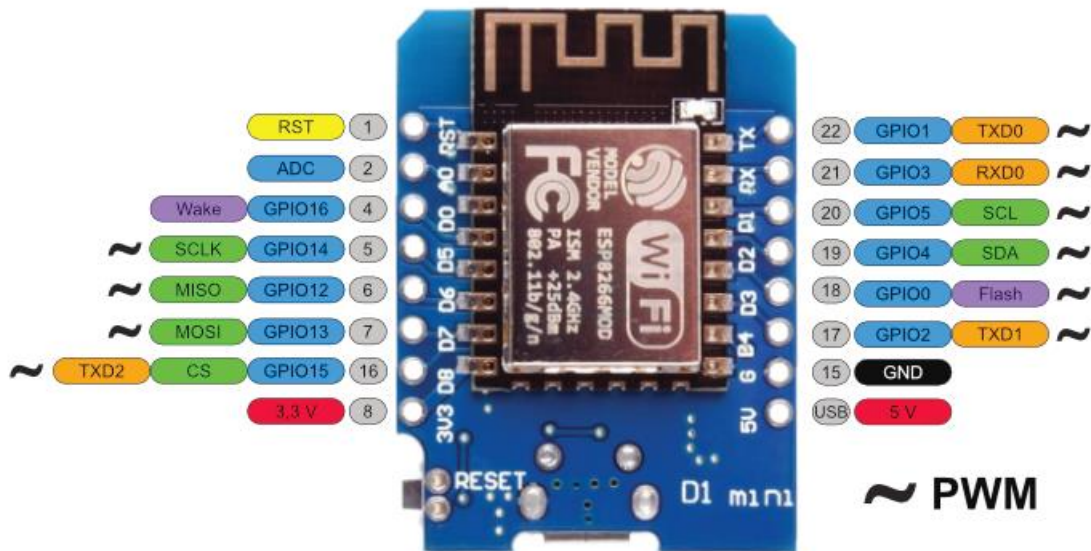


WEMOS D1 MINI E DISPLAY OLED

La scheda Wemos D1mini è una piccola ma potente scheda con microcontrollore a 32 bit ESP8266 con 4MB di memoria Flash ed 80 MHz di clock..

Questo micro è dotato di supporto wifi per la connessione ad una qualsiasi rete.

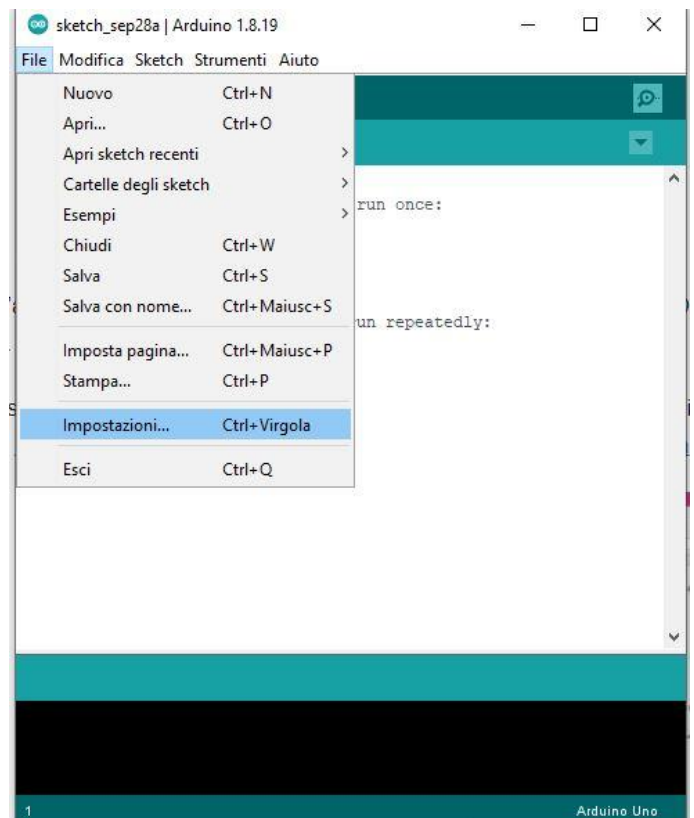
La



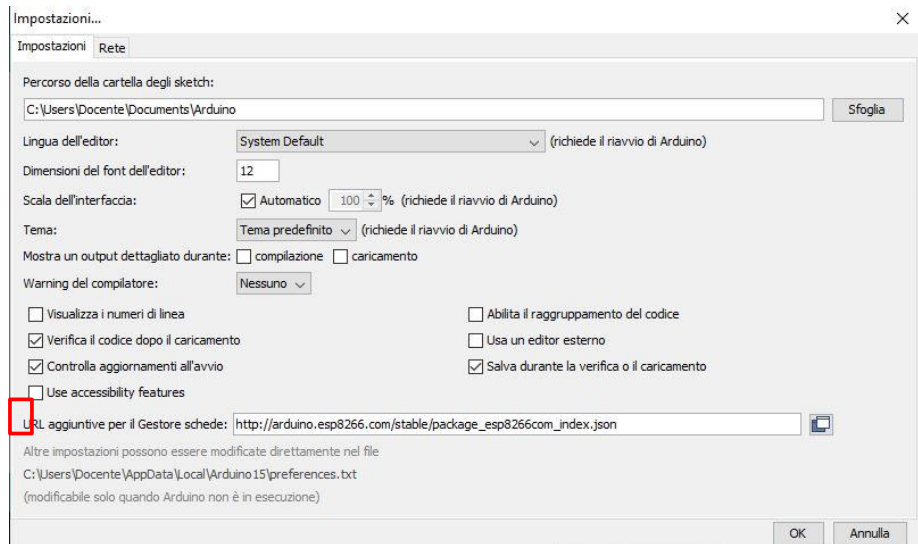
scheda è programmabile con la piattaforma ARDUINO.

Per la sua programmazione, occorre innanzitutto installare la scheda sulla piattaforma, per farlo seguire i seguenti passi:

1) Sul menù file andare su IMPOSTAZIONI

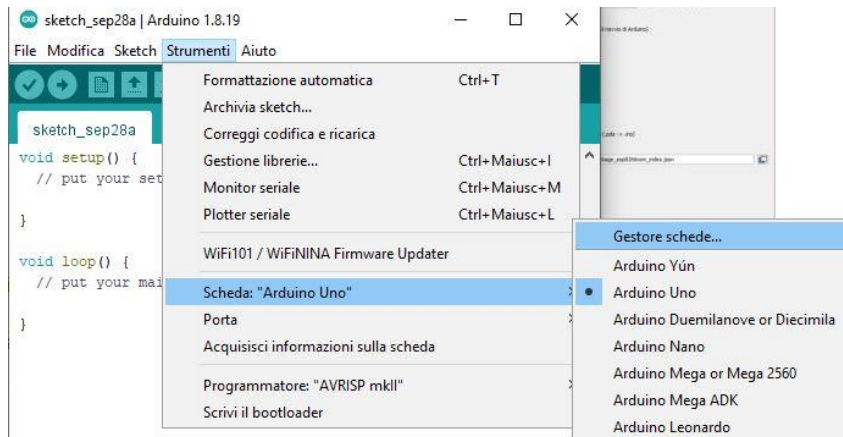


2) Sulla sezione **URL aggiuntive** scrivere il link evidenziato in figura:

