



Kepler 8.0

Curso 4 - Elementos Gráficos Kepler

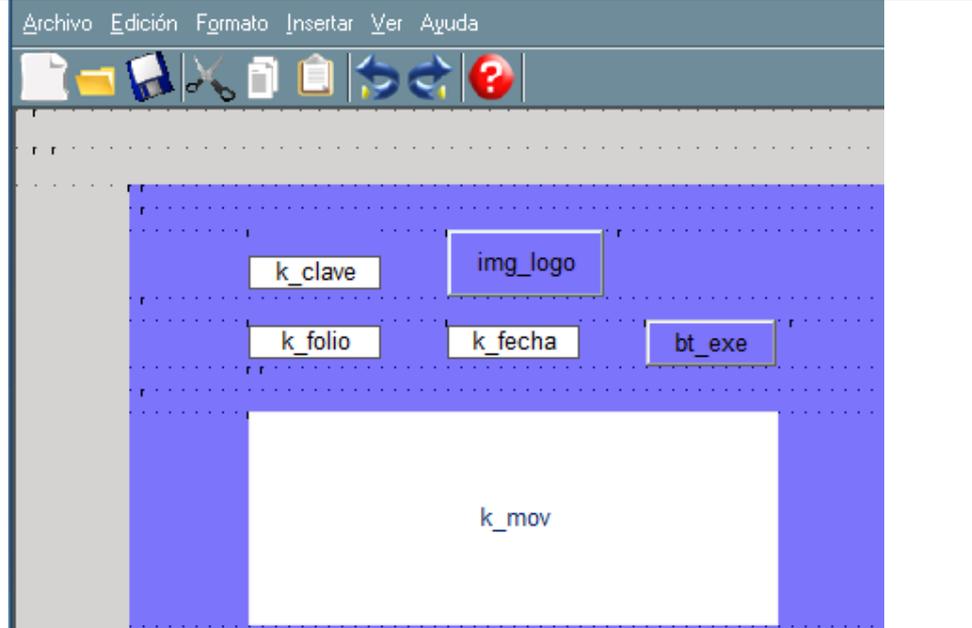
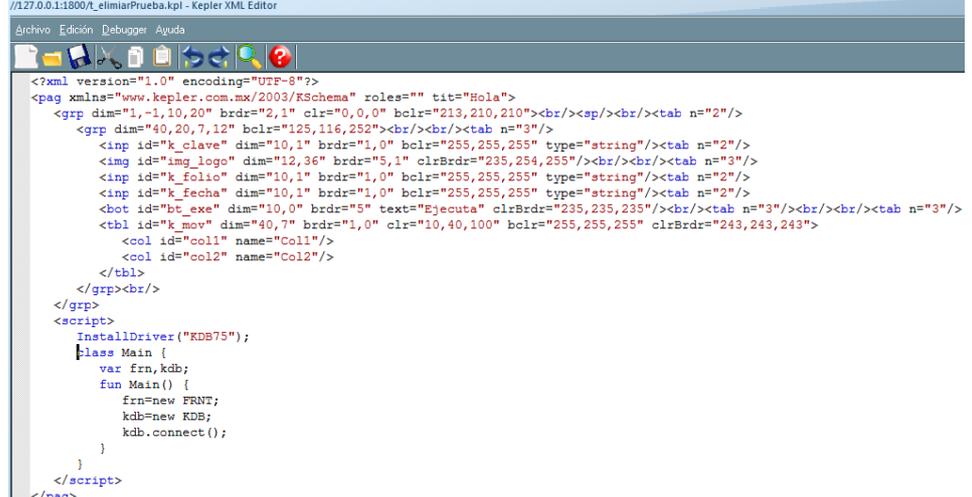
Contenido:

Creación de una nueva pantalla.....	4
ELEMENTOS GRÁFICOS KEPLER.....	9
Grupo	9
Input	15
Botón	22
Imagen	25
Check Box.....	27
Combo Box.....	32
Table	35
Radio Button	39
List Box.....	41
Horizontal Rule.....	44
Vertical Rule	45
Edit Box.....	46
Menús	52
Evaluación	58
Glosario.....	58

Kepler 8.0 Curso 4 “Elementos Gráficos”

Como parte de las nuevas ventajas que ofrece la versión 8.0 de Kepler se encuentra el nuevo modo gráfico de edición. Como se explicó en el Curso 3 de “**Uso del Sistema**”, Kepler ofrece tres modos de operación: **modo de ejecución**, **modo de edición gráfica**, **modo de edición XML** –como su nombre lo dice, estos dos últimos modos son para edición de opciones.

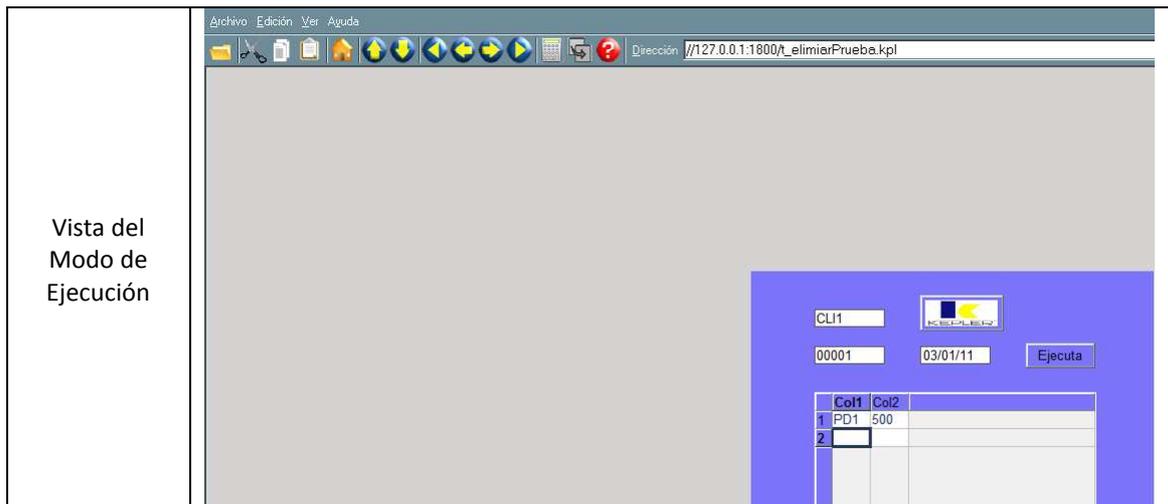
A continuación se muestran los dos modos de edición y el de ejecución para la misma pantalla, cabe resaltar que las tres pantallas son equivalentes:

<p>Vista del Modo de Edición Gráfica</p>	
<p>Vista del Modo de Edición XML</p>	 <pre> <?xml version="1.0" encoding="UTF-8"?> <pag xmlns="www.kepler.com.mx/2003/KSchema" roles="" tit="Hola"> <grp dim="1,-1,10,20" brdr="2,1" clr="0,0,0" bclr="213,210,210">
<sp/>
<tab n="2"/> <grp dim="40,20,7,12" bclr="125,116,252">

<tab n="3"/> <inp id="k_clave" dim="10,1" brdr="1,0" bclr="255,255,255" type="string"/><tab n="2"/>

<tab n="3"/> <inp id="k_folio" dim="10,1" brdr="1,0" bclr="255,255,255" type="string"/><tab n="2"/> <inp id="k_fecha" dim="10,1" brdr="1,0" bclr="255,255,255" type="string"/><tab n="2"/> <bot id="bt_exe" dim="10,0" brdr="5" text="Ejecuta" clrBrd="235,235,235"/>

<tab n="3"/> <tbl id="k_mov" dim="40,7" brdr="1,0" clr="10,40,100" bclr="255,255,255" clrBrd="243,243,243"> <col id="col1" name="Col1"/> <col id="col2" name="Col2"/> </tbl> </grp>
 </grp> </pag> <script> InstallDriver("KDB75"); class Main { var frn,kdb; fun Main() { frn=new FRNT; kdb=new KDB; kdb.connect(); } } </script> </pag> </pre>



Mediante estas dos formas de edición, se construye más rápida y fácilmente el estilo de las pantallas. Es decir que aunque las opciones aun no tengan funcionalidad, en poco tiempo se podrán diseñar sus interfaces gráficas. Lo anterior, gracias a que Kepler incorpora en su modo gráfico un editor tipo WYSIWYG. Acrónimo de *What You See Is What You Get* (en inglés, "lo que ves es lo que obtienes"), describe al estilo de editores que permiten escribir un documento mostrando al usuario directamente el resultado final, usualmente el resultado impreso. Un ejemplo son los procesadores de palabra como Microsoft Word y OpenOffice, donde lo que se va escribiendo se va desplegando en la pantalla prácticamente igual como se verá al estar impreso. Los tipos de letra (font), los colores y las proporciones se respetan en esta clase de aplicaciones.

En el caso de Kepler, la edición gráfica no sólo permite una visualización final de la pantalla sino que también facilita la manipulación de los objetos gráficos -ya sea para moverlos de lugar o modificar sus características o propiedades- presentes en ésta. Esto último es una ventaja permanente en la edición gráfica (CTRL^F), ya que al posicionarse sobre el elemento gráfico que se desee editar y posteriormente al dar clic en el menú y seleccionar la opción *Formato >> Objeto, grupo, página, renglones*, según sea el caso, se abrirá una pantalla especializada para el objeto seleccionado.

Creación de una nueva pantalla

Antes de comenzar a trabajar con cada uno de los elementos, se explica como crear una nueva pantalla.

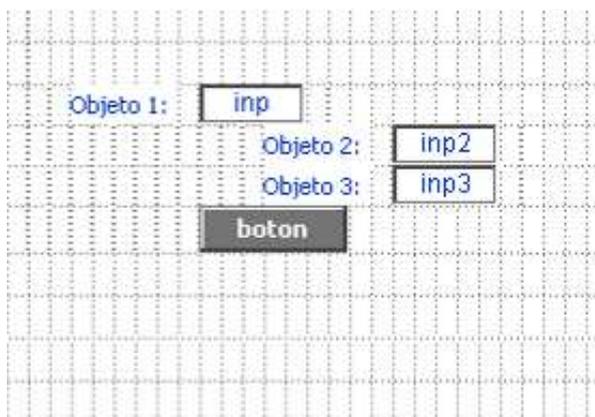
1. Partir de cualquier opción ya existente. Debemos ir al árbol de opciones y elegir alguna opción, ir al modo gráfico (CTRL^F).
2. A través del menú Archivo> Nuevo, se tienen las siguientes opciones:



- **Página Simple:** Se crea una página en blanco.
 - **Catálogo:** Genera de manera automática una página de catálogo en base a su definición (Tabla.def).
 - **Reporte:** Genera una plantilla de una página para generar un reporte.
 - **Formato:** Genera una plantilla de una página con un formato predefinido de reporte.
 - **Ayuda:** Genera una plantilla de una página de ayuda.
3. Después crear un nuevo documento, editarlo y salvarlo con “**Guardar Como**”. Es importante **NO** guardar la nueva pantalla con la opción de “**Guardar**”, es obligatorio guardar con “**Guardar Como**”, de no ser así no se guardará la nueva pantalla.
 4. Finalmente para ver la pantalla ya actualizada –y cargada en memoria– deberemos ir al modo de Ejecución (CTRL^E).

Respecto al directorio donde queda guardada la página: ésta se guardará con el nombre que se especifique; por ejemplo, si la página se llama miprueba.kpl se guardará por default en el servidor; es decir, en la **carpeta de trabajo (XSD)**. En caso de que el nombre venga precedido por un “_” (es decir: “_miprueba.kpl”) entonces se guardará en la **carpeta local (K80)**.

En la edición gráfica se presenta una cuadrícula que facilita la alineación de los objetos. Esta cuadrícula puede activarse/desactivarse a través del menú **Ver> Mostrar/Ocultar cuadrícula** en la edición gráfica. Es posible utilizar la tecla <TAB> para recorrer 3 cuadros y la combinación de teclas <SHIFT><SPACEBAR> para recorrer 1 cuadro. Para la alineación de los objetos, es conveniente utilizar la cuadrícula que simplemente utilizar la barra espaciadora, ya que se obtendrá un resultado más preciso.



