

My First Robot



My First Robot

Card 1 of 10

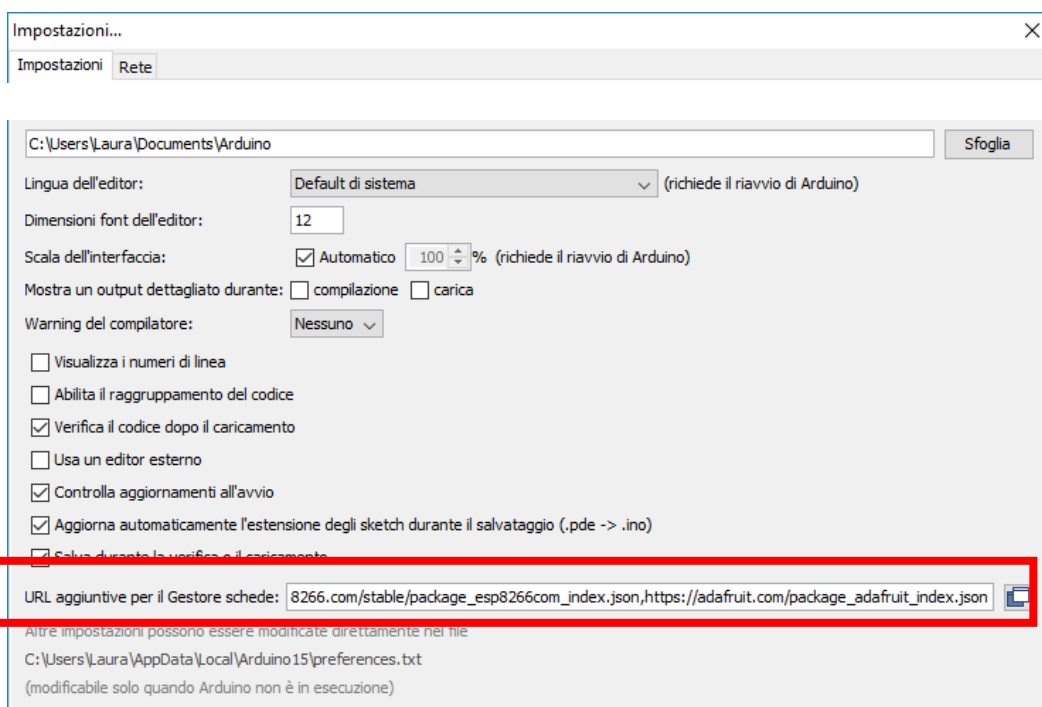
I'm Learning Robotics

1 Scarica l'ultima versione della Arduino IDE (<https://www.arduino.cc/en/Main/Software>)



2 Ora apri Arduino e seleziona **File->Preferences (File -> Impostazioni)** e copia questa url nel campo "Additional Board Manager URLs" ("URL aggiuntive per il gestore schede")

