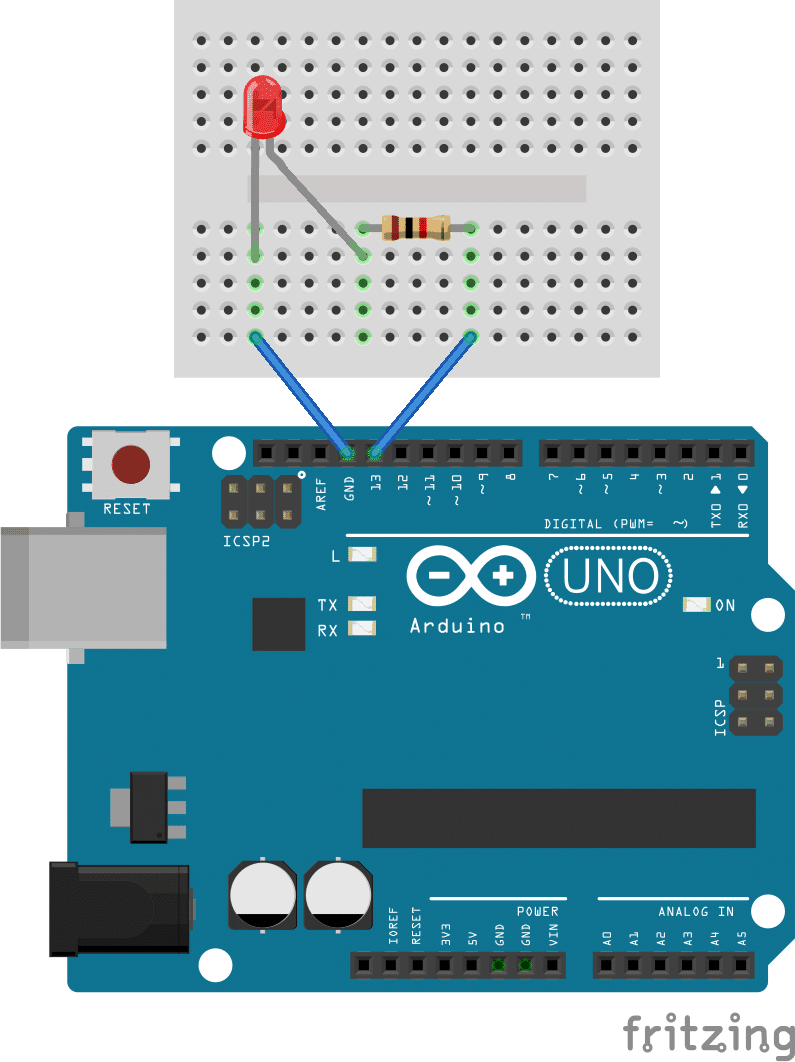
**Project**

**Lights:**

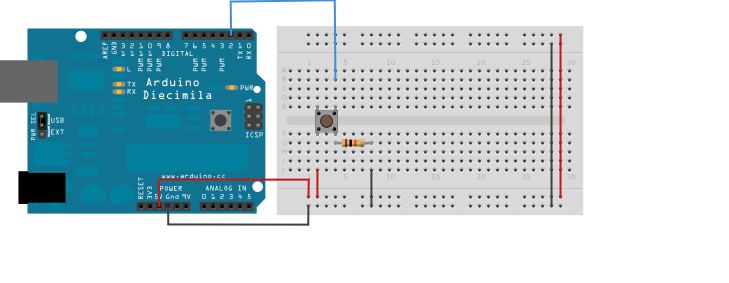
the LED’s shortest lead connects to the ground side. Use blink examples sketch.



**BUTTON:**

<https://www.arduino.cc/en/tutorial/button>

make button press



**Arduino Board Upload:**

\*\* Make sure node red is closed before uploading

<http://firmata.org/wiki/Main_Page>

File > examples > firmata > standard firmata

**Node Red:**

Command prompt to open: node-red

To close: Ctrl+c

Server: <http://127.0.0.1:1880/>

FLOW 1

Inject timestamp

Go to <https://home.openweathermap.org/>

Sign in to account, and click on account. Go to api keys tab and use WeatherKey

Change time according inside the SWITCH NODE

…

Added some nodes: <https://iotdesignpro.com/projects/interface-arduino-with-node-red-to-send-sensor-data-on-webpage>

* To **install the Arduino nodes**

**npm install node-red -node-arduino**

* To **install the serial port node**

**npm install node-red -node-serialport**

* To **install the Dashboard nodes**

**npm install node-red –dashboard**

* + serial is under network section of nodes
  + Arduino in, out is under Arduino section and Arduino board is under dashboard

