



# Learning Arduino

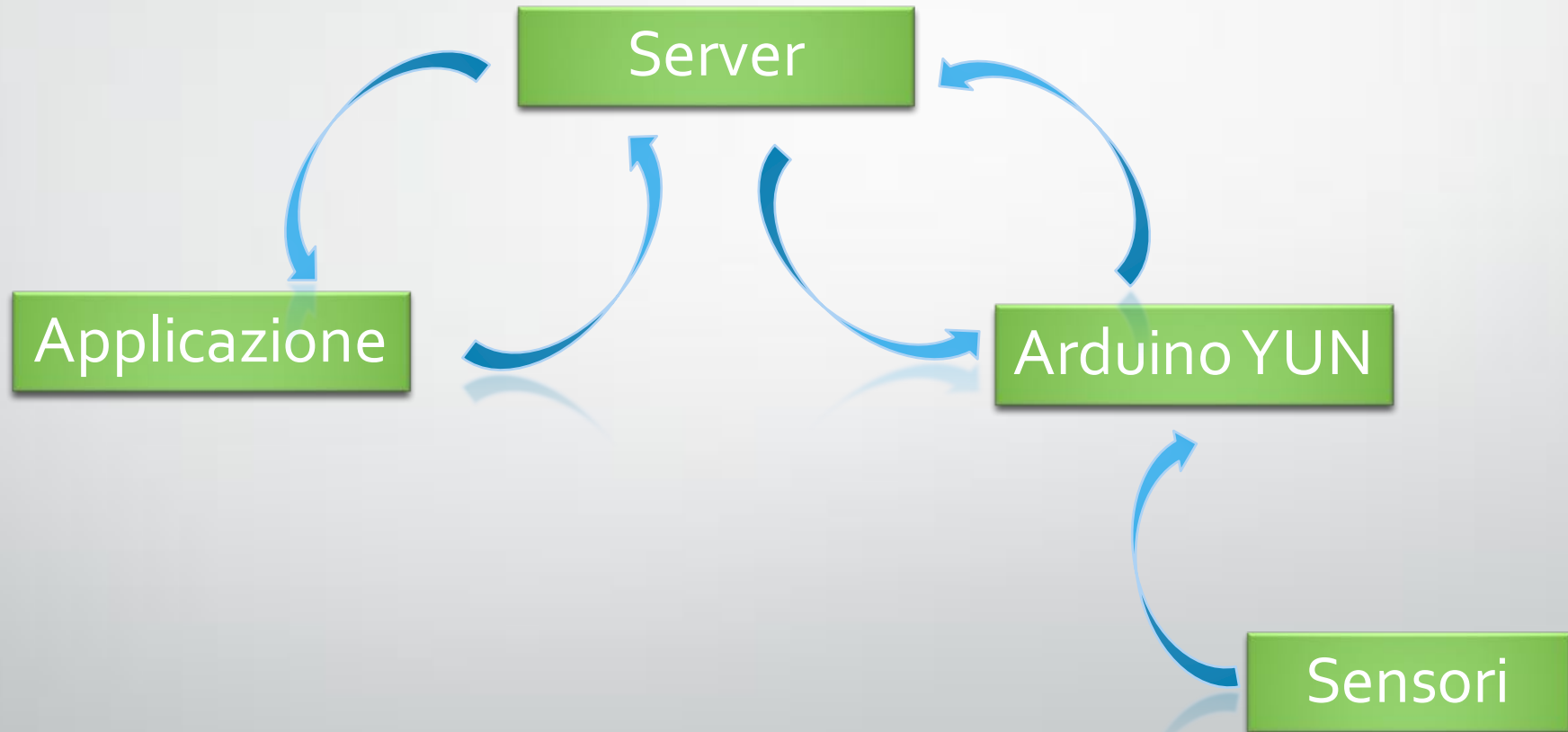
# Cosa facciamo oggi

- Creare un sistema IoT che rilevi delle grandezze fisiche e le mostri su una pagina web

# IoT

- Che cos'è l' IoT?
- Che utilità ha?

