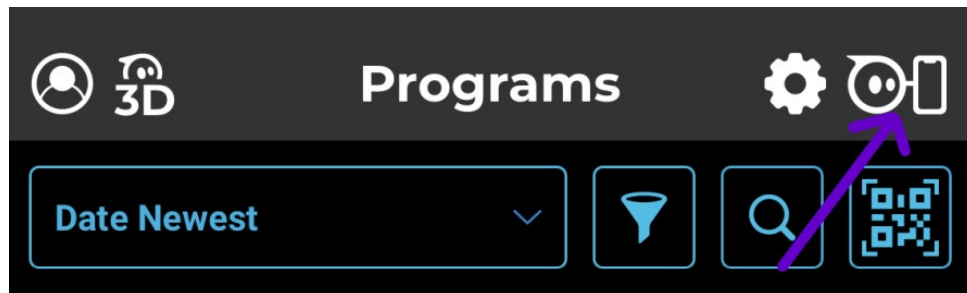


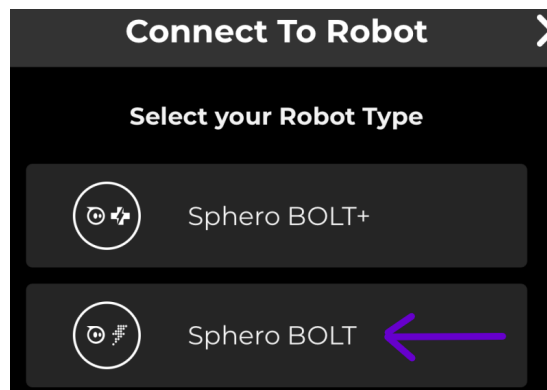
**GEECC Camp Activity Instructions**  
Computer Science

## Sphero Bolt Introduction (round robots, morning activity)

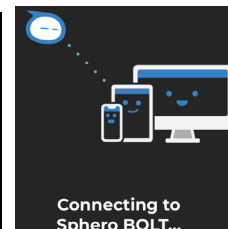
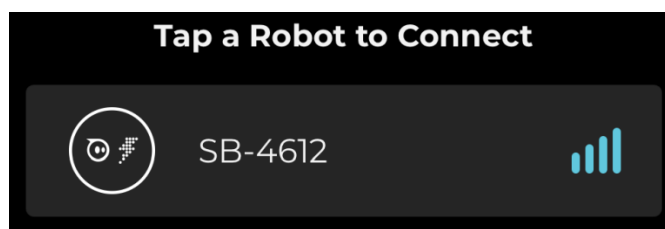
1. Form groups of two.
2. Navigate to <https://github.com/GSU-CS-Dept/EEA-Robot-Code/> in a web browser.
3. Open the GEECC CS Instructions.pdf file.
4. Retrieve a Sphero Bolt, a charging cradle, and the connected cable.
5. Plug the cable into a computer.
6. Place the cradle and the robot on top of the computer.
7. **If you are told that you have Bluetooth, skip to the Sphero Bolt with Bluetooth instructions. If you do not, continue to the next step.**
8. Download the Sphero EDU app on either Android or iPhone and open the app after installation is complete.
9. Open the Sphero EDU app, then click the top right icon. We need to connect to our Bolt to be able to program it.



10. Choose the Bolt, not the Bolt+.



11. Place your phone close to your Bolt and choose the one with the highest signal strength. You will see some connection confirmations. Your Bolt may also display a pattern to help you locate it.



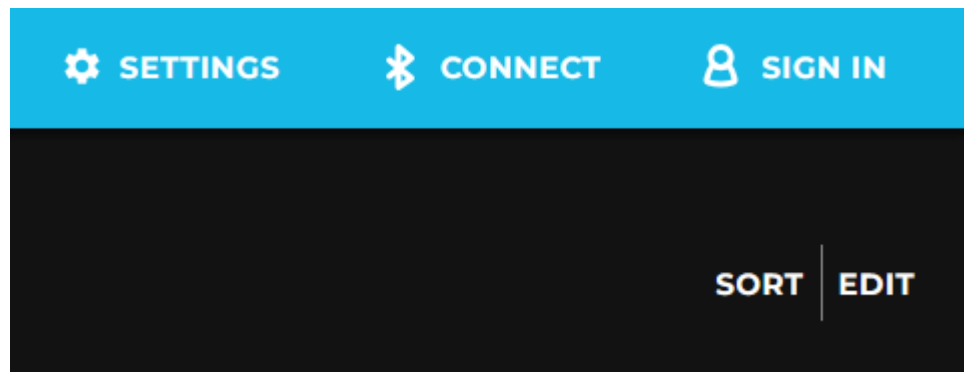
12. Click on the Drive in the bottom right corner and play around with driving the robot.
13. Once you have tested driving the robot, move on to the Sphero Bolt Programming.

