**ENTORNOS GRÁFICOS**

**Actividad 1**

**Profesora Nora Cabello**

**INTEGRANTES:**

* **Bressán Román - 40996**

**roman.09\_94@hotmail.com**

* **Ellena Manuel - 41018**

**manu\_landeta@hotmail.com**

**ACTIVIDAD 1: HTML**

**Introducción**

**Ejemplo 1**

1. Entrar a Dreamweaver y familiarizarse con el entorno.

2. Crear un Documento Nuevo (Página Básica HTML).

3. Ir a la Vista Código y analizar la estructura del documento generado.

4. Borrar todo el código y tipear el siguiente:

<html>

<head>

<title>Ejercicio 1</title>

</head>

<body>

<h1 align="center">ENTORNOS GRÁFICOS</h1>

<!--Notar que se utiliza la fuente por defecto-->

<hr>

<p align="center"><i><font color="#CC0000" size="3">Segundo Cuatrimestre

<br><b>2006</b></i></font></p>

<p align="justify">

<font color="#000000" size="2" face="Arial, Helvetica, sans-serif">Éste es un

ejercicio sobre <b>HTML</b>, escribiendo el código directamente en la

<b> Vista Código</b> del documento en <b>Dreamweaver</b>.<br>Luego, haremos

el mismo ejercicio pero trabajando en la<b>Vista Diseño</b>.</font>

</p>

<p>&nbsp;</p>

</body>

</html>

5. Mirar la Vista Diseño generada.

6. Analizar este código.

7. Guardar el documento y ver la Vista Previa en el Navegador.

**Ejemplo 2**

1. Repetir el ejercicio anterior pero trabajando directamente en la Vista Diseño.

2. Comparar.

**Ejemplo 3**

1. Acceder a la siguiente página en Internet: http://www.e-style.com.ar/entornos/

2. Consultar el código directamente desde el navegador (Ver Código fuente)

3. Descargar la página (Guardar como Página Web completa)

4. Abrir con Dreamweaver el archivo descargado y analizar el código.

**Ejercitación**

**Ejercicio 1: Responder el siguiente cuestionario**

**1. Qué es HTML, cuando fue creado, cuáles fueron las distintas versiones y cuál es la ultima?**

Para publicar información y distribuirla globalmente, se necesita un lenguaje entendido universalmente, una especie de lengua franca de publicación que todas las computadoras puedan comprender potencialmente. El lenguaje de publicación usado por la World Wide Web es el HTML (acrónimo de HyperText Markup Language, Lenguaje para el Formato de Documentos de Hipertexto).

El HTML fue desarrollado originalmente por Tim BernersLee y fue popularizado por el navegador Mosaic. Durante los años 90 ha proliferado con el crecimiento explosivo de la Web.

El HTML 2.0 (noviembre de 1995, ver [[RFC1866]](http://html.conclase.net/w3c/html401-es/references.html#ref-RFC1866)) fue desarrollado bajo los auspicios de la Internet Engineering Task Force (IETF) para codificar lo que era la práctica común a finales de 1994. HTML+ (1993) y HTML 3.0 (1995, ver [[HTML30]](http://html.conclase.net/w3c/html401-es/references.html#ref-HTML30)) propusieron versiones mucho más ricas de HTML. A pesar de no haber logrado nunca el consenso en las discusiones sobre estándares, estos borradores llevaron a la adopción de un número de nuevas características. Los esfuerzos del Grupo de Trabajo HTML del World Wide Web Consortium para codificar la práctica común en 1996 condujeron a HTML 3.2 (enero de 1997, ver [[HTML32]](http://html.conclase.net/w3c/html401-es/references.html#ref-HTML32)).

[HTML 4.0](https://es.wikipedia.org/wiki/HTML_4.0) implementa características como [XForms](https://es.wikipedia.org/wiki/XForms" \o "XForms) 1.0 que no necesitan implementar motores de navegación que eran incompatibles con algunas páginas web HTML. En 2004 la [W3C](https://es.wikipedia.org/wiki/W3C) reabrió el debate de la evolución del HTML, y se dieron a conocer las bases para la versión [HTML5](https://es.wikipedia.org/wiki/HTML5). No obstante, este trabajo fue rechazado por los miembros del [W3C](https://es.wikipedia.org/wiki/W3C) y se daría preferencia al desarrollo del XML.

