shows lines for drawing |
| print example | shows lcd's ability to print text |
| rect | draws rectangles |
| simple | simple |
| **Plots** | Show’s LCD’s ability to display different plots |
| pie | shows lcd's ability to display pie chart in different line styles |
| plot | displays a scatterplot |
| **Simple Interactive Displays** | Allow the user to touch the display |
| button | lets you press button |
| checkbox | has a checkbox |
| dial | shows a touch dial |
| touch | shows feedback of screen, what coordinates user touched |
| wstate | shows ability of button to be enabled or disabled |
| **Notes** |  |
| Interrupt options are available, like radio\_interrupt | shows lcd's ability to send to Serial Monitor |



(TOP) updated bash script with variable names (BOTTOM) libraryOptionalFields class and usage

(BOTTOM) ArLCD showing analogMeter example file for demonstration.



ARDUINO CLASS GENERATOR: Now that I am writing Arduno libraries again, I decided to revisit my Arduino Class Generator. First of all, I got my batch **script to correctly copy files using variable names for file paths -->**

This approach makes my script more readable, and should make it easier for my suite to be used on another computer.

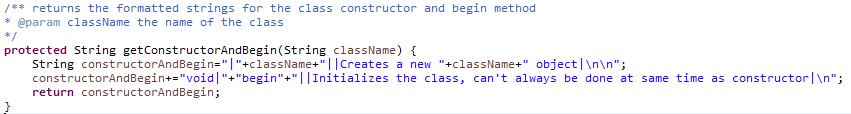
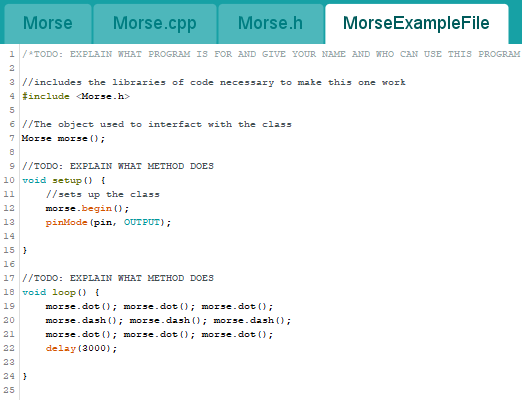
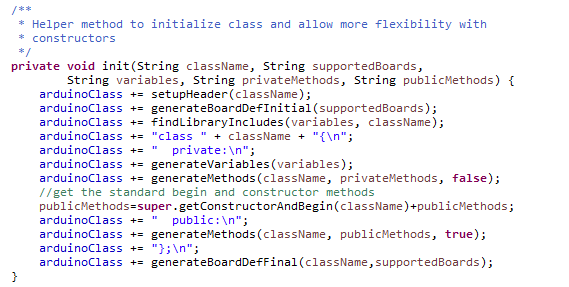
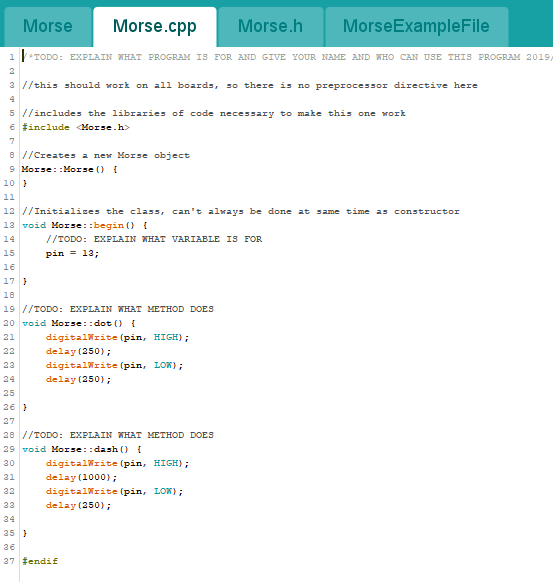
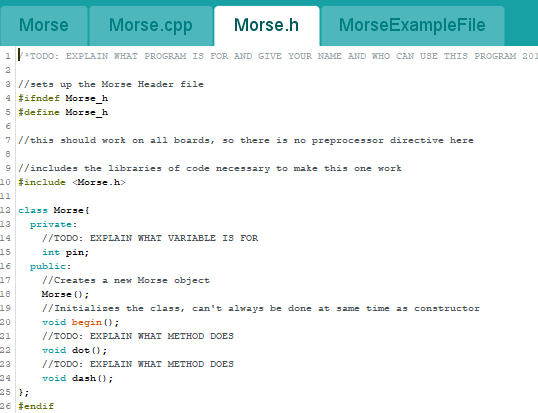
Another change I made was to avoid long parameter lists according to the arduino github code review. I followed this approach, <https://stackoverflow.com/questions/439574/whats-the-best-way-to-refactor-a-method-that-has-too-many-6-parameters> which is to wrap related fields into a class. I created the **libraryOptionalFields class -->** to do this for the class author, organization, hardcoded date, and supported boards.

