

M A K E
M A P
B L I N K

MAKE

John Keefe

MAP

@jkeefe
johnkeefe.net
john@johnkeefe.net

BLINK

MAKE

MAP

BLINK

**Class No. 3
Hello, Arduino**

MAKE
MAP
BLINK

#MakeMapBlink

MAKE

MAP

BLINK

Attendance !
Text your name to
646-699-3993

MAKE MAP BLINK

**Get to Know Arduino
a microcontroller**

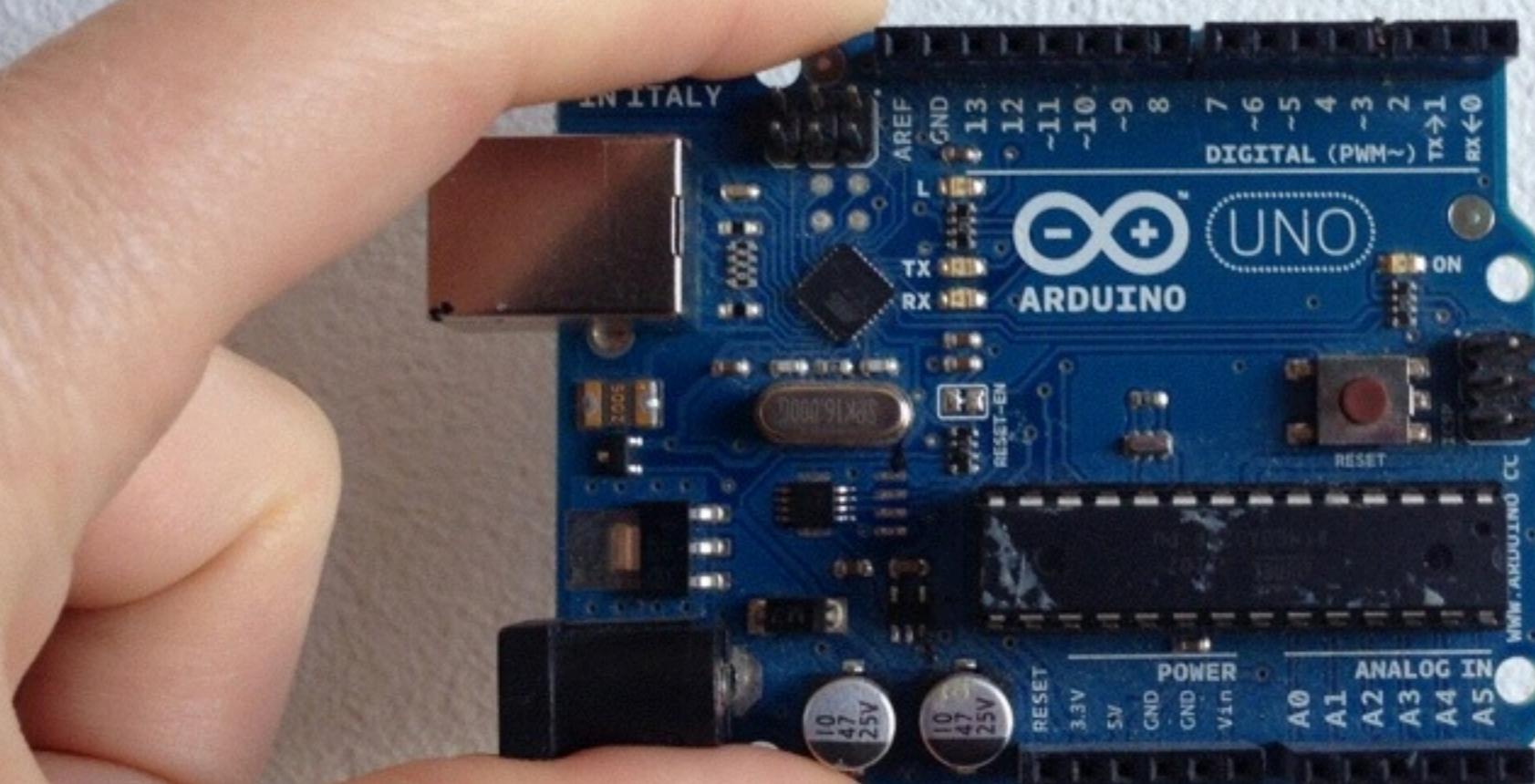
**MAKE
MAP
BLINK**

bit.ly/class-parts

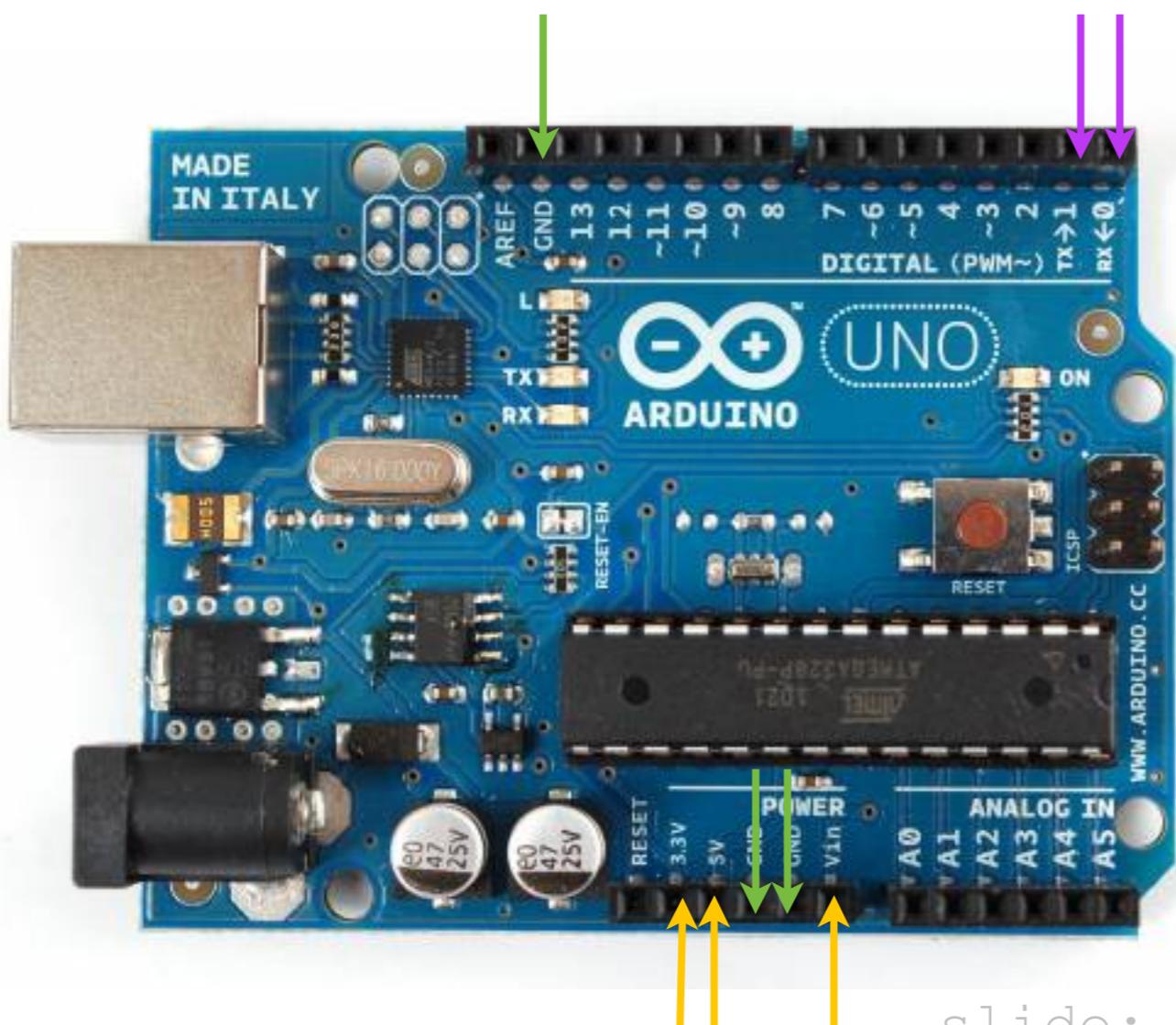
**Arduino Uno
USB A to B Cable**

\$30

3



a **pin** provides an input or output through which the controller can communicate with components.



