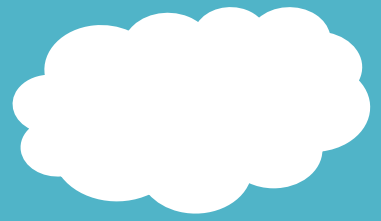
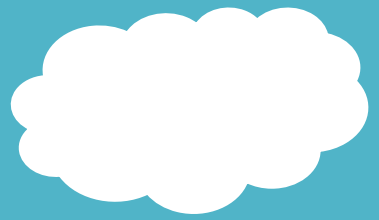


# Internet of THINGS

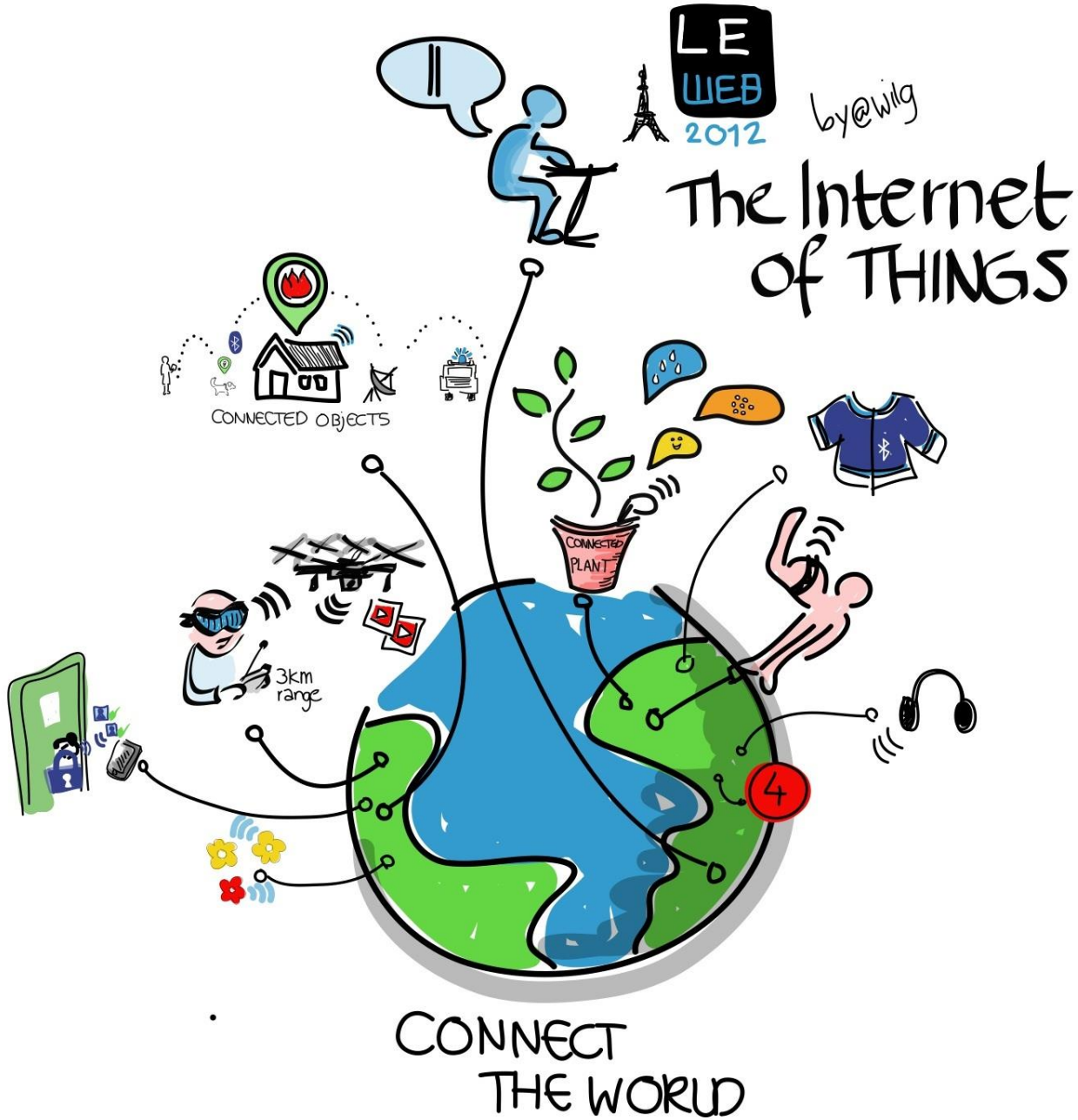


Julio Cesar Faracco

# WHAT IS "INTERNET OF THINGS"?

# WATCH_DOGS

# WHAT IS "INTERNET OF THINGS"?


LE WEB 2012
by@wilg
The Internet of THINGS
CONNECTED OBJECTS
CONNECTED PLANT
3km range
4
CONNECT THE WORLD

