



Programação para a Web - 2025.1 Lista de Exercícios

Questões

1) O código abaixo, feito em HTML, possui um grave erro sintático. Assinale qual é esse erro e descreva o que precisa ser feito para corrigi-lo:

```
<!DOCTYPE html>
<!-- Página do Instituto de Computação -->
<html>
  <head>
    <title>Instituto de Computação</title>
  </head>
  <body>
    <meta charset='UTF-8'>
    <h1>Instituto de Computação</h1>
    <img src='img/icom.png' width='260' height='111' alt='IComp'>
    <p>
      O <strong>Instituto de Computação (Icomp)</strong> é um
      instituto acadêmico da Universidade Federal do
      Amazonas que atua no ensino, pesquisa e extensão.
    </p>
  </body>
<!-- Fim da Página -->
</html>
```

Handwritten notes: tag <meta> está no lugar errado, ela deve estar no head do arquivo

2) No código acima, usamos os atributos **height** e **width** para informar as dimensões da imagem **imagens/icom.png** ao browser. Qual a importância desse procedimento, uma vez que o browser é capaz de identificar tais dimensões após o carregamento da imagem?

Handwritten note: É importante pois o browser irá fazer um "pré-carregamento" da imagem antes mesmo do carregamento completo do site

3) Escreva um código HTML para gerar a página da figura abaixo. Lembrando que as tags obrigatórias para criação de tabelas são **<table>**, **<tr>** e **<td>**. Você pode usar o atributo **colspan** de **<td>** para mesclar campos de uma mesma linha.

```
...
<table border="1" width="50%">
  <caption>Professores do Grupo...</caption>
  <thead>
    <tr>
      <td colspan="3">Prof. BDRI, UFAM</td>
    </tr>
    <tr>
      <th>Nome</th>
      <th>Cargo</th>
      <th>Email</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Altigran S. da Silva</td>
      <td>P. Assoc.</td>
      <td>alti@icomp.ufam.edu.br</td>
    </tr>
    <tr>
      <td>André Luis Carvalho</td>
      <td>P. Adj.</td>
      <td>andre@icomp.ufam.edu.br</td>
    </tr>
    <tr>
      <td>David F. de Oliveira</td>
      <td>P. Adj.</td>
      <td>david@icomp.ufam.edu.br</td>
    </tr>
    <tr>
      <td>Edleno S. de Moura</td>
      <td>P. Assoc.</td>
      <td>edleno@icomp.ufam.edu.br</td>
    </tr>
    <tr>
      <td>João Cavalcanti</td>
      <td>P. Adj.</td>
      <td>john@icomp.ufam.edu.br</td>
    </tr>
    <tr>
      <td>Marco Cristo</td>
      <td>P. Adj.</td>
      <td>marco@icomp.ufam.edu.br</td>
    </tr>
    <tr>
      <td>Moises de Carvalho</td>
      <td>P. Adj.</td>
      <td>moises@icomp.ufam.edu.br</td>
    </tr>
  </tbody>
</table>
```

Instituto de Computação - Google Chrome

Instituto de Computação

file:///var/www/pw/tabelas.html

Professores do Grupo Banco de Dados e Recuperação de Informação, do Instituto de Computação da UFAM

Professores do BDRI, UFAM		
Nome	Cargo	Email
Altigran S. da Silva	Professor Associado	alti@icomp.ufam.edu.br
André Luis Carvalho	Professor Adjunto	andre@icomp.ufam.edu.br
David F. de Oliveira	Professor Adjunto	david@icomp.ufam.edu.br
Edleno S. de Moura	Professor Associado	edleno@icomp.ufam.edu.br
João Cavalcanti	Professor Associado	john@icomp.ufam.edu.br
Marco Cristo	Professor Adjunto	marco@icomp.ufam.edu.br
Moises de Carvalho	Professor Adjunto	moises@icomp.ufam.edu.br

<table> + **<caption>** → legenda

thead → cabeçalho → título das colunas → **tr** e **th**

tbody → o corpo → informações das colunas → **tr** e **td** + **</table>**

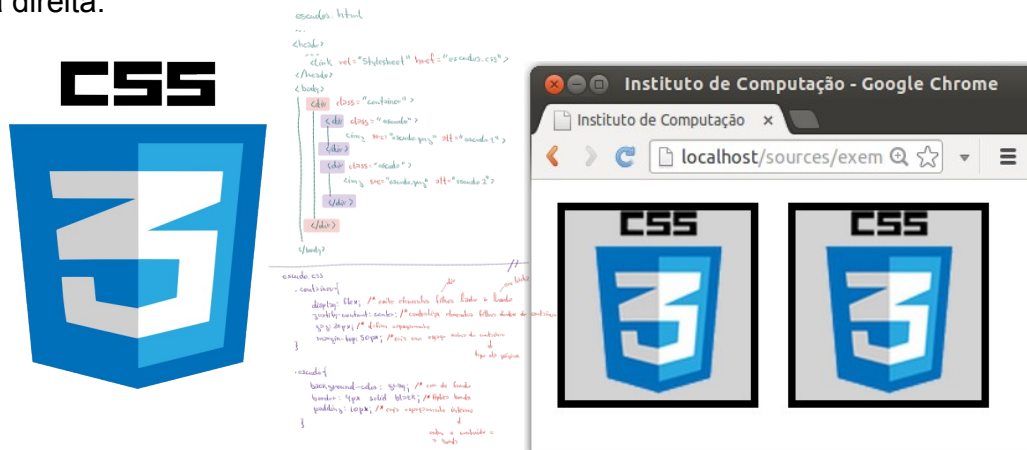
para título englobando todas as colunas: **<tr><th colspan="3"></th></tr>**

E-inline: `<h1 style="font-size: 22px; color: red;"> exemplo estilo inline </h1>`
 // `<head> <style> h1 { font-size: 22px; color: red; } </style> </head>`
 // `<head> <link rel="stylesheet" href="estilo.css"> </head>`
 // `estilo.css` → arquivo CSS
 // `h1 { font-size: 22px; color: red; }`

4) Descreva o que são **estilos inline**, **estilos embarcados**, e **estilos externos**. Escolha um dos três tipos de estilo e demonstre sua utilização em um código HTML/CSS.

estilos inline: são diretamente na tag html
 estilos embarcados: definidos no head do html com o elemento `<style>`
 estilos externos: diretrizes são construídas separadas do html. É utilizado um link no head do html
 5) Qual a diferença entre **unidades relativas** e **unidades absolutas**, usadas como medidas de tamanho na linguagem CSS? A unidade **em** é relativa ou absoluta? Como essa unidade é usada?
 Unidades Relativas: Dependem de um valor de referência, ou seja, se ajustam ao tamanho de um elemento da página. → em, rem, %
 Unidades Absolutas: São medidas fixas que não dependem de outro valor. → px, cm, in
 "em" é uma unidade relativa ao tamanho da fonte do elemento pai.

6) Considere o escudo CSS abaixo (à esquerda), que é uma imagem PNG com fundo transparente de nome **escudo.png**. A partir desta imagem, crie uma página Web semelhante à página da direita.



7) De que forma as 4 declarações CSS abaixo podem ser representadas através de um único comando CSS?

`padding-top: 5px;`
`padding-bottom: 15px;`
`padding-right: 0px;`
`padding-left: 10px;`
 padding: 5px 0px 15px 10px;

8) Analise o código HTML/CSS e a página resultante abaixo. Que tipo de seletor CSS pode ser usado para selecionar, de uma única vez, os dois trechos com borda mostrados na página?

```

<!DOCTYPE HTML>
<html>
  <head>
    <title>Example</title>
    <style type="text/css">
      {
        border: thin black solid;
        padding: 4px;
      }
    </style>
  </head>
  <body>
    <a class="cl1" href="http://icomp.ufam.edu.br">IComp</a>
    <p>Estou aprendendo <span class="cl1 cl2">CSS</span>.</p>
    <a href="http://w3c.org">Visite o website da W3C</a>
  </body>
</html>
  