TX/RX (serial - transmit/receive)

3 ground pins

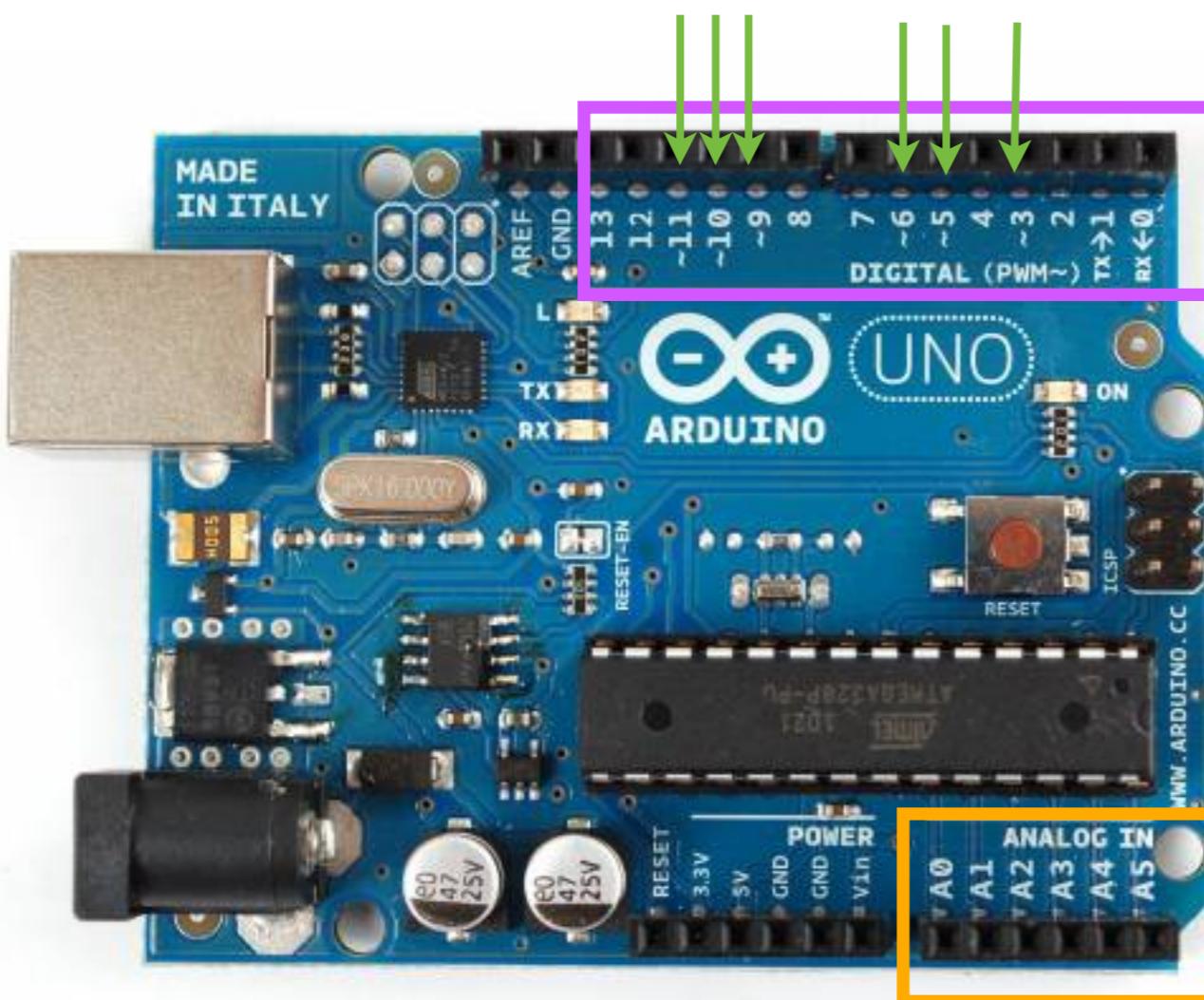
3 power pins

// 5 volts

// 3 volts

// VIN - can plug 9 volts here

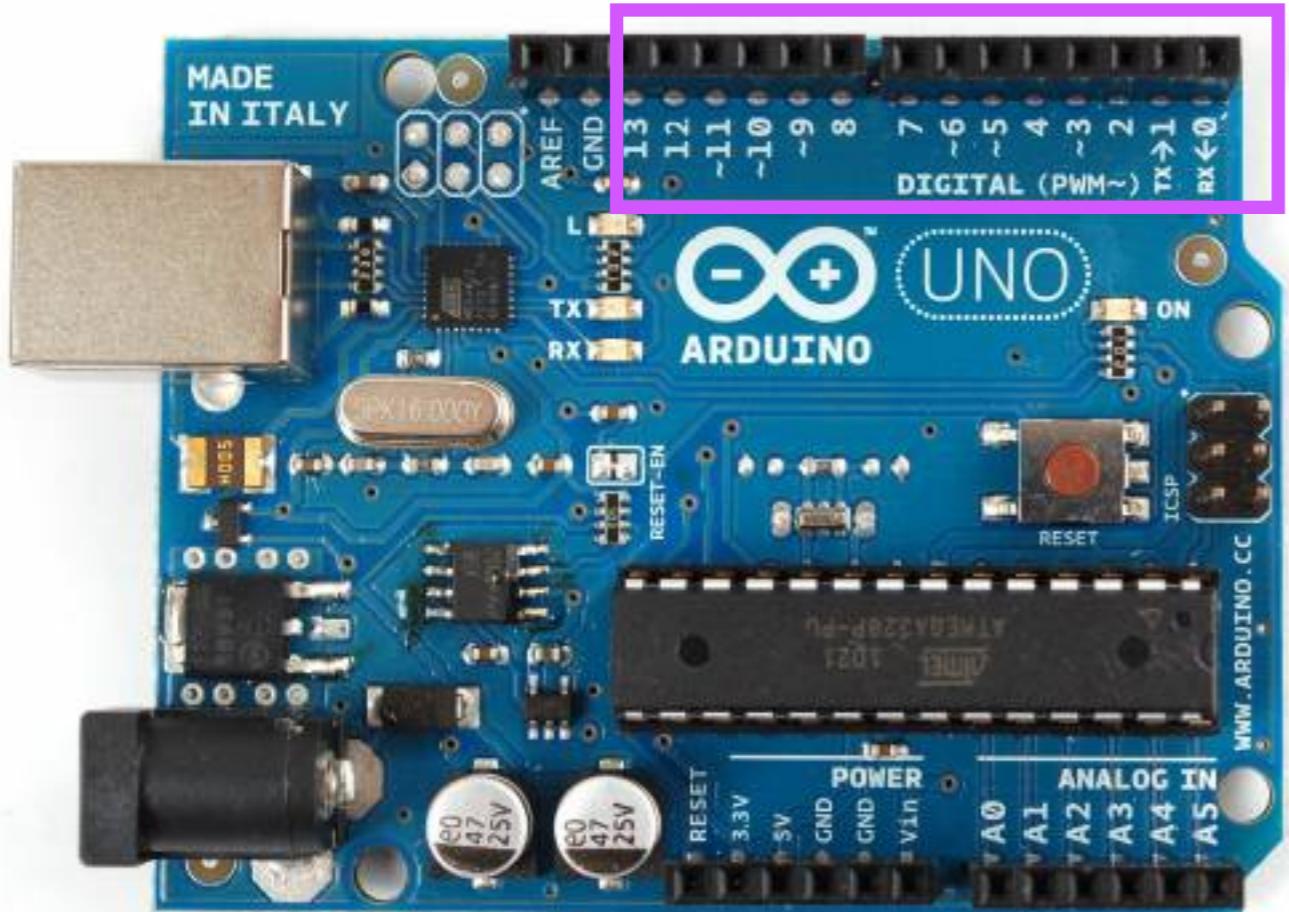
a **pin** provides an input or output through which the controller can communicate with components.



