

# Lektion 1: Bibliotheken einbinden / Seriellen Monitor öffnen

## Zusätzliche Arduino Bibliotheken einbinden

Wenn Sie mit der Arduino Entwicklungsumgebung und dessen Funktionen vertraut sind, möchten Sie möglicherweise die Möglichkeiten Ihres Arduinos mit zusätzlichen Bibliotheken erweitern.

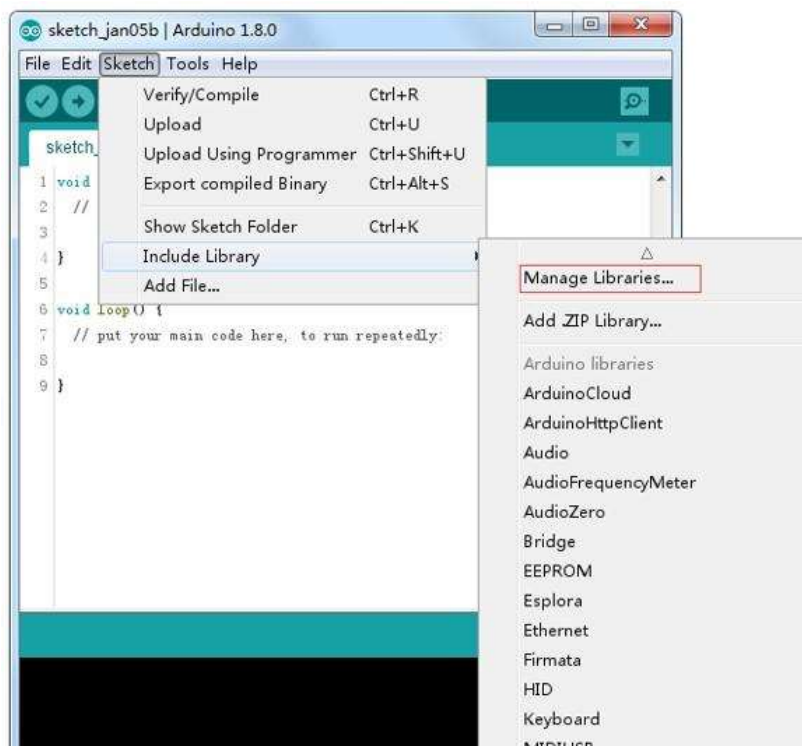
## Was sind Bibliotheken?

Bibliotheken sind Sammlungen von vorprogrammiertem Code, die es enorm einfach für Sie machen beispielsweise einen Sensor, ein Display oder ein Modul einzubinden. Zum Beispiel vereinfacht die vorinstallierte *LiquidCrystal* Bibliothek die Kommunikation zu LCD-Zeichenanzeigemodulen. Es gibt hunderte von zusätzlichen Bibliotheken für Arduino im Internet, die Sie sich einfach herunterladen können. Die mit der Arduino IDE mitkommenden Bibliotheken und einige zusätzliche Bibliotheken sehen Sie später zum Vergleich. Um zusätzliche Bibliotheken benutzen zu können, müssen diese erst installiert werden.

