



الفصل الخامس: التفاعل بين JavaScript وHTML

الصفحة	العنوان
3	1. التفاعل بين JavaScript و HTML
4	1.1 التعامل مع عناصر HTML من خلال JavaScript
7	2. الأحداث Events
8	1.2 الأحداث Events
9	2.2 معالجة أحداث جسم الوثيقة body
10	3.2 معالجة أحداث الزر
11	4.2 معالجة أحداث صندوق النص وصندوق كلمة السر
14	3. التحقق من الإدخال
15	1.3 التحقق من الإدخال
20	4. مثال تطبيقي

الكلمات المفتاحية

نموذج كائن الوثيقة، الأحداث.

الملخص

نستعرض في هذا الفصل آليات التعامل مع عناصر XHTML من خلال تعليمات JavaScript. كما نعرض لبعض آليات التحقق من القيم المدخلة من قبل المستخدم.

الأهداف التعليمية

يتعرف الطالب في هذا الفصل على:

- نموذج كائن الوثيقة.
- الأحداث.
- أحداث جسم الوثيقة.
- أحداث الأزرار.
- أحداث مربعات النص.
- التركيز.

المخطط

يضم فصل التفاعل بين JavaScript و HTML 4 وحدات (Learning Objects) هي:

- التفاعل بين JavaScript و HTML
- الأحداث Events
- التحقق من الإدخال
- مثال تطبيقي

التفاعل بين HTML و JavaScript

الأهداف التعليمية

- الوصول لعناصر HTML من خلال Java Script

التعامل مع عناصر HTML من خلال JavaScript

يُمكن الوصول لعناصر HTML من خلال تعليمات JavaScript بعدة طرق:

- نموذج كائن الوثيقة (DOM (Document Object Model
- استخدام أسماء العناصر
- استخدام الطريقة getElementById
- استخدام الطريقة getElementsByName

نموذج كائن الوثيقة (DOM (Document Object Model

- تُكون العناصر elements في هذا النموذج أغراض objects، وتكون واصفاتها attributes خصائص properties.
- يحوي الكائن document المصفوفة forms[] لكل النماذج المحتواة في الوثيقة والمحددة باستخدام المؤثر <form>.
- يحوي كل عنصر من المصفوفة السابقة على مصفوفة elements[] للوصول لجميع عناصر النموذج.
- بالتالي يُمكن الوصول لأي عنصر في الوثيقة باستخدام: document.forms[i].elements[j]

مثال:

يُبين المثال التالي استخدام هذه الطريقة في العنونة:

```
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> Access to form elements </title>
  </head>
  <body>
    <form action = "">
      <input type="button" name="turnItOn" />
    </form>
    <script type="text/javascript" >
      document.forms[0].elements[0].value="on";
    </script>
  </body>
</html>
```

من الواضح أن مشكلة هذه الطريقة في العنونة تكون عند إضافة أو حذف عناصر جديدة. فمثلاً، لو أضفنا زر جديد قبل الزر السابق سيُصبح العنوان السابق عنوان الزر الجديد.

استخدام أسماء العناصر

مثال:

في المثال السابق يُمكن كتابة:

```
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> Access to form elements </title>
  <body>
    <form name="myForm" action = "">
      <input type="button" name="turnItOn" />
    </form>
    <script "type=text/javascript" >
      myForm.turnItOn.value="on";
    </script>
  </body>
</html>
```

تُكمن المشكلة في هذه الطريقة أن المعيار XHTML لا يُسمح بإعطاء الوصفة name للنموذج (إلا أنه يُسمح بإعطائها لعناصر النموذج).

استخدام الطريقة getElementById

مثال:

يُمكن الوصول إلى عنصر عن طريق معرفه (يتم إعطاء معرف فريد للعنصر باستخدام الوصفة id). ويصبح المثال السابق في هذه الحالة:

```
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> Access to form elements </title>
  <body>
    <form action = "">
      <input type="button" id="turnItOn" />
    </form>
    <script "type=text/javascript" >
      document.getElementById("turnItOn").value="on";
    </script>
  </body>
</html>
```

نضع في غالب الأحيان لعناصر النموذج اسم name ومُعَرَّف id¹ (نضع نفس القيمة عادةً).

نقوم عادةً بإعطاء نفس الاسم لمجموعة من أزرار الخيار أو مجموعة من صناديق التحقق، وعندما يكون هذا الاسم مصفوفة يُمكن من خلالها الوصول لكل عنصر في المجموعة².