[Apple](https://es.wikipedia.org/wiki/Apple_Inc.), [Mozilla](https://es.wikipedia.org/wiki/Fundaci%C3%B3n_Mozilla) y [Opera](https://es.wikipedia.org/wiki/Opera_Software) anunciaron su interés en seguir trabajando en el proyecto bajo el nombre de [WHATWG](https://es.wikipedia.org/wiki/WHATWG),[20](https://es.wikipedia.org/wiki/HTML#cite_note-20) que se basa en la compatibilidad con tecnologías anteriores.

En 2006, el [W3C](https://es.wikipedia.org/wiki/W3C) se interesó en el desarrollo de [HTML5](https://es.wikipedia.org/wiki/HTML5), y en 2007 se unió al grupo de trabajo del [WHATWG](https://es.wikipedia.org/wiki/WHATWG) para unificar el proyecto. HTML 5 es la versión que se usa en la actualidad.

**2. ¿Cuáles son los principios básicos que el W3C recomienda seguir para la creación de**

**documentos con HTML?**

El W3C se guía por los principios de accesibilidad, internacionalización, e independencia de dispositivo, entre otros. Esto facilita que el acceso a la Web sea posible desde cualquier lugar, en cualquier momento y utilizando cualquier dispositivo. No importa si se utiliza hardware, software, o una infraestructura de red específica. Además de las posibles restricciones técnicas, se tiene en cuenta la existencia de múltiples idiomas, las diversas localizaciones geográficas, y las diferencias culturales o tradiciones, así como las posibles limitaciones físicas, psíquicas o sensoriales de los usuarios.

**3. En las Especificaciones de HTML, ¿cuándo un elemento o atributo se considera desaprobado?¿y obsoleto?**

* **Desaprobado:** anticuado por la presencia de estructuras nuevas.
* **Obsoleto:** es aquel para el cual no hay garantía de soporte.

**4. Qué es el DTD y cuáles son los posibles DTDs contemplados en la especificación de HTML**

**4.01?**

Una **declaración de tipo de documento** o **DOCTYPE** (del [inglés](https://es.wikipedia.org/wiki/Idioma_ingl%C3%A9s) *document type declaration*) asocia un documento [SGML](https://es.wikipedia.org/wiki/SGML)(HTML) o [XML](https://es.wikipedia.org/wiki/XML) particular con una [definición de tipo de documento](https://es.wikipedia.org/wiki/Definici%C3%B3n_de_tipo_de_documento). En la forma serializada de un documento, este se manifiesta como una cadena corta de caracteres de marcado (*[markup](https://es.wikipedia.org/wiki/Markup" \o "Markup)*) que se conforma con una sintaxis particular.

### HTML 4.01 DTDs

EL HTML 4.01 Strict no permite marcado de presentación con el argumento de que debería usar CSS en su lugar para eso. Así es como luce el DOCTYPE Strict:

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">

Las DTD transicionales (Transitional DTD) permiten algunos elementos y atributos antiguos que están en desuso:

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

Además, si está usando frames (marcos), para conseguir resultados válidos desde el validador SGML, necesitará el frameset DOCTYPE como el que sigue:

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">

**5. Qué son los metadatos y cómo se especifican en HTML?**

Son un **Marco de Descripción de Recursos(ResourceDescription Framework)** del W3C. Se convirtió en Recomendación del W3C e n febrero de 1999. Permite a los autores especificar metadatos legibles por máquina sobre documentos HTML y otros recursos accesibles por la red. Por ejemplo, para especificar el autor:

*<META name=”Author” content=”Dave Raggett”>*

El elemento META puede utilizarse para identificar propiedades de un documento (p.ej., el autor, la fecha de caducidad, una lista de palabras clave, etc.) y para asignar valores a esas propiedades. Puede utilizarse el atributo *lang* de META para especificar el idioma del valor del atributo *content*. Esto permite a los sintetizadores de voz aplicar reglas de pronunciación dependientes del idioma. Ejemplo: declara que el nombre del autor está en francés:

*<META name=”Author” lang=”fr” content=”A.Le Hors”>*

El siguiente ejemplo especifica la codificación de caracteres:

*<META http-equiv=”Content-Type” content=”text/html; charset=ISO-8859-5” >*

Un uso común de META es especificar palabras clave que pueden usar los motores de búsqueda para mejorar la calidad de los resultados de una búsqueda. Cuando se proporcionen varios elementos META con información para varios idiomas, los motores de búsqueda pueden utilizar el atributo lang como filtro para mostrar los resultados de la búsqueda usando las preferencias de idioma del usuario:

*<!-- Para hablantes de inglés americano -->*

*<META name=”keywords” lang=”en-us” content=”vacation, Greece, sunshine”>*

*<!-- Para hablantes de español -->*

*<META name=”keywords” lang=”es” content=”vacaciones, Grecia, sol”>*

**LECTURAS SUGERIDAS**

**Especificación HTML 4.01**

**http://html.conclase.net/w3c/html401-es/cover.html**

**Lista de tags HTML**

**http://www.htmlquick.com/es/reference/tags.html**

**Ejercicio 2:**

*HTML está compuesto por un conjunto de elementos que son la base de su estructura.*

*Los elementos están compuestos por dos tags (el de apertura y el de cierre) y el contenido en el medio (con excepción de los elementos vacíos).*

*Cada tag puede tener atributos (proporcionan ciertas características como altura, ancho, color, etc.) y eventos (asocian un script que se ejecuta cuando el evento ocurre).*

Analizar los siguientes segmentos de código indicando en qué sección del documento HTML se colocan, cuál es el efecto que producen y señalar cada uno de los elementos, etiquetas, y atributos (nombre y valor), aclarando si es obligatorio.

**<!-- Código controlado el día 12/08/2009 -->**

El tag <!-- --> inserta comentarios ocultos. Puede ir en cualquier parte del documento.

**<div id="bloque1">Contenido del bloque1</div>**

El tag <div> define un bloque de contenido y el atributo “id” asigna un identificador único a un elemento (lo cual puede ser verificado por un analizador SGML), no es obligatorio dentro del DIV.

**<img src="" alt="lugar imagen" id="im1" name="im1" width="32" height="32"**

**longdesc="detalles.htm" />**

El tag <img> inserta una imagen. El atributo “src” es obligatorio y se encarga de especificar la URL del archivo multimedia. El atributo “alt” especifica un texto alternativo que se representa cuando la imagen no puede ser mostrada. Los agentes de usuario deben representar el texto alternativo cuando no puedan soportar imágenes, o cuando estén configurados para no mostrar imágenes. El atributo “name” sirve para darle un nombre a la imagen. Los atributos “height” y “width” dan a los agentes de usuario una idea del tamaño de una imagen u objeto para que puedan reservar espacio para ellos y continuar la representación del documento mientras esperan a los datos de la imagen. El atributo “**longdesc”** complementa al atributo **alt** y se emplea para ofrecer una descripción más larga del elemento que la proporcionada por el atributo alt.

**<meta name="keywords" lang="es" content="casa, compra, venta, alquiler " />**

**<meta http-equiv="expires" content="16-Sep-2009 7:49 PM" />**

El tag <meta> nos da información sobre el documento. Este tag debe estar entre las etiquetas <head>. La etiqueta <meta> no necesita etiqueta de cierre. A través de esta etiqueta pueden especificarse los atributos “name” y “content”. El atributo “name” indica el tipo de información, y el atributo “content” indica el valor de dicha información. El atributo “http-equiv” se utiliza para indicarle al navegador alguna información o alguna acción que debe realizar. El atributo “lang” especifica el idioma del contenido del elemento.

**<a href="http://www.e-style.com.ar/resumen.html" type="text/html" hreflang="es" charset="utf-8" rel="help">Resumen HTML </a>**

El tag <a> sirve para insertar vínculos o marcadores. Luego del <a> se coloca el atributo “href” que es obligatorio y sirve para declarar el enlace a insertar. El atributo “type” especifica el tipo de elemento a enlazar. El atributo “hreflang” especifica el lenguaje del documento enlazado, si es en español se debe colocar el valor “es”, si es en ingles “en”. El atributo “charset” especifica la codificación de caracteres. El atributo “rel” especifica la relación entre el documento actual y el documento vinculado.

**<table width="200" summary="Datos correspondientes al ejercicio vencido">**

**<caption align="top">**

**Título**

**</caption>**

**<tr>**

**<th scope="col">&nbsp;</th>**

**<th scope="col">A</th>**

**<th scope="col">B</th>**

**<th scope="col">C</th>**

**</tr>**

**<tr>**

**<th scope="row">1º</th>**

**<td>&nbsp;</td>**

**<td>&nbsp;</td>**

**<td>&nbsp;</td>**

**</tr>**

**<tr>**

**<th scope="row">2º</th>**

**<td>&nbsp;</td>**

**<td>&nbsp;</td>**

**<td>&nbsp;</td>**

**</tr>**

**</table>**

La etiqueta <table> define una tabla. Una tabla va a poder tener diferentes filas (etiqueta <tr>) y dentro de cada fila vamos a poder definir diferentes celdas (etiquetas <th> si son la cabecera de la tabla o etiqueta <td> si es el propio contenido). Dentro de una tabla vamos a poder introducir elementos como imágenes, texto, enlaces, etc.

La etiqueta <summary> define un encabezado visible.

La etiqueta <caption> define una leyenda de la tabla y debe insertarse inmediatamente después de la etiqueta <table>.

La etiqueta <align> especifica la alineación de acuerdo con los elementos circundantes. Utilice CSS en su lugar.

**Ejercicio 3: En cada caso, explicar las diferencias entre los segmentos de código y sus**

**visualizaciones:**

**a)**

<a href="http://www.google.com.ar">Click aquí para ir a Google</a>

<a href="http://www.google.com.ar" target="\_blank">Click aquí para ir a Google</a>

<a href="http://www. google.com.ar" type="text/html" hreflang="es" charset="utf-8" rel="help">

<a href="#">Click aquí para ir a Google</a>

<a href="#arriba">Click aquí para volver arriba</a>

<a name="arriba" id="arriba"></a>

**En el segundo caso se agrega el atributo TARGET el cual nos permite mostrar el resultado de un enlace en el frame que queramos indicando el nombre de dicho frame, en este caso target=”\_blank” abrirá el enlace en una nueva pestaña.  
En el tercer caso el atributo “type” especifica el tipo de elemento a enlazar, “hreflang” el lenguaje del documento, “charset” la codificación de caracteres y “rel” especifica la relación entre el documento actual y el documento vinculado.  
En el cuarto caso se referencia a un id sin definir, pero no hay un redireccionamiento a alguna pagina, solo texto.  
En el quinto caso se hace referencia un id llamado “arriba”, pero tampoco redireccionará si este no está definido en otro lado.  
En el sexto caso falta el atributo href para el enlace, en este caso se define un id con nombre “arriba”.**

**b)**

<p><img src="im1.jpg" alt="imagen1" /><a href="http://www.google.com.ar">Click aquí</a></p>

<p><a href="http://www.google.com.ar"><img src="im1.jpg" alt="imagen1" /></a> Click aquí</p>

<p><a href="http://www.google.com.ar"><img src="im1.jpg" alt="imagen1" />Click aquí</a></p>

<p><a href="http://www.google.com.ar"><img src="im1.jpg" alt="imagen1" /></a> <a href=

"http://www.google.com.ar">Click aquí</a></p>

**Entre los dos primeros casos la diferencia es el orden de las etiquetas, en el primer caso la etiqueta <a> encierra a la palabra “Click aquí” por lo tanto este va a ser el enlace, y en el segundo caso el enlace está incluido en la imagen insertada. En el tercer caso, el enlace está incluido en la imagen y en la frase “Click aquí” y en el último caso realiza los enlaces por separado.**

**c)**

|  |  |  |  |
| --- | --- | --- | --- |
| <ul>  <li>xxx</li>  <li>yyy</li>  <li>zzz</li>  </ul> | <ol>  <li>xxx</li>  <li>yyy</li>  <li>zzz</li>  </ol> | <ol>  <li>xxx</li>  </ol>  <ol>  <li value="2">yyy</li>  </ol>  <ol>  <li value="3">zzz</li>  </ol> | <blockquote>  <p>1. xxx<br />  2. yyy<br />  3. zzz</p>  </blockquote> |

**Son 4 formas de listas.  
En el primer caso, se usan círculos como viñetas para dividir las filas. En el segundo caso con el tag <ol> se numeran las filas. En el tercer caso las filas también se numeran pero por separado dejando un espacio entre ellas y mediante el atributo “value”. En el último caso se usa el tag <blockquote> (que sirve para insertar citas textuales) y dentro de él se usan párrafos <p> donde se escriben los números y no se numeran las filas automáticamente.**

