



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Criação de Páginas de WEB

AD2 1º semestre de 2013.

Observações importantes:

1. **Atenção: A avaliação é individual. Caso existam duas ou mais implementações coincidentes, independente de qualquer motivo, todas as avaliações envolvidas receberão nota ZERO!**
 2. As questões a seguir continuam a **implementação do site da Agência de Viagens**. Algumas questões fazem referência a questões da primeira avaliação à distância.
 3. Você deve fazer as modificações pedidas na implementação do site e incluí-las no mesmo endereço onde ele foi publicado na primeira avaliação à distância (caso tenha sido publicado) ou entregar os arquivos modificados para o tutor.
 4. É imprescindível que o tutor receba todos os arquivos que compõem o site em alguma mídia (CD ou disquete), devidamente identificada com o nome do aluno. **Não serão aceitos trabalhos APENAS em papel.** Se desejar o aluno pode TAMBÉM entregar uma listagem da sua avaliação.
 5. Fazer as ADs é muito importante, não apenas pela nota, mas principalmente pela experiência que permitirá um melhor desempenho nas avaliações presenciais. Os assuntos abordados na AD podem cair na prova presencial mesmo não tendo sido abordados nos vídeos ou no material escrito.
-

1. Modifique a página aberta com a seleção do link “Destinos” (questão 3 da AD1) para a aparência mostrada na fig. 1. As listas internas foram retiradas e cada elemento da lista principal virou um link.



Fig. 1 - Nova aparência para a página do link Destinos

Ao ser ativado este link abre uma janela como a aparência mostrada na fig. 2. Esta janela tem a largura e altura de 300px, um título centralizado (igual ao texto do link), uma imagem (Arraial da Ajuda para destinos nacionais e Machu Picchu para destinos internacionais), a lista de destinos e um botão que fecha a própria janela.



Fig. 2 – Janela aberta pelo link Destinos Nacionais

- a) Utilize vetores em javascript para armazenar as informações de cada tipo de destino. Por exemplo, um vetor de strings poderia guardar as informações de destinos nacionais, onde a primeira posição seria o título da janela, a segunda posição o nome do arquivo de imagem e as demais posições os vários destinos. Outro vetor guardaria as informações de destinos internacionais. Percorra o vetor apropriado para montar a janela **[1 ponto]**.
- b) Escreva uma função em javascript que abra uma janela e monte o código HTML da página que será exibida nela. Esta função deve ser ativada através da seleção de um link **[1 ponto]**.

OBS. IMPORTANTE: No Firefox é necessário incluir na tag <a> o atributo target com valor "_self" para que se possa executar uma função a partir da seleção de um link.

2. Modifique a página aberta com a seleção do link “Pacotes” (questão 4 da AD1) para que a aparência da tabela seja a mostrada na fig. 3. O interior da célula ao lado de “opções de hospedagem” foi modificado para que apareçam 3 botões que representam os três tipos de quarto oferecidos pela pousada “Pelado na Praia”.

<p>7 dias em Arraial da Ajuda</p> 		
Transportes Inclusos	Aéreo	Teco Teco Air Lines
	Translado	Viação Boleia
Opções de Hospedagem	<p>Pousada Pelado na Praia</p> <p> <input type="button" value="Duplo"/> <input type="button" value="Triplo"/> <input type="button" value="Múltiplo"/> </p>	
Preço por Pessoa	<p>R\$ 500,00 + 7 x Diária do Quarto</p>	

Fig. 3 - Nova aparência para a página do link Pacotes

Faça com que a seleção de cada um dos botões ative uma função em javascript que modificará o conteúdo da célula ao lado de “Preço por Pessoa”. O valor que será colocado nesta célula contém o total da hospedagem por pessoa do tipo de quarto selecionado (fig. 4). As diárias de cada quarto foram especificadas na AD1: duplo – R\$ 70,00; triplo – R\$ 50,00; múltiplo – R\$ 30,00. **[2 pontos]**.

