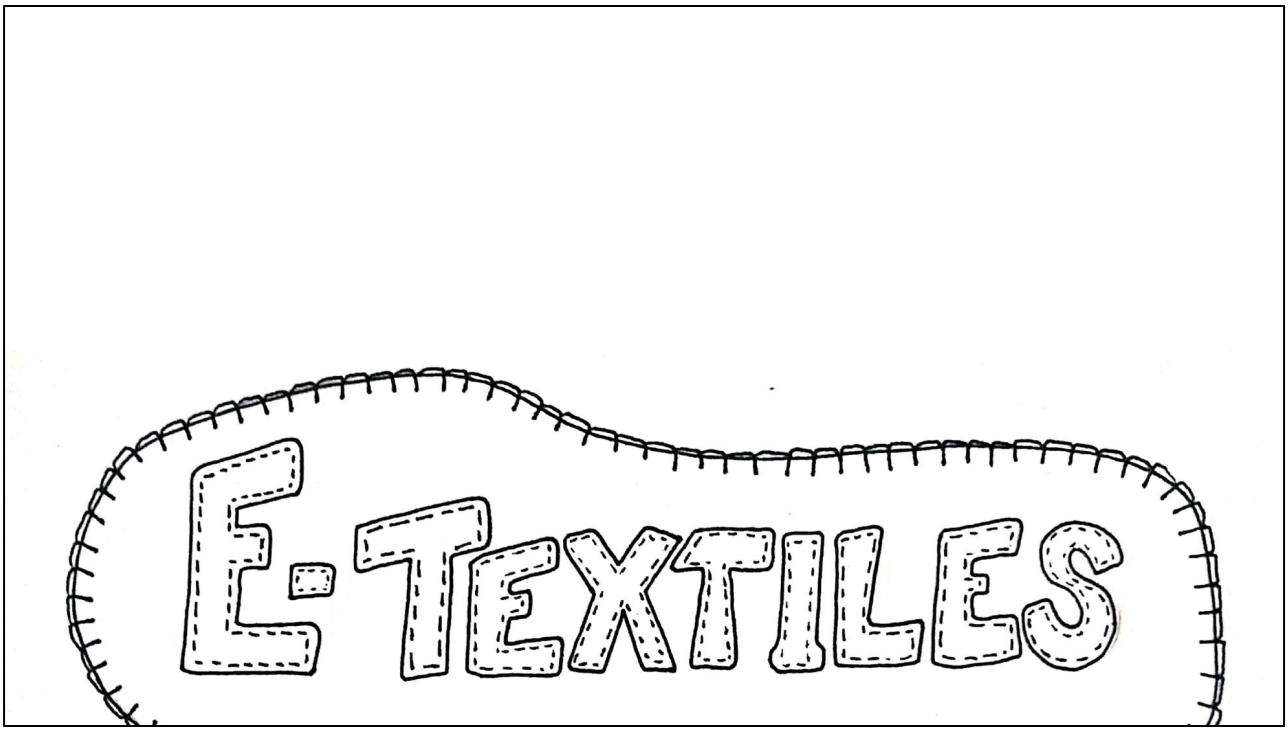
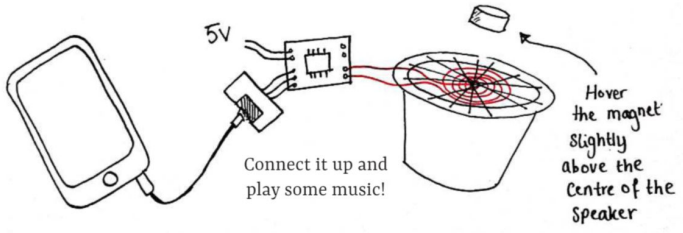
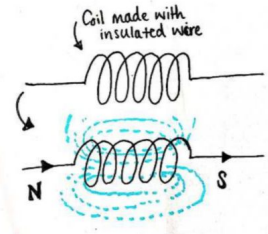
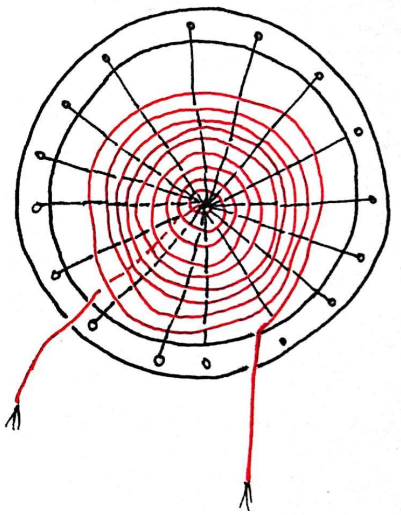