**d)**

|  |  |
| --- | --- |
| <table border="1" width="300">  <tr>  <th>Columna 1</th>  <th>Columna 2</th>  </tr>  <tr>  <td>Celda 1</td>  <td>Celda 2</td>  </tr>  <tr>  <td>Celda 3</td>  <td>Celda 4</td>  </tr>  </table> | <table border="1" width="300">  <tr>  <td><div align="center"><strong>Columna1</strong></div></td>  <td><div align="center"><strong>Columna 2</strong></div></td>  </tr>  <tr>  <td>Celda 1</td>  <td>Celda 2</td>  </tr>  <tr>  <td>Celda 3</td>  <td>Celda 4</td>  </tr>  </table> |

**La diferencia entre estos dos casos es que se usa la etiqueta <strong> en la primera fila de cada columna, lo que permite marcar con énfasis esa celda de la tabla(Poner la letra en negrita). En el segundo caso no se utilizan los atributos th para definir el texto de la cabecera de las columnas, sino que se utilizan div, con propiedades de alineación.**

**e)**

|  |  |
| --- | --- |
| <table width="200">  <caption>  Título  </caption>  <tr>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  <tr>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  </table> | <table width="200">  <tr>  <td colspan="3"><div align="center">Título</div></td>  </tr>  <tr>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  <tr>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  </table> |

**La diferencia entre estos casos es que en el Segundo se usa el atributo “colspan” lo que permite que la celda de “Título” se expanda en la cantidad de columnas que le indiquemos, en este caso se expande en las 3 columnas. En el primer caso la palabra titulo se define mediante “Caption” en cambio en el segundo caso se define dentro de una celda con td**

**f)**

|  |  |
| --- | --- |
| <table width="200">  <tr>  <td colspan="3"><div align="center">Título</div></td>  </tr>  <tr>  <td rowspan="2" bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  <tr>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  </table> | <table width="200">  <tr>  <td colspan="3"><div  align="center">Título</div></td>  </tr>  <tr>  <td colspan="2"  bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  <tr>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  <td bgcolor="#dddddd">&nbsp;</td>  </tr>  </table> |

**La diferencia entre estos dos casos es que en el primero se usa el atributo “rowspan” que permite expandir celdas a través de filas, y en el segundo caso se usa “colspan” que permite expandir celdas pero a través de columnas.**

**g)**

|  |  |
| --- | --- |
| <table width="200" border="1">  <tr>  <td colspan="3"><div  align="center">Título</div></td>  </tr>  <tr>  <td colspan="2"  rowspan="2">&nbsp;</td>  <td>&nbsp;</td>  </tr>  <tr>  <td width="50%">&nbsp;</td>  </tr>  </table> | <table width="200" border="1" cellpadding="0" cellspacing="0">  <tr>  <td colspan="2"><div align="center">Título</div></td>  </tr>  <tr>  <td rowspan="2">&nbsp;</td>  <td>&nbsp;</td>  </tr>  <tr>  <td width="50%">&nbsp;</td>  </tr>  </table> |

**Difieren en que en el Segundo caso se usan los atributos “cellpadding” (permite especificar la distancia entre el margen interior de la celda y su contenido) y**“**cellspacing” (permite especificar la distancia entre las celdas y el margen exterior de la tabla) por lo que la segunda tabla los tiene en valor 0.**

**h)**

|  |
| --- |
| <form id="form1" name="form1" action="procesar.php" method="post" target="\_blank">  <fieldset>  <legend>LOGIN</legend>  Usuario: <input type="text" id="usu1" name="usu1" value="xxx" /><br />  Clave: <input type="password" id="clave1" name="clave1" value="xxx" />  </fieldset>  <input type="submit" id="boton1" name="boton1" value="Enviar" />  </form> |
| <form id="form2" name="form2" action="" method="get" target="\_blank">  LOGIN<br />  <label>Usuario: <input type="text" id="usu2" name="usu2" /></label><br />  <label>Clave: <input type="text" id="clave2" name="clave2" /></label><br />  <input type="submit" id="boton2" name="boton2" value="Enviar" />  </form> |
| <form id="form3" name="form3" action="mailto:xx@xx.com” enctype=text/plain method="post" target="\_blank">  <fieldset>  <legend>LOGIN</legend>  Usuario: <input type="text" id="usu3" name="usu3" /><br />  Clave: <input type="password" id="clave3" name="clave3" />  </fieldset>  <input type="reset" id="boton3" name="boton3" value="Enviar" />  </form> |

