



Re-fashioning the Future

Aileen Drohan

Agenda

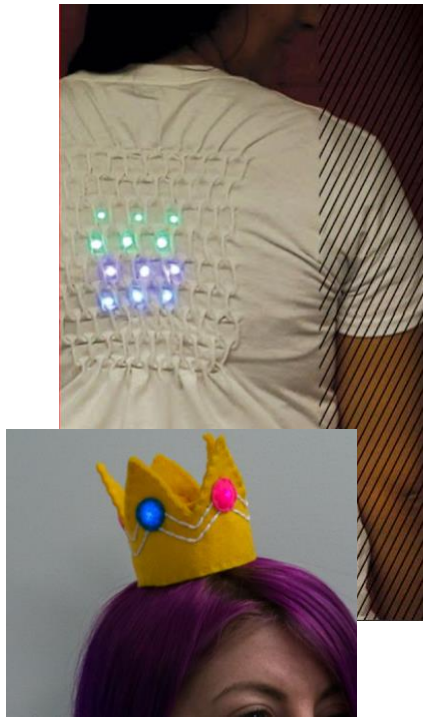
- Introduction
- Arduino setup
- Lilypad setup
- The Codebase
- Circuit Testing



Introduction

E-Textiles

- An Etextile is any fabric that has some electrical function.



The Lilypad family

- Designed with **sewable petals**, the boards can be integrated into fabric/garments with conductive thread.
- The boards are **washable**, except for the Buzzer the Vibe boards. Gentle hand washing is recommended. Batteries must be removed before washing.
- The Lilypad controller can accept operating **voltages between 2.7 and 5.5 volts**. Anything higher than 5.5 volts will fry the board.
- Commonly powered with a **3.7V LiPo battery with a JST connector**.
- Programming the Lilypad is easily achieved with **Arduino**.

https://www.sparkfun.com/lilypad_products

Your electronics workshop kit

1. Arduino LilyPad 328 Main Board
2. FTDI Basic Breakout (for uploading code)
3. Simple Power (for a LiPo battery)
4. Push button
5. Light sensor
6. Leds (static colours)
7. Pixel board (programmable colour)
8. Vibration Motor
9. Buzzers (to share)
10. Crocodile leads
11. 3.7V LiPo Battery