I find that this approach allows me to make more code more elegant, as I can pass an object through multiple levels and I can make fewer constructors

Finally, I renamed the arduinoClassGenerator package to be lowercase, which was complicated because of this:(see first post and windows operating system)

<https://bugs.eclipse.org/bugs/show_bug.cgi?id=168208> It worked to rename the package with a different name, then rename again to the all lowercase name

FUTURE WORK: I will make the June 14 2019 begin method feature and use it to make the Ulltrasonic and arLCD wrapper classe, or make them manually to save time.



show inclusion of begin method (TOPLEFT) morseExample.ino example file (BOTTOMLEFT) morese.cpp body file (TOPMIDDLE) getConstructorandBegin method of ArduinoClassmaster (MIDDLEMIDDLE) more.h class file (RIGHTBOTTOM) ARDUINOCLASS H init method

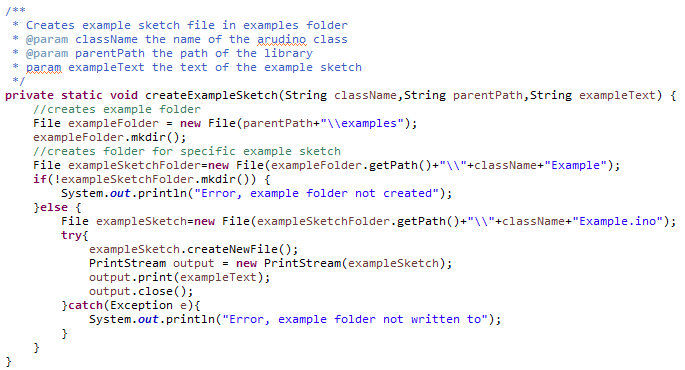
7/7/2019 4:33 AM Jacob Smith: CLASSGENERATOR: I **wrote the begin method for the class generator -->** after adding packages manually in eclipse. My approach was to insert a begin method header after the constructor. This way, all the code that was in the constructor is part of the begin method. Note that the logic to generate keywords does not have to be modified, because it reads the begin method as a public method and generates a keyword for it (Page 18).

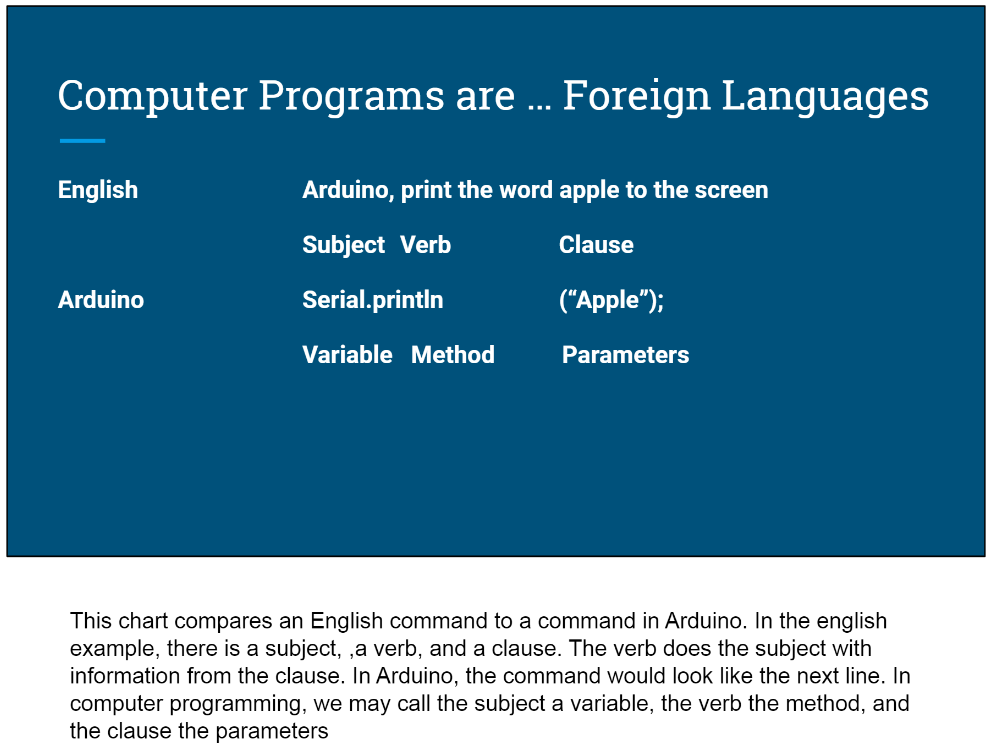
FUTURE WORK: I could add a feature which displays a warning if the begin method isn’t used, perhaps as an optional setting. In addition, it would be helpful to automatically include all instance variables as parameters in the begin method, as this saves the user the time of adding constructor logic manually. Also, the class generator puts gibberish into the setup and loop methods of the example files.