## Sphero Bolt Programming

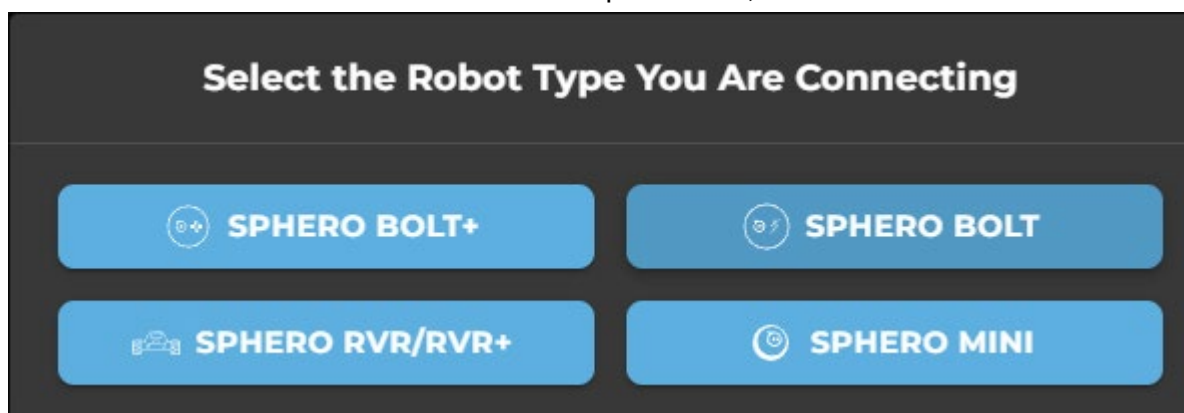
14. Click on Create Program. Take the offered tour, it will help you get oriented with the block programming. Give your program a name.
15. Place the Bolt on the floor (outside the classroom may be easier). You can use the Aim button next to Start to make sure you are connected to the correct Bolt.
16. Click the light icon to change the color of your LED to something different than the default. This will help you identify your robot.
17. Hit Start at the top to start executing code, Stop to stop the code executing.
18. Start experimenting with the code to navigate the obstacle course that will shown to you. Play around with the code until your robot can navigate the course. Tap on any numerical values to change them or drag new code actions from the bottom bar.
19. Leave this document open, after lunch you will want to move to the Smart Car section on page 6.