## Wie installiert man eine Bibliothek?

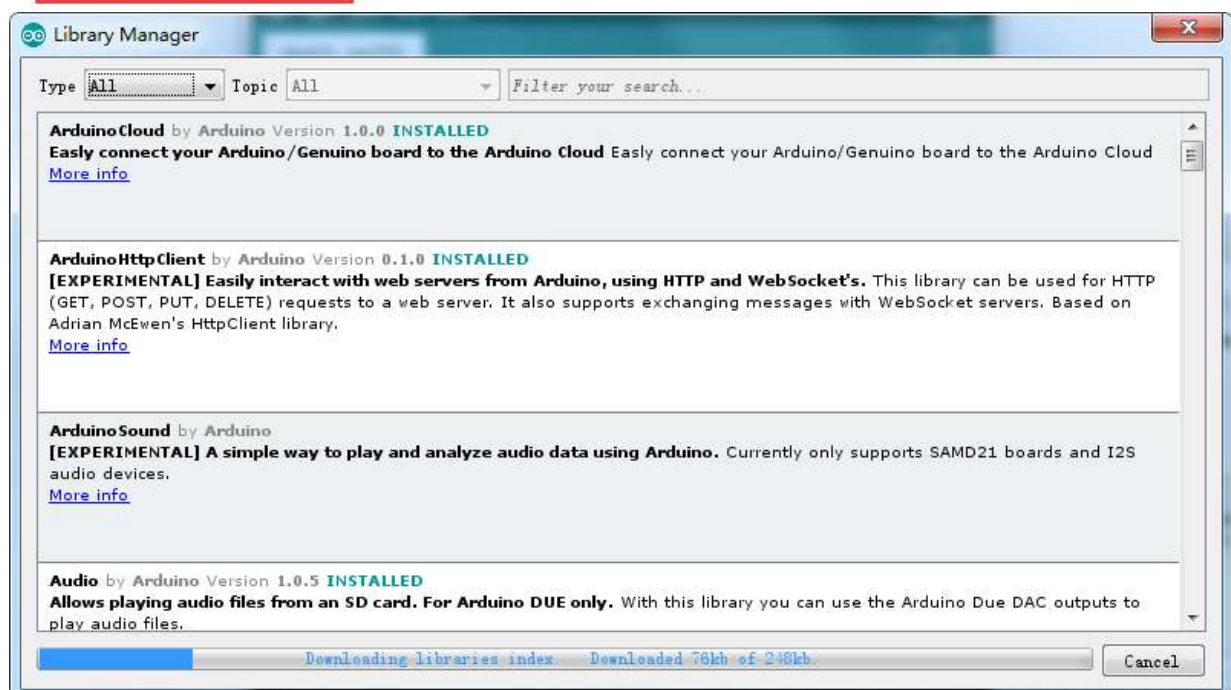
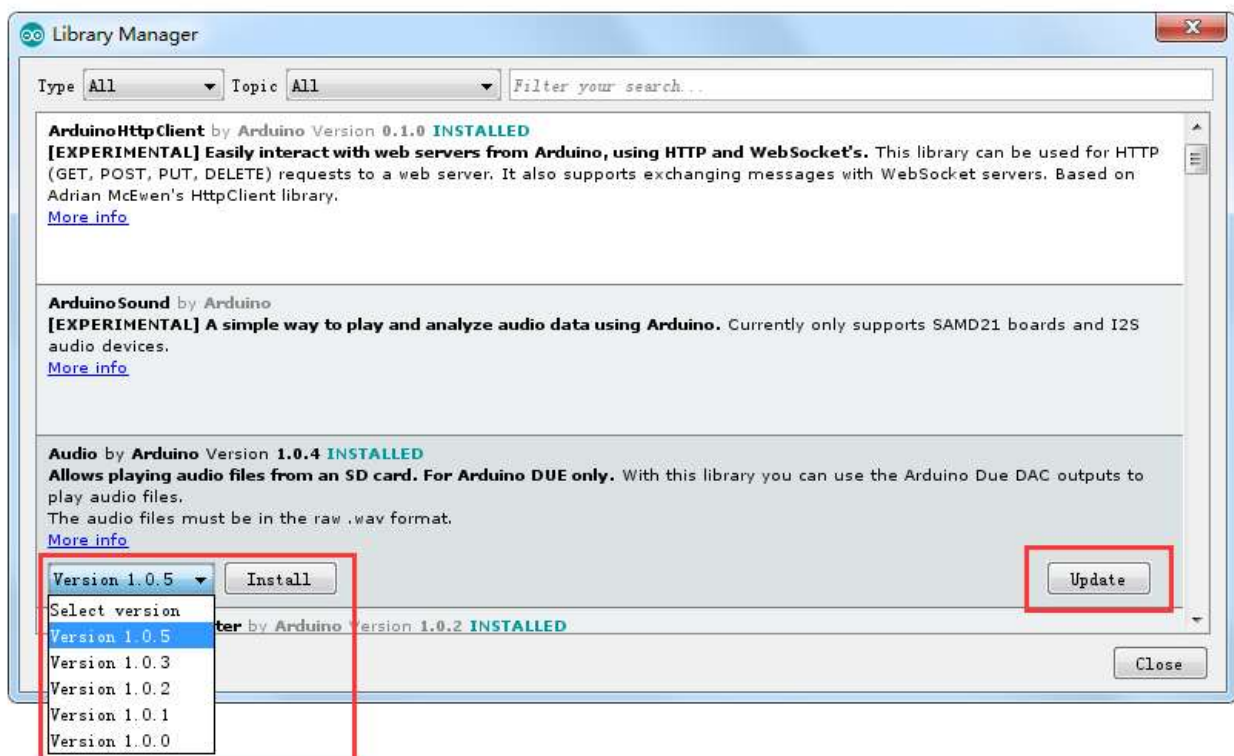
### Mit Hilfe des Bibliotheksverwalters

Um eine neue Bibliothek in Ihre Arduino IDE einzubinden, können Sie den *Bibliotheksverwalter* nutzen (verfügbar seit Arduino IDE 1.8.0). Öffnen Sie die IDE und klicken unter *Sketch > Bibliothek einbinden* auf „Bibliotheken verwalten“.