1 سنرى لاحقاً أن البرمجة من جهة المخدم تتعامل مع المعرف

2 بالطبع، يُمكن إعطاء معرف لكل عنصر في المجموعة إلا أن ذلك لن يكون عملياً للوصول لعناصر المجموعة

استخدام الطريقة getElementById

مثلاً لو كان لدينا مجموعة من صناديق التحكم:

```
<div id = "VehicleGroup">
<input type="checkbox" name="vehicles" value="car" /> Car
<input type="checkbox" name="vehicles" value="truck"/> Truck
<input type="checkbox" name="vehicles" value="bike"/> Bike
</div>
```

فيمكن الوصول لعناصرها كما يلي (لنحسب مثلاً عدد الصناديق المختارة):

```
var numChecked = 0;
var dom = document.getElementById("VehicleGroup");
for (index = 0; index < dom.vehicles.length; index++)
    if (dom.vehicles[index].checked)
        numChecked++;
```

استخدام الطريقة getElementsByName

يُمكن الوصول إلى مجموعة من العناصر التي لها نفس الاسم باستخدام الطريقة getElementsByName كما يُبين المثال التالي:

```
var numChecked = 0;
var dom = document.getElementsByName("vehicles");
for (index = 0; index < dom.length; index++)
    if (dom[index].checked)
        numChecked++;
```

الأحداث Events

الأهداف التعليمية

- التعرف على الأحداث

الأحداث Events

- يُخبر الحدث بأن شيء محدد قد حصل بفعل المتصفح أو بفعل المستخدم.
 - ندعو معالج الحدث Event handler الخطاطة التي تُنفذ عند وقوع الحدث.
 - ندعو عملية ربط حدث بمعالج حدث بالتسجيل Registration.
- يُبين الجدول التالي الأحداث والواصفات الموافقة والمؤثرات التي تتحسس هذه الأحداث:

الحدث	Event	Tag Attribute	Tag
ترك العنصر	blur	onblur	<a>
			<button>
			<input>
			<textarea>
			<select>
تغيير القيمة	change	onchange	<input>
			<textarea>
			<select>
النقر	click	onclick	<a>
			<input>
انتقال الفأرة إلى العنصر	focus	onfocus	<a>
			<input>
			<textarea>
			<select>
التحميل	load	onload	<body>
ضغط زر الفأرة	mousedown	onmousedown	Most elements
تحريك الفأرة	mousemove	onmousemove	Most elements
مغادرة الفأرة العنصر	mouseout	onmouseout	Most elements
توضع الفأرة فوق العنصر	mouseover	onmouseover	Most elements
تحرير زر الفأرة	mouseup	onmouseup	Most elements
تحديد العنصر	select	onselect	<input>
			<textarea>
الإرسال	submit	onsubmit	<form>
إنهاء التحميل	unload	onunload	<body>

يتم تسجيل معالج الحدث:

- إما بإسناد التعليمات لوصفة المؤثر:

```
<input type="button" name="myButton" id="myButton"
      onclick="alert('You clicked my button!');" />
```

- أو بكتابة تعليمات الحدث في وظيفة ومن ثم استدعاء الوظيفة في واصفة المؤثر:

```
<input type="button" name="myButton" id="myButton"
      onclick="myHandler( );" />
```

- أو بإسناد اسم الوظيفة في JavaScript إلى خاصية الحدث:

```
document.getElementById("myButton").onclick = myHandler;
```

معالجة أحداث جسم الوثيقة body

يُعتبر حدثي التحميل load وإنهاء التحميل unload من أهم أحداث المؤثر <body>. يُبين المثال التالي استخدام حدث التحميل لإظهار رسالة ترحيبية:

```
<!DOCTYPE html >
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> onLoad event handler </title>
    <script type = "text/javascript">
      <!--
      // The onload event handler
      function load_greeting () {
        alert("You are visiting the home page of \n" +
              "SVU \n" +
              "WELCOME!!!");
      }
      // -->
    </script>
  </head>
  <body onload="load_greeting();" >
    <p />
  </body>
</html>
```

حيث يُظهر المتصفح حال فتح الوثيقة:



معالجة أحداث الزر

يُعتبر الحدث onclick الأكثر استخداماً مع زر الأمر. ويتم تسجيل الحدث عادةً:

- إما بإسناد اسم الوظيفة إلى الوصفة onclick:

```
<input type="button" name="freeButton"
      id="freeButton" onclick="freeButtonHandler();">
```

- أو باستخدام:

```
document.getElementById("freeButton").onclick = freeButtonHandler();
```