14 Digital pins

6 PWM pins (this means you can FADE LEDs)

6 Analog input pins



14 Digital pins

You can read or write 2 different values to them:

HIGH

5 volts

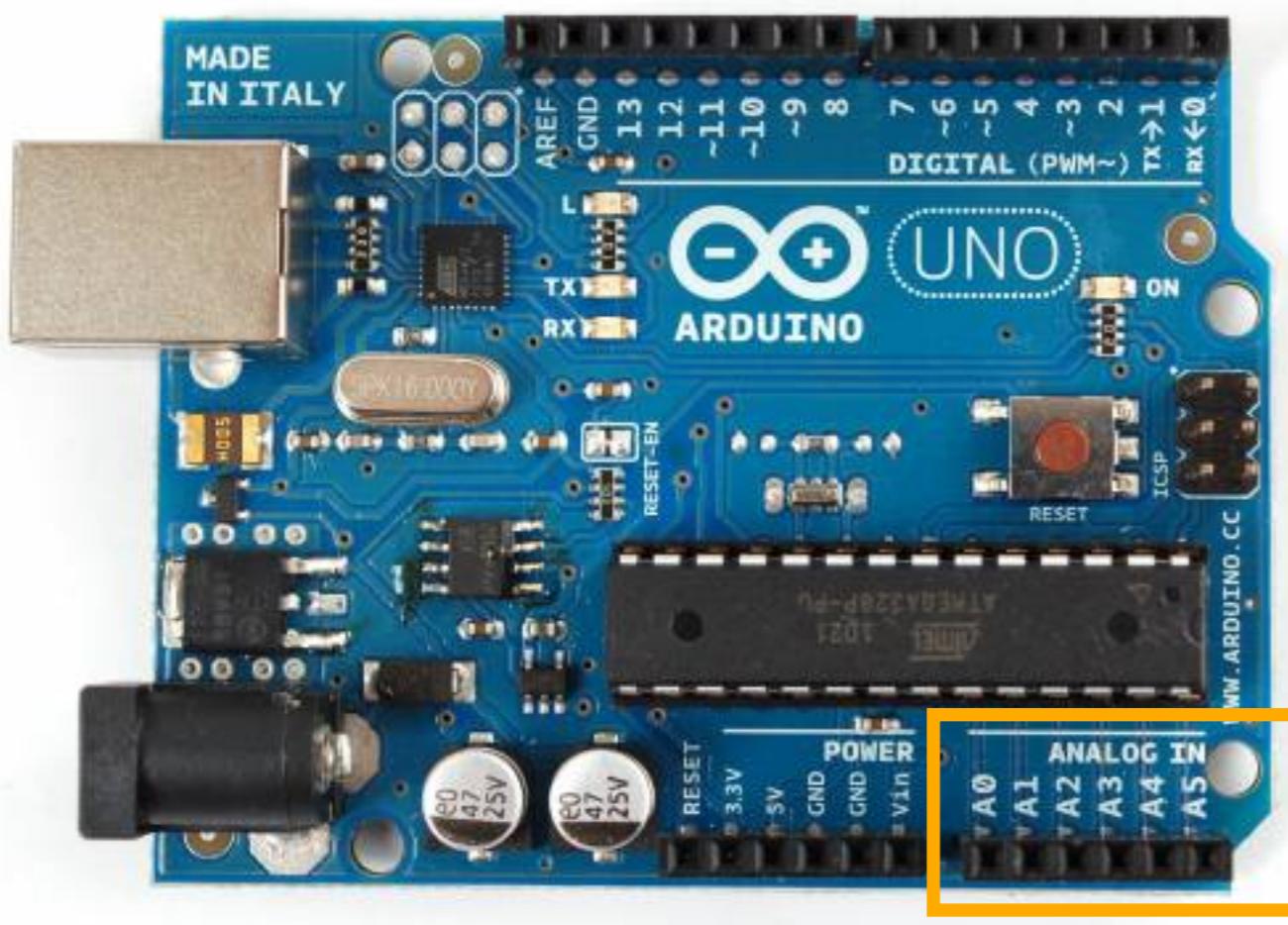
ON

LOW

0 volts

OFF

They can be configured as input OR output.



6 Analog Input pins

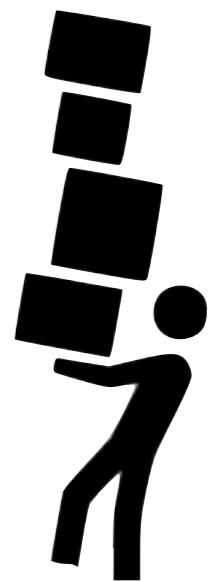
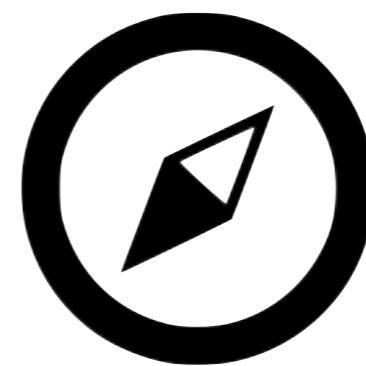
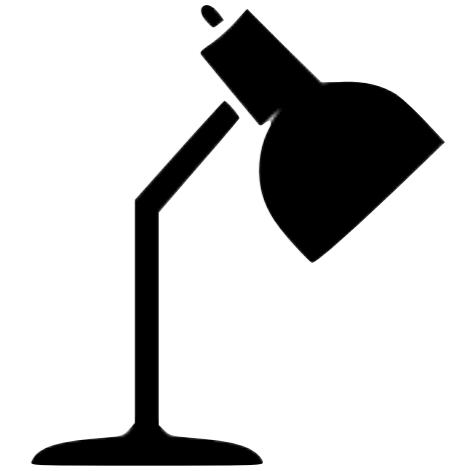
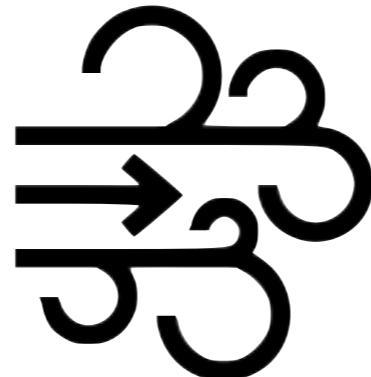
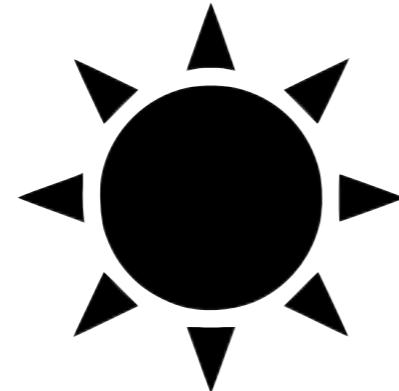
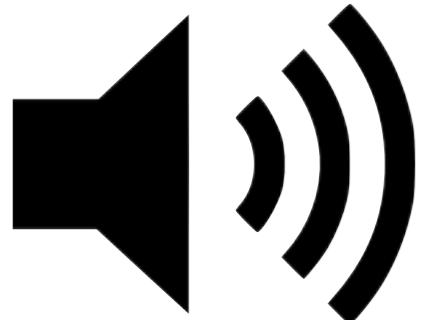
You can read or write a wide range of values