Cada renglón, se compone de una secuencia de elementos, **el tamaño del renglón será del tamaño del mayor objeto contenido dentro de ese renglón**. Sin embargo no hay que olvidar que un posible elemento del renglón puede ser un "grupo". A continuación se presenta un ejemplo de la importancia de este tipo de acomodo.



Figura 1. El renglón será del tamaño de la tabla ya que es el elemento más grande en el renglón.

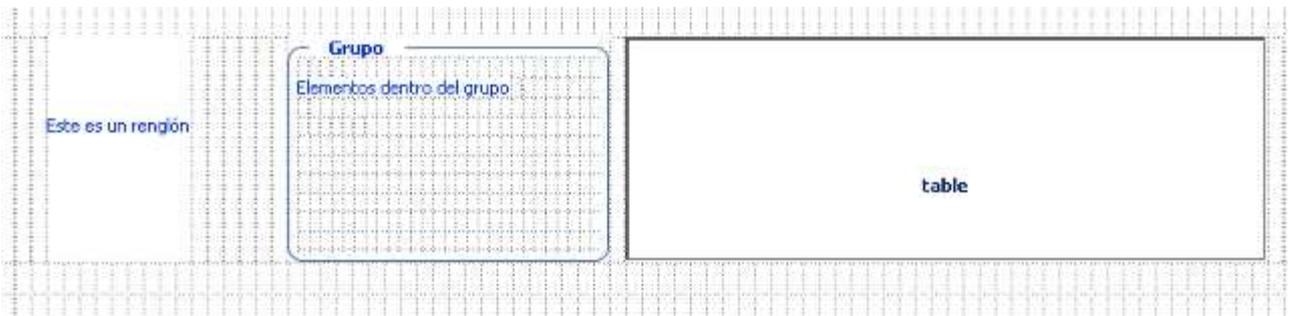
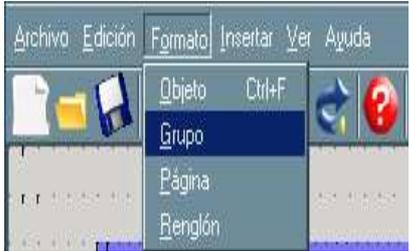
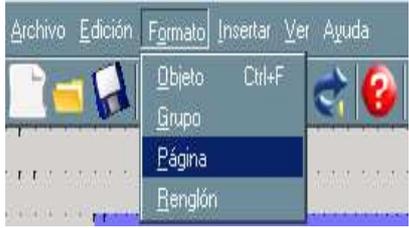
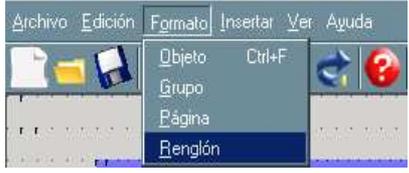


Figura 2. Dentro del grupo existen varios renglones de texto y no solo 1.

Para realizar la edición de elementos gráficos (ya sean creación de nuevas pantallas o de cualquiera de los demás elementos gráficos que se tratarán más adelante en el documento) será obligatorio trabajar desde el modo gráfico (CTRL^F) para así tener acceso a los menús de formato o bien trabajar directamente con los atributos en código en el modo de edición XML.

<p>Edición de Elementos gráficos</p>		<p>Una vez posicionado (seleccionado) sobre algún objeto gráfico en particular se abrirá una ventana especial para la modificación de las características principales de dicho elemento. Por ejemplo, si se seleccionara la imagen, se vería una pantalla como esta:</p>
---	---	---

		 <p>Se puede acceder a esta misma ventana, dando doble clic sobre el elemento.</p> <p>Para especificar otros atributos, así como la funcionalidad sobre los eventos, es posible definirlos a través de la edición XML. Para realizarlo se debe dar clic derecho sobre el elemento.</p>
<p>Edición de grupo</p>		<p>Una vez posicionado (seleccionado) sobre el grupo se abrirá una <i>ventana especial</i> para la modificación de características de <i>dicho grupo</i>.</p> <p>Por ejemplo, para un grupo se vería una pantalla como esta:</p> 
<p>Edición de página</p>		<p>Simplemente con estar en el modo gráfico y abriendo el menú aparecerá una pantalla como esta:</p> 
<p>Edición de Renglón</p>		<p>Una vez seleccionado el renglón se abrirá una ventana que permite especificar el (los) renglón(es) a un grupo de renglones de impresión. Esto se tratará más a detalle en otro curso. Por ejemplo, si se seleccionara un grupo de renglones aparecería una ventana como la</p>

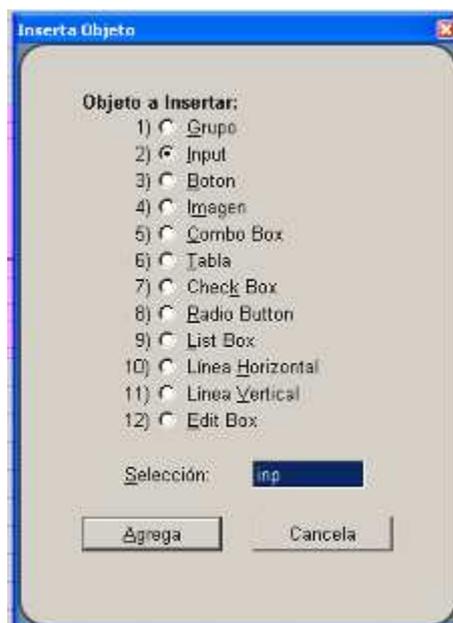


Ahora, pasemos al detalle de los elementos gráficos y al formato de presentación de dichos elementos en el presente documento. Se irá explicando a detalle cada uno de los elementos, su función y las características o atributos que lo componen. Para explicar sus atributos, se mostrará la descripción textual del atributo que se quiere modificar (i.e. font Arial negrita tamaño 12), su código XML (i.e. `fnt="Arial,12b">`), y su apariencia final en modo de ejecución:

Texto de Prueba

Dado que a través del menú de **Formato>Objeto, Grupo, Página o Renglón**, se abre una ventana especial para que se modifique el objeto, a partir de las características a definir o editar, en estas ventanas se irá haciendo la explicación de cada una de ellas; es decir, del valor que pueden tomar los elementos gráficos de un objeto. En algunos casos, se enriquecerá la explicación de los atributos de los elementos, mostrando ejemplos adicionales.

Respecto a los elementos gráficos que ofrece Kepler se encuentran los siguientes:



Para ver esta pantalla, lo podremos hacer desde el modo gráfico de cualquier opción. En el menú en **Insertar > Objeto**.

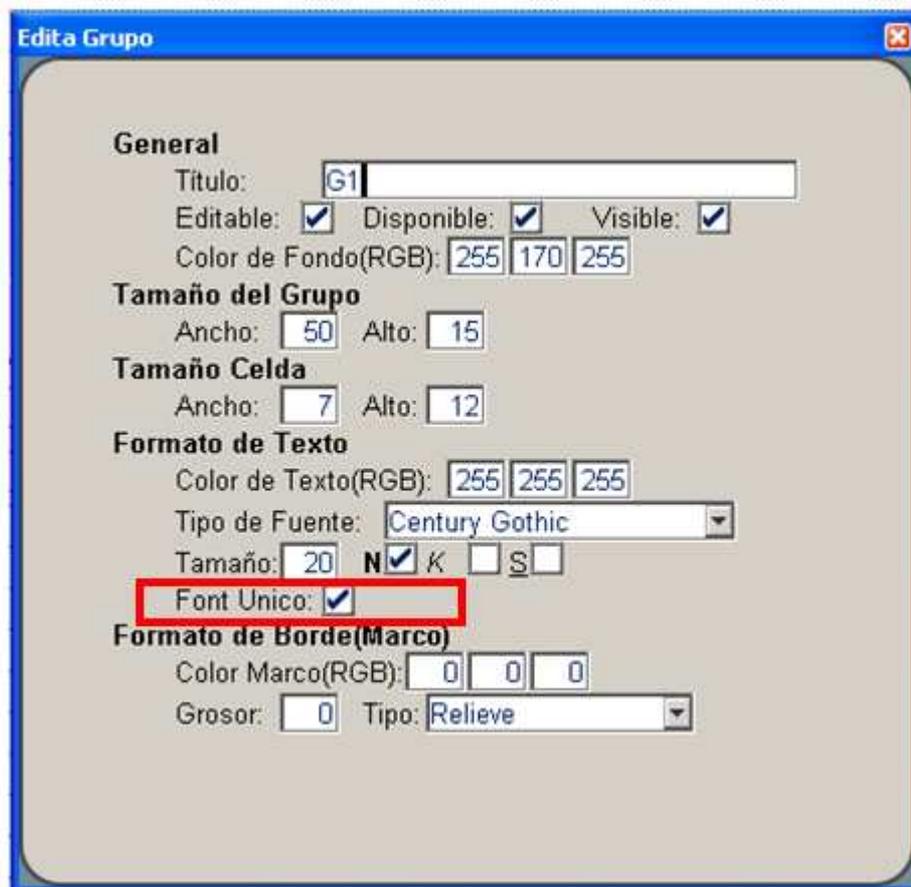
ELEMENTOS GRÁFICOS KEPLER

Grupo

Este elemento agrupa lógicamente a los objetos que estén contenidos en él. Define un área con ancho y alto particular donde se pueden contener cero o más elementos gráficos. Lo anterior permite estandarizar los atributos de dichos objetos.

Además de que en el grupo estén contenidos los objetos, al definir los atributos del grupo (desde la edición gráfica, en el menú ir a **Formato>Grupo**), éstos serán comunes para los elementos ahí contenidos. Por ejemplo si se desea que el tipo de letra, font, sea el mismo para todos los inputs, botones, radio buttons y demás elementos contenidos en el grupo se deberá palomear el **atributo Font Único**. (Ver imagen abajo).

Al final de la explicación de todos los atributos del grupo se muestra el resultado del grupo generado, con cada uno de los atributos especificados:



Edita Grupo

General
Titulo:
Editable: Disponible: Visible:
Color de Fondo(RGB):

Tamaño del Grupo
Ancho: Alto:

Tamaño Celda
Ancho: Alto:

Formato de Texto
Color de Texto(RGB):
Tipo de Fuente:
Tamaño: N K S
Font Único:

Formato de Borde(Marco)
Color Marco(RGB):
Grosor: Tipo:

Características y Atributos XML:

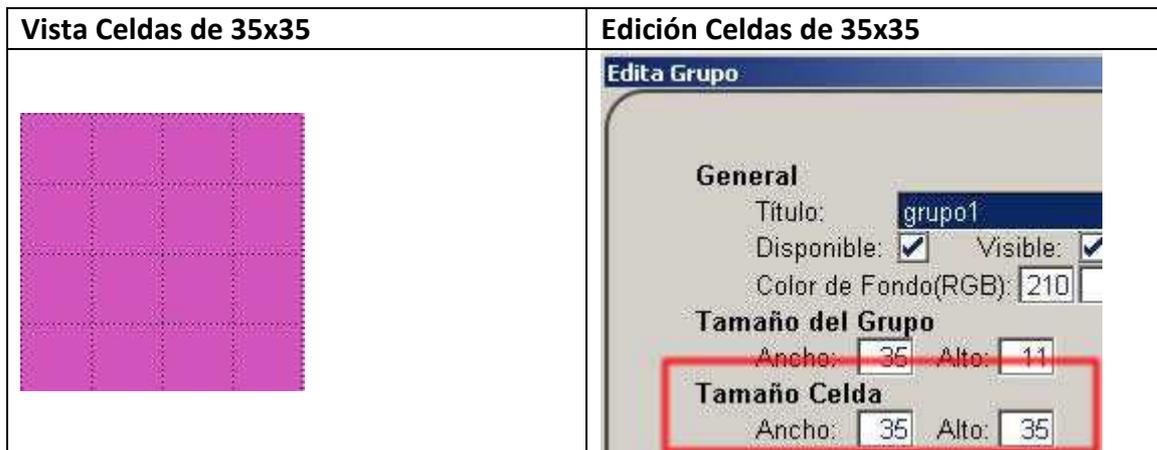
General

- **Título:** Es el título de ese grupo. Su equivalente en XML es:
tit: Más adelante se irá viendo que este título se visualizará en pantalla según el tipo de borde que seleccione para el grupo. De otro modo no se podrá ver en pantalla. Por ejemplo si deseáramos que el título fuera "G1", sería:
`<grp tit="G1">`
- **Editable:** Al palomear el checkbox de editable, el grupo y los elementos contenidos dentro de el mismo, podrán ser editables, de lo contrario no podrán modificarse. La equivalencia del atributo en XML es:
`<grp modify="N">`
- **Disponible:** Al palomear el checkbox de disponible, el grupo estará disponible para posicionarse sobre él. De otro modo, no se podrá acceder al grupo. La equivalencia del atributo en XML es:
available: Indica si es posible posicionarse en el grupo. Por ejemplo si quisiéramos que el campo no estuviera disponible para posicionar, sería:
`<grp available="N">`
- **Visible:** Al palomear el checkbox de visible, se podrá visualizar el grupo en pantalla. De otro modo, no se podrá ver el grupo en pantalla. La equivalencia del atributo en XML es:
visible: Por ejemplo si quisiéramos que el grupo fuera visible:
`<grp visible="Y">`
- **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo del objeto (en este caso el grupo). Su equivalencia XML es a través del atributo:
bclr: Indica el color de fondo en RGB*. Por ejemplo: para definir un grupo cuyo color de fondo sea azul claro se generaría el siguiente código XML.
`<grp bclr="0,255,255">`

Tamaño

- **Tamaño del Grupo:** Permite definir el tamaño del grupo. Se le pasa el ancho y el alto del grupo.
- **Tamaño de Celda:** Aquí se establece el ancho y alto de las celdas (en pixeles). Las celdas son los campos internos que componen al grupo. A continuación un ejemplo donde se ve la diferencia del tamaño en la celdas.

Vista Celdas de 15x15	Edición Celdas de 15x15
-----------------------	-------------------------



La equivalencia del atributo tamaño en XML es:

dim: Indica las dimensiones en pixeles del grupo en el siguiente orden: *ancho, alto, ancho_celda, alto_celda*.

Por ejemplo: para crear un grupo con un ancho de 100, alto de 45, y celdas de 10x10 se generaría el siguiente código XML.

```
<grp dim="100,45,10,10">
```

Formato de Texto

- Color del Texto (RGB):** A través de esta característica se establece el color del texto del grupo. Su equivalencia XML es a través del atributo: **clr:** El color default del texto se indica RGB*. Por ejemplo: para definir un grupo cuyo texto que tenga color negro se generaría el siguiente código XML.

```
<grp clr="0,0,0">
```
- Fuente:** Mediante esta característica se determina la tipografía default de los textos que aparezcan en el grupo. En el glosario se ahonda como definir las características de texto** del grupo. Su equivalencia en XML es:

fnt: Por ejemplo, para definir un grupo con letra Lucida Sans tamaño 22, con negrita, cursiva y subrayado sería:

```
<grp fnt="Lucida Sans,22bis">
```

- **Fuente Único:** Característica que permite definir el mismo tipo de letra para todos los objetos contenidos dentro del grupo. Su equivalencia en XML es:
Font único: Indica si todos los elementos contenidos en el grupo manejan el mismo formato de letra. Si toma el valor de 1 significa que se ignoran los formatos de cada elemento definido individualmente, y sólo se considera el formato del grupo. Si es 0, cada elemento puede definir su propio formato o tomar el formato default si no se establece alguno. A continuación la comparación de usar y no usar el **Unfont**.

Uso del Unfont



No se hace Uso del Unfont



Entonces para tener un grupo con un tipo de fuente diferente, el código XML sería:
<grp unfont="0">

Formato de Borde (Marco)

- **Color Marco (RGB):** Indica el color del borde del objeto (en este caso el grupo). Su equivalencia XML es a través del atributo:
clrBrdr: El color del borde (marco) se indica en formato RGB* del grupo.
Por ejemplo: para definir un grupo con un marco color negro sería:
<grp clrBrdr ="0,0,0">
- **Tipo de Borde (Grosor y Tipo):** Indica el tipo de borde del objeto (en este caso el grupo). Existen nueve tipos de borde para el tipo de objeto (grupo). Con respecto a la clasificación del tipo de borde, a continuación una descripción:

-1	Marco Default donde no se muestra borde en el grupo, sólo se deja el
----	--

	rectángulo con una línea delgada con el mismo color del grupo.
0	Efecto "Relieve" (Dibuja una línea delgada del color del marco, y deja un margen del color de fondo con el grueso indicado).
1	Efecto "Grabado" (Parecido al tipo 0, pero la línea izquierda y superior son un poco más gruesas)
2	Marco simple del color de borde (marco)
3	Efecto "3D"
4	Marco simple del color de borde, mostrando el título del grupo en la parte superior central. El formato del título es el mismo que el del texto del grupo. Resalta con negrita la primera línea del grupo.
5	Marco simple del color de borde, mostrando el título del grupo en la parte superior izquierda.
6	Igual al tipo 4, pero con esquinas redondeadas.
7	Igual al tipo 5, pero con esquinas redondeadas.

La equivalencia XML del **Tipo de Borde** es a través del atributo:

brdr: Indica el grueso del borde (marco) y el tipo de borde en el siguiente orden: grueso, tipo_borde. Por ejemplo, si quisiera un grupo con marco *tipo 1* con *grueso 10* sería:
<grp brdr ="10,1">

Finalmente, el resultado de establecer los atributos mostrados, se muestran a continuación, donde se define el tipo de borde, grosor y su visualización:

<p>Default Tipo=-1 Grosor=3</p>	
<p>Relieve Tipo=0 Grosor=3</p>	

<p>Grabado Tipo=1 Grosor=3</p>	 <p>DESCRIPCION: Pintura 1 Folio: 4502-368 Precio: 4,560.00</p>
<p>Simple Tipo=2 Grosor=3</p>	 <p>DESCRIPCION: Pintura 1 Folio: 4502-368 Precio: 4,560.00</p>
<p>3D Tipo=3 Grosor=3</p>	 <p>DESCRIPCION: Pintura 1 Folio: 4502-368 Precio: 4,560.00</p>
<p>Titulo1 Tipo=4 Grosor=3 (El título fue asignado a través del atributo tit previamente explicado)</p>	<p>Grupo 1</p>  <p>DESCRIPCION: Pintura 1 Folio: 4502-368 Precio: 4,560.00</p>
<p>Titulo2 Tipo=5 Grosor=3 (El título fue asignado a través del</p>	

<p>atributo tit previamente explicado)</p>	
<p>Título 1 Redondeado Tipo=6 Grosor=3 (El título fue asignado a través del atributo tit previamente explicado)</p>	
<p>Título 2 Redondeado Tipo=7 Grosor=3 (El título fue asignado a través del atributo tit previamente explicado)</p>	

* En todos los casos el grosor del marco de borde fue 3. También se aplicó la característica de Fuente Único (**Unfont**).

Input

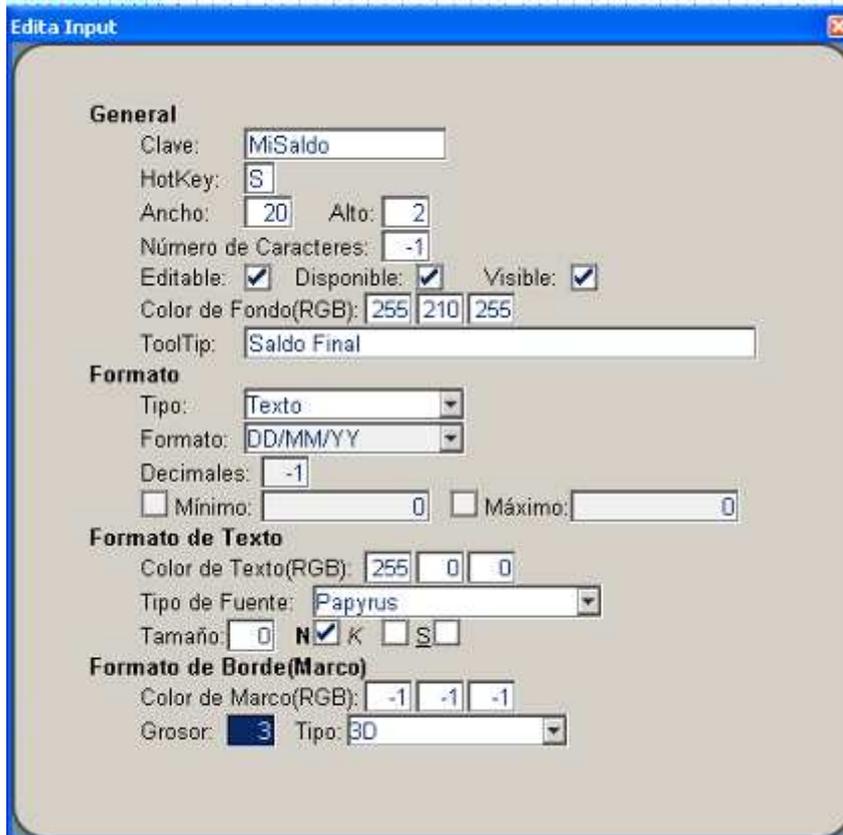
Es una caja de texto donde se podrán ingresar datos.

Este es una caja de texto:

A continuación se muestra un ejemplo de un input, más adelante y después de haber descrito sus atributos se explica qué se tuvo que modificar para que el input quedara así.



Para modificar las características de la caja de texto, primero hay que posicionarse en el input, y posteriormente ir a **Formato>Objeto**. Aparecerá una pantalla con las siguientes características:



```
<inp id="MiSaldo" dim="20,2" brdr="3,2" bclr="255,210,255" type="string"
fnt="Papyrus,0b" clr="255,0,0" hk="S" tooltip="Saldo Final"/><br/>
```

Características y Atributos XML:

General

- **Clave:** Esta característica permite identificar al input cuando se vaya a referenciar posteriormente. Su equivalente en XML es:

id: Si mas adelante se quiere hacer referencia a ese campo se haría con el nombre en id.

```
<inp id="MiSaldo">
```

- **Hot Key (Acceso Directo):** Esta característica permite posicionar el cursor sobre el objeto en cuestión (input), después de haberle asignado una letra en particular. Su equivalencia XML es:

hk: Indica si el input será (más fácilmente) accesible a través de un HotKey. En caso de que el HotKey fuera la letra U, entonces se tendría que presionar Alt+U para obtener un posicionamiento inmediato sobre ese input.

<inp hk="S">

- **Ancho y Alto:** Como lo indican sus nombres, estas dos características determinan el tamaño del input. Su equivalente en XML es:
dim: Indica las dimensiones calculadas en número de caracteres del input en el siguiente orden: ancho, alto. Para crear un input con un ancho de 25 y alto de 2.
<inp dim="20,2">
- **Número de Caracteres:** Restringe el número de caracteres en el input. Si el número de caracteres es mayor al tamaño de la pantalla (i.e. que el ancho esté calculado para 25 caracteres, pero el número de caracteres es 100 entonces se activará el scroll automáticamente).