يُبين المثال التالي استخدام الحدث onclick مع مجموعة من أزرار الخيار لإظهار رسالة موافقة عندما يقوم المستخدم بالنقر على زر خيار من المجموعة:

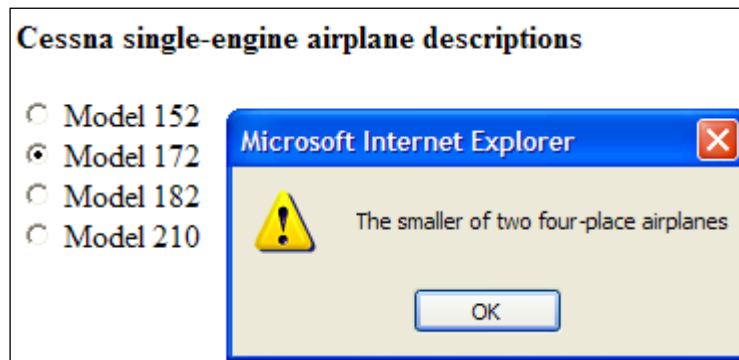
```
<!DOCTYPE html >
< An example of the use of the click event with radio buttons,
registering the event handler by assignment to the button attributes
-->
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> Illustrate messages for radio buttons </title>
    <script type = "text/javascript">
      <!--
      // The event handler for a radio button collection
      function planeChoice (plane) {
      // Produce an alert message about the chosen airplane
      switch (plane) {
        case 152:
          alert("A small two-place airplane for flight training");
          break;
        case 172:
          alert("The smaller of two four-place airplanes");
          break;
        case 182:
          alert("The larger of two four-place airplanes");
          break;
        case 210:
          alert("A six-place high-performance airplane");
          break;
        default:
          alert("Error in JavaScript function planeChoice");
          break;
      }
      }
      // -->
    </script>
  </head>
  <body>
    <h4> Cessna single-engine airplane descriptions </h4>
    <form id = "myForm" action = "handler">
      <p>
        <input type = "radio" name = "planeButton"
          value = "152" onclick = "planeChoice(152)" /> Model 152
        <br />
        <input type = "radio" name = "planeButton"
```

```

        value = "172" onclick = "planeChoice(172)" />
                                Model 172    <br />
<input type = "radio" name = "planeButton"
    value = "182" onclick = "planeChoice(182)" /> Model 182
<br />
<input type = "radio" name = "planeButton"
    value = "210" onclick = "planeChoice(210)" /> Model 210
</p>
</form>
</body>
</html>

```

عند النقر على أحد عناصر مجموعة أزرار الخيار تظهر الرسالة الموافقة:



معالجة أحداث صندوق النص وصندوق كلمة السر

يُمكن لصندوق النص إنشاء أربعة أحداث مختلفة:

- blur
- focus
- change
- select

حدث التركيز Focus

يُبين المثال التالي استخدام الحدث onfocus وذلك لمنع المستخدم من الكتابة في صندوق نص وذلك بإخراج التركيز من صندوق النص فور حدوثه (وذلك باستخدام الطريقة blur). يسمح هذا المثال للمستخدم بإدخال الكمية المطلوبة من كل منتج، وعند النقر على الزر "Total Cost" يظهر الإجمالي في صندوق النص الموافق. لن يتمكن المستخدم من الكتابة في صندوق النص المخصص للإجمالي إذ سنقوم بإخراج التركيز عن هذا الصندوق كلما قام المستخدم بالانتقال إليه.

```

<!DOCTYPE html >
<!-- This document illustrates using the focus event
      to prevent the user from changing a text field
-->
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head> <title> The focus event </title>
    <script type = "text/javascript">
      <!--
// The event handler function to compute the cost
function computeCost() {
var french = document.getElementById("french").value;
var hazlenut = document.getElementById("hazlenut").value;
var columbian = document.getElementById("columbian").value;
// Compute the cost
document.getElementById("cost").value =
totalCost = french * 3.49 + hazlenut * 3.95 +
              columbian * 4.59;
}  /* end of computeCost
-->
</script>
</head>
<body>
  <form action = "">
    <h3> Coffee Order Form </h3>
    <!-- A bordered table for item orders -->
    <table border = "border">
    <!-- First, the column headings -->
      <tr>
        <th> Product Name </th>
        <th> Price </th>
        <th> Quantity </th>
      </tr>
    <!-- Now, the table data entries -->
      <tr>
        <th> French Vanilla (1 lb.) </th>
        <td> $3.49 </td>
        <td> <input type = "text" id = "french"
                      size = "2" /> </td>
      </tr>
      <tr>
        <th> Hazlenut Cream (1 lb.) </th>
        <td> $3.95 </td>
        <td> <input type = "text" id = "hazlenut"
                      size = "2" /> </td>
      </tr>
      <tr>
        <th> Columbian (1 lb.) </th>
        <td> $4.59 </td>
        <td> <input type = "text" id = "columbian"
                      size = "2" /></td>
      </tr>
    </table>
    <!-- Button for precomputation of the total cost -->
    <p>
      <input type = "button" value = "Total Cost"
            onclick = "computeCost();" />
      <input type = "text" size = "5" id = "cost"
            onfocus = "this.blur();" />
    </p>
  </form>
</body>
</html>

