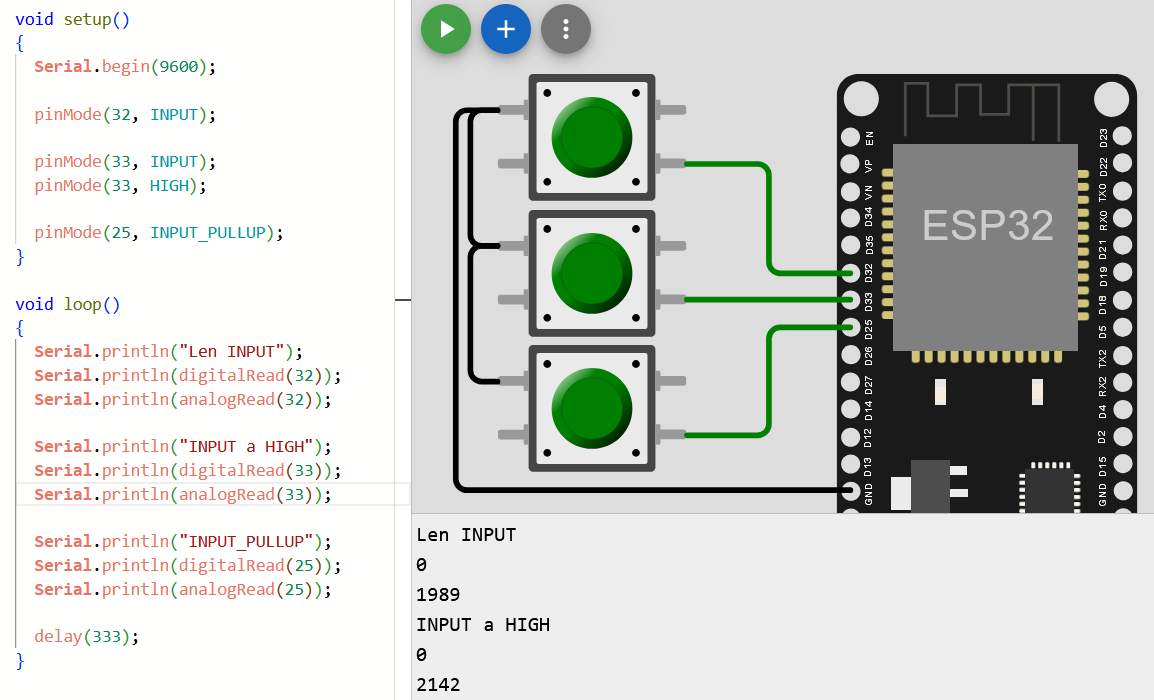
**ESP32**

Mikrokontrolér ESP32 je nástupcom ESP8266. Oproti Atmega328 má dve jadrá, integrovaný BT a WIFI, pracovnú frekvenciu cca 240 MHz, 4/16 MB Flash pamäť, 512 kB RAM a mnoho iný výhod, pričom rozmerovo a cenovo je porovnateľný. Existuje veľa vývojových dosiek (DEVKIT, NOTEMCU, atď.). Programovať sa dá cez Wiring (jazyk pre Arduino) alebo microPython.

Referencia pinov a periférii: <https://randomnerdtutorials.com/esp32-pinout-reference-gpios/>

Príklad na tlačidlo (internal pullup):



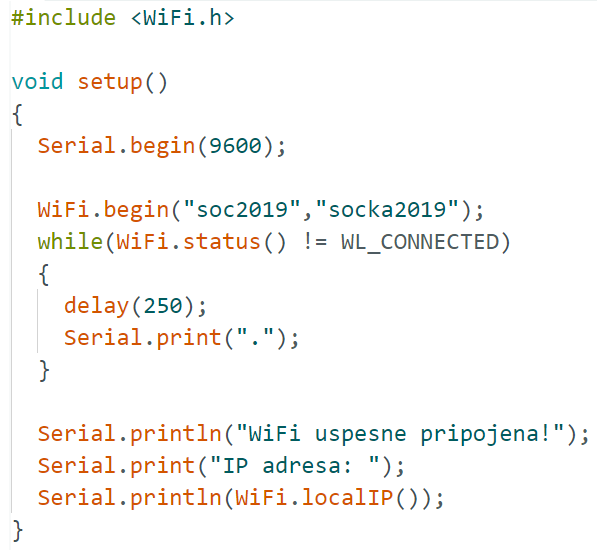
Pozor, niektoré piny sú len vstupné (GPIO 34,35,36,39)!