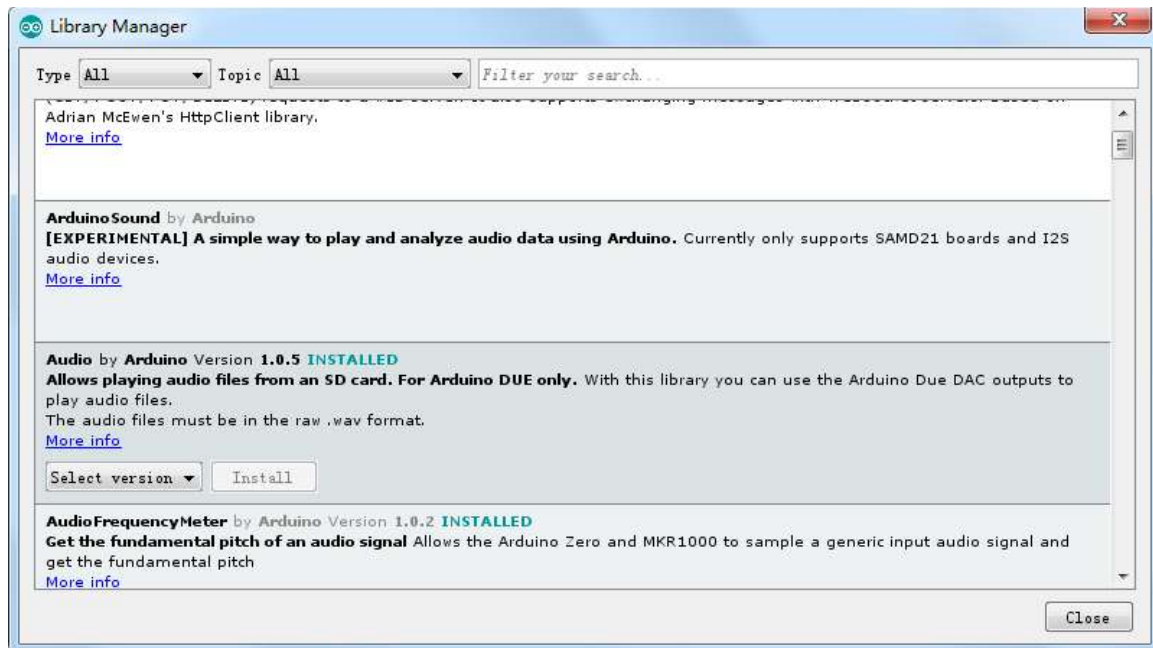


Dann wird sich der Bibliotheksverwalter öffnen und Sie finden eine Liste voller Bibliotheken, die bereits installiert sind oder heruntergeladen werden können. Als Beispiel werden wir die Bibliothek *Audio* installieren. Scrollen Sie die Liste herunter und suchen Sie die Bibliothek. Dann wählen Sie die Version aus, die Sie installieren wollen. Manchmal gibt es nur eine verfügbare Version und es lässt sich keine Version auswählen. Keine Sorge: Das ist normal.

Manchmal müssen Sie ein bisschen geduldig sein, da es sehr viele Bibliotheken gibt. Lassen Sie die Software die Liste aktualisieren und suchen Sie dann den Eintrag.



Klicken Sie zum Schluss auf *Installieren* und warten Sie, bis die Bibliothek installiert ist. Das Herunterladen kann abhängig von Ihrer Internetverbindungsgeschwindigkeit etwas dauern. Sobald die Installation abgeschlossen ist, sollte die Bibliothek im Verwalter den Zusatz „*Installiert*“ haben. Danach können Sie den Bibliotheksverwalter wieder schließen.

