

# Lejanías

Lejanías consiste en un libro interactivo, realizado en HTML/CSS y JavaScript, que invita al lector a construirse un recorrido sobre las estaciones y los tipos de poesía que se deseen sobre aquellos poemas que fueron realizados a lo largo del 2003 en las, entonces 81, estaciones de Cercanías de Renfe de la Comunidad de Madrid.

## Índice

Lejanías.....	1
Prólogo.....	1
Nacimiento.....	1
Código.....	2
Libros.....	2
Financiación.....	2
Donativos.....	3
Agradecimientos.....	3
Estructura del programa.....	3
Página index.html.....	3
Página lejanias.html e indice_lejanias().....	4
Función libro().....	4
Foema.....	5
Base de datos.....	5
Anexo I. lejanias.js.....	6
Anexo II. datos.js.....	19

## Prólogo

Por fin cercanías a Colmenar Viejo. Allí viven mis padres, a quienes va dedicado este libro. ¿Estaremos entonces, ahora, más cerca? Llego a la estación y pienso ¿Cómo sería un libro hecho en todas las estaciones de tren de Madrid, empezando en esta? ¿Me sentiría más cerca de lo que me rodea?

Cercanías o Lejanías: sólo depende de uno mismo. El tren no me ha acercado, pero el paso del tiempo compartido, el afecto incondicional, la comprensión y aceptación mutua, sí lo han hecho.

Te invito a que entrelaces tu camino entre los míos, a que visites este libro con lo que tú quieras que contenga, los lugares dónde detenerse y la forma en qué mirar o a que veas el recorrido que yo hice buscando poesía.

Este libro mío lo haces tú: acercándonos.

## Nacimiento

Lejanías es un proyecto que nace de la necesidad de demostrar que la distancia no existe y al mismo tiempo existirá siempre, que los medios de comunicación no comunican ni los medios de transporte transportan: seguimos siendo los humanos quienes hemos de comunicar o transportarnos... los medios son sólo eso, los que están en medio.

En el 2003 manejo la idea de recorrer todas las estaciones de tren de la comunidad de madrid acercándome a todos esos lugares que desconozco y gastando tiempo en ellos, realizando al menos 3 poemas y 3 fotografías. Le dedico Noviembre y Diciembre de ese año a la fase, llamémosla, recolectora, visitando casi azarosamente las 81 estaciones que en aquel entonces

formaban parte de la red de cercanías de Madrid, con libretas hechas con papeles reutilizados que voy haciéndome a cada día con una grapadora.

Después de Colmenar Viejo, no sé si casualmente, la siguiente estación en mi recorrido cronológico fue la UAM, tierra donde estudié, y, la tercera, Alcobendas, donde vivió mi primera novia digna de llamarse tal.

Entre Enero del 2004 y Mayo del 2004 paso a máquina casi todos los poemas textuales y comienzan las dudas acerca de cómo manejar los poemas más gráficos.

Una vez recopilado el material, me veo en la tesitura de presentarlo como libro de poemas y, desde muy el principio, veo que el tema del formato va a ser decisivo. Lo que también era deseable desde el principio era la posibilidad de recorrerlo de distintas maneras, para que cada persona pueda encontrar su propio camino, decidir dónde detenerse y qué mirar.

## Código

La cantidad acumulada de algo que he venido en llamar foemas es abrumadora y pienso, por momentos, en que necesitaría una base de datos o algo así, pero no me convence hacerlo dependiente de una instalación de un programa. Está claro que el libro-multimedia irá en CD, pero sé poco más. Entre unas cosas y otras, se perfila una decisión técnica que va acompañada de consecuencias tecnológicas: será un libro-web que pueda ser accesible desde cualquier lugar sin más que un navegador.

Pero a pesar de mi experiencia en el medio, yo soy un dinosaurio de estas cosas, alguien que sigue tecleando código HTML en un editor que el resto del mundo considera infernal: el VI. Me hago con una versión algo friki para mi sofisticado windows y me doy cuenta de todo lo que no sé.

Requiero conocer cosas que se llaman hojas de estilo o CSS y no tengo ni la más remota idea de qué es eso, así que a aprender. Practico con unas cuantas webs como la de Clave 53 o cuando le regalé la suya a mi amor, Carmen de la Rosa antes de que Lejanías pueda ver la luz, pero siempre pensando en ello. Más evidente aún era la proximidad de este libro cuando hice mi primera exposición de fotografía distribuida o el Proyecto Silla (Deconstrucción), con bastante código JavaScript para gestionar la información almacenada en arrays de objetos en local sin tener que hacer miles de páginas estáticas, ni consultas a un servidor.

Puede que esta morralla tecnológica se quede obsoleta en breve, pero eso dijeron del VI y yo sigo adorándolo.

Te puedes descargar el código con el que funciona este proyecto. Está comentado e incluye información adicional sobre la estructura interna de Lejanías.

## Libros

Lejanías no es un libro: es cada uno de los que tú quieras hacer. Cada día tendría una lectura posible y esa idea es la que motivó su desarrollo como HTML con JavaScript como te explico en la sección "Código".

No obstante, recopilé por formato los distintos poemas que pueden ser impresos y dio lugar a varios subproductos en forma de libros, pero llevan la siguiente advertencia que te hago:

Este libro no es un libro: Es un subproducto derivado del proyecto Lejanías que incluye poesía textual, visual, acciones, vídeos y fotografía. Lejanías está disponible de modo gratuito en [www.giuseppe.net](http://www.giuseppe.net)

Puedes solicitar ejemplares en papel de las distintas colecciones en: <http://giuseppe.bubok.com>

## Financiación

Lo he hecho por vicio, por puro vicio onanista. Podría decir que por necesidad o por capricho. Lo he hecho porque quise y de la manera que quise. Jamás pensé en reducir costes o en justificar gastos. Nunca pensé en su repercusión ni en su posible utilidad pública o social. Me llevó el tiempo que quise o el que le pude ir dedicando.

En algún momento me sugirieron hablar con el departamento de marketing de RENFE para contar con la compañía como patrocinadora, pero de alguna manera me resultaba contrario al sentido del proyecto: En primer lugar, no son los CERCANIAS los que nos mantienen cerca (ni lejos), así que el nombre y la referencia era sólo casual, aunque origina el proyecto; en segundo lugar, en varias estaciones tuve problemas de acceso para fotografiar e, incluso, escribir, sin autorización. Ahora, terminado el trabajo, no la quiero.

¿Por qué su gratuidad? ¿Es Lejanías un manifiesto en contra de la retribución del trabajo artístico o poético? Definitivamente, no. Contesto subscribiendo la frase de Isidoro Valcárcel Medina: El arte es una acción personal, que puede valer como ejemplo pero nunca tener un valor ejemplar.

Esto no es óbice para que me resulte interesante plantear una reflexión acerca de la subsistencia de los poetas y de su trabajo, que no veo mejor resuelto, hoy día, que en tiempos de Maiakovski. Parecemos abocados a vivir de la caridad institucional, de mecenazgos más o menos razonables (en ocasiones disacordes con la naturaleza misma del proyecto en cuestión) o a la autofinanciación más absoluta, lo que equivale a la desprofesionalización completa del artista o poeta.

## Donativos

¿Es Lejanías un proyecto gratuito?: Sí.

¿Puedo, no obstante, ayudar a su autor para que ponga en marcha futuros proyectos o bien agradecerle, económicamente, su proyecto?: Sí.

En un intento de volver a convertir el mecenazgo en una actividad individual, te insto a ser uno de mis colaboradores económicos, con la cantidad libre que tú decidas aportar.

Aun teniendo en cuenta lo dicho en la sección "Financiación", contemplo la posibilidad de aceptar Donativos a través de Paypal.

## Agradecimientos

A mis padres, a quienes va dedicado el libro, les debo tanto que sería impagable.

A Carmen, por sus consejos estéticos, su soporte constante, su amor, su paciencia, su comprensión, sus besos...

A los performers que realizaron algunas de las acciones que contiene el proyecto en formato vídeo, grabadas por mí o, en ocasiones, por ellos mismos: Carmen de la Rosa y Laura Beneitez, Beatriz Pagés y Toni Cárdenas, Paloma Calle, María Ginzo, Aída B. Márquez, Jana de Luque, Clara Graciolino.

A ti, lector, que lo haces posible.

## Estructura del programa

La primera página de inicio, index.html, contiene la imagen portada de los libros editados, así como la carátula de los Cds. Es una simple combinación de la L de Lejanías, con la C invertida de Cercanías. Esta combinación expresa la idea básica del proyecto que viene a decir que la proximidad nada tiene que ver con el convencional significado aportado a la palabra cercanía. Los colores, extraídos de la web de renfe.es, son con los que se ha realizado toda la imagen de Lejanías, pero sobre fondo negro, en lugar de blanco.