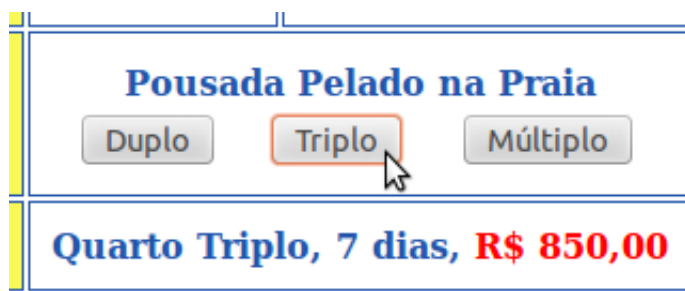


Fig. 4 – Valor da célula após a seleção do botão do quarto triplo

DICA: Pesquise o funcionamento do método `getElementById()` para descobrir como obter o objeto correspondente à célula que terá seu conteúdo modificado. Pesquise o funcionamento da propriedade `innerHTML` para descobrir como modificar o HTML do conteúdo de um elemento da página, após a página já ter sido criada.

3. Modifique a página de Cadastro (questão 5 da AD1) e crie uma função javascript que verifique a validade de um número de telefone ou celular digitado pelo usuário. Esta função deve ser chamada quando o valor do campo telefone for modificado (crítica de campo) e deve exibir uma mensagem de alerta caso o valor do campo não seja válido. A mensagem deve informar corretamente a qual campo se refere (celular ou telefone fixo). As verificações são:

- a) Se o usuário digitou 8 caracteres (um número de telefone é composto de 8 dígitos) como mostrado na fig. 5. **[1 ponto]**

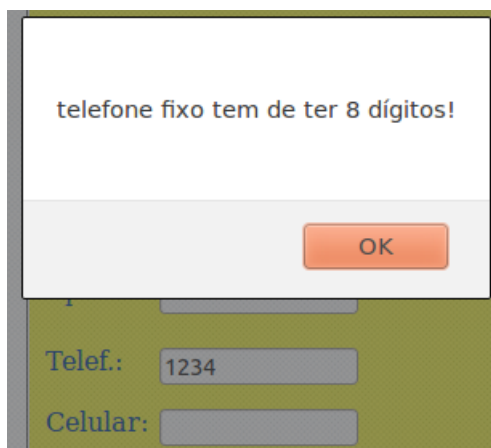


Fig. 5 - Erro ao digitar um telefone com menos de 8 dígitos.

- b) Se todos os caracteres digitados são dígitos, ou seja, se estão entre 0 e 9 (fig. 6).

[1 ponto]

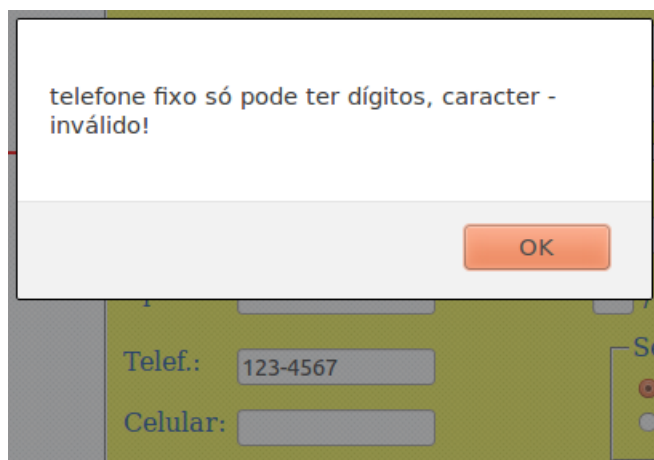


Fig. 6 - Erro ao digitar um telefone com caracteres que não são dígitos.

Caso o valor seja válido, o usuário pode passar para o próximo campo do formulário sem que nenhuma mensagem seja emitida (fig. 7).

A screenshot of a web form with a yellow background. It contains three input fields: "Cpf:" (empty), "Nasc.:" (empty), and "Telef.:" (containing "12345678"). Below these is a "Celular:" field (empty). The form is part of a larger interface with a green header.

Fig. 7- Campos de telefone que não geraram mensagem de erro.