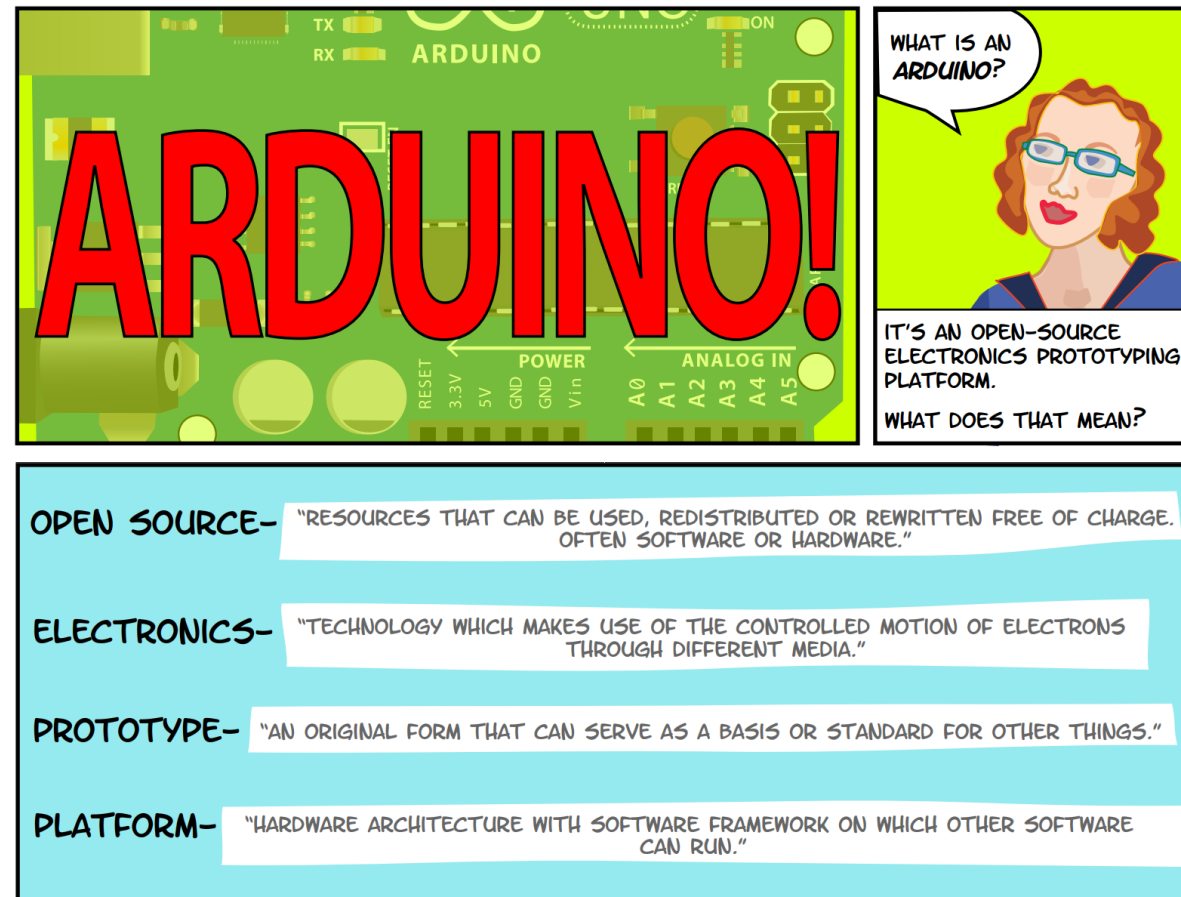


https://www.sparkfun.com/lilypad_products



Arduino Setup

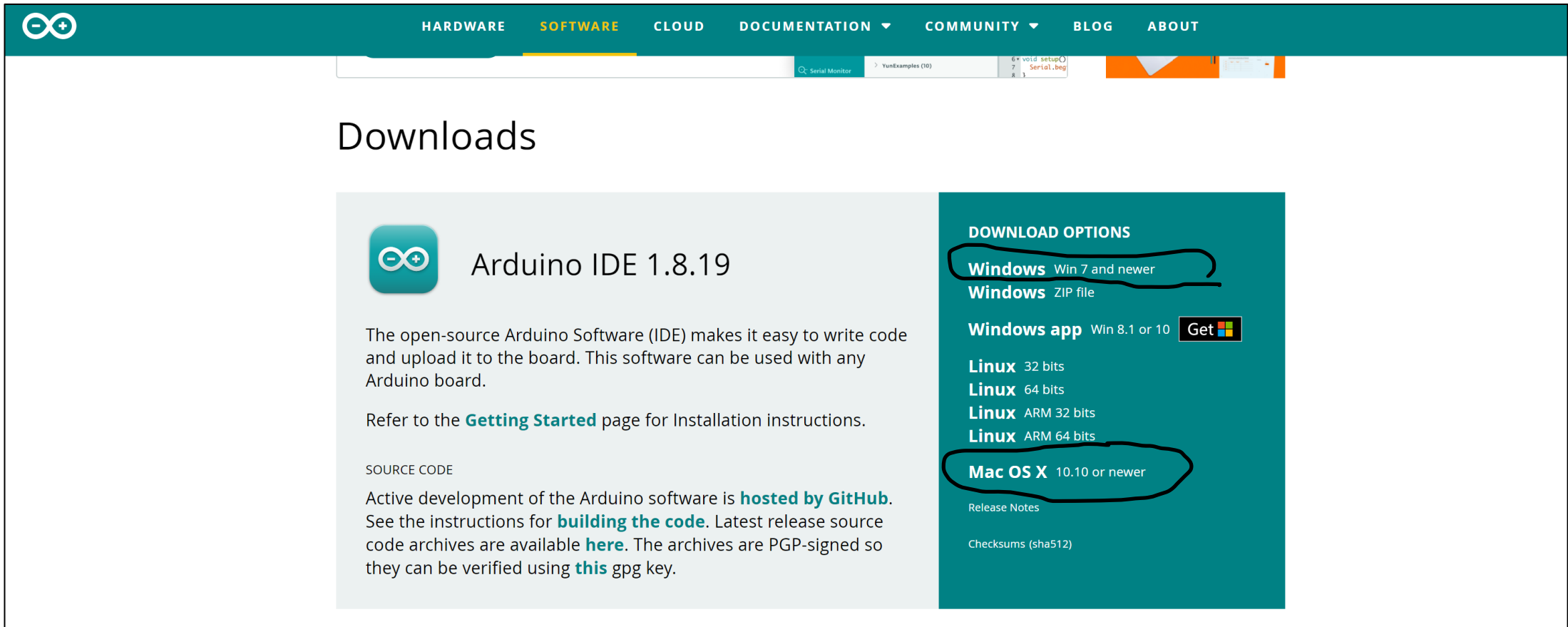
What is Arduino?



<http://www.jodyculkin.com/wp-content/uploads/2011/09/arduino-comic-latest3.pdf>


Installing Arduino

Go to Arduino Software page <https://www.arduino.cc/en/software>



The screenshot shows the Arduino Software page. The top navigation bar includes links for HARDWARE, SOFTWARE (highlighted), CLOUD, DOCUMENTATION, COMMUNITY, BLOG, and ABOUT. Below the navigation bar, the page title "Downloads" is displayed. The main content area features a large card for "Arduino IDE 1.8.19". The card includes the Arduino logo, a description of the IDE, and a link to the "Getting Started" page. To the right of the main card, a teal sidebar titled "DOWNLOAD OPTIONS" lists various download links. The "Windows" and "Mac OS X" options are circled in black. The "Windows" option includes links for "Win 7 and newer" and "ZIP file". The "Mac OS X" option includes a link for "10.10 or newer". Below the "Mac OS X" option, there are links for "Release Notes" and "Checksums (sha512)".

Downloads

 **Arduino IDE 1.8.19**

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file

Windows app Win 8.1 or 10 [Get](#)

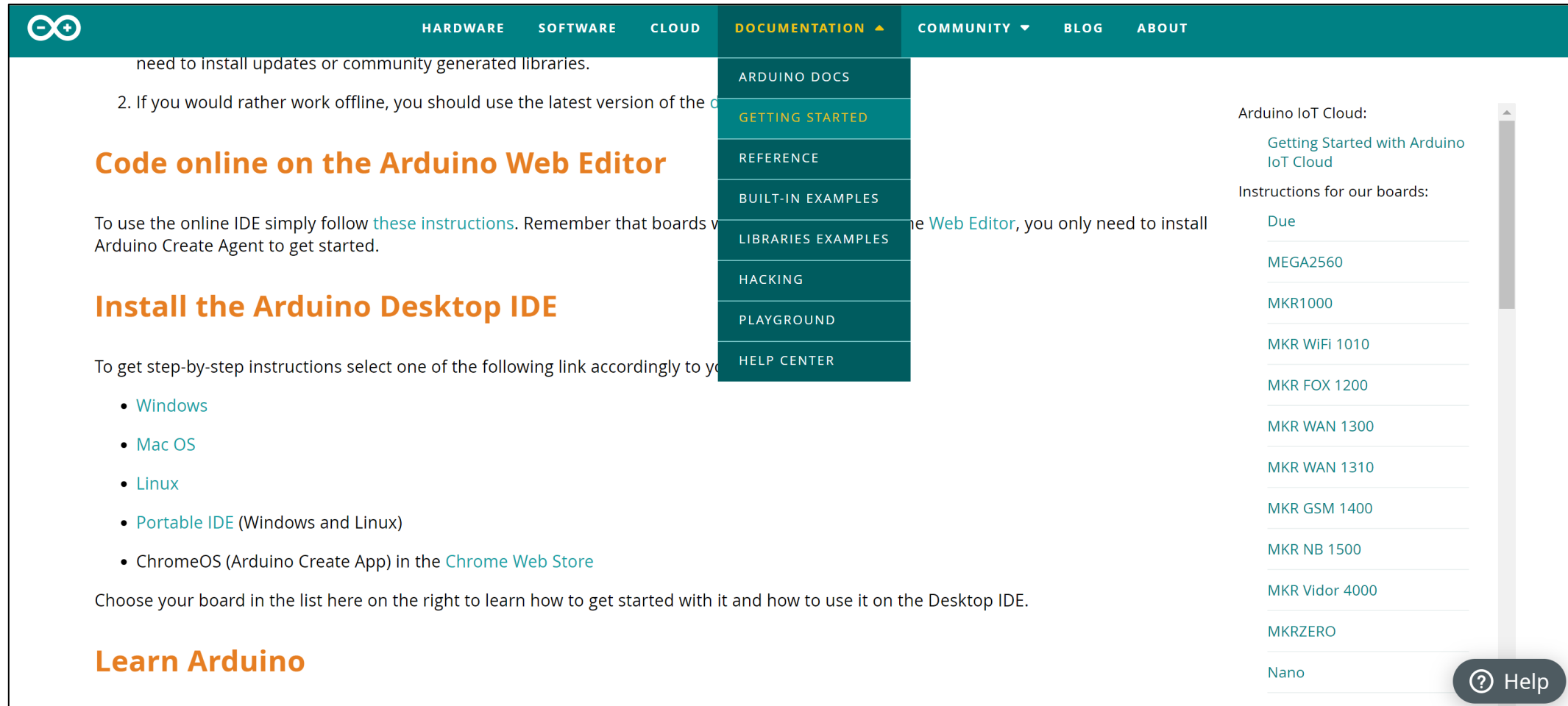
Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)

