**Enviando dados para WEB**

Em outro momento, aprendemos a programar o termistor e o ldr e monitoramos os dados enviados, pelo shell da IDE Thonny. Porém, dessa vez vamos realizar esse monitoramento pela WEB.

Você vai precisar de...

* ESP32;
* Resistor de 10KΩ;
* 1 Termistor NTC 10KΩ;
* 1 Protoboard;
* Jumpers;

Com esses materiais em mãos, realize a montagem a seguir...

Imagem

**Página HTML**

Na página HTML, vamos novamente usar o jQuery, Bootstrap e AJAX. Escreva o código abaixo no VisualStudio Code:

Texto

Descrição gerada automaticamente

Em 1 definimos o intervalo de 60.000 milissegundos (1 minuto) para atualização dos dados da página. E em 2, utilizaremos um arquivo em formato PNG, que representa a imagem de um termômetro.

Utilizando o utilitário a Ampy, escreva as linhas abaixo para copiar a página HTML e a o arquivo PNG para o sistema de arquivos interno do NodeMCU. Fique atento a página utilizada:





Quando abrir a página no seu navegador, a seguinte tela deve ser exibida:

FIGURA

**Código**

Na IDE Thonny, escreva o código abaixo, altere os campos ap e senha, a porta de acordo com sua montagem e execute-o:

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente



**Explicação**

Para iniciar, importaremos o módulo network, que nos possibilita a conexão com a internet:



Importaremos usocket com o nome de socket:



Chamaremos as funções ADC e Pin do módulo machine:



Também, precisaremos da biblioteca math para a função de conversão de temperatura:



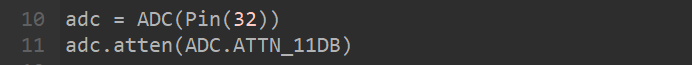
O último módulo importado será o gc, usado para gerenciamento de memória:



Chamaremos a função gc.collect(), que vai coletar e realocar as informações:



Agora, configuraremos o pino 32 como ADC e ajustar a atenuação, esse objeto receberá o nome de adc:



Após. Criaremos a função para conversão de resistência em temperatura:

Texto

Descrição gerada automaticamente

E, a função para implementar o código-fonte do servidor HTML, vamos criar a função obter\_arquivo, que obtém e retorna o conteúdo existente no sistema de Arquivos do NodeMCU:

Forma

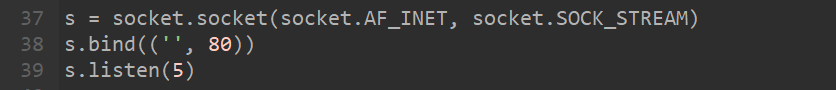
Descrição gerada automaticamente com confiança baixa

Por conseguinte, realizaremos a conexão WiFi:

Texto

Descrição gerada automaticamente

Também, criaremos o soquete:



Iniciaremos a variável html:



A seguir, criaremos um laço que permiti pedidos de conexão, processa a requisição efetuada pelo cliente e envia o conteúdo:

Texto

Descrição gerada automaticamente

Em 1 recebemos a requisição do cliente(navegador). Já em 2, o valor do termistor e, em 3 enviamos para o cliente, para que a temperatura seja atualizada na janela do navegador.

Por fim, a conexão é encerrada. Entretanto, a estação continua ativa aguardando novas conexões:

