

**INSTITUTO FEDERAL**

Mato Grosso do Sul

## CONSTRUÇÃO DE PÁGINAS WEB I

Ermerson Moraes

Prof. EBTT – Informática/Desenvolvimento - Substituto

Email: [ermerson.moraes@ifms.edu.br](mailto:ermerson.moraes@ifms.edu.br)

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Introdução

Imaginem que a internet é um grande país. Para que as cidades (servidores) e os cidadãos (navegadores) possam conversar e trocar mercadorias (dados), eles precisam de um idioma e um sistema de correios em comum. Esse sistema é o nosso protocolo.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Introdução

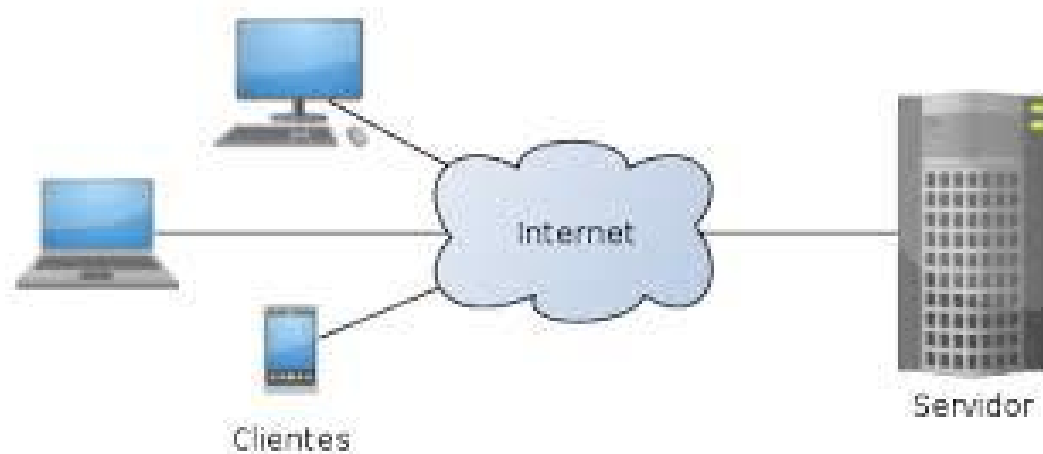
O que é o HTTP? Apresento o Hypertext Transfer Protocol como o idioma padrão da World Wide Web. É ele que define as regras de como os navegadores pedem informações e como os servidores as entregam.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Introdução

O HTTP é a base para que o cliente e um servidor web troquem informações. Ele permite a requisição e a resposta de recursos, como imagens, arquivos e as próprias páginas webs que acessamos, por meio de mensagens padronizadas.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Qual é a origem do HTTP?

Para falar da origem do HTTP, vamos lá para a década de 1980, quando o britânico Tim Berners-Lee, cientista da computação, estava trabalhando na Organização Europeia para Pesquisa Nuclear (CERN).



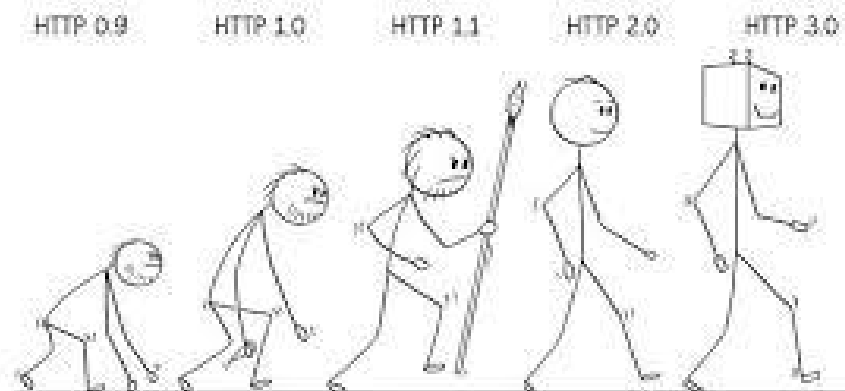
# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Qual é a origem do HTTP?

A primeira versão, a 0.9, uma versão simples que permitia apenas solicitações GET, isso é, os clientes poderiam solicitar recuperação de uma página de um servidor. A resposta não trazia cabeçalhos, apenas o conteúdo HTML solicitado.

### Evolution of HTTP protocol



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Qual é a origem do HTTP?