[Checksums \(sha512\)](#)

Installation Guides



The screenshot shows the Arduino website's navigation and content. The top navigation bar includes links for HARDWARE, SOFTWARE, CLOUD, DOCUMENTATION (highlighted with a dropdown menu), COMMUNITY, BLOG, and ABOUT. The DOCUMENTATION dropdown menu lists: ARDUINO DOCS, GETTING STARTED (highlighted), REFERENCE, BUILT-IN EXAMPLES, LIBRARIES EXAMPLES, HACKING, PLAYGROUND, and HELP CENTER. The main content area on the left features a section titled "Code online on the Arduino Web Editor" with instructions to follow [these instructions](#) and remember that boards need the Arduino Create Agent. Below this is a section titled "Install the Arduino Desktop IDE" with instructions to select a link based on the user's system. A list of links follows: Windows, Mac OS, Linux, Portable IDE (Windows and Linux), and ChromeOS (Arduino Create App) in the [Chrome Web Store](#). A note states: "Choose your board in the list here on the right to learn how to get started with it and how to use it on the Desktop IDE." The right sidebar contains a section "Arduino IoT Cloud:" with a link to "Getting Started with Arduino IoT Cloud", followed by "Instructions for our boards:" and a list of boards: Due, MEGA2560, MKR1000, MKR WIFI 1010, MKR FOX 1200, MKR WAN 1300, MKR WAN 1310, MKR GSM 1400, MKR NB 1500, MKR Vidor 4000, MKRZERO, and Nano. A "Help" button is located at the bottom right of the sidebar.

need to install updates or community generated libraries.

2. If you would rather work offline, you should use the latest version of the [Arduino IDE](#).

Code online on the Arduino Web Editor

To use the online IDE simply follow [these instructions](#). Remember that boards need the [Arduino Create Agent](#) to get started.

Install the Arduino Desktop IDE

To get step-by-step instructions select one of the following link accordingly to your operating system:

- [Windows](#)
- [Mac OS](#)
- [Linux](#)
- [Portable IDE](#) (Windows and Linux)
- [ChromeOS](#) (Arduino Create App) in the [Chrome Web Store](#)

Choose your board in the list here on the right to learn how to get started with it and how to use it on the Desktop IDE.

Learn Arduino

Arduino IoT Cloud:

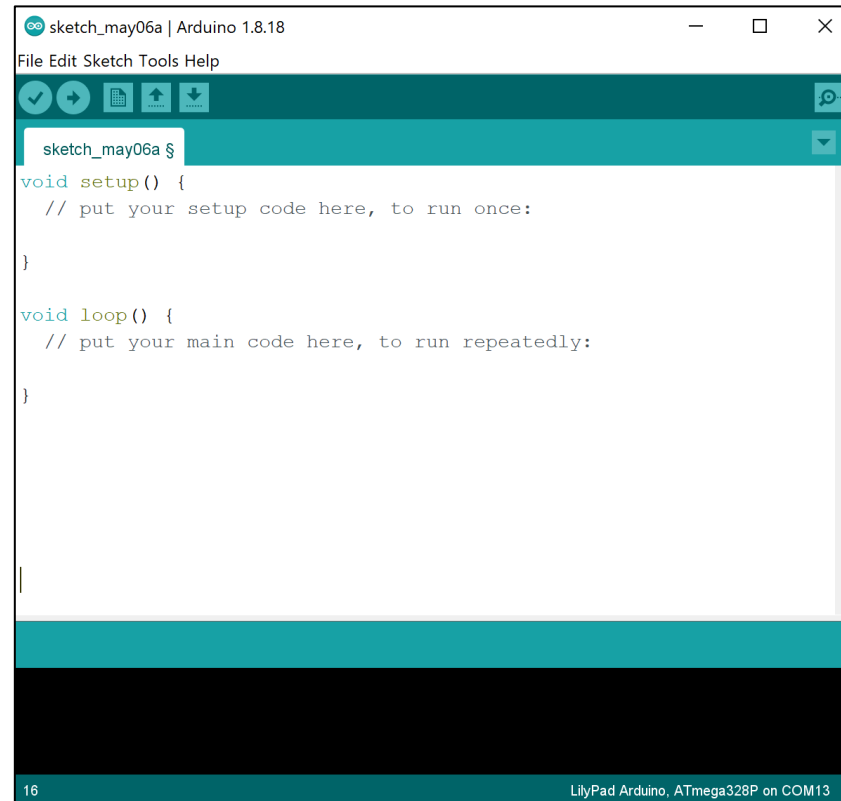
[Getting Started with Arduino IoT Cloud](#)