Pozor, INPUT a HIGH nefunguje pre aktiváciu pull-up odporu. Je nutný príkaz INPUT\_PULLUP!

Pozor, použitím analogRead() sa použije ADC, ktorý odpojí interný pull-up!

ÚLOHA: vytvorte príklad na analogRead, digitalRead, digitalWrite, Serial, LCD.

Príklad na WIFI (ping):



Uvedený príklad pripája ESP32 ku wi-fi sieti s SSID „soc2019“ a heslom „socka2019“ (pri nezabezpečenej stačí uviesť prázdny String). Pokiaľ sa nepodarí pripojiť, vypisuje sa v 250ms intervaloch do konzoly bodka. Po pripojení sa vypíše IP adresa. V tomto okamihu je možné pingnúť ESP32 (pokiaľ je daná sieť dostupná / lokálna).

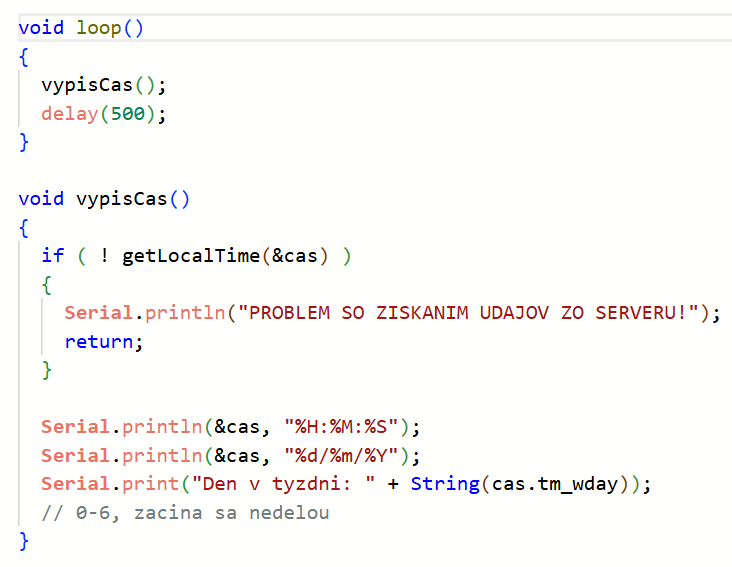
Príklad na WIFI (skenovanie SSID):



Príklad na WIFI (NTP):

