

3 - Un ejemplo con AJAX.

Confeccionaremos un ejemplo donde veremos que aparecen muchos conceptos, no se preocupe si no los comprende en su totalidad ya que los mismos se verán en forma detallada a lo largo de este curso.

La idea fundamental de este ejercicio es conocer como debemos estructurar nuestras páginas y ver que introduce de nuevo el empleo de AJAX.

Confeccionaremos un problema muy sencillo, imaginemos que tenemos una lista de hipervínculos con los distintos signos del horóscopo y queremos que al ser presionado no recargue la página completa sino que se envíe una petición al servidor y el mismo retorne la información de dicho signo, luego se actualice solo el contenido de un div del archivo HTML.

Este problema se puede resolver muy fácilmente si refrescamos la página completamente al presionar el hipervínculo, pero nuestro objetivo es actualizar una pequeña parte de la página y más precisamente el div que debe mostrar los datos del signo seleccionado.

Si bien nuestra página solo contendrá los hipervínculos a los distintos signos en un caso real la página puede contener muchos otros elementos HTML con imágenes, otros hipervínculos etc. los cuales no deberán sufrir cambios (ni parpadeo) ya que solo se modificará el elemento div respectivo mediante DHTML.

Esta actualización parcial de la página tiene muchas ventajas:

- Reducimos el ancho de banda requerido al no tener que recuperar toda la página.
- Agilizamos la actualización de la página.
- Reducimos el parpadeo de la página.
- Hacemos más natural la navegación del sitio.

La mayoría de los problemas requieren los siguientes archivos como mínimo:

1. El archivo HTML (es la página que se ve en el navegador)
2. El archivo JS (contiene todas las rutinas JavaScript que permiten actualizar dinámicamente la página HTML (mediante DHTML) y las rutinas que permiten comunicarse con el servidor para el envío y recepción de información)
3. La hoja de estilo, es decir el archivo CSS
4. La página que contiene el script que se ejecuta en el servidor(en nuestro caso emplearemos el lenguaje PHP)

Comencemos a presentar los distintos archivos para resolver este problema:

pagina1.html

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Problema</title>
  <script src="funciones.js"></script>
  <link href="estilos.css" rel="StyleSheet" type="text/css">
</head>

<body>
<h1>Signos del horóscopo.</h1>
<div id="menu">
  <p><a id="enlace1" href="pagina1.php?cod=1">Aries</a></p>
  <p><a id="enlace2" href="pagina1.php?cod=2">Tauro</a></p>
  <p><a id="enlace3" href="pagina1.php?cod=3">Geminis</a></p>
  <p><a id="enlace4" href="pagina1.php?cod=4">Cancer</a></p>
  <p><a id="enlace5" href="pagina1.php?cod=5">Leo</a></p>
  <p><a id="enlace6" href="pagina1.php?cod=6">Virgo</a></p>
  <p><a id="enlace7" href="pagina1.php?cod=7">Libra</a></p>
  <p><a id="enlace8" href="pagina1.php?cod=8">Escorpio</a></p>
  <p><a id="enlace9" href="pagina1.php?cod=9">Sagitario</a></p>
  <p><a id="enlace10" href="pagina1.php?cod=10">Capricornio</a></p>
  <p><a id="enlace11" href="pagina1.php?cod=11">Acuario</a></p>
  <p><a id="enlace12" href="pagina1.php?cod=12">Piscis</a></p>
</div>
<div id="detalles">Seleccione su signo.</div>
</body>
</html>

```

Esta página contiene HTML puro. Es importante notar que debemos incorporar los dos archivos externos .css y .js mediante los elementos HTML respectivos:

```

<script src="funciones.js"></script>
<link href="estilos.css" rel="StyleSheet" type="text/css">

```

La hoja de estilo solo tiene el objetivo de mejorar la presentación en la página de los doce hipervínculos de los signos del horóscopo. Puede probar de eliminar el archivo .css mediante el borrado del elemento link del archivo HTML y el problema debería continuar funcionando, por supuesto con una presentación mucho más pobre.

Podemos observar que cada hipervínculo solicita la misma página al servidor pero pasándole como parámetro un valor distinto, con esto podremos detectar en el servidor que signo eligió el operador.

El segundo archivo contiene las reglas de estilo que se definen para el archivo HTML:

estilos.css