4. Modifique a página de Cadastro (questão 5 da AD1) para que uma função javascript seja chamada quando o valor do CPF for modificado. Esta função deve fazer três verificações e exibir uma mensagem de alerta caso o valor do CPF não seja válido:

- a) Se o usuário digitou 11 caracteres (um CPF é composto de 9 dígitos de identificação e 2 dígitos de verificação) como mostrado na fig. 8. **[1 ponto]**

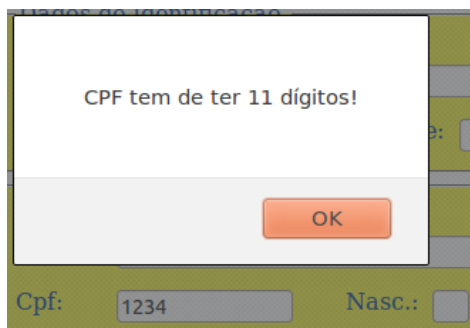


Fig. 8 - Erro ao digitar um CPF com menos de 11 dígitos.

- b) Se todos os caracteres digitados são dígitos (estão entre 0 e 9 - fig. 9). **[1 ponto]**

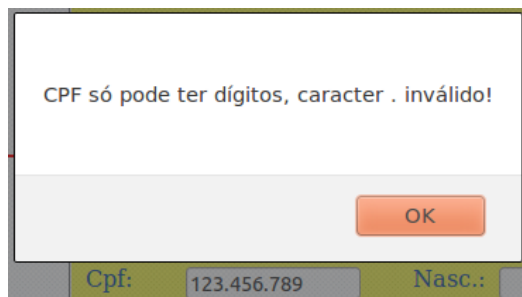


Fig. 9 - Erro ao digitar um CPF com caracteres que não são dígitos.

- c) Se os dígitos de verificação estão coerentes com os dígitos de identificação informados (fig. 10). **[1 ponto]**

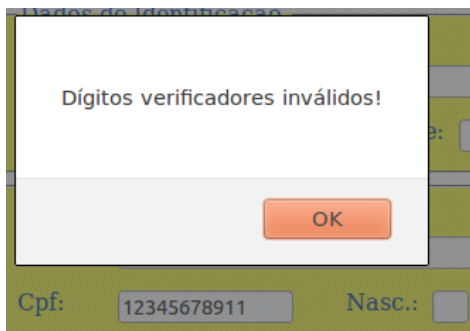


Fig. 10 - Erro de dígitos verificadores incorretos.

DICA: A função a seguir mostra como é feito o cálculo dos dígitos verificadores a partir dos nove dígitos de identificação do CPF (identCPF no código). Utilize-a para calcular quais dígitos você espera e comparar com os dígitos recebidos:

```
function calculaDV(num)
{
    var resto = 0, soma = 0;
    for (i = 2; i < 11; i++)
    {
        soma = soma + ((num % 10) * i);
        num = parseInt(num / 10);
    }
    resto = (soma % 11);
    return (resto > 1) ? (11 - resto) : 0;
}

primeiro_digito = calculaDV(identCPF)
segundo_digito  = calculaDV(identCPF * 10 + primeiro_digito)
```

Caso o valor do CPF seja válido, o usuário pode passar para o próximo campo do formulário sem que nenhuma mensagem seja emitida (figura 11).

Fig. 11 - Saída do campo com um CPF válido.

5. Modifique a página de Reservas (questão 6 da AD1) e associe uma função ao clique nos botões de rádio das formas de pagamento. Esta função deve controlar a exibição do fieldset “Dados do Cartão”. Se a forma de pagamento “cartão” for selecionada, o fieldset ficará visível (fig. 12). Se qualquer outra forma de pagamento for selecionada, o fieldset ficará invisível (fig. 13) **[1 ponto]**

Forma de Pagamento

☒ Cartão ☐ Cheque ☐ Dinheiro

Dados do Cartão

☒  ☐  ☐ 

Validade: /

Numero:

Fig. 12 – Fieldset visível com a forma de pagamento “cartão” selecionada.

Forma de Pagamento

☐ Cartão ☒ Cheque ☐ Dinheiro

Fig. 13 – Fieldset invisível para outra forma de pagamento selecionada.

DICA: Pesquise o funcionamento do método `getElementById()`. Pesquise a utilização da propriedade **style** nos objetos. Pesquise a utilização da propriedade CSS **visibility**.



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Criação de Páginas de WEB

Gabarito AD2 1º semestre de 2013.

