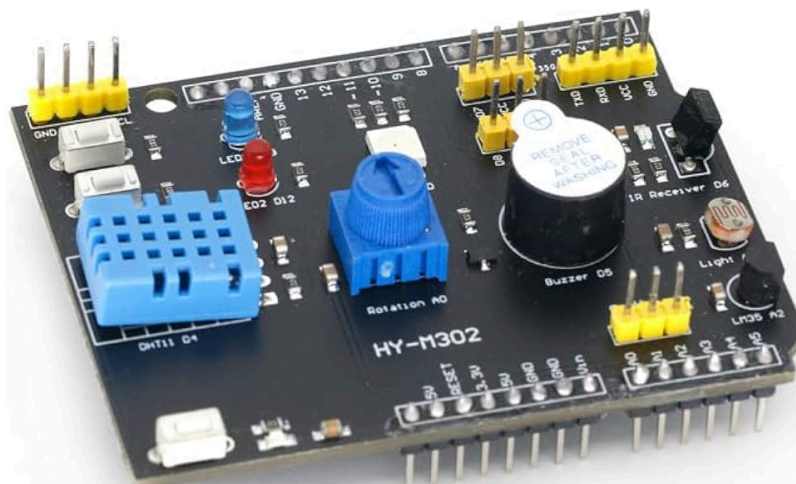


**Zadanie 1.** Sprawdź działanie czujników na płytce rozszerzeń podłączonej do modułu Arduino Mega 2560.



Pobierz program `ex01_01_sensor_board.ino`. Korzystając z narzędzia *Serial Monitor* sprawdź działanie modułu.

**Zadanie 2.** Instalacja środowiska Node-RED na platformie RaspberryPi.

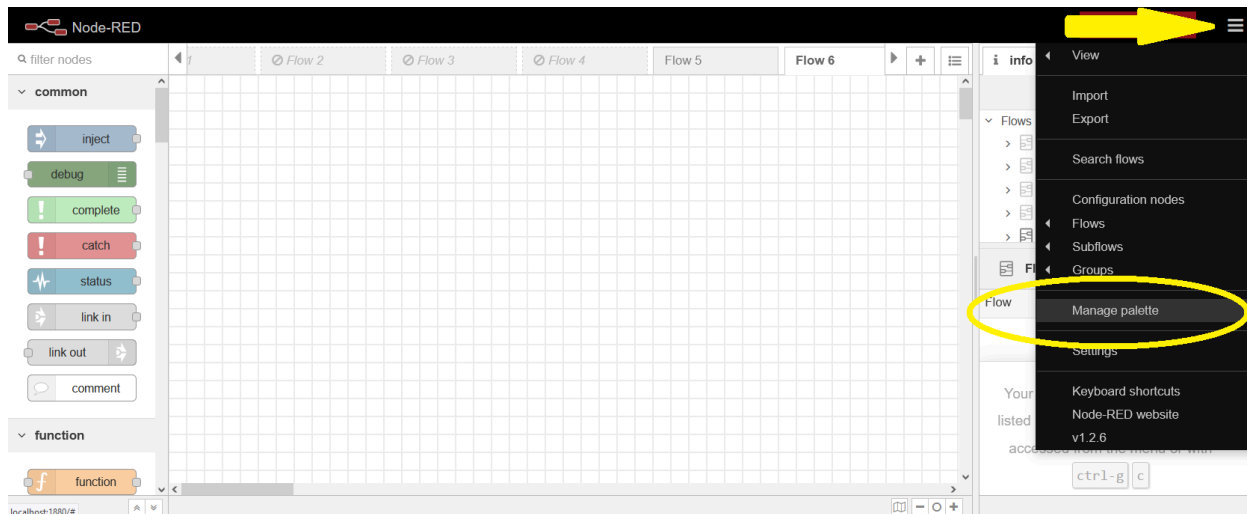
Node-RED jest narzędziem programistycznym, które integruje urządzenia, API oraz usługi sieciowe. Przed przystąpieniem do pracy ze środowiskiem zapoznaj się z instrukcją instalacji dostępną na stronie:

<https://nodered.org/docs/getting-started/local>

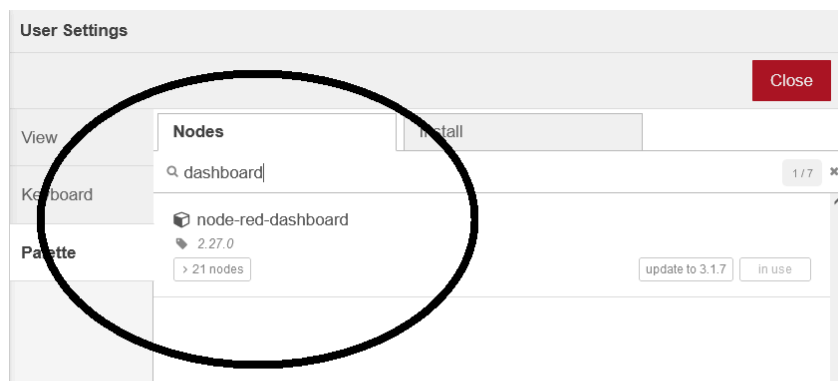
Po zakończeniu instalacji, uruchom dowolną przeglądarkę stron www. W pasku adresu wpisz `localhost:1880`. Interfejs użytkownika jest dostępny pod adresem `localhost:1880/ui`.

## Ćwiczenie nr 1: Wprowadzenie do Node-RED

Opcja *Manage palette* pozwala na zainstalowanie pakietów rozszerzeń:



Zainstaluj pakiety *dashboard* and *node-red-node-serialport*:



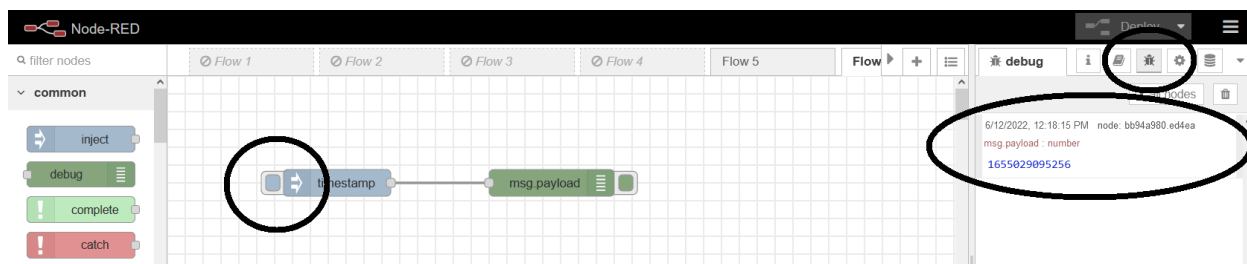
### Zadanie 3. Hello Node-RED.

Stwórz diagram przepływów - potrzebne bloki (*Inject*, *Debug*) znajdują się na palecie *Common*.



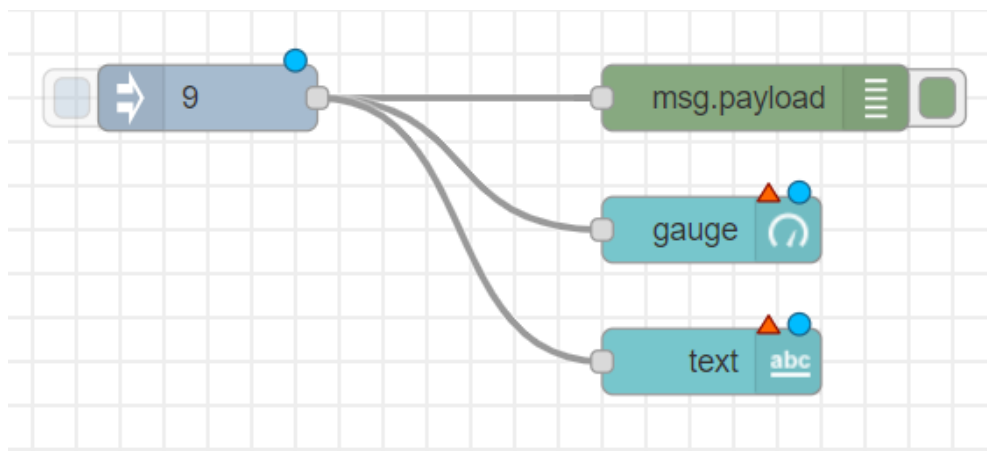
## Ćwiczenie nr 1: Wprowadzenie do Node-RED

Przetestuj diagram korzystając z *Debug messages*.