Instructions for our boards:

- [Due](#)
- [MEGA2560](#)
- [MKR1000](#)
- [MKR WIFI 1010](#)
- [MKR FOX 1200](#)
- [MKR WAN 1300](#)
- [MKR WAN 1310](#)
- [MKR GSM 1400](#)
- [MKR NB 1500](#)
- [MKR Vidor 4000](#)
- [MKRZERO](#)
- [Nano](#)

[Help](#)

Successful Installation



The screenshot shows the Arduino IDE window titled 'sketch_may06a | Arduino 1.8.18'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for opening, saving, uploading, and downloading. The main text area contains the following code:

```
sketch_may06a $  
  
void setup() {  
  // put your setup code here, to run once:  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
}
```

At the bottom of the window, a status bar indicates '16' and 'LilyPad Arduino, ATmega328P on COM13'.

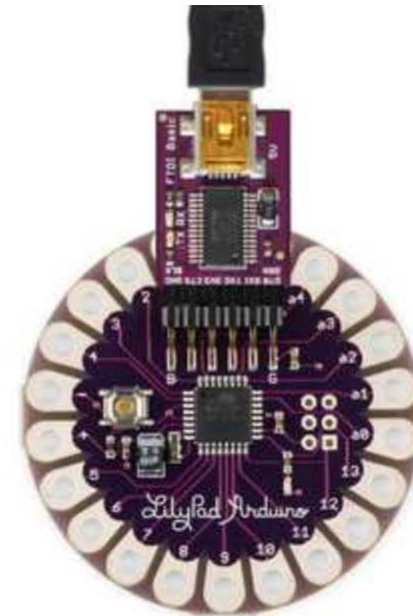
After installation and on launching the software you will be presented with an empty sketch. Programs in Arduino are called sketches



Lilypad 328 Main Board Setup

Lilypad 328 Main Board setup

1. Connect the FTDI breakout board to the LilyPad 328
2. Connect the breakout board to your computer using the USB cable.
3. An FTDI driver needs to be downloaded – one that is compatible with your operating system. The full list is here: <https://ftdichip.com/drivers/vcp-drivers/>
4. Installation guides for your operating system can be found here <https://ftdichip.com/document/installation-guides/>

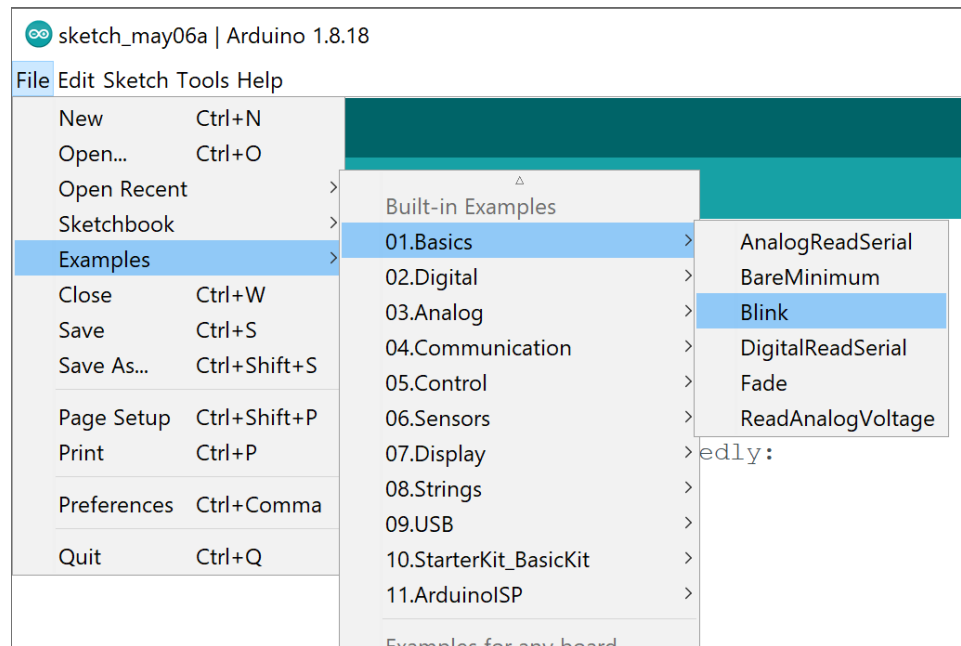


This is a complete getting started guide <https://docs.arduino.cc/retired/getting-started-guides/ArduinoLilyPad>

Uploading your first sketch

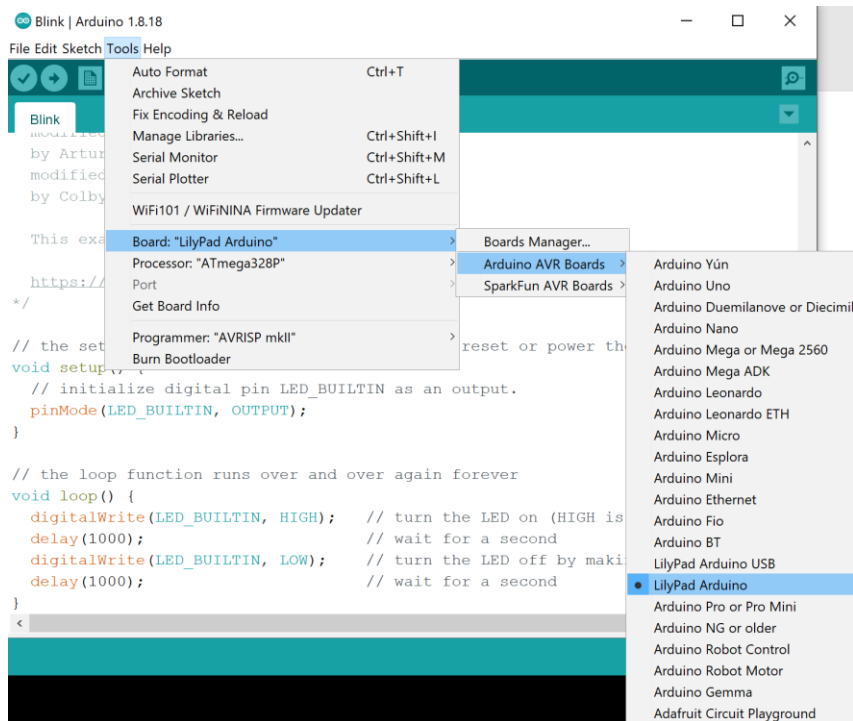
Go to **File > Examples > 01.Basics > Blink**