## Sphero Bolt with Bluetooth

1. Navigate to <https://edu.sphero.com/code> in the Chrome web browser.
2. Click on Connect.

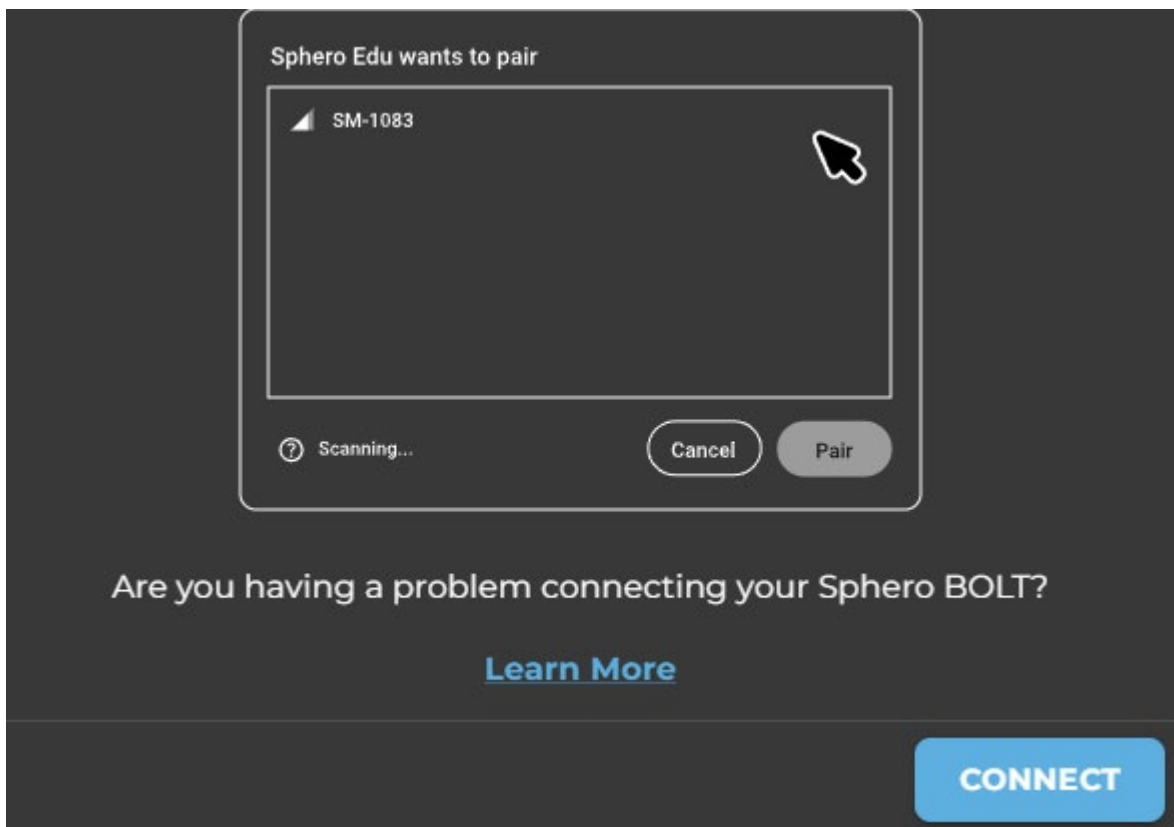


3. Make sure to choose the Sphero Bolt, not the Bolt+.

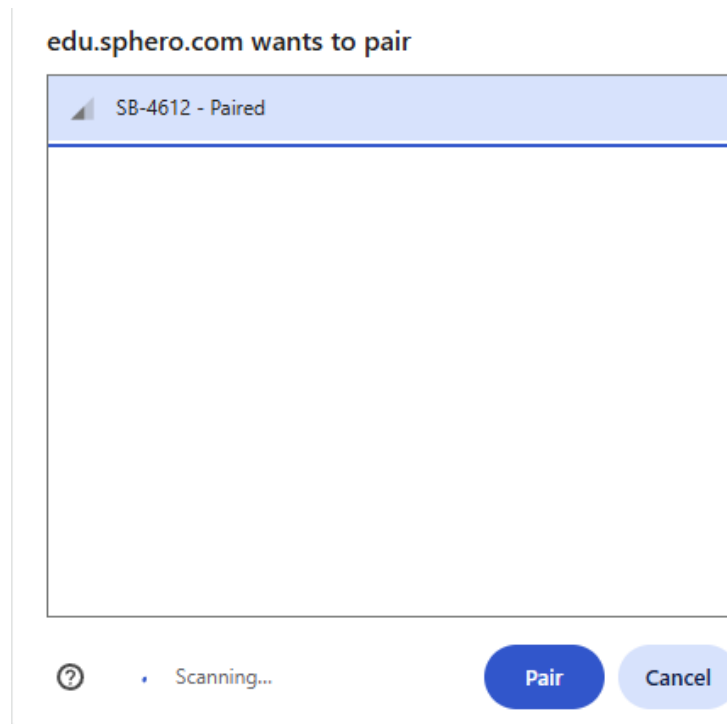


4. Move to next page.

5. Click Connect

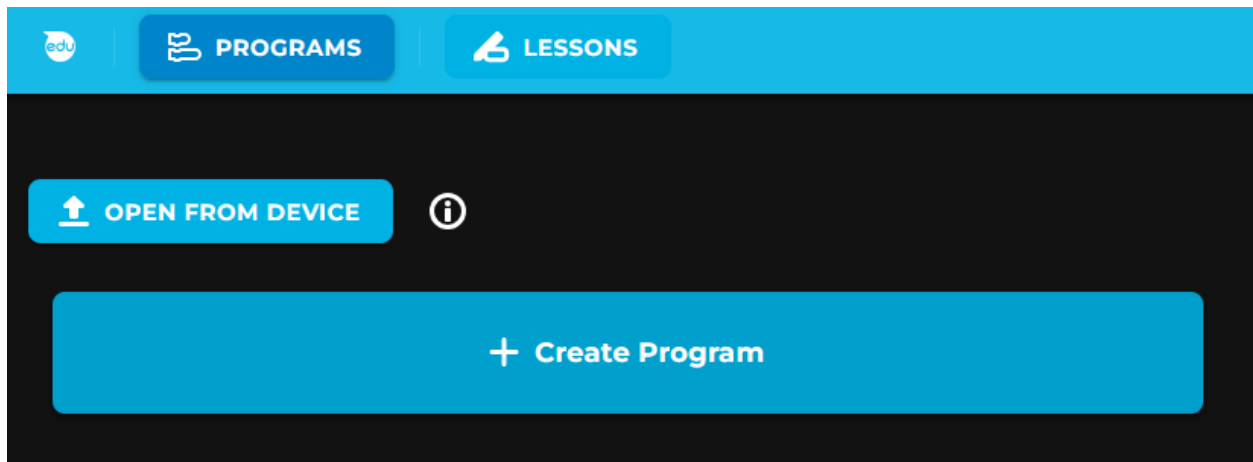


6. Connect to your Sphero Bolt (it is probably the one at the top of the list).



7. Move to next page.

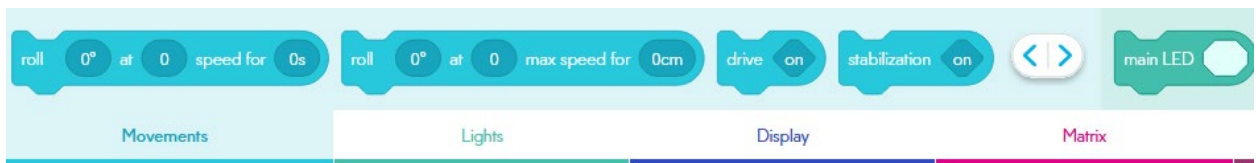
8. Click on Create Program



9. Give your program a name and make sure the chosen options match the following screenshot (see next page), then click Create.

