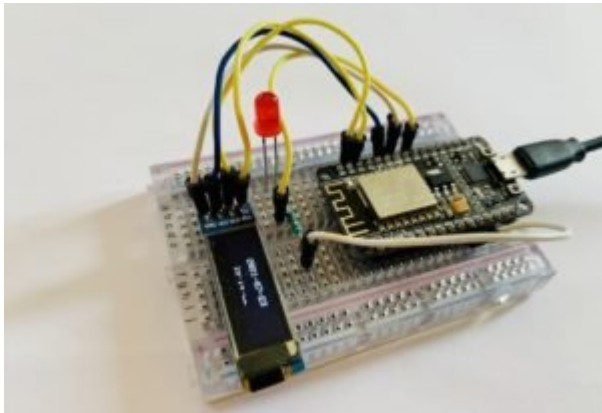


# ESP8266 programmieren mit der Arduino IDE



Ein ESP8266 oder auch ein Wemos D1 Mini ist ein echter Gewinn für deine Projekte. Mit diesen Microcontrollern stößt du die Tür zum Internet of Things ganz weit auf! **Und das beste: Du kannst ganz einfach mit deiner Arduino IDE deinen ESP8266 programmieren** – genauso wie deine Arduino-Boards. Ein paar Vorbereitungen musst du hierfür allerdings treffen.

Lerne hier, wie du in 5 Minuten deinen ESP8266 oder D1 Mini in deiner Arduino IDE installierst und anschließend mit dem Programmieren loslegen kannst.

## Deinen ESP8266 oder D1 Mini in der Arduino IDE installieren

Öffne zuerst die Einstellungen deiner Arduino IDE. Dort findest du das Feld **Zusätzliche Boardverwalter-URLs**. Trage hier die folgende Adresse ein:

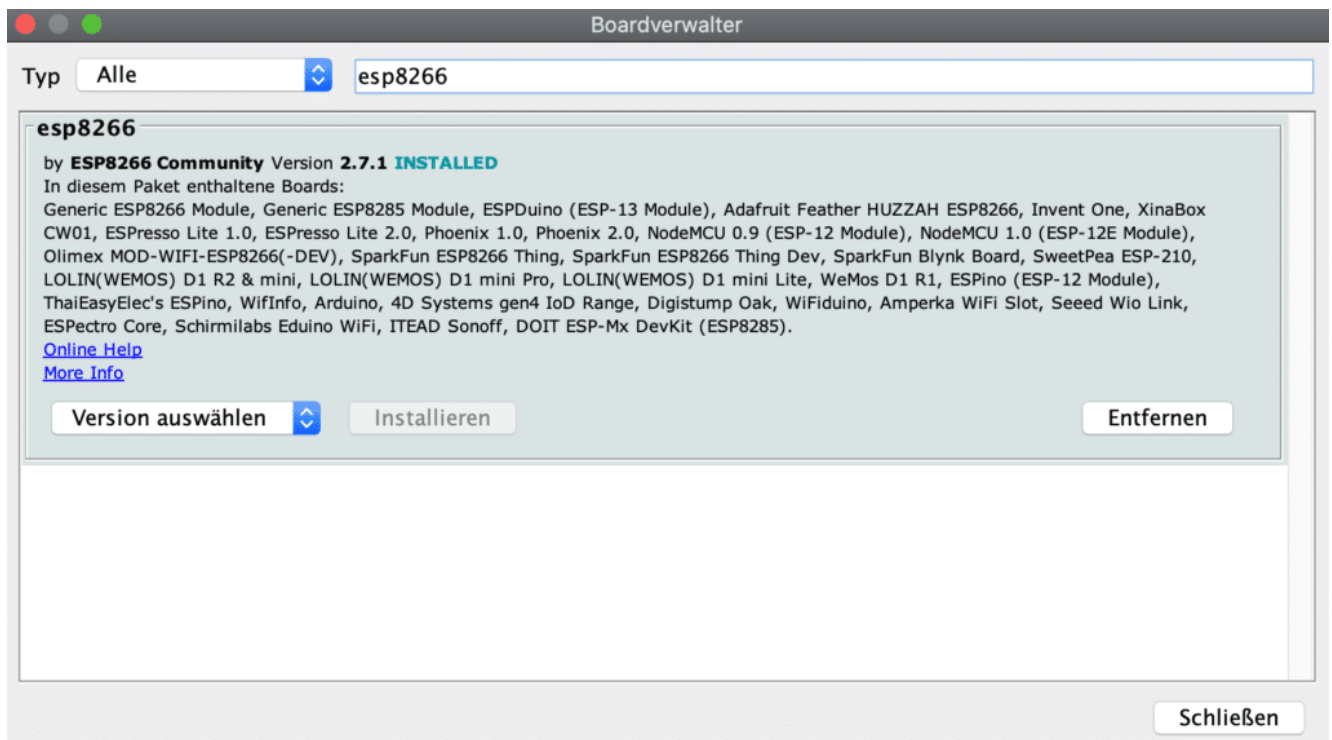
[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

**Tipp:** Wenn du dort schon die [URL deines ESP32](#) eingetragen

hast, schreibe die des ESP8266 einfach mit einem Komma getrennt dahinter. Dann verfügst du in der Arduino IDE über beide.

Schließe nun das Fenster mit einem Klick auf **OK**. Öffne als nächstes das Menü **Werkzeuge** und wähle dort den Menüpunkt **Boards** und anschließend **Boardverwalter**.

Suche in dem Fenster, das sich jetzt öffnet, nach **ESP8266**. Scrolle etwas nach unten, bis du den Eintrag **ESP8266 by ESP8266 Community** findest. Noch ein Klick auf **Installieren**, kurz warten und das sollte es gewesen sein. □



Das Paket für dein ESP8266 installieren

*Hinweis: Solltest du mit dem ESP8266 in der Arduino IDE Probleme beim Hochladen deiner Sketches haben, installiere probeweise eine frühere Version des Boards.*

# Deinen ESP8266 mit einem Sketch programmieren

Jetzt kannst du loslegen und deinen ESP8266 programmieren. Du hast ja bestimmt schon einmal die fest installierte LED auf einem Arduino blinken lassen – das funktioniert auch auf deinem ESP8266.

Erstelle also einen neuen Sketch mit folgendem Code:

```
/*  
Blinkende LED  
https://polluxlabs.net  
*/  
  
#define LED D0  
  
void setup() {  
  pinMode(LED, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(LED, LOW);  
  delay(500);  
  digitalWrite(LED, HIGH);  
  delay(500);  
}
```

Schließe nun deinen ESP8266 an den USB-Port deines Computers an. Wähle noch einmal im Menü den Punkt **Werkzeuge** und wähle unter **Board** den Eintrag **NodeMCU 1.0** und den Port, an dem dein ESP8266 angeschlossen ist.

Für deinen [Wemos D1 Mini\\*](#) wähle den Eintrag **LOLIN(WEMOS) D1**. Falls du eine Pro- oder Lite-Version hast, wähle natürlich den entsprechenden Eintrag.

Lade jetzt den Sketch hoch. Wenn alles funktioniert hat, blinkt

die Onboard-LED im Halbsekundentakt.

## Spannende ESP8266 Projekte

Zeit für etwas mehr Internet of Things? Hier sind ein paar spannende Projekte, die du mit deinem ESP8266 umsetzen kannst:

- [Dein eigener ESP8266 Web Server](#)
- [Erinnerung an deinen Regenschirm](#)