

1) Se você tivesse 5 diferentes arquivos de folhas de estilo, qual seria a melhor forma de integrá-los no site?

R: Supondo que o site necessite dos 5 diferentes estilos para diferentes áreas do site, poderia integrá-los conforme a necessidade das páginas, ou seja, se a página inicial utiliza apenas um dos estilos o HTML dessa página deve conter apenas um arquivo .css para evitar o excesso de requisições e manter a performance do site, o mesmo vale para as outras páginas.

2) Fale 3 formas de diminuir o page load (tempo de carregamento real e percebido).

R:

- Arquivos CSS e JS em arquivos separados;
- Otimizar ordem dos componentes;
- Reduzir número de scripts.

3) Quais ferramentas você usa para testar a performance do seu código?

R:

- Google Insights: pois analisa o conteúdo da página WEB e dá sugestões para melhorar a mesma;
- OctaGate SiteTimer: Exibe em cascata o carregamento dos objetos da página;
- GTmetrix: Gera relatórios detalhados do site, baixar os relatórios em PDF.

4) Considere o HTML5 como uma plataforma web aberta. Quais são os blocos de construção de HTML5?

R: O HTML5 trouxe importantes mudanças para padronização e semântica para páginas web. O HTML5 precisa de pelo menos 3 “pilares” que são as tags html, head, o e body, toda e qualquer tag mesmo as novas implementadas como a header, footer, nav, section necessita respeitar esses três pilares.

5) Você pode explicar a diferença entre GET e POST?

R: Ambos são métodos de comunicação HTTP.

GET: O método GET os parâmetros são passados no cabeçalho e por isso são vistos na URL, o tamanho máximo de dados passados geralmente é de 255 caracteres. O método GET tende a ser mais rápido (performance) mas não é uma regra. Um dos usos mais comuns do método é para formulários de busca e listagem de produtos cadastrados.

POST: Já o método POST envia os parâmetros no corpo da requisição, portanto não aparece na URL, porém pode ser vista quando inspecionada (para proteção usa-se o HTTPS, pois com ela se pode criptografar os dados). O método não tem um limite máximo de envio de dados e pode ser enviado grandes informações como imagem. Usado com frequência para envio de informações para processo, criação de conta ou produto.

6) Liste quantas propriedades display você puder lembrar.

R: propriedades de display:

- inline;
- block;
- inline-block;
- inline-grid;
- table;
- inline-table;
- table-caption;
- table-column;
- none;
- inherit;
- initial.

7) Qual a diferença entre inline e inline-block?

R: A propriedade inline permite que os elementos fiquem um do lado do outro, porém os mesmos ignoram as propriedades width e height, ou seja, seu conteúdo é definido pelo conteúdo que ele possui. A propriedade inline-block é a junção da propriedade inline + block, então além dos elementos ficarem lado a lado, podemos definir um limite de width e height.

8) Qual a diferença entre elementos posicionados de forma relativa, fixa, absoluta e estática?

R: Posicionamento relativo: quando aplicado por exemplo a propriedade `margin` ele terá uma distância em seu elemento pai, ele pode sobrepor os elementos irmãos que não possuem o mesmo tipo de posicionamento.

Posicionamento fixo: o elemento com essa propriedade irá se posicionar de acordo com o `body` e sempre será visível na página, caso a página tenha rolamento a mesma irá acompanhar no mesmo posicionamento.

Posicionamento absoluto: O mesmo ignora o posicionamento estático e/ou elementos que não possuem determinação independentemente de ser elemento pai de posicionamento, reconhece posicionamento fixo, absoluto e relativo.

Posicionamento estático: é posicionamento natural, se não for determinado nenhum posicionamento ele será estático.

9) Qual a diferença entre `.call` e `.apply`?

R: `.apply()`: permite que você invoque(chame) a função em que os argumentos são um array;

`.call()`: permite que você invoque(chame) a função e os parâmetros são listados explicitamente.

10) Qual a diferença entre `==` e `===`?

R: Operador `"=="`: Igualdade ampla, faz o comparativo de dois valores, independentemente de seu tipo.

Operador `"==="`: Igualdade restrita, faz o comparativo de dois valores, porém leva em consideração os valores e os tipos para a comparação.