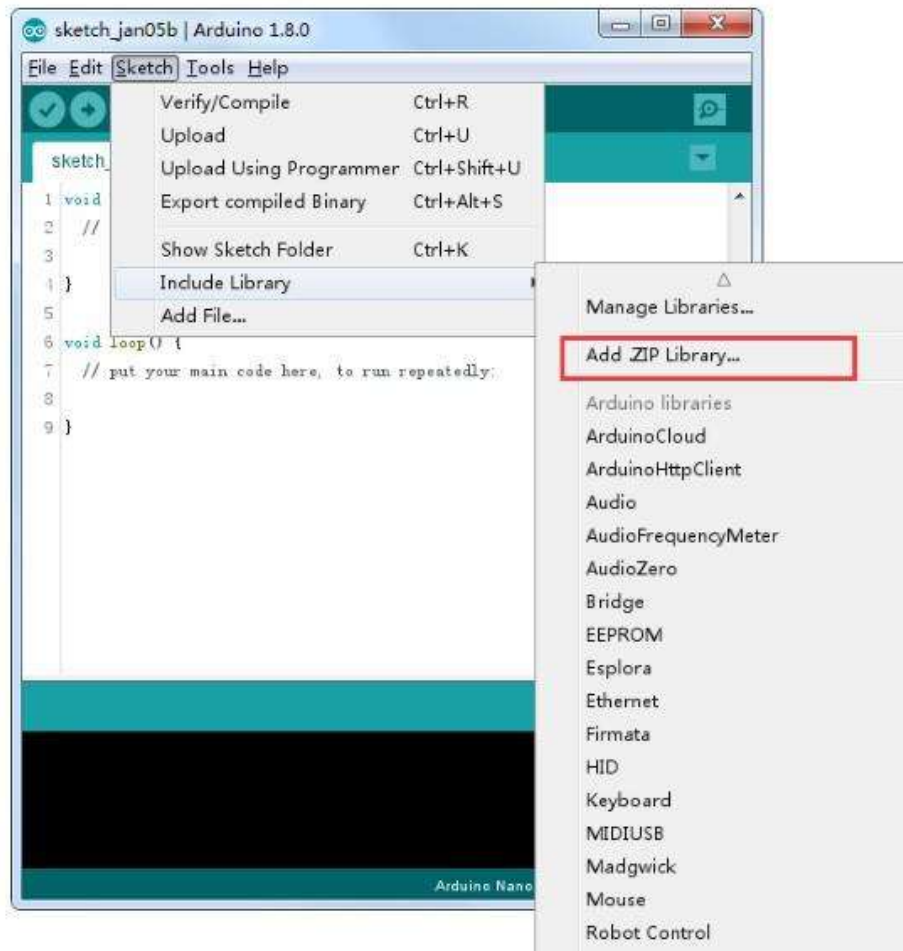


Jetzt können Sie die Bibliothek auch schon über das Einbindungsmenü hinzufügen. Wenn Sie Ihre eigene Bibliothek im Bibliotheksverwalter hinzufügen möchten, erstellen Sie einen Eintrag auf der [Github](#)-Seite von Arduino.