# IoT – Memory Safe



# Cos'è Arduino Yùn?



- Board che permette la connessione di sensori e attuatori e l'interazione tra loro
- Comprende un attacco Micro USB, USB, Ethernet e una scheda WiFi integrata
- Consente la connessione con tutti i KIT Arduino compatibili

# Come collegare il proprio Arduino al PC

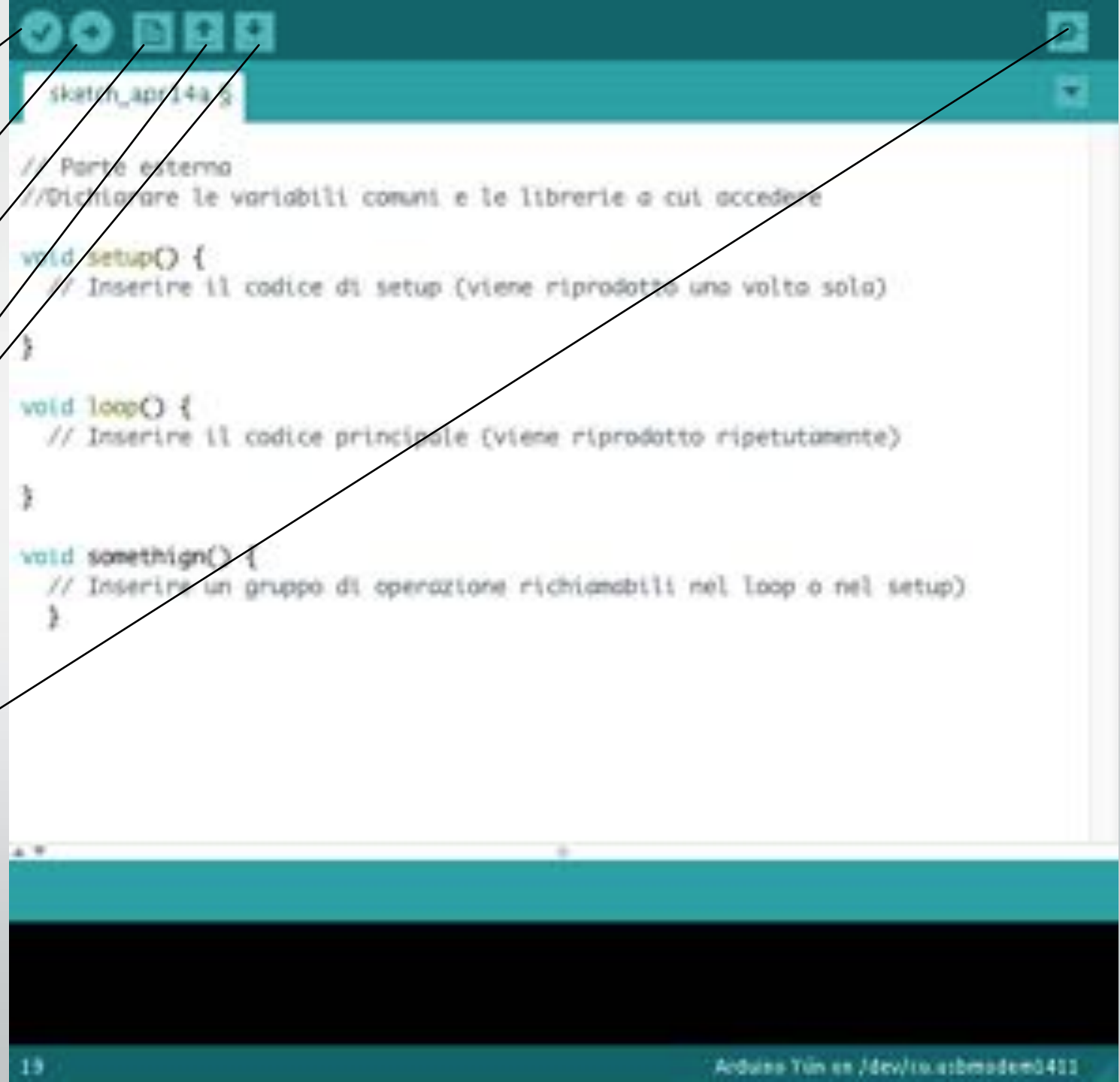
- È necessario un cavo USB/micro USB
- Il cavo funge sia da collegamento che da alimentazione
- Per ogni modifica sulla board (collegare/scollegare sensori) è necessario scollegarla dal PC