Read 0 - 1023

Written 0 - 255

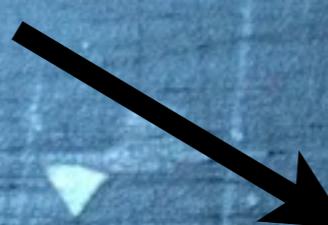
These can be only be configured as input.

the *really* cool
thing about this:
we can make the
arduino **sense** like
we can.

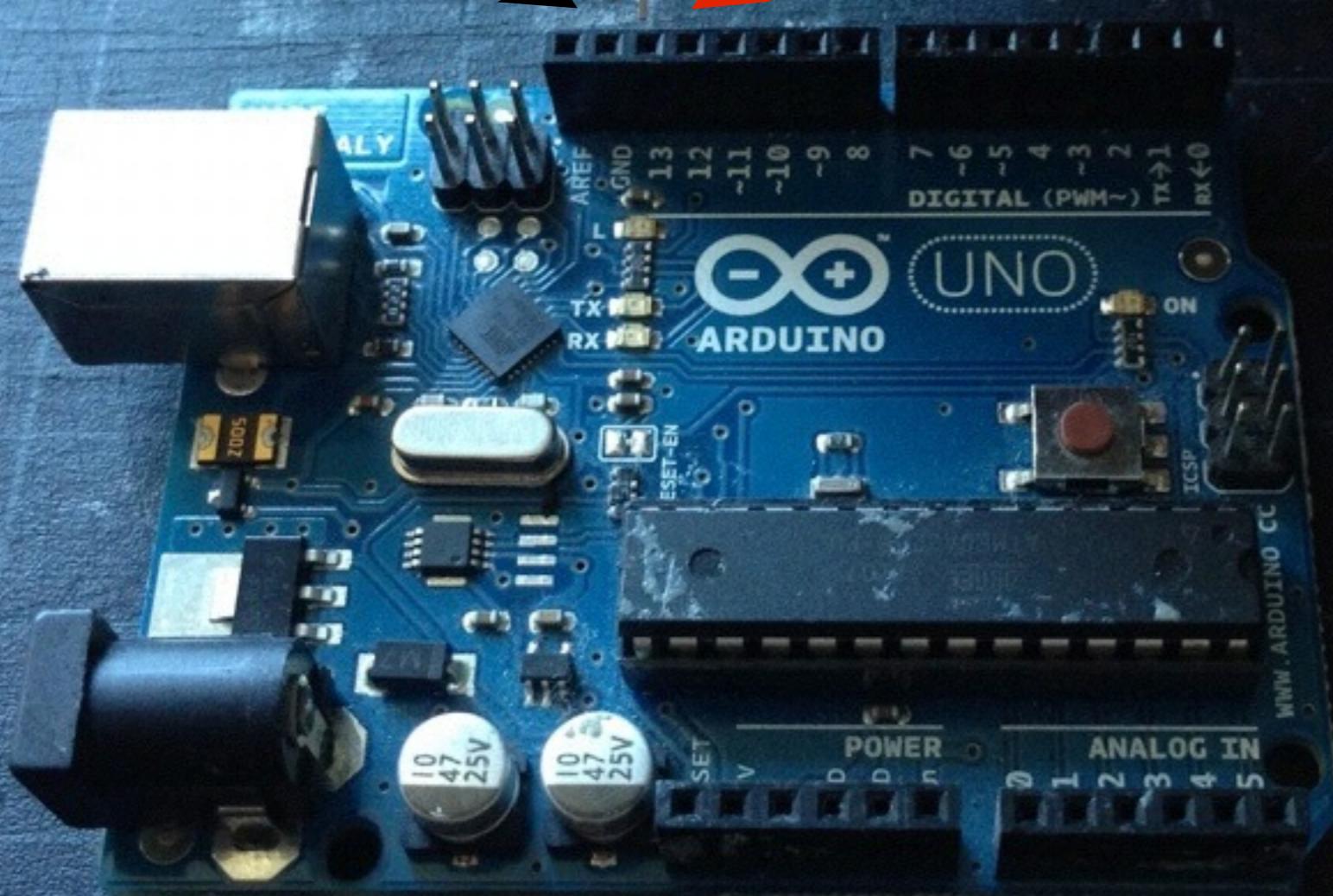


let's build!

- /GRD



+ /PIN 13



YOU CAN ONLY DO THIS ON PIN 13!

Arduino File Edit Sketch Tools Help

sketch_oct07a | Arduino 1.0.1

sketch_oct07a §

```
/*
My first sketch!
*/
```

ATtiny44 (internal 1 MHz clock) on /dev/tty.usbmodemfd131



sketch_oct07

```
/*
My first sketch
*/
```

New ⌘N
Open... ⌘O
Sketchbook

Examples ►

Close ⌘W
Save ⌘S
Save As... ⌘⌘S
Upload ⌘U
Upload Using Programmer ⌘⌘U

Page Setup ⌘⌘P
Print ⌘P

⌘N
⌘O

►

01.Basics ►

02.Digital
03.Analog
04.Communication
05.Control
06.Sensors
07.Display
08.Strings
09.USB(Leonardo)
ArduinoISP

IRremote
LPD8806
PCM

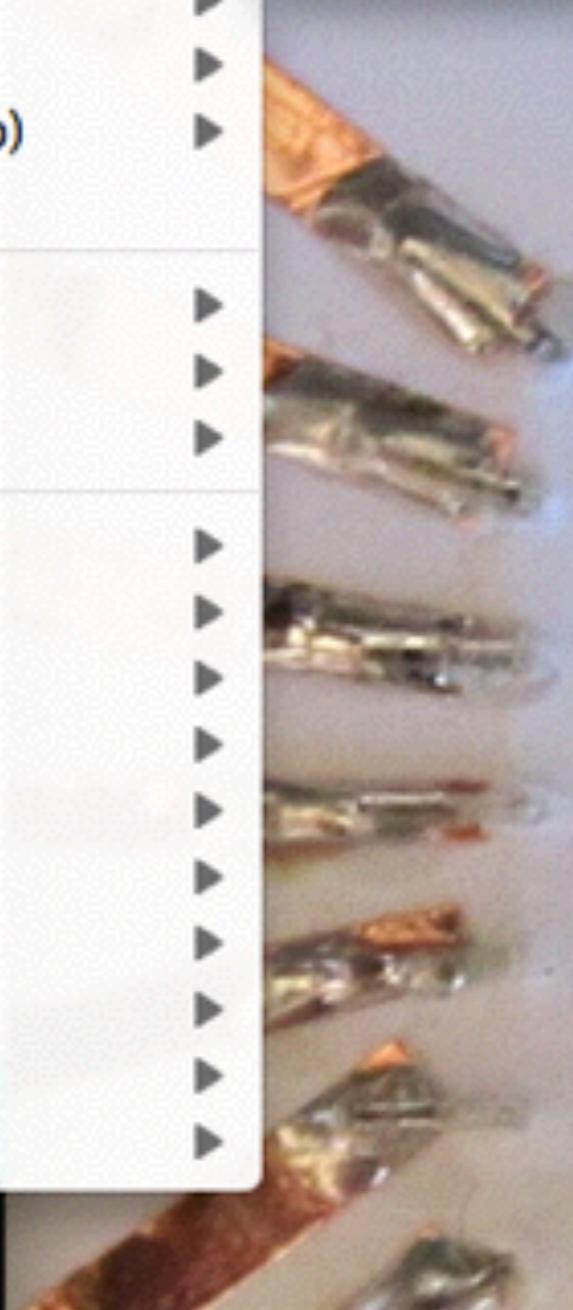
EEPROM
Ethernet
Firmata
LiquidCrystal
SD
Servo
SoftwareSerial
SPI
Stepper
Wire