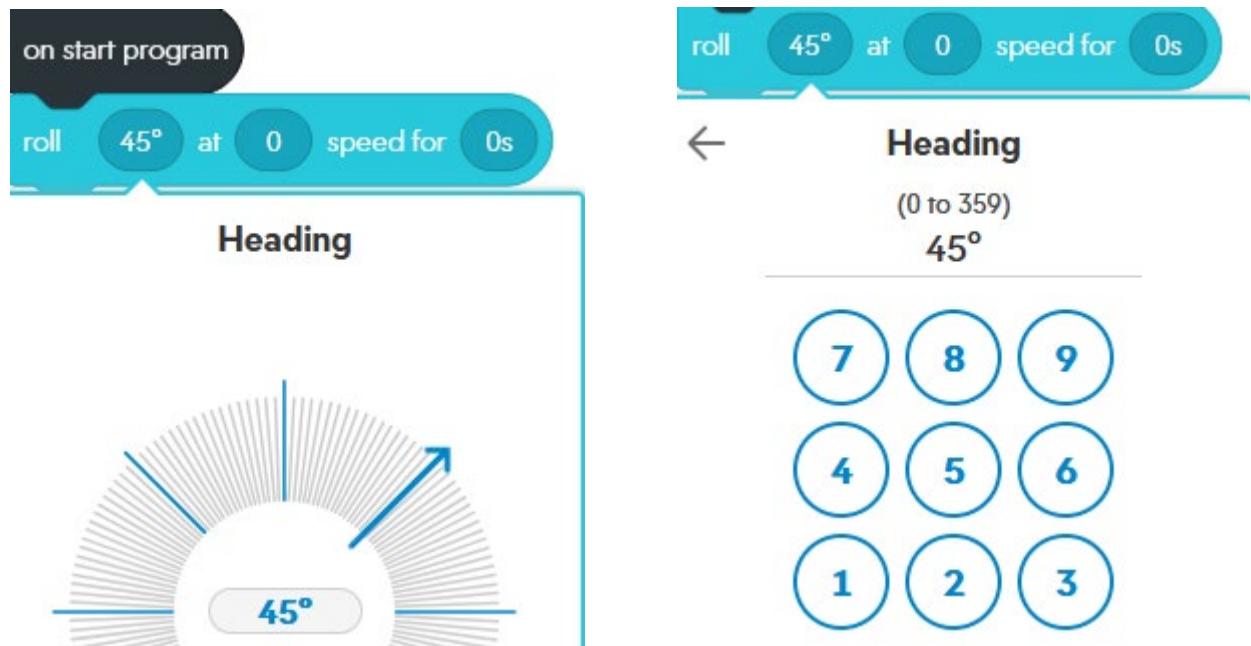
The image shows a form for creating a new program. It has a dark grey background. The first section is 'Program Name' with a text input field containing 'Untitled Program'. The second section is 'Choose Program Type' with three buttons: 'DRAW', 'BLOCKS' (which is highlighted in blue), and 'TEXT'. The third section is 'Choose Compatible Robots' with four buttons: 'SPHERO BOLT+', 'SPHERO BOLT' (highlighted in blue), 'SPHERO RVR/RVR+', and 'SPHERO MINI'.

10. Along the bottom of the page are various actions you can do on the bot. For each section, if you click on the left/right arrows, you can see all available actions. Either use the tilt function of the mouse's scroll wheel to scroll through all available options or hold down the Shift key and scroll the mouse wheel up or down.



11. Move to next page.

12. You can drag those items onto the “on start program” section like a puzzle piece, then choose options by clicking on each available section (see next page). This can be done visually by clicking on the image, or by keyboard by clicking on the number first which will bring up a number pad (see next page).