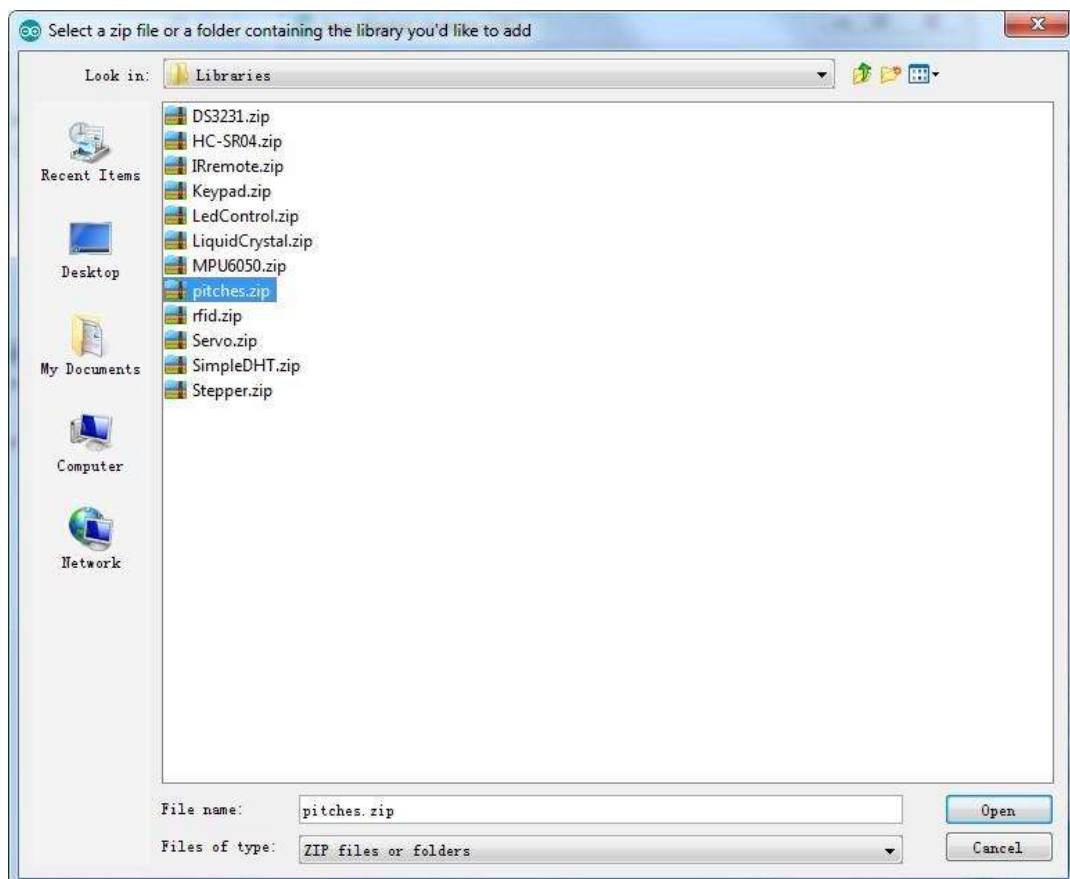
## Eine .zip Bibliothek einbinden

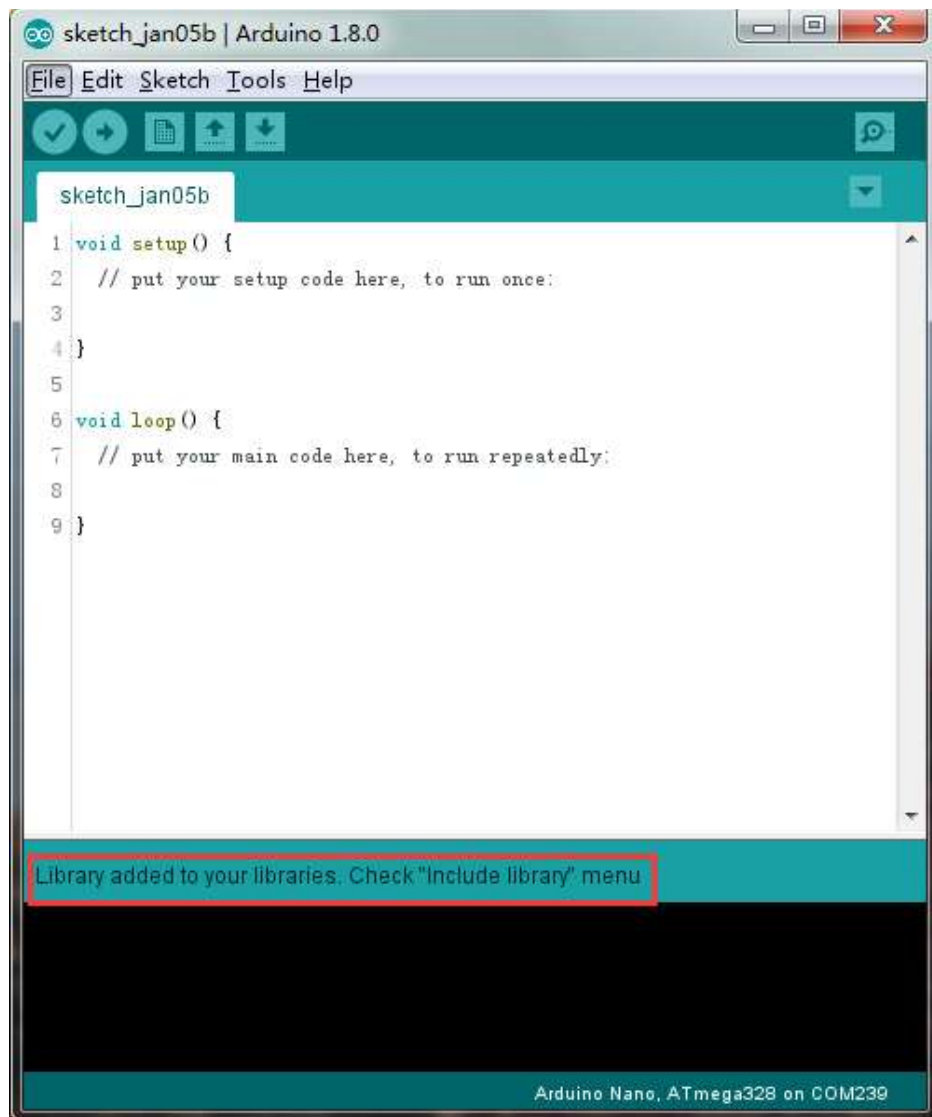
Weitere Bibliotheken werden im Internet häufig als ZIP Dateien zum Download angeboten. Der Name des ZIP Archivs ist meist der Name der Bibliothek. Im Archiv befindet sich eine *.cpp* Datei, eine *.h* Datei und oft eine *keywords.txt* Datei und ein *examples* Ordner mit Code-Beispielen und andere Dateien, die von der Bibliothek benötigt werden. Seit Version 1.0.5 können Drittanbieter-Bibliotheken in der Arduino IDE eingebunden werden. Entpacken Sie die ZIP Dateien der Bibliotheken nicht selber, sondern belassen Sie sie so wie sie sind. Die IDE wird das Archiv automatisch entpacken.

Navigieren Sie in der IDE zu *Sketch > Bibliothek einbinden* und wählen Sie „*ZIP Bibliothek hinzufügen...*“.



Sie werden dazu aufgefordert, die ZIP-Datei auszuwählen, die Sie hinzufügen wollen.  
Öffnen Sie die heruntergeladene .ZIP-Bibliothek.





Gehen Sie zurück zum Menü *Sketch > Bibliothek einbinden*. Sie sollten Ihre soeben hinzugefügte Bibliothek unten im Drop-Down-Menü wiederfinden. Die Bibliothek kann nun in Ihrem Sketch / Projekt eingebunden werden. Die ZIP-Bibliothek wurde automatisch zu den anderen Bibliotheken in Ihrem Arduino Projektordner hinzugefügt.

Info: Die neue Bibliothek kann direkt in Ihren Sketch eingebunden werden. Die Beispieldateien der neuen Bibliothek sind jedoch erst nach einem Neustart der IDE verfügbar.

Dies sind beiden geläufigsten Wege, um Bibliotheken hinzuzufügen. Bei Mac und Linux Systemen funktioniert es auf die gleiche Art. Die untenstehende manuelle Installation wird nur selten gebraucht und falls Sie diese nicht benötigen, können Sie sie überspringen.