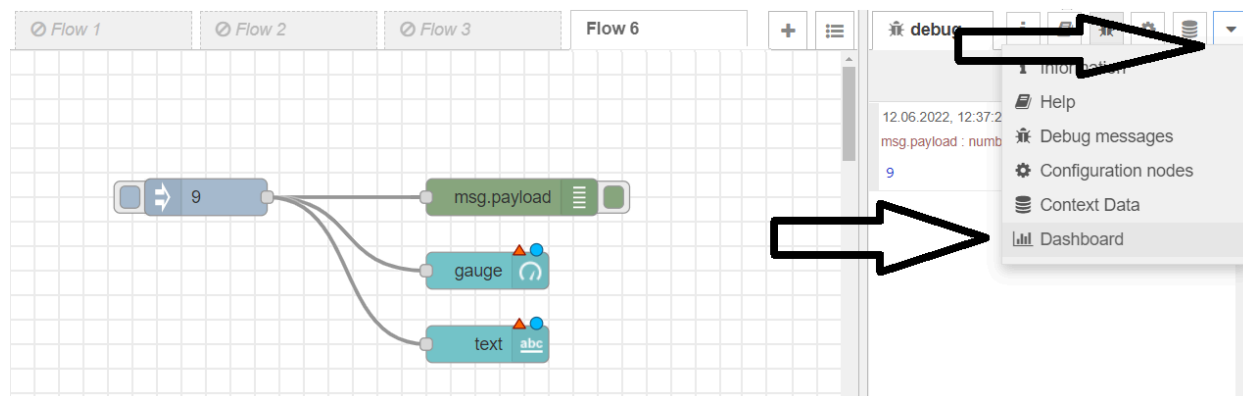
**Zadanie 4.** Budowa interfejsu użytkownika.

Dwa razy kliknij lewym przyciskiem myszy na bloku *Inject*. Zmień *msg.payload* z *timestamp* na *number*. Ustaw wartość liczbową na 9. Dodaj bloki *gauge* and *text* z zakładki *dashboard*.

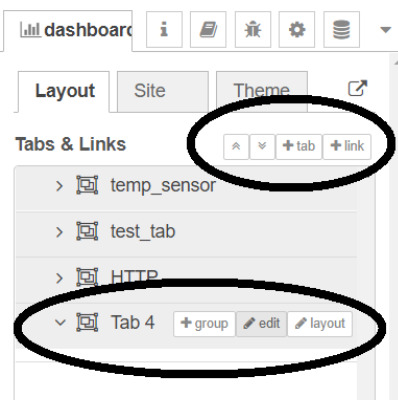


Nawiśnij przycisk *Down* i wybierz *Dashboard* z listy.

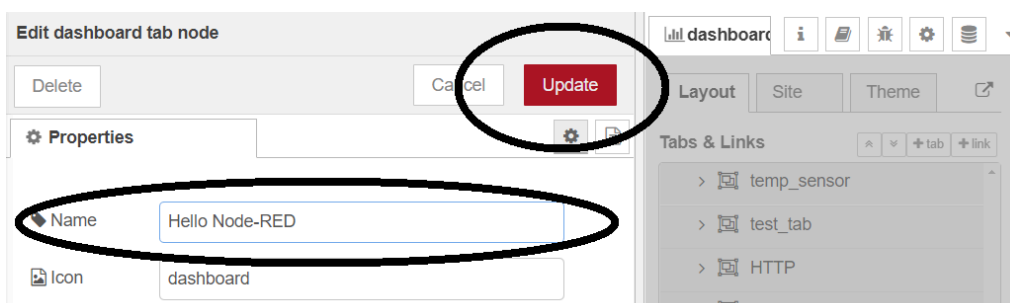
## Ćwiczenie nr 1: Wprowadzenie do Node-RED



Wybierz zakładkę *Layout*. Naciśnij przycisk *+tab* - dodane nowej zakładki. Naciśnij przycisk *edit* - zmiana nazwy zakładki.

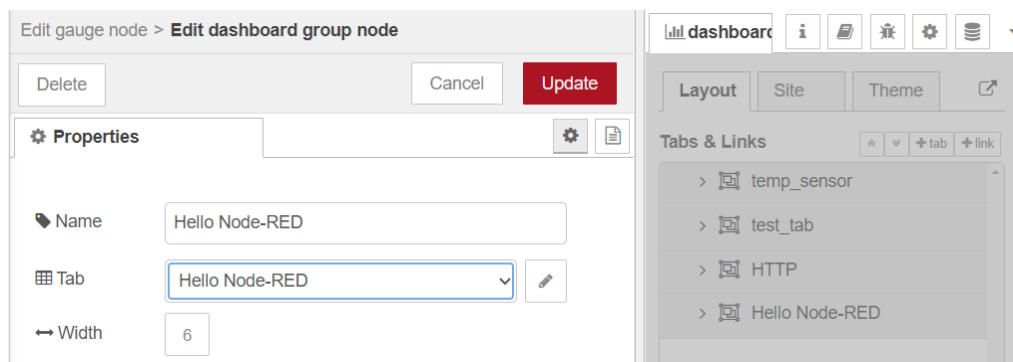


Wprowadź nową nazwę i naciśnij przycisk *Update*.