```

#menu {
  font-family: Arial;
  margin:5px;
}

#menu p {
  margin:0px;
  padding:0px;
}

#menu a {
  display: block;
  padding: 3px;
  width: 160px;
  background-color: #f7f8e8;
  border-bottom: 1px solid #eee;
  text-align:center;
}

#menu a:link, #menu a:visited {
  color: #f00;
  text-decoration: none;
}

#menu a:hover {
  background-color: #369;
  color: #fff;
}

#detalles {
  background-color:#ffc;
  text-align:left;
  font-family:verdana;
  border-width:0;
  padding:5px;
  border: 1px dotted #fa0;
  margin:20px;
  border-radius: 5px;
  box-shadow: 0px 0px 20px #aaa;
}

```

No haremos un análisis de estas reglas ya que corresponden al tema 28 del curso de [CSS Ya](#) "*Creación de un menú vertical configurando las pseudoclases*", puede refrescar los conceptos allí. Inclusive si todavía conoce poco de CSS y no quiere estudiarlo por ahora puede anular el archivo no incorporándolo en la página HTML suprimiendo el elemento link.

Ahora viene uno de los puntos claves donde debemos prestar más atención, esto se encuentra en las rutinas JavaScript que debemos implementar para comunicarnos con el servidor, además de lo ya conocido de DHTML para añadir elementos HTML en forma dinámica.

Veamos el archivo en su totalidad y expliquemos en forma muy global (recuerde que a lo largo de este curso iremos profundizando todos estos conceptos de comunicación con el servidor):

funciones.js

```
addEventListener('load',inicializarEventos,false);

function inicializarEventos()
{
    for(var f=1;f<=12;f++)
    {
        var ob=document.getElementById('enlace'+f);
        ob.addEventListener('click',presionEnlace,false);
    }
}

function presionEnlace(e)
{
    e.preventDefault();
    var url=e.target.getAttribute('href');
    cargarHoroscopo(url);
}

var conexion1;
function cargarHoroscopo(url)
{
    conexion1=new XMLHttpRequest();
    conexion1.onreadystatechange = procesarEventos;
    conexion1.open("GET", url, true);
    conexion1.send();
}

function procesarEventos()
{
    var detalles = document.getElementById("detalles");
    if(conexion1.readyState == 4)
    {
        detalles.innerHTML = conexion1.responseText;
    }
    else
    {
        detalles.innerHTML = 'Cargando...';
    }
}
```

En este punto si es indispensable haber realizado el curso de [DHTML Ya](#) para entender como registramos los eventos para los doce hipervínculos. Recordemos que siempre llamaremos a la función `addEventListener`.

Lo primero que se ejecuta es la llamada a la función `inicializarEventos()` inmediatamente luego que la página se a cargado por completo en el navegador:

```
function inicializarEventos()
{
  for(var f=1;f<=12;f++)
  {
    var ob=document.getElementById('enlace'+f);
    ob.addEventListener('click',presionEnlace,false);
  }
}
```

En esta función registramos el evento click para los doce enlaces de los signos del horóscopo. Para facilitar la codificación recordemos que todos tienen casi el mismo nombre, difieren por un número al final. Luego dentro de un for rescatamos la referencia a cada enlace y registramos el evento click indicando que se debe llamar a la función presionEnlace.

La función presión enlace:

```
function presionEnlace(e)
{
  e.preventDefault();
  var url=e.target.getAttribute('href');
  cargarHoroscopo(url);
}
```

Primero desactivamos el evento por defecto para el hipervínculo, luego llama a la función cargarHoroscopo pasándole como referencia la url que contiene el hipervínculo.

Todo lo comentado hasta acá se estudió en cursos anteriores: HTML, JavaScript, CSS, DHTML.

Veamos ahora la función cargarHoroscopo:

```
var conexion1;
function cargarHoroscopo(url)
{
  conexion1=new XMLHttpRequest();
  conexion1.onreadystatechange = procesarEventos;
  conexion1.open("GET", url, true);
  conexion1.send();
}
```

Previo a la definición de esta función definimos una variable global llamada conexion1 que será utilizada en esta y la siguiente función.

La función recibe como parámetro la url a la que debe hacer la petición de datos.

Lo primero que hacemos es crear un objeto de la clase XMLHttpRequest (luego veremos que este objeto nos permite comunicarnos con el servidor de forma asincrónica):

```
conexion1=new XMLHttpRequest();
```

La propiedad onreadystatechange se inicializa con la referencia de una función que será la encargada de procesar los datos enviados por el servidor, veremos el código de esta función más adelante.

Seguidamente llamamos al método open que tiene tres parámetros:

- Primero el método de envío de datos (GET o POST) Recordemos que si los datos se envían como parámetros (como es nuestro ejemplo) debemos indicar que utilizamos el método GET
- El segundo parámetro es la url y la página que procesará los datos que le enviemos.
- El tercer parámetro indicamos si se procesarán los datos de forma asíncrona (true) o síncrona

(false)

Por último nos falta llamar al método send para que comience el proceso:

```
conexion1.send();
```

Nos queda explicar la función procesarEventos que se ejecuta cada vez que el objeto conexion1 de la clase XMLHttpRequest cambia de estado. Tengamos en cuenta que los estados posibles de este objeto son:

- 0 No inicializado.
- 1 Cargando.
- 2 Cargado.
- 3 Interactivo.
- 4 Completado.

Para conocer el estado del objeto debemos acceder a la propiedad readyState que almacena alguno de los cinco valores que enunciamos.

Nuestra función procesarEventos es:

```
function procesarEventos()
{
    var detalles = document.getElementById("detalles");
    if(conexion1.readyState == 4)
    {
        detalles.innerHTML = conexion1.responseText;
    }
    else
    {
        detalles.innerHTML = 'Cargando...';
    }
}
```

Decíamos que cuando la propiedad readyState almacena 4 significa que todos los datos han llegado desde el servidor, luego mediante el método responseText recuperamos la información enviada por el servidor. Luego cualquier otro valor que contenga la propiedad readyState mostramos dentro del div el mensaje 'cargando...'.

Es seguro que muchas dudas han surgido de este primer pantallazo de AJAX, pero no se preocupe a medida que avancemos en el curso se irán aclarando e internalizando.

Pero todavía nos queda la página que contiene el programa en el servidor, en nuestro caso empleamos el lenguaje PHP (tener en cuenta que podemos emplear otro lenguaje de servidor para esto)

Veamos el código de esta página:

```

<?php
header('Content-Type: text/html; charset=utf-8');
if ($_REQUEST['cod']==1)
    echo "<strong>Aries:</strong> Hoy los cambios serán físicos, personales, de carácter, Te sentirás
if ($_REQUEST['cod']==2)
    echo "<strong>Tauro:</strong> Hoy los cambios serán privados, íntimos. Recuerdos. Ayuda, soli
if ($_REQUEST['cod']==3)
    echo "<strong>Géminis:</strong> Los asuntos de hoy tienen que ver con las amistades, reunione
if ($_REQUEST['cod']==4)
    echo "<strong>Cancer:</strong> Este día la profesión y las relaciones con superiores y con tu ma
if ($_REQUEST['cod']==5)
    echo "<strong>Leo:</strong> Este día los estudios, los viajes, el extranjero y la espiritualidad ser
if ($_REQUEST['cod']==6)
    echo "<strong>Virgo:</strong> Para este día toma importancia tu vida sexual, tal vez miedos, ten
if ($_REQUEST['cod']==7)
    echo "<strong>Libra:</strong> Hoy todo asunto tiene que ver con tu pareja, también con socios, e
if ($_REQUEST['cod']==8)
    echo "<strong>Escorpio:</strong> Hoy todo asunto tiene que ver con temas de trabajo y de salud
if ($_REQUEST['cod']==9)
    echo "<strong>Sagitario:</strong> Durante este día se vivirán cambios en relación a los noviazgo
if ($_REQUEST['cod']==10)
    echo "<strong>Capricornio:</strong> Los cambios del día tienen que ver con tu hogar, con la con
if ($_REQUEST['cod']==11)
    echo "<strong>Acuario:</strong> Hoy todo asunto tiene que ver con el entorno inmediato, herma
if ($_REQUEST['cod']==12)
    echo "<strong>Piscis:</strong> Durante este día se vivirán cambios en la economía, movimientos
?>

```

Mediante el vector asociativo \$_REQUEST recuperamos el valor del parámetro cod y mediante una serie de if verificamos si almacena el valor 1 procedemos a generar un texto referente al signo Aries, si tiene un 2 generamos un texto referente al signo Tauro y así sucesivamente.

Hay que tener en cuenta que no se estará enviando una página HTML completa, por eso no tiene los elementos Head, Body etc. sino es más bien un archivo de texto que luego será añadido en forma dinámica al div de la página HTML.

Debe quedar claro que los datos se podrían haber rescatado perfectamente de una base de datos, pero por simplicidad hemos dispuesto estos 12 if y generado el texto respectivo. Veremos más adelante problemas que acceden a bases de datos.

Hasta acá el primer problema de AJAX. Le recomiendo pasar a la sección de "Problemas Resueltos" y ejecutar este ejercicio, releer nuevamente estos conceptos y tratar de hacer modificaciones sencillas al problema.

Problema resuelto.

Confeccionar un problema que muestre una lista de hipervínculos con los distintos signos del horóscopo y luego al ser presionado no recargue la página completa sino que se envíe una petición al servidor y el mismo retorne la información de dicho signo, luego se actualice solo el contenido de un div del archivo HTML.

Para ver que solo se modifica el div "detalles" puede agregar otros elementos a la página haciéndola mas compleja (por ejemplo agregue 20 títulos h1) Observe luego que al presionar el hipervínculo solo se actualiza el div "detalles", sin el parpadeo de la página que ocurre cuando recargamos una página HTML completamente.

pagina1.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Problema</title>
  <script src="funciones.js"></script>
  <link href="estilos.css" rel="StyleSheet" type="text/css">
</head>

<body>
<h1>Signos del horóscopo.</h1>
<div id="menu">
  <p><a id="enlace1" href="pagina1.php?cod=1">Aries</a></p>
  <p><a id="enlace2" href="pagina1.php?cod=2">Tauro</a></p>
  <p><a id="enlace3" href="pagina1.php?cod=3">Geminis</a></p>
```

[Ejecutar ejemplo](#)

estilos.css

```
#menu {
  font-family: Arial;
  margin:5px;
}

#menu p {
  margin:0px;
  padding:0px;
}

#menu a {
  display: block;
  padding: 3px;
  width: 160px;
  background-color: #f7f8e8;
```

funciones.js


```
addEventListener('load',inicializarEventos,false);

function inicializarEventos()
{
    for(var f=1;f<=12;f++)
    {
        var ob=document.getElementById('enlace'+f);
        ob.addEventListener('click',presionEnlace,false);
    }
}

function presionEnlace(e)
{
    e.preventDefault();
    var url=e.target.getAttribute('href');
```

pagina1.php

```
<?php
header('Content-Type: text/html; charset=utf-8');
if ($_REQUEST['cod']==1)
    echo "<strong>Aries:</strong> Hoy los cambios serán físicos, personales, de carácter, Te sentirás impulsivo y
tomarás iniciativas. Período en donde considerarás unirte a agrupaciones de beneficencia, o de ayuda a los
demás.";
if ($_REQUEST['cod']==2)
    echo "<strong>Tauro:</strong> Hoy los cambios serán privados, íntimos. Recuerdos. Ayuda, solidaridad.
Asuntos en lugares de retiro. Tu cónyuge puede aportar buen status a tu vida o apoyo a tu profesión.";
if ($_REQUEST['cod']==3)
    echo "<strong>Géminis:</strong> Los asuntos de hoy tienen que ver con las amistades, reuniones, actividades
con ellos. Día esperanzado, ilusiones. Mucha energía sexual y fuerza emocional. Deseos difíciles de controlar.";
if ($_REQUEST['cod']==4)
    echo "<strong>Cancer:</strong> Este día la profesión y las relaciones con superiores y con tu madre serán de
importancia. Actividad en relación a estos temas. Momentos positivos con compañeros de trabajo. Actividad
```

[**Retornar**](#)