

## Atividade AV2 – Redes

### Questão 01: Servidor Web

**Validação 1:** O código Python foi completado com o tratamento correto do socket, requisições, 200 OK, 404 Not Found.

#### Evidência:

```
# Importa o módulo socket

from socket import *

import sys # Necessário para encerrar o programa


serverPort = 6789


# Cria o socket TCP (orientado à conexão)
serverSocket = socket(AF_INET, SOCK_STREAM)


# Configuração do Socket para escuta
serverSocket.bind(('', serverPort))
serverSocket.listen(1)


while True:

    # Estabelece a conexão
    print('Ready to serve...')

    # Aguarda e aceita uma nova conexão
    connectionSocket, addr = serverSocket.accept()


    try:

        # Recebe a mensagem do cliente (requisição HTTP)
        message = connectionSocket.recv(1024).decode()

        # Analisa a requisição para obter o nome do arquivo
        filename = message.split()[1]


        # Abre o arquivo local
```

```

f = open(filename[1:], encoding='utf-8')
outputdata = f.read()

# Envia a linha de status HTTP 200 OK
connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())

# Envia o cabeçalho Content-Type
connectionSocket.send("Content-Type: text/html; charset=utf-8\r\n".encode())
connectionSocket.send("\r\n".encode())

# Envia o corpo do arquivo
connectionSocket.send(outputdata.encode('utf-8'))

connectionSocket.send("\r\n".encode())

# Fecha a conexão após o envio bem-sucedido
connectionSocket.close()

except IOError:

    # Tratamento de erro: Envia a linha de status 404 Not Found
    connectionSocket.send("HTTP/1.1 404 Not Found\r\n\r\n".encode())

    # Envia o corpo da mensagem de erro
    connectionSocket.send("<html><head></head><body><h1>404 Not
Found</h1></body></html>\r\n".encode())

    # Fecha a conexão após o envio do erro
    connectionSocket.close()

serverSocket.close()

sys.exit()

```

## HTML/CSS:

```
<html>

<head>

<title>Teste do Servidor</title>

<style>

body{

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    background-color: #f4f4f4;

    color: #333;

    margin: 0;

    display: flex;

    justify-content: center;

    align-items: center;

    min-height: 100vh;

    flex-direction: column;

    text-align: center;

}

h1{

    color: #2c3e50;

    font-size: 3em;

    margin-bottom: 10px;

    text-shadow: 2px 2px 4px rgba(0,0,0,0.1);

}

p{

    font-size: 1.2em;

    color: #555;

}

.container{

    background-color: #ffffff;

    padding: 40px 60px;
```

```
border-radius: 12px;

box-shadow: 0 10px 25px rgba(0, 0, 0, 0.1);

max-width: 600px;

width: 90%;

}

</style>

</head>

<body>

  <div class="container">

    <h1>Hello, World!</h1>

    <p>Meu servidor web está funcionando!</p>

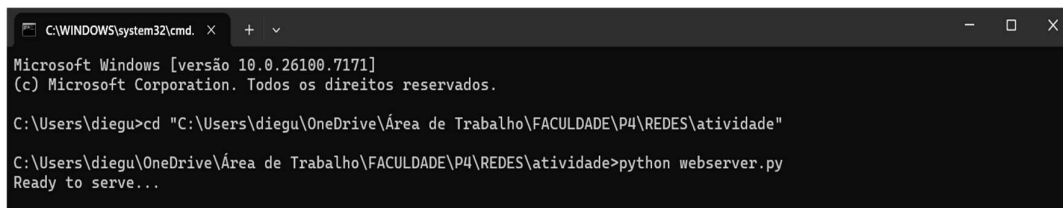
  </div>

</body>

</html>
```

**Validação 2:** O servidor foi executado com sucesso usando o comando python webserver.py e permaneceu no modo de escuta (mensagem Ready to serve...).