Kevin Ashton introduce the term IoT
during his work at MIT Auto-ID Labs in 1999

A global standard system/network
that uses RFID and other sensors

# 26+ BILLIONS
of connected devices until 2020

# AN ICREASE OF 22%
on investiments of IoT in 2016

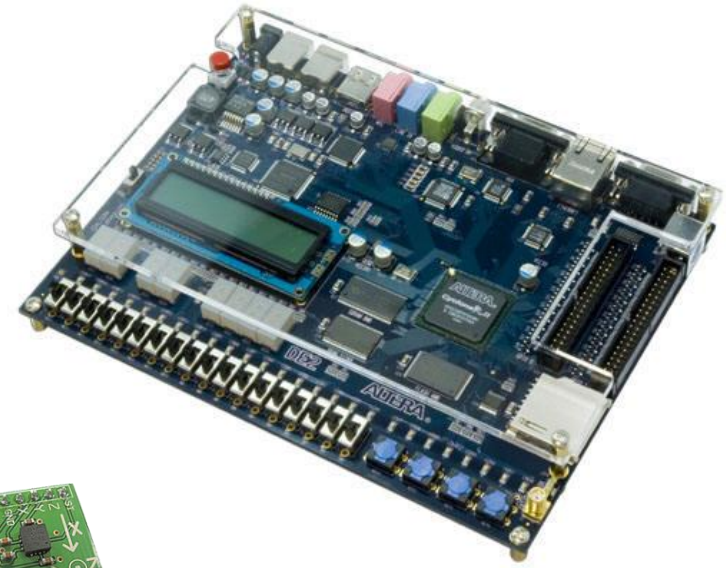
# 15 TRILLIONS
of global GDP until 2030


0101
1010
0010
NEWS

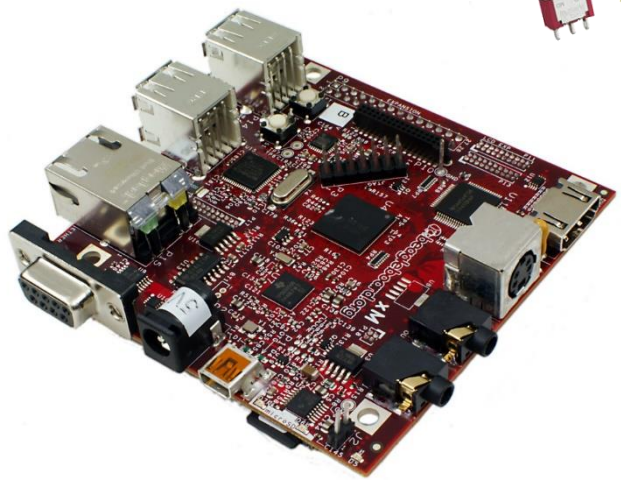
# Internet of Things

# versus

# Embedded Systems



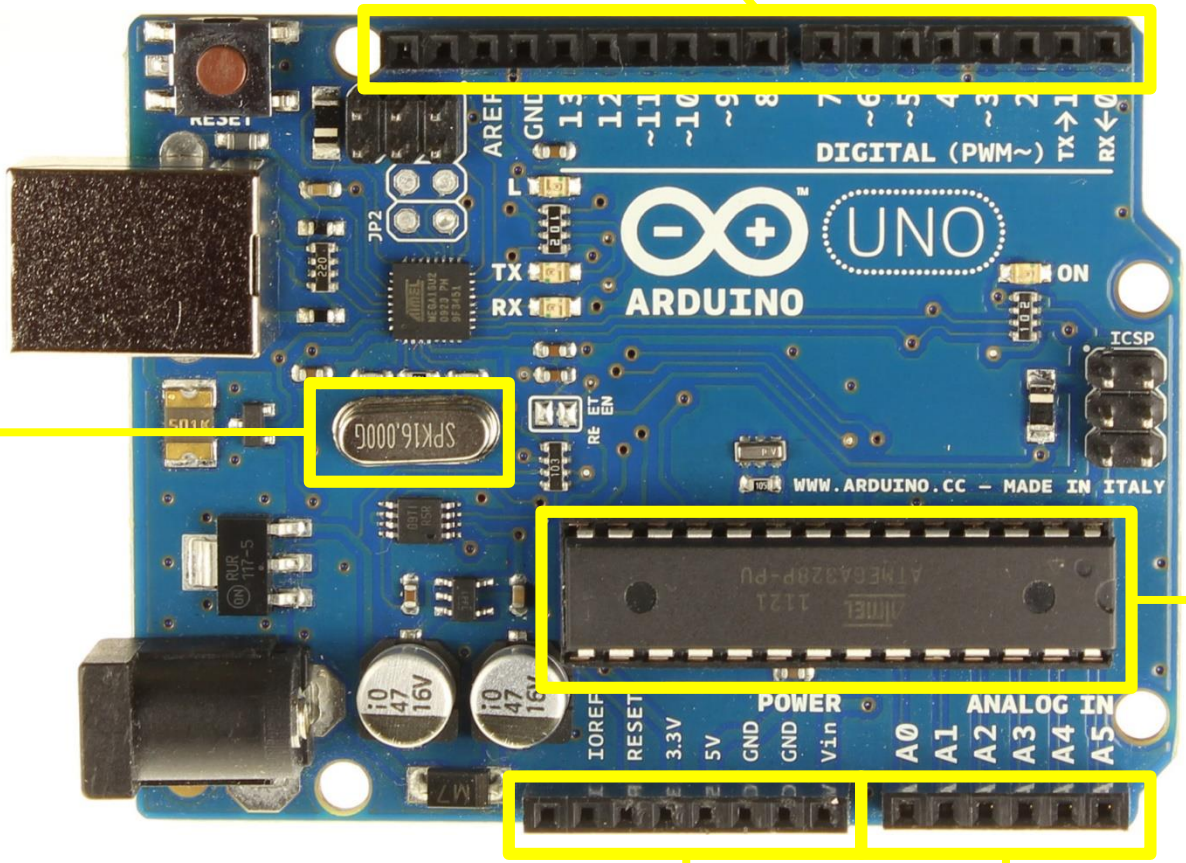

HMC6352
XBee




ARDUINO
YUN

# What is Arduino?

12 digital inputs (6 PWM)