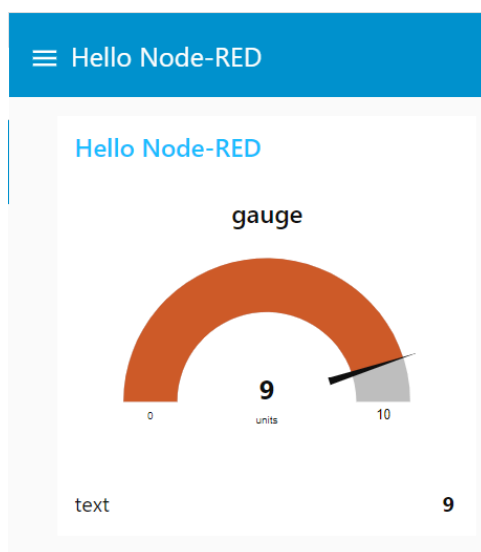


Dwukrotnie kliknij na bloku *gauge*. Wykonaj edycję *Group* i wybierz swoją zakładkę z listy rozwijalnej *Tab*. Zatwierdź zmiany za pomocą przycisku *Update*.

## Ćwiczenie nr 1: Wprowadzenie do Node-RED



Powtórz powyższe operacje dla bloku *text*. Naciśnij *Deploy* w celu aktualizacji diagramu przepływów. Wprowadź wartość za pomocą bloku *Inject*. Sprawdź wynik w *Debug messages*. Otwórz nowe okno w przeglądarce, a w pasku adresu wpisz *localhost:1880/ui*.



### Zadanie 5. Arduino i Node-RED.

Zaprogramuj płytkę Arduino programem testowym `ex01_05_test.ino`.

```
#define BAUDRATE 115200
```

## Ćwiczenie nr 1: Wprowadzenie do Node-RED

---

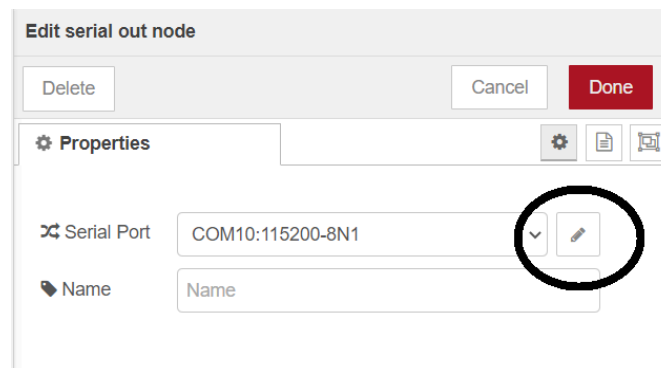
```
unsigned long p_millis = 0;

void setup() {
  Serial.begin(BAUDRATE); }

#define SERIAL_DELAY 1000

void loop() {
  if(millis() - p_millis > SERIAL_DELAY) {
    Serial.println(int(random(1,9)));
    p_millis = millis(); }
}
```

Zamień blok *inject* na blok *serial-in node* (paleta *network*). Wykonaj konfigurację parametrów transmisji.



## Ćwiczenie nr 1: Wprowadzenie do Node-RED

Edit serial out node > Edit serial-port node

Delete Cancel Update

Properties

Serial Port COM10

Settings

Baud Rate 115200 Data Bits 8 Parity None Stop Bits 1

Input

Optionally wait for a start character of , then

Split input on the character \n

and deliver ASCII strings

Output

Add character to output messages

Request

Default response timeout 10000 ms

Tip: the "Split on" character is used to split the input into separate messages. Can accept chars (\$), escape codes (\n), or hex codes (0x03).

Enabled 1 node uses this config On all flows

Sprawdź wynik w oknie *Debug messages*. W pasku adresu dowolnie wybranej przeglądarki wpisz *localhost:1880/ui* w celu obserwacji interfejsu użytkownika.

**Zadanie 6.** (do samodzielnego wykonania) Wyświetl dane z czujnika temperatury lub z potencjometru na interfejsie, w formie tekstowej, na wskaźniku oraz na wykresie.

### Dla zainteresowanych:

1. Node-RED:

[nodered.org](https://nodered.org)

2. Arduino - komunikacja z użyciem portu szeregowego:

[www.arduino.cc/reference/en/language/functions/communication/  
serial/](http://www.arduino.cc/reference/en/language/functions/communication/serial/)