TEMAS VISTOS

a) Estructura Html

b) Salto de linea

c) Parrafos

d) Encabezados

e)hipervinculo

a) a una pagina del mismo sitio

f)Enfasis

REPASAR RUTAS RELATIVAS Y ABSOLUTAS

Resolucion Ejercicio de repaso

TEMAS DE HOY

2) etiqueta imagen

4)hipervinculo en otra solapa y en la misma solapa

5) anclas llamadas desde la misma pagina

6)anclas llamadas desde otra pagina

7)lista ordenada

8)listadesordenada

9)listas anidadas

10)tablas

11)tablas con encabezado

12)tablas con titulo

13)combinacion de celdas

14)formulario

15)text

16)check

17)radio

18)select y select multiple

19)button

20)file

21)hidden

22)agrupamiento de controles

2) etiqueta imagen

imágenes dentro de una pagina

Para insertar una imagen dentro de una página debemos utilizar el elemento HTML <img>, la misma no tiene una marca de finalización (similar a la etiqueta <br>).

Generalmente, la imagen se encuentra en el mismo servidor donde se almacenan nuestras páginas HTML. Los formatos clásicos son los archivos con extensiones gif, jpg y png.

La sintaxis de esta etiqueta es:

<img src="foto1.jpg" alt="Pintura geométrica">

Como mínimo, debemos inicializar las propiedades src y alt de la etiqueta HTML "img".

En la propiedad src indicamos el nombre del archivo que contiene la imagen

Otra propiedad muy recomendada es alt, donde disponemos un texto que verán los usuarios que visiten el sitio con un navegador que sólo permite texto (o con un navegador que tenga desactivada la opción de descarga de imágenes). El texto debe describir el contenido de la imagen.

Confeccionemos una página que muestre una imagen llamada foto1.jpg (La imagen se encuentra almacenada en el servidor en la misma carpeta donde se localiza esta página)

Si la imagen se encuentra en una subcarpeta llamada imagenes, luego la sintaxis para recuperarla será:

<img src="imagenes/foto1.jpg" alt="Pintura geométrica">

Es decir, antecedemos al nombre de la imagen el nombre de la carpeta y la barra /

Si la imagen se encuentra en una carpeta padre de donde se encuentra la página HTML luego la sintaxis será:

<img src="../foto1.jpg" alt="Pintura geométrica">

Es decir, le antecedemos .. y la barra / al nombre de la imagen

Si queremos subir dos carpetas luego escribimos:

<img src="../../foto1.jpg" alt="Pintura geométrica">

Por último, si queremos acceder a una imagen que se encuentra en una carpeta llamada imagenes pero que está en una jerarquia superior:

<img src="../imagenes/foto1.jpg" alt="Pintura geométrica">

Primero le indicamos que subimos al directorio padre mediante los dos puntos .. y seguidamente indicamos el nombre de la carpeta y la imagen a mostrar.

**<img>** viene de la palabra **image**

**src** viene de de la palabra **source**

**alt** viene de la palabra **alternative**

4)hipervinculo en otra solapa y en la misma solapa

5)Anclas

Anclas llamadas desde la misma pagina

Otra posibilidad que nos brinda el HTML es disponer una referencia dentro de la página para poder posteriormente disponer un hipervínculo a dicha marca.

Es una práctica común cuanda queremos desplazarnos dentro de una página de gran tamaño. Se disponen hipervínculos a diferentes anclas.

La sintaxis para definir un ancla es:

<a name="nombreancla"></a>

No debemos confundir un ancla con un hipervínculo, más alla que se utiliza el mismo elemento a.

El ancla es el elemento de la pagina donde esta el hipervinculo Para un ancla inicializamos la propiedad name con el nombre del ancla.

Un ancla se la define en una parte de la página que queremos que el operador llegue a partir de un hipervínculo.

Ahora la sintaxis para ir a un ancla desde un hipervínculo es la siguiente:

<a href="#nombreancla">Introducción</a><br>

Vemos que en la propiedad href indicamos el nombre del ancla.

Haremos un ejemplo, donde dispondremos una lista de hipervínculos que llaman a una serie de anclas dispuestas en la misma página:

6)Anclas llamadas desde otra pagina

También es perfectamente válido la llamada a anclas desde otra página (no importa si se encuentra en el mismo sitio o en otro)

Debemos conocer el nombre de la página a llamar y el nombre del ancla, luego la sintaxis para la llamada al ancla es:

<a href="pagina2.html#introduccion">Introducción</a>

Es decir luego del nombre de la página que llamamos disponemos el caracter # y seguidamente el nombre del ancla.

