



AULA PRÁTICA 04 - SERVIDOR WEB

Para mais informações sobre o *NodeMCU* ESP-12E e a instalação dos módulos, reveja o material da aula prática 01.

INTRODUÇÃO

Essa prática servirá para orientar a utilização de um *NodeMCU* para criar um servidor *web* local, bem como uma página HTML simples, que poderá ser acessada por quaisquer dispositivos conectados à mesma rede (*smartphones*, PCs, etc.). Essa página servirá para controlar dois componentes conectados ao *NodeMCU* (dois LEDs, podendo ser substituídos por outros controladores).

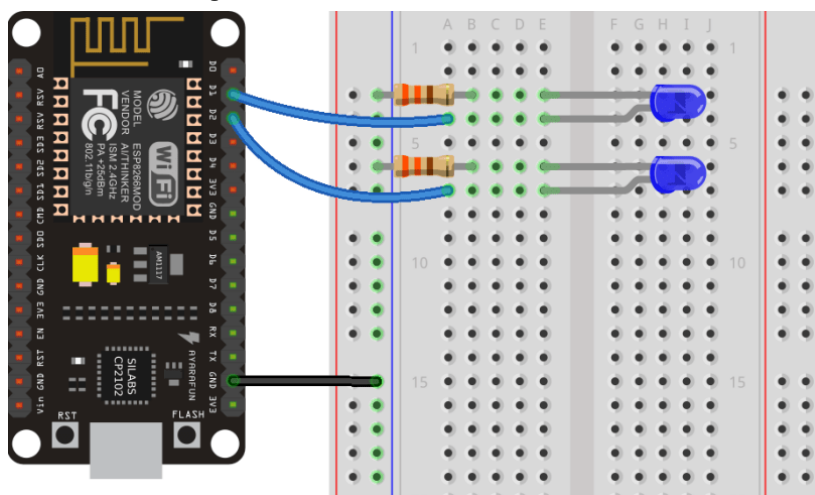
Há diversos exemplos de servidor web já feitos dentro da *Arduino* IDE, instalando-se os módulos do seu *NodeMCU*. Esses exemplos poderão servir como base para projetos futuros, bastando adaptá-los para o que for preciso. Para o ESP 12E, os exemplos podem ser acessados em:

Arquivos → Exemplos → Exemplos para NodeMCU 1.0 ESP12E Module → ESP8266WebServer

PARTE 1 – MONTAGEM DO HARDWARE E EXECUÇÃO DO CÓDIGO

Para essa prática, criaremos um servidor *web* que será acessado por uma página HTML e servirá para controlar seus LEDs. Primeiramente, você deve montar o circuito na placa utilizando 2 LEDs, dispostos como na Figura 1.

Figura 1 – Hardware montado com 2 LEDs.



As portas utilizadas para as conexões com os LEDs foram as portas **D1** e **D2** para o **ESP12** e GPIO **26** e **27** para o **ESP32**. Você pode escolher outras portas, como a D3 em diante, desde que seja propriamente mapeado e modificado no código fornecido. Dessa forma, é importante lembrar que os pinos do *NodeMCU* ESP12 possuem o seguinte mapeamento:

```
D0 = 16;  
D1 = 5;  
D2 = 4;  
D3 = 0;  
D4 = 2;  
D5 = 14;  
D6 = 12;  
D7 = 13;  
D8 = 15;  
D9 = 3;  
D10 = 1;
```

Após a montagem do *hardware*, faça o *download* do código fornecido no *Dropbox* da disciplina. Lá estarão disponíveis os códigos tanto para o ESP12 quanto para o ESP32. No código, você deverá inserir as informações da rede à qual o *NodeMUC* irá se conectar, basta substituir os dados em *SSID* e *Password*, como mostrado na Figura 2:

Figura 2 – Código – Substituição de dados da rede.

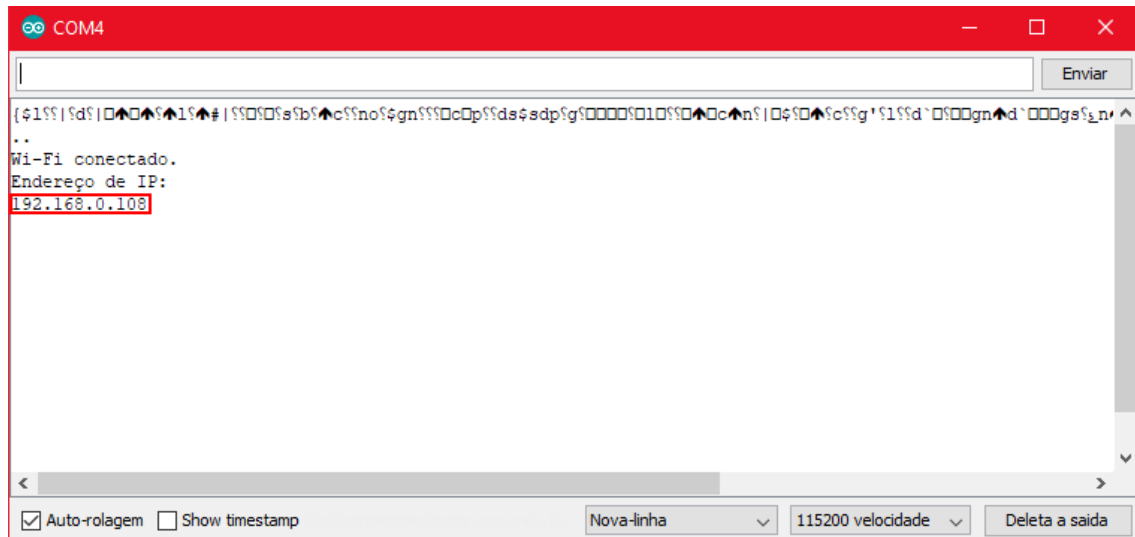


```
Pratica_WebServer_-_CLEAN_8266 | Arduino 1.8.6  
Arquivo Editar Sketch Ferramentas Ajuda  
Pratica_WebServer_-_CLEAN_8266  
// Bibliotecas  
#include <ESP8266WiFi.h>  
  
// Dados da Rede  
const char* ssid = "SSID do WiFi";  
const char* password = "Senha do WiFi";  
  
// Seta Porta 80 para o Servidor Web  
WiFiServer server(80);  
  
// Variável que guarda requisição HTTP  
String header;  
  
// Varáveis Auxiliares: Guarda Estado de Saída  
String outputD1State = "off";  
String outputD2State = "off";  
  
// Mapeamento de Pinos do Node a serem utilizados (Verificar no PDF)  
const int outputD1 = 5;  
const int outputD2 = 4;
```

