



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Criação de Páginas de WEB

AD2 2º semestre de 2019.

Observações importantes:

1. As questões a seguir continuam a implementação do site da loja de roupas. Antes de entregar o trabalho, **não se esqueça de consertar seus eventuais erros da AD1 conforme o gabarito divulgado**. Algumas questões fazem referência às questões da primeira AD.
 2. Você deve fazer as modificações pedidas no site, gerar um arquivo ZIP com o site completo e postá-lo na atividade correspondente na plataforma. Não se esqueça de confirmar o rascunho como versão definitiva.
 3. Lembrando que os trabalhos devem ser feitos utilizando editores simples, que não incluam código no texto. Utilize preferencialmente o Notepad (Windows) ou o Gedit (Linux). Outros editores, com esta característica, podem ser utilizados como Notepad++ e PsPad (Windows); Kate e Kwrite (Linux); Coda e TextMate (Mac); ou Gedit, Geany, Atom e Visual Studio Code (Multiplataforma).
 4. **ATENÇÃO:** A avaliação é individual. Caso existam duas ou mais implementações excessivamente coincidentes, independente de qualquer motivo, todas as avaliações envolvidas receberão nota ZERO!
 5. Não serão aceitos trabalhos em papel. O aluno deve postar na atividade determinada na plataforma. É imprescindível que o tutor receba todos os arquivos que compõem o site. **A entrega destes arquivos deve ser feita através DA PLATAFORMA na forma de um arquivo de extensão zip.**
 6. **Não serão aceitos trabalhos em arquivos de texto DOC ou PDF.** Os arquivos HTML, JS ou CSS devem estar individualizados (mesmo que agrupados num ZIP) para que o tutor possa testar seu site.
 7. **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 “Moletons” (questão 3 da AD1) para que as imagens de cores virem links, conforme podemos ver na figura 1, que ao ser selecionado causa a abertura de uma janela cujo conteúdo é exemplificado nas figuras 2 e 3. O conteúdo das janelas será sempre: um título indicando marca e cor, uma imagem, os tamanhos disponíveis, o preço e um botão para fechar a janela. **[2 pontos]**



Fig. 1 – Nova página do link Moletons



Fig. 2 – Seleção do link Hering Preta



Fig. 3 – Seleção do link Ripping Azul

DICA 1: Alguns preços e disponibilidades de tamanho mudaram em relação à questão 3 da AD1. Agora todos produtos da mesma marca têm mesmo preço e tamanhos (fig. 4).

DICA 2: Acrescente ao arquivo de estilos (.CSS) uma classe para a imagem mostrada dentro da janela com: largura e altura de 200px, borda sólida de 3px, padding de 12px e cor de fundo branca.

I. Hering:	
	- Cinza - P/M/G/XG - R\$ 70,00
	- Preto - P/M/G/XG - R\$ 70,00
II. Malwee:	
	- Cinza - M/G/XG - R\$ 40,00
III. Ripping:	
	- Azul - PP/P/M/G/GG - R\$ 80,00
	- Cinza - PP/P/M/G/GG - R\$ 80,00
	- Preto - PP/P/M/G/GG - R\$ 80,00

Fig. 4 – Nova tabela de preços e disponibilidade de tamanhos

2. Modifique a página aberta com a seleção do link “Camisetas” (questão 4 da AD1) para que a tabela fique com o aspecto mostrado na figura 5. Uma figura vazia (“Vazio.jpg”) aparece centralizada na terceira coluna da tabela (fig. 5).

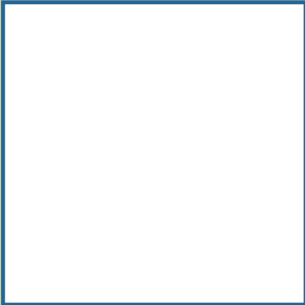
Camisas		
Marca	Cor	Foto/Tamanhos/Preço
Dudalina	<input type="checkbox"/>	
Hering	<input type="checkbox"/>	
	<input type="checkbox"/>	
	<input type="checkbox"/>	
Marfinno	<input type="checkbox"/>	
	<input type="checkbox"/>	