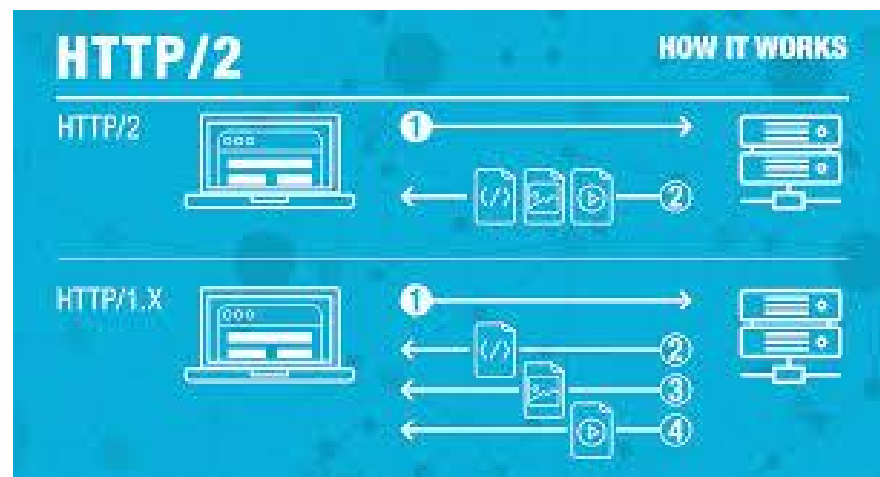
Em 1997, foi lançado o HTTP 1.1, apresentando aprimoramentos significativos. Uma das mudanças chave foi a introdução da persistência de conexão, uma funcionalidade que reduz a latência ao permitir que múltiplas solicitações e respostas sejam enviadas pela mesma conexão, evitando a necessidade de abrir uma nova conexão para cada recurso.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Qual é a origem do HTTP?

A versão 2.0 do HTTP, lançada em 2015, introduzindo a multiplexação para transmissão simultânea, a compactação de cabeçalhos em formato binário e recursos como priorização de fluxo e push do servidor, o HTTP 2 aumentou a eficiência, reduziu a latência e proporcionou uma experiência de navegação mais rápida.





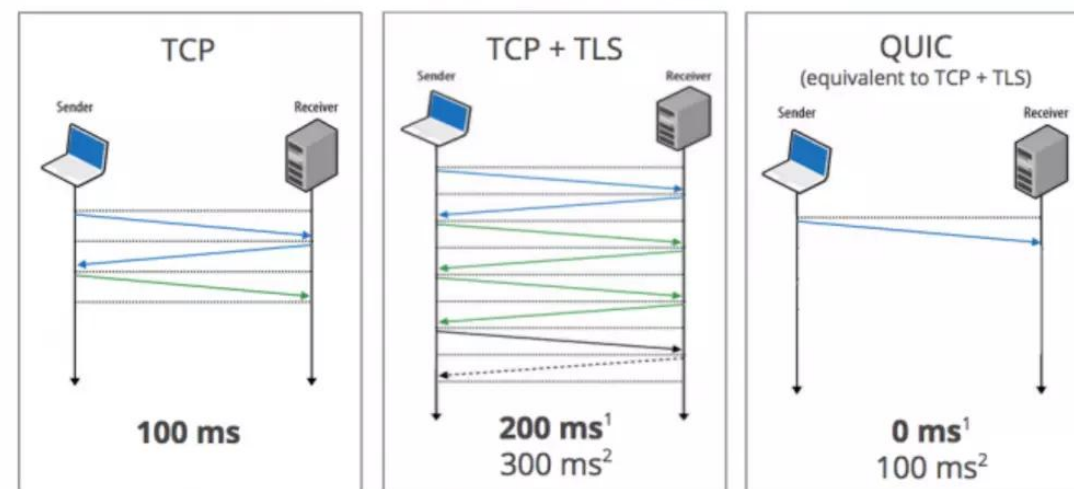
# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Qual é a origem do HTTP?

A versão atual, HTTP 3, foi lançada em 2018 em resposta à demanda por maior eficiência e desempenho, adotando o protocolo QUIC. Essa evolução, somada a características mantidas das versões anteriores, reforçou ainda mais a eficiência na entrega de conteúdo online.

### Zero RTT Connection Establishment



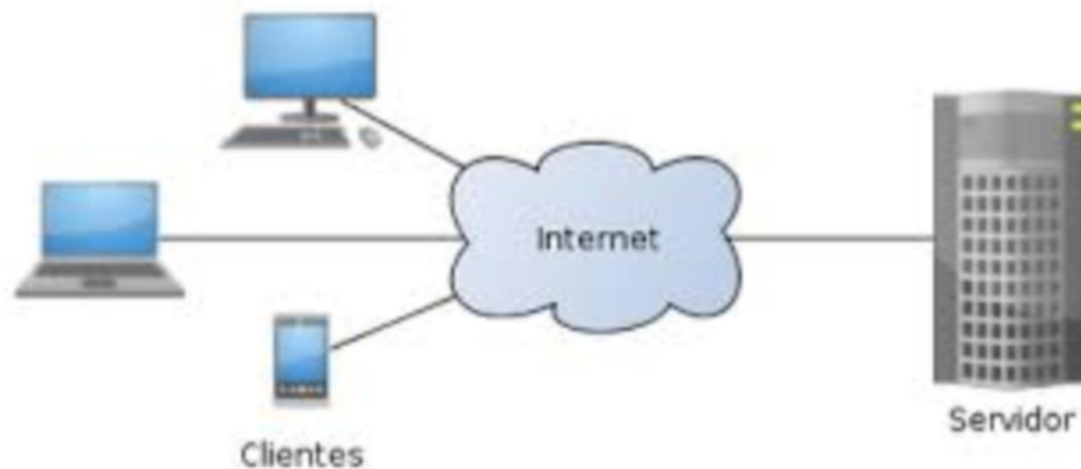
1. Repeat connection  
2. Never talked to server before

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Como funciona o HTTP?