**Evidência:**



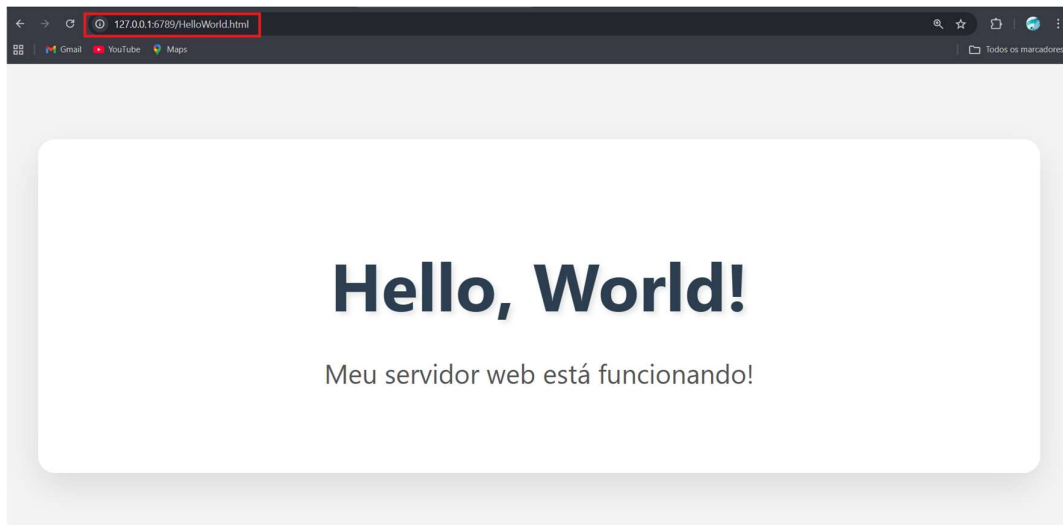
```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [versão 10.0.26100.7171]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\diegu>cd "C:\Users\diegu\OneDrive\Área de Trabalho\FACULDADE\P4\REDES\atividade"

C:\Users\diegu\OneDrive\Área de Trabalho\FACULDADE\P4\REDES\atividade>python webserver.py
Ready to serve...
```

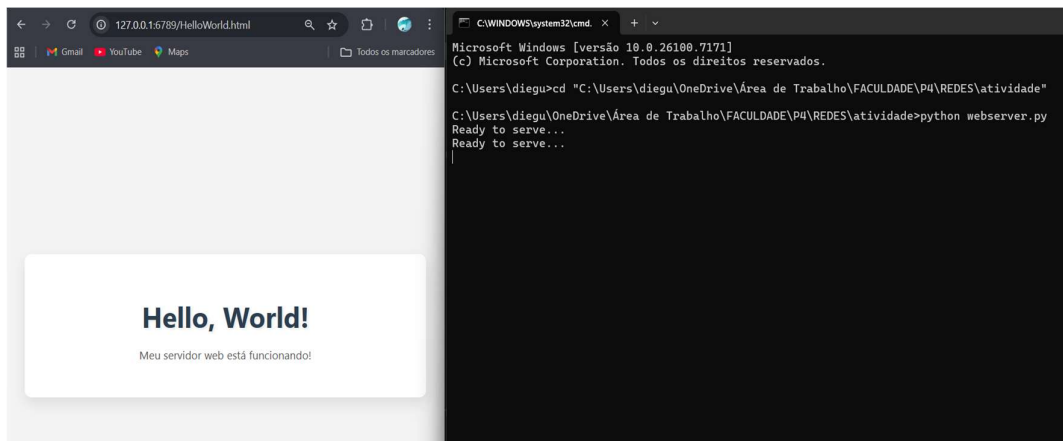
**Validação 3:** O servidor foi configurado para escutar na porta **6789**, conforme exigido pela atividade para evitar conflitos com outros servidores.