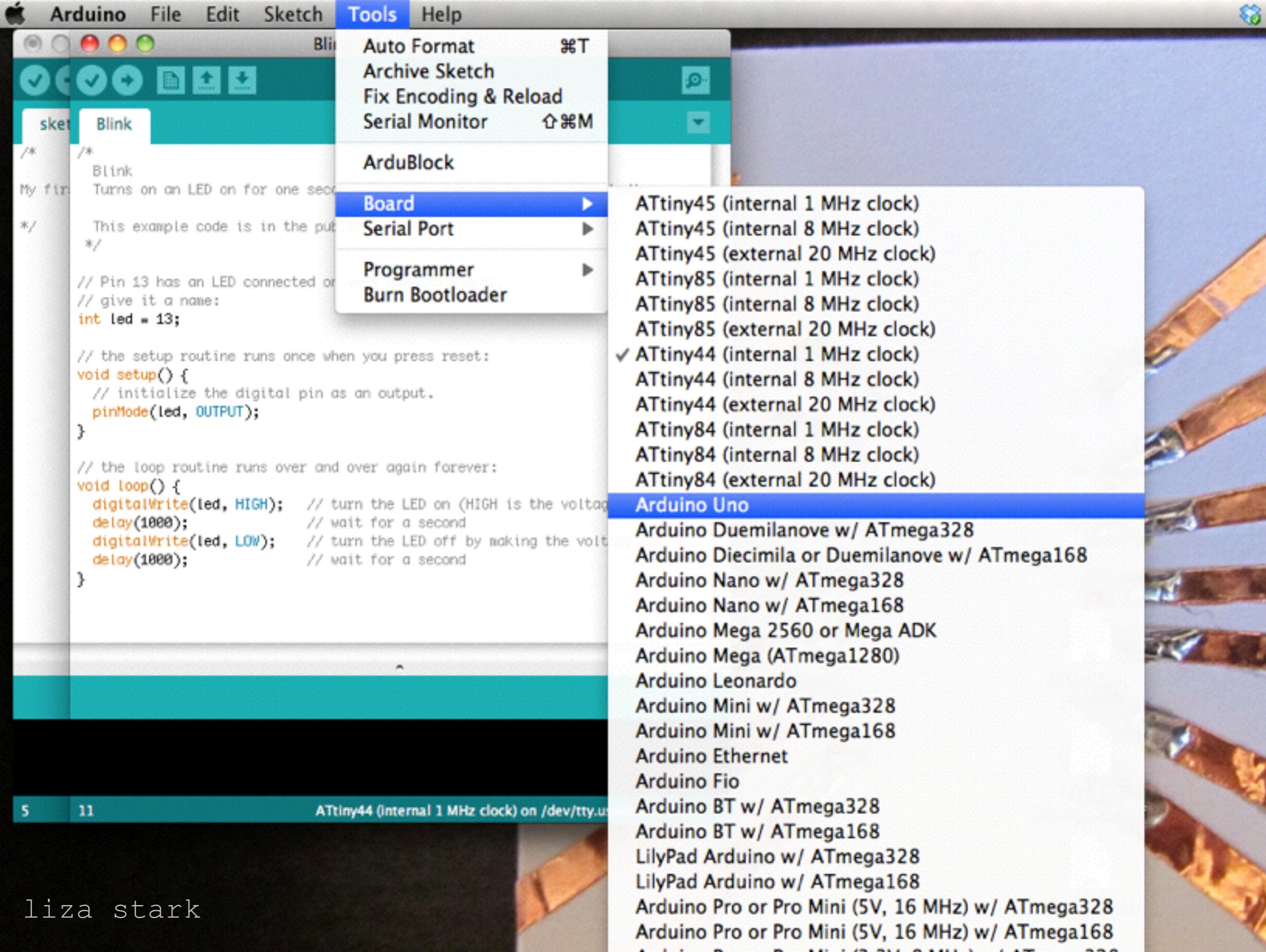
AnalogReadSerial

BareMinimum

Blink

DigitalReadSerial
Fade
ReadAnalogVoltage





Auto Format ⌘T
Archive Sketch
Fix Encoding & Reload
Serial Monitor ⌘M

ArduBlock

Board ▶
Serial Port ▶
Programmer ▶
Burn Bootloader ▶

- ATtiny45 (internal 1 MHz clock)
- ATtiny45 (internal 8 MHz clock)
- ATtiny45 (external 20 MHz clock)
- ATtiny85 (internal 1 MHz clock)
- ATtiny85 (internal 8 MHz clock)
- ATtiny85 (external 20 MHz clock)
- ✓ ATTiny44 (internal 1 MHz clock)
- ATTiny44 (internal 8 MHz clock)
- ATTiny44 (external 20 MHz clock)
- ATTiny84 (internal 1 MHz clock)
- ATTiny84 (internal 8 MHz clock)
- ATTiny84 (external 20 MHz clock)
- Arduino Uno
- Arduino Duemilanove w/ ATmega328
- Arduino Diecimila or Duemilanove w/ ATmega168
- Arduino Nano w/ ATmega328
- Arduino Nano w/ ATmega168
- Arduino Mega 2560 or Mega ADK
- Arduino Mega (ATmega1280)
- Arduino Leonardo
- Arduino Mini w/ ATmega328
- Arduino Mini w/ ATmega168
- Arduino Ethernet
- Arduino Fio
- Arduino BT w/ ATmega328
- Arduino BT w/ ATmega168
- LilyPad Arduino w/ ATmega328
- LilyPad Arduino w/ ATmega168
- Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
- Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
- Arduino Pro Mini (3.3V, 8 MHz) w/ ATmega328