```

```

<!-- The submit and reset buttons -->
<p>
    <input type = "submit" value = "Submit Order" />
    <input type = "reset" value = "Clear Order Form" />
</p>
</form>
</body>
</html>

```

Coffee Order Form

Product Name	Price	Quantity
French Vanilla (1 lb.)	\$3.49	<input type="text" value="2"/>
Hazlenut Cream (1 lb.)	\$3.95	<input type="text" value="1"/>
Columbian (1 lb.)	\$4.59	<input type="text" value="1"/>

Total Cost

15.52

Submit Order

Clear Order Form

التحقق من الإدخال

الأهداف التعليمية

- التحقق من القيم المدخلة من قبل المستخدم.

التحقق من الإدخال

تُعتبر عمليات التحقق من الإدخالات التي يقوم بها المستخدم من العمليات الأساسية التي يُحذ القيام بها على الزبون باستخدام JavaScript وذلك تلافياً للقيام بهذه التحققات على المخدم مما يوفر زمن معالجة على المخدم وزمن ترأسل على الشبكة. يجب عادة القيام بالخطوات الثلاثة التالية:

- اكتشاف الخطأ وإظهار رسالة alert موافقة.
- وضع التركيز على العنصر منشأ الخطأ باستخدام الطريقة ().focus()
- تحديد محتوى العنصر باستخدام الطريقة ().select() (ليقوم المستخدم بتعديله مباشرة).

مثال 1:

يقوم المثال التالي بإظهار صندوق نص لإدخال كلمة السر وتأكيد كلمة السر. يقوم الزر submit باستدعاء معالج الحدث. يقوم معالج الحدث أولاً بالتأكد من أن صندوق كلمة السر غير فارغ، وفي هذه الحالة يقوم بإظهار رسالة خطأ موافقة ومن ثم يضع التركيز على الصندوق. أما إذا كان صندوق كلمة السر غير فارغ، فيقوم بالتحقق من تطابق كلمة السر وتأكيد كلمة السر، وفي حال عدم تطابقهما، يقوم بإظهار رسالة خطأ موافقة ومن ثم يقوم بوضع التركيز على كلمة السر وتحديد القيمة المدخلة.

```
<!DOCTYPE html >
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> Illustrate password checking> </title>
    <script type = "text/javascript">
      <!--
        function chkPasswords() {
var init=document.getElementById("initial");
var sec=document.getElementById("second");
if (init.value==""){
  alert("You did not enter a password");
  init.focus();
  return false;
}
if (init.value != sec.value) {
  alert("The two passwords are not the same");
  init.focus();
  init.select();
  return false;
} else
return true;
        }
      // -->
    </script>
  </head>
  <body>
    <h3> Password Input </h3>
    <form id = "myForm"  action = "">
      <p>
        Your password
        <input type = "password"  id = "initial"  size = "10" />
        <br /><br />
        Verify password
        <input type = "password"  id = "second"  size = "10" />
      </p>
    </form>
  </body>
</html>
```



```

    <br /><br />
    <input type = "reset" name = "reset" />
    <input type = "submit" name = "submit" />
  </p>
</form>
<script type = "text/javascript">
  <!--
  // Set submit button onsubmit property to the event handler
  document.getElementById("second").onblur = chkPasswords;
  document.getElementById("myForm").onsubmit=chkPasswords;
  // -->
</script>
</body> </html>

```

مثال 2:

- يقوم المثال التالي من التحقق من القيمة المدخلة لصندوق الاسم ومن القيمة المدخلة لصندوق الهاتف.
- تقوم الوظيفة `chkName` من التحقق من أن الاسم المدخل له الشكل: Last Name, First Name, Middle initial.