**Evidência:**



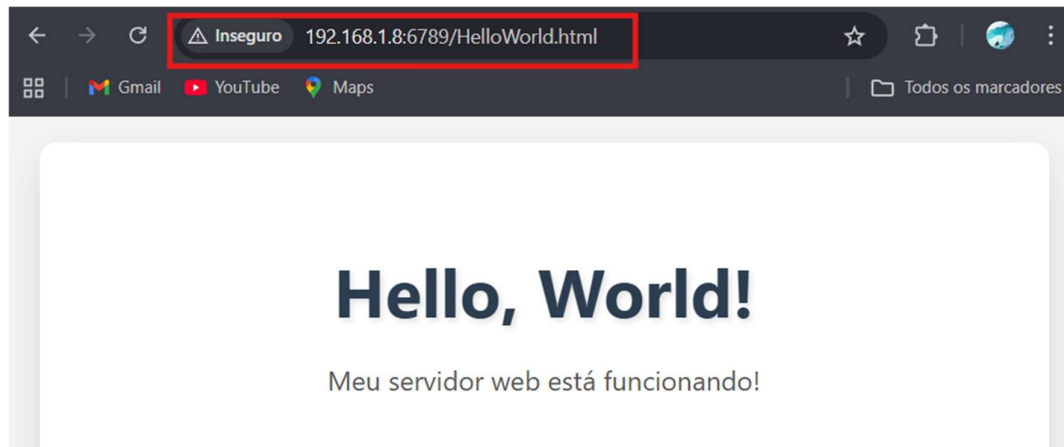
**Validação 4:** Foi acessada a URL `http://127.0.0.1:6789/HelloWorld.html`, e o navegador exibiu o conteúdo da página corretamente, provando que o servidor retornou o **200 OK**.

**Evidência:**



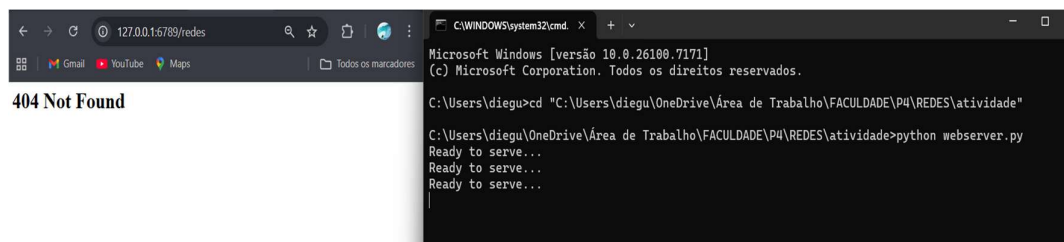
**Validação 5:** O servidor foi testado quanto ao requisito de acesso por hospedeiros diferentes.

**Evidência:**



**Validação 6:** Foi acessada uma URL inexistente, e o servidor respondeu corretamente com a mensagem **404 Not Found**, provando que o bloco except IOError está funcional.

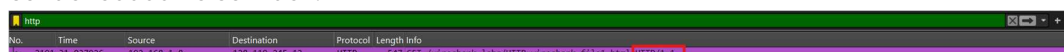
**Evidência:**



## Questão 02: Wireshark – HTTP

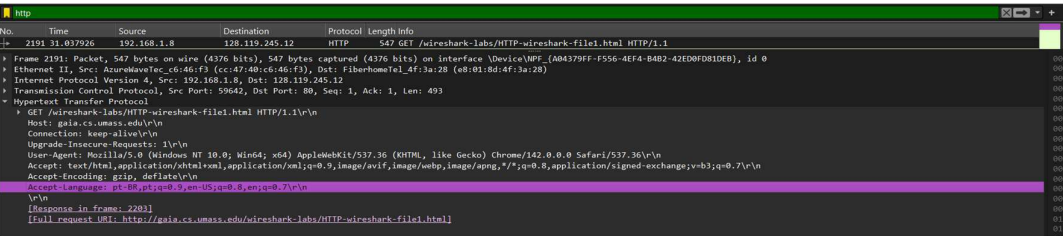
### 1. O Básico do GET do HTTP/interação de resposta

a. O seu navegador está rodando a versão 1.0 ou 1.1 do HTTP? Que versão do HTTP está sendo rodada no servidor? **R: HTTP/1.1**



b. Que linguagem (se tiver) o seu navegador indica que ele pode aceitar do servidor?