16 MHz

ATmega328P

31.5Kb Flash

2Kb SRAM

1Kb EEPROM

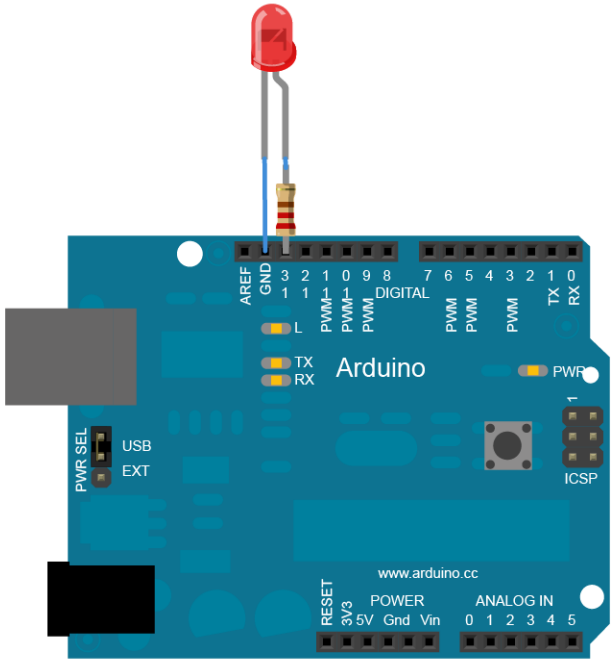
General I/O

6 Analog Inputs

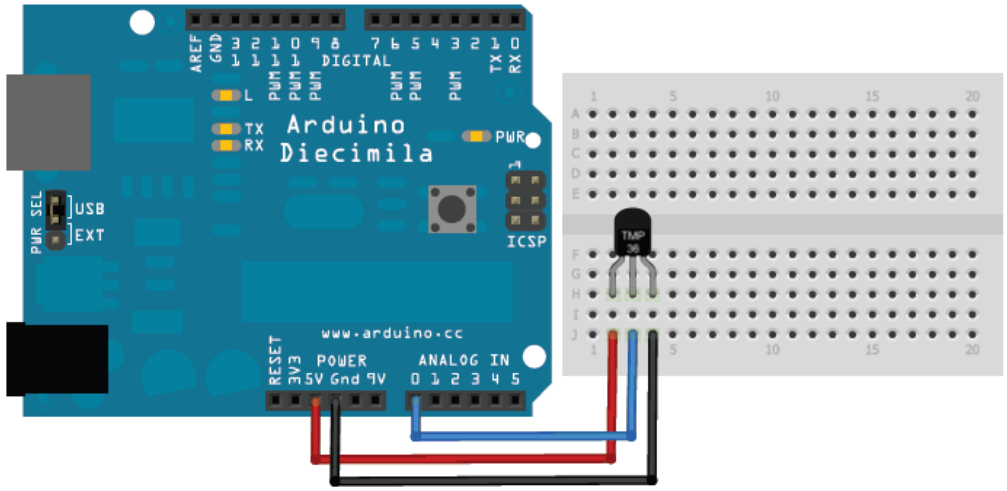
# Why Arduino is so cool?

Arduino is an Open-Source Hardware that anyone can develop electronic/digital projects

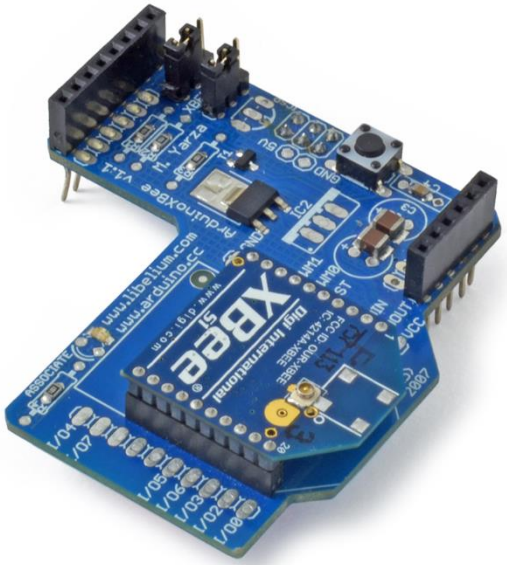
```ino
// the setup function runs once when you press reset or power the board.
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever.
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

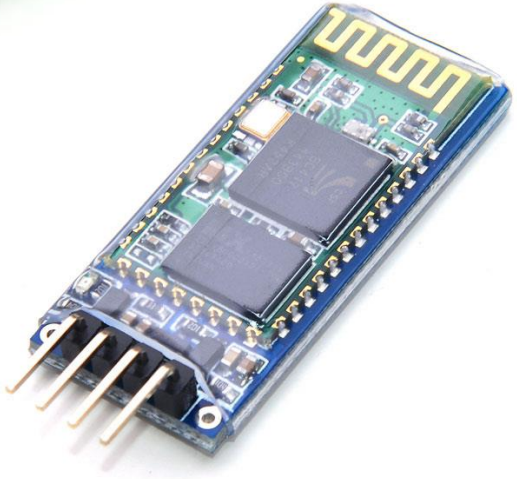