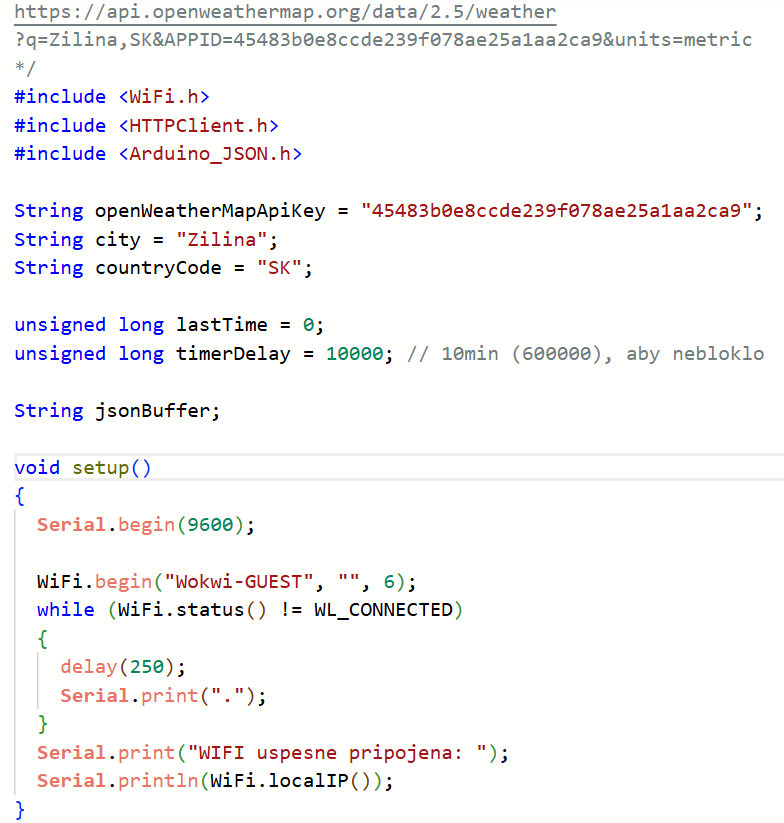






ÚLOHA: naprogramujte ESP32 tak, aby na LCD displeji vypisovalo každú sekundu zvyšujúci sa čas pomocou millis(). Popritom každých 10 sekúnd dôjde ku aktualizácii času pomocou NTP.

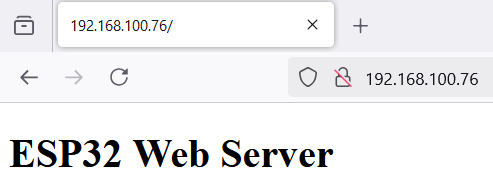
Príklad na WIFI (API + JSON):

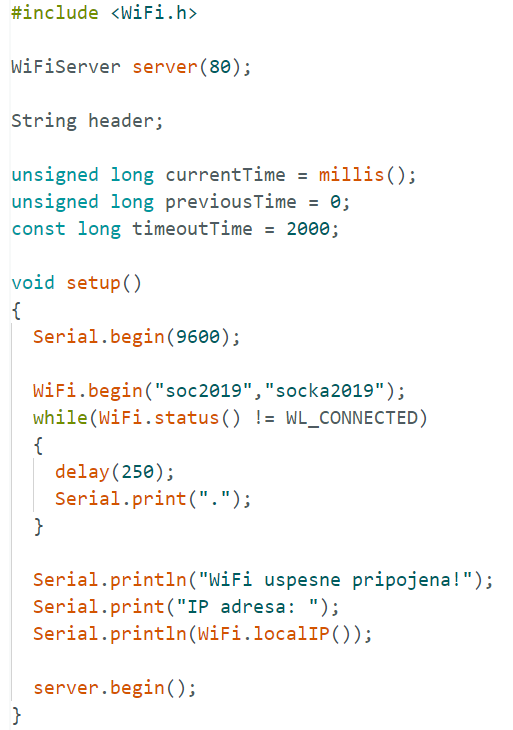






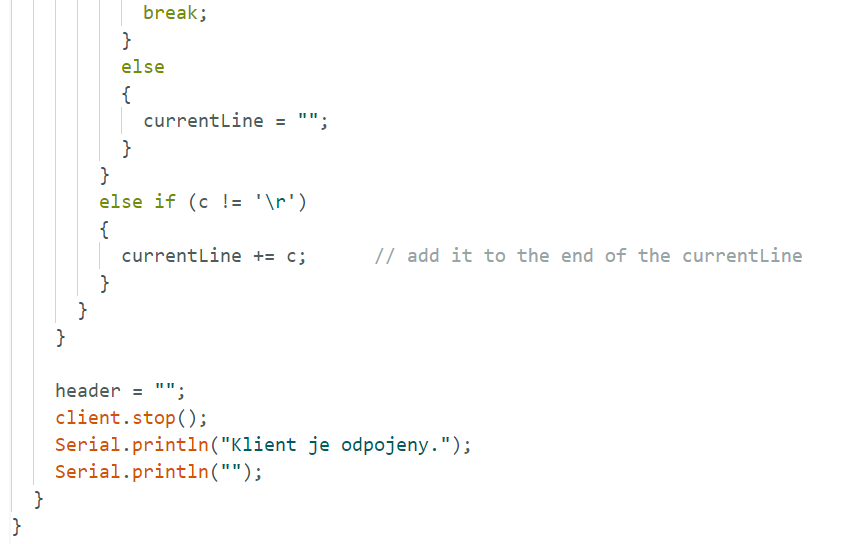
Príklad na WIFI (web server – odosielanie dát):



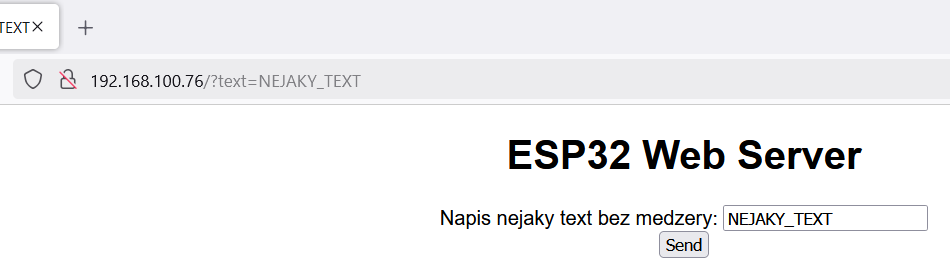






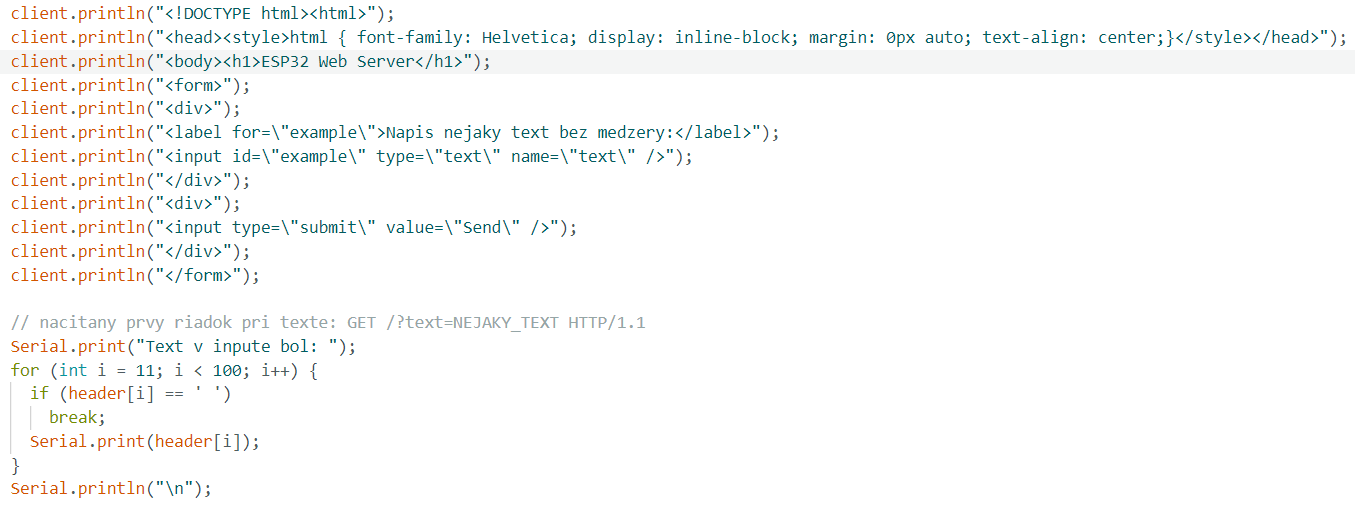


Príklad na WIFI (web server – prijímanie dát v INPUT):

