Arduino IDE Installation Guide
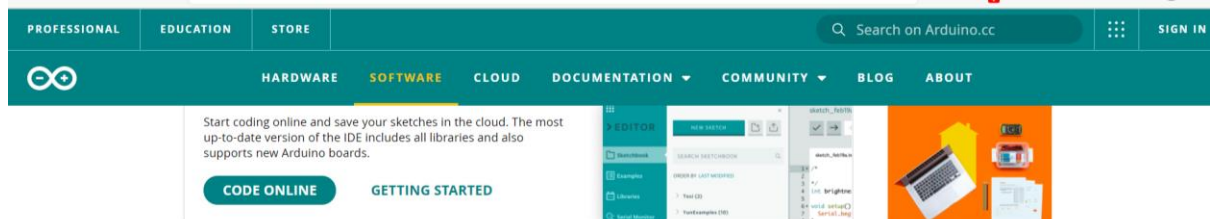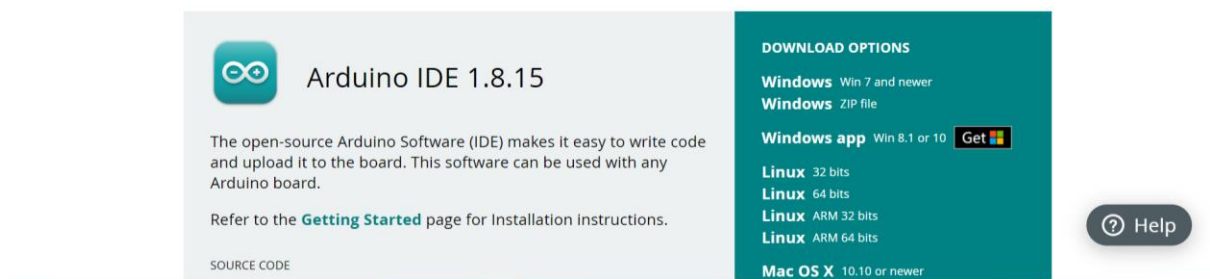
**For Windows:**

Link:

http://arduino.cc/en/Main/Software

Steps:

- Download the correct and latest version of Arduino IDE software from the Arduino website according to your PC's specifications. It is available for both 32-bit and 64-bit systems.
- After downloading the IDE software, open the .exe file. Then proceed with the installation and please wait for the driver installation process to finish.
- Click on the 'I Agree' button on the license agreement.
- Select all the components to install and click on the 'Next' button.
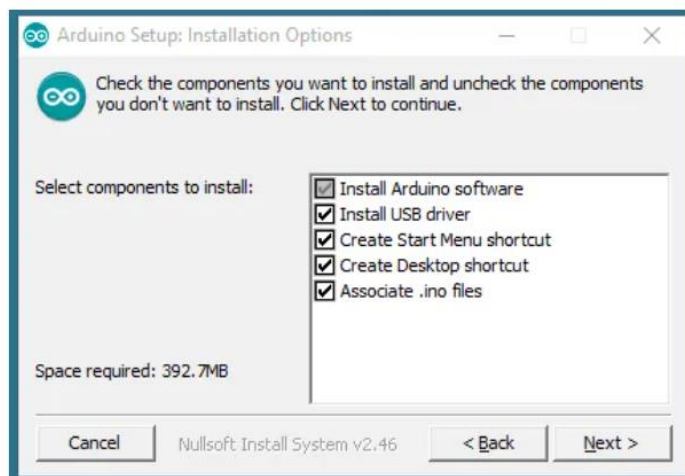
- Next, select the location in which you want the program to be installed by entering the path or by browsing the path. It's better to keep the default one.
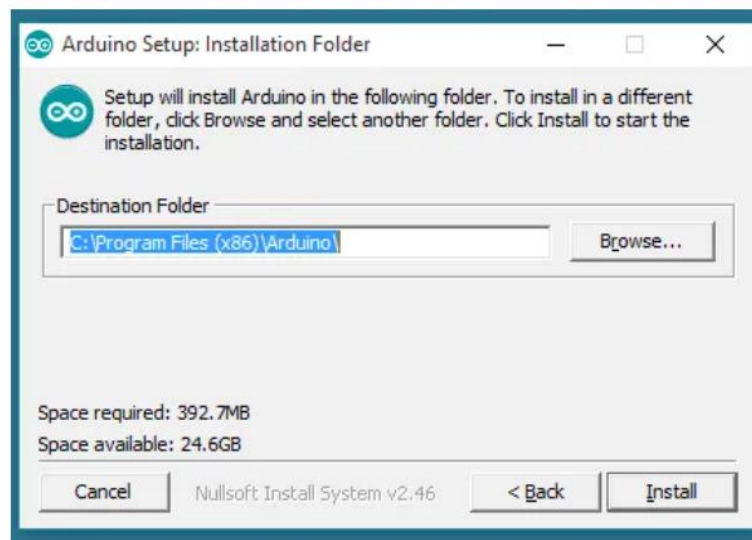
- After installation completes, click on the 'close' button. The setup window will extract and install all the required files.
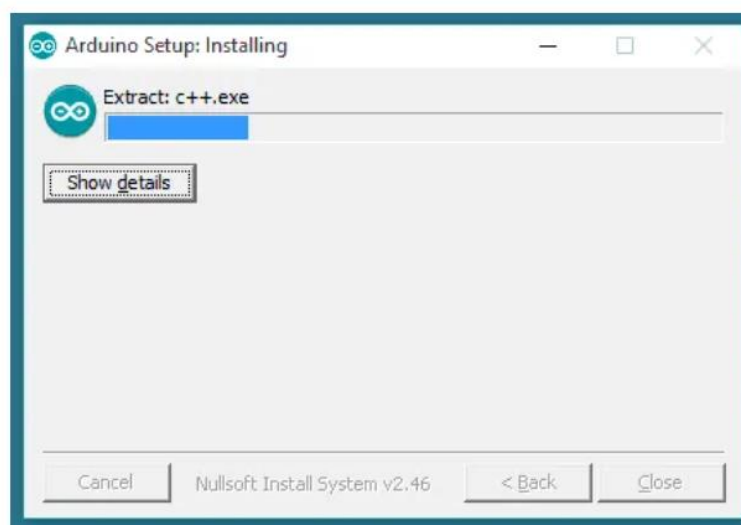
- Furthermore, create the shortcut folder on your desktop or wherever you want. Now, the Arduino IDE is installed on your PC. Just click on it, the IDE will open up.