Fig. 5 - Nova aparência da tabela da página Camisetas

- a) Crie um vetor (ou dois) em javascript contendo, para cada produto, duas informações: o nome do arquivo de sua foto e seu preço. **[1 ponto]**

Camisas		
Marca	Cor	Foto/Tamanhos/Preço
Dudalina	<input type="checkbox"/>	 <p>Tamanhos: P/M/G/XG/XXG</p> <p>Preço: R\$ 70,00</p>
Hering	<input type="checkbox"/>	
	<input type="checkbox"/>	
	<input type="checkbox"/>	
Marfinno	<input type="checkbox"/>	
	<input type="checkbox"/>	

Fig. 6 – Exibição da foto, tamanhos e preço da Hering verde.

- b) Os itens da coluna “Cor” devem virar links. Ao selecionar um dos links, será chamada uma função javascript que mostrará na terceira coluna a foto e o preço, como pode ser observado na figura 6. **[1 ponto]**

DICA: Use o método `getElementById` para obter o objeto associado a uma determinada tag. Use o campo `innerHTML` para modificar o conteúdo de uma tag Container.

3. Modifique a página aberta com a seleção do link “Continuar” na página de compras (questão 6 da AD1) e crie uma função javascript que verifique a validade dos números de telefone (fixo e celular) digitados 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. As verificações são:

- a) Se o usuário não digitou 8 ou 9 caracteres dependendo do campo que ativou a função (telefone fixo tem 8 dígitos e celular tem 9 dígitos) como mostrado nas figuras 7 e 8. **[1 ponto]**

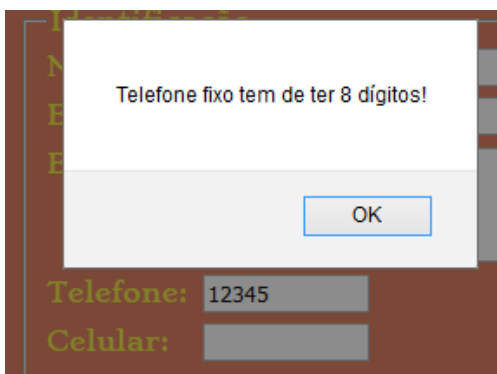


Fig. 7 – Erro ao digitar um telefone fixo com menos (ou mais) de 8 dígitos

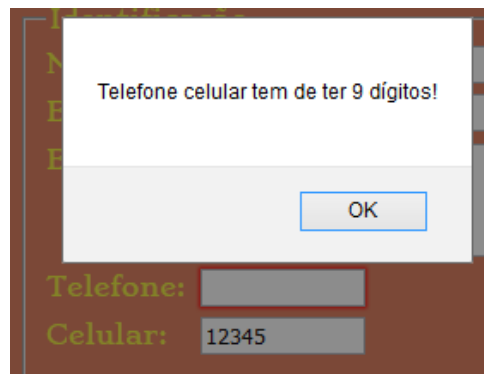


Fig. 8 – Erro ao digitar um celular com menos (ou mais) de 9 dígitos

- b) Se todos os caracteres digitados são dígitos, ou seja, se são caracteres entre 0 e 9 (figura 9). **[1 ponto]**

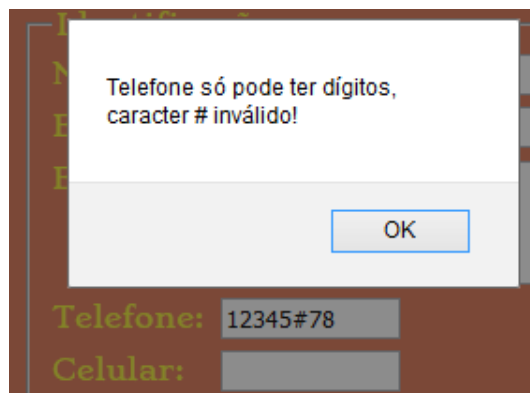


Fig. 9 - Erro ao digitar um telefone (fixo ou celular) com caracteres que não são dígitos.

Caso o valor seja válido, o usuário pode passar para o próximo campo do formulário sem que nenhuma mensagem seja emitida (Figura 10).

Endereço:

Telefone:

Celular:

Genero: ☒ Mas ☐ Fem ☐ Outro

Nascimento: / /

Fig. 10 - Campos de telefone (fixo ou celular) que não geraram mensagem de erro.

4. Modifique a página aberta com a seleção do link “Compras” (questão 5 da AD1) para que o usuário possa selecionar produtos para montar sua lista de compras e saber quanto vai pagar no total (figura 20). A página deve funcionar da seguinte maneira:

- i. Primeiro o usuário deve selecionar um produto (fig. 11):

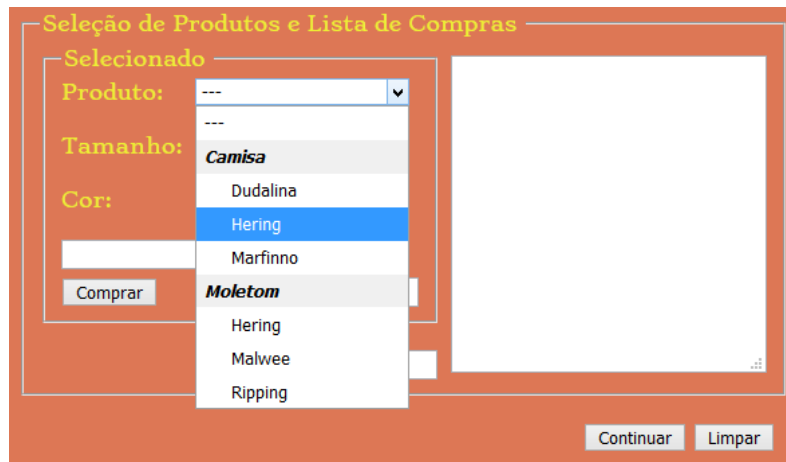


Fig. 11 – Selecionando uma camiseta Hering na lista de produtos.

- ii. Se o usuário tentar selecionar um tamanho ou uma cor antes de selecionar o produto a página deve emitir uma mensagem de erro e apagar a seleção (fig. 12).

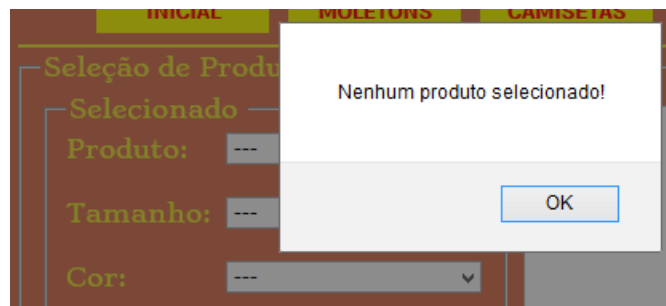


Fig. 12 – Erro de tentar selecionar tamanho ou cor antes de produto.

- iii. Depois de selecionado um produto (tipo e marca) na primeira lista, o usuário pode selecionar um tamanho (fig. 13) ou uma cor (fig. 14). Caso tamanho ou a cor não estejam disponíveis no estoque para o produto selecionado (tipo e marca) uma mensagem de erro deve ser exibida conforme é mostrado nas figuras 15 (tamanho não disponível) e 16 (cor não disponível).



Fig. 13 – Seleção de tamanho



Fig. 14 – Seleção de cor

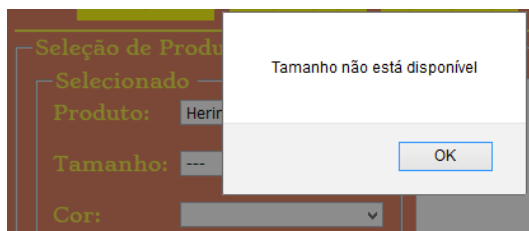


Fig. 15 – Tamanho não disponível para o produto selecionado

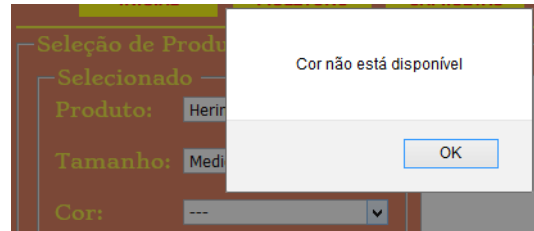


Fig. 16 – Cor não disponível para o produto selecionado

- iv. Após um tamanho válido e uma cor válida terem sido selecionados, deve ser mostrada a descrição completa do produto e seu preço unitário (figura 17).