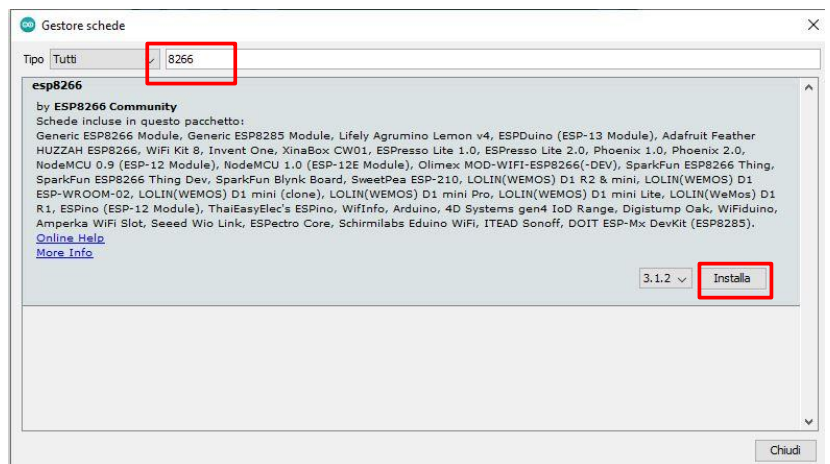


http://arduino.esp8266.com/stable/package_esp8266com_index.json

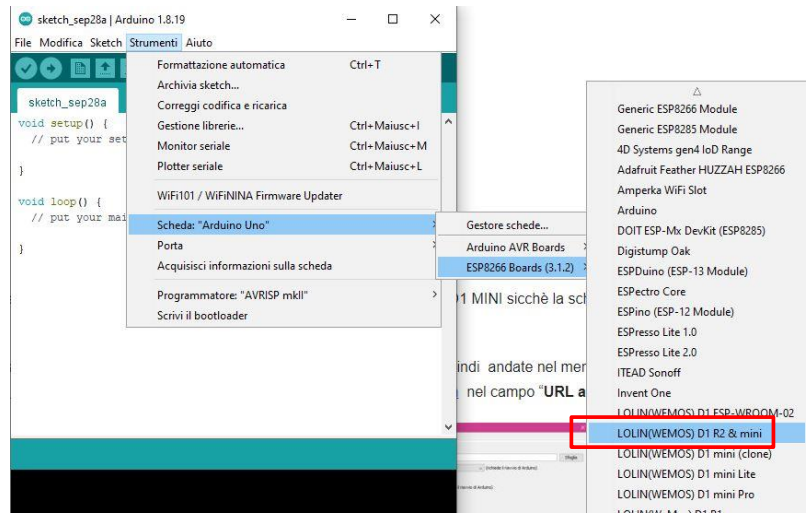
3) A questo punto sulla sezione **STRUMENTI**, selezionare **GESTIONE SCHEDE**



4) Inserire sul filtro di ricerca **8266** o anche **ESP8266** e verrà visualizzata la libreria **ESP8266**, fare click su **INSTALLA**

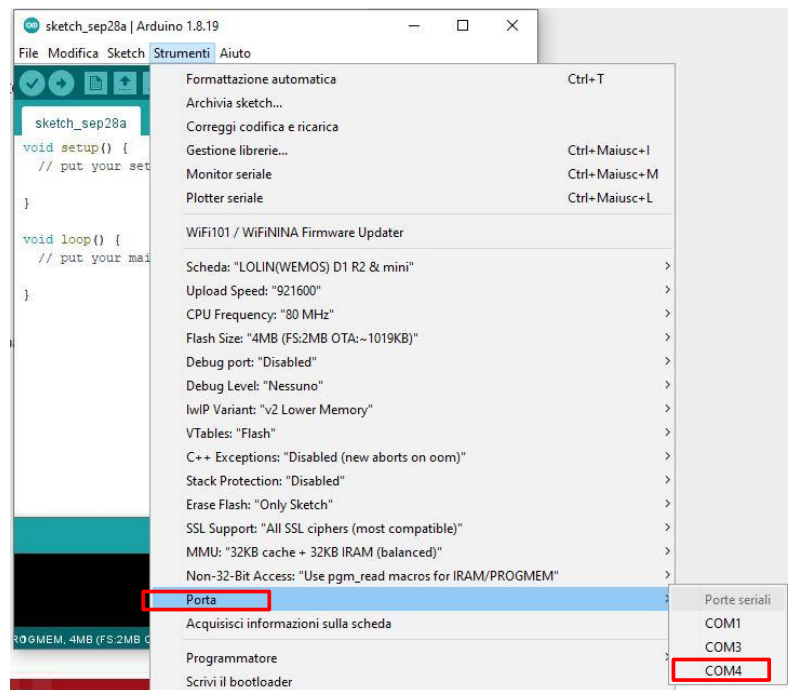


5) Una volta installata la scheda, selezionare su **STRUMENTI** il tipo di scheda **LOLIN WEMOS D1 R2 & mini**