## Manuelle Installation

Beenden Sie zuerst die Arduino Software, bevor Sie die Bibliothek manuell installieren. Dann entpacken Sie das ZIP-Archiv der Bibliothek. Wenn Sie zum Beispiel eine Bibliothek namens "*ArduinoParty*" heruntergeladen haben, entpacken Sie die *ArduinoParty.zip* Datei. Der entpackte Ordner sollte einen Ordner namens *ArduinoParty* enthalten, in welchem sich wiederum Dateien wie *ArduinoParty.cpp* und *ArduinoParty.h* befinden sollten. Wenn die .cpp und .h Dateien nicht in einem separaten Ordner liegen, müssen Sie einen erstellen. In diesem Fall würden Sie einen Ordner namens „*ArduinoParty*“ erstellen und verschieben alle Dateien, die im Zip-Archiv waren, in diesen Ordner (wie *ArduinoParty.cpp* und *ArduinoParty.h*). Verschieben Sie den *ArduinoParty* Ordner nun in Ihren Bibliotheksordner. Unter Windows und Mac ist dies standardmäßig unter *Dokumente\Arduino\libraries* . Unter Linux ist es ebenfalls *libraries* Ordner in Ihrem Projektordner.

Ihr Arduino Bibliotheks Ordner sollte nun so aussehen (unter Windows):

*Dokumente\Arduino\libraries\ArduinoParty\ArduinoParty.cpp*  
*Dokumente\Arduino\libraries\ArduinoParty\ArduinoParty.h*  
*Dokumente\Arduino\libraries\ArduinoParty\examples*

oder so (unter Mac und Linux):

*Dokumente/Arduino/libraries/ArduinoParty/ArduinoParty.cpp*  
*Dokumente/Arduino/libraries/ArduinoParty/ArduinoParty.h*  
*Dokumente/Arduino/libraries/ArduinoParty/examples*

....

Es können mehr Dateien als nur die .cpp and .h Dateien vorhanden sein, gehen Sie nur sicher, dass sich alle Dateien in Ihrem Ordner befinden. Die Bibliothek wird nicht funktionieren, wenn sich die .cpp und .h Dateien direkt im *libraries* Ordner oder in einem weiteren Unterordner befinden. Zum Beispiel:

*Dokumente\Arduino\libraries\ArduinoParty.cpp* und  
*Dokumente\Arduino\libraries\ArduinoParty\ArduinoParty\ArduinoParty.cpp*  
würden nicht funktionieren.

Wenn alles richtig ist, starten Sie die Arduino IDE neu. Stellen Sie sicher, dass die neue Bibliothek unter *Sketch > Bibliothek einbinden* zu finden ist. Dann sind Sie fertig.