The Blink example sketch should open

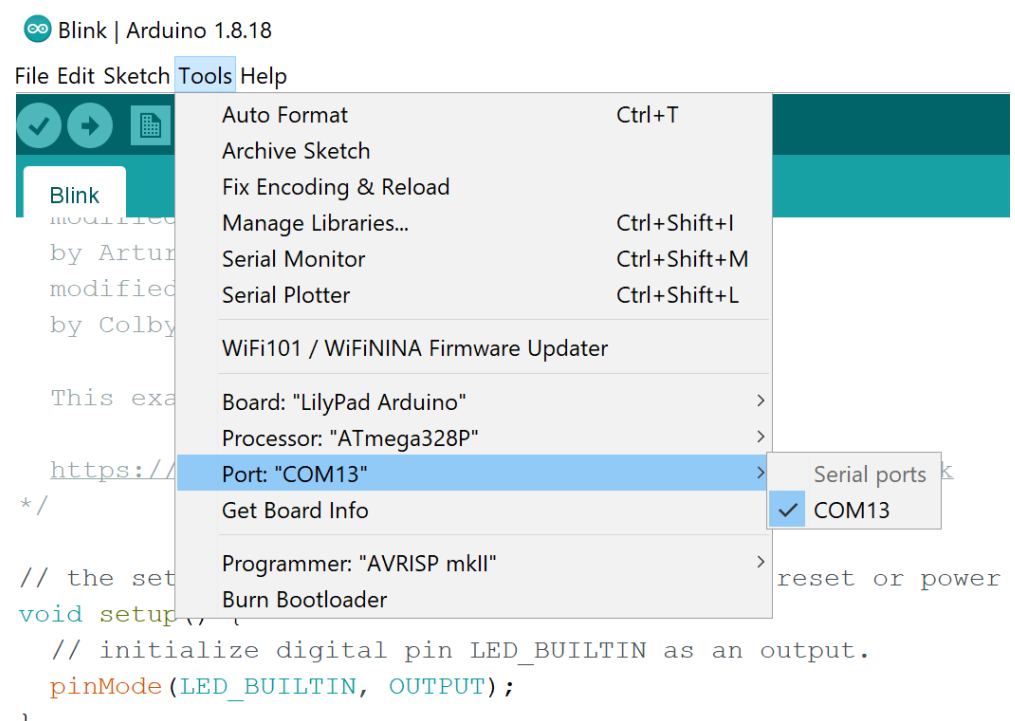


Uploading your first sketch

Select the board type under **Tools > Board**



Selected the COM port under **Tools > Port**



Uploading your first sketch

Click the verify button (this will check for code errors)



Click the upload button



Once uploaded the onboard LED should blink



```
File Edit Sketch Tools Help

Blink $

/*
 * Blink
 *
 * Turns an LED on for one second, then off for one second, repeatedly.
 *
 * Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
 * it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
 * the correct LED pin independent of which board is used.
 * If you want to know what pin the on-board LED is connected to on your Arduino
 * model, check the Technical Specs of your board at:
 * https://www.arduino.cc/en/Main/Products
 *
 * modified 8 May 2014
 * by Scott Fitzgerald
 * modified 2 Sep 2016
 * by Arturo Guadalupi
 * modified 8 Sep 2016
 * by Colby Newman
 *
 * This example code is in the public domain.
 * https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
 */

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}

Done compiling.
```