R: O navegador envia o cabeçalho Accept-Language com preferência para: **pt-BR** (seguido de pt, en-US, en).



c. Qual é o endereço IP do seu computador? E do servidor gaia.cs.umass.edu?

R: IP do meu PC é o 192.168.1.8. IP do servidor destino (gaia.cs.umass.edu) é o 128.119.245.12

Source	Destination
192.168.1.8	128.119.245.12

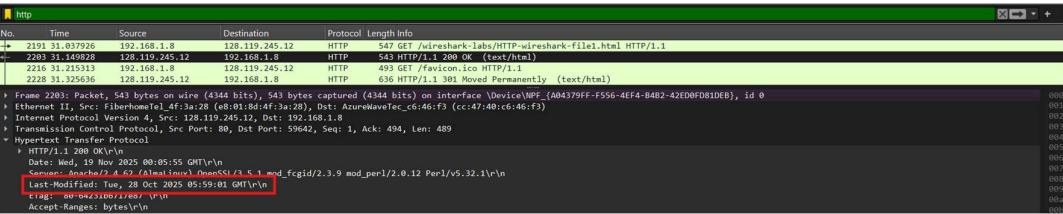
d. Qual é o código de estado retornado pelo servidor para o seu navegador?

R: 200 OK



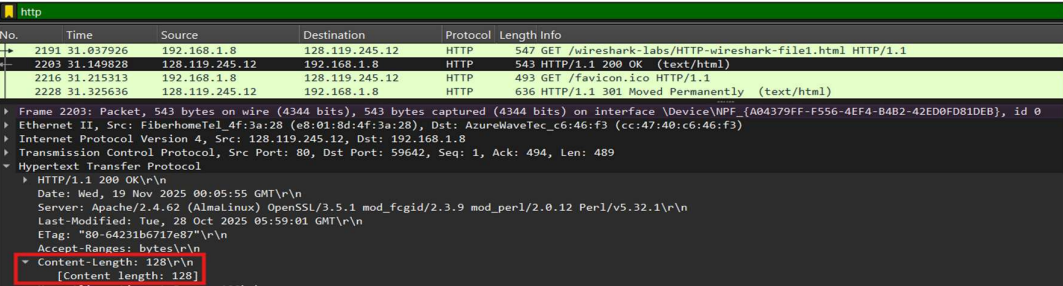
e. Quando foi a última alteração feita no arquivo HTML?

R: O cabeçalho Last-Modified indica a data da última modificação do arquivo: **Tue, 28 oct 2025 05:59:01 GMT.**



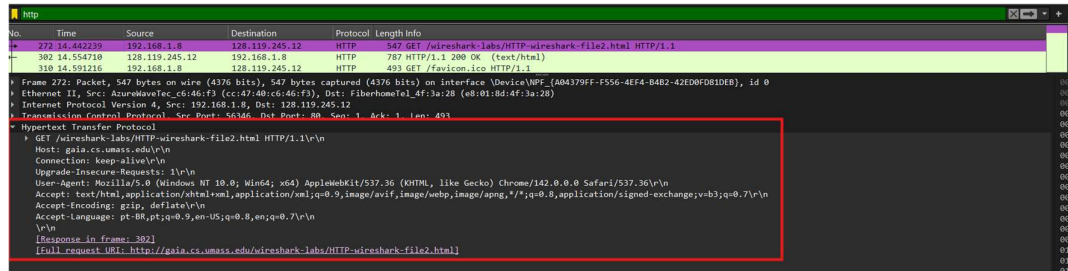
f. Quantos bytes de conteúdo são retornados para o seu navegador?