Arduino File Edit Sketch

Tools Help

Auto Format ⌘T
Archive Sketch
Fix Encoding & Reload
Serial Monitor ⌘M

ArduBlock

Board ▶

Serial Port ▶

Programmer ▶

Burn Bootloader

/dev/tty.usbmodemfa141
/dev/cu.usbmodemfa141
/dev/tty.Bluetooth-PDA-Sync
/dev/cu.Bluetooth-PDA-Sync
/dev/tty.Bluetooth-Modem
/dev/cu.Bluetooth-Modem

```
sketchbook/Blink.ino
/*
 * Blink
 * Turns on an LED on for one second,
 * then turns it off for one second,
 * repeatedly.
 */
// This example code is in the public domain.

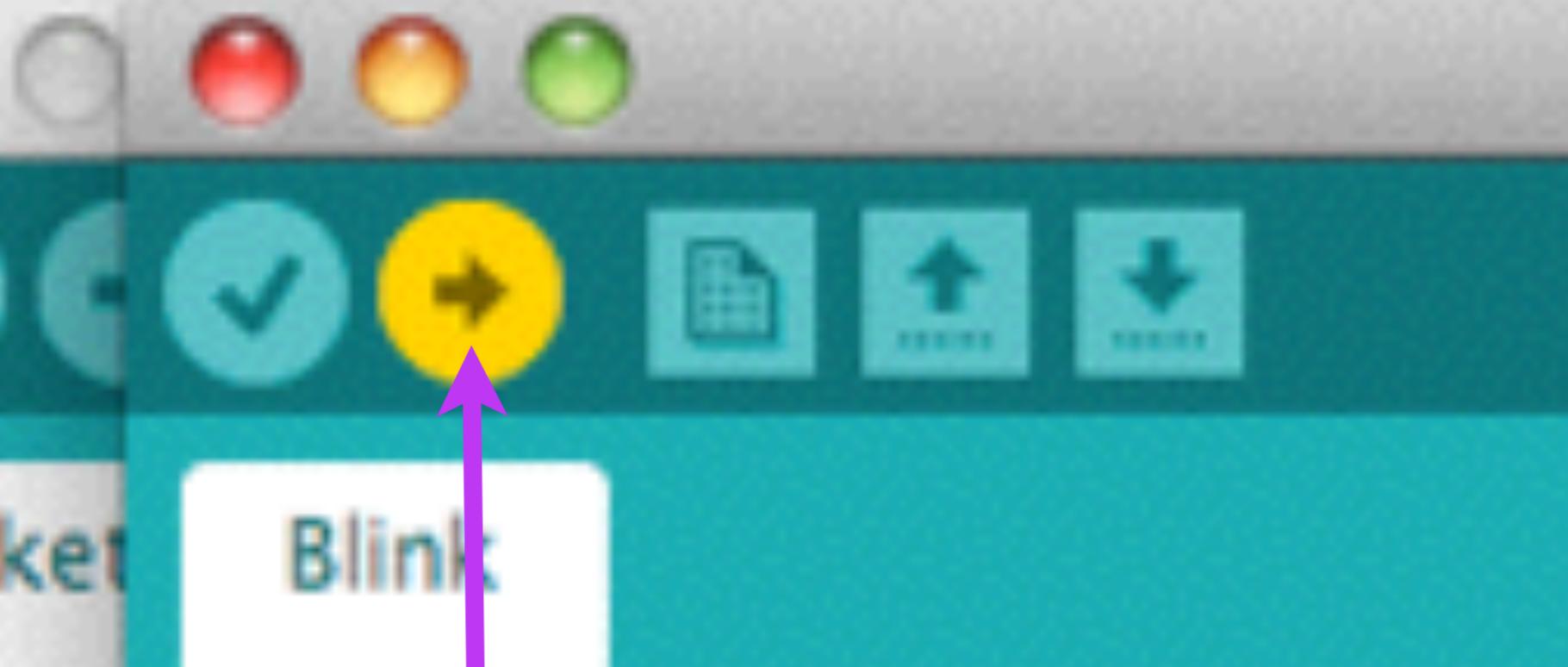
// Pin 13 has an LED connected or
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

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



Arduino File Edit Sketch