# Come programmare Arduino

- La board è programmabile tramite l'apposito software Arduino scaricabile dal link:  
<https://www.arduino.cc/en/Main/Software>
- Lo sketch è il tipo di software che si usa per controllare Arduino

# Lo Sketch

1. Verifica il codice
2. Carica il codice sulla board
3. Apri nuovo sketch
4. Apri sketch esistente
5. Salva sketch
6. Apri Monitor Seriale





sketch\_apr14a 5

```
// Parte esterna  
// Dichiarare le variabili comuni e le librerie a cui accedere
```

- Inserire variabili o costanti comuni a tutto lo sketch
- Richiamare le librerie che lo sketch dovrà utilizzare

```
void setup() {  
  // Inserire il codice di setup (viene riprodotto una volta sola)  
}
```

- Inserire tutto il codice necessario per impostare la board
- Il codice all'interno di questa sezione verrà eseguito una volta solamente dopo l'accensione.

```
void loop() {  
    // Inserire il codice principale (viene riprodotto ripetutamente)  
}
```

- Inserire il codice principale, cioè quella parte che serve per controllare i sensori durante tutta la fase del processo.
- È utile ogni tanto inserire un ritardo (funz. `delay(1000)`) per evitare un sovraccarico dei dati in input/output.



- Sezione di controllo.
- Errori rilevati dal software stesso durante la fase di debugging.



- Monitor Seriale
- Mostra messaggi rilevati dai sensori e opportunamente stampati

# Consigli finali

**Dopo ogni funzione, variabile o costante va inserito un punto e virgola.**

È sempre utile commentare le righe di codice scritte specificando di cosa si tratti, per inserire un commento (cioè una parte di codice che non verrà letta al momento dell'esecuzione) è sufficiente inserire prima del commento:

- `// commento` se si tratta di un commento di un'unica riga
- `/*  
*commento  
*/` se si tratta di un commento di più righe

# Project work 1

- 5 gruppi da 4 persone
- 1 Arduino Yun, 1 Tinker Kit, 2 sensori, 1 Led per ogni gruppo





# Obiettivi

Costruire un semplice esempio di un sistema hardware/software funzionante

A) Montare la parte hardware

B) Creare uno sketch che:

- Rilevi una o più grandezze fisiche attraverso sensore/i
- Stampi il risultato attraverso il monitor seriale
- Eventualmente piloti uno o più output (LED)