O HTTP opera em um modelo cliente-servidor, em que um cliente, geralmente um navegador web, faz solicitações a um servidor para obter recursos, como páginas da web, imagens ou arquivos.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Como funciona o HTTP?

O ciclo se inicia quando:

O cliente estabelece contato com o servidor, encaminhando uma requisição HTTP;

Nessa solicitação, o cliente especifica o método pretendido (por exemplo, GET para obter informações) e o caminho do recurso desejado;

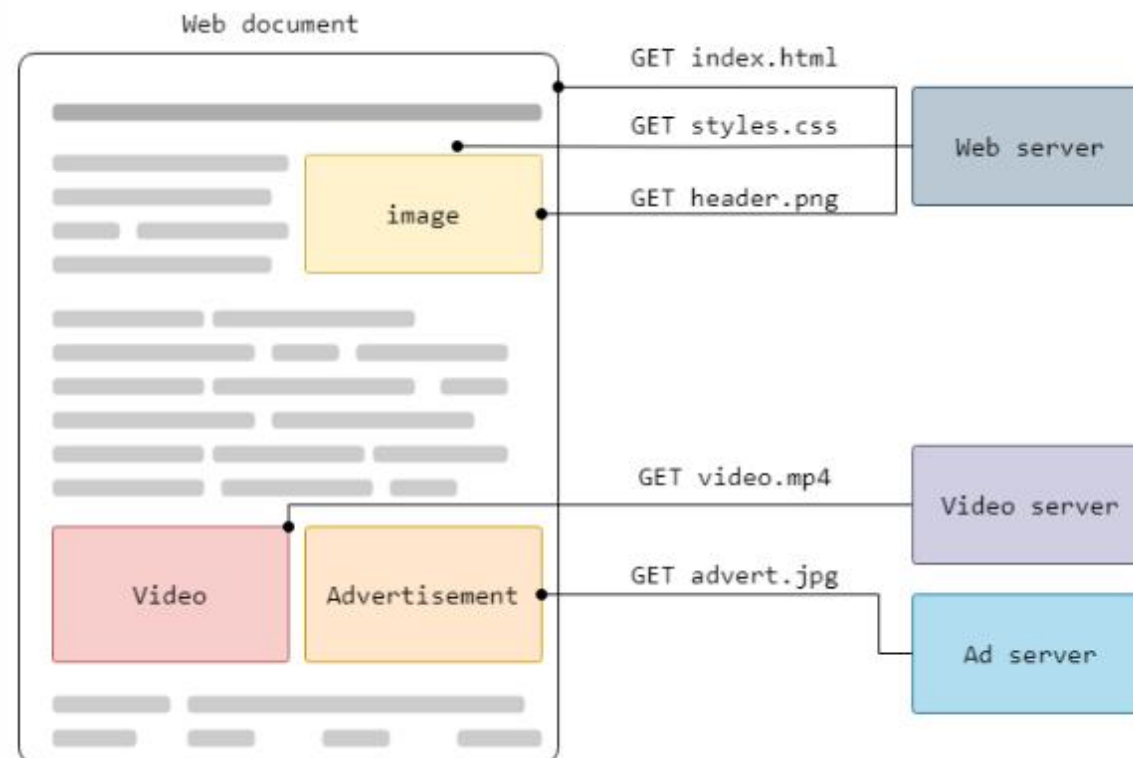
# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Como funciona o HTTP?

O ciclo se inicia quando:

Ao receber essa requisição, o servidor a processa e responde com uma mensagem HTTP, incluindo o recurso requisitado e informações adicionais no cabeçalho da resposta.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

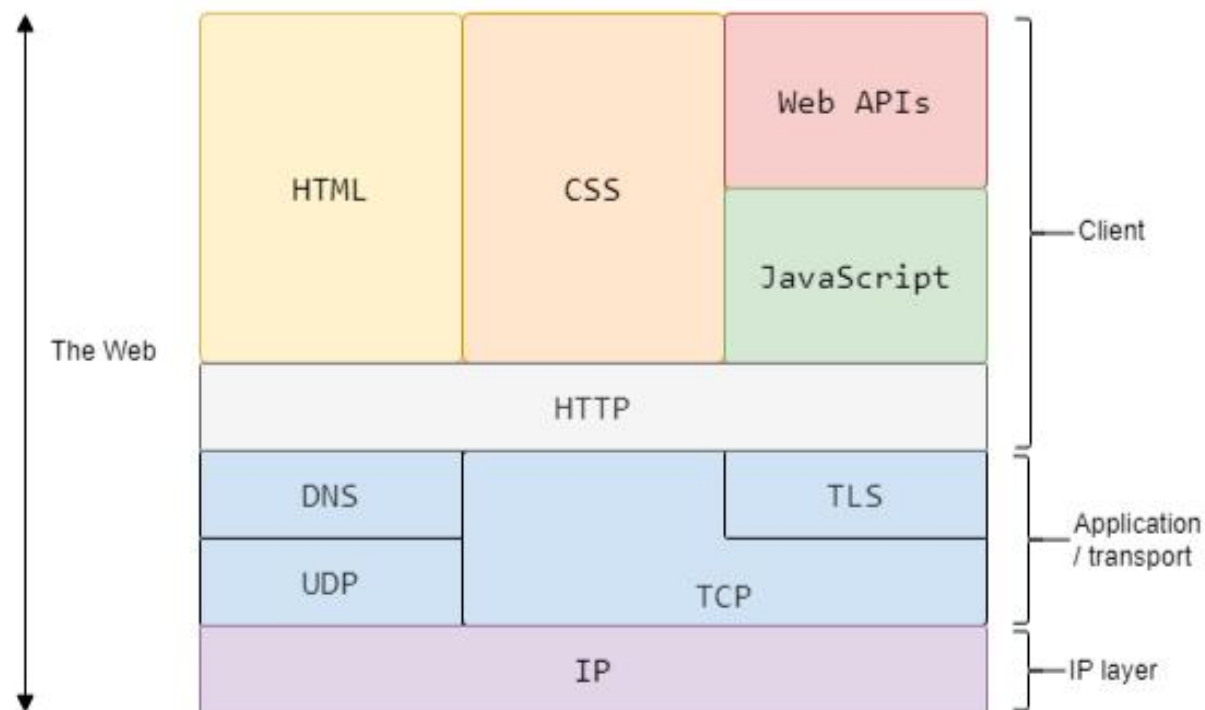
Como funciona o HTTP?