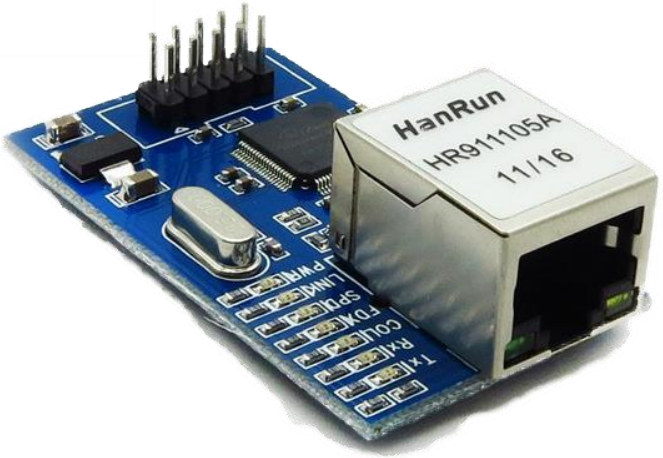
```
//TMP36 Pin Variables
int sensorPin = 0;        // The analog pin the TMP36's Vout (sense) pin is connected to the
                          // resolution is 10 mV / degree centigrade with a
                          // 500 mV offset to allow for negative temperatures

void setup() {
  Serial.begin(9600);     // Start the serial connection with the computer to view the results.
}

void loop() {
  // getting the voltage reading from the temperature sensor
  int reading = analogRead(sensorPin);