Confeccionemos dos página y que la primera llame a diferentes anclas definidas en la segunda:

7)lista ordenada

Este etiqueta HTML es útil cuando debemos numerar o listar una serie de datos.

Veamos con un ejemplo una lista ordenada para conocer su sintaxis. Mostraremos el orden de llegada de tres corredores:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<ol>

<li>Rodriguez Pablo</li>

<li>Gonzalez Raul</li>

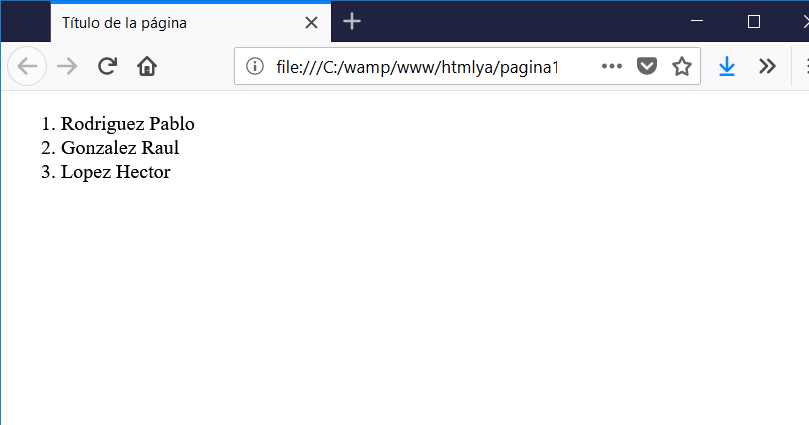
<li>Lopez Hector</li>

</ol>

</body>

</html>

El resultado en el navegador es:



8)Lista Desordenada

Una lista no ordenada como su nombre lo indica no utiliza un número delante de cada items sino un pequeño símbolo gráfico.

La forma de implementar este tipo de listas es idéntico a las listas ordenadas.

Veamos un ejemplo donde implementamos una lista no ordenada:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<h2>Lenguajes de programación.</h2>

<ul>

<li>C</li>

<li>C++</li>

<li>Java</li>

<li>C#</li>

</ul>

</body>

</html>

El resultado en el navegador es:



9) Lista de definiciones

Como su nombre lo indica se utiliza para asociar un término y la definición del mismo. El navegador se encarga de destacar y separa el término y su definición.

Crearemos una lista con la definición de varios lenguajes de programación:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<dl>

<dt>C++</dt>

<dd>Es un lenguaje de programación, diseñado a mediados de

los años 1980, por Bjarne Stroustrup, como extensión del lenguaje

de programación C.</dd>

<dt>Java</dt>

<dd>Es un lenguaje de programación orientado a objetos desarrollado

por Sun Microsystems a principios de los 90.</dd>

<dt>JavaScript</dt>

<dd>Es un lenguaje interpretado, es decir, que no requiere compilación,

utilizado principalmente en páginas web, con una sintaxis semejante a la

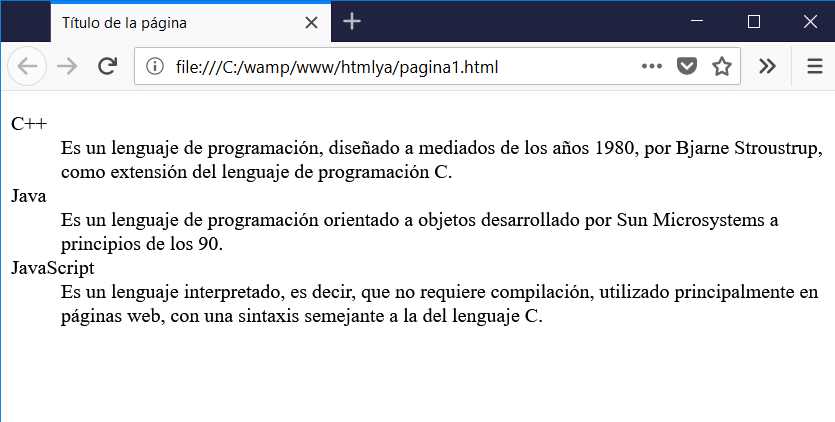
del lenguaje C.</dd>

</dl>

</body>

</html>

El resultado en el navegador es:



Como podemos observar intervienen más etiquetas que en los otros dos tipos de listas. Las marcas que encierran a la lista son <dl> (Definition List) y </dl>

Ahora debemos poner a pares estos dos elementos <dt> (Definition Term) y <dd> (Definition Description)

El navegador se encarga de hacer el sangrado del contenido del elemento dt

10 )listas anidadas

El lenguaje HTML nos permite insertar una lista dentro de otra. Se pueden anidar listas de distinto tipo, por ejemplo podemos tener una lista no ordenada y uno de los item puede ser una lista ordenada.