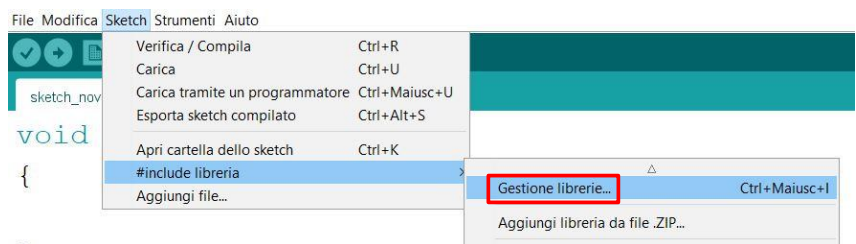
6) Collegare la scheda ed impostare la porta **COM** che appare sul menu **STRUMENTI-PORTA**

La porta COM da selezionare dipende dal PC e può essere differente da quella indicata in figura.



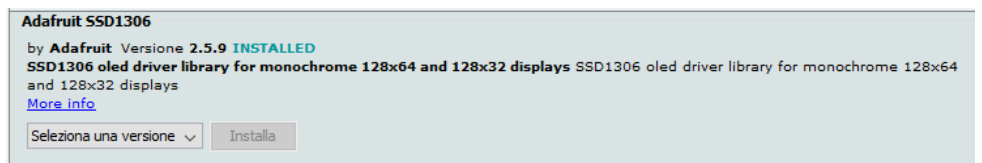
7) Installare la libreria **Adafruit SSD1306**.

Andare sul menù **SKETCH** poi **Gestione librerie**.



Inserire nel filtro la parola **SSD1306** per cercare la libreria.

Installare la libreria, al termine chiudere la finestra.

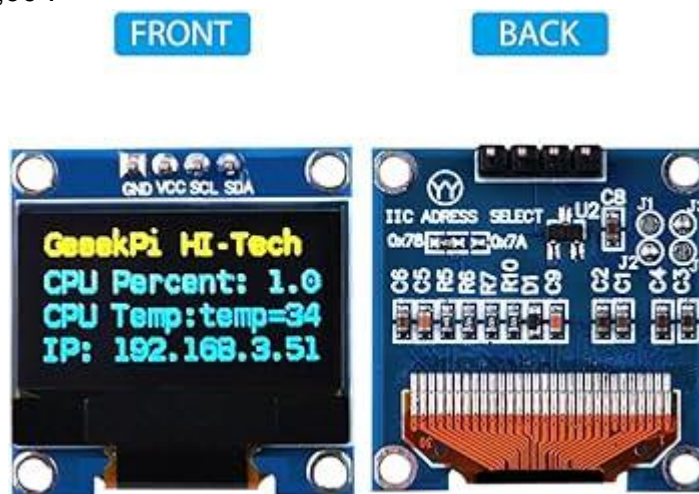


DISPLAY OLED (Organic LED)

I display OLED, sono realizzati con materiale organico (strutture prevalentemente composte da carbonio). Lo strato di materiale è molto sottile e viene interposto tra due piastre conduttrici, vengono chiamate “elettrodi”, di cui almeno una trasparente. Il materiale organico utilizzato è elettroluminescente, cioè in grado di emettere luce quanto percorso da corrente.

I display OLED sono in grado pertanto di emettere luce propria per le caratteristiche di elettroluminescenza del materiale organico che opportunamente drogato, consente l'emissione di differenti colorazioni.