E-TEXTILES

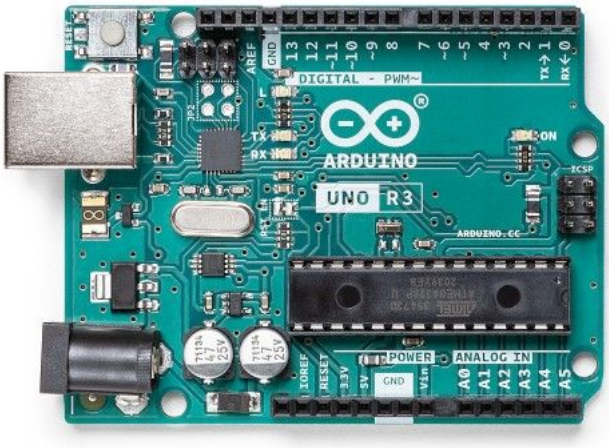




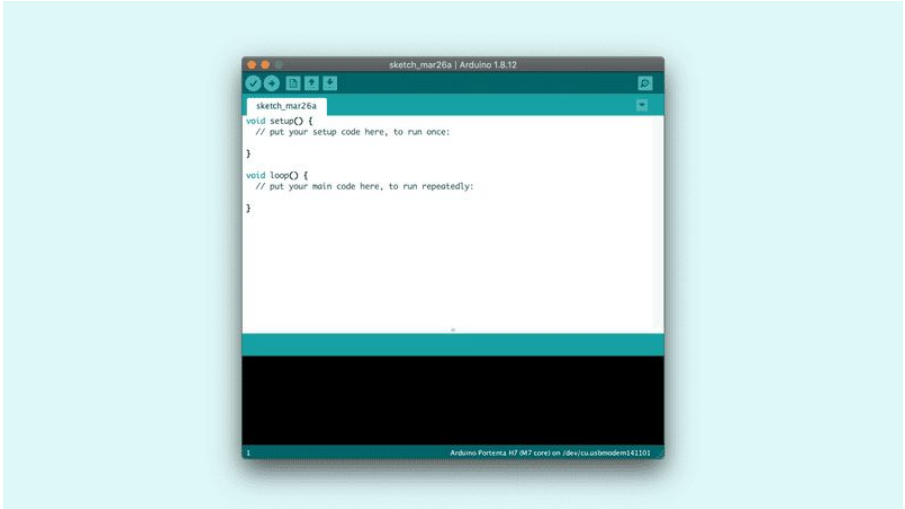


Recap over Day 5: Woven speaker and use of an electromagnet.