Número de Caracteres=10	Número de Caracteres=100 (se activa el scroll [barra de desplazamiento] automáticamente)
<p>General</p> <p>Clave: <input type="text" value="MiSaldo"/></p> <p>HotKey: <input type="text" value="U"/></p> <p>Ancho: <input type="text" value="25"/> Alto: <input type="text" value="2"/></p> <p>Número de Caracteres: <input type="text" value="10"/></p>	<p>General</p> <p>Clave: <input type="text" value="MiSaldo"/></p> <p>HotKey: <input type="text" value="U"/></p> <p>Ancho: <input type="text" value="25"/> Alto: <input type="text" value="2"/></p> <p>Número de Caracteres: <input type="text" value="100"/></p>
	

Su equivalencia en XML es:

nc: Por ejemplo: para definir un número máximo de caracteres sería:

<inp nc="3">

Nota: cuando el **nc** es mayor que el ancho del input, entonces automáticamente se hará el scrolling en la caja de texto.

- **Editable:** Esta característica define si se podrá editar el contenido del input. Al palomear esta característica el campo en cuestión será editable. La equivalencia en XML es:
modify: Indica si el input es editable; es decir, si podrá ser modificado. En el siguiente ejemplo se podría editar el contenido del input.
<inp modify="Y">
- **Disponible:** Al palomear el checkbox de disponible, el campo estará disponible para posicionarse sobre él. De otro modo, no se podrá dar clic en ese campo.
available: Indica si es posible posicionarse en el input. Por ejemplo si quisiéramos que el campo no estuviera disponible para posicionar, sería:
<inp available="N">
- **Visible:** Al palomear el checkbox de visible, se podrá visualizar el campo en campo en pantalla. De otro modo, no se podrá dar clic en ese campo.

visible: Indica si el input será visible o no. Por ejemplo si quisiéramos que el campo fuera visible:

```
<inp visible="Y">
```

- **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo del objeto (en este caso el input). Su equivalencia XML es a través del atributo:

bclr: Se indica el color de fondo en notación RGB*. Por ejemplo: para definir un input cuyo color de fondo sea rosa pálido se generaría el siguiente código XML.

```
<inp bclr="255,175,250">
```

- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir un input cuyo tooltip sea: "Nombre del cliente".

```
<inp tooltip=" Nombre del cliente">
```

Formato

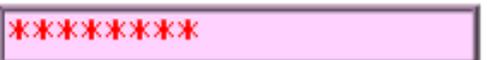
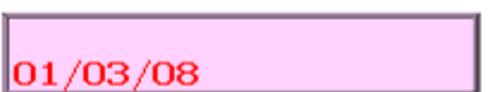
- **Tipo:** Mediante el tipo se especifica que tipo de máscara soporta el input. La utilidad de las máscaras se aprecia cuando se ingresa algún dato en el input, se captura y se espera que este dato sea de algún tipo en específico (es decir, se desea capturar números, texto, password, fecha, etc..). La equivalencia de esta característica en XML es:

type: A continuación se incluye una tabla donde se muestran los posibles valores que aceptan los inputs.

Por ejemplo, si el input recibiera texto sería así:

```
<inp type="string">
```

Tipos de Máscara que soporta un Input

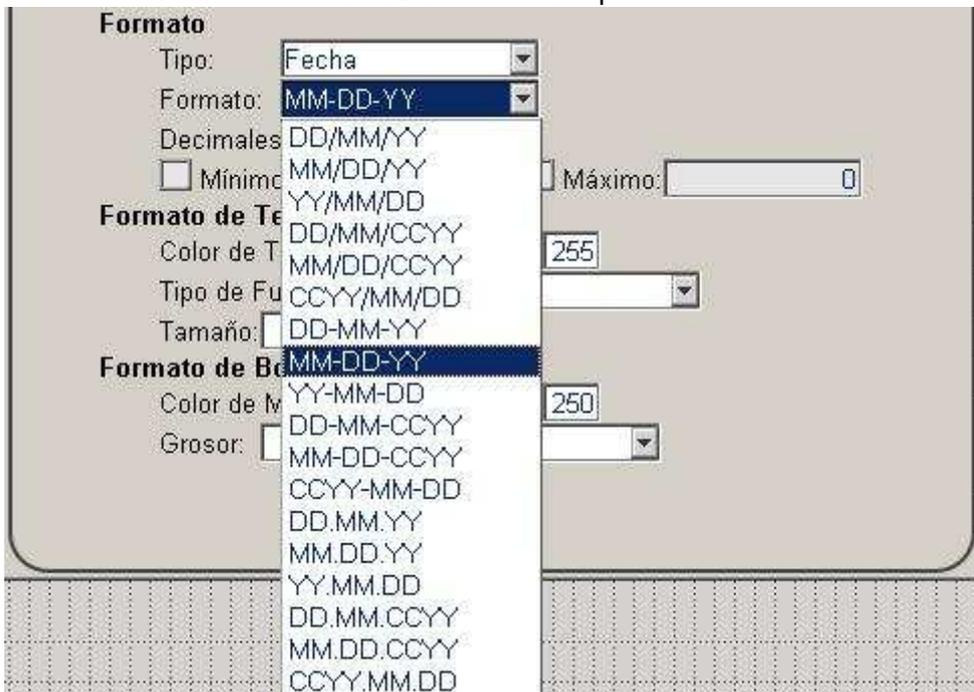
Texto (String)	
Mayúsculas	
Password	
Número	
Fecha (formato default: MM-DD-YY)* Otros: DD-MM-YY,DD-MM-CCYY)	
Hora (formato Horas:Minutos)	

<p>Folio (Completa el string introducido a un número fijo de caracteres, rellenando con ceros)</p>	<div style="border: 1px solid black; background-color: #f0f0f0; padding: 5px; text-align: center;"> 8000000001 </div>
---	---

* Los formatos de presentación de la fecha son 6:

- 1) día, mes, año (con 2 dígitos)
- 2) mes, día, año (con 2 dígitos)
- 3) año (con 2 dígitos), mes, día
- 4) día, mes, año (con 4 dígitos)
- 5) mes, día, año (con 4 dígitos)
- 6) año (con 4 dígitos), mes, día

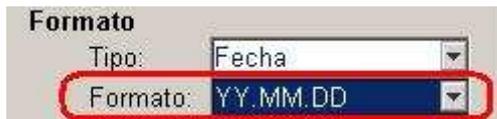
Estas 6 presentaciones se presentan con 3 tipos de separadores: '/', '-', '.'. A continuación se muestran las 18 variantes disponibles:



Importante:

→ Una vez seleccionado alguno de estos formatos, por ejemplo: MM/DD/YY, si se trata de escribir "23" en las posiciones asignadas para mes, el sistema no permitirá completar cualquier número una vez escrito el "2", ya que los meses no podrán pasar del último mes válido, es decir 12. Esta validación aplica para los días, donde el último posible del mes es "31". Para los años es válida cualquier combinación de máximo 4 dígitos.

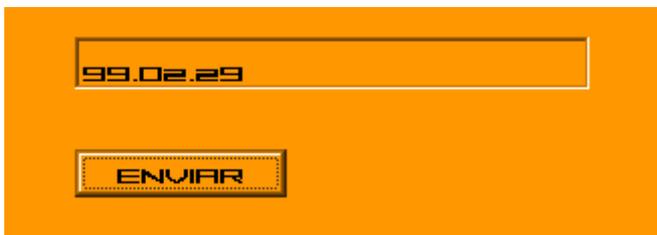
→ Otro punto es la corrección automática al formato establecido en el input. Es decir que si el input tiene un tipo de fecha establecido, pero el usuario no ingresa la fecha con formato correcto bastará con cambiar de campo para que se actualice inmediatamente al formato previamente establecido. Por ejemplo, suponga que se crea un input con formato YY.MM.DD.



Sin embargo, el usuario ingresa la fecha con el siguiente formato.



Aún con ello, al dar enter, el formato se respeta y se corrige:



Igualmente, cuando el campo es hora, dado el formato especificado *horas:minutos*, el máximo para estas posiciones, será 23 (horas) y 59 (minutos).



*Nota: Los siguientes tres atributos aplican cuando el **Tipo** es un número.*

- **Número de Decimales:** A través de este atributo se define el número de decimales que podrá tener un número en ese input. Su equivalencia en el código XML es:
nd: Automáticamente, en el input aparecerán ceros con el punto de decimal y *n* ceros indicando los decimales.
<inp nd="5">
- **Mínimo:** A través de esta característica se define el número mínimo que acepta ese input. Esto quiere decir que si en este atributo se especificara un “1”, entonces el input no aceptará números negativos. Su equivalencia en el código XML es:
min: Indica la cifra mínima que se guardará en el input. Para comprender mejor esta característica, observe el ejemplo donde el atributo de mínimo se le asigna un valor de 1.
<inp min="1">
Al establecer el valor de 1 para el atributo **min**, no se podrá poner ni 0 ni números negativos.
- **Máximo:** Esta característica nos permite definir el número máximo que aceptará el input. Su equivalencia en el modo XML es:

max: Este sería el ejemplo donde el atributo de máximo se le asigna un valor de 1.

```
<inp max="999">
```

Al establecer el valor de 999 para el atributo **max**, no se aceptará el número 1000 o superiores.

Por otro lado, es importante mencionar que aún cuando se establezca un rango –empleando un mínimo y un máximo específico- o una condición de cierto número de decimales para el contenido del input y ésta no se cumpla (en modo de ejecución) el sistema no marca error, en caso de salirse del valor especificado, simplemente se muestra el máximo o mínimo valor según sea el caso.

Formato de Texto

- **Color del Texto (RGB):** A través de esta característica se establece el color del texto del input. Su equivalencia XML es a través del atributo:

clr: Indica el color default en RGB*. Por ejemplo, para que el color del texto fuera azul agua sería:

```
<inp clr="0,255,255">
```

- **Fuente:** Mediante esta característica se determina la tipografía default de los textos que aparezcan en el input. En el glosario se ahonda como definir las características de texto**. Su equivalencia en XML es:

fnt: Por ejemplo, para definir un input con letra Impact sería:

```
<inp fnt="Impact">
```

Formato de Borde (Marco)

- **Color Marco (RGB):** Indica el color del borde del objeto (en este caso el input). Su equivalencia XML es a través del atributo:

clrBrdr: Indica el color del borde (marco) en RGB*. Por ejemplo: para definir un input con un marco color rosa fuerte sería:

```
<inp clrBrdr="255,150,250">
```

- **Tipo de Borde (Grosor y Tipo):** Indica el tipo de borde del objeto (en este caso el input). Existen tres tipos de borde para el input. Con respecto a la clasificación del tipo de borde, a continuación una descripción:

Tipos de Borde de Input

Simple*	
Grabado*	



* El grosor empleado con todos los tipos de borde fue 2.

brdr: Indica el grosor del borde (marco) y el tipo de borde en el siguiente formato: *grosor, tipo_borde*.

Por ejemplo, si quisiera un input con marco *tipo 2* con *grosor 3* sería:

```
<inp brdr ="3,2">
```

Botón

Es un objeto utilizado para indicar que al presionarlo se detonará algún evento. Para lograr su objetivo, se le asocia alguna tarea, de forma tal que cuando es oprimido se llame o ejecute alguna acción.

Como en los casos anteriores, editamos los atributos del botón en el menú en la opción **Formato>Objeto**.



El resultado de dichos cambios generaría un botón similar al siguiente.