  // converting that reading to voltage, for 3.3v arduino use 3.3
  float voltage = reading * 5.0;
  voltage /= 1024.0;

  // now print out the temperature
  // converting from 10 mv per degree wit 500 mV offset
  float temperatureC = (voltage - 0.5) * 100 ;
  // to degrees ((voltage - 500mV) times 100)
  Serial.print(temperatureC);
  Serial.println(" degrees C");

  delay(1000); //waiting a second
}
```



# Arduino and Internet of Things

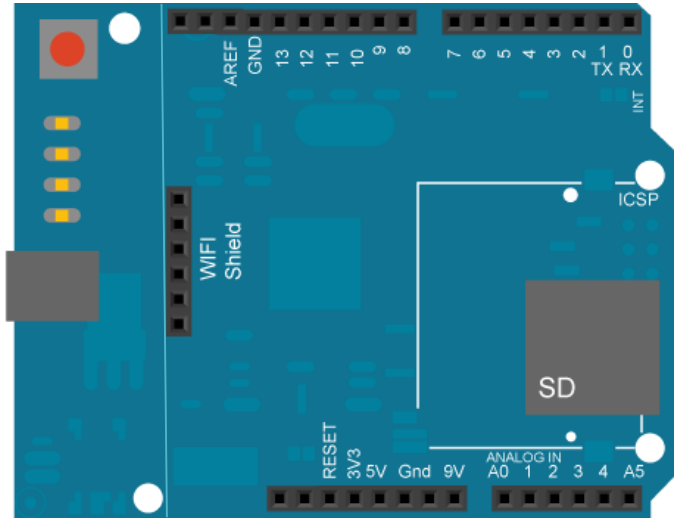


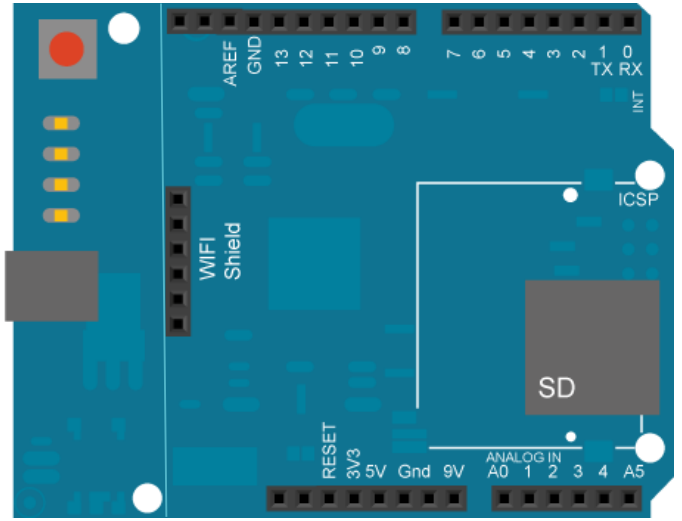


```
#include <SPI.h>
#include <WiFi.h>

void printMacAddress() {
  // the MAC address of your Wifi shield
  byte mac[6];

  // print your MAC address:
  WiFi.macAddress(mac);
  Serial.print("MAC: ");
  Serial.print(mac[5],HEX);
  Serial.print(":");
  Serial.print(mac[4],HEX);
  Serial.print(":");
  Serial.print(mac[3],HEX);
  Serial.print(":");
  Serial.print(mac[2],HEX);
  Serial.print(":");
  Serial.print(mac[1],HEX);
  Serial.print(":");
  Serial.println(mac[0],HEX);
}

void listNetworks() {
  // scan for nearby networks:
  Serial.println("** Scan Networks **");
  byte numSsid = WiFi.scanNetworks();

  // print the list of networks seen:
  Serial.print("number of available networks:");
  Serial.println(numSsid);