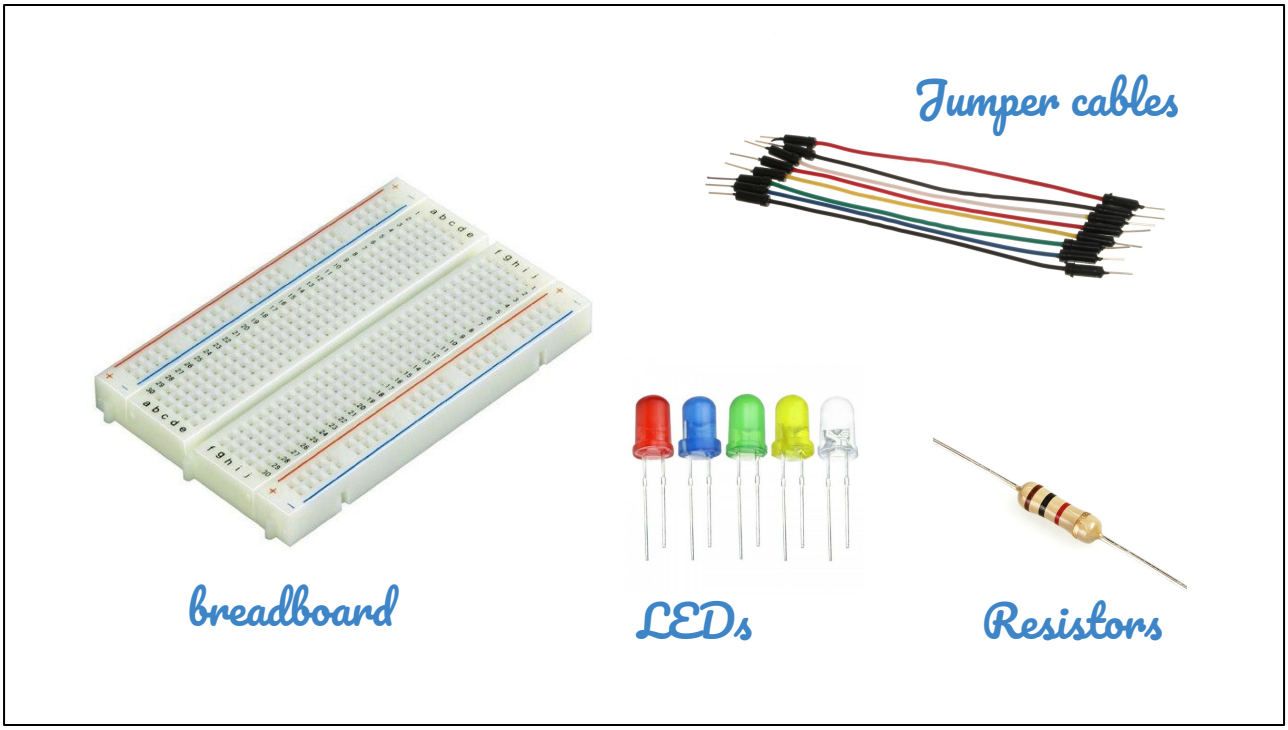
Day 6: Introduction to Arduino



This is an Arduino - A micro computer.
It allows you to programme circuits.
On the Arduino there are different ports down each side. You can connect components and circuits such as LEDs, sensors, motors, buzzers etc.
The Arduino is able to take INPUTs from these components (such as a sensor) and create an output (such as an LED lighting up). You can programme them to effect one another, such as an LED lighting up brighter as the temperature increases.