```
/*
 *  Sketch: Blink
 *  Turns on an LED on for one second
 *  This example code is in the
 */
// Pin 13 has an LED connected
```

The image shows the Arduino IDE interface with the title bar "Blink | Arduino 1.0.1". The main area displays the "Blink 5" sketch. The code is as follows:

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
    // initialize the digital pin as an output.
    pinMode(led, OUTPUT);
}

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

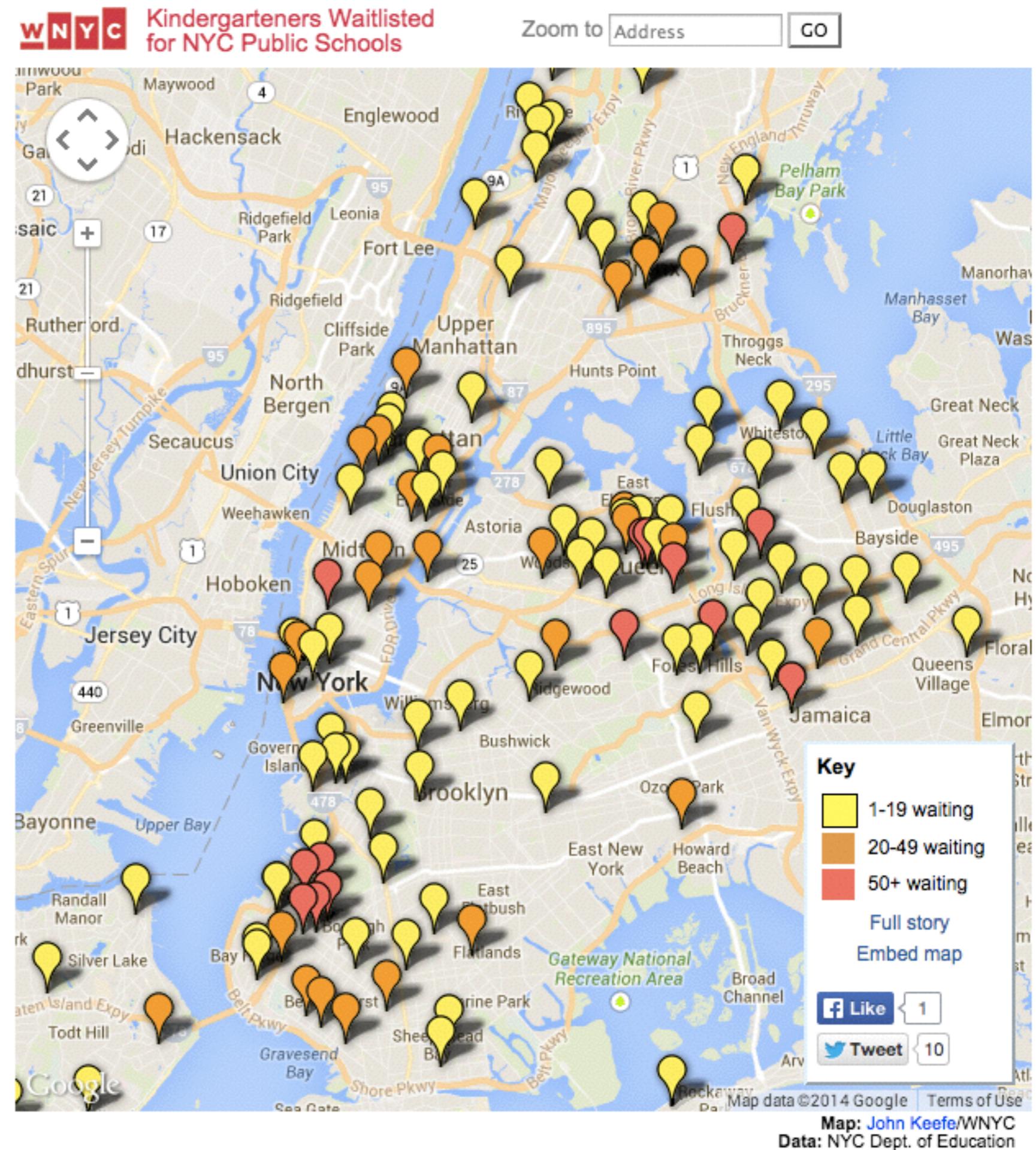