To install the stable version use the Menu - Manage palette - Install option

run npm uninstall to uninstall any additions

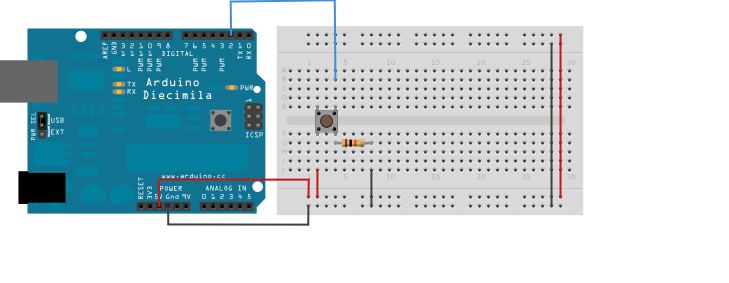
**Resources:**

[**Ohm's Law**](http://en.wikipedia.org/wiki/Ohm%27s_law)

**The Push Button Test:**

<https://www.arduino.cc/en/Tutorial/DigitalReadSerial>

Upload Digital read serial and use the following circuit: (can use any digital side pin, just change the code)



**Taking input from a user:**

See example SerialEvent in Arduino examples

Check if digit:

<https://www.arduino.cc/en/Tutorial/CharacterAnalysis>

Call serial once:

<https://forum.arduino.cc/index.php?topic=82603.0>

Serial inputs:

<https://www.norwegiancreations.com/2017/12/arduino-tutorial-serial-inputs/>

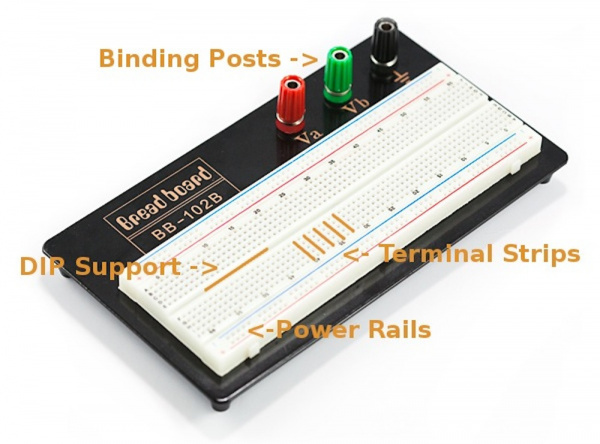
**Calling Functions:**

<https://www.arduino.cc/en/Reference/FunctionDeclaration>

**Current flow of a breadboard:**

<https://forum.arduino.cc/index.php?topic=305100.0>

<https://learn.sparkfun.com/tutorials/how-to-use-a-breadboard/all>



Terminal strip : components placed will be electrically connected to anything else placed in that row.

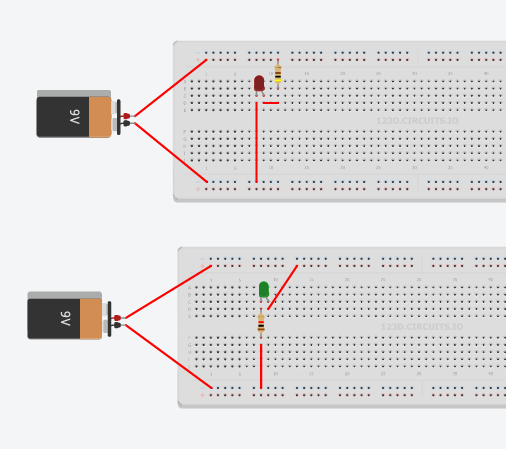
DIP Support: This ravine isolates both sides of a given row from one another, and they are not electrically connected.

* Many [integrated circuits](https://learn.sparkfun.com/tutorials/integrated-circuits), often referred to as ICs or, simply, chips, are manufactured specifically to fit onto breadboards.
* These DIP chips (salsa anyone?) have legs that come out of both sides and fit perfectly over that ravine. Since each leg on the IC is unique, we don’t want both sides to be connected to each other.
* we can connect components to each side of the IC without interfering with the functionality of the leg on the opposite side.

Power Rails: These power rails are metal strips that are identical to the ones that run horizontally, except they are typically all connected. The power rails give you lots of easy access to power wherever you need it in your circuit.



If I connect power to a terminal strip, the whole row has power (shown above). This is why placing an LED or other component that needs both POWER and GROUND for each leg should NOT be placed along the same terminal. They should be placed across terminals.



Above are some examples with LEDs

**Speaker:**

<https://www.programmingelectronics.com/an-easy-way-to-make-noise-with-arduino-using-tone/>

<https://maker.pro/arduino/projects/arduino-speaker>