**i)**

|  |
| --- |
| <label>Botón 1  <button type="button" name="boton1" id="boton1">  <img src="logo.jpg" alt="Botón con imagen " width="30" height="20" /><br />  <b>CLICK AQUÍ</b></button></label> |
| <label>Botón 2  <input type="button" name="boton2" id="boton2" value="CLICK AQUÍ" />  </label> |

**En el primer caso, dentro de la etiqueta <button> se adjunta una imagen de un logo y debajo se coloca la frase “Click aquí”. En el segundo caso se usa el elemento <input type= “button”> que es una versión específica del elemento <input>, que se usa para crear un botón en el que se puede hacer click sin ningún valor por defecto. Ha sido sustituido en HTML5 por el elemento <button>.**

**j)**

|  |
| --- |
| <p><label><input type="radio" name="opcion" id="X" value="X" />X</label><br />  <label><input type="radio" name="opcion" id="Y" value="Y" />Y</label></p> |
| <p><label><input type="radio" name="opcion1" id="X" value="X" />X</label><br />  <label><input type="radio" name="opcion2" id="Y" value="Y" />Y</label></p> |

**La única diferencia entre estos dos casos es el nombre (atributo “name”) que es oculto al usuario.**

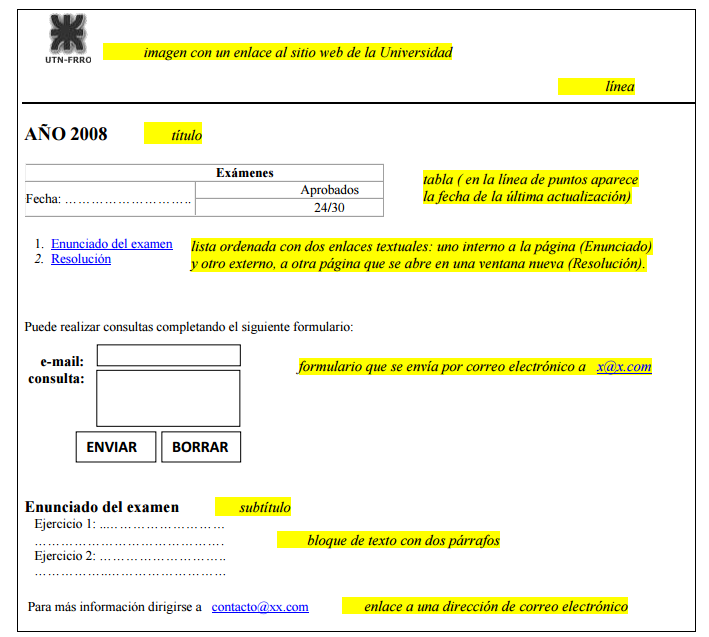
**k)**

|  |  |
| --- | --- |
| <select name="lista">  <optgroup label="Caso 1">  <option>Mayo</option>  <option>Junio</option>  </optgroup>  <optgroup label="Caso 2">  <option>Mayo</option>  <option>Junio</option>  </optgroup>  </select> | <select name="lista[]" multiple="multiple">  <optgroup label=" Caso 1">  <option>Mayo</option>  <option>Junio</option>  </optgroup>  <optgroup label=" Caso 2">  <option>Mayo</option>  <option>Junio</option>  </optgroup>  </select> |

**La diferencia está en el elemento SELECT que permite el atributo multiple para poder seleccionar más que una opción a la vez. (Las selecciones mútliples se hacen normalmente pulsando la tecla Control mientras se seleccionan las opciones con el ratón.) El atributo multiple no tiene valor. Si no está, entonces el elemento SELECT sólo permite seleccionar una opción.**

**Ejercicio 4:**

Escribir el código HTML completo necesario para visualizar la siguiente salida en el navegador:

****

**Ejercicio 5**

Trabajando en Vista Diseño, crear un documento con marcos que presente la siguiente disposición:

|  |  |  |
| --- | --- | --- |
| encabezado | | Logo |
| menu | principal | |
| pie | | |

En el marco correspondiente al logo, debe aparecer una imagen, y en el marco encabezado solo texto.

En el marco correspondiente al menú, una lista no ordenada con enlaces a las páginas

Articulos.htm y MasInfo.htm que se deben abrir en el marco principal.

En Articulos.htm, se deben mostrar los artículos en existencia y sus precios, en una tabla.

En MasInfo.htm debe aparecer la siguiente dirección de correo electrónico, para realizar las

consultas: consultas@empresa.com

En el marco pie debe aparecer nuevamente el menú de navegación.