7/8/2019 12:04 PM: CLASS GEN:To learn how to add save as feature to the IDE, I Iooked at the Arduino IDE class files. Line 622 of Editor.java shows how the save as function is used (I traced function calls to sketch controller).

With this, I got the class maker toautomatically save as when a new class is created (sketchController save as method) -->which gives user a chance to put the sketch in the right folder and name it correctly.

Then, I looked into Editor Line 721 and traced that function call to arduino core/scr/cc/arduino/ contrubtions/libraries/libraryinstaller.





The above trace showed me that file objects are used for creating files for the Arduino IDE, so I used that approach in **automatically creating examples folder in class generation-->**

In addition, I Edited runArduino.bat batch script tohandle if the interface file is there or not (requires different responces)

WORKSHOP:I also wrote slides to introduce computer programming for the workshop. I am emphasizing a **langauge based approach -->**, because students already know a bit of grammar.

I then used the class generator to make ultrasonic sensor and display libraries for a demonstration. This is the first time that I used the generator to make a class I want to use. The biggest hang-ups where finding the example file because of read only formats and an extra #endif in the body file.

BACKGROUND: I found this library, which is very similar to my Arduino education library <https://github.com/arduino-libraries/EducationShield>. I will continue to develop my own library as it will let me test the class generator, and I will try to write other capabilities than this library.

Mbot (we have those kits too) <https://www.makeblock.com/steam-kits/mbot> and Blockly @arduino <https://technologiescollege.github.io/Blockly-at-rduino/>provide graphical interfaces for writing Arduino code

This Arduino build seems to integrate the above interfaces into the Arduino IDE <https://github.com/technologiescollege/arduino>.

(LEFT) Slide from Arduino education workshop showing grammer based approach (RIGHT) createExampleSketch method showing how class generator includes save as feature



Finally, Userlibrary class in aduino-core/src/processing/app/packages seems to be the logic of how third party libraries are displayed by the ARDUINO IDE. These background resources provide interesting applications of the class generation and workshop, like integrating a graphical interface to generate Arduino classes.

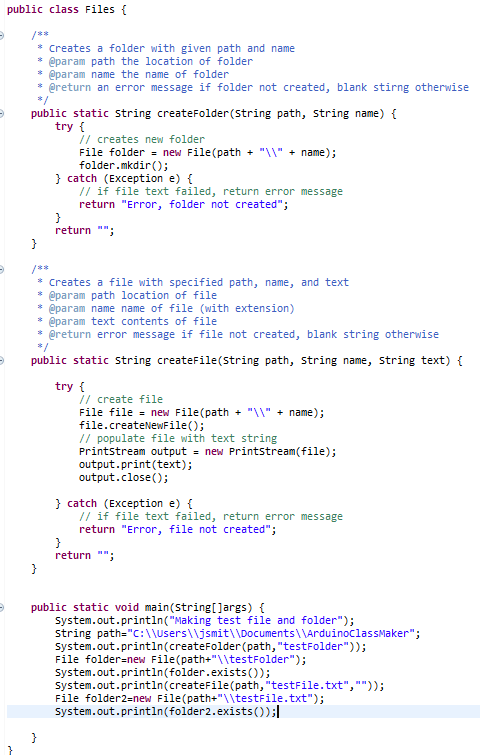
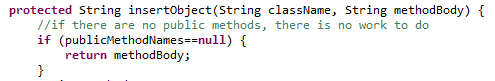
7/12/2019 12:55 PM: CLASS MAKER: EXAMPLESKETCH: I **fixed the gibberish printing problem from page 33** 🡪, which was caused by there being no public methods and the insertObjects methods therefore replacing a blank string with the className. This let me create a blank file test case set.