13. If you need to delete any blocks, grab them and drag them to the bottom of the screen then click the trash can.



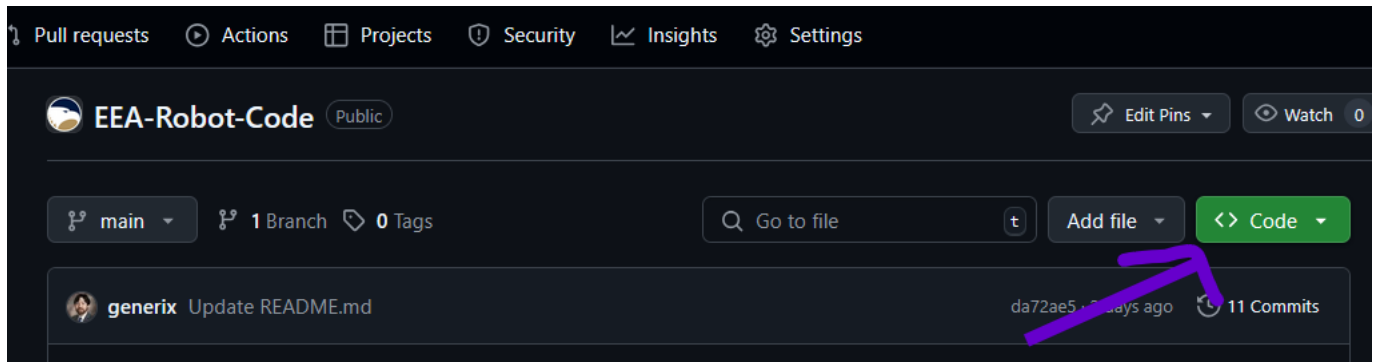
14. When you are ready to run the program, place the Sphero Bolt on the ground, then click the Start button on the Sphero Edu website.

## Smart Car Assembly (after lunch)

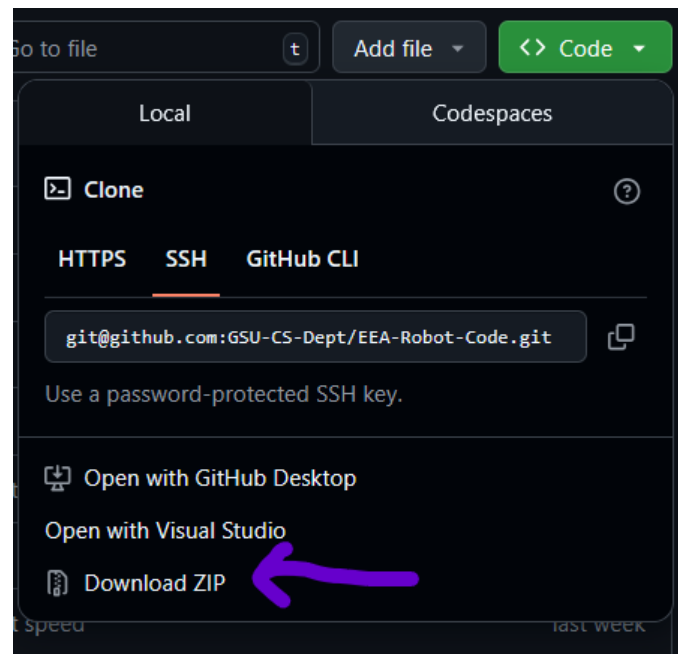
1. Form groups of three.
2. Navigate to <https://github.com/GSU-CS-Dept/EEA-Robot-Code/> in a web browser.
3. Open the Smart Car Box, remove parts from the internal boxes, but do not start opening packages until the instructions instruct you to.
4. Click on the [Assembly Video](#) link and use that along with the paper instructions to assemble the car. The paper instructions are also included in the GitHub repository linked in Step 2 in PDF format (Filename: ELEGOO Smart Robot Car Kit V3.0 Assembly Tutorial--20190314.pdf).
5. Pay attention to orientation of parts, screws used (see the paper instructions for details), and make sure to tighten any parts using the provided needle nose pliers to hold the nuts.
6. If you get done assembling before the allotted time, download the ElegooKit app to your phone. If the allotted time has elapsed, move to the Smart Car Code and Testing section.
7. Connect to your robot using WiFi. It will usually be the first in the list and should show up as ELEGOO-XXXXXXXXXX with numbers and letters in place of the Xs.
8. Experiment with the different modes (line tracking, obstacle avoidance, follow, FPV).

## Smart Car Code and Testing

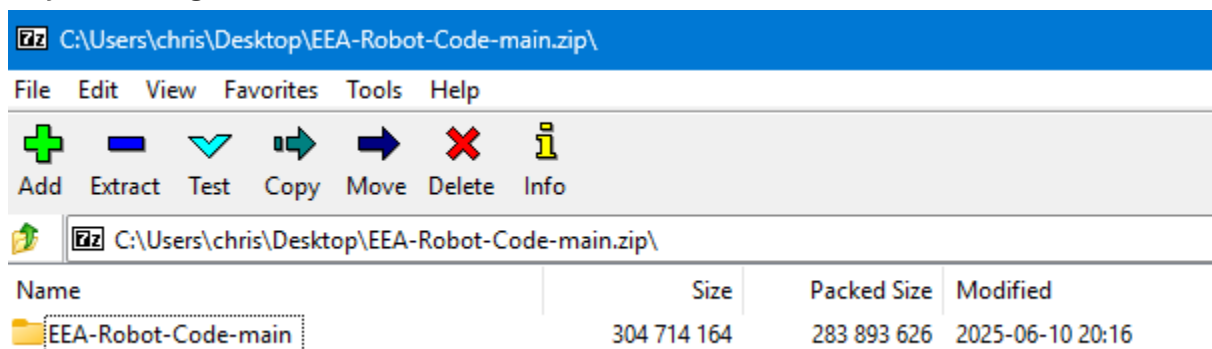
1. Stay with your assembly group.
2. Navigate to <https://github.com/GSU-CS-Dept/EEA-Robot-Code/> in a web browser.
3. Click on the green Code button near the top of the screen.