Esta página, reenvía a la página index.html de la carpeta base contenedora del proyecto web Lejanías, dividida en las secciones: Prólogo, Nacimiento, Código, Libros, Financiación, Donativos, Agradecimientos e Inicio.

## Página index.html

Además de llamar a algunas funciones definidas en el archivo lejanias.js con la intención de conocer el entorno del cliente, como info\_navegador() y resolucion(), llama a la función **ventanacompleta(lejanias.html)** que abre la ventana wlejanias.

Tras este paso, la navegación está realizada, casi por completo, bajo el gobierno de las funciones contenidas en lejanias.js.

Si no está activado JavaScript en el navegador, en este momento aparece un enlace a una página que nos dice que es preciso tenerlo activo y la manera de activarlo (javascript.html). Es difícil mantener esa página actualizada, porque los navegadores van cambiando más rápidamente de lo deseable.

## Página `lejanias.html` e `indice_lejanias()`

`Lejanias.html` tan solo contiene una llamada a la función **`indice_lejanias()`**. `indice_lejanias()` es la función de entrada en este programa (`lejanias.js`) desde la página `lejanias.html`. Es la encargada de escribir el formulario (FORM) SELESTACIONES. Con él es con quien el lector va a poder seleccionar su recorrido de estaciones y tipos de foemas.

Crea el formulario utilizando las funciones

- `crea_checktipos()`: Escribe los checks para seleccionar los tipos. Usa el método del `crea_checktipo()` del objeto `tipof poema`; `tipof poema.crea_checktipo()`.
- `crea_checkfechas()`: En caso de haber elegido la visita cronológica, escribe los checks de las fechas disponibles, aquellas en las que escribí los foemas, mediante la función `crea_checkcalendario()`, sobre el mes de Noviembre y Diciembre de 2003.
- `crea_checkmapa()`: En caso de desear realizar un recorrido propio, utilizando el método `crea_checkmap()` del objeto `estacion` (`estacion.crea_checkmap()`), escribe los checks sobre el mapa esquemático de las líneas de cercanías de la comunidad de Madrid, vigente durante noviembre de 2003, fondo de la capa "mapaestaciones".
- `crea_checkestaciones()`: Esta función sólo tiene utilidad nostálgica, pues ya no se utiliza. Creaba los checks de las estaciones con el método del objeto `estacion` `estacion.crea_checkest()`. Las estaciones aparecen en orden, pero no sobre el mapa.

Por último, escribe las herramientas de navegación, asignando la función **`libro()`** al comando "seguir", representado por un icono de un tren avanzando hacia la derecha.

A partir de este momento el libro generado es el resultado de la elección del lector. Va a ser `libro()` quien se encargue de saber qué es lo que ha pedido y crearlo para la ocasión.

## Función `libro()`

`libro()` Se encarga de las dos tareas más importantes del proyecto, saber qué quiere el lector y dárselo.

En primer lugar, las funciones "get" (`get_checktipos()`, `get_checkdia()` y `get_checkestaciones()`) se encargan de revisar el formulario SELESTACIONES detectando qué seleccionó el lector.

Esa información la almacena en los arrays `recorrido_tipos[]` y `recorrido[]` que indican que tipos de foema y qué estaciones, respectivamente, se eligieron.

En segundo lugar, a partir de estas dos listas, genera la lista de foemas que el lector ha elegido mediante la función **`crea_foemas()`** en lo que es el verdadero corazón del proyecto. Almacena esta lista en el array `foemas[]` que debe permanecer visible para otras páginas generadas a partir de aquí.

Por último, **`escribe_foemas()`** se ocupa de escribir los enlaces en el libro seleccionado en lo que vendría a ser un índice a la carta, que nos dirige a las distintas páginas posibles, cada una de las cuales tendrá que seguir permitiéndonos avanzar.

Para ello llama a las siguientes funciones:

- **`texto_cabecera()`**: Genera el contenido de cabecera necesario para una nueva página HTML.
- **`recrea_foemas()`**: Inserta en la página nueva la creación de los objetos del array `foemas[]`. Deben estar disponibles para las funciones que deban consultarlos.
- **`indice_de_foemas()`**: En función de la lista almacenada en `foemas[]` crea los enlaces que apuntan a los distintos foemas del libro resultante. Puede

comenzarse a leer. Cualquiera de ellos es abierto en una ventana nueva mediante la función `ventanafoema()`.

- o **texto\_cabecera()**: Por último, cierra el documento HTML.

**ventanafoema()**: es la entrada a la lectura de los foemas desde el índice generado en la fase anterior. Se encarga de abrir una ventana conteniendo dos partes: la superior, para mostrar los foemas, y la inferior, para la paginación, con la página `recorrido.html`.

La paginación se realiza con las funciones `crea_recorrido()`, llamada desde `recorrido.html`, que en primer lugar muestra en la ventana de foemas aquel desde el que se llama, y `haz_recorrido()`, para abrir al foema anterior, el siguiente o, simplemente, cerrar la ventana.

## Foema

Es el ente menor, que no por ello menos importante, que maneja el proyecto. Consiste en un objeto con los atributos: Estación, Tipo, Identificador y URL. Es, por así decir, el vínculo entre la escritura y la informática, es por donde agarro los poemas, los convierto en enlace y los muestro en pantalla. Conseguir una serie de ellos elegidos por ti en función de los criterios mencionados arriba y manejarlos en las páginas web generadas dinámicamente es lo que ha constituido la tarea esencial de este proyecto.

Todo foema tiene una página HTML, que puede ser una plantilla (todos los visuales, fotografía o vídeo lo son) o contener el texto del mismo (para los foemas textuales y las partituras de acción). La estructura de nombres y carpetas que los alojan es la que contiene la información que permitirá asociar su ubicación completa, o url, a datos sobre el mismo, como qué estación fue en la que se realizó, la fecha y hora y el tipo de foema del que se trata.

`foemas/EE/EEMMDDHHTII.html`

- El código EE es un par de dígitos único para cada estación y correspondiente a su orden alfabético.
- MM es 11 para noviembre y 12 para diciembre. DD es el código para el día, con 2 dígitos. HH, para la hora aproximada de realización del foema. No es un dato que esté siendo usado ahora mismo, pero, en versiones ulteriores, podría ser.
- T es un código numérico que identifica el tipo de foema del que se trata, pudiendo ser textual, visual, fotográfico, partitura de acción o vídeo.
- II es el identificador, también de 2 dígitos, de ese foema en concreto.

## Base de datos

Estaciones y tipos son dos listas o arrays de objetos que, respectivamente, guardan información referente a las estaciones o a los distintos tipos de foemas que puede haber. Son consultadas por la mayoría de las funciones para obtener datos o mostrarlos. Son, por decirlo de alguna manera, una base de datos simple definida en el fichero `datos.js` que las construye posiblemente más veces de las necesarias.

Cada objeto estación tiene la siguiente información: Nombre, Código, Fecha(MMDD), Hora (La Hora no se usa, pero quizá algún día), Tipo de Posición y Posición sobre el mapa y un array con el Número de Fotos, Acciones, Vídeos, Visuales y Textuales (en ese orden)

Cada objeto tipo tiene la siguiente información: Código, Nombre del Tipo, Ubicación de los archivos de ese tipo y Extensión.

Fechas es un array de objetos de datos para la presentación cronológica del libro.

Cada objeto Fecha tiene un Identificador, Mes, Día y un array con los códigos de las estaciones de ese día (en orden cronológico).