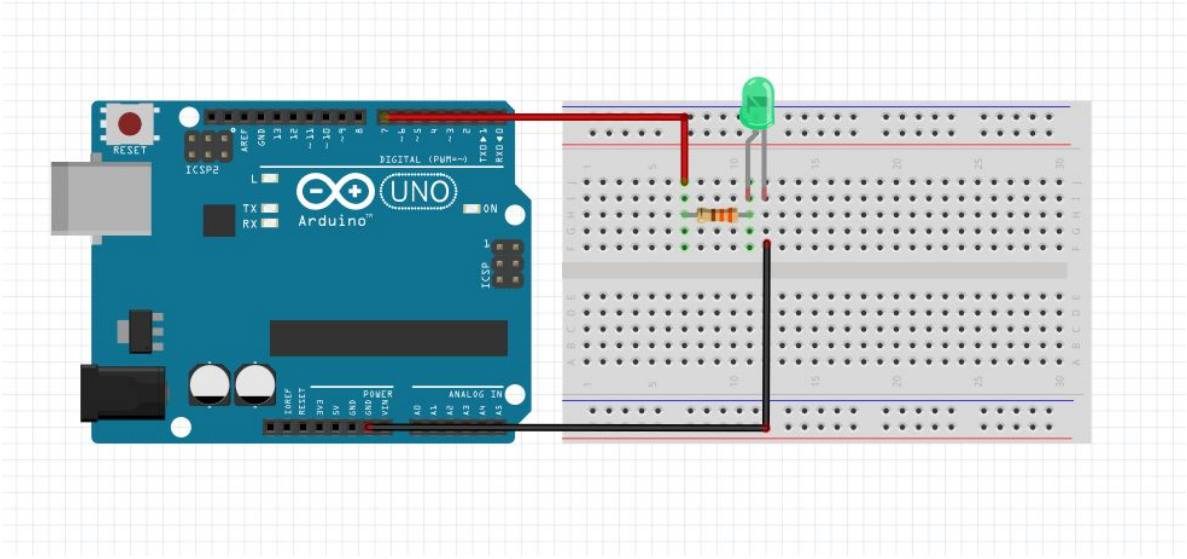


To programme your Arduino you need to use the Arduino IDE software.
This is a programme that allows you to write your code. It then compiles it into a format that the Arduino can understand and sends it to your connected arduino.
You can install Arduino IDE, for free online. Just go on to the Arduino website and click on the software tab > download.



For the electronic prototyping (temporary circuits that can be used with you Arduino to make sure the code runs correctly) you need the following equipment.
Breadboard: the object that allows you to create circuits without permanent connections. By entering the component into the board, where components are connected along the horizontal line and not connected along the vertical line.
Jumper cables: used with the breadboard to form connections with components and the arduino
LEDs and Resistors: we previously discussed on Day 2

Blinking LED



The first of the activities to try out, if you are new to Arduino is the blinking LED.
It is important to start with very simple tasks to make sure you have a real understanding of what is happening and refresh your memory about the language we use.
All the links for these activities are given in the tutorial book on the Introduction to Arduino page

Code

```
int led = 7;
void setup() {
pinMode(led, OUTPUT);
}

void loop() {
digitalWrite(led, HIGH);
delay(1000);
digitalWrite(led, LOW);
delay(1000);
}
```

This is the code that is used to make sure the LED is able to blink.
It is important to be able to go through each line of code and understand what is happening

Code

int

pinMode(_)

digitalWrite(_)

delay(_)

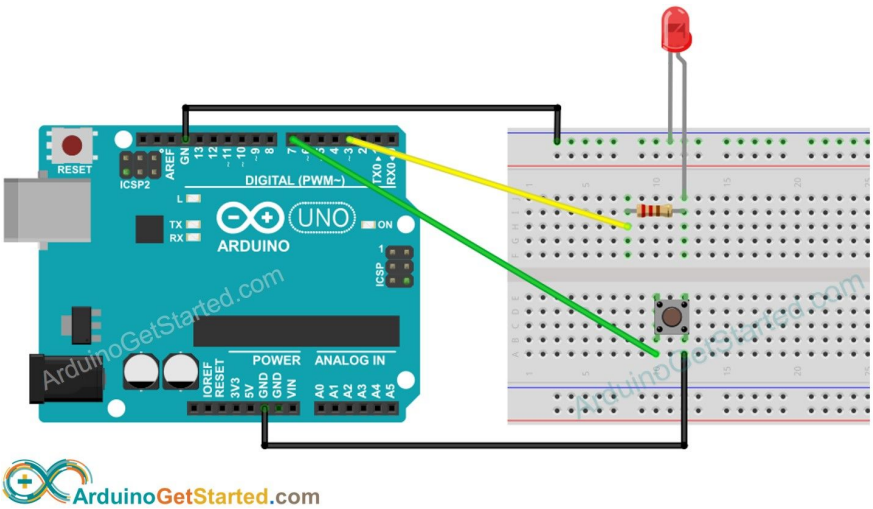
These are the main functions we should understand.
int = means we are creating a variable that is an integer. An integer is a whole number, such as 1, 2 or 3. Integers can not take a value that is a decimal. It will be rounded up or down to the closest whole number.

pinMode(_) = this command is used to define how a pin will behave when a component is connected. The pin can be receiving an INPUT or be providing an OUTPUT. For each pin being used on your Arduino, you need to define how it is behaving.

digitalWrite(_) = this is used when you pin is connected to a component which is providing an output. Use digitalWrite to tell it what action to do. Here, it is used to tell the LED to light up

delay(_) = the delay command is needed as the arduino goes through the code incredibly fast. We need to make sure it pauses after the LED is turned on, to make sure that the light stays on long enough so that it is visible to us.