Para el anidamiento de listas solo debemos tener cuidado en la correcta apertura y cerrado de las marcas

Implementaremos una página que enumere una serie de paises en una lista ordenada y luego en cada pais dispondremos una lista de hipervínculos de periódicos de dicho pais:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<ol>

<li>Argentina

<ul>

<li><a href="http://www.lanacion.com.ar">La Nación</a></li>

<li><a href="http://www.clarin.com.ar">Clarín</a></li>

<li><a href="http://www.pagina12.com.ar">Página 12</a></li>

</ul>

</li>

<li>España

<ul>

<li><a href="http://www.elpais.es">El País Digital</a></li>

<li><a href="http://www.abc.es">ABC</a></li>

<li><a href="http://www.elmundo.es">El Mundo</a></li>

</ul>

</li>

<li>México

<ul>

<li><a href="http://www.jornada.unam.mx">La Jornada</a></li>

<li><a href="http://www.el-universal.com.mx">El Universal</a></li>

</ul>

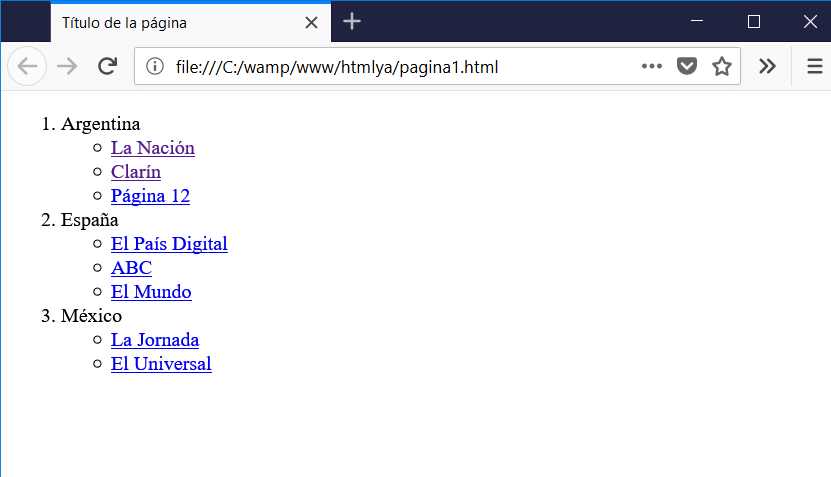
</li>

</ol>

</body>

</html>

El resultado en el navegador es:



10)

Tabla

El objetivo fundamental de las tablas es mostrar una serie de datos en forma ordenada, organizado en filas y columnas.

Algo importante es que no utilizaremos las tablas para organizar la página completa (esto se hace en la actualizad mediante CSS, es decir hojas de estilo. Una vez que avance en este curso puede visitar [CSS Ya](http://www.tutorialesprogramacionya.com/cssya/))

Para la creación de una tabla intervienen una serie de elementos:

<table> Es la marca de comienzo de la tabla. Este elemento requiere la marca de cierre.

<tr> Es la marca de comienzo de una fila. Esta marca debe estar dentro del elemento table. Este elemento requiere la marca de cierre.

<td> Es la marca de comienzo de una celda. Esta marca debe estar dentro del elemento tr. Este elemento requiere la marca de cierre.

Para recordar el nombre de estos elementos HTML:

**<table>**

**<tr>** viene de **table row** que significa fila de la tabla.

**<td>** viene de **table data** que significa dato de la tabla.

La mejor forma de entender y dominar este concepto es implementar tablas y ver su resultado dentro del navegador. Confeccionemos una tabla que muestre los nombre de paises en una columna y su cantidad de habitantes en otra:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<table border="1">