```

IComp

CSS



IComp e CSS possuem classe cls
 -cls { /* Per classe
 ...
 }

9) O que é **tipagem dinâmica** de variáveis, adotada pela linguagem JavaScript? Escreva um breve código que demonstre essa característica da linguagem.

tipagem dinâmica: mudança de tipo durante a execução
 let b=10 // tipo: number
 b="a letra não é mais 10" // tipo: string
 let x; // não inicializado
 x=10; // number
 x="deu" // string
 x=true; // boolean

10) Descreva o que faz a seguinte sequência de código:

```
<div id="icomp">Instituto de Computação</div>
<button id="add">Adicionar Estilos</button>
<button id="clean">Limpar Estilo</button>
<script>
  var icomp = document.getElementById("icomp");
  document.getElementById("add").onclick = function() {
    icomp.style.setProperty("background-color", "lightgray");
    icomp.style.setProperty("color", "red");
    icomp.style.setProperty("font-size", "32px");
  }
  document.getElementById("clean").onclick = function() {
    icomp.style.removeProperty("background-color");
    icomp.style.removeProperty("color");
    icomp.style.removeProperty("font-size");
  }
</script>
```

Na divs botões e um texto
 estiliza após clicar no botão
 com fundo cinza claro
 cor vermelho
 fonte com 32px
 o botão clean
 remove esses
 estilos

11) Cite duas vantagens que o **método externo**, usado para incluir código javascript sobre uma página, possui sobre o método embarcado.

1. Reutilização do código
2. Melhor organização e legibilidade

12) A função **window.alert(string)** é usada para apresentar uma caixa de alerta com o conteúdo da string passada por parâmetro. Construa uma página HTML contendo um único botão. Quando o usuário clicar nesse botão, deverá ser exibida uma caixa de alerta com o texto **Botão Presionado**. Use o método **addEventListener** para programar o evento **click**.

13) Qual é o resultado da execução do código abaixo? Explique o motivo de dar o resultado informado.

```
(function() {
  var a = 5;
})();
console.log(a);
```

a foi declarada com var dentro do escopo da função logo ela é local a função por isso o erro — pois ela não existe no escopo global se fosse let ou const, daria o mesmo erro
 a is not defined

Diferença entre:
 * getElementById() — método do document, que busca elementos
 * addEventListener() — método de um elemento HTML, para reagir a eventos.

12) <body>
 <button id="bt">Clique aqui</button>
 <script src="script.js"></script>
 </body>

```
var botao=document.getElementById("bt");
botao.addEventListener("click",function(){
  window.alert("bt clicado");
});
```

14) Qual é o resultado da execução do código abaixo? Esse resultado sofreu a influência de **hoisting**?

*rola e proprio antes de ser executado
ele pega as declarações e vai as atribuições
para o topo do scope*

Interpretação hoisting:

```
function teste() {
  console.log(a);
  console.log(foo());


  var a = 1;
  function foo() {
    return 2;
  }
}

teste();
```

→ undefined
→ 2

Hoive hoisting, a variável a é criada com var, mas sem valor
A função foo() é criada com o corpo inteiro, então pode ser usada antes de sua linha no código

```
function teste() {
  var a; // criada, valor = undefined
  function foo() { // criada com definição
    return 2;
  }
  console.log(a); // undefined
  console.log(foo()); // 2
  a = 1; // só aqui recebe valor
}
```



15) Qual é o resultado da execução do código abaixo? Por que?

```
var fullname = 'John Doe';

var obj = {
  fullname: 'Colin Ihrig',
  prop: {
    fullname: 'Aurelio De Rosa',
    getFullname: function() {
      return this.fullname;
    }
  }
};

console.log(obj.prop.getFullname());
var test = obj.prop.getFullname;
console.log(test());
```

o this aponta para obj.prop
getFullname é chamada como um método do objeto obj.prop

this cai no escopo global = John Doe
função é copiada para test e executada fora de um objeto, ou seja, não aponta mais para obj.prop

Aurelio de Rosa
John Doe

16) Na linguagem Javascript, qual é a diferença entre **==** e **===** ?

↳ compara valores e tipos de valores

*olha somente
os valores das variáveis*