4. Choose Download .zip. This file will contain all of the original tutorials provided by Elegoo and custom code that we will use in our robot demo today. Feel free to look through all the files. The 02 Main Program folder contains the original code that was used in the Elegoo Kit app.



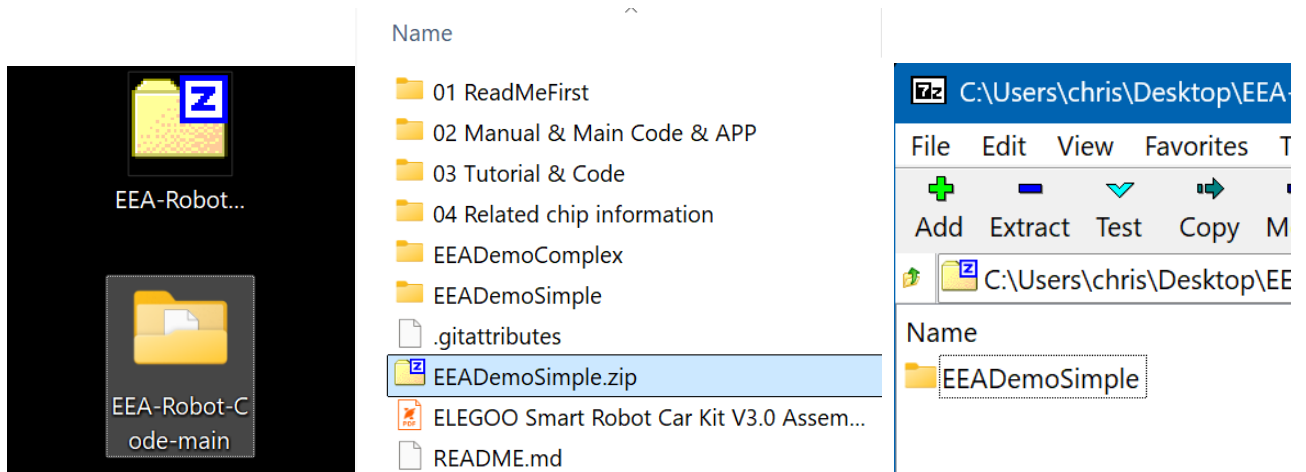
- When the file is done downloading, double-click the file to open the zip archive. The location of file may vary according to browser, use the Downloads section inside the browser to locate it.