E toda essa interação acontece por meio do protocolo TCP (Transmission Control Protocol), que garante uma comunicação confiável e orientada à conexão.

Sem a necessidade de gerir o estado entre as requisições, o protocolo mantém uma certa simplicidade, promovendo também uma escalabilidade maior.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP



Como funciona o HTTP?

Contudo, para abordar a demanda de manter estados em determinadas aplicações web, são comumente empregados mecanismos suplementares, tais como cookies e sessões.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Exercícios:

1. Explique em suas próprias palavras o que é o HTTP e qual é a sua função na comunicação entre cliente (navegador) e servidor na Web. Utilize a analogia do “idioma” e “sistema de correios” apresentada na aula.
2. Crie uma tabela de evolução do HTTP, relacionando cada versão às suas principais características:

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Componentes de sistemas baseados em HTTP

O HTTP é um protocolo cliente-servidor: as requisições são enviadas por uma entidade, o agente-usuário (ou um proxy em nome dele). A maior parte do tempo, o agente-usuário é um navegador da Web.





# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Componentes de sistemas baseados em HTTP

Cada requisição individual é enviada para um servidor, que irá lidar com isso e fornecer um resultado, chamado de resposta. Entre a solicitação e a resposta existem várias entidades, os proxies, que executam operações diferentes e atuam como gateways (intermediários) ou caches, por exemplo.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Cliente: o agente-usuário

O agente-usuário é qualquer ferramenta que age em nome do usuário. Essa função é predominantemente realizada pelo navegador Web; algumas poucas exceções são programas usados por engenheiros e desenvolvedores Web para debugar as suas aplicações.

```
Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:47.0) Gecko/20100101 Firefox/47.0
```

```
Mozilla/5.0 (Macintosh; Intel Mac OS X x.y; rv:42.0) Gecko/20100101 Firefox/42.0
```

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Cliente: o agente-usuário

Para mostrar uma página Web, o navegador envia uma requisição para buscar o documento HTML da página.

Depois o navegador interpreta esses recursos para mostrar ao usuário a página completa. Existem scripts executados pelo navegador que buscam mais recursos em fases subsequentes e conforme o uso da página e o navegador atualiza a página de acordo.

```
.hljs-number {color: #1c00cf;
}.hljs-section,
.hljs-meta {color: #643820;
}.hljs-title.class_,
.hljs-class .hljs-title,
.hljs-type,
.hljs-built_in,
.hljs-params {color: #5c2699;
}.hljs-attr {color: #836c28;
}.hljs-subst {color: #000;
}.hljs-formula {background-color: #eee;font-style: italic;
}.hljs-addition {background-color: #baeeba;
}.hljs-deletion {background-color: #ffc8bd;
}.hljs-selector-id,
.hljs-selector-class {color: #9b703f;
}.hljs-doctag,
.hljs-strong {font-weight: bold;
}.hljs-emphasis {font-style: italic;
}
</style><style type="text/css" data-fela-rehydration="482" data-
<script src="https://cdn-client.medium.com/lite/static/js/report
<script src="https://cdn-client.medium.com/lite/static/js/5049.c
<script src="https://cdn-client.medium.com/lite/static/js/4810.6
<script src="https://cdn-client.medium.com/lite/static/js/6618.c
<script src="https://cdn-client.medium.com/lite/static/js/1386.6
<script src="https://cdn-client.medium.com/lite/static/js/9977.c
<script src="https://cdn-client.medium.com/lite/static/js/5250.9
<script src="https://cdn-client.medium.com/lite/static/js/8261.9
<script src="https://cdn-client.medium.com/lite/static/js/7975.9
<script src="https://cdn-client.medium.com/lite/static/js/2648.4
<script src="https://cdn-client.medium.com/lite/static/js/2712.c
<script src="https://cdn-client.medium.com/lite/static/js/2793.2
```