Button controlled LED



The next activity we always look at is the button controlled LED.
This is a slightly more complex activity where we are introducing a button which provides an INPUT signal.

Code

```
const int BUTTON_PIN = 7;  // the number of the pushbutton pin
const int LED_PIN =  3;   // the number of the LED pin

int buttonState = 0;   // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(LED_PIN, OUTPUT);
  // initialize the pushbutton pin as an pull-up input:
  // the pull-up input pin will be HIGH when the switch is open and LOW
when the switch is closed.
  pinMode(BUTTON_PIN, INPUT_PULLUP);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(BUTTON_PIN);

  // control LED according to the state of button
  if(buttonState == LOW)         // If button is pressing
    digitalWrite(LED_PIN, HIGH); // turn on LED
  else                           // otherwise, button is not pressing
    digitalWrite(LED_PIN, LOW);  // turn off LED
}
```

Here is the code that is used - it is important to go through every line to make sure you understand what is happening

Code

```
const int

int

void setup() {
  pinMode(_,OUTPUT);

  pinMode(_,INPUT);
}

void loop() {
  digitalRead(_);

  if( == )
digitalWrite(_)

  else
digitalWrite(_)
}
```

These are the main commands that we can extract. let's have a look in more detail…

const int = by stating 'const' before 'int' this stands for 'constant' and means that the integer value that this variable has been given can not change at any point during the code. It is used for defining the pin numbers, as this will not change

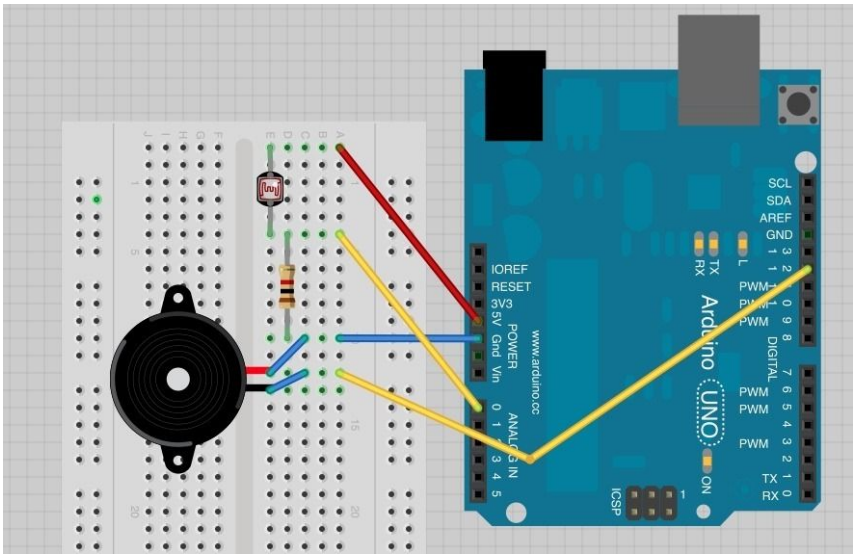
pinMode(_) = In this code we can see that INPUT and OUTPUT pins have been defined using this command. LEDs providing an OUTPUT and the buttons providing an INPUT.

digitalRead(_) = this is when a component is providing an input signal (such as the button). The arduino need to 'read' the signal coming from this pin. Here we are asking the Arduino to 'read' the signal being provided by the button.

if/else = in this code we use an 'if' and 'else' clause. This means that if the condtion in the brackets is met, the Arduino should follow this first part of the code. But if the condition is not met, the arduino will skip this section of the code and instead complete the code which follows the 'else' command.