## Anexo I. lejanias.js

```
/*
Archivo JavaScript Principal de Lejanías.

Este archivo se divide en varias partes:
1.- Definición de Atributos y Métodos para los OBJETOS
Estaciones, Tipos, Fechas y Foemas
2.- Funciones de Manejo y Creación del formulario SELESTACIONES
3.- Funciones para la Creación y Gestión de las páginas dinámicas
de los Foemas.
4.- Funciones Auxiliares para crear los Recorridos en ventanas
separadas, que se abren con los foemas seleccionados.
5.- Funciones de traza_ utilizadas durante el desarrollo.
6.- Funciones de interés general, ventanas, navegador...
*/

// Construcción de OBJETOS ESTACIONES
// Construcción de OBJETOS ESTACIONES
function url_estacion() {
    // Nos forma la URL del Foema antes de encontrarse con los valores
    // correspondientes al tipo y número del foema.
    var url_est='foemas/'+this.IdEst+'/'+this.IdEst+this.Fecha+this.Hora;
    return (url_est);
}
function lugarfecha() {
    // Nos forma la una cadena con el Lugar y la Fecha de creación del Foema
    var lyf=this.Nombre+', 2003'+this.Fecha;
    return (lyf);
}
function crea_checkest() {
    // Nos escribe en el documento un checkbox para la estacion
    document.writeln('<div class=checkestacion
style="position:relative;width:150px;height:50px;">');
    document.writeln('<input type="checkbox" name="chk_est"
value="'+this.IdEst+'">'+this.Nombre);
    document.writeln('</div>');
}
function crea_checkmap() {
    // Nos escribe en el documento un checkbox sobre el mapa para la estacion
    document.write('<div class=checkestacion style="'+this.PosicionMapa+'">');
    if (this.TipoPos==0) document.write('<input type="checkbox" name="chk_est"
value="'+this.IdEst+'">'+this.Nombre);
    else if (this.TipoPos==1) document.write(this.Nombre+'<input type="checkbox"
name="chk_est" value="'+this.IdEst+'>');
    else if (this.TipoPos==2) document.write('<input type="checkbox" name="chk_est"
value="'+this.IdEst+'><br>'+this.Nombre);
    else if (this.TipoPos==3) document.write(this.Nombre+'<br><input type="checkbox"
name="chk_est" value="'+this.IdEst+'>');
    document.write('</div>');
}
function e_props() {
    // Escribe en un formato legible (para trazas, por ejemplo) las propiedades del
objeto
    var props='IdEst= '+this.IdEst+'\nNombre= '+this.Nombre+'\nFecha= '+this.Fecha;
    return (props);
}
function estacion(Nombre,IdEst,Fecha,Hora,TipoPos,PosicionMapa,TMAX) {
    this.Nombre=Nombre
    this.IdEst=IdEst
    this.Fecha=Fecha
    this.Hora=Hora
    this.TipoPos=TipoPos
    this.PosicionMapa=PosicionMapa
    this.TMAX=TMAX
    // estacion.url_estacion()
    // Devuelve la parte de la URL dependiente de la estacion
    this.url_estacion=url_estacion
    this.lugarfecha=lugarfecha
    this.e_props=e_props
    this.crea_checkest=crea_checkest
    this.crea_checkmap=crea_checkmap
}
```

```

}
// Construccion de Objetos TIPOS
// Construccion de Objetos TIPOS
function crea_checktipo() {
    // Nos escribe en el documento un checkbox para la estacion
    document.writeln('<input type="checkbox" name="chk_tipo"
value="'+this.Id+'"'>'+this.Nombre);
}
function t_props() {
    // Escribe en un formato legible (para trazas, por ejemplo) las propiedades del
objeto
    var props='Id= '+this.Id+'\nNombre= '+this.Nombre+'\nCarpeta=
'+this.carpeta+'\nExtensi3n= '+this.ext;
    return (props);
}
function tipofoema(Id,Nombre,carpeta,ext) {
    this.Id=Id
    this.Nombre=Nombre
    this.carpeta=carpeta
    this.ext=ext
    this.t_props=t_props
    this.crea_checktipo=crea_checktipo
}
// Construccion de Objetos FOEMAS
// Construccion de Objetos FOEMAS
function f_props() {
    // Escribe en un formato legible (para trazas, por ejemplo) las propiedades del
objeto
    var props='IdEst= '+this.IdEst+'\nIdTipo= '+this.IdTipo+'\nIdFoema=
'+this.IdFoema+'\nURL= '+this.url;
    return (props);
}
function f_unido() {
    var aux = this.IdEst+', '+this.IdTipo+', '+this.IdFoema+', \''+this.url+'\'';
    return (aux);
}
function foema(IdEst,IdTipo,IdFoema,url) {
    this.IdEst=IdEst
    this.IdTipo=IdTipo
    this.IdFoema=IdFoema
    this.url=url
    this.f_props=f_props
    this.f_unido=f_unido
}
// Construccion de Objetos FECHA
// Construccion de Objetos FECHA
function fech_exacta() {
    var aux='Día '+this.dia+' de '+mes_str(this.mes-1)+' de 2003';
    return (aux);
}
function fech_props() {
    // Escribe en un formato legible (para trazas, por ejemplo) las propiedades del
objeto
    var props='Fecha Id: '+this.Id+'Fecha: '+this.fech+'\nRecorrido de esa fecha:
'+this.rec.join();
    return (props);
}
function fecha(Id,mes,dia,rec) {
    this.Id=Id
    this.mes=mes
    this.dia=dia
    this.rec=rec
    if (dia<10) {var aux="0"+dia} else {var aux=dia};
    this.fech=""+mes+aux;
    this.f_input='<input class=fecha type="radio" name="chk_fecha" value="'+Id+'"'>';
    this.fech_props=fech_props
    this.fech_exacta=fech_exacta
}

// Funciones de Gestión de los FOEMAS seleccionados en el array foemas[]
// Funciones de Gestión de los FOEMAS seleccionados en el array foemas[]

// recrea_foemas()
// sirve para Escribir el Array de foemas[] en cada ventana que abra
// Llamada desde escribe_foemas()
function recrea_foemas() {
    var aux = "";

```

```

document.writeln('<script>');
for (var i_foemas=0; i_foemas<foemas.length; i_foemas++) {
    aux = foemas[i_foemas].f_unido();
    document.writeln('foemas['+i_foemas+']=new foema('+aux+');');
}
document.writeln('</script>');
}

// indice_de_foemas()
// escribe enlaces que para los elementos de foemas[] separándolos por estaciones
// Llamada desde escribe_foemas()
function indice_de_foemas() {
    var flink = "", bolita="";
    var ultima_estacion = 0;
    var num_estacion = 0;

    document.writeln('<div class=caja_foemas>');

    for (var i_foemas=0; i_foemas<foemas.length; i_foemas++) {
        if (foemas[i_foemas].IdEst != ultima_estacion) {
            ultima_estacion = foemas[i_foemas].IdEst;
            if (num_estacion++ != 0) document.writeln('</div></div>'); //
Cerramos caja_estacion
            document.writeln('<div class=caja_estacion>');
            document.writeln('');
            document.writeln('<div class=nombre_estacion>');
            document.writeln(estaciones[ultima_estacion].Nombre);
            document.writeln('</div>'); // Cerramos nombre_estacion
            document.writeln('<div class=foemas_estacion>');
        }
        bolita="img/bolitatipo'+foemas[i_foemas].IdTipo+'.gif'";
        flink='<img class=bolita_foema src=' + bolita + '
onClick="ventanafoema('+i_foemas+');">';
        document.writeln(flink);
    }
    if (num_estacion != 0) document.writeln('</div></div>'); // Cerramos última
caja_estacion

    document.writeln('</div>'); // Cerramos caja_foemas
}

// Funciones get_check... para recoger los checks de día, tipos o estaciones
// llamadas desde libro(is_krono) (tipos y (día o estaciones))
// Se encargan de llenar los arrays recorrido_tipos y recorrido
// que se usan en crea_foemas() generando el array foemas[]

// get_checkdia()
// Función que devuelve el valor del Día seleccionado
function get_checkdia() {
    var dia=-1, i=0;
    for (i=0; i < document.SELESTACIONES.chk_fecha.length; i++) {
        // Si el DIA está seleccionado, añadirlo al recorrido
        if (document.SELESTACIONES.chk_fecha[i].checked) {
            dia = document.SELESTACIONES.chk_fecha[i].value;
        }
    }
    if (dia < 0) alert('No has seleccionado Ninguna Fecha');
    return (dia);
}

// get_checktipos()
// Función que genera un array con los tipos seleccionados
function get_checktipos() {
    var j=0, i=0;
    for (i=0; i < document.SELESTACIONES.chk_tipo.length; i++) {
        // Si el TIPO está seleccionado, añadirlo al recorrido_tipos
        if (document.SELESTACIONES.chk_tipo[i].checked) {
            recorrido_tipos[j] = document.SELESTACIONES.chk_tipo[i].value;
            j++;
        }
    }
    if (j == 0) {alert('No has seleccionado Ningún Tipo'); return (j); }
    return (recorrido_tipos.length);
}

// get_checkestaciones()
// Función que recorre todos los checkboxes del indice y extrae de los