Observe, também, que o mapeamento dos pinos foram definidos no código, podendo definir-se um novo mapeamento, caso deseje trocar ou adicionar mais portas no sistema. Isso será útil na **Parte 2** da prática.

Após carregar o código, abra o monitor serial e observe se a tela da Figura 3 aparece. Caso a tela esteja em branco, confira se a velocidade mostrada no monitor serial é a mesma do código (no fornecido está em **115200**). Outra opção é dar **RESET** no *NodeMCU*, assim a tela irá aparecer.

Figura 3 – Tela do monitor serial.



Observe que o endereço de IP do seu dispositivo foi exibido. **Copie** esse endereço.

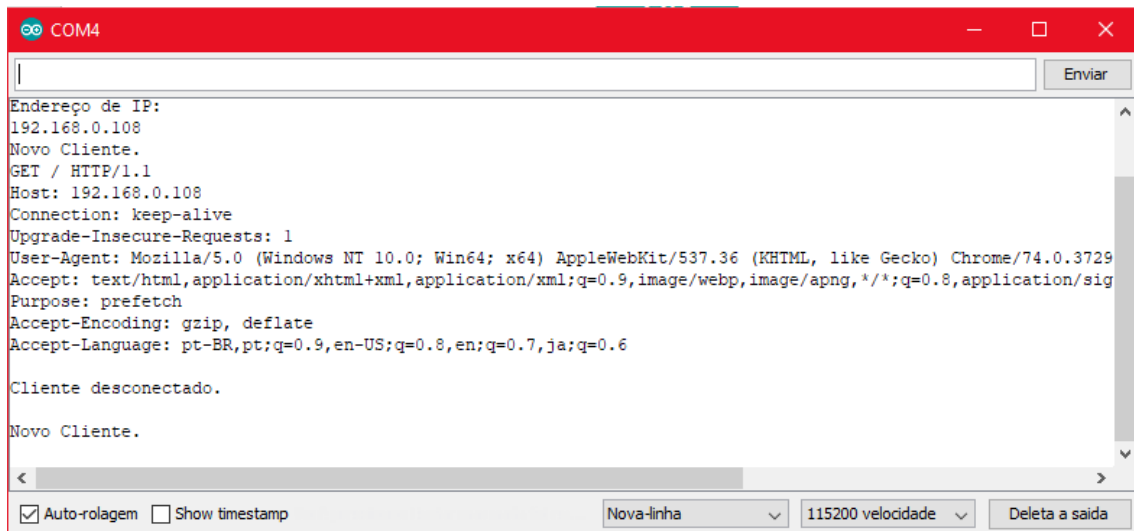
Em algum dispositivo conectado à Internet (na mesma rede), como seu *Smartphone* ou PC, abra uma janela no Navegador e cole o endereço IP do seu dispositivo. A página da Figura 4 deverá abrir, e é nela que você poderá controlar seus LEDs.

Figura 4 – Página HTML para controle dos LEDs.



Observe, também, que seu *NodeMCU* criou um servidor *web* e já possui um cliente conectado, como mostrado na Figura 5. As informações do dispositivo conectado e da conexão também são exibidas, bem como os comandos executados na página.

Figura 5 – Monitor serial após abrir a página.



Na página, cada botão representa um dos LEDs conectados. Clicando nesses botões, os LEDs deverão acender ou apagar.

Uma outra maneira de fazer uso dessa prática, de modo similar ao que foi feito até aqui, seria o *NodeMCU* enviando informações, como dados de sensores (umidade e temperatura, por exemplo), que serão vistos em uma *webpage*.

PARTE 2 – ADICIONANDO MAIS LEDS

Para a segunda parte da prática, adicione pelo menos **mais dois** LEDs na montagem do seu hardware.

DICA: Observe bem os comentários do código e tente seguir o modelo de mapeamento, tanto na declaração das **variáveis** de estado e auxiliares quanto nos **nomes** utilizados na página **HTML** e **Servidor Web**. Você pode também estilizar a página utilizando HTML ou CSS, trocando fontes, cores ou títulos.

Entregar o código-fonte e um pequeno relatório (máx. 3 págs.) via SIGAA.

LITERATURA CONSULTADA

RANDOM NERD TUTORIALS - Build an ESP8266 Web Server. Disponível em: < <https://randomnerdtutorials.com/esp8266-web-server/> >. Acesso em: 09 abr. 2024.

RANDOM NERD TUTORIALS - ESP32 Web Server – Arduino IDE. Disponível em: < <https://randomnerdtutorials.com/esp32-web-server-arduino-ide/> >. Acesso em: 10 abr. 2024.