FILES: I also created the **Files class🡪**, which makes creating files and example folders much easier, although there is similar logic here to the ScriptEditor class.

With the Files class, I was able to create the **optionalClassFileMaker class** **(next page)**, which creates keywords, .development, .gtiignore, library.properties, and example files. Abstracting these behaviors from the interface class lets me verify their behavior with unit tests.

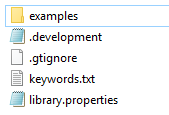
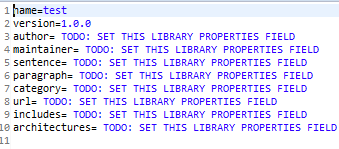
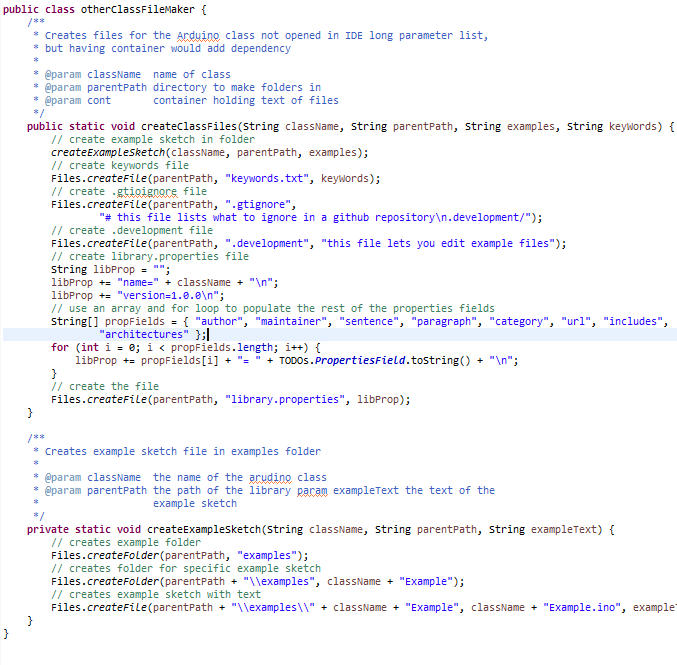
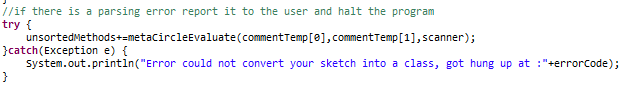
Also, this class means that the Arduino IDE doesn’t have to open other file types, which should make the pull request easier.

The generate **library.properties file (next page)** includes ToDos or fields that the class generator does not know like the maintainer.



[Top] getPublicMethodNames and insertObject methods of ArduinoClassExamplSketch.java showing handling if public methods are blank.

[Bottom] Files Class



With all of the work on files, I learned how to create temporary files with Junit.

Explanatory Blog Post:<https://garygregory.wordpress.com/2010/01/20/junit-tip-use-rules-to-manage-temporary-files-and-folders/>

Checking if file was created: [0]<https://stackoverflow.com/questions/18397166/junit-test-case-to-check-if-file-was-created>

This is the first time I’ve had to test whether a file was created, and temporary files allow me to not build up my project after repeated unit tests.

BACKGROUND: While there aren’t any other Arduino Class generators, there seem to be C++ class generators, see these examples:

[1]<https://www.codeproject.com/Articles/671306/A-Cplusplus-Class-Code-Generator>

[2]<https://www.lazycplusplus.com/>

[3] http://ring

REFACTORING: I removed a stray end if and changed ArduinoClass hardcoded enum to files testing, which saves on java code, is easier to format, and is consistent with other tests. I also replaced string literal comparisons and avoided long parameter lists lang.sourceforge.net/doc/codegenerator.html, Daniel helped with reassigning parameter**. I also added a try catch** 🡪 so the sketchParser class can handle formatting errors.