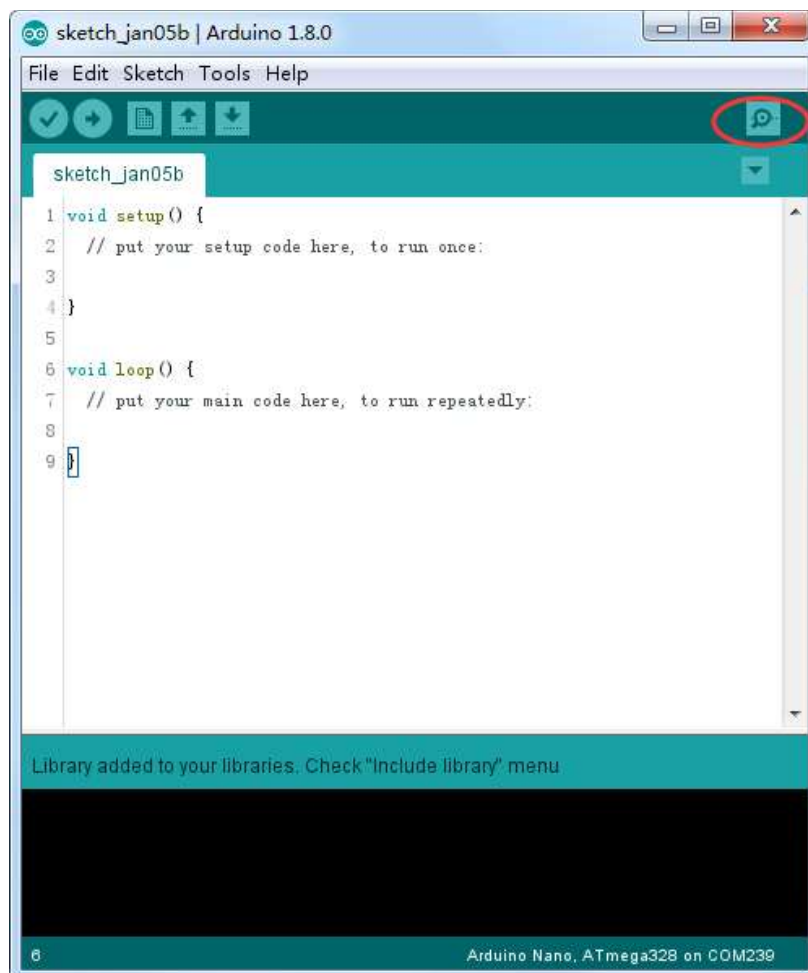


## Arduino Serieller Monitor (Windows, Mac, Linux)

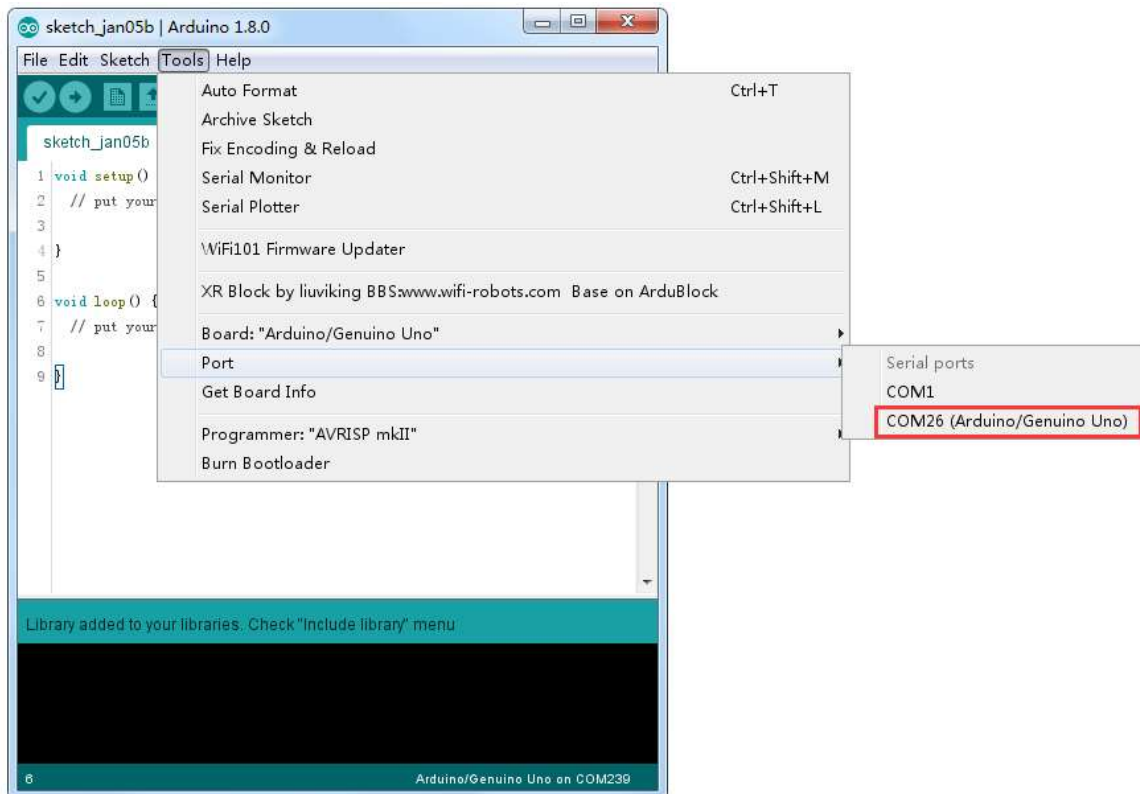
Die Arduino Entwicklungsumgebung (IDE) ist die Software Seite der Arduino Plattform. Und weil das Arbeiten mit einem Terminal ein so bedeutender Teil beim Programmieren mit Arduino und anderen Mikrocontrollern ist, haben sich die Entwickler der Arduino IDE dazu entschieden, ein Terminal (Kommandozeile) zu integrieren. In der Arduino Entwicklungsumgebung wird dies „Serieller Monitor“ genannt.