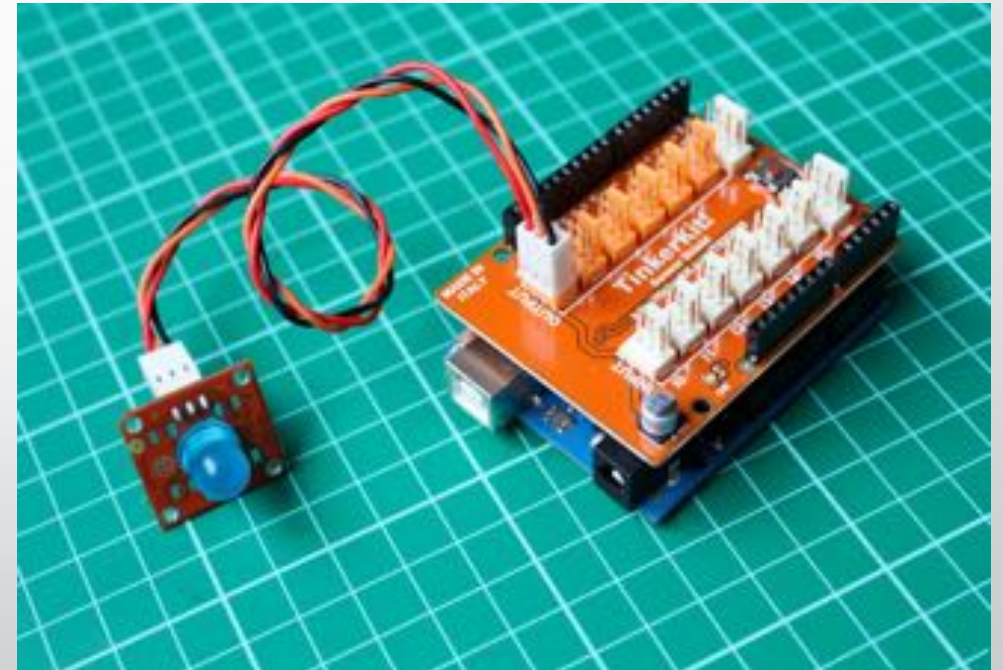
# Montaggio – Fase 1

Inserite la board del tinkerkit  
sullo Yun come da figura



# Montaggio – Fase 2

- Collegate i sensori/attuatori alle prese della board
- Sensori (temperatura ecc.) ad una presa di input
- Attuatori (led, buzzer ecc.) ad una presa di output
- Le prese a cui collegate i sensori dovranno corrispondere nello Sketch



# Montaggio – Fase 3

- Collegate lo Yun al vostro computer tramite cavo USB / Micro USB



# Programmazione iniziale

- Aprire gli esempi già forniti:  
File → Esempi → Tinkerkits
- Includere fin dall'inizio la libreria del tinkerkits:  
Sketch → Include library → Tinkerkits

# Creare una variabile

- Digitare: `int nomeVariabile;`
- La prima parte rappresenta il tipo di variabile di cui avete bisogno, in questo caso un intero (int)
- La seconda parte rappresenta invece il nome che volete assegnargli
- Non utilizzate caratteri particolari ne spazi altrimenti il software darà errore
- Attenzione è case sensitive

# Come visualizzare una variabile sul monitor

- Nella sezione di setup:  
Serial.begin inizializza il monitor seriale  
Il while serve invece ad attendere che voi abbiate aperto il monitor prima di far partire il programma
- Nella sezione di loop:  
Serial.print stamperà il vostro dato  
Serial.println farà la medesima operazione ma dopo andrà a capo  
In caso vogliate visualizzare una stringa di caratteri inserite al posto di nome variabile la stringa prescelta tra virgolette

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

```
Serial.print(nomeVariabile);  
  
Serial.println(nomeVariabile);
```

# Project work 2

- Utilizzare lo Sketch creato in precedenza per inviare dati ad un server e visualizzarli su una freeboard



# Fase 1

Aggiungere allo sketch le librerie:

- PubSubClient
- Wifi
- Bridge

Lo si può fare manualmente o digitando le seguenti righe di codice:

```
#include <SPI.h>

#include <PubSubClient.h>

#include <WiFi.h>

#include <YunClient.h>
```

# Fase 2

Impostare la connessione al server digitando le seguenti righe di codice:

```
#define MQTT_SERVER "40.87.158.158"
YunClient yun;
void callback(char* topic, byte* payload, unsigned int length) {
    // handle message arrived
}
PubSubClient client(MQTT_SERVER, 80, callback, yun);
```

# Fase 3

- Inizializzare la connessione digitando all'interno del setup:  
`Bridge.begin();`

# Fase 4

Inserire all'interno del loop un if che verifichi la connessione al server:

```
void loop() {  
  if (mqtt.connect("YUNClient", "test", "test")) {  
  
    //qui inserite il vostro codice che legge i dati del sensore  
  
  }  
}
```

# Fase 5

Convertire il vostro dato in uno leggibile dal server

```
void loop() {  
  if (mqtt.connect("YUNClient", "test", "test")) {  
  
    //qui inserite il vostro codice che legge i dati del sensore  
    String nomeString = String(nomeVariabile);  
  
    char nomeChar[10];  
  
    nomeString.toCharArray(nomeChar, 10);  
  }  
}
```

# Fase 6

Publicate il dato sul server

- La prima parte rappresenta il Topic o nome con cui volete pubblicare il dato
- La seconda parte corrisponde al nome del Char da pubblicare

```
mqtt.publish("nomeTopic", nomeChar);
```

# Topic da utilizzare

- Gruppo 1: “lumgruppo1”  
“tempgruppo1”
- Gruppo 2: “lumgruppo2”  
“tempgruppo2”
- Gruppo 3: “lumgruppo3”  
“tempgruppo3”
- Gruppo 4: “lumgruppo4”  
“tempgruppo4”
- Gruppo 5: “lumgruppo5”  
“tempgruppo5”