Reed Switch

- reed switches are triggered by a nearby magnet, and are “on” only when the magnet is present.
- reed switches have no polarity
- use a breadboard for connecting it to your Arduino

- note: there is an additional “ground” pin on the digital pin side of the board, used in this schematic. There is no difference to the other two “ground” pins.

*Try both programs shown below with your Arduino and learn about momentary and permanent switching, and debouncing.*
Sample Code: push-on-only

/* Basic Digital Read
 * ------------------
 * turns on and off a light emitting diode(LED) connected to digital
 * pin 13, when activating a reed switch attached to pin 7. It illustrates the
 * concept of Active-Low, which consists in connecting buttons using a
 * 1K to 10K pull-up resistor.
 *
 * Created 1 December 2005
 * copyleft 2005 DojoDave <http://www.0j0.org>
 * http://arduino.berlios.de
 *
 */

int ledPin = 13; // choose the pin for the LED
int inPin = 7; // choose the input pin
int val = 0; // variable for reading the pin status

void setup() {
    pinMode(ledPin, OUTPUT); // declare LED as output
    pinMode(inPin, INPUT); // declare as input
}

void loop(){
    val = digitalRead(inPin); // read input value
    if (val == HIGH) { // check if the input is HIGH (button released)
        digitalWrite(ledPin, LOW); // turn LED OFF
    } else {
        digitalWrite(ledPin, HIGH); // turn LED ON
    }
}

Source:

http://www.arduino.cc/en/Tutorial/Pushbutton
Sample Code: stay-on, stay-off
When using this type of switch for this kind of application, it needs to be debounced. Debouncing is important to understand in-depth: read up about it [HERE](http://www.arduino.cc/en/Tutorial/Switch).

```c
/* switch
 * Each time the input pin goes from LOW to HIGH
 * the output pin is toggled from LOW to HIGH or HIGH to LOW. There's
 * a minimum delay between toggles to debounce the circuit (i.e. to ignore
 * noise).
 *
 * David A. Mellis
 * 21 November 2006
 */

int inPin = 7;       // the number of the input pin
int outPin = 13;     // the number of the output pin
int state = HIGH;    // the current state of the output pin
int reading;         // the current reading from the input pin
int previous = LOW;  // the previous reading from the input pin

// the follow variables are long's because the time, measured in milliseconds,
// will quickly become a bigger number than can be stored in an int.
long time = 0;       // the last time the output pin was toggled
long debounce = 200; // the debounce time, increase if the output flickers

void setup()
{
  pinMode(inPin, INPUT);
  pinMode(outPin, OUTPUT);
}

void loop()
{
  reading = digitalRead(inPin);

  // if the input just went from LOW and HIGH and we've waited long enough
  // to ignore any noise on the circuit, toggle the output pin and remember
  // the time
  if (reading == HIGH && previous == LOW && millis() - time > debounce) {
    if (state == HIGH)
      state = LOW;
    else
      state = HIGH;

    time = millis();
  }

  digitalWrite(outPin, state);

  previous = reading;
}
```

Source: