

HTTP - HYPERTEXT TRANSFER PROTOCOL

1 - Principal protocolo na comunicação de computadores através da internet. 2 - Protocolo da camada de aplicação, sempre trabalhando junto a outros dois protocolos, o TCP (responsável pela transmissão das informações) e o IP (cuida do encaminhamento dos dados) 3 - Esse protocolo se baseia em pedido e resposta (Request e Response), após o pedido e a resposta do servidor a conexão estabelecida é fechada, sendo assim, deverá ser necessário repetir o ciclo toda vez que uma request é feita. 4 - Request: Composto por: Linha de pedido, cabeçalhos e corpo/mensagem.

1. **Linha de pedido:** contém o Identificador do método (tipo de ação esperada do servidor), URI do recurso: Endereço no qual será enviado a request e a Versão do protocolo: Atualmente a mais usada (1.1)
2. **Cabeçalho:** Podem ser enviados dezenas de campos para o servidor através do cabeçalho e ele pode responder de alguma maneira diferente de acordo com o que foi enviado. Os campos mais comuns são:
 1. **Date:** Informa a data do envio da requisição.
 2. **Cache-Control:** Envia diretivas para o mecanismo de Cache.
 3. **Transfer-Encoding:** Especifica a forma de decodificar o corpo da requisição.
 4. **Cookie:** Envia informações sobre os cookies.
 5. **Accept:** Especifica o tipo de resposta.
 6. **User-Agent:** Envia informações sobre o client (Navegador)
 7. **Body:** Exemplo, o envio de um formulário.

Exemplo de requisição completa:

```
POST / HTTP/1.1

Host: www.codigofonte.com.br/
User-Agent: Mozilla/5.0 (X11; Linux i686) AppleWebKit/537.36 (KHTML, like Gecko) Ubuntu Chromium/76.0.3809.87 Chrome/76.0.3809.87 Safari/537.36
Accept: text/html,application/xml;q=0.9,*/*;q=0.8
Accept-Language: pt-br
Accept-Encoding: gzip,deflate
Accept-Charset: utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

nome1=vanessa&nome2=gabriel
```



Está definido em 3 partes:

- Cabeçalho Geral
- Cabeçalho de Requisição
- Cabeçalho de Entidade

Response: Está definido em 3 partes também:

- **Linha de status:** Versão do protocolo usada no servidor, código numérico do status (Como nosso pedido foi condicionado no servidor) e Texto associado ao

status. Um dos mais importantes é o código do status, pois através dele você consegue visualizar como seu pedido foi processado no servidor, Categorias:

1. 1xx Informational (pedido recebido e ainda está sendo processado)
2. 2xx Success (pedido recebido => aceito => processado)
3. 3xx Redirection (ações adicionais precisam ser feitas para completar seu pedido)
4. 4xx Client Error (informações incorretas e ou não pode ser processado)
5. 5xx Server Error (servidor não conseguiu processar seu pedido)

- **Cabeçalho:** Exemplos de campos:

1. Content-Type: Tipo de conteúdo enviado na requisição
2. Access-Control-Allow: Se a resposta pode ser acessada pela origem do seguinte (CORS)
3. Date: Data da resposta enviada

- **Corpo:** Mensagem que o nosso navegador vai interpretar (html, xml, json)

Exemplo de resposta completa vindo do servidor: ![Dicionario do Programador - HTTP] (<https://i.imgur.com/gc8DdJa.png> =600x450)

Ciclo HTTP finalizado.

Todo o conteúdo foi retirado do canal: Código Fonte TV

(<https://www.youtube.com/c/codigofontetv>) Agradeço imensamente por todo conteúdo que eles proporcionam para nós devs.