Blink Code: key things to note

`/*this is a comment */`

`// this is also a comment`

`void setup(){ }` is the function that runs once, when the Arduino starts up

`void loop(){ }` is the function that repeats forever, after the setup has run

Every opening curly brace { must have a closing curly brace }

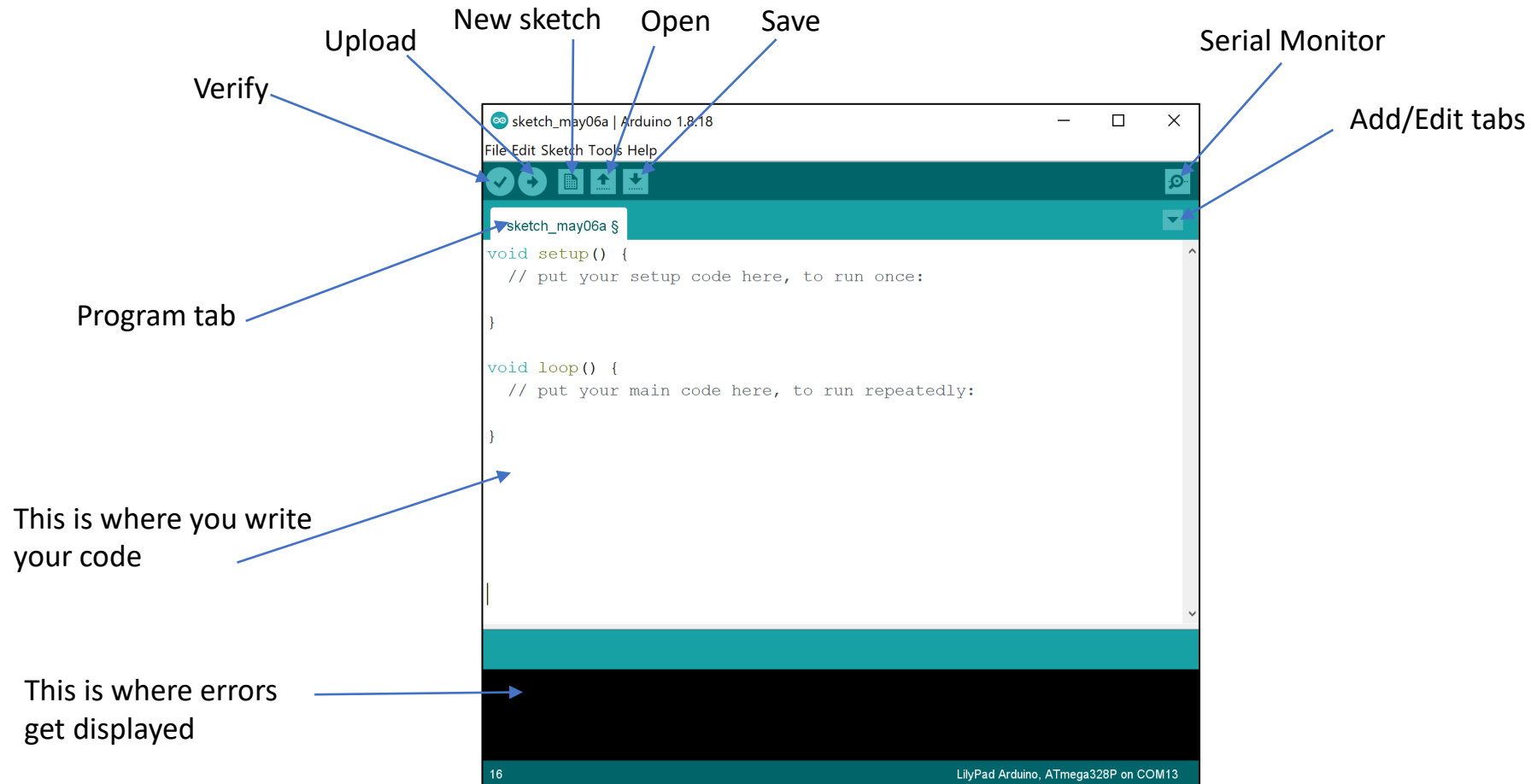
Every opening bracket (must have a closing bracket)

Every statement ends with a semicolon ;

`digitalWrite(pin, state);` sets a given pin high (5v) or low (0v). State can be HIGH or LOW, 1 or 0, true or false.

`delay(time);` waits for a given amount of time, in milliseconds.