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



and click **OK**

My First Robot



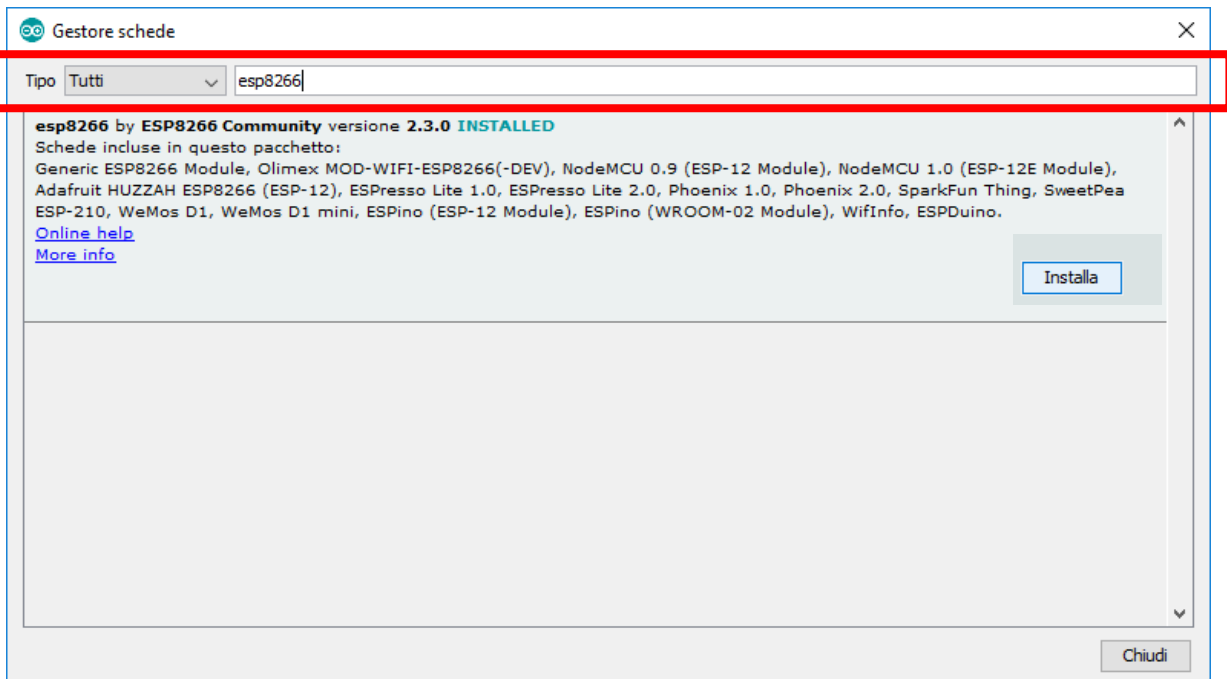
My First Robot

Card2of 10

I'm Learning Robotics

3 Infine vai sul menu **Tools -> Boards -> Boards Manager (Strumenti -> Schede -> Gestore Schede)**

Scrivi `esp8266` nel campo di ricerca



Seleziona le voci con `esp8266` e clicca **Install**

Ben Fatto! Iniziamo a programmare il tuo comPVterBot!

My First Robot



My First Robot

Card3of 10

I'm Learning Robotics

In questo esercizio impariamo a collegare ed accendere un led di comPVterBot.

- 1 Collega il led verde al PIN D8.
- 2 Collega il led verde al PIN D8.
- 3 Apri la IDE Arduino.
- 4 Specifichiamo I PIN cui sono associati i LED

```
/*
```

```
  Blink
```

```
  Accende e spegne un led verde e uno rosso per un secondo di continuo
```

```
*/
```

```
#define ledVerde D8 // qui specifichiamo quale pin dell Arduino collegare il led: D8 per il verde...
#define ledRosso D7 // ...D7 per il rosso
// la funzione setup viene eseguita SOLO UNA VOLTA all'avvio (e di conseguenza quando viene premuto il tasto di reset)
void setup() {
  // specifichiamo che i pins che abbiamo scelto saranno di OUTPUT (molti pin possono essere sia input che output)
  pinMode(ledVerde, OUTPUT);
  pinMode(ledRosso, OUTPUT);
  Serial.begin(9600);
}
```



My First Robot



My First Robot

Card4of 10

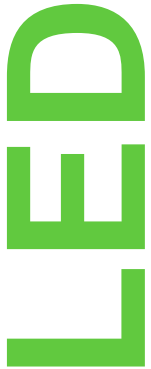
I'm Learning Robotics

5 Nel loop (eseguito continuamente) accendiamo e spegniamo il lwd verde ed accendiamo e spegniamo il led rosso

// la funzione "loop" viene eseguita continuamente

```
void loop() {  
  Serial.println("Serial connection OK!!");  
  digitalWrite(ledVerde, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(200); // wait for a second  
  digitalWrite(ledVerde, LOW); // turn the LED off by making the voltage LOW  
  delay(200); // wait for a second  
  digitalWrite(ledRosso, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(200); // wait for a second  
  digitalWrite(ledRosso, LOW); // turn the LED off by making the voltage LOW  
  delay(200);  
}
```