The status bar at the bottom left says "Done uploading." and the bottom right shows the name "liza stark".

Make an account at
cartodb.com
(free)

MAKE MAP BLINK

PINS

3

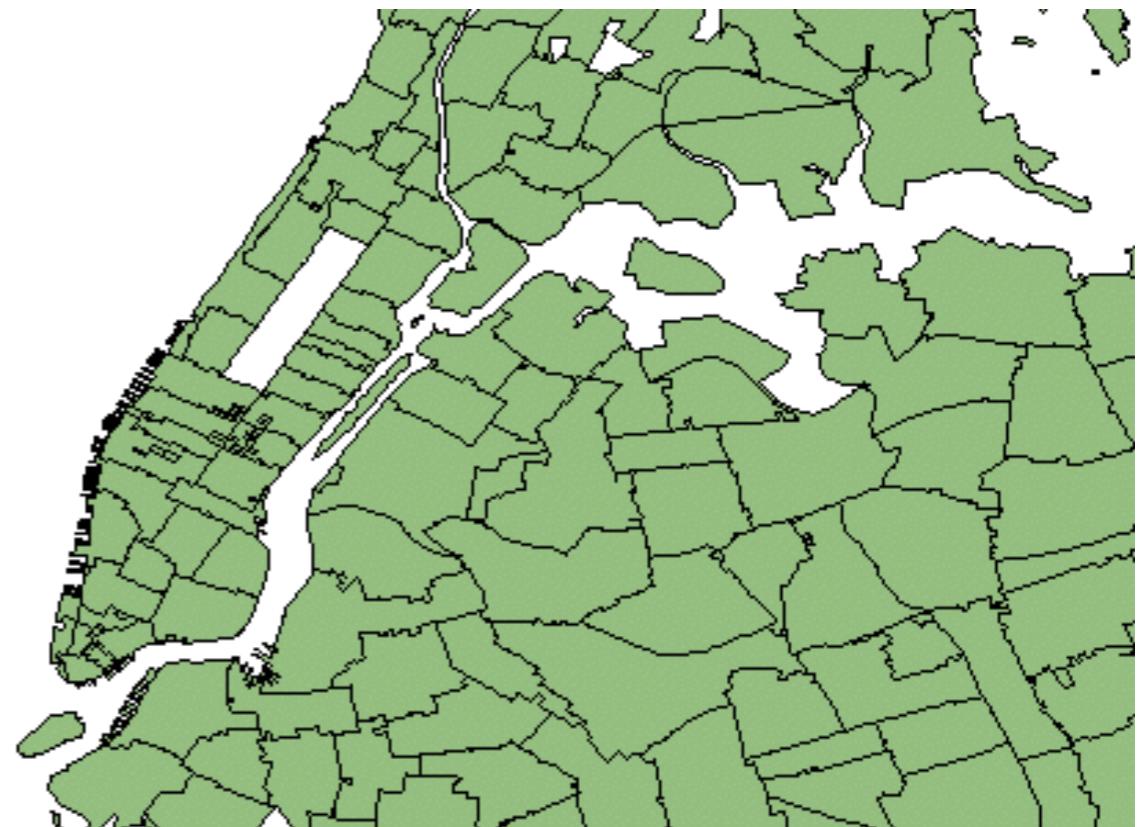


MAKE
MAP
BLINK

DATA

5-Digit Zip	Amt
10013	\$800
10014	\$1,600
10016	\$0
10017	\$3,200

SHAPES

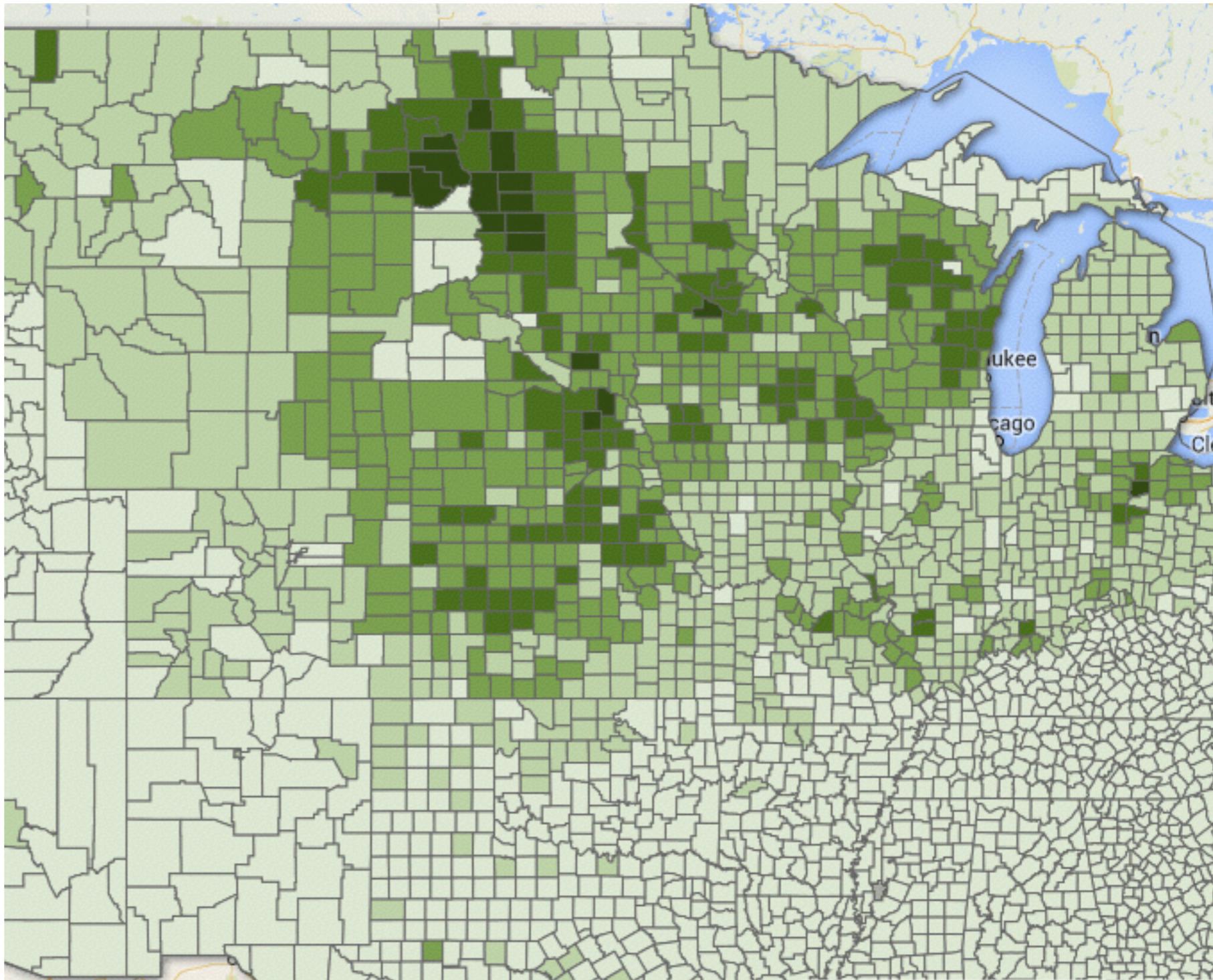


2

**MAKE
MAP
BLINK**

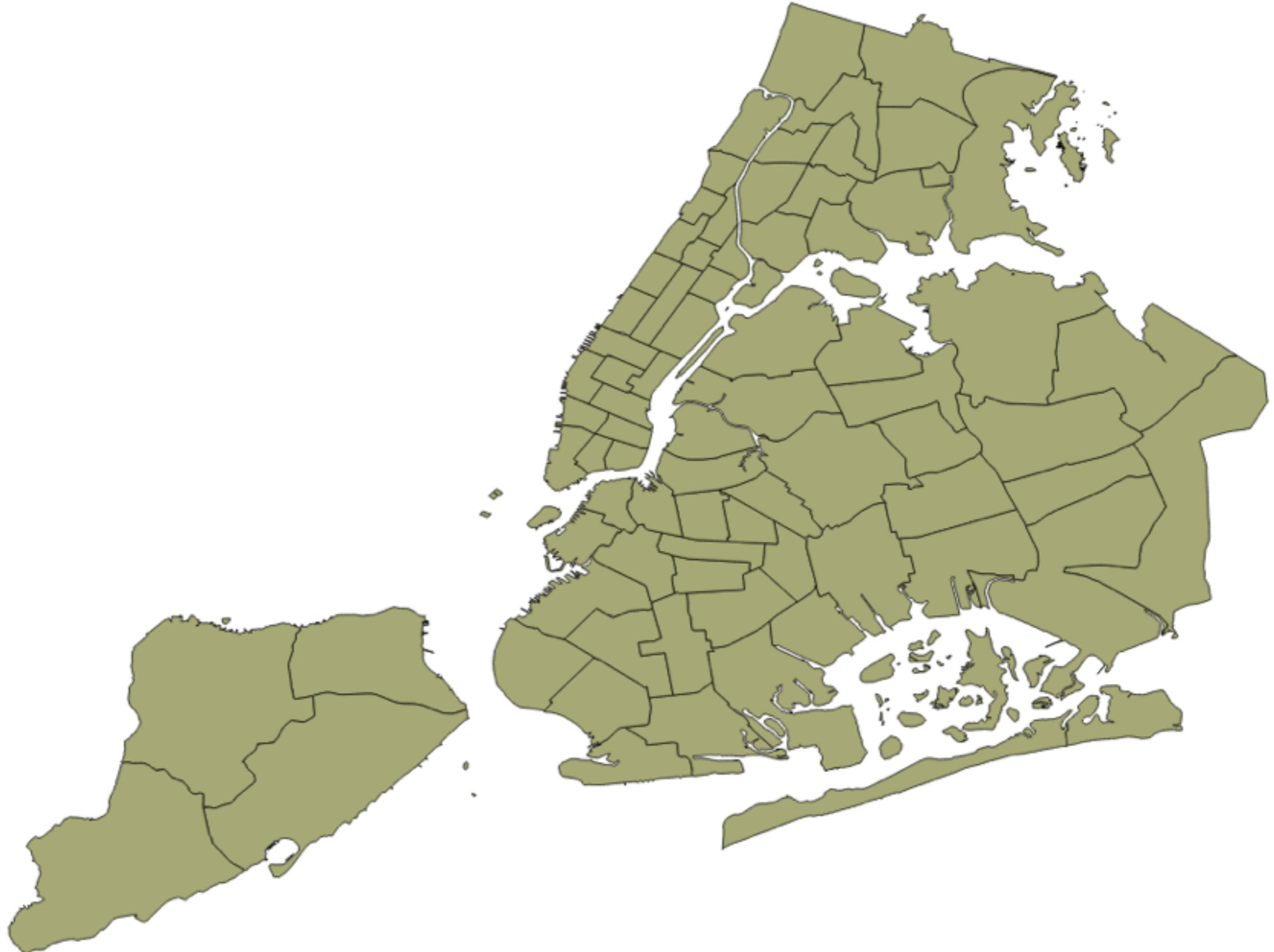
SHAPES

3



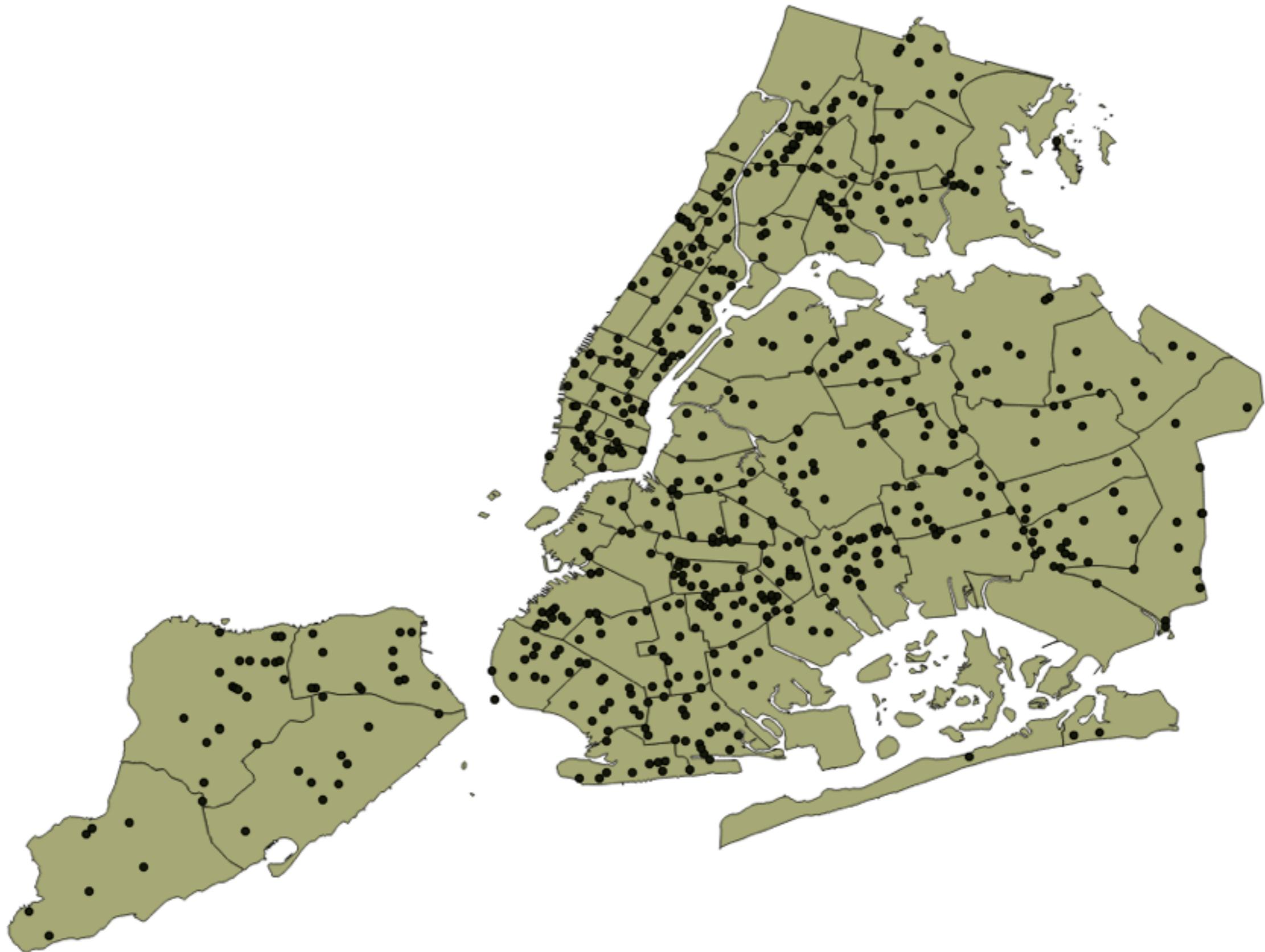
**MAKE
MAP
BLINK**

3



**MAKE
MAP
BLINK**

3



MAKE

John Keefe

MAP

@jkeefe
johnkeefe.net
john@johnkeefe.net

BLINK