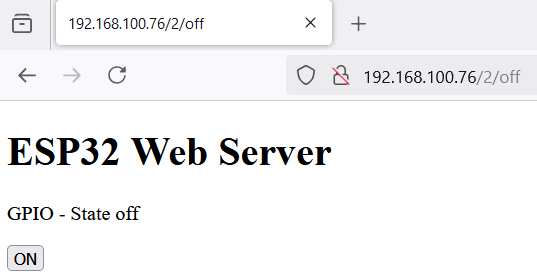


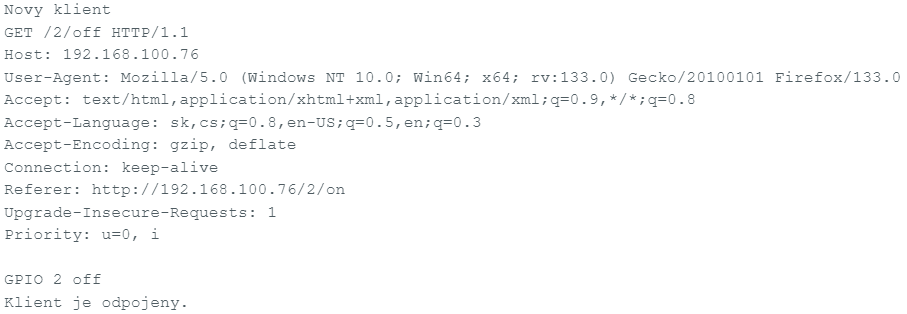


Pridanie kódu do teba web stránky a spracovanie parametrov v URL:



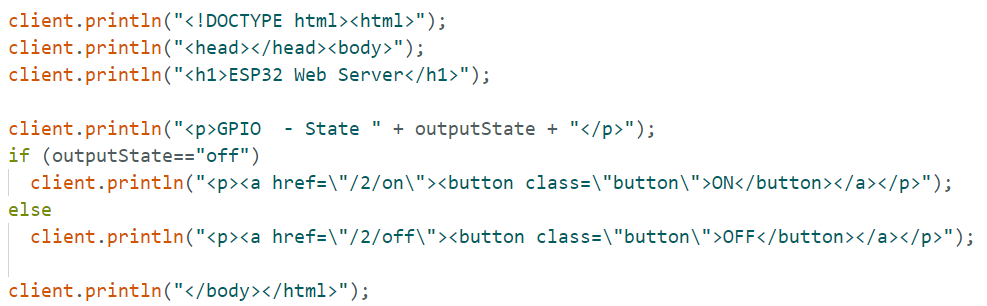
Príklad na WIFI (web server – ovládanie LED cez button):





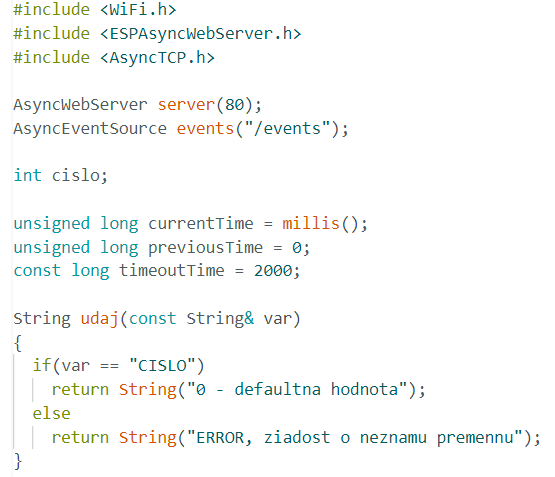
V úvode si treba vytvoriť globálnu premennú outpuState a pin s LED nastaviť ako výstupný. V HTML:





Príklad na WIFI (web server – dynamicky príjem dát pomocou SEE(Server-Sent Events)):