6 Evviva! Il nostro robot può **parlare coi colori!**



My First Robot



My First Robot

Card5of 10

I'm Learning Robotics

1 Apri la Arduino IDE.

Questo è il codice di setup. Dobbiamo dire al microcontrollore cosa c'è agganciato a pin. D1 e D2 controllano la velocità dei due motori. D3 e D4 la direzione dei motori. Sono tutti pin di OUTPUT.

```
#define pin_speedA D1
#define pin_speedB D2
#define pin_dirA D3
#define pin_dirB D4
```

```
int speedA=255;
int speedB=255;
bool dirA=0;
bool dirB=0;
```

```
void setup()
{
    pinMode(pin_speedA,OUTPUT);
    pinMode(pin_speedB,OUTPUT);
    pinMode(pin_dirA,OUTPUT);
    pinMode(pin_dirB,OUTPUT);
}
```

2 Ora creiamo le funzioni per avanti, indietro, sinistra e destra

```
void avanti()
{
    analogWrite(pin_speedA, speedA);
    digitalWrite(pin_dirA, dirA);
    analogWrite(pin_speedB, speedB);
    digitalWrite(pin_dirB, dirB);
}
```

```
void indietro()
{
    analogWrite(pin_speedA, speedA);
    digitalWrite(pin_dirA, !dirA);
    analogWrite(pin_speedB, speedB);
    digitalWrite(pin_dirB, !dirB);
}
```

```
void destra()
{
    analogWrite(pin_speedA, speedA);
    digitalWrite(pin_dirA, !dirA);
    analogWrite(pin_speedB, speedB);
    digitalWrite(pin_dirB, dirB);
}
```

```
void sinistra()
{
    analogWrite(pin_speedA, speedA);
    digitalWrite(pin_dirA, dirA);
    analogWrite(pin_speedB, speedB);
    digitalWrite(pin_dirB, !dirB);
}
```

MOTORI

3 Dobbiamo poterci fermare

```
void Stop()
{
    analogWrite(pin_speedA, 0);
    analogWrite(pin_speedB, 0);
}
```

4 Ora mettiamo tutto insieme

```
void loop()
{
    avanti();
    delay(1000);
    indietro();
    delay(1000);
    destra();
    delay(1000);
    sinistra();
    delay(1000);
    stop();
    delay(1000);
...e }
risolviamo il labirinto!
```

MOTORI

In questo esercizio eseguiamo un **test della distanza misurata** con il sensore ad ultrasuoni HC-SR04.

Questo sensore può inviare un **onda ultrasonica** e **misurare il tempo impiegato** a ricevere l'onda riflessa.

Collegamento dei pins tra il sensore e l'Arduino:

- VCC deve essere collegato a 5v
- GND a GND
- Trig al pin D7 : inviano un impulso a questo pin del sensore, questo fa partire l'onda
- Echo al pin D6 : su questo pin riceveremo un impulso della durata del tempo impiegato

1 Apri la IDE Arduino.

2 Crea il file Ultrasonic.ino.

Prima impostiamo le variabili e i pin che andremo ad usare

```
#define echoPin D6 // Echo Pin
#define trigPin D5 // Trigger Pin

#define echoPin D6 // Echo Pin
#define trigPin D5 // Trigger Pin
#define LEDPin D7 // Pin dove collegare il LED (si consiglia verde con il
catodo a GND, così che faccia da 'semaforo': verde intenso ="tutto libero")
```

```
long durata, distanza; // misureremo la durata dell'impulso restituito dal
sensore per calcolare la distanza
```

```
void setup() {
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(LEDPin, OUTPUT); // Use LED indicator (if required)
}
```

