



Protocolo HTTP - Requisições

Autor : Marcelo Passos dos Santos





Objetivo

Apresentar as requisições no navegador.



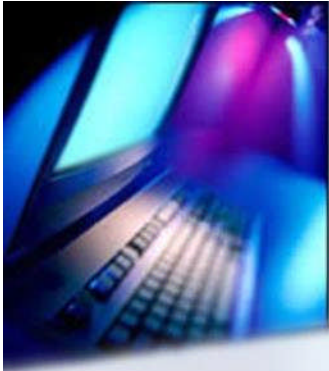
Protocolo HTTP

- Vamos fazer um teste e usar o navegador para mostrar mais detalhes sobre a comunicação HTTP.
- Os navegadores mais populares como Google Chrome, Mozilla Firefox ou Microsoft Internet Explorer possuem ferramentas e plugins que visualizam como o navegador trabalha e usa o HTTP.
- Para habilitar as ferramentas do desenvolvedor no Chrome vá ao menu: Visualizar -> Desenvolvedor Ferramentas do desenvolvedor (Selecione a aba Network)
- No Firefox vá ao menu: Ferramentas -> Desenvolvedor web -> Exibir/Ocultar ferramentas
- Para o Internet Explorer aperte a tecla F12 para abrir o console do desenvolvedor e selecione aba Rede (ou Network).



Método Get no HTTP

- Vamos abrir o console de desenvolvedor e acessar o `http://www.uol.com.br`.
- Aqui usaremos o navegador Chrome, mas nos outros navegadores o comportamento é bem parecido.
- No console podemos ver todas as requisições HTTP executadas pelo Chrome.
- Mas não só isso, também aparecem alguns códigos e métodos, além do tempo de execução para cada requisição.
- Repare que chamamos apenas o `http://www.uol.com.br`, mas foram feitas várias outras requisições em seguidas.



Exercícios

- Qualquer resposta HTTP possui um número que informa sobre o status da requisição.
- Qual dos códigos abaixo indica que a requisição foi bem sucedida?

() 100

() 404

() 200

() 300



Exercícios

- Na documentação oficial se diz a respeito da classe de códigos que começam com 2xx:
- **2xx** – Sucesso. Esta classe de códigos de status indica a ação solicitada pelo cliente foi recebida, compreendida, aceita e processada com êxito.
- **200**
 - Padrão de resposta para solicitações HTTP sucesso.
- **201 Criado**
 - O pedido foi cumprido e resultou em um novo recurso que está sendo criado.
- **202 Aceito**
 - O pedido foi aceito para processamento, mas o tratamento não foi concluído. O pedido poderá ou não vir a ser posta em prática, pois pode ser anulado quando o processamento ocorre realmente.



Exercícios

- **Problema no servidor**
- Quais desses códigos representa algum problema gerado no servidor?

() 500

() 302

() 402

() 404

() 302



Exercícios

- **500 Erro interno do servidor (Internal Server Error)**
 - Indica um erro do servidor ao processar a solicitação. Na grande maioria dos casos está relacionada as permissões dos arquivos ou pastas do software ou script que o usuário tenta acessar e não foram configuradas no momento da programação/construção do site ou da aplicação
- **501 Não implementado**
 - O servidor ainda não suporta a funcionalidade ativada.
- **502 Bad Gateway**
 - Em regra, o erro quando há uma configuração imprecisa entre os computadores de rede, possivelmente incluindo o servidor Web no site visitado. Antes de analisar este problema, é necessário limpar o cache do navegador, completamente.
- **503 Serviço indisponível**
 - O servidor está em manutenção ou não consegue dar conta dos processamentos de recursos devido à sobrecarga do sistema. Isto deve ser uma condição temporária
- **504 Gateway Time-Out**
 - É caracterizado por erros particulares do site em questão. Pode ser que o site esteja em manutenção ou não exista.
- **505 HTTP Version not supporte**
 - A maioria dos browsers assumem que os servidores de rede suportam versões 1.x do protocolo HTTP.