FUTURE WORK: Verify behavior of IDE and console printing and come up with some way of handling TODOS, perhaps an interactive format

[Left] try catch statement in SketchParser to handle parsing errors

[Top] generated library.properties file by otherClassFilesMaker class and folder structure

[Bottom] otherClassFile maker class

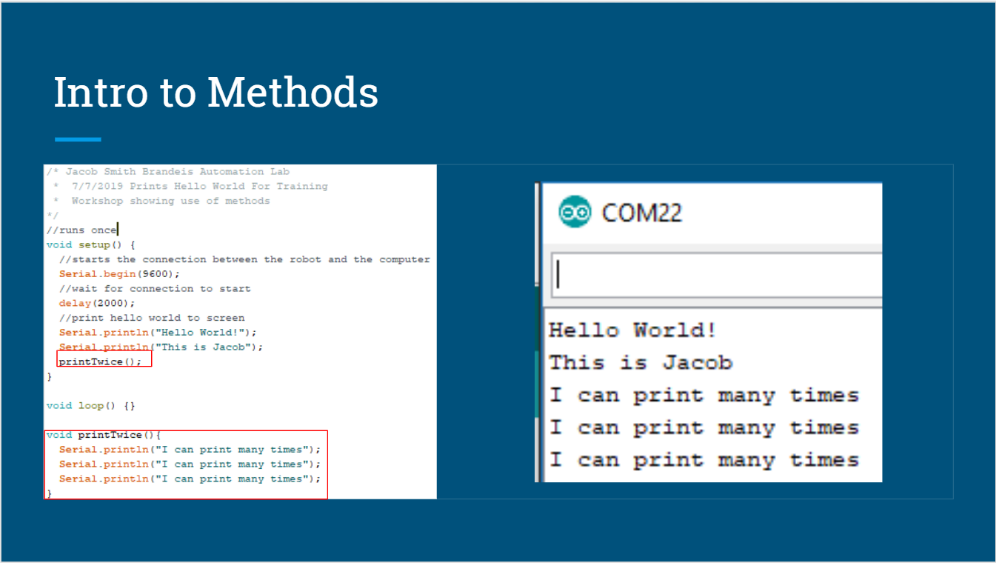
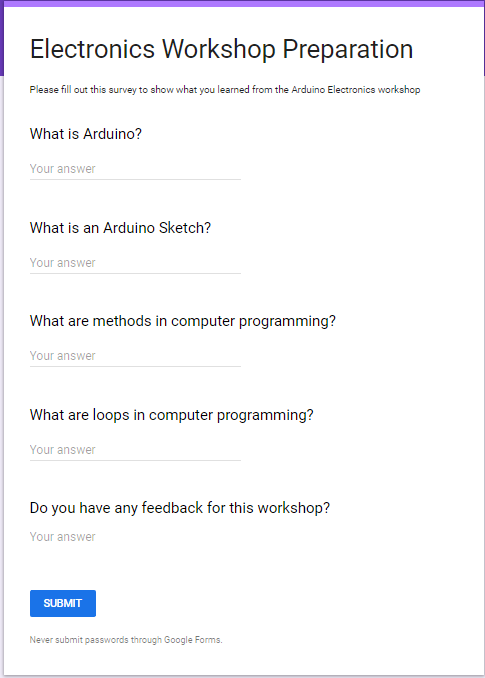
WORKSHOP: Yesterday, I added explanations of Arduino Sketches, **methods 🡪** and loops. I also added **a quiz** 🡪 . Then, I gave the tutorial to Fontaine, it took about 30 minutes, but I would make it an hour for students new to programming. His feedback was to replace text printing with an LED to make it more electronics based. After talking to Tim and Daniel, the next step is for me to create a hardware module for my workshop, add the example projects, and try out the tutorial on someone who hasn’t programmed before. The advantage of the multiple module approach is that it gives me time to focus on each element, and people who already know one module can skip it.

The electronics workshop will focus on analog and digital signals and Morse code, potentiometer, digital output, and PWM examples.

BACKGROUND: Also, I found this library for an Arduino workshop

[3]<https://github.com/arduino/EduIntro>

Publicity: Also this week, I met with Tim and Ian to go over publicity for my workshop. The next steps are for me to make a flyer for it and meet with the science librarians. The workshop will probably be weekly, with an electronics section, a computing section, and a project section. The workshop will be capped at 3 students a week. Reserving rooms is also a long term consideration.