<tr>

<td>China</td><td>1300 millones</td>

</tr>

<tr>

<td>India</td><td>1080 millones</td>

</tr>

<tr>

<td>Estados Unidos</td><td>295 millones</td>

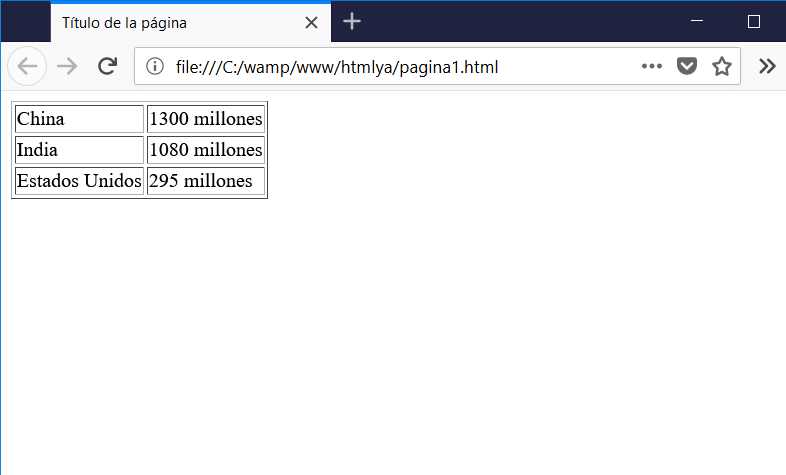
</tr>

</table>

</body>

</html>

El resultado en el navegador es:



Lo primero que aparece es la apertura del elemento table, donde inicializamos la propiedad border con el valor 1, con esto el contorno de cada celda será visible (pruebe de asignarle el valor cero y otros valores superiores a 1)

<table border="1">

Seguidamente viene la apertura de la primer fila de la tabla con el elemento tr:

<tr>

Continuamos con la apertura de la primer celda de la primer fila de la tabla:

<td>

Luego el dato propiamente dicho de la celda:

India

Cerramos la celda:

</td>

Abrimos la proxima celda:

<td>

Disponemos el valor de la celda:

1300 millones

Cerramos la celda:

</td>

Ahora cerramos la primer fila de la tabla:

</tr>

El mecanismos de la siguiente fila es similar a la primera.

Tabla con Encabezado

La primer característica que le vamos a agregar a una tabla son las celdas de encabezado. Normalmente la primer fila de una tabla puede representar los títulos para cada columna.

Para indicar que se trata de una celda de encabezado utilizamos el elemento <th> en lugar de <td>

El navegador representa distinto las celdas de datos y las celdas de encabezamiento.

Confeccionemos el mismo problema del concepto anterior pero agregando un título a cada columna de la tabla mediante celdas de encabezamiento:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<table border="1">

<tr>

<th>Paises</th><th>Cantidad de habitantes</th>

</tr>

<tr>

<td>China</td><td>1300 millones</td>

</tr>

<tr>

<td>India</td><td>1080 millones</td>

</tr>

<tr>

<td>Estados Unidos</td><td>295 millones</td>

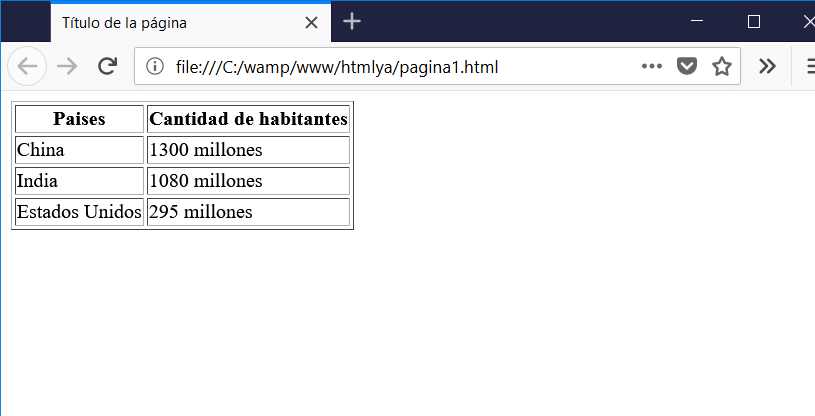
</tr>

</table>

</body>

</html>

resultado



**Tabla con combinacion de celdas**

En algunas situaciones se necesita que una celda ocupe el lugar de dos o más celdas en forma horizonzal o vertical, para estos casos el elemento td o th dispone de dos propiedades llamadas rowspan y colspan.

A estas propiedades se les asigna un valor entero a partir de 2.