InCase of Driver error

The install includes drivers, so in theory, you should be good to go straight away. If that fails for some reason, try these steps to install the drivers manually:

- Plug in your board and wait for Windows to begin its driver installation process. After a few moments, the process will fail, despite its best efforts.

- Click on **Start Menu** > **Control Panel**.

- Navigate to **System and Security** > **System**. Once the System window is up, open the **Device Manager**.

- Under **Ports** (COM & LPT), you should see an open port named **Arduino UNO (COMxx)**.

- Right click on **Arduino UNO (COMxx)** > **Update Driver Software**.

- Choose **Browse my computer for Driver software**.

- Navigate to and select the Uno's driver file, named **ArduinoUNO.inf**, located in the **Drivers** folder of the Arduino Software download.

- Windows will finish up the driver installation from there.

Installing the Arduino Software Package on Mac OS

Download the Arduino software for Mac from the **Arduino website**. Extract the contents of the **.zip** file and run the app. You can copy it into your applications folder, but it will run just fine from your **desktop** or **downloads** folders. You don't need to install any additional drivers for the Arduino UNO.

**Installing the Arduino Software on Package Ubuntu/Linux**

- Install **gcc-avr** and **avr-libc**:

- sudo apt-get install gcc-avr avr-libcIf you don't have openjdk-6-jre already, install and configure that too:

- sudo apt-get install openjdk-6-jre
  sudo update-alternatives --config javaSelect the correct **JRE** if you have more than one installed.

- Go to the **Arduino website** and download the Arduino Software for Linux. You can **untar** and run it with the following command:

- tar xzvf arduino-x.x.x-linux64.tgz
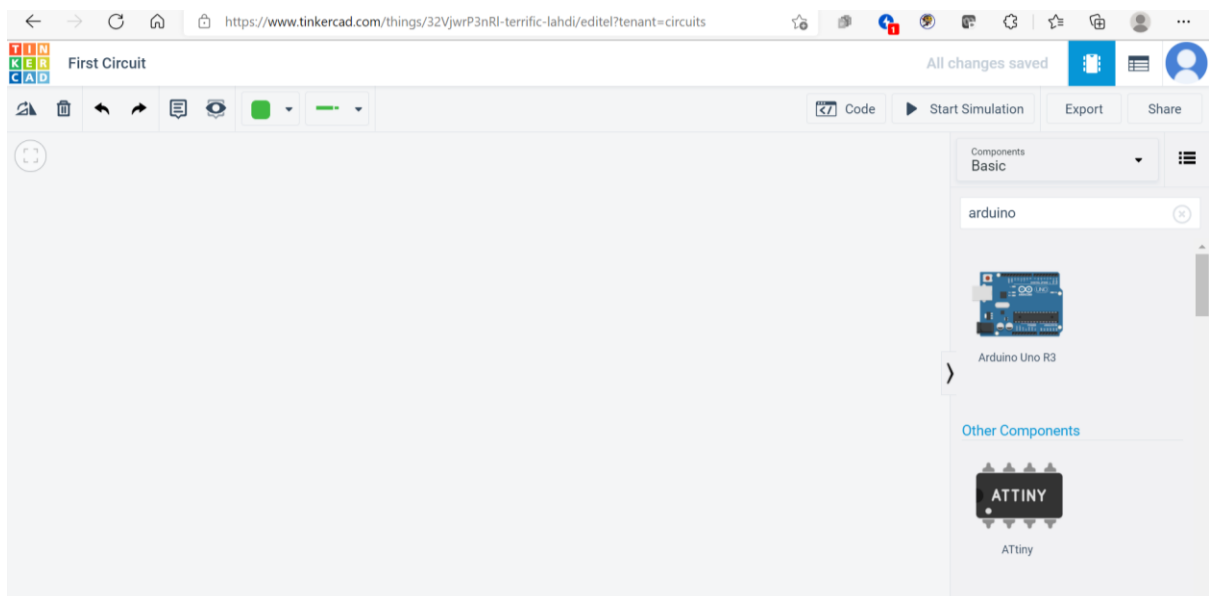 cd arduino-1.0.1
  ./arduino

TinkerCad:

Go to: https://www.tinkercad.com/

Sign Up and make an account.
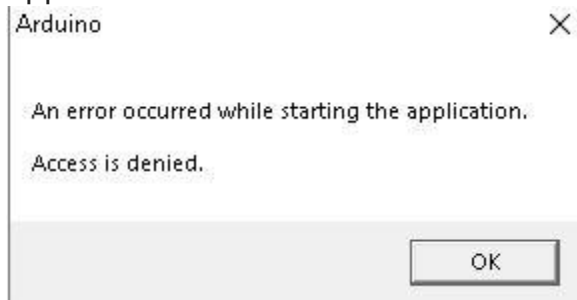
Login to your Dashboard.
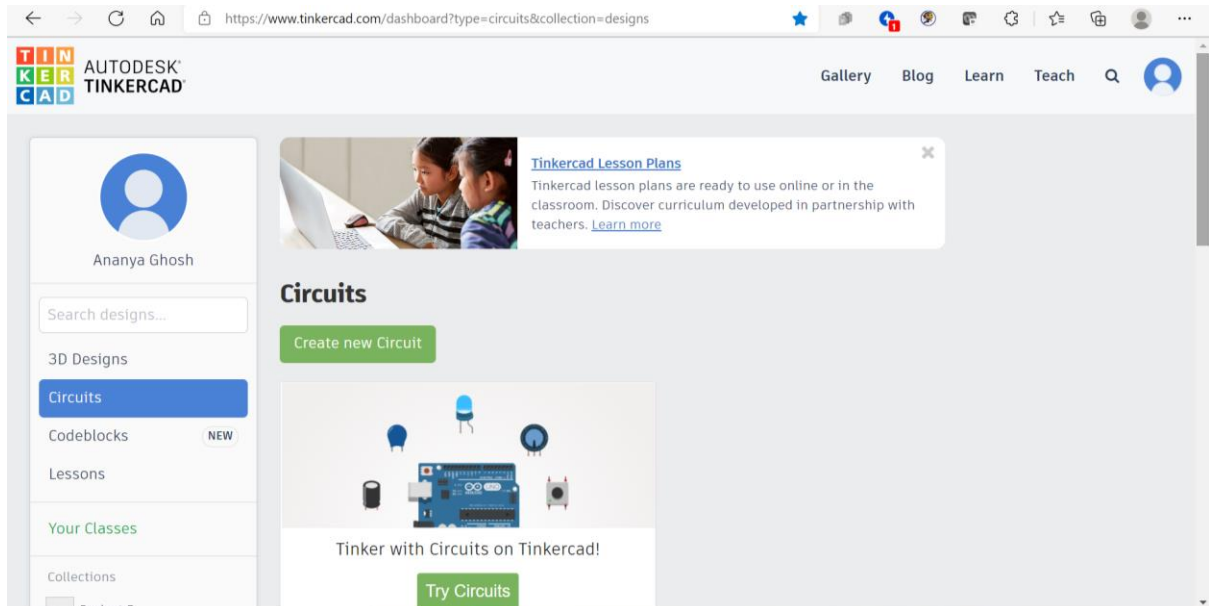
Create new Circuit

Arduino Installation errors

1. "An error occurred while starting the application." "Access is denied". Followed by the ever helpful "OK" box. When you click OK, the application closes.