Observações importantes:

1. **Atenção: A avaliação é individual. Caso existam duas ou mais implementações coincidentes, independente de qualquer motivo, todas as avaliações envolvidas receberão nota ZERO!**
 2. As questões a seguir continuam a **implementação do site da Agência de Viagens**. Algumas questões fazem referência a questões da primeira avaliação à distância.
 3. Você deve fazer as modificações pedidas na implementação do site e incluí-las no mesmo endereço onde ele foi publicado na primeira avaliação à distância (caso tenha sido publicado) ou entregar os arquivos modificados para o tutor.
 4. É imprescindível que o tutor receba todos os arquivos que compõem o site em alguma mídia (CD ou disquete), devidamente identificada com o nome do aluno. **Não serão aceitos trabalhos APENAS em papel.** Se desejar o aluno pode TAMBÉM entregar uma listagem da sua avaliação.
 5. Fazer as ADs é muito importante, não apenas pela nota, mas principalmente pela experiência que permitirá um melhor desempenho nas avaliações presenciais. Os assuntos abordados na AD podem cair na prova presencial mesmo não tendo sido abordados nos vídeos ou no material escrito.
-

1. Modifique a página aberta com a seleção do link “Destinos” (questão 3 da AD1) para a aparência mostrada na fig. 1. As listas internas foram retiradas e cada elemento da lista principal virou um link.

Ao ser ativado este link abre uma janela como a aparência mostrada na fig. 2. Esta janela tem a largura e altura de 300px, um título centralizado (igual ao texto do link), uma imagem (Arraial da Ajuda para destinos nacionais e Machu Picchu para destinos internacionais), a lista de destinos e um botão que fecha a própria janela.

- a) Utilize vetores em javascript para armazenar as informações de cada tipo de destino. Por exemplo, um vetor de strings poderia guardar as informações de destinos nacionais, onde a primeira posição seria o título da janela, a segunda posição o nome do arquivo de imagem e as demais posições os vários destinos. Outro vetor guardaria as informações de destinos internacionais. Percorra o vetor apropriado para montar a janela **[1 ponto]**.
- b) Escreva uma função em javascript que abra uma janela e monte o código HTML da página que será exibida nela. Esta função deve ser ativada através da seleção de um link **[1 ponto]**.

OBS. IMPORTANTE: No Firefox é necessário incluir na tag <a> o atributo target com valor "_self" para que se possa executar uma função a partir da seleção de um link.

```
<!DOCTYPE html>
<html><head><title>Pé na Estrada</title>
<base target="_blank" />
<link rel="stylesheet" type="text/css" href="penaestrada.css" />
<script>
var tabDestinos = new Array();

tabDestinos[0] = [ "Nacionais",
                  "pac_ajuda",
                  "Arraial da Ajuda",
                  "Ibitipoca",
                  "Visconde de Maua" ];

tabDestinos[1] = [ "Internacionais",
                  "pac_mpicchu",
                  "Colonia de Sacramento",
                  "Deserto de Atacama",
                  "Machu Picchu" ];
```

```

function MostraDestinos( nd ) {
    var jan = open("", "destino",
        "location=no,status=no,width=300,height=300");
    with (jan.document) {
        write("<html><head><title>Pe na Estrada</title>");
        write("<link rel='stylesheet' type='text/css'");
        write(" href='penaestrada.css'>");
        write("</head><body>");
        write("<h3 class='titDest'>Destino " +
            tabDestinos[nd][0] + "</h3>");
        write("<p class='apres'><img src='" +
            tabDestinos[nd][1] + ".jpg' /></p>");
        write("<ul class='listaDest'>");
        for ( i = 2; i < tabDestinos[nd].length; i++ )
            write("<li>" + tabDestinos[nd][i] + "</li>");
        write("</ul>");
        write("<form>");
        write("<input type='button' value='Fechar' ");
        write("onClick='window.close();' />");
        write("</form></body></html>");
        close();
    }
}

</script>

</head>
<body>
<header> ... </header>
<nav> ... </nav>

<section>
<h3 class="titDest"> Pacotes de 7 Dias</h3>
<ol>
<li><a target="_self"
    href="javascript:MostraDestinos(0);">
    Destinos Nacionais</a></li>
<li><a target="_self"
    href="javascript:MostraDestinos(1);">
    Destinos Internacionais</a></li>
</ol>
<br/><br/><br/>
</section>

<footer> ... </footer>
</body>
</html>

```

2. Modifique a página aberta com a seleção do link “Pacotes” (questão 4 da AD1) para que a aparência da tabela seja a mostrada na fig. 3. O interior da célula ao lado de “opções de hospedagem” foi modificado para que apareçam 3 botões que representam os três tipos de quarto oferecidos pela pousada “Pelado na Praia”.