Si queremos que una celda ocupe tres columnas luego inicializamos la propiedad colspan con el valor 3:

<td colspan="3">Facturación de los últimos tres meses</td>

Si por el contrario queremos que una celda se extienda a nivel de filas luego hacemos:

<td rowspan="3">Secciones</td>

Veamos un ejemplo empleando el concepto de combinación de celdas:

<!DOCTYPE html>

<html>

<head>

<title>Título de la página</title>

<meta charset="UTF-8">

</head>

<body>

<table border="1">

<tr>

<td rowspan="4">Recursos</td><td colspan="4">Facturación de los últimos tres meses</td>

</tr>

<tr>

<td>Discos Duros</td><td>23000</td><td>27200</td><td>26000</td>

</tr>

<tr>

<td>CPU</td><td>73000</td><td>67300</td><td>51000</td>

</tr>

<tr>

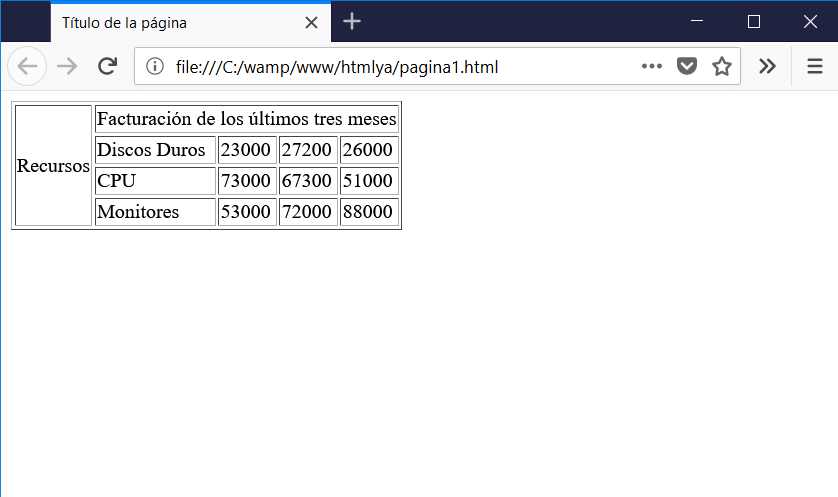
<td>Monitores</td><td>53000</td><td>72000</td><td>88000</td>

</tr>

</table>

</body>

</html>



**Contenido de la cabeza de la pagina**

Hemos definido un elemento meta fundamental donde indicamos el tipo de codificación de caracteres de la página:

<meta charset="UTF-8">

Veamos ahora otras propiedades de la marca meta y los valores más comunes con lo que podemos inicializarlos:

<meta name="nombre de la propiedad" content="valor de la propiedad">

Name almacena el nombre de la propiedad y content el valor de la propiedad.

No existe ninguna especificación de la W3C que defina los valores posibles para el atributo name, si bien existen algunos que son estandares de facto (description, keywords, author etc.)

Veamos las propiedades y valores más comunes

<meta name="keywords" content="html, programación, webmaster">

Los buscadores tienen en cuenta los metadatos. Si en la página inicializamos la propiedad name del elemento meta con el valor keywords luego buscará en la propiedad content las distintas palabras claves más representativas para dicha página. Esto es muy útil para posicionar nuestra página según el contenido que provee.

Veamos otras inicializaciones del elemento meta:

<meta name="description" content="El objetivo de este tutorial

es presentar los conceptos básicos de HTML. Es objetivo prioritario respetar

los estándares del W3C">

En este caso especificamos una descripción de la página, pudiendo ser del sitio si se trata de la página principal.

<meta name="author" content="juan perez ">

<meta name="copyright" content="xx.inc.">

**Formulario**

Un formulario permite que el visitante al sitio cargue datos y sean enviados al servidor.

Es el medio ideal para registrar comentarios del visitante sobre el sitio, solicitar productos, sacar turnos etc.

De todos modos veremos que el lenguaje HTML solo tiene el objetivo de crear el formulario. El HTML no tiene la responsabilidad de registrar los datos en el servidor, esta actividad está delegada a un lenguaje que se ejecute en el servidor (PHP, ASP, ASP.Net, JSP,NodeJS etc)

Si recien comienza en el mundo del desarrollo de páginas web es recomendable primero aprender y conocer todos los elementos para la creación de formularios en HTML y en un paso posterior estudiar la registración en el servidor

Veamos la sintaxis básica para crear un formulario donde ingresemos nuestro nombre.

Para crear un formulario debemos utilizar el elemento form, que tiene marca de comienzo y fin. Dentro de la marca form veremos otros elementos para crear botones, editores de línea, cuadros de chequeo, radios de selección etc.