Pasemos al detalle de los atributos del botón.

```
<bot id="bot1" dim="15,0" brdr="5" text="Enviar " hk="K" default="1" step="1" fnt="Chiller,32b" />
```

Características y Atributos XML:

General

- **Clave:** Esta característica permite identificar al botón para ser llamado posteriormente. Su equivalente en XML es:
id: Indica el nombre bajo el que se identificara dicho botón.
<bot id="Enviar">
- **Texto a Desplegar:** Esta característica muestra el texto del botón. Su equivalente en XML es:
text: Si se deseara que el botón desplegara el texto de "Enviar" el código sería:
<bot text="Enviar">
- **Acceso Directo:** Esta característica permite posicionar el cursor sobre el objeto en cuestión (botón), después de haberle asignado una letra en particular. Su equivalencia XML es:
hk: Por ejemplo si el (hk) HotKey fuera la letra O, entonces se tendría que presionar Alt+O para obtener un posicionamiento inmediato sobre ese botón.
<bot hk="O">
- **Tamaño:** Permite definir el ancho del botón. Su equivalente en XML es
dim: Indica las dimensiones (ancho) en pixeles del botón. Por ejemplo, para crear un botón con un ancho de 15 el código es:
<bot dim="15">
- **Default:** Al palomear el checkbox de default, indica si el cursor se posicionará en el botón, teniendo prioridad sobre los demás objetos en la pantalla. Esta característica es equivalente a la siguiente etiqueta XML.
default: En este ejemplo, el botón será el primero en tener el enter posicionado sobre los demás objetos.
<bot default="1">
- **Step:** Indica si el botón se verá presionado o sin presionar. Si queremos que al apretarlo conserve el aspecto de estar oprimido entonces el valor del atributo step deberá ser 1. En caso contrario deberá ser 0. En XML esta característica se conoce como:
step: En este ejemplo, el botón se quedará oprimido para simular que continua realizándose alguna acción.
<bot step="1">
- **Disponible:** Al palomear el checkbox de disponible, el botón estará disponible para posicionarse sobre él. De otro modo, no se podrá dar clic en el botón. Esta característica es equivalente a la siguiente etiqueta XML.

available: Indica si es posible posicionarse en el botón. Por ejemplo si quisiéramos que el botón no estuviera disponible (o pareciera desactivado) para posicionar, sería:
<bot available="N">

- **Visible:** Al palomear el checkbox de visible, se podrá visualizar el botón en pantalla. En XML esta característica se conoce como:

visible: Indica si el botón será visible o no. Por ejemplo si quisiéramos que el botón fuera invisible:
<bot visible="N">

- **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo del objeto (en este caso el botón). Su equivalencia XML es a través del atributo:

bclr: Indica el color de fondo en RGB* del input.

Por ejemplo: para definir un botón cuyo color de fondo sea azul rey se generaría el siguiente código XML.

<bot bclr="20,65,190">

- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir un botón cuyo tooltip sea: “Ejecuta la acción”.

<bot tooltip=" Ejecuta la acción">

Formato del Texto

- **Color del Texto (RGB):** A través de esta característica se establece el color del texto del botón. Su equivalencia XML es a través del atributo:

clr: Indica el color default del texto en RGB* dentro del botón. Por ejemplo, para que el color del texto fuera azul agua sería:

<bot clr="0,255,255">

- **Fuente:** Mediante esta característica se determina la tipografía default del texto que aparezca en el botón (i.e., el título del botón). En el glosario se ahonda como definir las características de texto**. Su equivalencia en XML es:

fnt: Por ejemplo, para definir un botón con letra Chiller tamaño 32, con negrita sería:

<bot fnt="Chiller,32b" >

Formato de Borde (Marco)

- **Color Marco (RGB):** Indica el color del borde del objeto (en este caso el botón). Su equivalencia XML es a través del atributo:

clrBrdr: Indica el color del borde (marco) en RGB* del botón. Por ejemplo: para definir un botón con un marco color gris fuerte sería:

<bot clrBrdr="80,80,80">

- **Grosor (Borde):** Indica el grosor del borde del botón. Entre más grande sea este número irá aumentando el grosor del borde del botón: Su equivalencia XML es:
brdr: Por ejemplo, si quisiera un botón con marco con *grosor 3* sería:
<bot brdr="3">

Imagen

Dentro de las nuevas capacidades de Kepler se encuentra la posibilidad de incluir imágenes dentro de la pantalla. Dichas imágenes se les deberá de especificar la ruta, si no se escribe explícitamente alguna ruta entonces se tomarán de la **carpeta IMG** ubicada **dentro de la carpeta de trabajo (XSD)**.

Cabe mencionar que una imagen también tiene la capacidad de **comportarse como un botón**. Por ejemplo, se puede presionar y ver "hundido" cuando esto ha sucedido, también se puede establecer el color y tipo de borde, entre otras características.



Como en los casos anteriores, podemos acceder a la de edición del objeto mediante el menú **Formato>Objeto**.



Y con respecto al código en XML que se genera, éste es:

```
<img id="img1" dim="20,75" brdr="2,1" img="colores.jpg" clrBrdr="0,255,0"  
tooltip="imagen" available="Y" hk="I"/>
```

Características y Atributos XML:

General

- **Clave:** Esta característica permite identificar la imagen para ser usada posteriormente. Su equivalente en XML es:

id: Es decir que si mas adelante se quiere hacer referencia a este objeto se haría con el nombre en id.

```
<img id="img1">
```

- **Acceso Directo:** Esta característica permite posicionar el cursor sobre la imagen después de haberle asignado una letra en particular. Su equivalencia XML es:

hk: Indica si la imagen será (más fácilmente) accesible a través de un HotKey. Por ejemplo, si el HotKey fuera la letra B, entonces se tendría que presionar Alt+B para obtener un posicionamiento inmediato sobre la imagen.

```
<img hk="I">
```

- **Ancho y Alto:** Permite definir el ancho y alto de la imagen. Su equivalente en XML es:

dim: Indica las dimensiones en pixeles de la imagen en el siguiente orden: *ancho, alto*. Para crear una imagen con un ancho de 10 y alto de 55 sería:

```
<img dim="10,55">
```

- **Disponible:** Esta característica define si se podrá modificar (presionar) la imagen. La equivalencia en XML es:

Modify: Indica si la imagen es editable; es decir, si podrá ser modificada (Presionada)

```
<img available="Y">
```

- **Visible:** Al palomear el checkbox de visible, se podrá visualizar la imagen en pantalla. En XML esta característica se conoce como:

visible: Por ejemplo si quisiéramos que la imagen fuera visible:

```
<img visible="Y">
```

- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir una imagen cuyo tooltip sea: "Presionar...".

```
<img tooltip=" Presionar...">
```

Formato de Borde (Marco)

- **Color Marco (RGB):** Indica el color del borde de la imagen. Su equivalencia

XML es a través del atributo:

clrBrdr: Indica el color del borde (marco) en RGB* de la imagen. Por ejemplo: para definir una imagen con un marco color verde como se muestra en el ejemplo sería:

```
<img clrBrdr="0,255,0">
```

- **Archivo:** Aquí es donde se especifica la ruta donde se encuentre la imagen a mostrar en pantalla. A continuación se incluye su equivalencia XML y las extensiones de imágenes que acepta el sistema.

img: Los formatos de imagen que soporta Kepler son los siguientes diez:

- jpeg/jif,
- gif,
- png,
- bmp,
- ico,
- mng,
- pcx,
- psd (photoshop),
- tga (targa) y
- tiff

El código que genera se presenta a continuación:

```
<img img="colores.jpg">
```

Como se comentó anteriormente, por default la ruta donde se deberán guardar las imágenes es en la carpeta del sistema (XSD/imágenes), para que el sistema las encuentre.

Check Box

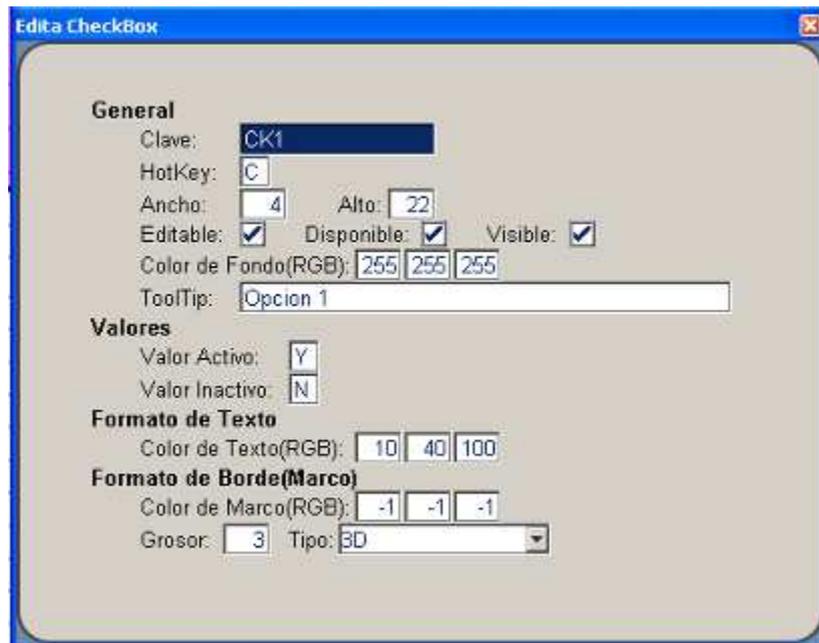
Es un recuadro que se puede seleccionar y palomear. A continuación un ejemplo de cómo se ve este recuadro cuando se le solicita al usuario que lo seleccione para posteriormente realizar una acción en base a dicha información.



Especifique el [los] filtro[s] del reporte

<input checked="" type="checkbox"/>	Clave
<input checked="" type="checkbox"/>	Grupo
<input type="checkbox"/>	Tipo

Podemos acceder a la de edición del objeto mediante el menú **Formato>Objeto**.



Finalmente el código generado en el modo XML, producto de insertar un objeto de tipo CheckBox fue:

```
<chck id="CK1" dim="4,22" brdr="3,2" clr="10,40,100" bclr="255,255,255" YN="10"
hk="C" tooltip="Opcion 1"/>
```

Características y Atributos XML:

General

- **Clave:** Esta característica permite identificar al checkbox para ser llamado posteriormente. Su equivalente en XML es:
id: Indica el nombre bajo el que se identificara dicho checkbox. Es decir que si mas adelante se quiere hacer referencia a ese objeto se haría con el nombre en id.

```
<chck id=" CK1">
```

- **Acceso Directo (HotKey):** Esta característica permite posicionar el cursor sobre el objeto en cuestión (checkbox), después de haberle asignado una letra en particular. Esto permitirá al checkbox ser (más fácilmente) accesible a través de un HotKey. Su equivalencia XML es:
hk: Por ejemplo, en caso de que el HotKey fuera la letra "c", entonces se tendría que presionar Alt+c para obtener un posicionamiento inmediato sobre ese checkbox.

```
<chck hk="c">
```

- **Ancho y Alto:** Estas características permiten definir el alto y ancho del checkbox. Su equivalente en XML es:

dim: Indica las dimensiones en pixeles del checkbox en el siguiente orden: *ancho, alto*. Por ejemplo un checkbox de ancho 4 y alto 22.

```
<chck dim="4,22">
```

▪ **Editable:** Esta característica define si se podrá editar el contenido del checkbox. La equivalencia en XML es:

modify: Indica si el checkbox es editable; es decir, si podrá ser modificado.

```
<chck modify="Y">
```

▪ **Disponible:** Al palomear el checkbox de disponible, el checkbox estará disponible para posicionarse sobre él. De otro modo, no se podrá dar clic en el objeto. Esta característica es equivalente a la siguiente etiqueta XML.

available: Indica si es posible posicionarse en el checkbox. Por ejemplo si quisiéramos que el checkbox no estuviera disponible (o pareciera desactivado) para posicionar, sería:

```
<chck available="Y">
```

▪ **Visible:** Al palomear el checkbox de visible, se podrá visualizar el checkbox en pantalla. En XML esta característica se conoce como:

visible: Por ejemplo si quisiéramos que el checkbox fuera visible:

```
<chck visible="Y">
```

▪ **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo del checkbox. Su equivalencia XML es a través del atributo:

bclr: Indica el color de fondo en RGB* del objeto. Por ejemplo: para definir un checkbox cuyo color de fondo sea azul rey se generaría el siguiente código XML.

```
<chck bclr="20,65,190">
```

▪ **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir un checkbox cuyo tooltip sea: "Ejecuta la acción".

```
<chck tooltip=" Ejecuta la acción">
```

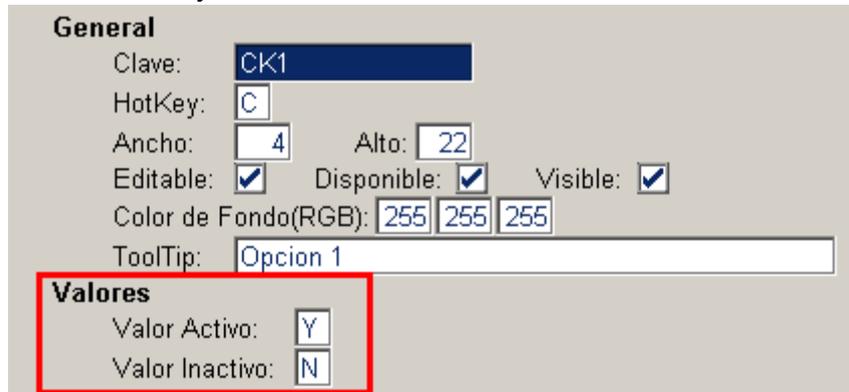
Valores

▪ **Valor Activo y Valor Inactivo:** Por default cuando se palomea un checkbox se activa un "1" y si se deja desactivado queda en "0". Sin embargo, es posible que cuando programemos en XML resulte de mayor utilidad guardar algún otro carácter, más representativo. Por ello, este atributo permite que se le asigne un valor particular cuando el checkbox ha sido seleccionado (**Yes**) y cuando no (**No**). De tal forma que se puedan asociar letras arbitrarias que sean más útiles; una letra para el valor Activo y otra para el valor inactivo. La equivalencia de esta característica en XML se llama:

YN: Por ejemplo, para asignar una **A** (de Autorizado) a **Yes** y una **N** (de No Autorizado) para No sería:

```
<chck YN="AN">
```

Lo anterior se puede establecer a través del atributo en la etiqueta **YN** del XML, o bien a través de Formato>>Objeto.



