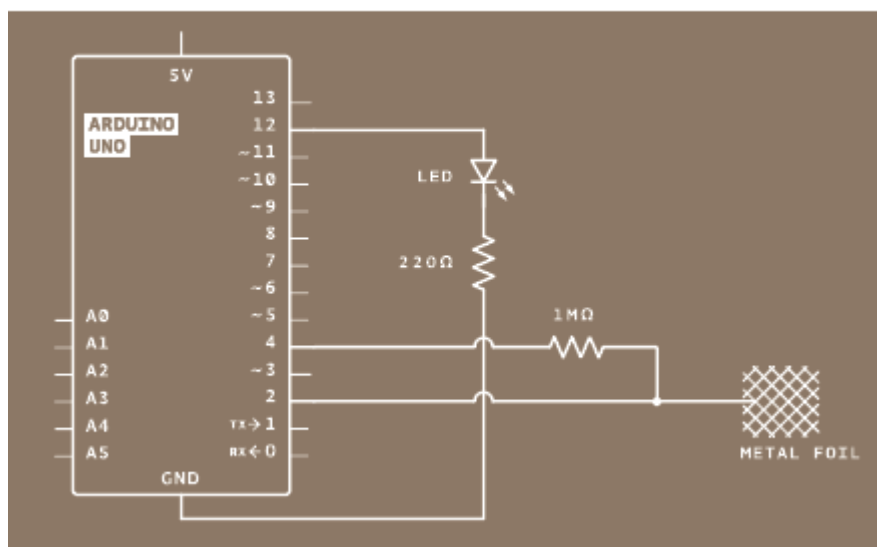
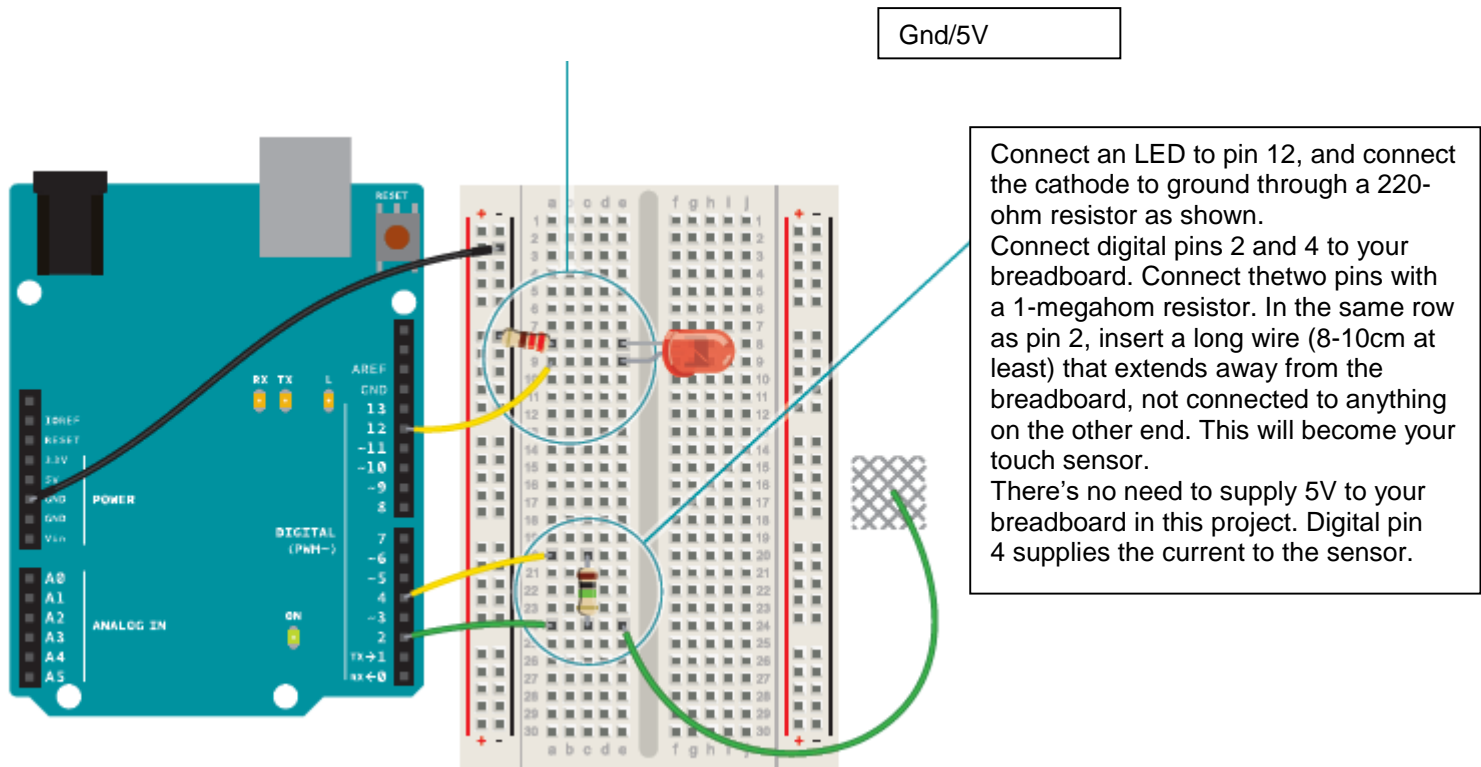