Confeccionaremos un formulario para el ingreso de nuestro nombre y un botón para el envío del dato ingresado al servidor:

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="20">

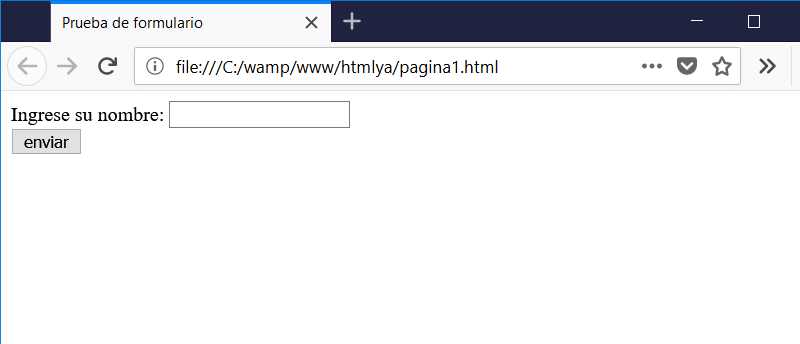
<br>

<input type="submit" value="enviar">

</form>

</body>

</html>



Veamos detenidamente la estructura de un formulario elemental, lo primero la apertura de la marca form donde debemos definir dos propiedades (action y method):

<form action="registrardatos.php" method="post">

La propiedad action se inicializa con el nombre de la página que contiene el programa que procesará los datos en el servidor. Como no conocemos la sintaxis de un lenguaje de servidor almacené en este sitio una página codificada en PHP que recibe los datos del formulario y los imprime en una nueva página (recordar que en este curso no se ve como programar en el servidor)

Todo los formularios que implemento y los que usted implementará como ejercicios deben llamar a esta página: "/registrardatos.php" más adelante cuando conozca un lenguaje de servidor podrá almacenarlos en una base de datos, consultar otros datos, modificar datos existentes etc.

La segunda propiedad que debemos inicializar es method. Esta propiedad puede almacenar únicamente dos valores (post o get)

Normalmente un formulario se envía mediante post (los datos se envían con el cuerpo del formulario) En caso de utilizar get los datos se envían en la cabecera de la petición de la página, utilizando el método get estamos limitados en la cantidad de datos a enviar, no asi con el método post.

Ahora veamos el cuadro de texto donde se ingresa el nombre:

Ingrese su nombre:

<input type="text" name="nombre" size="20">

<br>

El mensaje "Ingrese su nombre:" es un texto fijo.

El elemento input permite definir un cuadro de texto (editor de línea) si asignamos a la propiedad type el valor "text".

Todo cuadro de texto debe inicializar la propiedad name con un nombre para el cuadro de texto. Este es un dato fundamental para poder recuperar el dato ingresado en el servidor (por ejemplo mediante PHP)

Por último inicializamos la propiedad size con el valor 20, esto significa que el cuadro de texto se dimensiona para permitir mostrar 20 caracteres (no se limita la cantidad de caracteres a ingresar por parte del visitante sino la cantidad de caracteres que se pueden visualizar)

Seguidamente:

<input type="submit" value="enviar">

También mediante el elemento input definimos un botón para el envío de datos al servidor. Debemos inicializar la propiedad type con el valor submit, con esto ya tenemos un botón para el envío de datos.

La propiedad value almacena la etiqueta que debe mostrar el botón.

Finalmente cerramos el formulario:

</form>

**Formulario con input pass**

En el concepto anterior vimos como implementar un formulario básico.

Veamos ahora con más detenimiento el elemento input. Este elemento hemos visto que nos permite definir cuadros de texto y botón para subir los datos al servidor. Ahora veremos que también podemos definir cuadros para el ingreso de una clave y botones para borrar el contenido de todos los controles del formulario.

Confeccionaremos un formulario que solicite el ingreso del nombre de un usuario y su clave:

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="20">

<br>

Ingrese su clave:

<input type="password" name="clave" size="12">

<br>

<input type="submit" value="enviar">

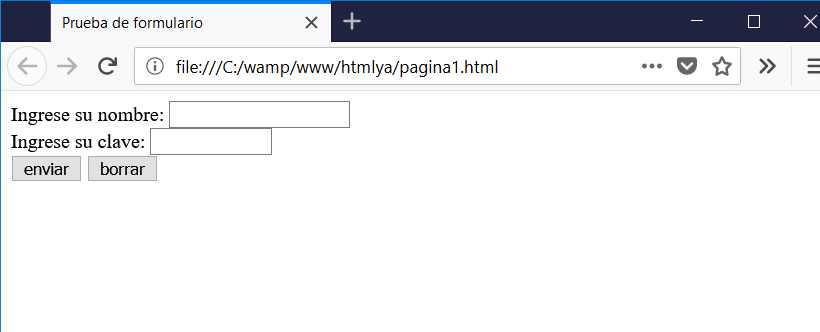
<input type="reset" value="borrar">

</form>

</body>

</html>

El resultado en el navegador es:



Veamos la sintaxis nueva para definir un cuadro de texto para el ingreso de una clave:

<input type="password" name="clave" size="12">

Utilizamos el mismo elemento input pero inicializamos la propiedad type con el valor "password", con esto logramos que cuando el visitante ingrese la clave se visualicen asteriscos en lugar de los caracteres que tipeamos.