Formato del Texto

- **Color del Texto (RGB):** A través de esta característica se establece el color del interior del checkbox. Su equivalencia XML es a través del atributo:

clr: Indica el color default del texto en RGB* dentro del checkbox.

Para que el color del texto fuera azul agua sería:

```
<chck clr="10,40,100">
```

Formato de Borde (Marco)

- **Color Marco (RGB):** Indica el color del borde del objeto (checkbox). Su equivalencia XML es a través del atributo:

clrBrdr: Indica el color del borde (marco) en RGB* del checkbox.

Por ejemplo: para definir un checkbox grupo con un marco color rosa fuerte sería:

```
<chck clrBrdr="0,5,5">
```

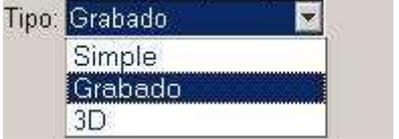
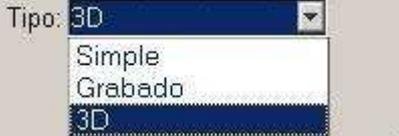
- **Grosor y Tipo (Borde):** Indica el grosor y el tipo del borde del checkbox.

Existen 3 tipos de borde para el checkbox:

- Simple
- Grabado y
- 3D

Clasificación de Tipos de Borde de CheckBox

Tipo de Borde (para todos los casos donde grueso=3)	Selección en el Menú (Formato>> Objeto)
 <p>Clave CheckBox Tipo: Simple</p>	

 <p>CheckBox Tipo: Grabado</p>	
 <p>CheckBox: 3D</p>	

Su equivalencia XML es:

brdr: Indica el grueso del borde (marco) y el tipo de borde en el siguiente orden: *grueso, tipo_borde*. Por ejemplo, si quisiera un checkbox con marco *tipo 2* con *grueso 3* sería:

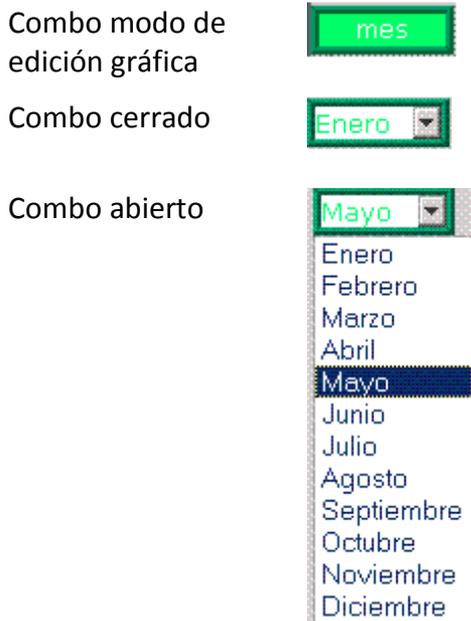
```
<chck brdr="3,2">
```

Finalmente, comentar que al final del código de XML para el Checkbox de Clave, se agregó la siguiente etiqueta (marcada en negritas) que muestra el texto en pantalla.
`<chck id="CK1" dim="4,22" brdr="3,2" clr="10, 40,100" bclr="255,255,255" hk="C" clr="255,0,0" clrBrdr="0,255,0" YN="SN" available="Y" visible="Y"/><txt> Clave</txt>`

Combo Box

Es una caja de texto que despliega distintos valores, para posteriormente ser seleccionados por el usuario. Los valores pueden ser distintos, ya sea texto, números, fechas, etc.

Un ejemplo de un combobox con los meses del año:



Podemos acceder a la de edición del objeto mediante el menú **Formato>Objeto**.



Bien, ahora en el modo de edición XML se generó el siguiente código:

```
<cmb id="mes" dim="7,1" brdr="2,1" bclr="0,255,110" fnt="Verdana,16"  
clrBrdr="0,255,175" modify="Y" available="Y" visible="Y"/>
```

Detallemos cada uno de estos atributos a continuación:

Características y Atributos XML:

General

- **Clave:** Es el nombre bajo el que se identifica el combobox.
id: Indica el nombre bajo el que se identificara dicho combo. Es decir que si mas adelante se quiere hacer referencia a ese campo se haría con el nombre en id.

```
<cmb id="mes">
```
- **Acceso directo:** Indica si el combo será (más fácilmente) accesible a través de un HotKey.
hk: Por ejemplo, en caso de que el HotKey fuera la letra R, entonces se tendría que presionar Alt+R para obtener un posicionamiento inmediato sobre ese combo.

```
<cmb hk="R">
```
- **Tamaño:** Permite definir el ancho del combobox. Su equivalente en XML es **dim:** Indica el ancho del combo en pixeles del combo. Para crear un combo con un ancho de 36 sería:

```
<cmb dim="36">
```
- **Acepta Vacío:** Esta característica permite que el combo pueda quedarse vacío, de otro modo siempre deberá contener algún valor. Cuando esta característica *Acepte Vacío*, al dar clic en la barra espaciadora se limpiará el combo y se podrá proceder sin tener que elegir alguna de las opciones disponibles. Esta característica en XML se expresa:
null: Por ejemplo si quisiéramos definir un combo que aceptara vacío el código sería:

```
<cmb null="Y">
```
- **Editable:** Esta característica define si se podrá editar el combo (cambiar la selección). La equivalencia en XML es:
modify: Indica si el combo es editable; es decir, si podrá ser modificado.

```
<cmb modify="Y">
```
- **Disponible:** Al palomear el checkbox de disponible, el combobox estará disponible para posicionarse sobre él. Esta característica es equivalente a la siguiente etiqueta XML.
available: Esto es importante porque de no permitir el posicionamiento entonces no se podrían mostrar los valores del combo. Por ejemplo si quisiéramos que el combo estuviera disponible (o pareciera activado) para posicionar, sería:

```
<cmb available="Y">
```

- **Visible:** Al palomear el checkbox de visible, se podrá visualizar el combo en pantalla. En XML esta característica se conoce como:
visible: Indica si el combo será visible o no. Por ejemplo si quisiéramos que el combo fuera visible:

`<cmb visible="Y">`

- **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo del combo. Su equivalencia XML es a través del atributo:
bclr: Indica el color de fondo en RGB* del combo.
Por ejemplo: para definir un combo cuyo color de fondo sea verde fosforescente se generaría el siguiente código XML.

`<cmb bclr="0,255,110">`

- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir un combo cuyo tooltip sea: "Selecciona algo".

`<cmb tooltip="Selecciona algo">`

Formato del Texto

clr: Indica el color default del texto en RGB* dentro del combo.

Para que el color del texto fuera blanco sería:

`<cmb clr="255,255,255">`

- **Fuente:** Mediante esta característica se determina la tipografía default de los textos que aparezcan en el combo. En el glosario se ahonda como definir las características de texto**. Su equivalencia en XML es:

fnt: Por ejemplo, para definir un combobox con letra Verdana tamaño 16 sería:

`<cmb fnt="Verdana,16">`

Formato del Borde

- **Color Marco (RGB):** Indica el color del borde del objeto (en este caso el combo). Su equivalencia XML es a través del atributo:

clrBrdr: Indica el color del borde (marco) en RGB* del combo. Por ejemplo: para definir un combo con un marco verde fuerte sería:

`<cmb clrBrdr="0,255,175">`

- **Tipo de Borde (Grosor y Tipo):** Indica el tipo de borde del combo. Existen tres tipos de borde para el input. Con respecto a la clasificación del tipo de borde, a continuación una descripción:

brdr: Indica el grosor del borde (marco) y el tipo de borde en el siguiente orden: *grosor, tipo_borde*. Por ejemplo, si quisiera un combo con marco *tipo 2* con *grosor 4* sería:

`<cmb brdr="4,2">`

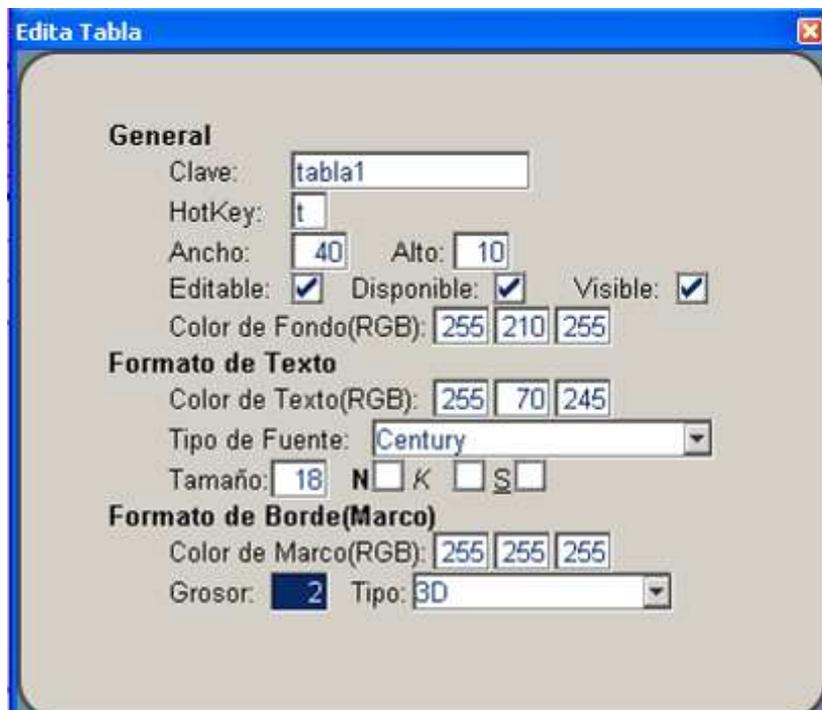
Table (Grid)

Similar a una tabla de una hoja de cálculo, las tablas en Kepler nos permiten ingresar diversos datos, definiendo un número específico de columnas con número de renglones infinito. A continuación se muestra un ejemplo de una tabla, la modificación de sus atributos –como se ha explicado en los objetos anteriores- y en particular la definición del número de columnas que obligatoriamente se realiza a través del modo XML.