Fig. 17 – Apresentação da descrição e do preço do produto.

- v. Se o usuário selecionar o botão comprar esta descrição deve ser incluída na lista de compras (à direita da janela) e o preço somado ao campo total (figura 18).

The screenshot shows a window titled "Seleção de Produtos e Lista de Compras". On the left, under the "Selecionado" section, there are three dropdown menus for "Produto:", "Tamanho:", and "Cor:". Below these is a text input field containing "Camisa Hering M Verde". A "Comprar" button is next to a price field showing "R\$ 70". At the bottom, a "Total R\$" field also shows "70". On the right, a list box contains the text "Camisa Hering M Verde".

Fig. 18 – Compra de um produto.

- vi. Se o usuário selecionar um novo produto (figura 19) e selecionar o botão comprar, a descrição do novo produto deve ser incluída no final da lista de compras e o preço somado ao campo total (figura 20)

The screenshot shows the same window as Fig. 18, but with different selections. The "Produto:" dropdown is set to "Ripping", "Tamanho:" to "Medio", and "Cor:" to "Azul". The text input field now contains "Moleton Ripping M Azul". The "Comprar" button is next to a price field showing "R\$ 80". The "Total R\$" field still shows "70". The list box on the right still contains "Camisa Hering M Verde".

Fig. 19 – Seleção de um novo produto.

The screenshot shows the window after purchasing the second product. The "Produto:" dropdown is empty, "Tamanho:" is empty, and "Cor:" is empty. The text input field is empty. The "Comprar" button is next to a price field showing "R\$". The "Total R\$" field now shows "150". The list box on the right now contains two items: "Camisa Hering M Verde" and "Moleton Ripping M Azul".

Fig. 20 – Compra de um novo produto.

Para implementar este comportamento acrescente os seguintes códigos à página:

- I. Crie as variáveis e estruturas de dados que serão utilizadas na página. A descrição do que representa cada variável e estrutura está nos comentários.

```
/*
 * variáveis globais à página:
 *      produto selecionado, tamanho e cor.
 * Armazenam, respectivamente, os índices das
 * tabelas: tabProdutos, tabTamanhos e tabCores.
 * São inicializadas com -1 (nada selecionado)
 */
var prodSel, tamSel, corSel;

prodSel = tamSel = corSel = -1;

/*
 * Tabelas de tipos de produto, marcas, cores e tamanhos
 */
var tabTipos = [ "Camisa", "Moleton" ];

var tabMarcas = [ "Dudalina", "Hering", "Malwee",
                  "Marfinno", "Ripping" ];

var tabCores = [ "Azul", "Branco", "Chumbo",
                  "Cinza", "Preto", "Verde" ];

var tabTamanhos = [ "XP", "P", "M", "G", "XG", "XXG" ];

/* Tabela de produtos: Cada entrada armazena tipo, marca,
 * lista de cores, lista de tamanhos e preço de um produto
 */
var tabProdutos = [
[ 0, 0, [ 1 ],          [ 1, 2, 3, 4 ],          100],
[ 0, 1, [ 0, 2, 5 ],    [ 1, 2, 3, 4, 5 ],        70],
[ 0, 3, [ 0, 1 ],       [ 1, 2, 3, 4 ],          30],
[ 1, 1, [ 3, 4 ],       [ 1, 2, 3, 4 ],          70],
[ 1, 2, [ 3 ],          [ 2, 3, 4 ],             40],
[ 1, 4, [ 0, 3, 4 ],    [ 0, 1, 2, 3, 4 ],        80] ];
```