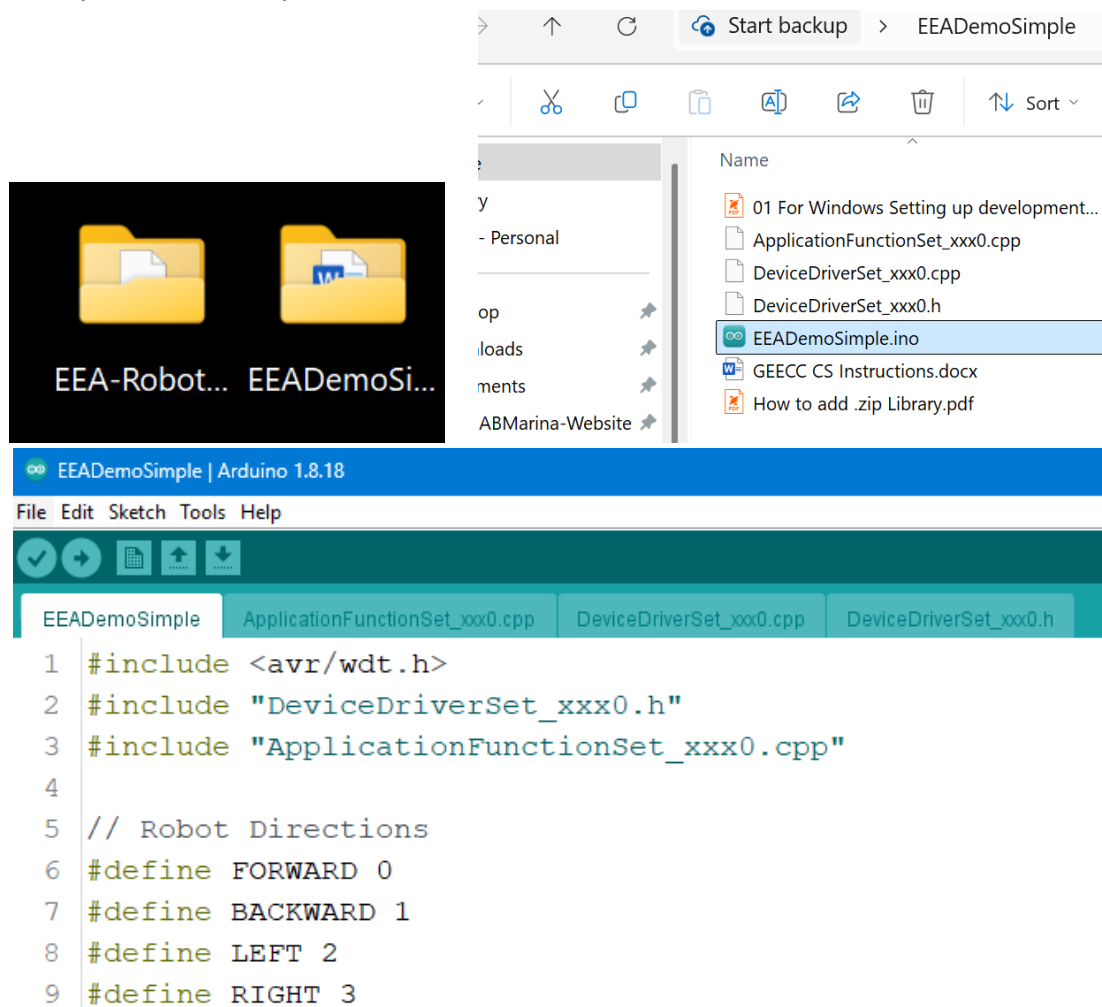
- Drag the EEA-Robot-Code-main folder to the Desktop.



- Open the folder, then open the EEADemoSimple.zip file and drag the EEADemoSimple folder to the Desktop.

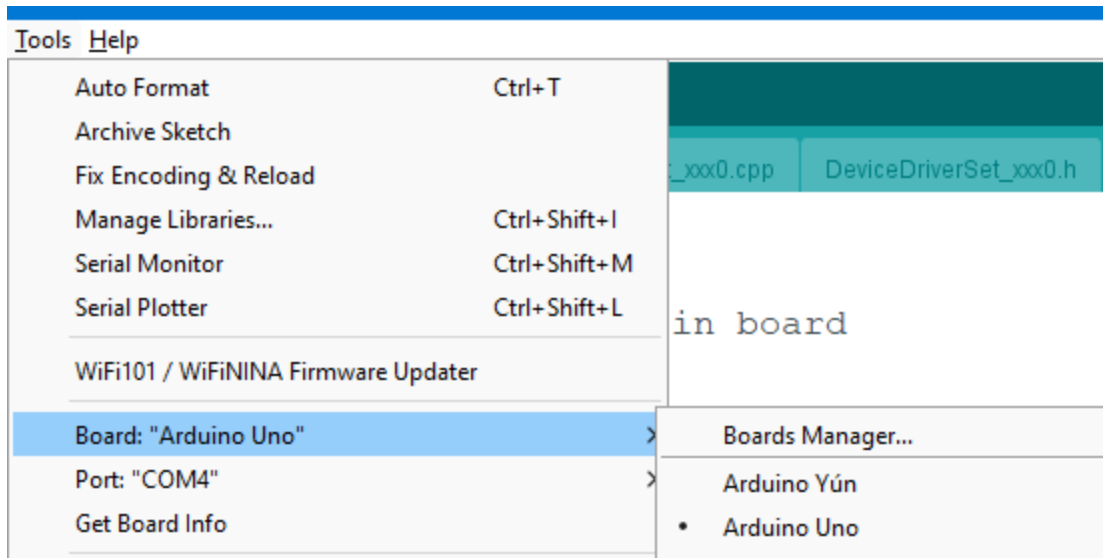


8. Double click the EEADemoSimple folder on the Desktop, then double click the EEADemoSimple.ino file to open the Arduino IDE.

