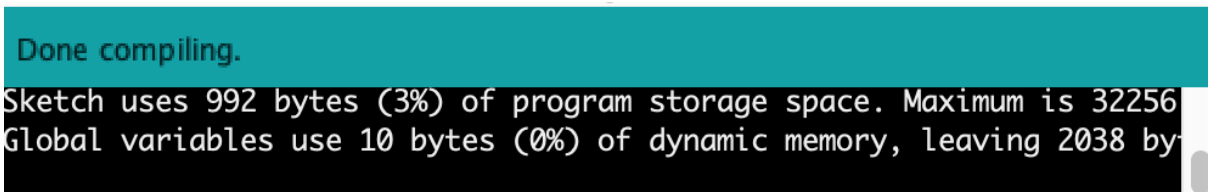
1. To verify your syntax is ok in your code (no typos) you can push the TICK button.

NOTE: If it's a new project, this step might also open up a SAVE DIALOG BOX, allowing you to save your project. Give it a suitable name if you want to bring it back later.

If your code is good it looks like this at the bottom of the window.

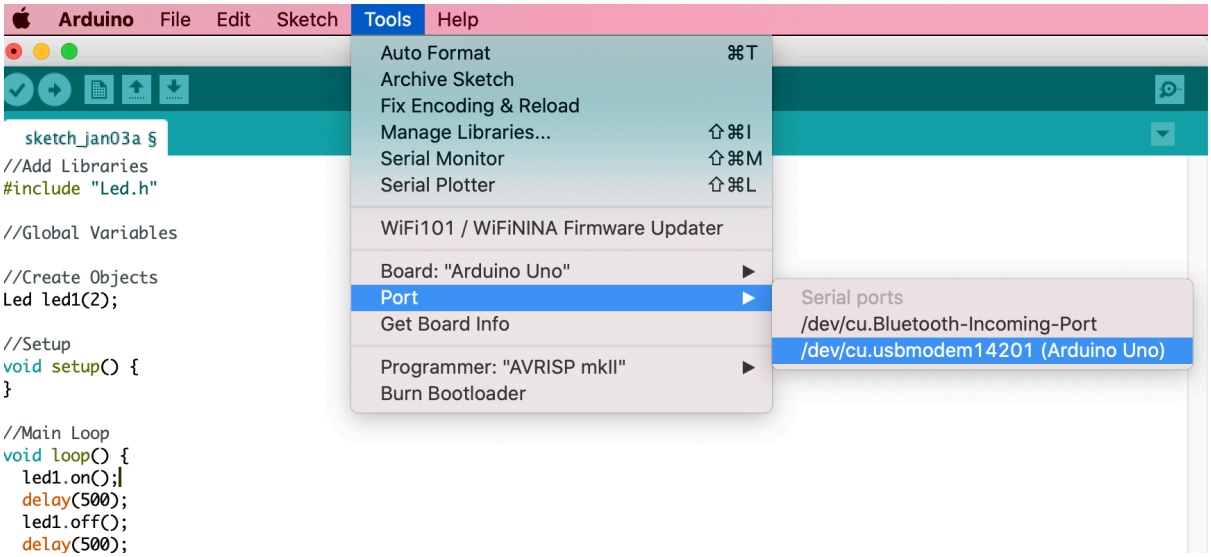


But if you have a typo, it will look like this (in example below, a semi-colon is left out on first line inside main loop.

Note the orange warning at the bottom. Also, the explanation is that a semi-colon is needed. Note, in the code, the line where the error was encountered is highlighted pink. Typically, the error is the line above the pink line, because only when you tried typing the next command, to delay, before finishing the previous command with a semi-colon, did the computer note an issue. There should be a semi-colon after led1.on() on the line above.