```



```

// marcados el recorrido seleccionado.
// Las índices de las estaciones marcadas se almacenan en recorrido[]
function get_checkestaciones() {
    var i=0;
    var i_est=0;

    for (i=0; i < document.SELESTACIONES.chk_est.length; i++) {
        // Si la ESTACION está seleccionada, añadirla al recorrido
        if (document.SELESTACIONES.chk_est[i].checked) {
            recorrido[i_est] = eval(document.SELESTACIONES.chk_est[i].value);
            i_est++;
        }
    }
    if (i_est == 0) {alert('No has seleccionado Ninguna Estación'); return (i_est); }
    return (i_est);
}

// crea_foemas()
// Función que rellena el array global foemas[] a partir de los arrays
// Los índices de las estaciones marcadas se almacenan en recorrido[]
// Los índices de los tipos marcados se almacenan en recorrido_tipos[]
function crea_foemas() {
    var i=0, num_estacion, num_tipo, max_tipo;
    var i_est=0, i_tipos=0, i_foemas=0;
    var url_estacion="", url_foema="", url_esttipo="";
    var aux;

    for (i_est=0; i_est < recorrido.length; i_est++) {
        num_estacion = recorrido[i_est];
        url_estacion = estaciones[num_estacion].url_estacion();
        for (i_tipos=0; i_tipos < recorrido_tipos.length; i_tipos++) {
            num_tipo = recorrido_tipos[i_tipos];
            url_esttipo = url_estacion + num_tipo;
            max_tipo = estaciones[num_estacion].TMAX[num_tipo-1];

            if (TRAZA>=4) traza_tipos_estacion(num_estacion, num_tipo);

            // Formar el Array de FOEMAS...
            // FOR para el TIPO num_tipo de la ESTACION num_estacion
            for (i=1; i<=max_tipo; i++) {
                // Ha de ser de 2 cifras al convertirse en cadena
                if (i<10) {aux = "0" + i; } else aux = "" + i;
                url_foema = url_esttipo + aux + '.html';
                foemas[i_foemas] = new foema(num_estacion, num_tipo, i,
url_foema);
                i_foemas++;
            }
        }
    }
    if (i_foemas == 0) {alert('No hay Foemas para la Selección Actual: Prueba de
nuevo'); return (i_foemas); }
    else if (TRAZA>=3) alert('Foemas Totales: ' + i_foemas);

    if (TRAZA>=4) traza_foemas();

    return (i_foemas);
} // Fin de crea_foemas()

// Funciones de generación de la página HTML
// resultante tras la elección del recorrido en el índice
// La función principal es escribe_foemas()

// texto_cabecera()
// Crea la Cabecera y el final del documento base en función del argumento es_inicio
function texto_cabecera(es_inicio) {
    if (es_inicio) {
        document.writeln('<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">');
        document.writeln('<HTML lang=es><HEAD><TITLE>'+TITULO+'</TITLE>');
        document.writeln('<META http-equiv=Content-Type content="text/html; charset=iso-
8859-1">');
        document.writeln('<link rel="shortcut icon" href="favicon.ico">');

        document.writeln('<script type="text/javascript"
src="js/lejanias.js"></script>');
    }
}

```

```

        document.writeln('<link rel="stylesheet" href="css/lejanias.css"
type="text/css">');

        document.writeln('</head><body>');
    } else {
        document.writeln('</body></html>');
        document.close();
    }
}

// leyenda_tipos()
// Genera una cajita con la leyenda de los tipos de los foemas en pantalla.
function leyenda_tipos() {
    var bolita, flink;

    document.writeln('<table class=tabla_tipos>');
    for (var i=1; i<tipos.length; i++) {
        bolita='img/bolitatipo'+tipos[i].Id+'.gif';
        flink='<img src=' + bolita + '>'+tipos[i].Nombre;
        document.writeln('<tr><td>'+flink+'</td></tr>');
    }
    document.writeln('</table>');
}

// escribe_foemas()
// Función que genera la página con el recorrido seleccionado
// almacenado en el array global foemas[]
function escribe_foemas(losfoemas) {
    texto_cabecera(true);

    // Introducimos losfoemas como variable de la función
    // y reescribimos el array global con este
    foemas=losfoemas;

    // Escribe los constructores de Foemas para el libro seleccionado
    // en la página web generada, estando así disponibles para esta ventana
    recrea_foemas();

    // Escribe la estructura de la parte de arriba de la página
    // incluyendo imagen de cerrar y título en pantalla
    indice_cabecera(true, 'libro');

    // Escribe una cajita con la leyenda de los tipos de foemas
    leyenda_tipos();

    // Genera los enlaces de los distintos foemas
    indice_de_foemas();

    indice_cabecera(false, 'libro');

    document.writeln('<div class="caja_pie">');
    document.writeln('<hr class=barra_titulo>');
    var frase='Has elegido este libro a las '+hora_actual()+ ' del '+fecha_actual();
    document.writeln('<div class=lugar_fecha>'+frase+'</div>');
    document.writeln('</div>'); // Fin de caja_pie

    texto_cabecera(false);
}

// libro(is_krono)
// Llama a la función que genera el recorrido de foemas seleccionados
// Llama a la función que genera la página web con el recorrido
function libro(is_krono) {
    // Llamamos a la función que rellena el array recorrido_tipos[]
    // en función de los tipos seleccionados
    if (get_checktipos() == 0) return (0); // No seguimos si no hay seleccion

    if (is_krono) {
        var dia_elegido=get_checkdia();
        // Asignamos el recorrido[] de estaciones al array fecha.rec[]
        recorrido=fechas[dia_elegido].rec;
    } else {
        // Llamamos a la función que rellena el recorrido[] de estaciones
        if (get_checkestaciones() == 0) return (0); // No seguimos si no hay
seleccion
    }
}

```

```

// Llamamos a la función que genera el contenido del
// array global foemas[] en función de recorrido[] y recorrido_tipos[]
var i_foemas = crea_foemas();
if (i_foemas == 0) return (i_foemas); // No seguimos si no hay foemas

// Creamos la variable local losfoemas, la igualamos a la global
var losfoemas=new Array();
losfoemas=foemas;

// Llamamos a la función que genera la página web con el recorrido de foemas
// seleccionado y almacenado en el array global foemas[]
escribe_foemas(losfoemas);

if (TRAZA>=2) alert ('El Libro está terminado');
// window.location.reload(); // Para solucionar un problema de repintado CAMBIAR
}

// crea_checkmapa()
// Crea los CheckBoxes sobre el mapa usando el método de estaciones[]
function crea_checkmapa() {
    var i=1;
    var num=estaciones.length-1;
    for (i=1;i<=num;i++) estaciones[i].crea_checkmapa();
    return(num); // Devuelve el número de CheckBoxes creados
}

// crea_checkestaciones()
// Crea los CheckBoxes de las estaciones usando el método de estaciones[]
function crea_checkestaciones() {
    var i=1;
    var num=estaciones.length-1;
    for (i=1;i<=num;i++) estaciones[i].crea_checkest();
    return(num); // Devuelve el número de CheckBoxes creados
}

// crea_checktipos()
// Crea los CheckBoxes de los tipos usando el método de tipos[]
function crea_checktipos() {
    document.writeln('<table class=tabla_tipos>');
    for (var i=1;i<tipos.length;i++) {
        document.write('<tr><td>');
        tipos[i].crea_checktipo();
        document.write('</td></tr>');
    }
    document.writeln('</table>');

    // Marca todos los tipos seleccionados por defecto
    for (i=0;i<document.SELESTACIONES.chk_tipo.length;i++) {
        document.SELESTACIONES.chk_tipo[i].checked=true;
    }

    return(tipos.length-1); // Devuelve el número de CheckBoxes creados
}

// crea_checkfechas() y crea_checkcalendario()
// Crean los CheckBoxes de las fechas usando el método de fecha[]
IND_FECHA=0;
// fecha_ini ha de ser lunes, fecha_fin ha de ser domingo.
function crea_checkcalendario(mes, fecha_ini, fecha_fin) {
    var dia_semana=0;
    var cadena_semana, cadena_dia;
    document.write('<tr><th colspan=7>'+mes+'</th></tr>');
    for (var i_fecha=fecha_ini; i_fecha<=fecha_fin; i_fecha++) {
        if ((dia_semana % 7) == 0) { // LUNES
            cadena_semana='<tr>';
        }
        cadena_dia='<td>';
        // Si el día es uno de los que hice foemas...
        if (i_fecha == fechas[IND_FECHA].dia) {
            cadena_dia=cadena_dia+fechas[IND_FECHA].f_input;
            if (IND_FECHA < (fechas.length-1)) IND_FECHA++;
        }
        cadena_dia=cadena_dia+'<div class=fecha_calendario>'+i_fecha+'</div></td>';
        cadena_semana=cadena_semana+cadena_dia;
        if ((dia_semana % 7) == 6) { // DOMINGO
            cadena_semana=cadena_semana+'</tr>';
            document.write(cadena_semana);
            if (TRAZA >= 3) alert ('Hoy es Domingo '+i_fecha+' de '+mes+' y la
cadena es \n'+cadena_semana);
        }
    }
}

```

```

        dia_semana++;
    }
}
function crea_checkfechas() {
    document.write('<table class=tabla_calendario>');

document.write('<tr><th>L</th><th>M</th><th>X</th><th>J</th><th>V</th><th>S</th><th>D</t
h></tr>');
    document.write('<tr height=20px></tr>');

    crea_checkcalendario('Noviembre', 3, 30);
    crea_checkcalendario('Diciembre', 1, 7);

    document.write('</table>');

    return(fechas.length);
}
// indice_cabecera()
// Crea la Cabecera y el final del documento base en función del argumento es_inicio
function indice_cabecera(es_inicio, llamada) {
    var opcs="";

    if (es_inicio) {
        document.writeln('<div class="caja_cabecera">');
        document.writeln('<div class="volver">');
        if (llamada=='libro') { opcs=' alt="Volver"
onClick="redireccion(\'lejanias.html\')"; }
        else if (llamada=='indice') { opcs=' alt="Cerrar"
onClick="parent.window.close()"; }

        opcs = '<img class=icono_navegacion src=img/cerrar.gif '+ opcs +>';
        document.writeln(opcs);
        document.writeln('</div>');
        document.writeln('<H2 class=titulo_libro>'+TITULO+'</H2>');
        document.writeln('<hr class=barra_titulo>');
        document.writeln('</div>'); // Fin de caja_cabecera

        if (llamada=='libro') { document.writeln('<div class=caja_checks>'); }
        else if (llamada=='indice') { document.writeln('<div
class=caja_checks>'); }
        } else {
            document.writeln('</div>'); // Fin del div caja_checks o caja_foemas
        }
    }

// Función de Entrada en este programa desde LEJANIAS.HTML
// Es la encargada de Crear el formulario SELESTACIONES
// Llama a crea_checkfechas() o crea_checkmapa() según proceda
// También recurre a crea_checktipos()
// Antiguamente, se usaban las funciones crea_checkestaciones() y crea_checks_mapa()
//
// Asigna la función libro(KRONO) al onClick del boton
function indice_lejanias(is_mapa,is_krono) {
    if (TRAZA>=3) alert('MAPA: '+MAPA+'\nKRONO: '+KRONO);

    // BROWSER=browser(); // Todavía no utilizo el valor de BROWSER, pero puede que
luego sí.

    indice_cabecera(true, 'indice');

    if (TRAZA>=3) traza_tipos();
    if (TRAZA>=3) traza_estaciones();

    // Comienzo el FORMULARIO de ESTACIONES y TIPOS (SELESTACIONES)
    document.writeln('<FORM id=form_indice NAME=SELESTACIONES>');

    // Escribo los Checks de los Tipos de foemas para elegir.
    crea_checktipos();

    // Escribo los Checks de Estaciones o Dias según sea cronológico o no
    if (is_krono) {
        document.writeln('<div class=nomapaestaciones>');
        crea_checkfechas(); // Ponemos los checks de las fechas con foemas en el
calendario
        document.writeln('</div>');
        var cadena_boton=' alt="Esto es lo que hice ese día"
onClick="libro('+is_krono+)'";

```

```

        var cadena_leyenda='Elige una fecha para ver el recorrido que hice.';
    }
    else {
        if (is_mapa) {
            document.writeln('<div class=mapaestaciones>');
            crea_checkmapa(); // Ponemos los Checks de las estaciones sobre el
Mapa de la Clase mapaestaciones
            document.writeln('</div>');
        }
        else {
            document.writeln('<div class=nomapaestaciones>');
            crea_checkestaciones(); // Ponemos los checks de las estaciones en
orden... pero sin mapa.
            document.writeln('</div>');
        }
        var cadena_boton=' alt="Crea tu propio Libro"
onClick="libro('+is_krono+')"';
        var cadena_leyenda='Elige las estaciones y tipos de poemas que quieres
leer.';
    }

    // Escribimos la caja_pie con los botones de navegación y la leyenda
    document.writeln('<div class="caja_pie">');
    document.writeln('<hr class=barra_titulo>');
    document.writeln('<div class=botones_formulario>');
    document.writeln('');
    document.writeln('');
    document.writeln('</div>');

    document.writeln('<div class=leyenda>');
    document.writeln(cadena_leyenda);
    document.writeln('</div>');
    document.writeln('</div>'); // Fin de caja_pie

    // Termino el FORMULARIO de ESTACIONES y TIPOS (SELESTACIONES)
    document.writeln('</FORM>');

    indice_cabecera(false, 'indice');
}
// get_foema()
// Función que devuelve el valor del Id del FOEMA
// a partir de su URL (extrayendo la parte relativa, sin la extensión)
function get_foema() {
    var archivo = rutarelativa();
    var posicionPunto = archivo.indexOf(".");
    return(archivo.substring(0,posicionPunto));
}
// html_foema_cabecera(titulo)
// Función llamada desde foema_visual() para formar la cabecera de página
function html_foema_cabecera(titulo) {
    document.write('<div class=titulo_poema>'+titulo+'</div>');
    document.write('<hr class=barra_titulo>');

    document.write('<div class=caja_foema>');
}
// html_foema_pie()
// Función llamada desde foema_visual() para formar el pie de página
function html_foema_pie() {
    var idfoema = get_foema();
    var id_est = idfoema.substring(0,2); // El Índice de la Estacion
    var i_est = eval(id_est); // Para Pasarlo a número para usarlo en estaciones[]

    document.write('</div>'); // Cerramos la caja del Foema
    document.write('<hr class=barra_pie>');
    document.write('<div class=lugar_fecha_poema>');
    document.write(estaciones[i_est].lugarfecha());
    document.write('</div>');
}

// foema_visual()
// Función llamada desde todos los FOEMAS VISUALES (foto, vídeo, visual)
// que genera, en función del nombre del archivo, el contenido.
// Recupera valores de la estación, el tipo, el id del foema.
function foema_visual(titulo) {
    var idfoema = get_foema();

```

```

var id_est =idfoema.substring(0,2); // El Índice de la Estacion
var fechtime=idfoema.substring(2,8); // Fecha y Hora juntos
var id_tipo =eval(idfoema.substring(8,9)); // Tipo del foema
var i_foe =idfoema.substring(9,11); // Índice del foema de este tipo

// Obtengo el índice del Array de tipos[] en función del Id
var num_tipo=0;
while (id_tipo != tipos[num_tipo].Id) { num_tipo++; }

var
archivo=tipos[num_tipo].carpeta+id_est+'/'+id_est+fechtime+i_foe+tipos[num_tipo].ext;

html_foema_cabecera(titulo);

document.write('<div class=poema_foto>');
switch (id_tipo) {
case 1: // foema_foto
document.write('<a href="'+archivo+'" alt="Pincha sobre la foto para
verla a tamaño completo">');
document.write('');
document.write('</a>');
break;
case 3: // foema_video
document.write('<object type="application/x-shockwave-flash"
id="FlowPlayer" data="../../js/player.swf" width="420" height="363">');
document.write('<param name="movie" value="../../js/player.swf">');
document.write('<param name="scale" value="noScale">');
document.write('<param name="wmode" value="transparent">');
document.write('<param name="allowScriptAccess"
value="sameDomain">');
document.write('<param name="quality" value="high">');
document.write('<param name="flashvars" value="config={ loop: false,
autoplay:true, initialScale: \'fit\','>');
// if (TRAZA>=4) alert('PATH del player: ../../js/player.swf \nPATH
del videoFile: '+archivo);
document.write('<videoFile: \''+archivo+'\' , }">');
document.write('</object>');
break;
case 4: // foema_visual
document.write('<a href="'+archivo+'" alt="Pincha sobre el poema
para verlo a tamaño completo">');
document.write('');
document.write('</a>');
break;
}
document.write('</div>');

html_foema_pie();
}

// Otras FUNCIONES ÚTILES
// Otras FUNCIONES ÚTILES
// Función para abrir una ventana con distintas opciones
// Llamada desde index.html para abrir lejanias.html
function ventanacompleta(pagina) {
BROWSER=browser();
var
opciones='toolbar=no,location=no,directories=no,status=no,menubar=no,resizable=no,scroll
bars=no';
if ((BROWSER=='SFRI') || (BROWSER=='CHRM')) {
var ancho=screen.width-30;
var alto =screen.height-80;
opciones=opciones+',width='+ancho+',height='+alto;
// opciones='fullscreen, scrollbars=auto'; // CAMBIAR
}
else if (BROWSER=='MZFF') { opciones='fullscreen'; }
else if (BROWSER=='MSIE') { opciones=opciones+',fullscreen=yes'; }
else { } // Navegador no contemplado... No puedo prever todos los posibles!

wlejanias=window.open(pagina,"wlejanias",opciones);
if ((BROWSER=='SFRI') || (BROWSER=='CHRM')) wlejanias.moveTo(10,0);
wlejanias.focus();
}

function ventanafoema(i_foema) {
FOEMABIERTO=i_foema;
var ancho=900, alto=720;