  // print the network number and name for each network found:
  for (int thisNet = 0; thisNet<numSsid; thisNet++) {
    Serial.print(thisNet);
    Serial.print(") ");
    Serial.print(WiFi.SSID(thisNet));
    Serial.print("\tSignal: ");
    Serial.print(WiFi.RSSI(thisNet));
    Serial.print(" dBm");
    Serial.print("\tEncryption: ");
    Serial.println(WiFi.encryptionType(thisNet));
  }
}
```

```
void setup() {
  // initialize serial and wait for the port to open:
  Serial.begin(9600);
  while(!Serial);

  // attempt to connect using WEP encryption:
  Serial.println("Initializing Wifi...");
  printMacAddress();

  // scan for existing networks:
  Serial.println("Scanning available networks...");
  listNetworks();
}

void loop() {
  delay(10000);
  // scan for existing networks:
  Serial.println("Scanning available networks...");
  listNetworks();
}
```



```
#include <SPI.h>
#include <WiFi.h>

char ssid[] = "yourNetwork";      //  your network SSID (name)
char pass[] = "secretPassword";   // your network password
                                  // (use for WPA, or use as key for WEP)
int keyIndex = 0; // your network key Index number (needed only for WEP)

int status = WL_IDLE_STATUS;
char server[] = "www.google.com"; // name address for Google (using DNS)

// Initialize the Ethernet client library
WiFiClient client;
```



```
void setup() {
  Serial.begin(9600);
  while (!Serial);
  }

  // check for the presence of the shield:
  if (WiFi.status() == WL_NO_SHIELD) {
    Serial.println("WiFi shield not present");
    while(true);
  }

  // attempt to connect to Wifi network:
  while (status != WL_CONNECTED) {
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(ssid);
    // Connect to WPA/WPA2 network.
    // Change this line if using open or WEP.
    // status = WiFi.begin(ssid, keyIndex, pass);
    status = WiFi.begin(ssid, pass);

    // wait 10 seconds for connection:
    delay(10000);
  }
  Serial.println("Connected to wifi");
  printWifiStatus();

  Serial.println("\nConnecting to server...");
  // if you get a connection, report back via serial:
  if (client.connect(server, 80)) {
    Serial.println("connected to server");
    // Make a HTTP request:
    client.println("GET /search?q=arduino HTTP/1.1");
    client.println("Host: www.google.com");
    client.println("Connection: close");
    client.println();
  }
}
```

```
void loop() {
  // if there are incoming bytes available
  // from the server, read them and print them:
  while (client.available()) {
    char c = client.read();
    Serial.write(c);
  }

  // if the server's disconnected, stop the client:
  if (!client.connected()) {
    Serial.println("\ndisconnecting from server.");
    client.stop();

    // do nothing forevermore:
    while(true);
  }
}


void printWifiStatus() {
  // print the SSID of the network you're attached to:
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());

  // print your WiFi shield's IP address:
  IPAddress ip = WiFi.localIP();
  Serial.print("IP Address: ");
  Serial.println(ip);

  // print the received signal strength:
  long rssi = WiFi.RSSI();
  Serial.print("signal strength (RSSI):");
  Serial.print(rssi);
  Serial.println(" dBm");
}
```

# The Code Without Verbose Mode (Serial) And Comments...

```
void setup() {
  if (WiFi.status() == WL_NO_SHIELD) {
    while(true);
  }

  while (status != WL_CONNECTED) {
    // status = WiFi.begin(ssid, keyIndex, key);
    status = WiFi.begin(ssid, pass);
    delay(10000);
  }

  if (client.connect(server, 80)) {
    client.println("GET /search?q=arduino HTTP/1.1");
    client.println("Host: www.google.com");
    client.println("Connection: close");
    client.println();
  }
}

void loop() {
  while (client.available()) {
    char c = client.read();
  }

  if (!client.connected()) {
    client.stop();

    while(true);
  }
}
```

JohnnyFivehidapiBlocklyDuinonoduinoi
noradardublockdinoduinohwioJArduino
PlatformIOTinyGPSSoulissPJONOpenHA
BOpenIoTOpenRemoteTheThingSystem

# Links

https://123d.circuits.io/

https://easyeda.com/

http://www.falstad.com/circuit/

http://www.docircuits.com/circuit-editor

https://www.arduino.cc/en/Tutorial/HomePage


T.Hanks