R: 128 bytes



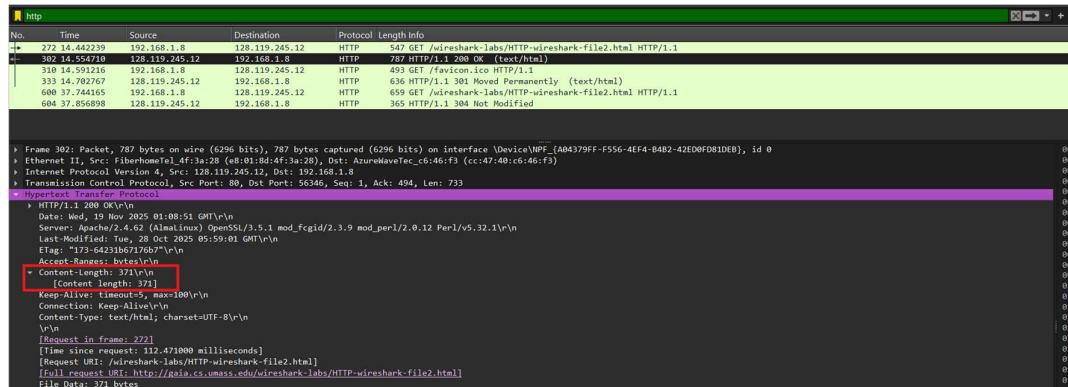
g. Inspeção o conteúdo da primeira requisição GET do HTTP de seu navegador para o servidor. Você viu uma linha com “IF-MODIFIED-SINCE” no GET do HTTP?

R: Não. Como é o primeiro acesso, o navegador não tem dados de cache e não envia este cabeçalho.



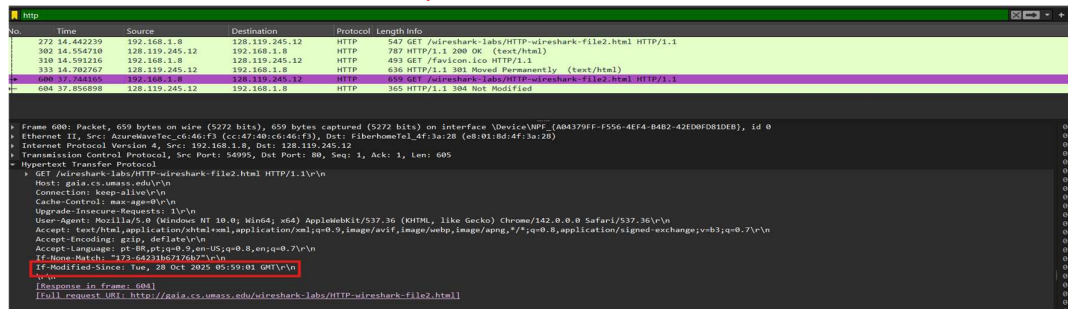
h. Inspeção o conteúdo da resposta do servidor. O servidor retornou o conteúdo do arquivo explicitamente? Como isso se observou?

R: Sim. No pacote 302, a resposta foi 200 OK e o servidor enviou o corpo do arquivo.



i. Agora investigue o conteúdo da segunda requisição GET do HTTP de seu navegador para o servidor. Você viu uma linha com “IF-MODIFIED-SINCE” no GET do HTTP? Se sim, que informação segue o cabeçalho “IF-MODIFIED-SINCE:”?

R: Sim. No pacote 600, o navegador enviou o cabeçalho condicional If-Modified-Since com a data de Last-Modified da resposta anterior.





j. Qual é o código de estado do HTTP e a frase retornada do servidor em resposta ao segundo GET do HTTP? O servidor explicitamente retornou o conteúdo do arquivo? Explique:

R: O servidor retornou **304 Not Modified** (pacote 604) e **não** retornou o corpo do arquivo. Isso indica que o arquivo não foi alterado, permitindo que o navegador utilize a cópia do **cache local** sem a necessidade de retransferência de dados.