[LEFT] workshop quiz

[RIGHT] WORKSHOP LOOPS SLIDE

|  |  |  |  |
| --- | --- | --- | --- |
| Resource | useful chapter | title | has ... |
| **Arduino Projects** |  |  |  |
| [Arduino By Example](https://learning.oreilly.com/library/view/arduino-by-example/9781785289088/) |  |  |  |
| [Arduino Electronics Blueprints](https://learning.oreilly.com/library/view/arduino-electronics-blueprints/9781784393601/?ar) |  |  |  |
| [Practical Arduino Engineering](https://learning.oreilly.com/library/view/practical-arduino-engineering/9781430238850/) |  |  |  |
| [Beginning Arduino](https://learning.oreilly.com/library/view/practical-arduino-engineering/9781430238850/) | 6 | Interlude: Home Engineering | voltage divider |
| **Arduino Resources** |  |  |  |
| [Arduino Development Cookbook](https://learning.oreilly.com/library/view/arduino-development-cookbook/9781783982943/) | 5 | Motor Control | Controlling Speed with PWM |
| [Arduino Cookbook](https://learning.oreilly.com/library/view/arduino-cookbook-2nd/9781449321185/) | 5 | Simple Digital and Analog Input |  |
| [Arduino Essentials](https://learning.oreilly.com/library/view/arduino-essentials/9781784398569/) | 3,4 | Interact Digital,Analog Outputs |  |
| [Getting Started with Arduino](https://learning.oreilly.com/library/view/getting-started-with/9781449316358/) | 5,Appendix C | Advanced Input/Output, Arduino Quick Reference | PWM diagram |
| Make: Getting Started with Intel Galileo | Page 44 | Analog/Digital |  |
| **General Electronics** |  |  |  |
| [Voltage Divider](https://en.wikipedia.org/wiki/Voltage_divider) |  |  |  |
| [Quick Dirty Introduction to Electronics](http://douglasrepetto.com/classes/basic_sensors_mC_electronics/) |  |  | Electronics Tutorial for beginners using Arduino |
| [Olin’s Electronics Course](http://faculty.olin.edu/bstorey/isim.pdf) |  |  |  |
| [Practical Electronics Handbook](https://ebookcentral.proquest.com/lib/brandeis-ebooks/detail.action?docID=285825) | 16 | Digital Signal Processing |  |
| [Electricity Electronics Fundamentals](https://ebookcentral.proquest.com/lib/brandeis-ebooks/detail.action?docID=3239060) | 6 | Computers and MicroProcessors | diagram of computer |
| [Electronics Explained](https://learning.oreilly.com/library/view/electronics-explained/9781856177009/) | 4,5,6 | Linear/Analog,Digital,MicroComputers |  |
| [Electronics Simplified](https://learning.oreilly.com/library/view/electronics-simplified-3rd/9780080970639/) | 1 | Electricity, Waves, Pulses | diagram of water flow, introduction |
| Make: More Electronics Charles Plat |  |  | electronics projects |
| **Scientific Application** |  |  |  |
| [Arduino Control of a Pulsatile Flow Rig](https://www.sciencedirect.com/science/article/abs/pii/S1350453317302618) |  |  |  |
| **Educational Applications** |  |  |  |
| [First Exposure to Arduino Through Peer Coaching](https://www.sciencedirect.com/science/article/pii/S0747563217304193) |  |  |  |
| [Where does Arduino’s Power Come From?](http://joltida.org/index.php/joltida/article/view/44/109) |  |  |  |
| [Basic Arduino Programming for Training High School Students](http://jurnal.narotama.ac.id/index.php/scj/article/view/456) |  |  |  |
| **Further Reading with a Specific Focus** |  |  |  |
| [Basic Electronics Math](https://learning.oreilly.com/library/view/basic-electronics-math/9780750697279/) |  |  | Math Resource |
| [Beginning C for Arduino](https://learning.oreilly.com/library/view/beginning-c-for/9781484209400/?ar) | 14 | Object Oriented Programming | Programming Resource |
| [Arduino Projects Book](https://bastiaanvanhengel.files.wordpress.com/2016/06/arduino_projects_book.pdf) |  |  | Projects Resource |
| [Arduino A Technical Reference](https://learning.oreilly.com/library/view/arduino-a-technical/9781491934319/) |  |  | Components Resource |
| **Needed Books** |  |  |  |
| [Amazon Wishlist](https://www.amazon.com/hz/wishlist/dl/invite/aYmmbbg?ref_=wl_share) |  |  |  |