



Programação para a Web - 2019.1 Primeria Prova Parcial – Prova 02

Instruções Iniciais

Use o comando **git clone** para clonar seu repositório da disciplina na máquina local, caso ainda não possua este repositório em sua máquina. Acesse seu repositório na máquina local, e crie um diretório chamado **prova01**. Todas as questões da prova deverão ser respondidas dentro desse diretório. Após o término de cada questão, use os comandos **git add** e **git commit** para adicionar os arquivos da questão em seu repositório git. Quando terminar sua prova, não esqueça de usar o comando **git push origin master** para salvar todos os seus arquivos no GitHub.

Questões

1) **(2.0)** O código abaixo, feito em HTML e CSS, possui 4 (quatro) erros sintáticos. Identifique as linhas onde tais erros ocorreram, e crie um arquivo chamado **questao_01.txt** cujo conteúdo deve conter uma descrição dos quatro erros, as linhas onde tais erros ocorreram, e o que deve ser feito para corrigi-los. Após isso, use os comandos **git add** e **git commit** para adicionar o arquivo **questao_01.txt** no repositório GIT.

```
01 <!DOCTYPE html>
02 <html>
03   <head>
04     <comment>Isto é um comentário, não deve aparecer na página</comment>
05     <meta charset="utf-8">
06     <meta name="keywords" content="HTML,CSS,JavaScript">
07     <meta name="author" content="John Doe">
08     <title>Instituto de Computação</title>
09     <style>
10       width {
11         #para: 400px;
12       }
13     </style>
14   </head>
14   <h1>Instituto de Computação</h1>
15   <p id="para">
16     O Instituto de Computação (IComp), antigo
17     <span style="font-weight:bold">Departamento de Ciência
18     da Computação (DCC), é um instituto acadêmico que agrega
19     os professores da área de computação. Como todo instituto
20     acadêmico o IComp atua no ensino, pesquisa e
21     extensão, além de desempenhar atividades administrativas.
22   </p>
23   <script>
24     (function () {
25       var para = document.getElementById("para");
26       para.style.color = "red";
27     })();
28   </script>
29 </html>
```

2) (2.0) Entre 1928 e 2013, uma lei Turca proibiu o uso das letras Q, W, X em documentos oficiais. Essa lei entrou em vigor durante o conflito entre Curdos e Turcos, e teve como objetivo reprimir o uso da língua Curda, que faz uso dessas três letras. Desta forma, faça uma página Web contendo um único INPUT do tipo TEXT, que aceita todas os símbolos do teclado, com exceção das três letras proibidas pela lei Turca. Em outras palavras, essas três letras não podem aparecer no INPUT mesmo que o usuário pressione as teclas correspondentes. Um dica importante: Quando um usuário pressiona uma tecla dentro de um INPUT, a **ação padrão** desse evento é aparecimento do símbolo correspondente dentro do INPUT. Para impedir a ação padrão, você pode usar o método **event.preventDefault()**.

Após isso, crie um arquivo chamado **questao_02.html** e escreva o código HTML/JS gerado, e então use os comandos **git add** e **git commit** para adicionar o arquivo desta questão no repositório GIT. Use JavaScript embarcado, e não externo.

3) (2.0) Crie uma página HTML/CSS com os mesmos elementos da página ao lado. Tais elementos devem seguir as mesmas disposições, cores (amarelo no canto superior esquerdo, vermelho no superior direito, azul no inferior esquerdo, e verde no inferior direito) e estilos da página de exemplo. Use posicionamento absoluto (**position: absolute**) para dispor os quadrados em seus respectivos lugares. Além disso, você também vai precisar das propriedades CSS **top**, **bottom**, **right** e **left**. Cada quadrado deve ser representado por um elemento **div**, cujas propriedades **width** e **height** deverão ser iguais a **100px**. Após isso, crie um arquivo chamado **questao_03.html** e escreva o código HTML/CSS gerado, e então use os comandos **git add** e **git commit** para adicionar o arquivo desta questão no repositório GIT. Use CSS embarcado, e não externo.



4) (2.0) No diretório **prova01**, crie um arquivo chamado **questao_04.txt** contendo uma breve explicação sobre o significado de **hosting**. Ainda neste arquivo, descreva como o hosting afeta a interpretação de um código JavaScript, e dê um exemplo de código que explora (usa) essa característica da linguagem.

5) (2.0) Neste exercício, você pode optar por usar Classes ES6 ou Funções Construtoras. Crie uma classe chamada **Venda** para representar a venda de um determinado item de uma loja qualquer. Uma venda deve incluir os seguintes atributos: (i) o **ID** do item, (ii) a **quantidade** comprada do item e (iii) o **preço** unitário do item. Considere que uma instância da classe Venda representa a venda de um único item da loja. Sua classe deve ter um construtor que inicialize os três atributos. Forneça um método **set** e **get** para cada atributo. Além disso, forneça um método chamado **getValorTotal** que retorna o valor da venda (isto é, multiplica a quantidade pelo preço do item). Crie duas instâncias da classe Venda e exiba o valor total de cada uma delas através da função **console.log()**. Crie um arquivo chamado **questao_05.js** com o programa criado, e depois use os comandos **git add** e **git commit** para adicionar o arquivo desta questão no repositório GIT.

Observação: Quando finalizar a prova, use o comando **git push origin master** para salvar o conteúdo de sua prova no repositório remoto. Antes de ir embora, verifique com o professor se seu repositório está salvo corretamente no servidor.