Faça com que a seleção de cada um dos botões ative uma função em javascript que modificará o conteúdo da célula ao lado de “Preço por Pessoa”. O valor que será colocado nesta célula contém o total da hospedagem por pessoa do tipo de quarto selecionado (fig. 4). As diárias de cada quarto foram especificadas na AD1: duplo – R\$ 70,00; triplo – R\$ 50,00; múltiplo – R\$ 30,00. [2 pontos].

DICA: Pesquise o funcionamento do método `getElementById()` para descobrir como obter o objeto correspondente à célula que terá seu conteúdo modificado. Pesquise o funcionamento da propriedade `innerHTML` para descobrir como modificar o HTML do conteúdo de um elemento da página, após a página já ter sido criada.

```
<!DOCTYPE html>
<html><head><title>Pe na Estrada</title>
<base target="_blank" />
<link rel="stylesheet" type="text/css" href="penaestrada.css" />
<script>
tabPrecos = [ 70, 50, 30 ];
tabQuartos = [ "Duplo", "Triplo", "Múltiplo" ];

function MostraPreco( nq ) {
    var preco = document.getElementById("precoTotal");
    preco.innerHTML = "Quarto " + tabQuartos[nq] +
        ", 7 dias, <span class='destac'>R$ " +
        (7 * tabPrecos[nq] + 500) +
        ",00</span>";
}

function CriaBotoes() {
    for ( i = 0; i < tabQuartos.length; i++ ) {
        document.write('<th><input type="button" value="">');
        document.write(tabQuartos[i]);
        document.write('" OnClick="MostraPreco(' + i);
        document.write(')";' /></th>');
    }
}
</script></head>
```

```

<body>
<header> ... </header>
<nav> ... </nav>

<section>
<form>
<table class="tabPac">
<tr><th colspan="3" class="linImgPac">
    <span class="destac">7 dias</span>
    em Arraial da Ajuda<br/><br/>
    </th>
</tr>
<tr><th rowspan="2" class="colLegPac">
    Transportes<br/>Inclusos</th>
    <td class="colDestac">Aéreo</td>
    <td>Teco Teco Air Lines</td></tr>
<tr><td class="colDestac">Translado</td>
    <td>Viação Boleia</td></tr>
<tr><th class="colLegPac">Opções de<br/>Hospedagem</th>
    <td colspan="2">
    <table class="tabInterna">
    <tr><th colspan="3">Pousada Pelado na Praia</th>
    <tr><script>CriaBotoes();</script></tr>
    </table>
    </td></tr>
<tr><th class="colLegPac">Preço por Pessoa</th>
    <th colspan="2" class="colDestac" id="precoTotal">
    <span class="destac">R$ 500,00</span>
    + 7 x Diária do Quarto</th></tr>
</table>
<br/>
</section>

<footer> ... </footer>
</body>
</html>

```

3. Modifique a página de Cadastro (questão 5 da AD1) e crie uma função javascript que verifique a validade de um número de telefone ou celular digitado pelo usuário. Esta função deve ser chamada quando o valor do campo telefone for modificado (crítica de campo) e deve exibir uma mensagem de alerta caso o valor do campo não seja válido. A mensagem deve informar corretamente a qual campo se refere (celular ou telefone fixo). As verificações são:

- a) Se o usuário digitou 8 caracteres (um número de telefone é composto de 8 dígitos) como mostrado na fig. 5. **[1 ponto]**
- b) Se todos os caracteres digitados são dígitos, ou seja, se estão entre 0 e 9 (fig. 6). **[1 ponto]**

Caso o valor seja válido, o usuário pode passar para o próximo campo do formulário sem que nenhuma mensagem seja emitida (fig. 7).

```
<!DOCTYPE html>
<html><head><title>Pe na Estrada</title>
<base target="_blank" />
<link rel="stylesheet" type="text/css" href="penaestrada.css" />
<script>

function VerificaTelef( ident, campo ) {
    var i, c;
    var strTel = campo.value;

    if ( strTel.length != 8 ) {
        alert(ident + " tem de ter 8 dígitos!");
        return false;
    }

    for ( i = 0; i < 8; i++ ) {
        c = strTel.charAt(i);
        if ( (c < '0') || (c > '9')) {
            alert(ident + " só pode ter dígitos, caracter " +
                c + " inválido!");
            return false;
        }
    }

    return true;
}
</script>
```

```

</head>
<body>
<header> ... </header>
<nav> ... </nav>

<section>

<form method="POST">

<fieldset class="fsForm">
<legend> Dados de Identificação&nbsp;</legend>
...
</fieldset>

<fieldset class="fsForm">
<legend> Dados Pessoais&nbsp;</legend>
...
<br/><label for="tel" class="lab70">Telef.: </label>
    <input type="text" class="ent120Form"
        id="tel" name="TxtTelefone"
        onchange="VerificaTelef('telefone fixo', this);" /><br/>
<br/><label for="cel" class="lab70">Celular: </label>
    <input type="text" class="ent120Form"
        id="cel" name="TxtCelular"
        onchange="VerificaTelef('celular', this);" /><br/>
</fieldset>

<p class="entDireita">
    <input type="submit" value="Enviar" name="BtnEnviar">
    <input type="reset" value="Limpar" name="BtnLimpar">
</p>
</form>

</section>

<footer> ... </footer>
</body>
</html>

```

4. Modifique a página de Cadastro (questão 5 da AD1) para que uma função javascript seja chamada quando o valor do CPF for modificado. Esta função deve fazer três verificações e exibir uma mensagem de alerta caso o valor do CPF não seja válido:

- a) Se o usuário digitou 11 caracteres (um CPF é composto de 9 dígitos de identificação e 2 dígitos de verificação) como mostrado na fig. 8. **[1 ponto]**
- b) Se todos os caracteres digitados são dígitos (estão entre 0 e 9 - fig. 9). **[1 ponto]**
- c) Se os dígitos de verificação estão coerentes com os dígitos de identificação informados (fig. 10). **[1 ponto]**

DICA: A função a seguir mostra como é feito o cálculo dos dígitos verificadores a partir dos nove dígitos de identificação do CPF (identCPF no código). Utilize-a para calcular quais dígitos você espera e comparar com os dígitos recebidos:

Caso o valor do CPF seja válido, o usuário pode passar para o próximo campo do formulário sem que nenhuma mensagem seja emitida (figura 11).

```
<!DOCTYPE html>
<html><head><title>Pe na Estrada</title>
<base target="_blank" />
<link rel="stylesheet" type="text/css" href="penaestrada.css" />
<script>

function VerificaTelef( ident, campo ) {
    ...
}

function calculaDV(num)
{
    var resto = 0, soma = 0;
    for ( i = 2; i < 11; i++ ) {
        soma = soma + ((num % 10) * i);
        num = parseInt(num / 10);
    }

    resto = (soma % 11);
    return (resto > 1) ? (11 - resto) : 0;
}
```



```

function VerificaCPF( campo ) {
    var i, c, cpf;
    var iniCPF;
    var strCPF = campo.value;

    if ( strCPF.length != 11 ) {
        alert("CPF tem de ter 11 dígitos!");
        return false;
    }

    for ( i = 0; i < 11; i++ ) {
        c = strCPF.charAt(i);
        if ( (c < '0') || (c > '9')) {
            alert("CPF só pode ter dígitos, caracter " +
                c + " inválido!");
            return false;
        }
    }

    iniCPF = strCPF.substring(0, 9);

    pd = calculaDV(iniCPF);
    sd = calculaDV(iniCPF * 10 + pd);

    if ((pd != strCPF.charAt(9)) || (sd != strCPF.charAt(10))) {
        alert("Dígitos verificadores inválidos!");
        return false;
    }

    return true;
}

</script>
</head>
<body>
<header> ... </header>
<nav> ... </nav>

<section>

<form method="POST">

<fieldset class="fsForm">
<legend> Dados de Identificação&nbsp;&nbsp;&nbsp;</legend>
...
</fieldset>

<fieldset class="fsForm">
<legend> Dados Pessoais&nbsp;&nbsp;&nbsp;</legend>
<br/><label for="nome" class="lab70">Nome: </label>
    <input type="text" class="ent360Form"
        id="nome" name="TxtNome" /><br/>