Nei più commerciali display OLED, sono presenti due tipologie di controller (circuiti integrati che si interpongono tra il display e l'eventuale microcontrollore) uno è il modello SSD1306 l'altro SH1106. Con questi controller troviamo abbinati display di dimensioni modeste, uno dei più comuni ha una dimensione di 0,96”.



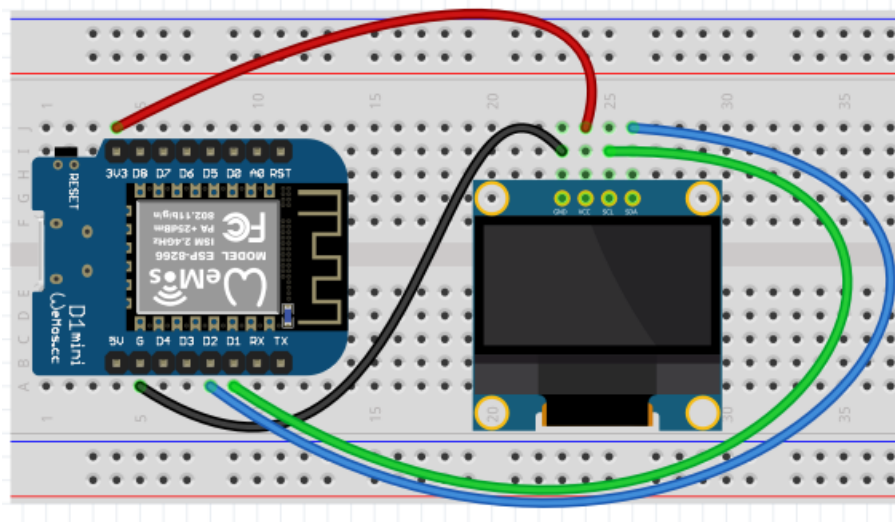
Questi display vengono gestiti tramite un collegamento con bus a 2 fili I2C.

In un bus I2C i dati viaggiano sulla linea SDA mentre sulla linea SCL viaggia un segnale di clock per sincronizzare la trasmissione.

Oltre alle due linee il display ha bisogno di un'alimentazione a 3,3Volt.

Sulla scheda WEMOS D1Mini, il bus I2C fa capo ai pin D1 (SCL) e D2 (SDA).

Pertanto il collegamento è il seguente:



DISPLAY OLED (Organic LED)

Il seguente codice integra le funzioni principali per gestire il display.

Nella prima parte vanno inserite le librerie necessarie,

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128 // OLED larghezza, in pixels
#define SCREEN_HEIGHT 64 // OLED altezza, in pixels

#define OLED_RESET      -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
```

Nel setup viene inizializzato il monitor seriale per il debug, ed inizializzato il display con il suo indirizzo 0x3C.

```
void setup() {
  Serial.begin(9600);

  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println(F("SSD1306 allocation failed"));
    for(;;);
  }
  display.clearDisplay();
}
```

Di seguito le principali funzioni per effettuare semplici disegni.

```
//*****************************************************************************
//DISEGNA SINGOLI PIXEL
//color=0 WHITE      color=1 INVERSE
void disegna_pixel(int coordX, int coordY, int colore){
  display.clearDisplay();
  if(colore==0) display.drawPixel(coordX, coordY, WHITE);
  if(colore==1) display.drawPixel(coordX, coordY, INVERSE);
  display.display();
}

//*****************************************************************************
//DISEGNA UNA LINEA
void disegna_linea(int coordX1, int coordY1, int coordX2, int coordY2, int colore){
  display.clearDisplay();
  if(colore==0) display.drawLine(coordX1, coordY1, coordX2, coordY2, WHITE);
  if(colore==1) display.drawLine(coordX1, coordY1, coordX2, coordY2, INVERSE);
  display.display();
}
```