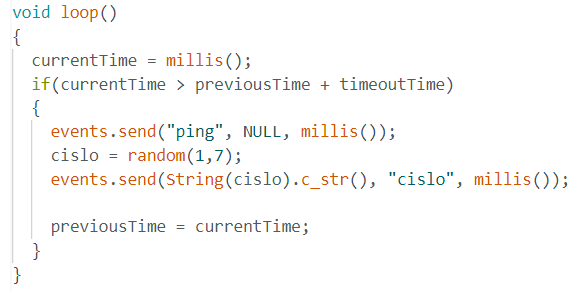
* kód do ESP32









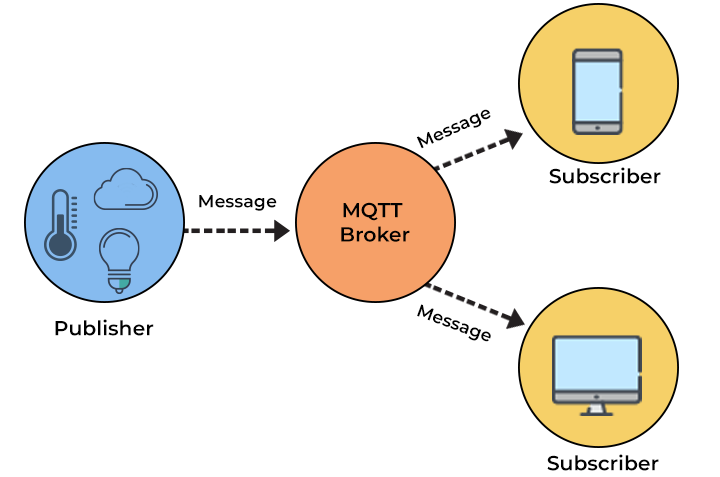


* cez internetový prehliadač sa treba pripojiť na IP ESP32, ktorá sa zobrazí v Serial-y



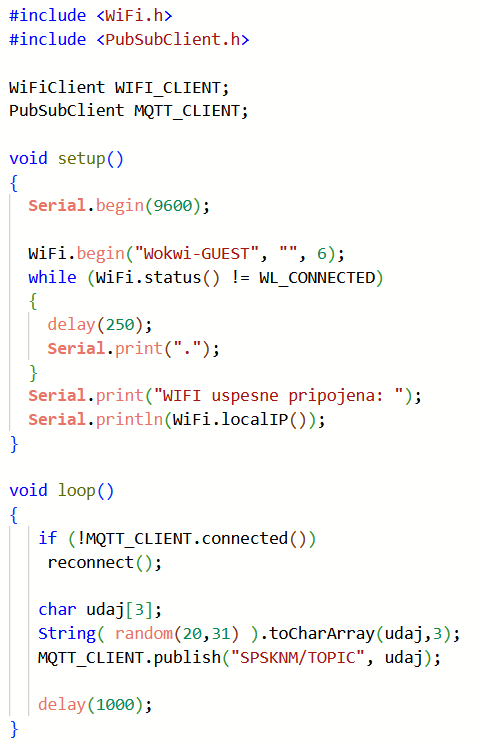
**MQTT**

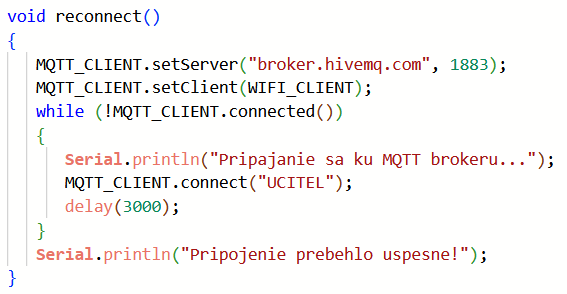
MQTT umožňuje v IoT projektoch prenos dát medzi odosielateľom (publisher) a prijímateľom (subscriber). (Nejde o prenosovú technológiu ako Lora, či ZigBee, ale o prenosový protokol.) Využíva na to prostredníka – server (broker), ktorý zaznamenáva dáta, ak nejaké sú a informuje o nich prijímateľa. V prípade nedostupnosti odosielateľa spojenie od prijímateľa nespadne, akurát sa odošlú posledné aktuálne dáta.