Luego para definir un botón que permita borrar todos los datos ingresados hasta el momento lo hacemos mediante:

<input type="reset" value="borrar">

Es decir inicializamos la propiedad type con el valor "reset", con esto sabe el navegador que cuando dicho botón sea presionado debe borrar todos los controles de ingreso de datos de dicho formulario.

Otra cosa que hay que tener en cuenta que la propiedad name de cada elemento input debe tener un nombre distinto (esto debido a que en el servidor se lo rescata a partir de este nombre)

**formulario textarea**

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="30"><br>

Ingrese su mail:

<input type="text" name="mail" size="50"><br>

Comentarios:<br>

<textarea name="comentarios" rows="5" cols="60"></textarea>

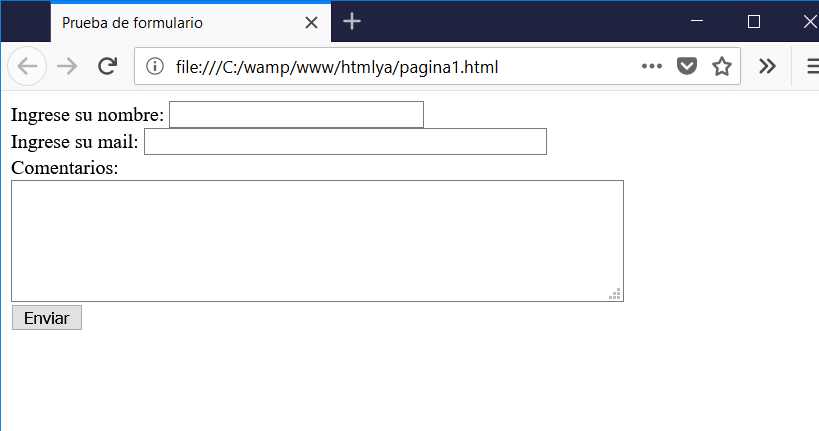
<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>



**checkbox**

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione los lenguajes que conoce:

<br>

<input type="checkbox" name="java">Java<br>

<input type="checkbox" name="cmasmas">C++<br>

<input type="checkbox" name="c">C<br>

<input type="checkbox" name="csharp">C#<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione el máximo nivel de estudios que tiene:

<br>

<input type="radio" name="estudios" value="1">Sin

estudios<br>

<input type="radio" name="estudios" value="2">Primario<br>

<input type="radio" name="estudios" value="3">Secundario<br>

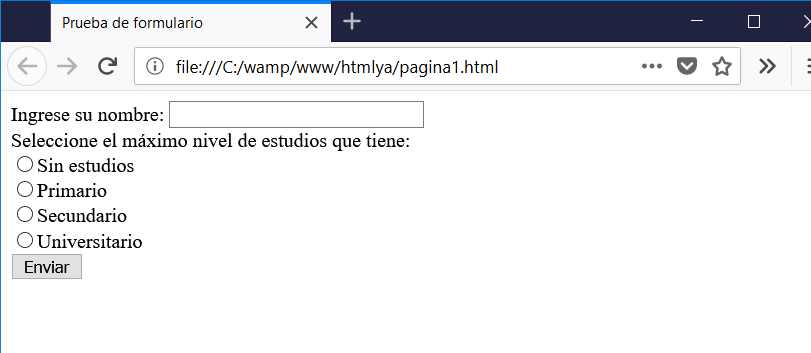
<input type="radio" name="estudios" value="4">Universitario<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>



<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione su pais:

<select name="pais">

<option value="1">Argentina</option>

<option value="2">España</option>

<option value="3">México</option>

<option value="4">Guatemala</option>

<option value="5">Honduras</option>

<option value="7">El Salvador</option>

<option value="8">Venezuela</option>

<option value="9">Colombia</option>

<option value="10">Cuba</option>

<option value="11">Bolivia</option>

<option value="13">Perú</option>

<option value="14">Ecuador</option>

<option value="15">Paraguay</option>

<option value="16">Uruguay</option>

<option value="17">Chile</option>

</select>

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

**Cuadro de selección multiple**

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Seleccione uno o varios colores (Presione Ctrl para seleccionar varios colores)<br>