```
//*****  
//DISEGNA UN RETTANGOLO  
void disegna_rettangolo(int coordX, int coordY, int larghezza, int altezza, int colore){  
    display.clearDisplay();  
    if(colore==0) display.drawRect(coordX, coordY, larghezza, altezza, WHITE);  
    if(colore==1) display.drawRect(coordX, coordY, larghezza, altezza, INVERSE);  
    display.display();  
}//disegna_rettangolo()  
  
//*****  
//DISEGNA UN CERCHIO  
void disegna_cerchio(int centroX, int centroY, int diametro, int colore){  
    display.clearDisplay();  
    if(colore==0) display.drawCircle(centroX, centroY, diametro, WHITE);  
    if(colore==1) display.drawCircle(centroX, centroY, diametro, INVERSE);  
    display.display();  
}//disegna_cerchio()  
  
//*****  
//DISEGNA UN RETTANGOLO ARROTONDATO  
void disegna_rettangolo_arrotondato(int coordX, int coordY, int larghezza, int altezza, int raggio, int colore){  
    display.clearDisplay();  
    if (colore==0) display.drawRoundRect(coordX, coordY, larghezza, altezza, raggio, WHITE);  
    if (colore==1) display.drawRoundRect(coordX, coordY, larghezza, altezza, raggio, INVERSE);  
    display.display();  
}//disegna_rettangolo_arrotondato()
```

Nel ciclo loop eseguo scritte e disegni vari.

```
void loop() {  
  
    int larghezza, altezza;  
    int valore;  
  
    //LEGGO LARGHEZZA ED ALTEZZA DEL DISPLAY  
    larghezza=display.width();  
    altezza=display.height();  
  
    //visualizzo un valore in un rettangolo  
    valore=100;  
    display.clearDisplay();  
    disegna_rettangolo_arrotondato(10,10,100,20,3,0);  
    display.setTextSize(2);  
    display.setTextColor(WHITE);  
    display.setCursor(12,12);  
    display.println(valore);  
    display.display();  
    delay(2000);  
  
    //disegno un singolo pixel  
    disegna_pixel(10,20,0);  
    delay(200);  
    disegna_pixel(20,30,1);  
    delay(200);  
  
    //disegno una linea  
    disegna_linea(0,0, larghezza, altezza, 0);  
    delay(200);  
    disegna_linea(0,0, larghezza, altezza, 1);  
    delay(200);  
}
```

```
//disegno un rettangolo
disegna_rettangolo(0,0,larghezza,altezza,0);
delay(200);
disegna_rettangolo(0,0,larghezza,altezza,1);
delay(200);
//disegno un cerchio
disegna_cerchio(larghezza/2,altezza/2,30,0);
delay(200);
disegna_cerchio(larghezza/2,altezza/2,30,1);
delay(200);
//disegno un rettangolo arrotondato
disegna_rettangolo_arrotondato(0,0,larghezza-2,altezza-2,altezza/4,0);
delay(200);
disegna_rettangolo_arrotondato(0,0,larghezza-2,altezza-2,altezza/4,1);
delay(200);