```

```

var desplazamiento=(screen.width-ancho)/2; // Para centra la ventana

var opciones='scrollbars=no,width='+ancho+',height='+alto;
wfoema=window.open('framesfoema.html','wFoema',opciones);

wfoema.moveTo(desplazamiento,10);
wfoema.focus();
}

// crea_recorrido()
// Función llamada desde el frame recorrido.html y que crea el recorrido con siguiente,
etc...
// Podría estar todo su código dentro de la página html, pero así es más bonito.
function crea_recorrido() {
    // Recojo el array de foemas[] de la ventana anterior y el Foema de comienzo.
    foemas=parent.window.opener.foemas;
    FOEMABIERTO=parent.window.opener.FOEMABIERTO;

    // Nos sitúa en la ventana de los Foemas el FOEMABIERTO
    parent.window.document.getElementById('vframe').src=foemas[FOEMABIERTO].url;

    // En el OnClick modifíco el this.HREF hacia el siguiente o el anterior con la
funcion
    // haz_recorrido(direccion) que modifica el FOEMABIERTO
    document.write('<div class=indice_recorrido>');
    document.write('<a href="nada.html" id=href_anterior
onClick="this.href=haz_recorrido(\'anterior\');" target=ventana>');
    document.write('<img class=icono_navegacion src=img/anterior.gif
alt="Anterior"></a>');
    document.write('<a href="nada.html" target=ventana
onClick="parent.window.close()">');
    document.write('<img class=icono_navegacion src=img/cerrar.gif
alt="Cerrar"></a>');
    document.write('<a href="nada.html" id=href_siguiete
onClick="this.href=haz_recorrido(\'siguiete\');" target=ventana>');
    document.write('<img class=icono_navegacion src=img/siguiete.gif
alt="Siguiete"></a>');
    document.write('</div>');
}

// haz_recorrido(direccion)
// Función llamada desde la función crea_recorrido() para ir al anterior foema o al
siguiete
// modificando el valor de FOEMABIERTO y devolviendo la URL del nuevo foema.
function haz_recorrido(direccion) {
    if (direccion == "anterior") {
        FOEMABIERTO--;
        if (FOEMABIERTO < 0 ) FOEMABIERTO = foemas.length - 1; // Para no ser
negativo
    }
    else if (direccion == "siguiete") {
        FOEMABIERTO++;
        if (FOEMABIERTO >= foemas.length) FOEMABIERTO = 0; // Para no ser excesivo
    }
    else alert("Imposible ir en esa dirección: " + direccion);

    if (TRAZA>=4) alert('URL del foema nuevo['+FOEMABIERTO+']: ' +
foemas[FOEMABIERTO].url);
    // Devuelvo la URL del nuevo FOEMABIERTO
    return (foemas[FOEMABIERTO].url);
}

// Funciones de TRAZA
// Funciones de TRAZA
// traza_foemas()
// sirve para mostrar las propiedades de todos los foemas en el array foemas[]
function traza_foemas() {
    var aux = "";
    for (var i_foemas=0; i_foemas<foemas.length; i_foemas++) {
        aux = foemas[i_foemas].f_props();
        alert('Propiedades del FOEMA['+i_foemas+']\n' + aux);
    }
    alert('Número Total de Foemas' + i_foemas);
    return (i_foemas);
}
// traza_estaciones()

```



```

// sirve para mostrar cuantos foemas hay de cada tipo en las estaciones en el array
estaciones[]
function traza_estaciones() {
    var photos = 0, videos = 0, textos = 0, accion = 0, visual = 0;
    var aux = "";

    for (var i_estaciones=1; i_estaciones<estaciones.length; i_estaciones++) {
        photos = photos + estaciones[i_estaciones].TMAX[0];
        accion = accion + estaciones[i_estaciones].TMAX[1];
        videos = videos + estaciones[i_estaciones].TMAX[2];
        visual = visual + estaciones[i_estaciones].TMAX[3];
        textos = textos + estaciones[i_estaciones].TMAX[4];
        if (TRAZA>=4) {
            aux = estaciones[i_estaciones].e_props();
            alert('Propiedades de la ESTACION['+i_estaciones+']\n' + aux);
        }
    }
    i_estaciones--;
    alert('Número Total de Estaciones: ' + i_estaciones
        + '\nNúmero Total de Foemas: ' + (photos+accion+videos+visual+textos));
    alert('Fotos: ' + photos + '\nAcciones: ' + accion + '\nVideos: ' + videos +
        '\nVisuales: ' + visual + '\nTextuales: ' + textos);
    return (i_estaciones);
}

// traza_tipos_estacion()
// sirve para mostrar cuantos tipos hay y foemas máximos de ese tipo por cada estación
function traza_tipos_estacion(num_estacion, num_tipo) {
    var max_tipo = estaciones[num_estacion].TMAX[num_tipo-1];
    alert('Nombre de la Estacion: ' + estaciones[num_estacion].Nombre
        + '\nNombre del TIPO: ' + tipos[num_tipo].Nombre
        + '\nNúmero del Tipo: ' + num_tipo
        + '\nNúmero de Foemas de ese tipo: ' + max_tipo);
}

// traza_tipos()
// sirve para mostrar las propiedades de todas las tipos en el array tipos[]
function traza_tipos() {
    var aux = "";
    alert('Función de TRAZA de Tipos');
    for (var i_tipos=1; i_tipos<tipos.length; i_tipos++) {
        aux = tipos[i_tipos].t_props();
        alert('Propiedades del TIPO['+i_tipos+']\n' + aux);
    }
    i_tipos--;
    alert('Número Total de Tipos' + i_tipos);
    return (i_tipos);
}

// traza_fechas()
// sirve para mostrar las propiedades de todas las fechas en el array fechas[]
function traza_fechas() {
    var aux = "";
    alert('Función de TRAZA de fechas');
    for (var i=0; i<fechas.length; i++) {
        aux = fechas[i].fech_props();
        alert('Propiedades de la Fecha '+fechas[i].fech_exacta());
        alert('Propiedades de la Fecha['+i+']\n' + aux);
    }
    i--;
    alert('Número Total de Fechas' + i);
    return (i);
}

// traza_recorrido() y traza_recorrido_tipos() no se usan: BORRAR
function traza_recorrido() {
    alert('Muestra Recorrido con ' + recorrido.length + ' estaciones');
    for (var i=0; i<recorrido.length; i++) alert('recorrido['+i+']= ' +
        recorrido[i]);
}

function traza_recorrido_tipos() {
    alert('Muestra Tipos con ' + recorrido_tipos.length + ' Tipos');
    for (var i=0; i<recorrido_tipos.length; i++) alert('recorrido_tipos['+i+']= ' +
        recorrido_tipos[i]);
}

// Funciones de Propósito General, válidas para otros proyectos
// Funciones de Propósito General, válidas para otros proyectos
// Función que devuelve una cadena de Mes en funcion del número (de 0 a 11)

```



```

function mes_str(num) {
    var meses = new
Array('Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio', 'Julio', 'Agosto', 'Septiembre', 'Oc
tubre', 'Noviembre', 'Diciembre');
    return(meses[num]);
}
// Función que devuelve una cadena de Día del Mes del Año
function fecha_actual() {
    var fecha = new Date();
    var mes = mes_str(fecha.getMonth());
    return (fecha.getDate() + ' de '+mes+' del '+fecha.getFullYear());
}
// Función que devuelve una cadena de Hora:Minuto
function hora_actual() {
    var ahora = new Date();
    hora = ahora.getHours();
    if (hora<10) hora = "0" + hora;
    minuto = ahora.getMinutes();
    if (minuto<10) minuto = "0" + minuto;
    // segundo = momentoActual.getSeconds();
    // if (segundo<10) segundo = "0" + segundo;

    return (hora + ':' + minuto);
}
// Función para alertarnos sobre la resolución de la pantalla
function resolucion() {
    alert("La resolución de tu pantalla es de: " + screen.width + " x " +
screen.height);
}
// Función para Redireccionar la página a una URL indicada en la variable pagina
function redireccion(pagina) {
    document.location.href=pagina;
    // También se podría hacer con la propiedad location de la ventana
    // window.location = pagina;
    // También el método window.assign(pagina); Podría servir...
}
// Función que muestra información relevante sobre el Navegador que estamos usando
function browser() {
    var x = navigator;
    var br;
    if (x.appName=="Netscape")
    {
        if (x.userAgent.indexOf('Firefox')!=-1) { br="MZFF"; }
        else if (x.userAgent.indexOf('Chrome')!=-1) { br="CHRM"; }
        else if (x.userAgent.indexOf('Safari')!=-1) { br="SFRI"; }
        else br="OTRO";
    }
    else if (x.appName=="Microsoft Internet Explorer") br="MSIE";
    else br="OTRO";

    return (br);
}
function info_navegador() {
    var x = navigator;

    document.writeln("<br>BROWSER= " + browser() + "<hr>");
    document.writeln("<br>CodeName= " + x.appCodeName);
    document.writeln("<br>Name= " + x.appName);
    document.writeln("<br>Version= " + x.appVersion);
    document.writeln("<br>CookieEnabled= " + x.cookieEnabled);
    document.writeln("<br>CPUClass= " + x.cpuClass);
    document.writeln("<br>OnLine= " + x.onLine);
    document.writeln("<br>Platform= " + x.platform);
    document.writeln("<br>JavaEnabled= " + x.javaEnabled());
    document.writeln("<br>UserAgent= " + x.userAgent);
    document.writeln("<br>BrowserLanguage= " + x.browserLanguage);
    document.writeln("<br>SystemLanguage= " + x.systemLanguage);
    document.writeln("<br>UserLanguage= " + x.userLanguage);
    info_pantalla();
}
function info_pantalla() {
    document.writeln('<br>La reolución de la pantalla es la siguiente:');
    document.writeln('<br>Resolución Ancho: '+screen.width);
    document.writeln('<br>Resolución Alto: '+screen.height);
}
// Función que devuelve la parte Relativa del archivo en curso
function rutarelativa() {

```

```

        var rutaAbsoluta = self.location.href;
        var posicionUltimaBarra = rutaAbsoluta.lastIndexOf("/");
        return (rutaAbsoluta.substring(posicionUltimaBarra + "/" ,
rutaAbsoluta.length));
    }
    // Función para mostrar una capa, en función de su ID
    function mostrarcapa(capa){
        document.getElementById(capa).style.visibility="visible";
    }
    // Función para ocultar una capa, en función de su ID
    function ocultarcapa(capa){
        document.getElementById(capa).style.visibility="hidden";
    }
    // Función que no hace nada. Sí, así es. Es una función relajada
    function nada() {}

```

```

////////////////////////////////////
////////

```

```

// Variables GLOBALES de interés general
// Variables GLOBALES de interés general
var foemas=new Array(); // Se llena en crea_foemas()
var recorrido=new Array(); // Se llena en get_checkestaciones()
var recorrido_tipos=new Array(); // Se llena en get_checktipos()

```

```

var url_indice="lejanias.html";
var TITULO="Lejan&iacute;as";
var TRAZA=1;
var wlejanias;
var wfoema;
var BROWSER;
var MAPA=true;
var KRONO=false;
var FOEMABIERTO=0;
var NUM_ESTACIONES=81;
var NUM_TIPOFOEMAS=5;

```

```

////////////////////////////////////
////////

```

```

// Incluimos, por último, los datos de estaciones, tipos y fechas, definidos en datos.js
if (self.location.href.indexOf('foemas')!=-1) { document.write("<script
src='../..js/datos.js' language='JavaScript' type='text/javascript'></script>"); }
else document.write("<script src='js/datos.js' language='JavaScript'
type='text/javascript'></script>");

```

## Anexo II. datos.js

```
// Creación de ARRAYS de Estaciones, TiposFoemas y Foemas.
// Creación de ARRAYS de Estaciones, TiposFoemas y Foemas.
estaciones=new Array();

// Cada estacion tiene los siguientes datos:
// Nombre, Código, Fecha(MMDD), Hora (La Hora no se usa, pero quizá algún día)
// Tipo de Posición (0,1,2,3) y Posicion sobre el mapa
// Número de: Fotos, Acciones, Vídeos, Visuales y Textuales (en ese orden)
estaciones[0]=new estacion("Nombre de la Estación","00","0101","00",0,"",[1,1,1,1,1]);
estaciones[1]=new estacion("Alcalá de
Henares","01","1119","14",0,"top:215px;left:650px;",[5,0,0,3,10]);
estaciones[2]=new estacion("Alcalá de H.
Universidad","02","1110","13",0,"top:200px;left:660px;",[4,0,0,1,8]);
estaciones[3]=new estacion("Alcobendas","03","1103","11",0,"top:20px;left:675px;",
[5,0,0,2,7]);
estaciones[4]=new estacion("Alcorcón","04","1113","11",1,"top:422px;right:642px;",
[5,0,0,4,5]);
estaciones[5]=new estacion("Alpedrete","05","1117","16",0,"top:50px;left:330px;",
[5,0,0,2,6]);
estaciones[6]=new estacion("Aluche","06","1103","16",1,"top:352px;right:572px;",
[4,0,0,0,5]);
estaciones[7]=new estacion("Aranjuez","07","1124","11",0,"top:505px;left:685px;",
[12,0,0,5,4]);
estaciones[8]=new estacion("Aravaca","08","1204","10",1,"top:265px;right:510px;",
[9,0,0,4,7]);
estaciones[9]=new estacion("Atocha","09","1106","16",3,"top:280px;left:510px;text-
align:left",[6,2,2,2,8]);
estaciones[10]=new estacion("Cercedilla","10","1117","14",1,"top:0px;right:607px;",
[16,1,0,3,15]);
estaciones[11]=new estacion("Chamartín","11","1204","12",0,"top:165px;left:510px;",
[10,0,0,6,12]);
estaciones[12]=new estacion("Ciempozuelos","12","1124","11",0,"top:490px;left:685px;",
[7,0,0,2,6]);
estaciones[13]=new estacion("Collado Mediano","13","1117","12",0,"top:25px;left:330px;",
[9,0,0,3,10]);
estaciones[14]=new estacion("Colmenar viejo","14","1103","09",1,"top:0px;right:355px;",
[3,0,0,1,6]);
estaciones[15]=new estacion("Coslada","15","1201","12",0,"top:260px;left:610px;",
[6,2,2,2,11]);
estaciones[16]=new estacion("Cuatro
vientos","16","1103","18",1,"top:394px;right:614px;",[4,0,0,1,6]);
estaciones[17]=new estacion("Delicias","17","1106","10",2,"top:340px;right:445px;text-
align:right",[3,0,0,1,7]);
estaciones[18]=new estacion("Doce de
Octubre","18","1128","14",1,"top:370px;right:440px;",[9,1,1,2,7]);
estaciones[19]=new estacion("El Barrial. CC
Pozuelo","19","1127","12",1,"top:225px;right:510px;",[7,1,1,7,6]);
estaciones[20]=new estacion("El Escorial","20","1120","13",1,"top:45px;right:668px;",
[8,1,1,7,17]);
estaciones[21]=new estacion("El Goloso","21","1110","17",1,"top:55px;right:325px;",
[9,1,0,2,4]);
estaciones[22]=new estacion("El Pozo","22","1201","14",0,"top:320px;left:550px;",
[8,1,0,4,11]);
estaciones[23]=new estacion("El
Tejar","23","1120","16",3,"bottom:360px;right:475px;text-align:right",[6,1,1,6,9]);
estaciones[24]=new estacion("Embajadores","24","1113","10",1,"top:324px;right:544px;",
[4,1,4,1,3]);
estaciones[25]=new estacion("Entrevías","25","1113","16",0,"top:335px;left:535px;z-
index:2",[7,1,2,7,6]);
estaciones[26]=new estacion("Fanjul","26","1204","17",1,"top:366px;right:586px;",
[6,2,2,9,9]);
estaciones[27]=new estacion("Fuencarral","27","1103","12",0,"top:130px;left:550px;",
[3,0,0,0,9]);
estaciones[28]=new estacion("Fuenlabrada","28","1128","11",1,"top:507px;right:513px;",
[10,1,1,5,9]);
estaciones[29]=new estacion("Galapagar-La
Navata","29","1120","10",1,"top:120px;right:570px;",[5,2,0,4,7]);
estaciones[30]=new estacion("Getafe Central","30","1106","15",0,"top:465px;left:510px;",
[4,1,0,1,3]);
estaciones[31]=new estacion("Getafe
Industrial","31","1124","13",0,"top:415px;left:625px;",[9,0,0,3,6]);
estaciones[32]=new estacion("Getafe Sector
3","32","1106","13",0,"top:485px;left:520px;",[3,0,0,0,3]);
```

```
estaciones[33]=new estacion("La Serna","33","1113","14",1,"top:480px;right:495px;",
[4,0,0,2,9]);
estaciones[34]=new estacion("Laguna","34","1204","17",1,"top:338px;right:558px;",
[5,1,1,3,5]);
estaciones[35]=new estacion("Las Aguilas","35","1204","16",1,"top:380px;right:600px;",
[8,2,1,3,10]);
estaciones[36]=new estacion("Las Margaritas-
Universidad","36","1106","14",0,"top:430px;left:480px;width:120px;text-align:left;",
[3,1,0,6,6]);
estaciones[37]=new estacion("Las Matas","37","1127","10",1,"top:150px;right:540px;",
[10,1,1,0,3]);
estaciones[38]=new estacion("Las Retamas","38","1113","13",1,"top:436px;right:656px;",
[5,1,0,1,3]);
estaciones[39]=new estacion("Las Rozas","39","1127","11",1,"top:185px;right:510px;",
[8,0,0,6,7]);
estaciones[40]=new estacion("Las Zorreras","40","1120","12",1,"top:65px;right:620px;",
[8,1,0,3,8]);
estaciones[41]=new estacion("Leganés","41","1128","12",1,"top:460px;right:475px;",
[9,0,0,4,11]);
estaciones[42]=new estacion("Los Molinos","42","1117","13",0,"top:5px;left:330px;",
[5,1,1,2,11]);
estaciones[43]=new estacion("Los Negrales","43","1117","16",0,"top:75px;left:330px;",
[4,0,0,1,9]);
estaciones[44]=new estacion("Majadahonda","44","1127","12",1,"top:205px;right:510px;",
[8,0,0,4,4]);
estaciones[45]=new estacion("Meco","45","1110","12",0,"top:185px;left:670px;",
[9,0,0,3,4]);
estaciones[46]=new estacion("Méndez
Alvaro","46","1106","11",2,"top:340px;left:470px;text-align:left;",[7,0,0,1,6]);
estaciones[47]=new estacion("Móstoles","47","1103","18",1,"top:450px;right:670px;",
[3,4,3,3,6]);
estaciones[48]=new estacion("Móstoles-E]
Soto","48","1103","17",1,"top:473px;right:709px;",[5,0,0,0,2]);
estaciones[49]=new estacion("Nuevos
Ministerios","49","1204","15",0,"top:205px;left:510px;",[12,0,0,6,5]);
estaciones[50]=new estacion("Orcasitas","50","1128","13",1,"top:390px;right:440px;",
[7,0,0,4,8]);
estaciones[51]=new estacion("Parla","51","1106","13",0,"top:505px;left:520px;",
[4,1,0,4,2]);
estaciones[52]=new estacion("Parque de
Ocio","52","1114","11",0,"top:450px;left:775px;width:120px;",[3,0,0,0,3]);
estaciones[53]=new estacion("Pinar","53","1127","10",1,"top:165px;right:525px;",
[5,0,0,0,4]);
estaciones[54]=new estacion("Pinto","54","1114","11",0,"top:430px;left:650px;",
[7,1,0,0,9]);
estaciones[55]=new estacion("Pirámides","55","1103","14",1,"top:340px;right:465px;",
[7,0,0,1,7]);
estaciones[56]=new estacion("Pitis","56","1201","11",3,"bottom:360px;left:430px;",
[6,0,0,2,6]);
estaciones[57]=new estacion("Pozuelo","57","1204","10",1,"top:245px;right:510px;",
[6,4,1,1,5]);
estaciones[58]=new estacion("Príncipe Pío","58","1117","10",1,"top:285px;right:510px;",
[7,2,0,2,9]);
estaciones[59]=new estacion("Puente
Alcocer","59","1128","13",1,"top:410px;right:440px;",[7,3,1,2,7]);
estaciones[60]=new estacion("Ramón y
Cajal","60","1107","09",3,"bottom:360px;left:460px;text-align:left;width:80px;",
[4,1,0,1,8]);
estaciones[61]=new estacion("Recoletos","61","1107","09",0,"top:245px;left:510px;",
[3,1,0,4,7]);
estaciones[62]=new estacion("San Cristobal de los
Angeles","62","1124","14",3,"bottom:105px;left:520px;text-align:left",[5,0,0,0,9]);
estaciones[63]=new estacion("San Cristobal
Industrial","63","1124","13",0,"top:400px;left:610px;",[5,2,2,2,6]);
estaciones[64]=new estacion("San Fernando","64","1110","13",0,"top:245px;left:625px;",
[5,2,2,3,4]);
estaciones[65]=new estacion("San José de
Valderas","65","1113","12",1,"top:408px;right:628px;",[4,0,0,2,5]);
estaciones[66]=new estacion("San Martín de la
Vega","66","1114","11",0,"top:480px;left:785px;width:90px;",[7,0,0,2,4]);
estaciones[67]=new estacion("San Yago","67","1120","11",1,"top:85px;right:600px;",
[10,3,3,3,11]);
estaciones[68]=new estacion("Santa Eugenia","68","1201","14",0,"top:290px;left:580px;",
[9,1,0,4,10]);
estaciones[69]=new estacion("Torrejón de
Ardoz","69","1110","11",0,"top:230px;left:640px;",[6,0,0,2,8]);
```

```

estaciones[70]=new estacion("Torrelodones","70","1117","11",1,"top:135px;right:555px;",
[5,3,3,0,9]);
estaciones[71]=new estacion("Tres Cantos","71","1110","18",1,"top:20px;right:325px;",
[4,0,0,2,6]);
estaciones[72]=new estacion("UAM","72","1103","11",0,"top:100px;left:580px;",
[4,0,0,1,3]);
estaciones[73]=new estacion("UP Comillas","73","1110","16",0,"top:70px;left:610px;",
[4,1,0,1,8]);
estaciones[74]=new estacion("Valdelasfuentes","74","1110","15",0,"top:50px;left:635px;",
[5,0,0,0,7]);
estaciones[75]=new estacion("Valdemoro","75","1124","12",0,"top:460px;left:670px;",
[7,1,0,3,6]);
estaciones[76]=new estacion("Vallecas","76","1110","10",0,"top:305px;left:565px;",
[4,0,0,1,6]);
estaciones[77]=new estacion("Vicálvaro","77","1201","13",0,"top:275px;left:595px;",
[5,0,0,4,8]);
estaciones[78]=new estacion("Villalba","78","1117","17",0,"top:95px;left:330px;",
[5,2,1,4,8]);
estaciones[79]=new estacion("Villaverde
Alto","79","1106","12",1,"top:425px;right:440px;",[4,2,0,0,6]);
estaciones[80]=new estacion("Villaverde
Bajo","80","1124","09",0,"top:370px;left:510px;",[5,1,0,3,7]);
estaciones[81]=new estacion("Zarzaquemada","81","1113","15",1,"top:445px;right:460px;",
[4,1,0,5,4]);

tipos=new Array();
// Cada tipo tiene los siguientes datos:
// Código, Nombre del Tipo, Ubicación del archivo origen y Extensión
tipos[0]=new tipofoema(0,"Tipo de Foema","carpeta","extensión");
tipos[1]=new tipofoema(5,"Poema Textual","../textos/",".txt");
tipos[2]=new tipofoema(4,"Poema Visual","../pvisuales/",".gif");
tipos[3]=new tipofoema(3,"Vídeo","../pvideo/",".flv"); // Relativo al player.swf en js/
tipos[4]=new tipofoema(2,"Acciones","../textos/",".pdf");
tipos[5]=new tipofoema(1,"Fotografía","../fotos/",".jpg");

fechas=new Array();
// Cada Fecha tiene un Identificador, Mes, Día
// y un array con los códigos de las estaciones de ese día (en orden cronológico)
fechas[0]=new fecha(0,11,03,[14,72,3,27,55,6,48,47,16]);
fechas[1]=new fecha(1,11,06,[17,46,79,51,32,36,30,9]);
fechas[2]=new fecha(2,11,07,[61,60]);
fechas[3]=new fecha(3,11,10,[76,69,45,2,64,74,73,21,71]);
fechas[4]=new fecha(4,11,13,[24,4,65,38,33,81,25]);
fechas[5]=new fecha(5,11,14,[66,52,54]);
fechas[6]=new fecha(6,11,17,[58,70,13,42,10,5,43,78]);
fechas[7]=new fecha(7,11,19,[1]);
fechas[8]=new fecha(8,11,20,[29,67,40,20,23]);
fechas[9]=new fecha(9,11,24,[80,7,12,75,31,63,62]);
fechas[10]=new fecha(10,11,27,[37,53,39,44,19]);
fechas[11]=new fecha(11,11,28,[28,41,59,50,18]);
fechas[12]=new fecha(12,12,01,[56,15,77,68,22]);
fechas[13]=new fecha(13,12,04,[8,49,11,57,35,26,34]);

```