<select name="colores[]" size="4" multiple>

<option value="1">Rojo</option>

<option value="2">Verde</option>

<option value="3">Azul</option>

<option value="4">Amarillo</option>

<option value="5">Blanco</option>

<option value="6">Negro</option>

<option value="7">Naranja</option>

<option value="8">Violeta</option>

</select>

<br>

<input type="submit" value="Enviar">

</form>

</body>



agrupamiento de opciones

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Seleccione una fruta o verdura:

<select name="articulo">

<optgroup label="Frutas">

<option value="1">Naranjas</option>

<option value="2">Manzanas</option>

<option value="3&guot;Sandia</option>

<option value="4">Frutilla</option>

<option value="5">Durazno</option>

<option value="6">Ciruela</option>

</optgroup>

<optgroup label="Verduras">

<option value="7">Lechuga</option>

<option value="8">Acelga</option>

<option value="9">Zapallo</option>

<option value="10">Papas</option>

<option value="11">Batatas</option>

<option value="13">Zanahorias</option>

<option value="14">Rabanitos</option>

<option value="15">Calabaza</option>

</optgroup>

</select>

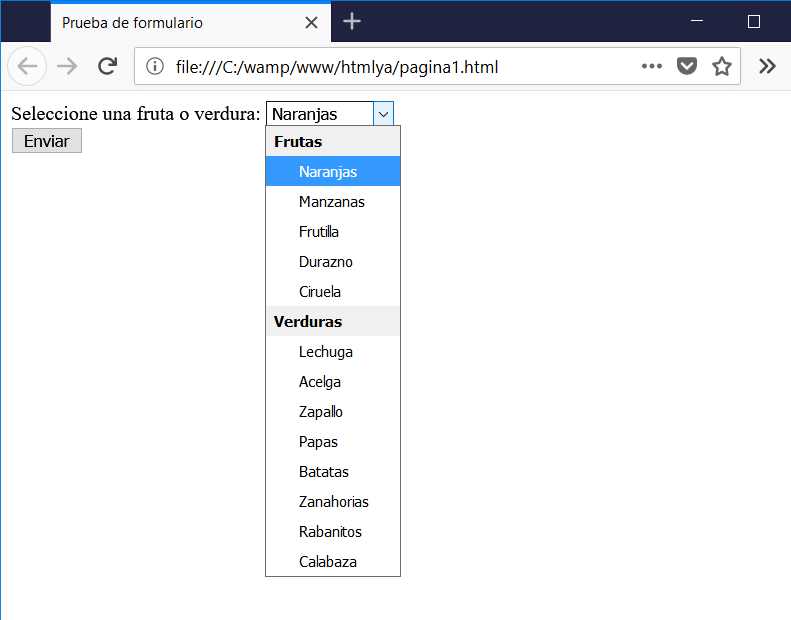
<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>



formulario con boton

<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post">

Ingrese su nombre:

<input type="text" name="nombre" size="20">

<br>

<button type="submit">Enviar<img src="enviar.gif" alt="envio"></button>

<button type="reset">Borrar<img src="borrar.gif" alt="borrado"></button>

</form>

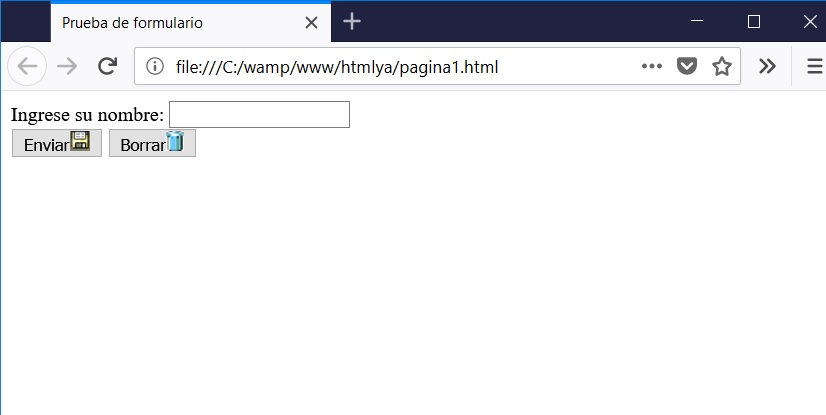
</body>

</html>

boton con imagen

<button type="submit">Enviar<img src="enviar.gif"

alt="envio"></button>



<!DOCTYPE html>

<html>

<head>

<title>Prueba de formulario</title>

<meta charset="UTF-8">

</head>

<body>

<form action="registrardatos.php" method="post" enctype="multipart/form-data">

Ingrese su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione la foto:

<input type="file" name="foto">

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