    Sol:
    *   On Windows, and assuming you installed the normal version (not the windows app), there is a debug version of the Arduino IDE: C:\Program Files (x86)\Arduino\arduino_debug.exe; You will have to run the debug version from the command line.
    *   I went to my C: drive and did a hard search for any files (including hidden files) that fit the criteria *arduino*. Then, I deleted all of them that I could. Then I re-installed the IDE. Shut down the computer and then restarted it. Then, when I launched the IDE, it worked!

2. Error launching installer
    Sol:
    *   Please run the program arduino_debug.exe, which is located in the Arduino IDE installation folder, from the command line
    *   Download the "Windows ZIP file for non admin install": https://www.arduino.cc/download_handler.php?f=/arduino-1.8.5-windows.zip 4Unzip the downloaded file. You can move the unzipped folder to any convenient location. Run the file arduino.exe that's inside the unzipped folder. You can create a shortcut to it for convenience if you like.

Arduino IDE won't open; Error occurred during initialization of VM

If the Arduino IDE won't start on Windows (the initialization window appears and disappears without showing any error message or warning), try running arduino_debug.exe from within the Arduino root folder, which is in C:\Program Files (x86)\Arduino\ by default.

If you get something like the following output there is an issue with the Java memory allocation.


```
D:\Arduino\arduino-1.8.13-windows\arduino-1.8.13>arduino_debug.exe
Picked up _JAVA_OPTIONS: -Xmx2G
Error occurred during initialization of VM
Could not reserve enough space for 2097152KB object heap

D:\Arduino\arduino-1.8.13-windows\arduino-1.8.13>_
```

Follow these steps to work around it:

1. In the Arduino root folder, open the file `arduino.l4j.ini` with a text editor.
2. Find the line starting with `-Xmx`. By default, it should be `-Xmx512M`. If this is not the case, change the line to read `-Xmx512M`.
3. Try running `arduino_debug.exe` again.

If it still gives the same error, the reason may be that the setting is being overridden by the `_JAVA_OPTIONS` environment variable. To test this, run the following command from the `cmd` command line in the Arduino root folder.

```
set _JAVA_OPTIONS=
```

From the same `cmd` session, run the exe.

```
arduino_debug.exe
```

The IDE should now start. But note that the `_JAVA_OPTIONS` is only unset in the current `cmd` session. To make the change permanent, it needs to be unset on the system and/or user level, which is outside the scope of this article. Note that setting the variable in this way may interfere with other applications on your system.

Arduino IDE won't open; The cloud file provider is not running

If the Arduino IDE doesn't start on Windows (the initialization window appears and disappears without showing any error message or warning), try running `arduino_debug.exe` from the Command Prompt.
The `arduino_debug.exe` is located in the Arduino installation folder, which is in C:\Program Files (x86)\Arduino\ by default.

1. Press Win+R.
2. Type in 'cmd' and press enter.
3. Navigate to the installation folder with the `cd` command.

```
4. cd <path>
```

You may be able to copy the path into CMD by dragging the folder onto the CMD window.

5. Run the executable.

```
6. arduino_debug.exe
```

7. Observe the output. If you happen to get the following message

```
8. java.io.IOException: The cloud file provider is not running
```

there is most likely a problem with the OneDrive On-Demand synchronization. Proceed with the steps below.

Arduino IDE and Microsoft OneDrive On-Demand

The OneDrive On-Demand feature can result in essential files not being available or synchronized, which can stop the IDE from working.

There are numerous ways to fix this issue.

- Disable On-Demand for the IDE location.
- Install the IDE somewhere else (that does not have On-Demand enabled).
- Using the portable version will make the above easier, as it's all contained in one folder. But if that folder has On-Demand enabled, there could still be issues.

Error "Could not find the main class." when launching the Arduino IDE

If you get this error when launching Arduino:

```
Java Virtual Machine Launcher: Could not find the main class.
Program will exit.
```

make sure that you correctly extracted the contents of the Arduino .zip file - in particular that the `lib` directory is directly inside of the Arduino directory and contains the file `pde.jar`.

Error UnsatisfiedLinkError when launching Arduino IDE

If you get this error when launching Arduino:

```
 Uncaught exception in main method:
java.lang.UnsatisfiedLinkError: Native Library
/Users/anu/Desktop/arduino-0002/librxtxSerial.jnilib already
loaded in another classloader
```

you probably have an old version of the communications library lying around. Search for `comm.jar` or `jcl.jar` in `/System/Library/Frameworks/JavaVM.framework/` or in directories in your `CLASSPATH` or `PATH` environment variables.