

Nome do Aluno: _____		Matrícula: _____	Nota: _____
Disciplina: Desenvolvimento Web	Período: _____	Prova: _____	
Professor: Jonh Carvalho			Ass. Professor
Curso: _____		Data: _____	2025.1

ATENÇÃO!

1. Se a prova for escrita a lápis, total ou parcialmente, o aluno não terá direito a solicitar revisão. As folhas de rosto e de perguntas deverão ser devolvidas juntamente com a prova.
2. Apenas os alunos cujos nomes constem na lista de frequência (ou seja, que estejam regularmente matriculados na disciplina) podem realizar esta prova.
3. Quando solicitado, o aluno deverá apresentar documento de identificação (carteira de identidade ou de habilitação) para que possa realizar a prova.
4. Não é permitido sair de sala nos primeiros 30 minutos do horário da prova. Após esse tempo, o aluno que sair de sala estará necessariamente concluindo a sua prova.
5. Ao terminar a prova, o aluno deve permanecer sentado e solicitar ao professor que a recolha.
6. A lista de frequência deve ser assinada pelo aluno no momento em que entregar a prova ao professor.
7. Após a correção, apenas o próprio aluno poderá examinar ou receber a prova. No ato da entrega da prova pelo professor, o aluno deverá assinar a lista de frequência.

Secretaria Acadêmica

Declaro estar ciente dos procedimentos acima

Assinatura do aluno

1. Uma empresa de tecnologia está desenvolvendo um site para uma biblioteca comunitária. O site deve ser acessível, responsivo e seguir as melhores práticas de desenvolvimento web. A equipe de desenvolvimento precisa garantir que o código HTML esteja bem estruturado, semântico e que atenda aos padrões de acessibilidade. O site terá uma página inicial com um cabeçalho, uma seção de livros em destaque, um formulário de cadastro para novos usuários e um rodapé com informações de contato. (0.5 ponto)

Analise o trecho de código HTML abaixo, que representa parte da estrutura do site da biblioteca comunitária, e responda à questão que segue:

```
```html
<!DOCTYPE html>
<html lang="pt-BR">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Biblioteca Comunitária</title>
</head>
<body>
 <header>
 <h1>Biblioteca Comunitária</h1>
 <nav>

 Livros em Destaque
 Cadastro
 Contato

 </nav>
 </header>

 <section id="destaques">
 <h2>Livros em Destaque</h2>

 O Pequeno Príncipe
 1984
 Dom Quixote

 </section>

 <section id="cadastro">
 <h2>Cadastro de Usuário</h2>
 <form action="#" method="post">
 <label for="nome">Nome:</label>
 <input type="text" id="nome" name="nome" required>
 <label for="email">E-mail:</label>
 <input type="email" id="email" name="email" required>
 <label for="senha">Senha:</label>
 <input type="password" id="senha" name="senha" required>
 <button type="submit">Cadastrar</button>
 </form>
 </section>

 <footer id="contato">
 <p>Contato: biblioteca@comunidade.com | (11) 1234-5678</p>
 </footer>
</body>
</html>
```
```

Com base no código HTML apresentado, qual das alternativas abaixo descreve corretamente um problema ou uma melhoria que pode ser aplicada ao código?

- A) O uso da tag `<section>` para dividir as seções do site está incorreto, pois essa tag só deve ser utilizada para conteúdo independente e que faz sentido por si só. O correto seria utilizar `<div>`.
- B) O atributo `for` nas tags `<label>` está sendo utilizado corretamente para associar os rótulos aos campos do formulário, garantindo acessibilidade e usabilidade.
- C) O código não utiliza a tag `<main>` para envolver o conteúdo principal da página, o que poderia melhorar a semântica e a organização do documento.
- D) O atributo `action` no formulário está definido como `#`, o que é uma prática recomendada para garantir que o formulário seja enviado corretamente.
- E) A tag `<footer>` está sendo utilizada de forma incorreta, pois ela só deve ser usada para informações de direitos autorais e não para informações de contato.

2. Uma empresa de design está desenvolvendo um site para uma cafeteria chamada "Café & Aroma". O site deve ser moderno, responsivo e seguir as melhores práticas de desenvolvimento web. A equipe de desenvolvimento precisa garantir que o código HTML e CSS estejam bem estruturados, semânticos e que atendam aos padrões de acessibilidade. O site terá uma página inicial com um cabeçalho, uma seção de destaques do menu, uma seção de depoimentos e um rodapé com informações de contato. (0.5 ponto)

Analise o trecho de código HTML e CSS abaixo, que representa parte da estrutura e estilização do site da cafeteria, e responda à questão que segue:

```
```html
<!DOCTYPE html>
<html lang="pt-BR">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Café & Aroma</title>
 <style>
 body {
 font-family: Arial, sans-serif;
 margin: 0;
 padding: 0;
 }
 header {
 background-color: #4CAF50;
 color: white;
 padding: 10px 0;
 text-align: center;
 }
 nav ul {
 list-style-type: none;
 padding: 0;
 }
 nav ul li {
 display: inline;
 margin: 0 15px;
 }
 nav ul li a {
```

```

 color: white;
 text-decoration: none;
 }
 main {
 padding: 20px;
 }
 section {
 margin-bottom: 20px;
 }
 footer {
 background-color: #333;
 color: white;
 text-align: center;
 padding: 10px 0;
 position: fixed;
 bottom: 0;
 width: 100%;
 }
</style>
</head>
<body>
 <header>
 <h1>Café & Aroma</h1>
 <nav>

 Destques
 Depoimentos
 Contato

 </nav>
 </header>

 <main>
 <section id="destaques">
 <h2>Destques do Menu</h2>

 Café Espresso
 Capuccino
 Latte Macchiato

 </section>

 <section id="depoimentos">
 <h2>Depoimentos</h2>
 <p>"Melhor café da cidade!" - Cliente Satisfeito</p>
 </section>
 </main>

 <footer id="contato">
 <p>Contato: contato@cafeearoma.com | (11) 9876-5432</p>
 </footer>
</body>
</html>

```

Com base no código HTML e CSS apresentado, qual das alternativas abaixo descreve corretamente um problema ou uma melhoria que pode ser aplicada ao código?

- A) O uso da propriedade `position: fixed` no rodapé faz com que ele fique sempre visível na parte inferior da tela, o que pode cobrir parte do conteúdo principal em dispositivos com telas pequenas.
- B) A propriedade `display: inline` aplicada aos itens da lista de navegação é a forma mais recomendada para criar menus horizontais, pois garante a melhor acessibilidade.
- C) A falta de uma media query no CSS impede que o site seja responsivo, já que não há ajustes específicos para diferentes tamanhos de tela.
- D) O uso da tag `` para envolver o conteúdo principal da página está incorreto, pois essa tag só deve ser utilizada para conteúdos secundários.
- E) A propriedade `text-align: center` aplicada ao cabeçalho e ao rodapé é desnecessária, pois o alinhamento centralizado não é uma prática recomendada em design web.

3. Um desenvolvedor está criando um sistema de cadastro para um evento acadêmico. O formulário deve coletar:

- Nome completo (obrigatório)
- E-mail (obrigatório, com validação)
- Telefone (opcional)
- Tipo de inscrição (estudante, profissional ou palestrante)
- Aceite dos termos (obrigatório)

O formulário precisa ser **acessível**, **responsivo** e seguir **boas práticas de HTML/CSS**. (0.5 ponto)

```
```html
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Inscrição - Evento Acadêmico</title>
  <style>
    body {
      font-family: 'Segoe UI', sans-serif;
      max-width: 600px;
      margin: 0 auto;
      padding: 20px;
    }
    .form-group {
      margin-bottom: 15px;
    }
    label {
      display: block;
      margin-bottom: 5px;
      font-weight: bold;
    }
    input[type="text"],
    input[type="email"],
    input[type="tel"],
    select {
      width: 100%;
    }
  </style>
</head>
<body>
  <div class="form-group">
    <label>Nome completo</label>
    <input type="text">
  </div>
  <div class="form-group">
    <label>E-mail</label>
    <input type="email">
  </div>
  <div class="form-group">
    <label>Telefone</label>
    <input type="tel">
  </div>
  <div class="form-group">
    <label>Tipo de inscrição</label>
    <select>
      <option value="estudante">Estudante</option>
      <option value="profissional">Profissional</option>
      <option value="palestrante">Palestrante</option>
    </select>
  </div>
  <div class="form-group">
    <label>Aceite dos termos</label>
    <input type="checkbox">
  </div>
</body>
</html>```
```

```

        padding: 8px;
        box-sizing: border-box;
    }
    button {
        background-color: #2E7D32;
        color: white;
        padding: 10px 15px;
        border: none;
        cursor: pointer;
    }
    .required:after {
        content: " *";
        color: red;
    }
}
</style>
</head>
<body>
    <h1>Formulário de Inscrição</h1>
    <form id="cadastro" action="/inscricao" method="POST">
        <div class="form-group">
            <label for="nome" class="required">Nome Completo</label>
            <input type="text" id="nome" name="nome" required>
        </div>

        <div class="form-group">
            <label for="email" class="required">E-mail</label>
            <input type="email" id="email" name="email" required>
        </div>

        <div class="form-group">
            <label for="telefone">Telefone</label>
            <input type="tel" id="telefone" name="telefone">
        </div>

        <div class="form-group">
            <label for="tipo" class="required">Tipo de Inscrição</label>
            <select id="tipo" name="tipo" required>
                <option value="">Selecione...</option>
                <option value="estudante">Estudante</option>
                <option value="profissional">Profissional</option>
                <option value="palestrante">Palestrante</option>
            </select>
        </div>

        <div class="form-group">
            <input type="checkbox" id="termos" name="termos" required>
            <label for="termos" class="required">Aceito os termos de
participação</label>
        </div>

        <button type="submit">Enviar Inscrição</button>
    </form>
</body>
</html>

```

Analisando o código do formulário, qual das alternativas abaixo **aponta um problema ou melhoria necessária**?

- A) O uso de `class="required"` nos rótulos (*labels*) é redundante, pois os campos já possuem o atributo `required` no HTML, tornando a estilização desnecessária.
- B) O campo de telefone (`type="tel"`) não possui um padrão de máscara (ex: `(00) 00000-0000`), o que pode levar a inconsistências nos dados enviados.
- C) O botão de envio não está estilizado para feedback visual (ex: `:hover`), o que prejudica a experiência do usuário.
- D) O uso de `box-sizing: border-box` nos campos de entrada (*input*) é incorreto, pois deve ser aplicado apenas em contêineres (*containers*).
- E) A validação do e-mail (`type="email"`) é suficiente, não sendo necessário usar `required`, pois navegadores modernos já bloqueiam envios sem um e-mail válido.

4. Um desenvolvedor está criando um site para um pequeno café usando técnicas de responsividade sem Flexbox ou Grid. O layout deve se adaptar a diferentes tamanhos de tela usando apenas: (0.5 ponto)

- Media Queries para breakpoints
- Unidades relativas (% , em , rem)
- Técnicas de float e display

```
```html
<!DOCTYPE html>
<html lang="pt-BR">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Café Artesanal</title>
 <style>
 * {
 box-sizing: border-box;
 margin: 0;
 padding: 0;
 }

 body {
 font-family: 'Arial', sans-serif;
 font-size: 1rem;
 line-height: 1.6;
 }

 .container {
 width: 90%;
 max-width: 1200px;
 margin: 0 auto;
 }

 header {
 background-color: #8B4513;
 color: white;
 padding: 1.5em 0;
 text-align: center;
 }

 .menu-item {
```

```

 width: 30%;
 float: left;
 margin-right: 5%;
 margin-bottom: 2em;
 padding: 1em;
 background: #f5f5f5;
 border: 1px solid #ddd;
 }

 .menu-item:nth-child(3n) {
 margin-right: 0;
 }

 footer {
 clear: both;
 background-color: #333;
 color: white;
 text-align: center;
 padding: 1.5em 0;
 }

 /* Tablet */
 @media (max-width: 900px) {
 .menu-item {
 width: 47%;
 margin-right: 6%;
 }

 .menu-item:nth-child(3n) {
 margin-right: 6%;
 }

 .menu-item:nth-child(2n) {
 margin-right: 0;
 }
 }

 /* Mobile */
 @media (max-width: 600px) {
 .menu-item {
 width: 100%;
 margin-right: 0;
 float: none;
 }
 }
</style>
</head>
<body>
 <header>
 <div class="container">
 <h1>Café Artesanal</h1>
 </div>
 </header>

 <main class="container">
 <div class="menu-item">
 <h2>Expresso</h2>
 <p>Café puro e encorpado</p>
 <p>R$ 4,50</p>
 </div>

 <div class="menu-item">

```



```

 <h2>Cappuccino</h2>
 <p>Café com leite vaporizado</p>
 <p>R$ 6,00</p>
 </div>

 <div class="menu-item">
 <h2>Latte</h2>
 <p>Café com leite cremoso</p>
 <p>R$ 7,50</p>
 </div>
</main>

<footer>
 <div class="container">
 <p>Horário de funcionamento: 8h às 20h</p>
 </div>
</footer>
</body>
</html>
...

```

Analisando o código do site do Café Artesanal, qual das alternativas abaixo identifica corretamente um problema ou melhoria necessária na implementação da responsividade?

- A) O uso de unidades relativas (em, rem) para padding e margens é inadequado, pois deveriam ser usados pixels (px) para maior precisão visual.
- B) A media query para tablets (max-width: 900px) está mal configurada, pois não redefine corretamente as margens dos itens na terceira coluna.
- C) O clear: both no footer é desnecessário, pois os floats já são naturalmente contidos pelo elemento main.
- D) A abordagem usando float é eficiente para layouts responsivos e não apresenta problemas de compatibilidade com navegadores modernos.
- E) A falta de uma unidade relativa para o max-width do container limita a adaptabilidade do layout em telas muito grandes.

**Bom teste!**