### Die Verbindung herstellen

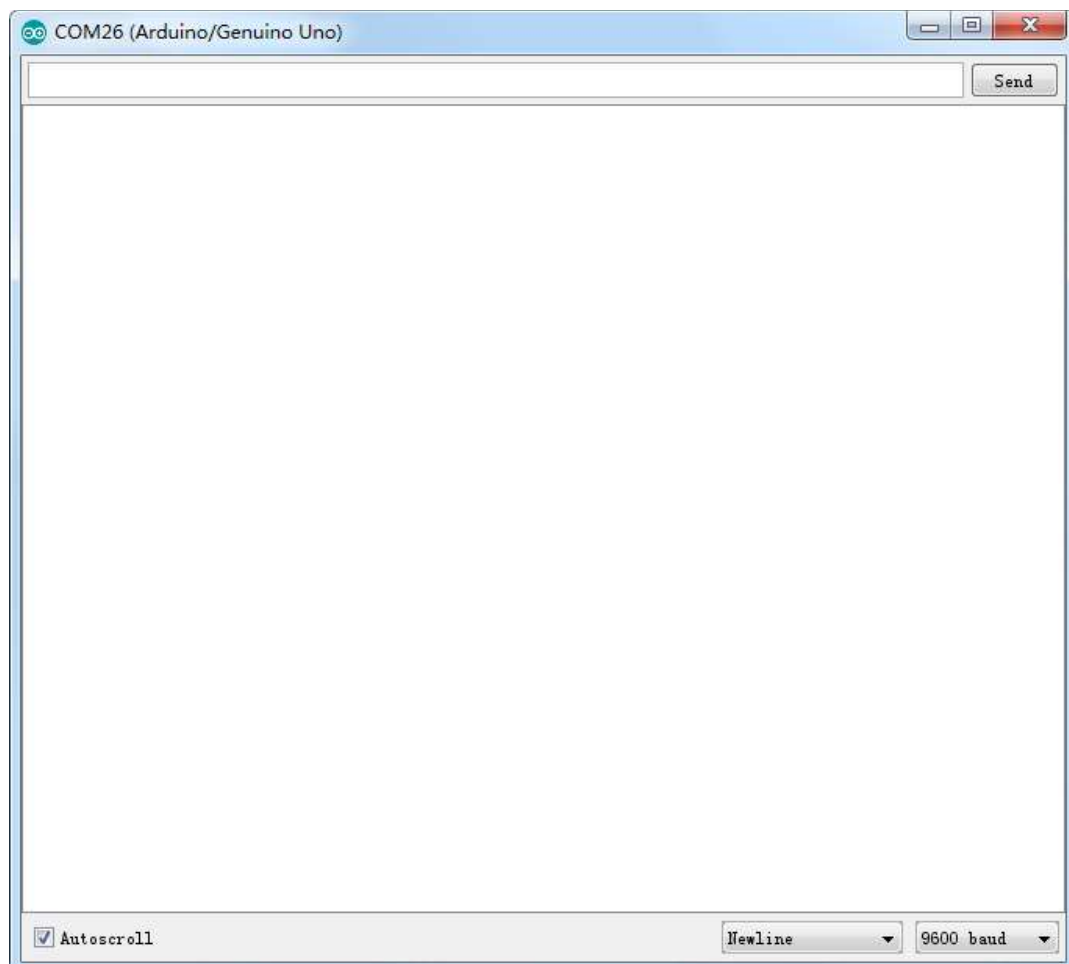
Der Serielle Monitor ist in jeder Version der Arduino IDE vorhanden. Um ihn zu öffnen, müssen Sie lediglich auf das im Bild unten markierte Symbol klicken.



Bevor die Verbindung möglich ist, müssen Sie den richtigen COM-Port auswählen. Gehen Sie zu *Werkzeuge* -> *Port* und wählen Sie den richtigen Port aus. Normalerweise ist nur einer vorhanden. Sonst schauen Sie im Geräte-Manager nach.



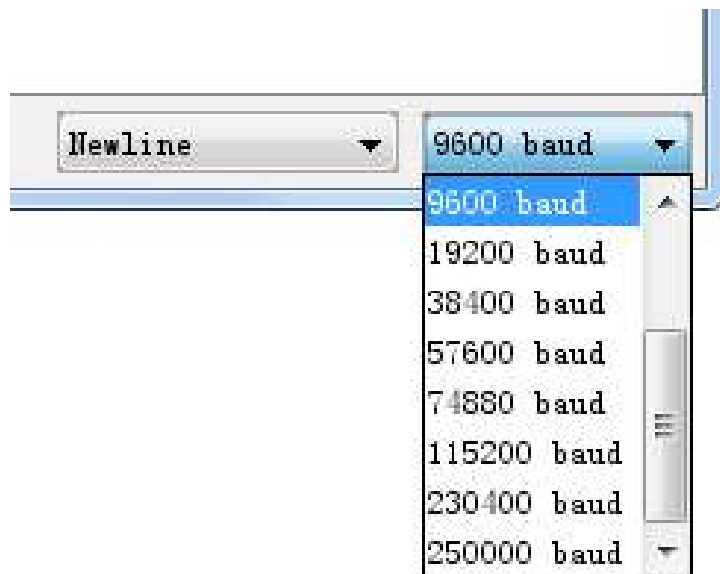
Sobald Sie den Seriellen Monitor dann geöffnet haben, erscheint dies:



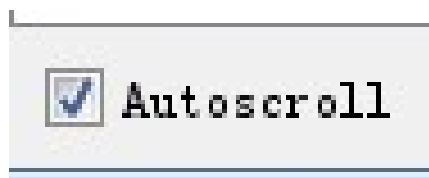


## Einstellungen

Die Optionen des Seriellen Monitors sind sehr limitiert, aber reichen aus, um die wichtigsten Optionen der Seriellen Kommunikation festzulegen. Die erste Einstellung, die Sie machen, ist die Festlegung der *Baud-Rate*. Klicken Sie auf das im Bild zu sehene Drop-Down Menü und wählen Sie die richtige Baudrate aus (9600 baud).



Schließlich können Sie durch Anklicken der Checkbox *Autoscroll* einstellen, ob der Serielle Monitor automatisch mitscrollen soll.



## Vorteile

Der Serielle Monitor ist ein schneller und einfacher Weg eine Serielle Verbindung zu Ihrem Arduino aufzubauen. Wenn Sie bereits mit der Arduino IDE arbeiten, brauchen Sie kein weiteres Terminal zu installieren, um sich die serielle Kommunikation Ihres Arduinos anzuschauen.

## Nachteile

Die minimalistischen Einstellungen des Seriellen Monitors lassen zu wünschen übrig und für fortgeschrittene Serielle Kommunikation kann dies unzureichend sein.