Navigating the Arduino IDE

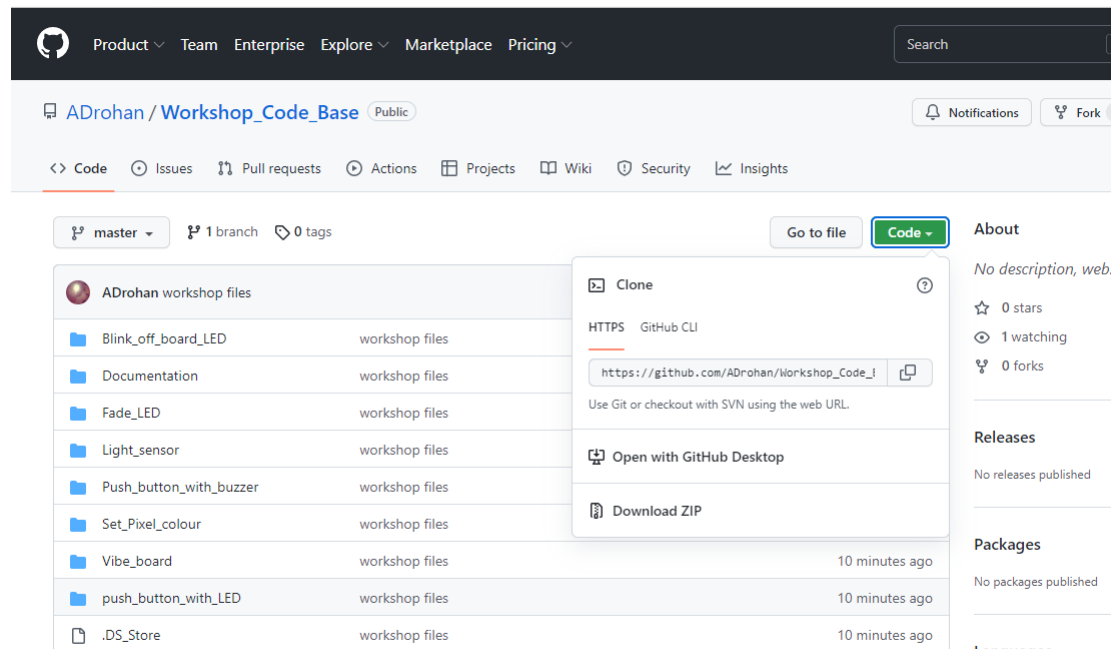




The Codebase

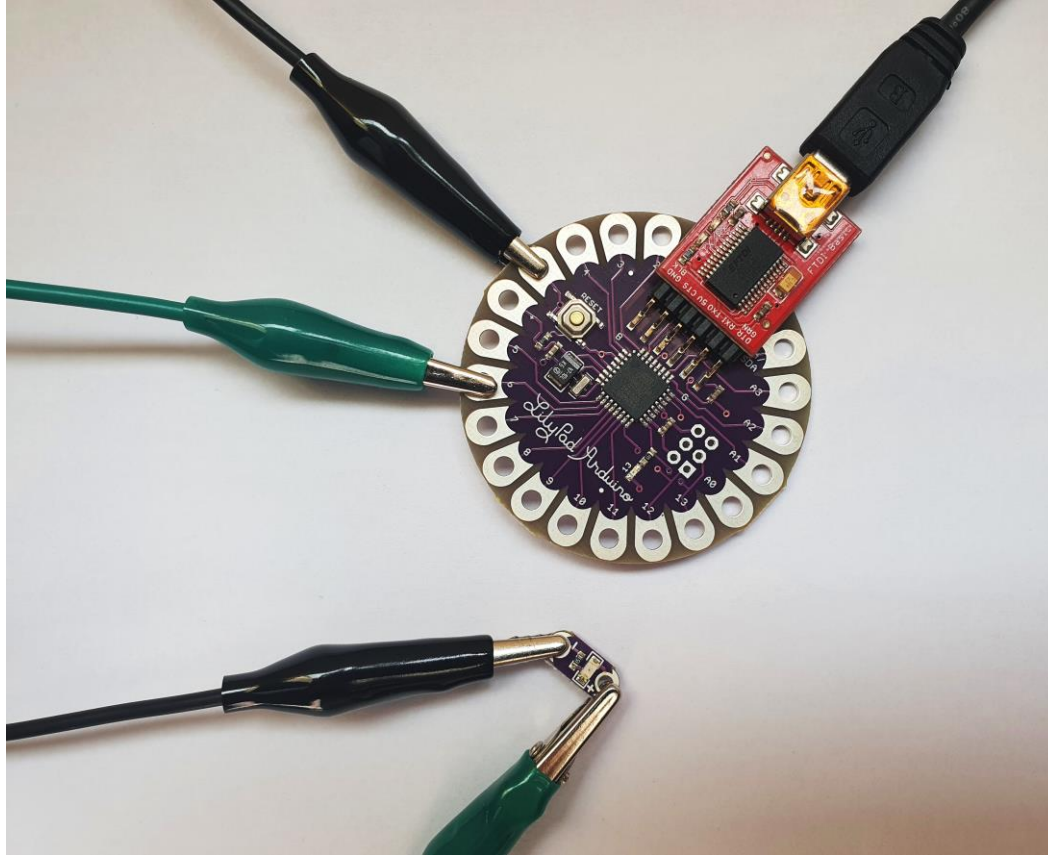
Download the Code Base

Go to : https://github.com/ADrohan/Workshop_Code_Base

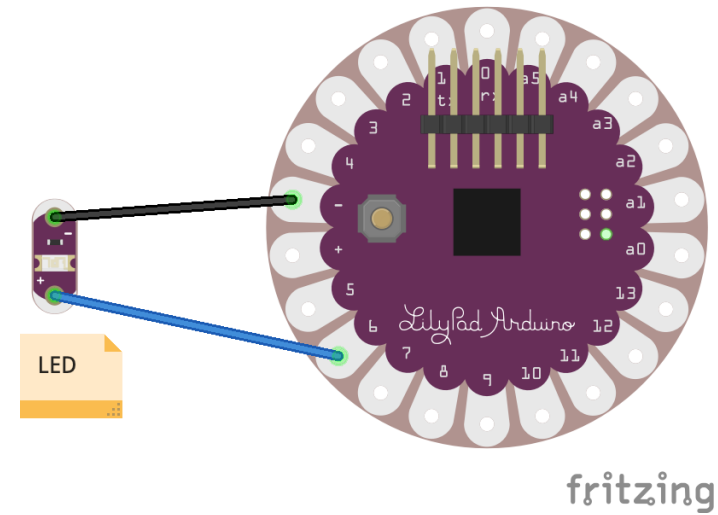


Click the green code button
and navigate to Download Zip

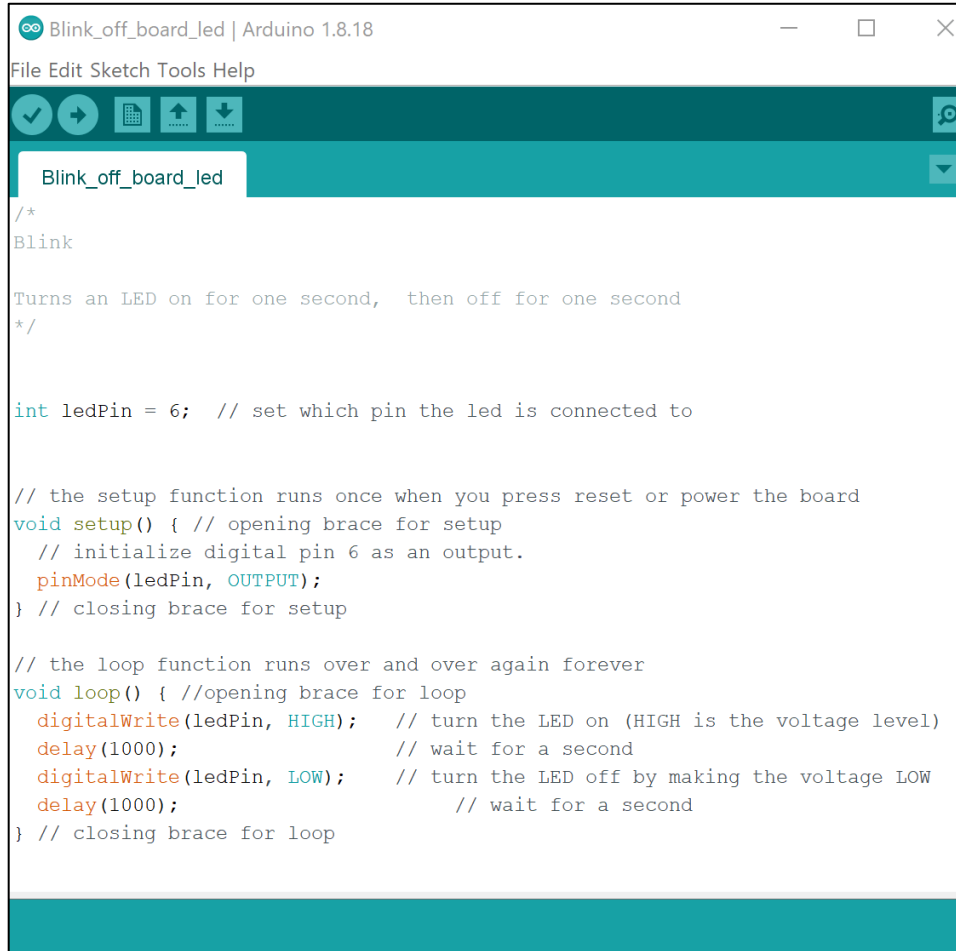
Attaching components with Crocodile leads



- Connect an led with Crocodile leads. Attach the led's – petal to the main boards – petal and the led's + petal to the main boards petal number 6.