- II. Defina as seguintes funções que serão chamadas quando um item novo for selecionado nas listas de produtos, tamanhos e cores.

```
/*
 * Função chamada quando um item é
 * selecionado na lista de produtos
 */
function selecProd(selec) {
    var f = selec.form;
    var nl = selec.selectedIndex - 1;
```

```

    /* preenche var global prodSel */
    prodSel = nl;

    /* limpa descrição, preço, tamanho e cor */
    f.txtDesc.value = "";
    f.TxtPreco.value = "";
    f.selTamanho.selectedIndex = -1;
    f.selCor.selectedIndex = -1;
}

/*
 * Função chamada quando um item é
 * selecionado na lista de tamanhos
 */
function selecTam(selec) {
    /* preenche var global tamSel */
    /* 3 é tamanho */
    tamSel = verificaCampo(selec, 3, "Tamanho");
    mostraDesc(selec.form);
}

/*
 * Função chamada quando um item é
 * selecionado na lista de cores
 */
function selecCor(selec) {
    /* preenche var global corSel */
    /* 2 é cor */
    corSel = verificaCampo(selec, 2, "Cor");
    mostraDesc(selec.form);
}

```

III. Modifique as respectivas tags do formulário para incluir as chamadas a estas funções:

```

<Select name="selProduto" class="colSelProd"
        onChange="selecProd(this);">
[...]
<Select name="selTamanho" class="colSelProd"
        onChange="selecTam(this);">
[...]
<Select name="selCor" class="colSelProd"
        onChange="selecCor(this);">
[...]
<input class="elemEsq" type="button" value="Comprar"
        onClick="CompraItem(this.form);"/>

```

Após estas inclusões ficou faltando definir o código de quatro funções:

- a) Defina o código da função `prodTem` que verifica se uma cor ou tamanho está disponível para um produto indicado. Esta função recebe três parâmetros: o índice de um produto na tabela de produtos, a característica procurada (valor 2 para cor e 3 para tamanho) e o índice (de `tabCores` ou `tabTamanhos`) da característica sendo pesquisada. Esta função retorna o valor deste terceiro parâmetro se o produto tem a característica pesquisada (cor ou tamanho) e -1 caso contrário. **[1 ponto]**

Por exemplo:

A chamada `prodTem(1, 2, 3)` verifica se as camisetas Hering (primeiro parâmetro) possuem a cor (segundo parâmetro) azul (terceiro parâmetro). O resultado será -1.

A chamada `prodTem(4, 3, 2)` verifica se os moletons Malwee (primeiro parâmetro) possuem o tamanho (segundo parâmetro) M (terceiro parâmetro). O resultado será 2 (valor do terceiro parâmetro) pois este produto possui este tamanho.

- b) A função `verificaCampo` é chamada pela seleção de uma cor ou tamanho. Esta função faz três coisas: verifica se um produto já foi selecionado na lista de produtos (caso contrário emite uma mensagem de erro), chama a função `prodTem` e caso `prodTem` tenha retornado -1 emite uma mensagem de erro. Esta função retorna o valor retornado por `prodTem`. Complete o código nos locais comentados. **[1 ponto]**

```
function verificaCampo(selec, campo, nome) {
    var ind = selec.selectedIndex - 1;

    /* verifica se algum produto foi selecionado, emite mensagem e
     * limpa o campo caso não tenha sido selecionado
     */

    var prdTem = prodTem(prodSel, campo, ind);

    /* Se prodTem retornou -1 emite mensagem
     * correspondente e limpa o campo
     */

    return prdTem;
}
```

- c) A função `mostraDesc` é chamada pelas funções `selecTam` e `selecCor`, e recebe como parâmetro o objeto formulário. Caso a cor e o tamanho já tenham sido definidos pelo usuário, utiliza as tabelas `tabProdutos`, `tabTipos`, `tabMarcas`, `tabTamanhos` e `tabCores` para preencher os campos de descrição e preço unitário de um produto. Escreva o código desta função. **[1 ponto]**
- d) A função `CompraItem` é chamada quando o botão de compra é acionado. Esta função verifica se algum produto foi selecionado, emitindo uma mensagem de erro caso não tenha sido. Se um produto foi selecionado acrescenta sua descrição à lista de pedidos (pulando uma linha) e soma o preço unitário ao valor total da compra. Depois limpa os campos para uma nova seleção de produtos (figura 20). Escreva o código desta função. **[1 ponto]**

DICA: Não esqueça que a soma de string com número é uma concatenação de strings.

DICA: Para mudar de linha no textarea use um caractere de mudança de linha ‘\n’.