وذلك عن طريق التحقق من مطابقة القيمة المدخلة للنموذج `Pattern`:

`^[A-Z][a-z]+, ?[A-Z][a-z]+, ?[A-Z]\.?$`

- كذلك تقوم الوظيفة `chkPhone` من التحقق من مطابقة القيمة المدخلة في صندوق الهاتف مع النموذج: `d{3}-\d{3}-\d{4}`.
- يتم تسجيل الوظيفة مع حدث `onchange` لصندوق النص الموافق. عند إدخال قيمة في صندوق لا تتطابق مع النموذج، يتم إظهار رسالة تنبيه موافقة ومن ثم وضع التركيز على الصندوق وتحديد القيمة المدخلة.

```

<!DOCTYPE html >

<!-- An example of input validation using the change and submit
      events
      -->

<html xmlns = "http://www.w3.org/1999/xhtml">

  <head>

```

```

<title> Illustrate form input validation> </title>

<script type = "text/javascript">

    <!--
// The event handler function for the name text box

    function chkName() {

        var myName = document.getElementById("custName");

// Test the format of the input name
// Allow the spaces after the commas to be optional
// Allow the period after the initial to be optional

        var pos = myName.value.search(

            /^[A-Z][a-z]+, ?[A-Z][a-z]+, ?[A-Z]\.?$ /);

        if (pos != 0) {

            alert("The name you entered (" + myName.value +

                ") is not in the correct form. \n" +

                "The correct form is: " +

                "last-name, first-name, middle-initial \n" +

                "Please go back and fix your name");

            myName.focus();

            myName.select();

            return false;

        } else

            return true;

    }

// The event handler function for the phone number text box

    function chkPhone() {

        var myPhone = document.getElementById("phone");

// Test the format of the input phone number

        var pos = myPhone.value.search(/^\d{3}-\d{3}-\d{4}$/);

        if (pos != 0) {

            alert("The phone number you entered (" + myPhone.value +

                ") is not in the correct form. \n" +

                "The correct form is: ddd-ddd-dddd \n" +

                "Please go back and fix your phone number");

            myPhone.focus();

            myPhone.select();

```

```

        return false;
    } else
        return true;
    }

    // -->
</script>
</head>
<body>
    <h3> Customer Information </h3>
    <form action = "">
        <p>
            <input type = "text" id = "custName"
                onchange = "chkName();" />

            Name (last name, first name, middle initial)
            <br /><br />
            <input type = "text" id = "phone" />

            Phone number (ddd-ddd-dddd)
            <br /><br />
            <input type = "reset" id = "reset" />
<input type = "submit" id = "submit" />
        </p>
    </form>
    <script type = "text/javascript">
        <!--
// Set form element object properties to their
// corresponding event handler functions
document.getElementById("custName").onchange = chkName;
document.getElementById("phone").onchange = chkPhone;

        // -->
    </script>
</body>
</html>

```

Customer Information	
<input type="text" value="Alkhatib, Bassel, M."/>	Name (last name, first name, middle initial)
<input type="text" value="999-555-3333"/>	Phone number (ddd-ddd-dddd)
<input type="button" value="Reset"/>	<input type="button" value="Submit Query"/>

مثال تطبيقي

الأهداف التعليمية

- مراجعة مفاهيم الفصل.

تهدف الصفحة التالية إلى حساب سعر الإقامة في فندق وذلك وفق مجموعة من الخيارات التي يُمكن أن يقوم بها المستخدم:

- تصنيف الفندق: *, **, ***, ****, *****
 - إطلالة الفندق: البحر، الغابة، الجبل.
 - الوجبات المطلوبة: فطور، غداء، عشاء.
 - خدمات إضافية: تلفزيون، مسبح، مطبخ.
- ولتكن مثلاً الأسعار التالية المعتمدة:

Stars		Where		Food		Extra	
Stars	Price	Sea	+50%	Breakfast	50	TV	50
*	100	Forest	+30%	Lunch	100	Pool	100
**	200	Mountain	+20%	Dinner	200	Kitchen	150
***	300						
****	400						
*****	500						

يُمكن في هذه المسألة البسيطة تثبيت الأسعار في قيم عناصر HTML للرجوع إليها في خطاطة Java المستخدمة لحساب سعر الليلة:

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head> <title>Hotel Club HTML</title>
<script language="javascript" type="text/javascript">
function ChangeFlag()
{
document.getElementById("imgFlag").src=
document.getElementById("SelectFlag").value
}

function getPrice()
{
var x;
x=document.getElementById("SelectStars").value;

if (document.getElementById("RadWhere1").checked)
x= x* document.getElementById("RadWhere1").value;
else if (document.getElementById("RadWhere2").checked)
x= x* document.getElementById("RadWhere2").value;
else if (document.getElementById("RadWhere3").checked)
x= x* document.getElementById("RadWhere3").value;

if (document.getElementById("chkBreak").checked)
x = x + parseInt( document.getElementById("chkBreak").value);

if (document.getElementById("chkLunch").checked)
x = x + parseInt( document.getElementById("chkLunch").value);
}
```

```

if (document.getElementById("chkDinner").checked)
    x = x + parseInt( document.getElementById("chkDinner").value);

var i;
for (i=0; i<document.getElementById("SelectExtra").length; i++)
    if (document.getElementById("SelectExtra")[i].selected)
        x= x + parseInt(document.getElementById("SelectExtra")[i].value);

document.getElementById("txtOut").value=x;
}
</script>

</head>
<body bgcolor="#ffffcc">
<form id="frmHotel" name="frmHotel" action="">
<table border="1">
<caption>Hotel Club (HTML Controls/ Client)</caption>
<tr>
    <td> Country:</td>
    <td>
        <select name="SelectFlag" id="SelectFlag" onchange="ChangeFlag()" >
            <option value="images\USA.PNG">USA</option>
            <option value="images\FRANCE.PNG">France</option>
            <option value="images\ITALY.PNG">Italy</option>
            <option value="images\GERMANY.PNG">Germany</option></select></td>
    <td> </td></tr>
<tr>
    <td> Stars:</td>
    <td>
        <select id="SelectStars" name="SelectStars">
            <option value="100">*</option>
            <option value="200">**</option>
            <option value="300">***</option>
            <option value="400">****</option>
            <option value="500">*****</option>
        </select></td><td></td></tr>
<tr>
    <td>Where:</td>
    <td>
        <input id="RadWhere1" name="RadWhere" checked type="radio" value="1.5"
/> Sea<br />
        <input id="RadWhere2" type="radio" name="RadWhere" title="Forest"
value="1.3" /> Forest<br />
        <input id="RadWhere3" type="radio" name="RadWhere" title="Mountain"
value="1.2" /> Mountain<br />
    </td><td></td></tr>
<tr>
    <td>Meals:</td>
    <td>
        <input id="chkBreak" name="chkBreak" type="checkbox" title="Breakfast"
value="50" />Breakfast<br />
        <input id="chkLunch" name="chkLunch" type="checkbox" title="Lunch"
value="100" />Lunch<br />
        <input id="chkDinner" name="chkDinner" type="checkbox" title="Dinner"
checked="CHECKED" value="200" />Dinner</td>
    <td></td></tr>
<tr>
    <td>Extra:</td>
    <td>

```

```

<select id="SelectExtra" multiple name="SelectExtra" dir="ltr"
size="3">
<option selected="selected" value="50">TV</option>
<option value="75">Pool</option>
<option value="100">Kitchen</option>
</select><br /></td><td></td></tr>
<tr><td>
<input id="Submit1" type="button" value="How Much" onclick="getPrice()"
/></td>
<td>
<input id="txtOut" name="txtOut" type="text" />
</td><td></td></tr>
</table></form>
</body> </html>

```

Hotel Club (HTML Controls/ Client)		
Country:	France ▼	
Stars:	*** ▼	
Where:	<input checked="" type="radio"/> Sea <input type="radio"/> Forest <input type="radio"/> Mountain	
Meals:	<input type="checkbox"/> Breakfast <input type="checkbox"/> Lunch <input checked="" type="checkbox"/> Dinner	
Extra:	<div> TV Pool Kitchen </div>	
How Much	800	

- توجد أعلام الدول في المجلد images.
- تكون قيمة كل خيار من قائمة البلدان مسار ملف صورة علم البلد.
- تقوم وظيفة Java المسماة ChangeFlag() بإسناد قيمة الخيار المحدد من قائمة البلاد (أي اسم ملف صورة علم البلد المحدد) إلى الخاصية src للصورة التي يظهر فيها علم البلد.
- يتم استدعاء هذه الوظيفة على الحدث onchange لقائمة البلاد.
- تقوم الوظيفة getPrice() بحساب سعر الإقامة استناداً لخيارات المستخدم. تعود هذه الوظيفة إلى قيمة كل عنصر لمعرفة سعر الخيار المحدد.