ULTRASONIC

3 Inviamo un impulso basso-alto-basso sul pinTrigger per lanciare l'ultrasuono, quindi calcoliamo la distanza sulla base del tempo impiegato dal segnale a tornare

```
void loop() {  
  ////inviamo un impulso basso-alto-basso sul pinTrigger per lanciare l'ultrasuono  
  digitalWrite(trigPin, LOW);  
  delayMicroseconds(2);  
  digitalWrite(trigPin, HIGH);  
  delayMicroseconds(10);  
  digitalWrite(trigPin, LOW);  
  
  durata = pulseIn(echoPin, HIGH);  
  //Calcoliamo la distanza (in cm) in base alla velocità del suono  
  distanza = durata/58.2;  
  
  // inviamo la misura della distanza al computer via seriale  
  Serial.println(distanza);  
  // utilizziamo la luminosità del led per avere un'idea della distanza misurata  
  analogWrite(LEDpin, distanza);  
  //è necessario un ritardo di 50ms prima della prossima lettura del sensore  
  delay(50);  
}
```

ULTRASONIC

My First Robot



My First Robot

Card9of 10

I'm Learning Robotics

In questo esercizio accendiamo un led dallo smartphone o PC via WiFi!

1 collegare il led sul pin 8

2 modificare il programma seguente specificando il nome del WiFi e la password a cui ci si collegherà con il PC o lo smartphone

```
#include <ESP8266WiFi.h>
#define ledPin D8

const char* ssid = "MyWiFiName";
const char* password = "MyPassword";

WiFiServer server(80);
```

3 modificare il programma seguente specificando il nome del WiFi e la password a cui ci si collegherà con il PC o lo smartphone

```
void setup() {
  Serial.begin(9600);
  delay(10);

  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  // Connect to WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);
```

```
  while (WiFi.status() !=
  WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");

  // Start the server
  server.begin();
  Serial.println("Server started");

  // Print the IP address
  Serial.print("Use this URL to
  connect: ");
  Serial.print("http://");
  Serial.print(WiFi.localIP());
```

WiFi

My First Robot



My First Robot

Card 10 of 10

I'm Learning Robotics

- 4 dopo aver caricato il programma sull'Arduino, aprire il monitor seriale
- 5 Dopo qualche puntino ("..."), comparirà l'indirizzo IP assegnato all'Arduino
- 6 connettersi a quell'indirizzo da pc o smartphone
- 7 comparirà una pagina con 2 pulsanti: uno per accendere e uno per spegnere il led
- 8 enjoy it!

```
void loop() {
  // Check if a client has connected
  WiFiClient client = server.available();
  if (!client) {
    return;
  }

  // Wait until the client sends some data
  Serial.println("new client");
  while(!client.available()){
    delay(1);
  }

  // Read the first line of the request
  String request =
  client.readStringUntil('\r');
  Serial.println(request);
  client.flush();
  // Match the request

  int value = LOW;
  if (request.indexOf("/LED=ON") != -1) {
    digitalWrite(ledPin, HIGH);
    value = HIGH;
  }
  if (request.indexOf("/LED=OFF") != -1)
  {
    digitalWrite(ledPin, LOW);
    value = LOW;
  }
}
```

```
// Set ledPin according to the request
//digitalWrite(ledPin, value);

// Return the response
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println(""); // do not forget this one
client.println("<!DOCTYPE HTML>");
client.println("<html>");

client.print("Led pin is now: ");

if(value == HIGH) {
  client.print("On");
} else {
  client.print("Off");
}
client.println("<br><br>");
client.println("<a
href='\"/LED=ON\"'><button>Turn On
</button></a>");
  client.println("<a
href='\"/LED=OFF\"'><button>Turn Off
</button></a><br />");
  client.println("</html>");

  delay(50);
  Serial.println("Client disconnected");
  Serial.println("");
}
```

WiFi