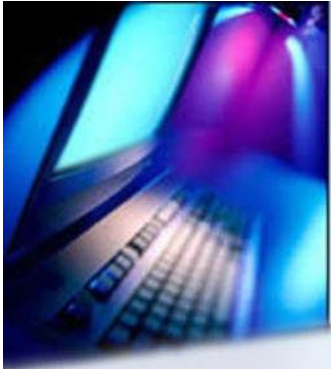
Exercício

- **Abra uma nova aba no navegador e tente acessar:**
- **<http://g1.globo.com/algo-que-nao-existe>**
- **Qual foi o código da resposta?**
 - **() 405**
 - **() 404**
 - **() 200**
 - **() 500**
- **Obs: Você precisa depurar a requisição HTTP para descobrir o código da resposta.**



Exercícios

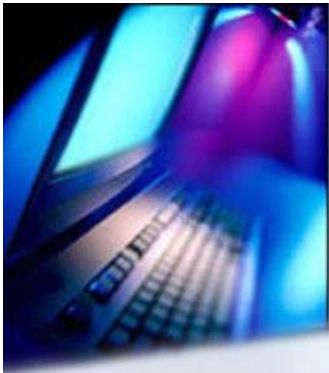
- **Pesquise o código de erro 3XX e 4XX descreva-os :**



Parâmetros do HTTP

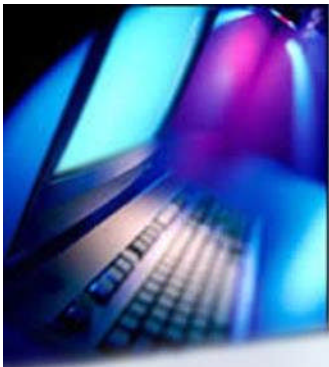
- Vamos ver um outro exemplo de uma URL e acessar <http://www.google.com>.
- Vamos pesquisar, por exemplo por Neymar.
- Repare que, ao pesquisar no google, a URL mudou um pouco.
- O recurso acessado pela busca se chama/results (os resultados da pesquisa) mas agora temos um parâmetro da requisição indicado pela ?

https://www.google.com.br/?gws_rd=ssl#q=neymar



Parâmetros do HTTP - GET

- Reparem que no console sempre aparece o tipo (ou método) da requisição, sendo GET.
- Uma característica da requisição GET é enviar os parâmetros pela URL!
- Isso é útil quando queremos deixar os parâmetros visíveis.
- Assim podemos facilmente guardar a URL com os parâmetros para repetir a requisição algum momento depois.
- Mas será que isso também é uma boa opção na hora de enviar credencias como login e senha?
- Queremos que aparece a senha de um usuário na URL?
- Imagina que você efetue o login no seu banco e na URL apareça: `https://www.bb.com.br/login?login=nico&password=supersecreto`



Parâmetros do HTTP - POST

- Vamos efetuar um login na área restrita do UniFOA para ver como esse sistema envia dados do usuário para o servidor.
- Repare que a URL para enviar o login senha se chama `http://portal.unifoa.edu.br/PortalSagres/Acesso.aspx`.
- Repare também que o método HTTP utilizado mudou.
- Estamos um usando o HTTP POST!
- Usando o POST, o navegador envia os dados do formulário no corpo da requisição e não na URL!
- Se fosse um GET, todos os dados seriam enviados através da URL.
- Como o UniFOA não deseja que os dados do login e senha apareçam na URL do navegador, foi utilizado um HTTP POST. Faz sentido?



Exercícios

- Vamos testar o envio de parâmetros através da requisição, fazendo uma busca no Google pela palavra UNIFOA.
- Para isso, na URI do Google vamos enviar uma requisição o parâmetro com o valor UNIFOA. Ou seja:
- <https://www.google.com.br/?q=UNIFOA>
- Ao entrar nessa URL, qual método HTTP foi usado?



Exercícios

- **Enviando parâmetros de forma correta**
- Como enviar de forma correta os parâmetros em uma URL?
 - () - <http://calculadordeimc.com.br/&peso=44?altura=1.50>
 - () - <http://calculadordeimc.com.br/?peso=44,altura=1.50>
 - () - <http://calculadordeimc.com.br/?peso=44&altura=1.50>
 - () - <http://calculadordeimc.com.br/?peso=44&altura=1.50>
 - () - <http://calculadordeimc.com.br/?peso:44&altura:1.50>



Exercícios

- **Enviando parâmetros de forma correta**
- Como enviar de forma correta os parâmetros em uma URL?

(X) - <http://calculadordeimc.com.br/&peso=44?altura=1.50>

() - <http://calculadordeimc.com.br/?peso=44,altura=1.50>

() - <http://calculadordeimc.com.br/?peso=44&altura=1.50>

() - <http://calculadordeimc.com.br/?peso=44&altura=1.50>

() - <http://calculadordeimc.com.br/?peso:44&altura:1.50>



Exercícios

- **Pesquise e explique a diferença entre os métodos GET e POST ?**