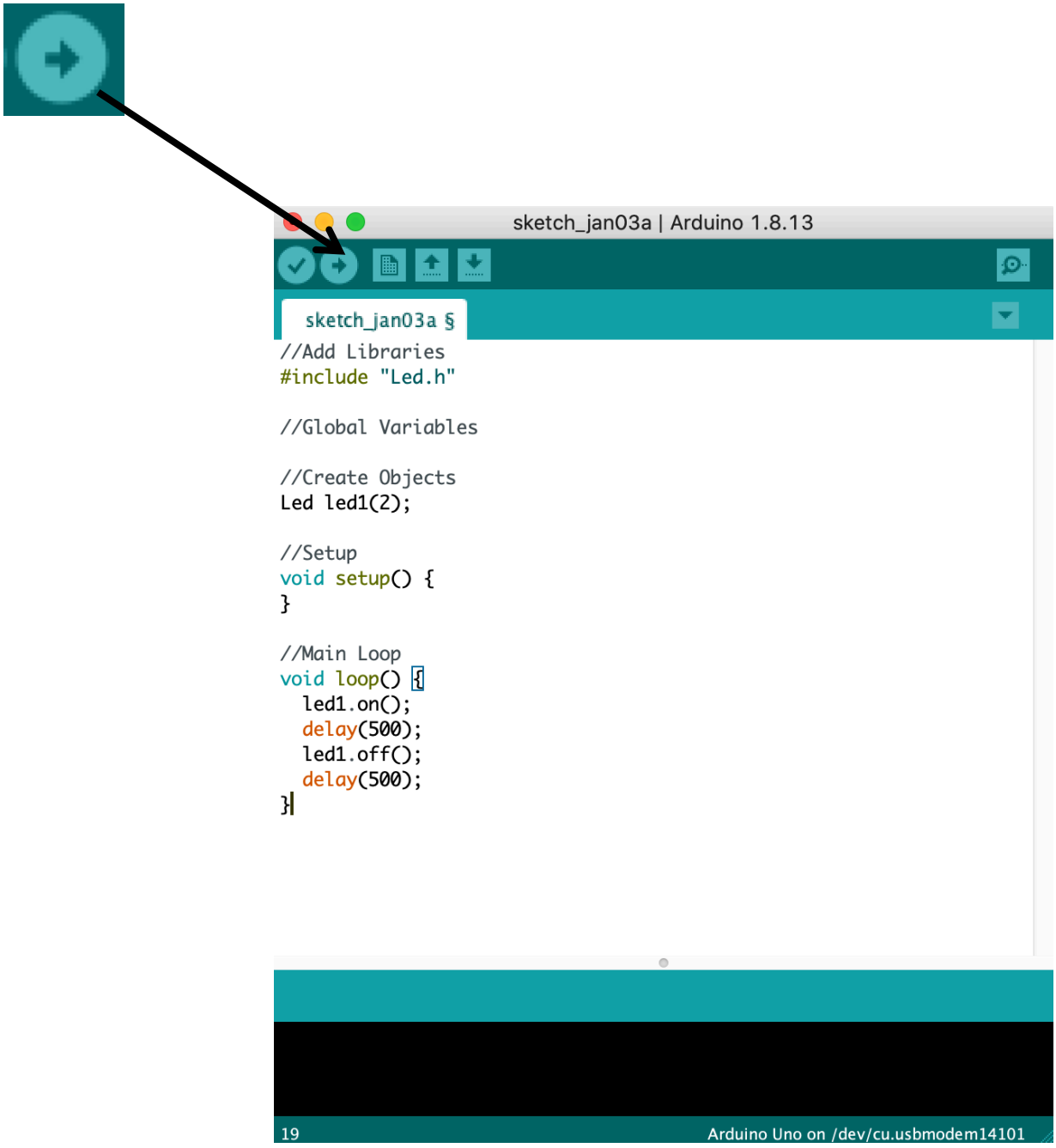
Once you are happy with your code, and are confident it has no typos in it, we are ready to upload it.

You'll need to let the laptop know which USB Port to send the data through. You can do this by going to TOOLS / Port / and then selecting the option that has Arduino Uno at the end.



* Sometimes, you go to Tools / Port, and there is no option with Arduino in it. In this case, check to ensure you have actually connected the Arduino to the laptop with a USB cable/adaptor. Once you connect it, wait a few seconds, then click off, then go back and try again with Tools / Port and hopefully the Arduino option will be there.

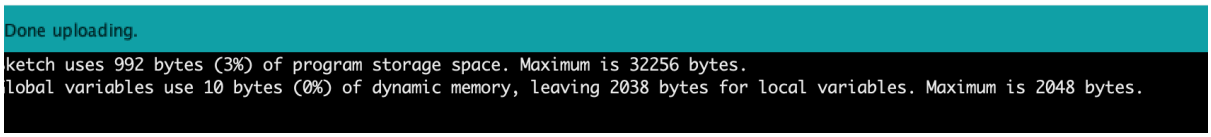
Finally, it’s time to upload using the arrow button.



Note, the other options you see are NEW , OPEN, SAVE, which can all be accessed via the FILE tab anyway.

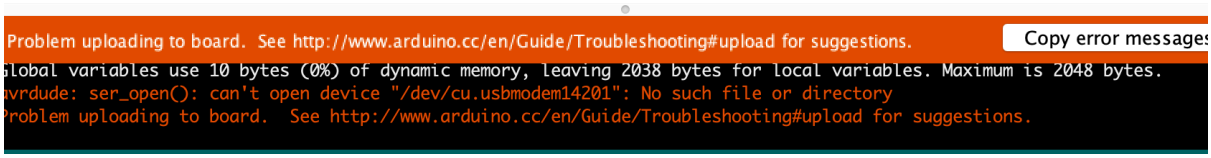
NEW OPEN. SAVE

Once done, it should give you some status down the bottom.



If there is an error, you will get an error message.

e.g. If you try UPLOADING without the Arduino connected and the port selected, you'll get this error message:



Note, the orange writing in black says 'can't open device'.

The other issue that can occur is that you get the Arduino being 'busy' because the program running on it doesn't pause enough to let a new program upload. Typically trying to reset first, then quickly uploading fixes this.