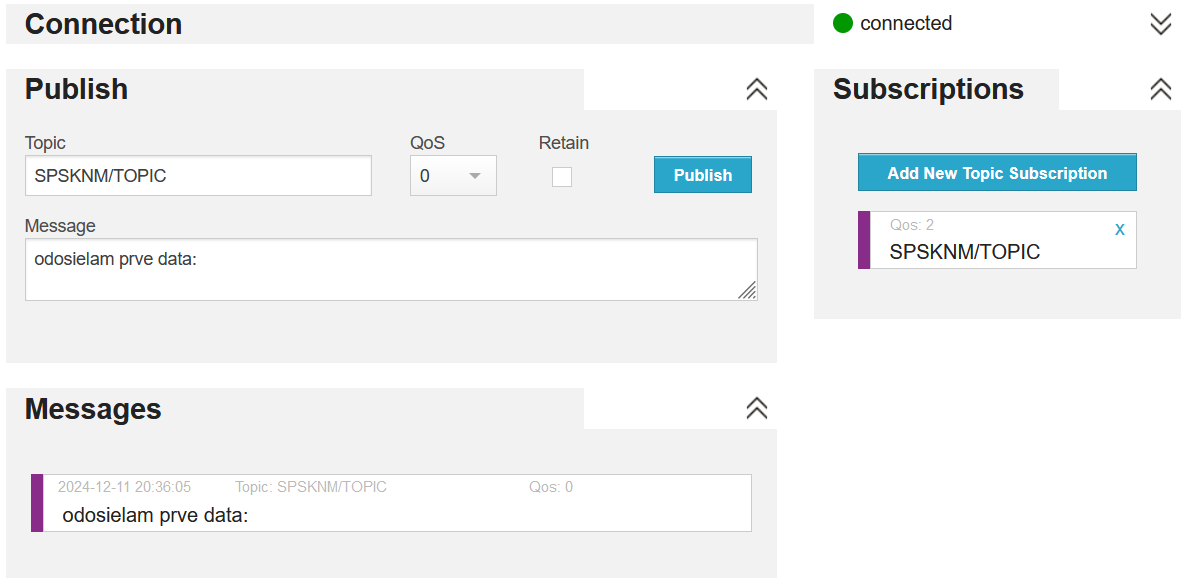
Príklad na WIFI (MQTT):

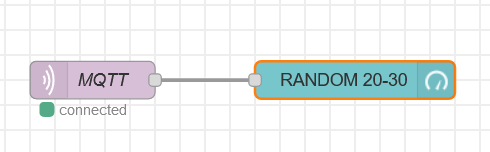
V príklade sa generujú náhodné čísla od 20 do 30 a posielajú sa na MQTT broker v 1 sek. intervale.

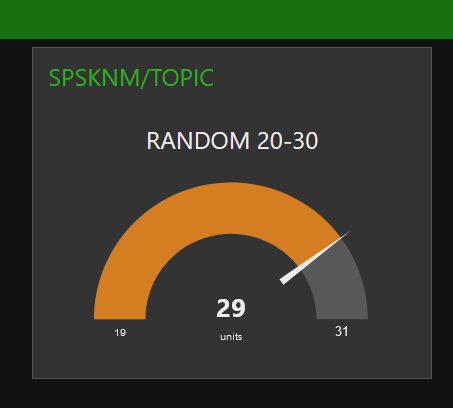




Ako klient sa môže použiť HiveMQ klient na adrese: *//http://hivemq.com/demos/websocket-client/* alebo Node-red server.







Príklad na WIFI (MQTT – prijímanie dát):

Do kódu treba vložiť obsluhu callback(). Tá sa zavolá vždy, keď prídu nejaké dáta do topicu, teda aj od seba samých.