A continuación el ejemplo de una tabla:

	COLUMNA1	COLUMNA2	
1	dato1	dato2	
2	dato 3	numero 1	
3	dato 4		
4	dato 5		
5	dato 6		
6	dato 7		
7	dato 8		
8	dato 9		

Para editar en el menú sería en **Formato>Objeto**, para hacerlo en el modo gráfico sería:



Edita Tabla

General
 Clave:
 HotKey:
 Ancho: Alto:
 Editable: Disponible: Visible:
 Color de Fondo(RGB):

Formato de Texto
 Color de Texto(RGB):
 Tipo de Fuente:
 Tamaño: N K S

Formato de Borde(Marco)
 Color de Marco(RGB):
 Grosor: Tipo:

Ahora, en el modo XML se genera el siguiente código:

```
<tbl id="tabla1" dim="40,10" brdr="2,2" clr="255,70,245" bclr="255,210,255" hk="t"
fnt="Century,18" clrBrdr="255,255,255"><col id="col1" name="Columna1"/><col
id="col2" name="Columna2"/></tbl>
```

Analicemos ahora los atributos:

Características y Atributos XML:

General

- **Clave:** Indica el nombre bajo el que se identificara dicho el radio button. Su equivalente en XML es:

id: Indica el nombre bajo el que se identificara la tabla. Es decir que si mas adelante se quiere hacer referencia a la tabla se haría con el nombre en id.

```
<tbl id="tabla">
```

- **Acceso Directo:** Esta característica permite posicionar el cursor sobre la tabla después de haberle asignado una letra en particular. Su equivalencia XML es:

hk: Indica si la tabla será (más fácilmente) accesible a través de un HotKey. En caso de que el HotKey fuera la letra t, entonces se tendría que presionar Alt+t para obtener un posicionamiento inmediato sobre la tabla.

```
<tbl hk="t">
```

- **Ancho y Alto:** Estas características permiten definir el alto y ancho de la tabla.

Su equivalente en XML es:

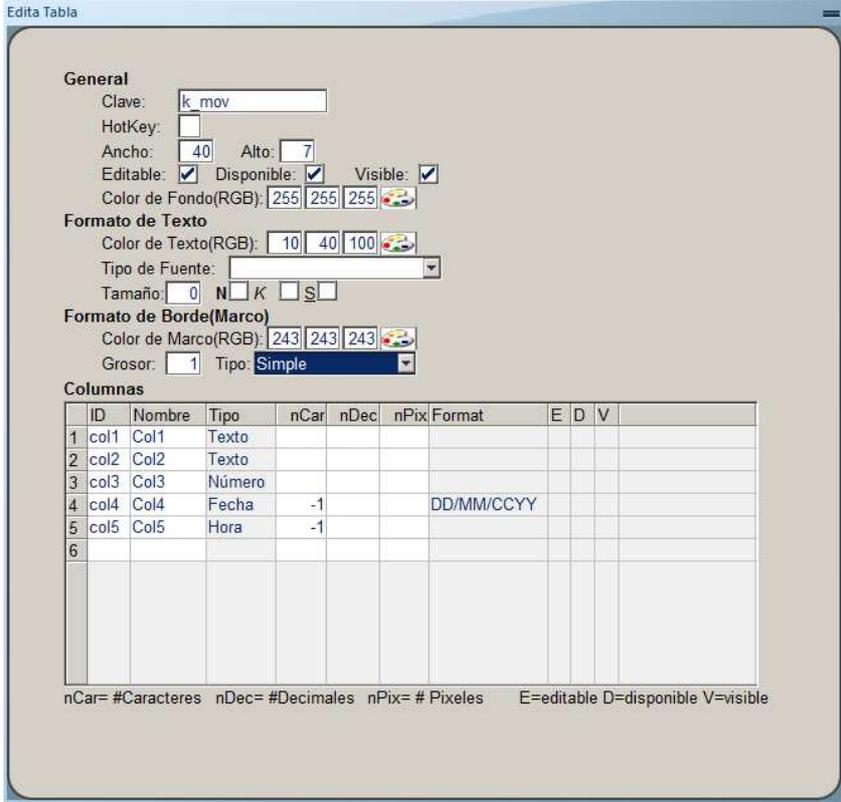
dim: Indica el ancho de la tabla en pixeles, donde el primer valor corresponde al ancho mientras que el segundo al alto. Así que, para crear una tabla con un ancho de 40 y un alto de 10 sería:

```
<tbl dim="40,10">
```

Columnas y Renqlones de una tabla

El agregar columnas a una tabla se puede realizar de dos formas:

1. Por **edición grafica**, aquí se agrega el **ID** (identificador) de la columna, su **nombre** y **características** para dicha columna.



General
Clave:
HotKey:
Ancho: Alto:
Editable: Disponible: Visible:
Color de Fondo(RGB):

Formato de Texto
Color de Texto(RGB):
Tipo de Fuente:
Tamaño: N K S

Formato de Borde(Marco)
Color de Marco(RGB):
Grosor: Tipo:

Columnas

ID	Nombre	Tipo	nCar	nDec	nPix	Format	E	D	V
1	col1	Col1							
2	col2	Col2							
3	col3	Col3							
4	col4	Col4		-1		DD/MM/CCYY			
5	col5	Col5		-1					
6									

nCar= #Caracteres nDec= #Decimales nPix= # Pixeles E=editable D=disponible V=visible

2. La segunda forma de agregar columnas es por medio de **edición XML**, permite hacer la definición del número de columnas que tendrá la tabla. Forzosamente deberemos usar el modo XML y agregar explícitamente tantos atributos como columnas se deseen. Por ejemplo en la tabla mostrada previamente, se declararon dos columnas y el código que se agregó fue:

```
<col id="col2" name="Col2"/><col id="col3" name="Col3"
type="number"/><col id="col4" name="Col4" type="date" nc="-1"
fmt="DD/MM/CCYY"/><col id="col5" name="Col5" type="time" nc="-1"/>
```

Como en todos los casos anteriores, la **columna** tiene el atributo **id**, bajo el cual **será identificada y podrá ser referenciada**. El atributo **name** será el texto que aparecerá como **nombre o título de la columna**. En caso de omitirse se pondrá la letra del alfabeto correspondiente a dicha columna (i.e., en la cuarta columna sin nombre, aparecería una letra "D" como título de la columna).

Finalmente respecto a los renglones, no es necesario definirlos ya que éstos serán infinitos para el número de columnas que haya sido previamente definido.

- **Editable:** Esta característica define si se podrá editar el contenido de la tabla. Al palomear esta característica el campo en cuestión será editable. La equivalencia en XML es:
modify: Indica si el input es editable; es decir, si podrá ser modificado. En el siguiente ejemplo se podría editar el contenido de la tabla.

```
<tbl modify="Y">
```

- **Disponible:** Al palomear el checkbox de disponible, la tabla estará disponible para posicionarse sobre ella. Su equivalencia en XML es:
available: Indica si es posible posicionarse en la tabla. Esto es importante porque de no permitir el posicionamiento entonces no se podrían escribir valores en la tabla. Por ejemplo si quisiéramos que la tabla no estuviera disponible, sería:

```
<tbl available="N">
```

- **Visible:** Al palomear el checkbox de visible, se podrá visualizar la tabla en pantalla. El código equivalente en XML es:
visible: Indica si la tabla será visible o no. Por ejemplo si quisiéramos que el combo fuera visible:

```
<tbl visible="N">
```

- **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo de la tabla. Su equivalencia XML es a través del atributo:
bclr: Por ejemplo: para definir una tabla cuyo color de fondo sea rosa fuerte se generaría el siguiente código XML.

```
<tbl bclr="230,0,110">
```

Formato de Texto

- **Color del Texto (RGB):** A través de esta característica se establece el color del texto de la tabla. Su equivalencia XML es a través del atributo:
clr: Indica el color default del texto en RGB* dentro de la tabla.
Para que el color del texto fuera fucsia el código XML generado sería:

```
<tbl clr="255,70,245">
```

- **Fuente:** Establece el tipo de letra que se usará en la tabla. En el glosario se profundiza en como definir las características de texto**. Su equivalencia en XML es:
font: Por ejemplo, si quisiera una tabla con tipo de letra Century, tamaño 18, entonces el valor del atributo sería:

```
<tbl font="Century,18">
```

Formato de Borde

- **Color Marco (RGB):** Indica el color del borde de la tabla en RGB*. Su equivalencia XML es a través del atributo:
clrBrdr: Por ejemplo: para definir una tabla con un marco blanco sería:

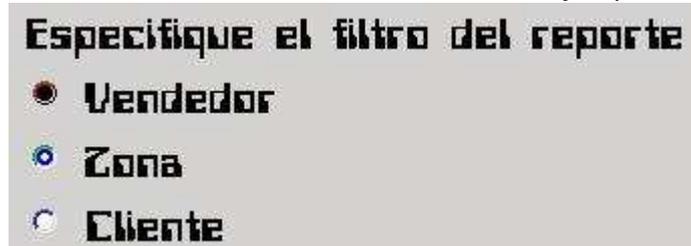
```
<tbl clrBrdr="255,255,255">
```

- **Tipo de Borde (Grosor y Tipo):** Indica el tipo de borde y grosor de la tabla.
brdr: Indica el grosor del borde (marco) y el tipo de borde en el siguiente orden: *grosor, tipo_borde*. Por ejemplo, si quisiera una tabla con marco *tipo 2* con *grosor 3* sería:

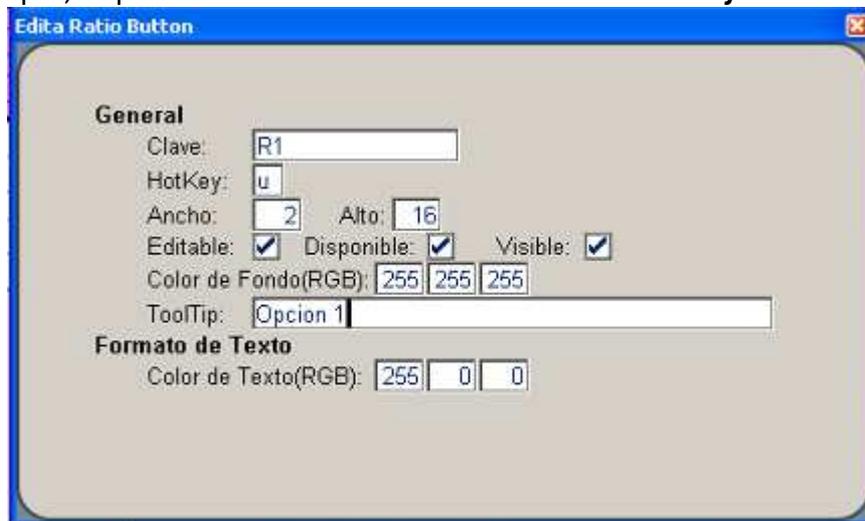
```
<tbl brdr="3,2">
```

Radio Button

Es un objeto circular que permite realizar la selección de una y sólo una opción. A diferencia del checkbox, por su naturaleza permite una selección única por cada grupo. Esto implica que si se tienen dos (o más grupos en la pantalla) se tendrían dos conjuntos diferentes de radio buttons. A continuación un ejemplo:



Como siempre, se permite la edición desde el menú **Formato>Objeto**.



Por otro lado, el código en XML para el primer radio button (acompañado con el input Vendedor) que se genera es:

```
<ratio id="R1" dim="2,16" clr="255,0,0" bclr="255,255,255" hk="u" tooltip="Opcion 1" modify="Y" available="Y" visible="Y"/>
```

Características y Atributos XML:

General

- **Clave:** Indica el nombre bajo el que se identificara dicho el radio button. Su equivalente en XML es:
id: Es decir que si más adelante se quiere hacer referencia a ese radio button se haría con el nombre en id.

```
<ratio id="R1">
```

- **Acceso Directo:** Esta característica permite posicionar el cursor sobre el objeto en cuestión (radio button), después de haberle asignado una letra en particular. Su equivalencia XML es:
hk: En caso de que el HotKey fuera la letra u, entonces se tendría que presionar Alt+u para obtener un posicionamiento inmediato sobre ese radio.