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### O servidor de páginas Web

Do outro lado do canal de comunicação está o servidor que serve o documento requisitado pelo usuário. Um servidor se apresenta virtualmente apenas como uma máquina: isto porque o servidor pode ser uma coleção de servidores dividindo a carga (através de uma técnica chamada balanceamento de carga) ou também como um programa complexo que acessa outros servidores (como um cache, um servidor de banco de dados, servidores de e-commerce (lojas virtuais), etc.), gerando toda ou parte do documento solicitado.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Proxies (ou representantes)

Entre o navegador Web e o servidor, vários computadores e máquinas transmitem as mensagens HTTP. Essas máquinas que operam na camada de aplicação são normalmente conhecidas como proxies (ou representantes, ou procuradores, etc). Eles podem ser transparentes ou não (alterações nas requisições não passam por eles), e podem desempenhar várias funções:



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Proxies (ou representantes)

- cacheamento
- filtragem
- balanceamento de carga
- autenticação
- autorização
- registro de informação

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

HTTP é simples

Mesmo com mais complexidade introduzida no HTTP/2.0 por encapsular mensagens HTTP em quadros (frames), o HTTP foi projetado para ser simples e legível às pessoas. As mensagens HTTP podem ser lidas e entendidas por qualquer um, provendo uma maior facilidade para desenvolvimento e testes, e reduzir a complexidade para os estudantes.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

HTTP é extensível

Introduzidos no HTTP/1.0, os cabeçalhos HTTP fazem com que este protocolo seja fácil para estender e usá-lo para experimentos. Novas funcionalidades podem até ser introduzidas pelo simples acordo entre um cliente e um servidor sobre a nova semântica de um cabeçalho.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Aspectos básicos do HTTP

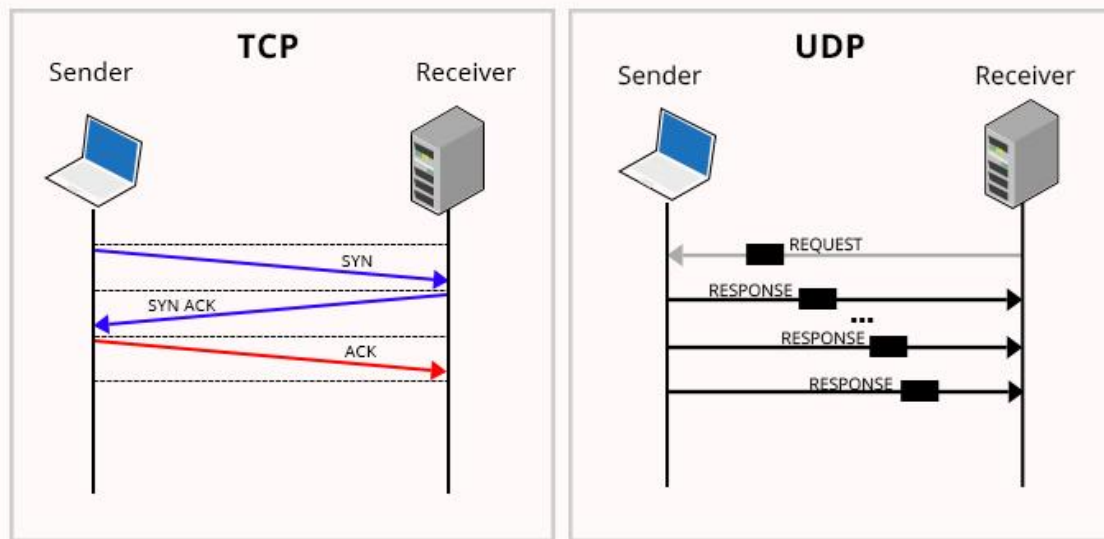
HTTP não tem estado, mas tem sessões

HTTP é sem estado: não existe uma relação entre duas requisições sendo feitas através da mesma conexão. Mas como o fundamento básico do HTTP é não manter estados, cookies HTTP permitem que as sessões tenham estados. Usando a extensibilidade dos cabeçalhos, os cookies são adicionados ao fluxo do HTTP, permitindo que a criação de sessão em cada requisição HTTP compartilhem o mesmo contexto, ou o mesmo estado.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### TCP Vs UDP Communication



Aspectos básicos do HTTP

HTTP e conexões

O HTTP não requer que o protocolo de transporte utilizado seja baseado em conexões, só requer que seja confiável ou não perca mensagens (sem pelo menos apresentar erros). Dentre os dois protocolos de transporte mais comuns na internet, o TCP é confiável e o UDP não. Portanto, o HTTP utiliza o padrão TCP, que é baseado em conexão, mesmo que nem sempre seja obrigatório o uso de uma conexão.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Fluxo HTTP

Quando o cliente quer comunicar com um servidor, este sendo um servidor final ou um proxy, ele realiza os seguintes passos:

1-Abre uma conexão TCP: A conexão TCP será usada para enviar uma requisição, ou várias, e receber uma resposta. O cliente pode abrir uma nova conexão, reusar uma conexão existente, ou abrir várias conexões aos servidores.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Fluxo HTTP

2-Envia uma mensagem HTTP. Com o HTTP/2.0, essas mensagens simples são encapsuladas dentro de quadros (frames), tornando-as impossíveis de ler diretamente, mas o princípio se mantém o mesmo.