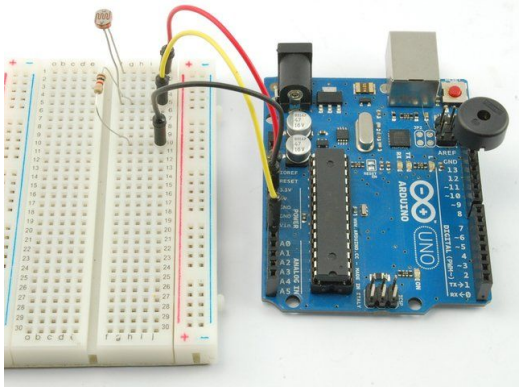
'==' this notation is used when we want to say that it has the value defined.
if three == 3
Means that the following code would happen if the variable 'three' has the value 3.

Theremin

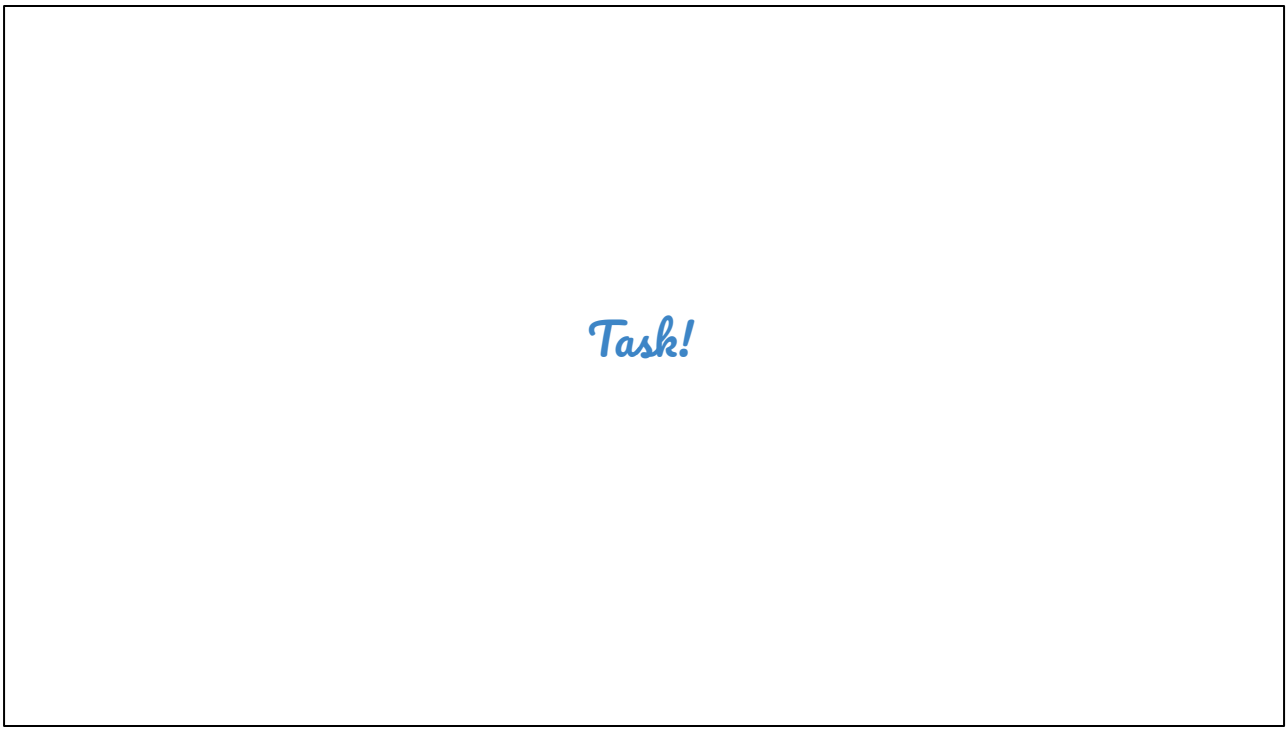


The final activity to try out and fully understand is the Theremin activity.
This is slightly more complex in terms of building a circuit. But the code remains very simple.

Theremin



It looks like this!



Find the details of the Task for day 6 in the Huge academy platform.