```

```

<br/><label for="cpf" class="lab70">Cpf: </label>
  <input type="text" class="ent120Form"
    id="cpf" name="TxtCpf"
    onChange="VerificaCPF(this);" />
  <label for="aniv" class="labNasc"> Nasc.: </label>
  <input type="text" class="ent20Form"
    id="aniv" name="TxtDiaNasc" /> /
  <input type="text" name="TxtMesNasc" class="ent20Form"/> /
  <input type="text" name="TxtAnoNasc" class="ent40Form"/>
<br/><fieldset class="fsSexo">
  <legend> Sexo&nbsp;  </legend>
  <input type="radio" value="M" checked
    name="TglSexo"> Masculino<br/>
  <input type="radio" name="TglSexo" value="F"> Feminino<br/>
</fieldset>
<br/><label for="tel" class="lab70">Telef.: </label>
  <input type="text" class="ent120Form"
    id="tel" name="TxtTelefone"
    onchange="VerificaTelef('telefone fixo', this);" /><br/>
<br/><label for="cel" class="lab70">Celular: </label>
  <input type="text" class="ent120Form"
    id="cel" name="TxtCelular"
    onchange="VerificaTelef('celular', this);" /><br/>
</fieldset>

<p class="entDireita">
  <input type="submit" value="Enviar" name="BtnEnviar">
  <input type="reset" value="Limpar" name="BtnLimpar">
</p>
</form>

</section>
<footer> ... </footer>
</body>
</html>

```

5. Modifique a página de Reservas (questão 6 da AD1) e associe uma função ao clique nos botões de rádio das formas de pagamento. Esta função deve controlar a exibição do fieldset “Dados do Cartão”. Se a forma de pagamento “cartão” for selecionada, o fieldset ficará visível (fig. 12). Se qualquer outra forma de pagamento for selecionada, o fieldset ficará invisível (fig. 13) [1 ponto]

DICA: Pesquise o funcionamento do método `getElementById()`. Pesquise a utilização da propriedade `style` nos objetos. Pesquise a utilização da propriedade CSS `visibility`.

```
<!DOCTYPE html>
<html><head><title>Pe na Estrada</title>
<base target="_blank" />
<link rel="stylesheet" type="text/css" href="penaestrada.css" />
<script>

function MostraCartoes() {
    var cart = document.getElementById("cartoes");
    cart.style.visibility = 'visible';
}

function EscondeCartoes() {
    var cart = document.getElementById("cartoes");
    cart.style.visibility = 'hidden';
}

</script>
</head>
<body>
<header> ... </header>
<nav> ... </nav>

<section>

<form method="POST">

    <fieldset class="fsTopo">
        <legend> Dados de Identificação&nbsp;</legend>
        ...
    </fieldset>
```

```

<fieldset class="fsResDir">
<legend> Forma de Pagamento&nbsp;</legend>
  <br/><input type="radio" name="RadForma" id="car"
    <b>OnClick="MostraCartoes();" checked />
    <label for="car">Cartão</label>
    <input type="radio" name="RadForma" id="che"
    <b>OnClick="EscondeCartoes();" />
    <label for="che">Cheque</label>
    <input type="radio" name="RadForma" id="din"
    <b>OnClick="EscondeCartoes();" />
    <label for="din">Dinheiro</label>
  <br/><br/>
  <fieldset id="cartoes">
  <legend> Dados do Cartão&nbsp;</legend>
  <br/><input type="radio" name="RadBand" id="visa" checked />
    <label for="visa"></label>
    <input type="radio" name="RadBand" id="mast" />
    <label for="mast"></label>
    <input type="radio" name="RadBand" id="amex" />
    <label for="amex"></label><br/>
  <br/><label for="val" class="lab90">Validade: </label>
    <input type="text" class="ent20Form"
      id="val" name="TxtValMes" /> /
    <input type="text" class="ent40Form"
      name="TxtValAno" /><br/>
  <br/><label for="num" class="lab90">Numero: </label>
    <input type="text" class="ent120Form"
      id="num" name="TxtNumero" /><br/>
  </fieldset>
</fieldset>

  <fieldset class="fsResEsq">
  <legend> Passeio Escolhido&nbsp;</legend>
  ...
  </fieldset>

<p class="entDireita">
  <input type="submit" value="Enviar" name="BtnEnviar">
  <input type="reset" value="Limpar" name="BtnLimpar">
  </p>
</form>

</section>
<footer> ... </footer>
</body>
</html>

```