//scroll nelle due direzioni
display.startscrollright(0x00, 0x0F);
delay(2000);
display.stopscroll();
delay(1000);
display.startscrollleft(0x00, 0x0F);
delay(2000);
display.stopscroll();
delay(1000);
}
```

Di seguito il programma.

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define SCREEN_WIDTH 128 // OLED larghezza, in pixels
#define SCREEN_HEIGHT 64 // OLED altezza, in pixels
#define OLED_RESET -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

void setup() {
  Serial.begin(9600);
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println(F("SSD1306 allocation failed"));
    for(;;);
  }
  display.clearDisplay();
}
```

```
//*****
```

```
//DISEGNA SINGOLI PIXEL
```

```
//color=0 WHITE color=1 INVERSE
```

```
void disegna_pixel(int coordX, int coordY, int colore){  
    display.clearDisplay();  
    if(colore==0) display.drawPixel(coordX, coordY, WHITE);  
    if(colore==1) display.drawPixel(coordX, coordY, INVERSE);  
    display.display();  
}  
//disegna_pixel()
```

```
//*****
```

```
//DISEGNA UNA LINEA
```

```
void disegna_linea(int coordX1, int coordY1, int coordX2, int coordY2, int colore){  
    display.clearDisplay();  
    if(colore==0) display.drawLine(coordX1, coordY1, coordX2, coordY2, WHITE);  
    if(colore==1) display.drawLine(coordX1, coordY1, coordX2, coordY2, INVERSE);  
    display.display();  
}  
//disegna_linea()
```

```
//*****
```

```
//DISEGNA UN RETTANGOLO
```

```
void disegna Rettangolo(int coordX, int coordY, int larghezza, int altezza, int colore){  
    display.clearDisplay();  
    if(colore==0) display.drawRect(coordX, coordY, larghezza, altezza, WHITE);  
    if(colore==1) display.drawRect(coordX, coordY, larghezza, altezza, INVERSE);  
    display.display();  
}  
//disegna Rettangolo()
```

```
//*****
```

```
//DISEGNA UN CERCHIO
```

```
void disegna_cerchio(int centroX, int centroY, int diametro, int colore){  
    display.clearDisplay();  
    if(colore==0) display.drawCircle(centroX, centroY, diametro, WHITE);  
    if(colore==1) display.drawCircle(centroX, centroY, diametro, INVERSE);  
    display.display();  
}  
//disegna_cerchio()
```



```
//*****
```

```
//DISEGNA UN RETTANGOLO ARROTONDATO
```

```
void disegna_rettangolo_arrotondato(int coordX, int coordY, int larghezza, int altezza, int raggio, int colore){  
    display.clearDisplay();  
    if (colore==0) display.drawRoundRect(coordX, coordY, larghezza, altezza, raggio, WHITE);  
    if (colore==1) display.drawRoundRect(coordX, coordY, larghezza, altezza, raggio, INVERSE);  
    display.display();  
}  
//disegna_rettangolo_arrotondato()
```

```
void loop() {
```

```
    int larghezza,altezza;  
    int valore;
```

```
    //LEGGO LARGHEZZA ED ALTEZZA DEL DISPLAY  
    larghezza=display.width();  
    altezza=display.height();
```

```
    //visualizzo un valore in un rettangolo
```

```
    valore=100;  
    display.clearDisplay();  
    disegna_rettangolo_arrotondato(10,10,100,20,3,0);  
    display.setTextSize(2);  
    display.setTextColor(WHITE);  
    display.setCursor(12,12);  
    display.println(valore);  
    display.display();  
    delay(2000);
```

```
    //disegno un singolo pixel
```

```
    disegna_pixel(10,20,0);  
    delay(200);  
    disegna_pixel(20,30,1);  
    delay(200);
```

```
    //disegno una linea
```

```
    disegna_linea(0,0,larghezza,altezza,0);  
    delay(200);  
    disegna_linea(0,0,larghezza,altezza,1);  
    delay(200);
```

```
//disegno un rettangolo
disegna_rettangolo(0,0,larghezza,altezza,0);
delay(200);
disegna_rettangolo(0,0,larghezza,altezza,1);
delay(200);
//disegno un cerchio
disegna_cerchio(larghezza/2,altezza/2,30,0);
delay(200);
disegna_cerchio(larghezza/2,altezza/2,30,1);
delay(200);
//disegno un rettangolo arrotondato
disegna_rettangolo_arrotondato(0,0,larghezza-2,altezza-2,altezza/4,0);
delay(200);
disegna_rettangolo_arrotondato(0,0,larghezza-2,altezza-2,altezza/4,1);
delay(200);

//scroll nelle due direzioni
display.startscrollright(0x00, 0x0F);
delay(2000);
display.stopscroll();
delay(1000);
display.startscrollleft(0x00, 0x0F);
delay(2000);
display.stopscroll();
delay(1000);
}
```