HTTP

```
GET / HTTP/1.1
```

```
Host: developer.mozilla.org
```

```
Accept-Language: fr
```

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Fluxo HTTP

3-Lê a resposta do servidor

4-Fecha ou reutiliza a conexão para requisições futuras.

### HTTP

HTTP/1.1 200 OK

Date: Sat, 09 Oct 2010 14:28:02 GMT

Server: Apache

Last-Modified: Tue, 01 Dec 2009 20:18:22 GMT

ETag: "51142bc1-7449-479b075b2891b"

Accept-Ranges: bytes

Content-Length: 29769

Content-Type: text/html

<!DOCTYPE html... (here comes the 29769 bytes of the requested web page)

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

HTTP/1.1 e mensagens mais antigas HTTP são legíveis às pessoas. No HTTP/2.0, essas mensagens são embutidas numa nova estrutura binária, um quadro, permitindo otimizações como compressão de cabeçalhos e multiplexação. É portanto útil entender as mensagens HTTP/2.0 no formato da versão HTTP/1.1.

# CONSTRUÇÃO DE PÁGINAS WEB I

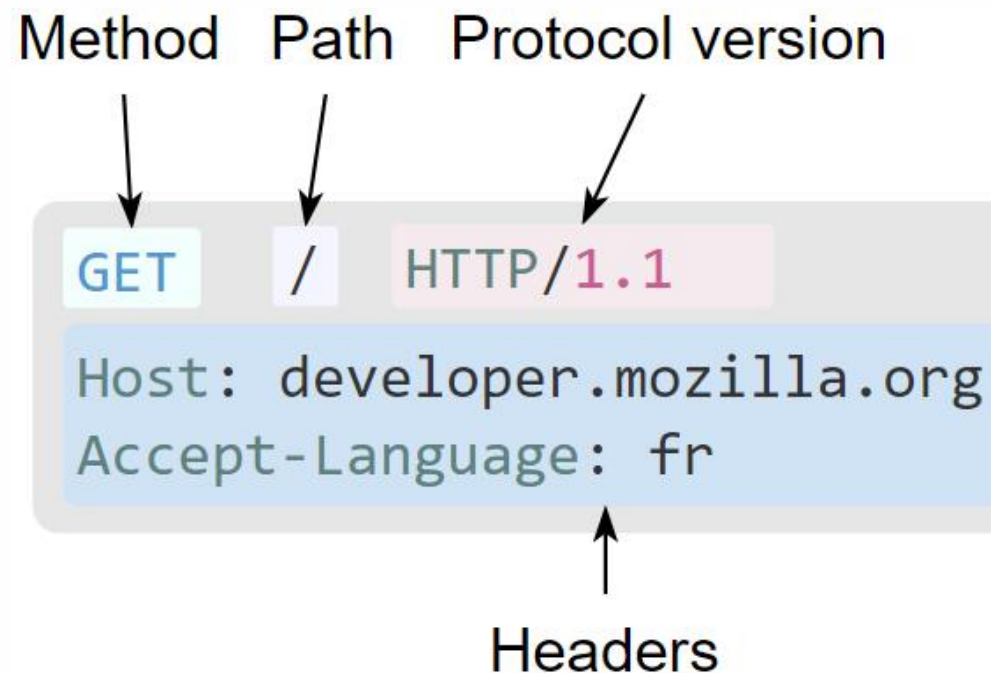
## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

Requisições

Exemplo de uma requisição HTTP:



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

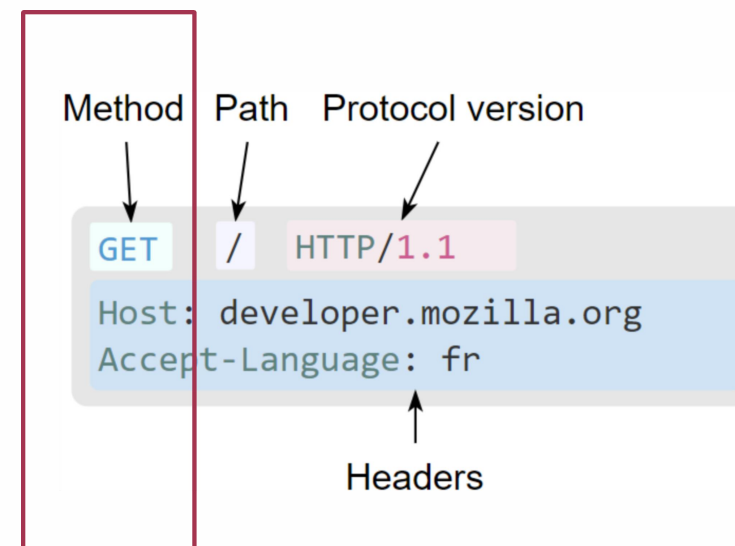
Aspectos básicos do HTTP

Mensagens HTTP

Requisições

As requisições consistem dos seguintes elementos:

Um método HTTP, geralmente é um verbo como GET, POST, DELETE, PUT, etc, ou um substantivo como OPTIONS ou HEAD que define qual operação o cliente quer fazer.





# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

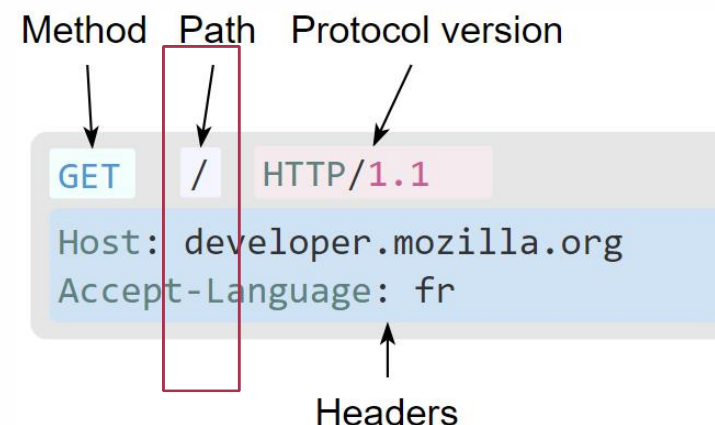
Aspectos básicos do HTTP

Mensagens HTTP

Requisições

As requisições consistem dos seguintes elementos:

O caminho do recurso a ser buscado; a URL do recurso sem os elementos que são de contexto, o domínio domain ou a porta port TCP (aqui indicada pelo 80 que é ocultado por ser o número da porta padrão)



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

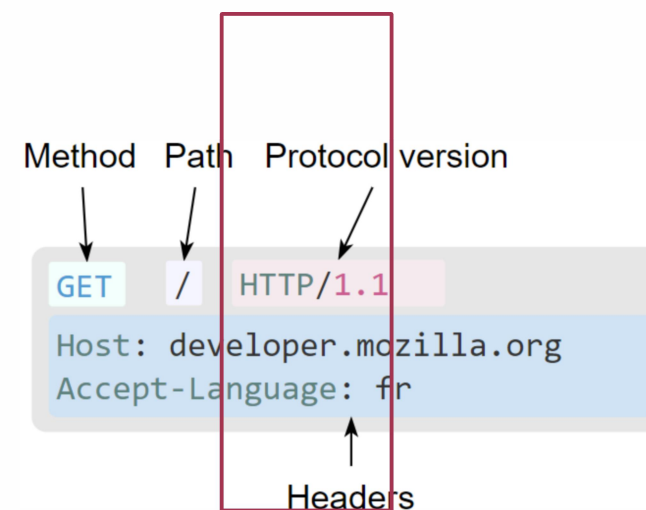
Aspectos básicos do HTTP

Mensagens HTTP

Requisições

As requisições consistem dos seguintes elementos:

A versão do protocolo HTTP.



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

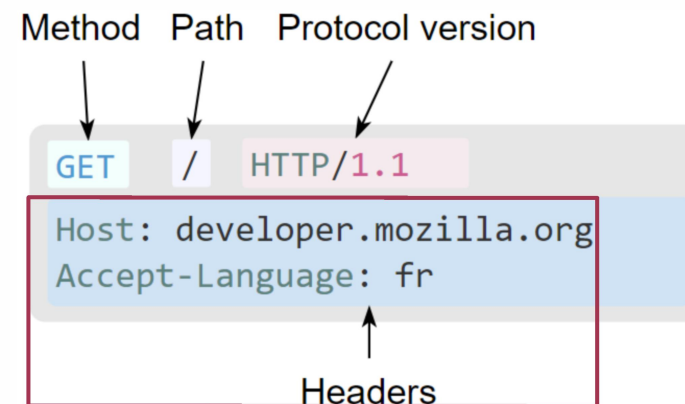
Mensagens HTTP

Requisições

As requisições consistem dos seguintes elementos:

Cabeçalhos opcionais que contém informações adicionais para os servidores.

Ou um corpo de dados, para alguns métodos como POST, similares aos corpos das respostas, que contém o recurso requisitado.