A reworked Blink sketch turning an led on attached to petal 6.



```
Blink_off_board_led | Arduino 1.8.18
File Edit Sketch Tools Help

Blink_off_board_led

/*
Blink

Turns an LED on for one second, then off for one second
*/

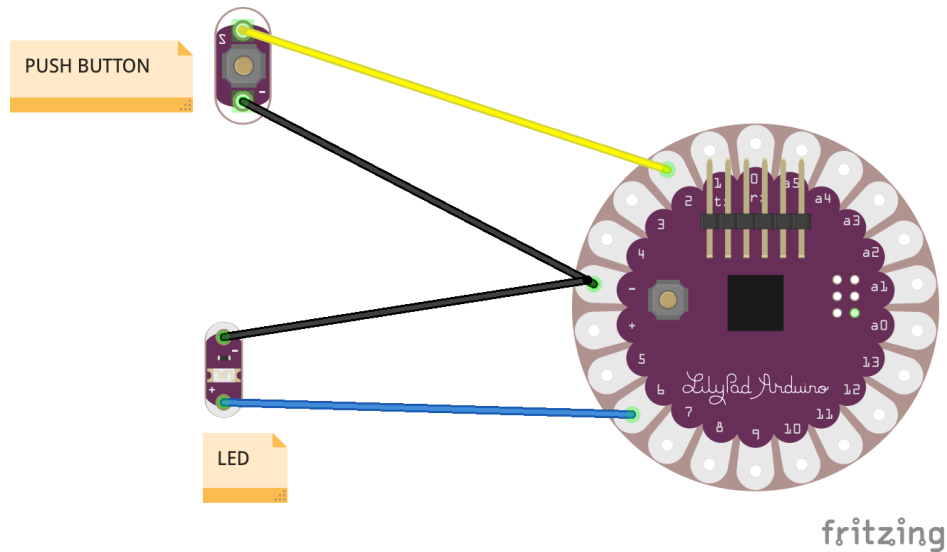
int ledPin = 6; // set which pin the led is connected to

// the setup function runs once when you press reset or power the board
void setup() { // opening brace for setup
  // initialize digital pin 6 as an output.
  pinMode(ledPin, OUTPUT);
} // closing brace for setup

// the loop function runs over and over again forever
void loop() { //opening brace for loop
  digitalWrite(ledPin, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
} // closing brace for loop
```

- Go to the code base you downloaded
- Open the sketch **Blink_off_board_LED.ino** This will be inside the Blink_off_board_LED folder.
- **Verify** with the verify button
- **Upload** with the upload button
- Your Led board should now be blinking instead of the onboard led as we had previously.

Add a Push Button



- Open the sketch **push_button_with_LED.ino**
- **Verify** with the verify button
- **Upload** with the upload button
- Your Led should now turn on when the push button is pressed

```
push_button_with_LED | Arduino 1.8.18
File Edit Sketch Tools Help

push_button_with_LED

/*
Push button code
When the button is pressed the led turns on. Otherwise the led is off
*/

const int ledPin = 6;      //The LED is connected to pin 6 - this is a PWM pin
const int buttonPin = 2;   //The Button is connected to pin 2
int buttonState = 0;       // This is a variable to store the button's state

void setup() {
  Serial.begin(9600);       // required to output to the serial monitor
  pinMode (ledPin, OUTPUT); //Set the LED pin as an output
  pinMode (buttonPin, INPUT_PULLUP); //Set button as input pulling it HIGH
}

void loop() {
  buttonState = digitalRead(buttonPin); // read the state of the button pin
  Serial.println(buttonState);          // out put the state to the serial monitor

  //check if the button is pressed
  if (buttonState == LOW) // if it is the button state is LOW
  {
    digitalWrite(ledPin, HIGH); // turn the led on
  }
  else
  {
    digitalWrite(ledPin, LOW); // or else turn the led off
  }
}
```



Circuit Testing

Multimeters and continuity

What is a multimeter?

A multimeter is a measuring instrument that can measure multiple electrical properties.

What is Continuity?

Continuity works by poking a little voltage into the circuit and seeing how much current flows. Continuity test with a multimeter when your circuit is NOT powered.

Testing continuity with a multimeter. What is it good for?

- Determine if your conductive wire/thread connections are good
- Determine if a conductive pathway is broken
- Making sure something isn't connected



<https://learn.adafruit.com/multimeters>

Continuity testing

How to continuity test with a digital multimeter

- Make sure the circuit is unpowered before testing
- Turn the dial to the continuity setting. Look for the sound symbol on your multimeter.
- Connect the black lead to the common ground terminal (COM)
- Connect the red lead to the mAΩ terminal (note: there may be a slight variation in the naming of this with different multimeters)
- Touch the two probes together (i.e. the metallic parts) You should hear an audible beep. You're set to start testing.
- Probe two points in the circuit you want to test for continuity.
- If the multimeter detects continuity, an audible beep will occur. If there is no continuity no beep will occur. This means your circuit is not connected electrically.



Continuity sound symbol

<https://learn.adafruit.com/multimeters>

Resources:

- Lilypad parts supplier: <https://www.mouser.ie/c/?series=LilyPad>
- Conductive Thread and fabric suppliers:
<https://serigraf.ie/shop/category/embroidery-madeira-thread-specialists-thread-220>
<https://www.bart-francis.be/en/search?query=conductive>
<https://www.mouser.ie/c/?q=adafruit%20conductive%20fabric>
https://www.shieldex.de/en/products_categories/fibers-yarns/
https://www.shieldex.de/en/products_categories/fabrics/
- E-textile How To Guides:
<http://thesoftcircuitteer.net/>
<https://www.kobakant.at/DIY/>
<https://www.instructables.com/How-To-videos-for-eTextiles-soft-circuits-and-we/>
<https://www.youtube.com/c/Etextilelounge/playlists>
- Lilypad tutorials: <https://www.sparkfun.com/search/results?term=lilypad>



Thank you

Aileen Drohan

aileen.drohan@waltoninstitute.ie