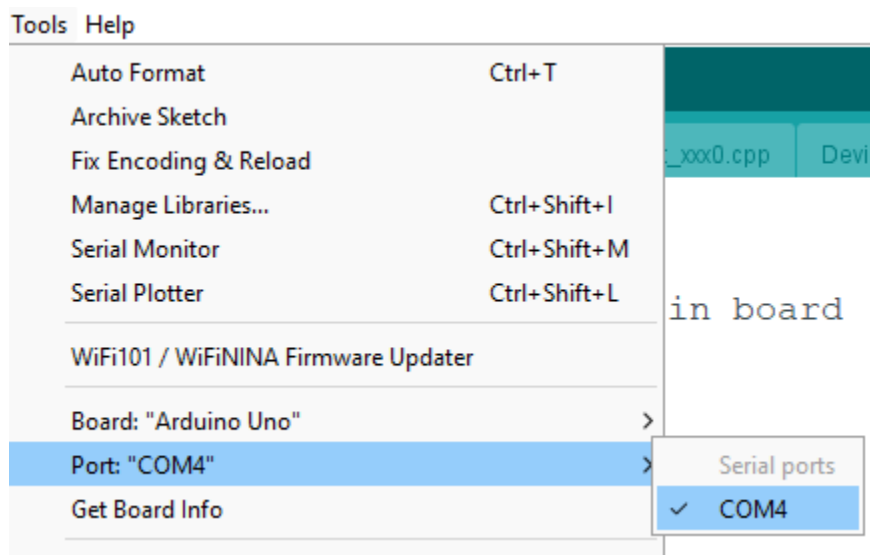


9. Check that the Board setting is set to Arduino Uno by checking Tools -> Board.





10. Set the COM Port to the highest COM listed on your computer at Tools -> Port:.



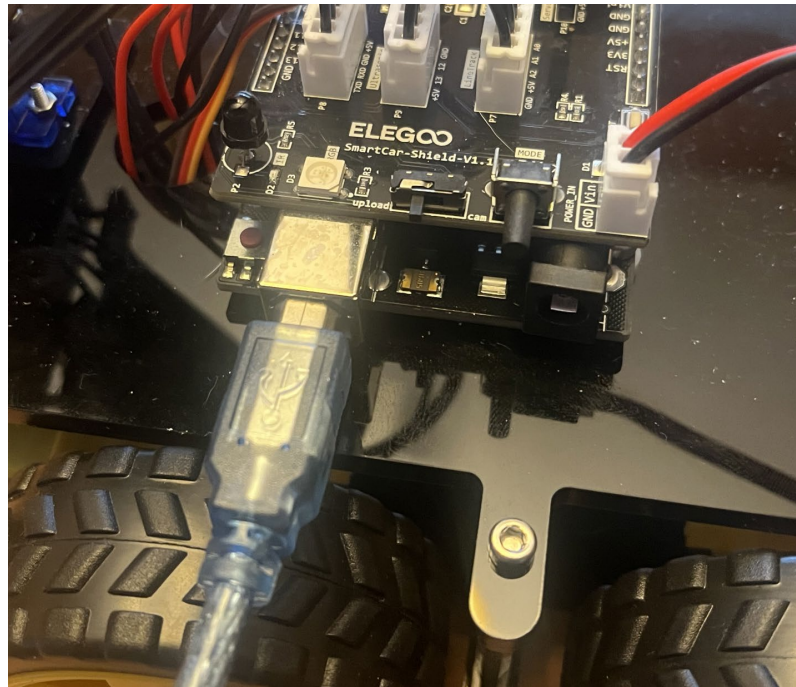
11. Examine the Arduino code in the file and note how the robot is controlled by C++ code in the setup() function. Always tell the robot to STOP at the end of the function to prevent it from moving after executing its code.

```
// set movement speed (150 @ minimum, movements get less precise the slower it goes, 255 is optimal)
uint8_t motorSpeed = 255; // change this value to speed up or slow down car, min 0, max of 255

// move forward
ApplicationFunctionSet_SmartRobotCarMotionControl(FORWARD /*direction*/, motorSpeed /*speed*/);
delay(2500); // wait

// turn
ApplicationFunctionSet_SmartRobotCarMotionControl(RIGHT /*direction*/, motorSpeed /*speed*/);
delay(3000); // wait
```

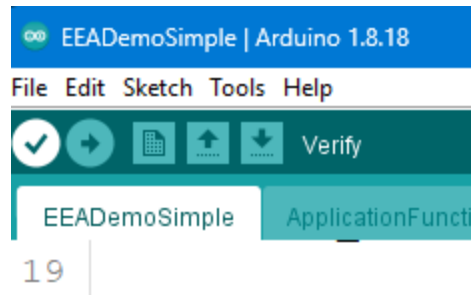
12. Plug the non-charging USB cable into the silver USB port on the mainboard of the Smart Car.



13. Locate the switch next to the USB port and make sure it is moved to the Upload side and NOT the Cam Position.



14. First click the Verify button in the Arduino IDE. If there are no errors, this will complete successfully. If there are no errors, move to the next step. If there are errors, ask for help.



15. Pick the car up off the ground, then click the Execute button. This will load the code onto your robot. This may involve the wheels moving. Once the car quits moving, unplug the USB cable.
16. Place the car on the ground, then turn the battery on. Whatever program was last loaded on the robot will now execute.
17. Try experimenting with the code directions and delay times (measured in milliseconds) and re-verify and re-upload the code.