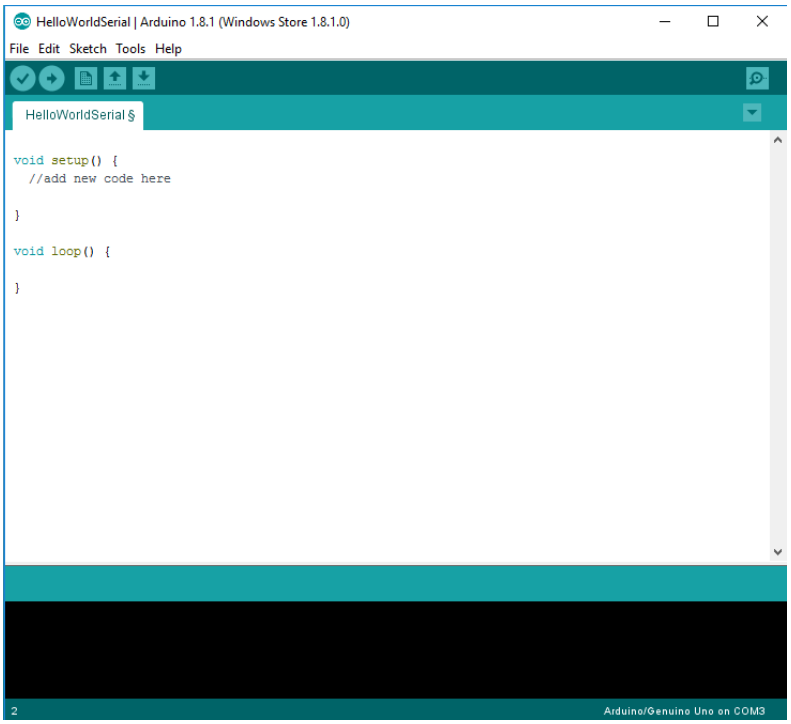
Intro to Arduino IDE and Hello World

Arduino IDE



The Arduino Integrated Development Environment (IDE) is a suite of open source tools used to write, load, and debug code for the Arduino. An Arduino program is refrerred to as a 'sketch.' Open up the provided sketch HelloWorldSerial.ino. Make sure your Arduino is connected to the computer using the USB cable.

This sketch, as currently written, doesn't actually *do* anything; but that doesn't mean we can't compile, load, and run it. Start by pressing the verify button. This compiles the code; which means it takes the human readable source code, and turns it into machine executable binary file. During this step, we will discover if our code contains any errors; if errors exist, the compilation will fail.

Next, use the upload button to send this code to the Arduino. You should see the process complete and the IDE will report that the upload is done. It should be noted that pressing upload automatically compiles your code before uploading, you don't need ot press verify first if you are ready to upload.

Hello World

It's traditional for programmers encountering a new language to write a very simple program, called a 'hello world' program in which they confirm that their system is succefully configured to run their code. This program typically just prints the phrase 'hello world' to the screen. Since we are working with an Arduino, wich does not have a screen, our Hello World program will be slightly more complicated. It will print the phrase "Hello World" to the serial port, and our computers will then print it to the screen.

Locate the comment that shows us where in the sketch to start adding new code. Start by adding the following line *exactly* as written:

```
Serial.begin(9600);
```

This line tells the Arduino to start a serial connection with the computer, at a speed of 9600 baud.

Go down to the next line and write:

```
while(!Serial) {}
```

We will talk more about control structures and loops later. For now, it should suffice ot say that this line of code waits for the serial connection to be established before proceeding

```
Serial.println("Hello World");
```

This line writes the text Hello World to the serial line; if everything goes well, we should see it appear on the serial monitor when we run this program.

Save your progress then press the upload button. When the upload completes, open the serial monitor. Make sure that the baud rate for the monitor matches the baud rate we defined in our program (9600). You should see the text 'Hello World' appear on screen.