13. Touchy-feel lamp

You'll be using the CapacitiveSensor library by Paul Badger for this project. This library allows you to measure the capacitance of your body.

Capacitance is a measure of how much electrical charge something can store. The library checks two pins on your Arduino (one is a sender, the other a receiver), and measures the time it takes for them to have the same state. These pins will be connected to a metal object like aluminum foil. As you get closer to the object, your body will absorb some of the charge, causing it to take longer for the two pins to be the same.



```

// import the library (must be located in the
// Arduino/libraries directory)
#include <CapacitiveSensor.h>

// create an instance of the library
// pin 4 sends electrical energy
// pin 2 senses a change
CapacitiveSensor capSensor = CapacitiveSensor(4, 2);

// threshold for turning the lamp on
int threshold = 1000;

// pin the LED is connected to
const int ledPin = 12;

void setup() {
  // open a serial connection
  Serial.begin(9600);
  // set the LED pin as an output
  pinMode(ledPin, OUTPUT);
}

void loop() {
  // store the value reported by the sensor in a variable
  long sensorValue = capSensor.capacitiveSensor(30);

  // print out the sensor value
  Serial.println(sensorValue);

  // if the value is greater than the threshold
  if (sensorValue > threshold) {
    // turn the LED on
    digitalWrite(ledPin, HIGH);
  }
  // if it's lower than the threshold
  else {
    // turn the LED off
    digitalWrite(ledPin, LOW);
  }

  delay(10);
}

```

At the beginning of your program, include the CapacitiveSensor library. You include it the same way you would a native Arduino library like the Servo library in the earlier projects.

Create a named instance of the library.

When you use this library, you tell the instance what pins it will be using to send and receive information. In this case, pin 4 sends to the conductive sensor material through the resistor, and pin 2 is the sense pin. Set up a variable for the sensing threshold at which the lamp will turn on. You'll change this number after you test the sensor's functionality.

Then define the pin your LED will be on.

In the setup() function, open a Serial connection at 9600 bps.

You'll use this to see the values the sensor reads. Also, make your ledPin an OUTPUT.

In the `loop()` function, create a variable of type `long` to hold the sensor's value. The library returns the sensor value using a command called `CapacitiveSensor()` that takes an argument identifying the number of samples you want to read. If you read only a few samples, it's possible you'll see a lot of variation in the sensor. If you take too many samples, you could introduce a lag as it reads the sensor multiple times. 30 samples is a good starting value. Print the sensor value to the serial monitor.

With an `if()...else` statement, check to see if the sensor value is higher than the threshold. If it is, turn the LED on. If it is not, turn it off.

Then add a small `delay()` before ending the `loop()`.

After programming the Arduino, you'll want to find out what the sensor values are when it's touched. Open the serial monitor and note the value coming from the sensor when you're not touching it. Press gently on the bare wire you have exposed from your breadboard. The number should increase. Try pressing more firmly and see if it changes. Once you have an idea of the range of values you're getting from the sensor, go back to the sketch and change the threshold variable to a number that is greater than the sensor's value when it is not touched, but less than its value when pressed. Upload the sketch with the new value. The light should come on reliably when you touch the wire, and turn off when it's left alone. If you aren't getting the light to turn on, try lowering the threshold a little more.