# CONSTRUÇÃO DE PÁGINAS WEB I

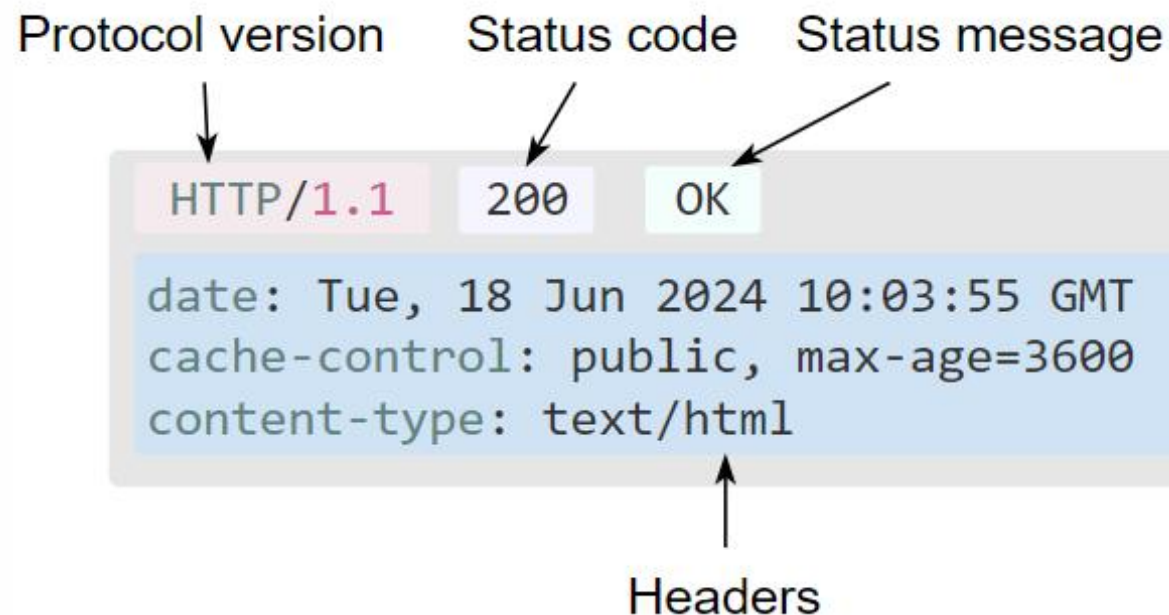
## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

Respostas

Exemplo de resposta HTTP:



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

Códigos de status de respostas HTTP

Os códigos de status de resposta HTTP indicam se uma solicitação HTTP específica foi concluída com êxito.

As respostas são agrupadas em cinco classes:

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

Códigos de status de respostas HTTP

Respostas Informativas (100 – 199)

Respostas bem-sucedidas (200 – 299)

Mensagens de redirecionamento (300 – 399)

Respostas de erro do cliente (400 – 499)

Respostas de erro do servidor (500 – 599)

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

Códigos de status de respostas HTTP

Respostas informativas - 100

100 Continue - Essa resposta provisória indica que o cliente deve continuar a solicitação ou ignorar a resposta se a solicitação já estiver concluída.

101 Switching Protocols - Esse código é enviado em resposta a um cabeçalho de solicitação Upgrade do cliente e indica o protocolo para o qual o servidor está mudando.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens HTTP

Códigos de status de respostas HTTP

Respostas bem sucedidas - 200

200 OK - A solicitação foi bem-sucedida. O significado do resultado de "sucesso" depende do método HTTP:



# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

GET: O recurso foi obtido e transmitido no corpo da mensagem.

HEAD: Os cabeçalhos de representação são incluídos na resposta sem nenhum corpo de mensagem.

PUT ou POST: O recurso que descreve o resultado da ação é transmitido no corpo da mensagem.

TRACE: O corpo da mensagem contém a mensagem de requisição recebida pelo servidor.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Aspectos básicos do HTTP

201 Created - A requisição foi bem sucedida e um novo recurso foi criado como resultado. Esta é normalmente a resposta enviada após as solicitações POST ou algumas solicitações PUT.

202 Accepted - A solicitação foi recebida, mas ainda não foi atendida. Destina-se a casos em que outro processo ou servidor manipula a solicitação ou processamento em lote.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Mensagens de redirecionamento - 300

300 Multiple Choices - A solicitação tem mais de uma resposta possível.

301 Moved Permanently- A URL do recurso solicitado foi alterada permanentemente. A nova URL é fornecida na resposta.

302 Found - Este código de resposta significa que o URI do recurso solicitado foi alterado temporariamente.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Respostas de erro do cliente - 400

400 Bad Request - A requisição foi malformada ou contém sintaxe inválida, impedindo o servidor de processá-la.

401 Unauthorized - O cliente precisa autenticar-se para obter a resposta solicitada. Usado em conjunto com autenticação básica ou por tokens.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

### Aspectos básicos do HTTP

#### Respostas de erro do cliente - 400

403 Forbidden - O servidor entendeu a requisição, mas se recusa a autorizá-la. Isso pode ocorrer devido a permissões insuficientes.

404 Not Found - O recurso solicitado não foi encontrado no servidor. Esse é o código mais conhecido e é retornado quando uma URL é inválida ou o recurso foi removido.

405 Method Not Allowed - O método HTTP utilizado não é permitido para o recurso solicitado. Por exemplo, um servidor pode permitir GET mas não POST em um recurso específico.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Respostas de erro do servidor - 500

500 Internal Server Error - O servidor encontrou uma situação com a qual não sabe lidar.

501 Not Implemented - O método da requisição não é suportado pelo servidor e não pode ser manipulado.

# CONSTRUÇÃO DE PÁGINAS WEB I

## Protocolo HTTP

Aspectos básicos do HTTP

Respostas de erro do servidor - 500

502 Bad Gateway - O servidor, atuando como gateway ou proxy, recebeu uma resposta inválida do servidor upstream (servidor original ou outro serviço intermediário).

503 Service Unavailable - O servidor não está disponível para processar a requisição, geralmente devido a manutenção ou sobrecarga. É um erro temporário.