<ratio hk="u">

- **Ancho y Alto:** Indica las dimensiones en pixeles del radio, en el orden: ancho, alto. Permite definir el ancho del botón. Su equivalente en XML es:
dim: Indica las dimensiones en pixeles del radio en el siguiente orden: *ancho*, *alto*. Para crear un radio con un ancho de 2 y alto de 16.

<ratio dim="2,16">

- **Editable:** Esta característica define si se podrá modificar (seleccionar) el radio. La equivalencia en XML es:

modify: Indica si el radio button es editable; es decir, si podrá ser seleccionado.

<ratio modify="Y">

- **Disponible:** Al palomear el checkbox de disponible, el radio estará disponible para posicionarse sobre él. Esta característica es equivalente a la siguiente etiqueta XML.

available: Por ejemplo si se deseara que el radio no estuviera disponible (o pareciera desactivado) para posicionar, sería:

<ratio available="N">

- **Visible:** Al palomear el checkbox de visible, se podrá visualizar el radio en pantalla. En XML esta característica se conoce como:

visible: Indica si el radio será visible o no. Por ejemplo si quisiéramos que el radio fuera invisible:

<ratio visible="N">

- **Color de Fondo (RGB):** Indica el color interno (del fondo) en RGB* del radio button. Su equivalencia XML es a través del atributo:

bclr: En el ejemplo anterior, el color de fondo del primer radio (con input Vendedor) es negro, mientras que en el segundo y tercer radio (con inputs Zona y Cliente respectivamente) el color de fondo es blanco -aún cuando el segundo radio fue seleccionado. A continuación se pone el código generado en modo XML tanto del primer radio (negro) como del segundo (blanco) para apreciar la diferencia.

<ratio bclr="0,0,0"> (primero)

<ratio bclr="255,255,255"> (segundo)

- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir un radio button cuyo tooltip sea: "Selecciona una opción".

<ratio tooltip="Selecciona una opción">

Formato del Texto

- **Color del Texto (RGB):** Indica el color en RGB* del borde del radio. Su equivalencia XML es a través del atributo:

clr: Por ejemplo, para que el color del borde del radio fuera rojo el código sería:

```
<radio clr="255,0,0">
```

- **Default:** Indica si el radio button aparecerá "seleccionado" desde que se cargue dicha pantalla. Su equivalente XML es:

default: En A continuación un ejemplo este ejemplo, el radio será el primero en tener el enter posicionado sobre los demás objetos.

```
<radio default="1">
```

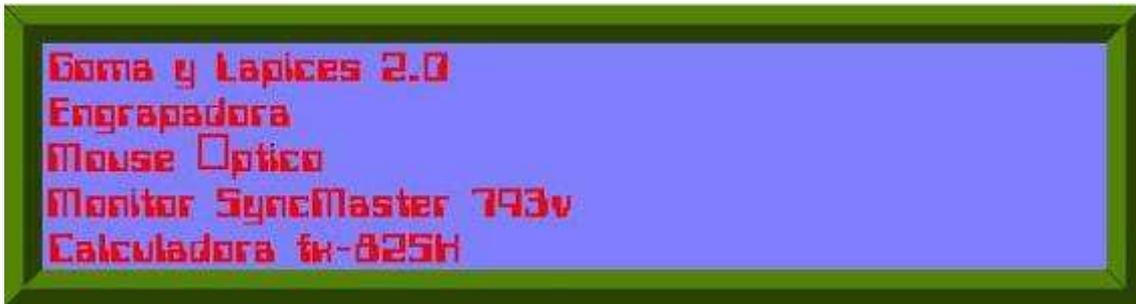
List Box

Es una caja de texto que despliega distintos valores, para posteriormente permitir la selección de uno y solo un elemento. Los valores pueden ser distintos, ya sea texto, números, fechas, etc. Tiene prácticamente la misma funcionalidad que un combo a diferencia que desde el inicio se puede definir el alto de éste (cuando en el input sólo se puede definir el ancho y no el alto). Dicha edición se puede realizar desde el menú **Formato>Objeto**.



En caso de que el número de elementos alojados en la lista exceda el alto definido, realizará automáticamente el scrolling correspondiente. A continuación un ejemplo de cómo se vería una lista con autoscrolling:

A continuación un ejemplo de una lista:



* En este curso no se mostrará como llenar la lista con diversos elementos.

Por otro lado, el código en XML para la lista que se genera es:
<list id="lista1" dim="60,5" brdr="20,2" clr="10,40,100" bclr="128,128,255" hk="L"
tooltip="Seleccione un elemento de la lista" fnt="Algerian,20bis"
clrBrdr="128,200,0"/>

Características y Atributos XML:

- **Clave:** Indica el nombre bajo el que se identificara dicha lista. Su equivalente en XML es:
id: Por ejemplo, a continuación se crea una lista cuya clave es "lista1".
`<list id="lista1">`
- **Acceso Directo:** Esta característica permite posicionar el cursor sobre el objeto en cuestión (lista), después de haberle asignado una letra en particular. Su equivalencia XML es:
hk: Indica si la lista será (más fácilmente) accesible a través de un HotKey. En caso de que el HotKey fuera la letra L, entonces se tendría que presionar Alt+L para obtener un posicionamiento inmediato sobre esa lista.
`<list hk="L">`
- **Ancho y Alto:** Indica las dimensiones en pixeles de la lista, en el orden: *ancho*, *alto*. Su equivalente en XML es:
dim: Indica las dimensiones en pixeles de la lista en el siguiente orden: ancho, alto. Para crear una lista con un ancho de 60 y alto 5.
`<list dim="60,5">`
- **Editable:** Esta característica define si se podrá modificar la selección de la lista. La equivalencia en XML es:
modify: Indica si la lista será editable; es decir, si podrá ser modificado su valor seleccionado.
`<list modify="Y">`
- **Disponible:** Al palomear el checkbox de disponible, indica si es posible posicionarse en la lista. Esta característica es equivalente a la siguiente etiqueta XML.
available: Por ejemplo si quisiéramos que el lista no estuviera disponible (o pareciera desactivada) para posicionar, sería:

<list available="N">

- **Visible:** Al palomear el checkbox de visible, se indica si la lista será visible o no. En XML esta característica se conoce como:

visible: Por ejemplo si quisiéramos que la lista estuviera visible:

<list visible="Y">

- **Color de Fondo (RGB):** Indica el color interno (del fondo) en RGB* de la lista.
bclr: Por ejemplo: para definir una lista cuyo color de fondo sea azul se generaría el siguiente código XML.

<list bclr="128,128,255" >

- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:

tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento .

Por ejemplo: para definir una lista cuyo tooltip sea: "Selecciona un opción".

<list tooltip="Selecciona un opción">

Formato de Texto

- **Color del Texto (RGB):** A través de esta característica se establece el color del texto en RGB* dentro de la lista. Su equivalencia XML es a través del atributo:

clr: Indica el color default del texto Para que el color del texto fuera rojo sería:

<list clr="255,0,0">

- **Fuente:** Establece el tipo de letra que se usará en la lista. En el glosario se profundiza en como definir las características de texto**. Su equivalencia en XML es:

fnt: Por ejemplo, si quisiera una lista con tipo de letra Algerian tamaño 20 entonces el valor del atributo sería:

<list fnt="Algerian,20">

- **Color Marco (RGB):** Indica el color del borde (marco) en RGB* de la lista. Su equivalencia XML es a través del atributo:

clrBrdr: Por ejemplo: para definir una lista con un marco color verde sería:

<list clrBrdr="128,200,0">

Tipo de Borde (Grosor y Tipo): Indica el tipo de borde de la lista. En XML, el atributo es:

brdr: Indica el grosor del borde (marco) y el tipo de borde en el siguiente orden: *grosor, tipo_borde*. Por ejemplo, si quisiera una lista con marco *tipo 2* con *grosor 20* sería:

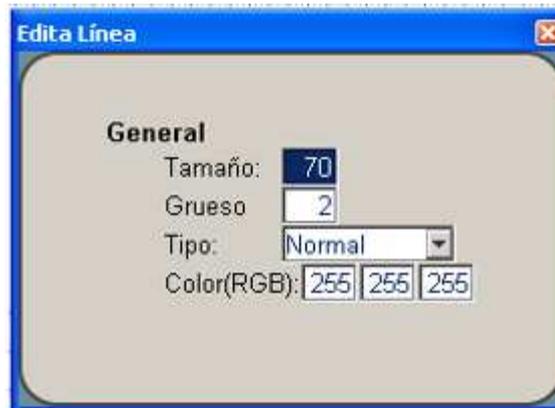
<list brdr="20,2">

Horizontal Rule

Es una regla horizontal estilo HTML. Sirve para dar formato dentro de una pantalla. A continuación un ejemplo de regla horizontal.



Como cualquiera de los demás objetos, se permite la edición desde el menú **Formato>Objeto**.



El código en XML sería:
<hLine dim="70,0" g="2" type="2" clr="255,255,255"/>

Características y Atributos XML:

- **Tamaño:** En este campo se pone el valor del ancho de la regla horizontal. Este estará dado en pixeles. Su atributo correspondiente en el modo XML es: **dim:** Indica las dimensión, ancho, en pixeles de la línea. Para crear una regla horizontal con un ancho de 70.

<hLine dim="70">

- **Tipo:** El valor del tipo presenta dos opciones: normal y hundido. A continuación un ejemplo de ambas líneas:

Línea Horizontal con tipo: normal



Línea Horizontal con tipo: hundido



Su atributo correspondiente en el modo XML es:

type: Indica el tipo de la regla horizontal.

Para que el tipo de la regla horizontal fuera "Hundido" el código sería:

<hLine type="2">

- **Grueso:** En este campo se pone el valor del ancho de la regla horizontal. Este ancho está dado en pixeles. Su atributo correspondiente en el modo XML es: **g:** Indica el grueso de la regla horizontal

A continuación el código de un grosor 2.

```
<hLine g="2">
```

NOTA: Para el caso del ejemplo de la regla horizontal donde el tipo es hundido, se le asigno un grosor de 2 y un tipo "hundido", cuando se seleccione el tipo el grosor es fijo.

- **Color (RGB):** Es a través de esta característica que se define el color de la línea. El color se define en notación RGB. Su atributo correspondiente en el modo XML es:

clr: Indica el color en RGB* del borde de la regla horizontal.

Para que el color del borde de la regla horizontal fuera blanco el código sería:

```
<hLine clr="255,255,255">
```

Vertical Rule

Es una regla vertical estilo HTML. Sirve más que nada para dar formato dentro de una pantalla. A continuación un ejemplo de regla horizontal.



Como cualquiera de los demás objetos, se permite la edición desde el menú *Formato>>Objeto*.



El código en XML sería:

```
<vLine g="3" clr="255,0,0"/>
```

Características y Atributos XML:

A diferencia de la regla horizontal (hline), a este objeto gráfico sólo se le puede definir el grosor pero no se puede definir su alto.

- **Color (RGB):** Es a través de esta característica que se define el color de la línea. El color se define en notación RGB. Su atributo correspondiente en el modo XML es:

clr: Indica el color en RGB* del borde de la regla vertical.

Para que el color del borde de la regla vertical fuera rojo el código sería:

```
<vLine clr="255,0,0">
```

- **Grueso:** En este campo se pone el valor del ancho de la regla vertical. Este ancho estará dado en pixeles. Su atributo correspondiente en el modo XML es:
g: Indica el grueso de la regla vertical.
A continuación el código de un grosor 3.

```
<vLine g="3">
```

Edit Box

Prácticamente igual al ComboBox, el EditBox muestra un conjunto de valores con la facilidad de escribir algún valor (que no se encuentre contenido en la lista que se despliegue originalmente) en tiempo de ejecución. La principal ventaja de esto es que ese valor "nuevo" ingresado puede ser capturado (aun cuando no se haya desplegado en la lista original).

Por ejemplo, suponga que se quiere conocer cual fue la forma de conocimiento de la Empresa Arizona Instruments y despliega las siguientes opciones en el EditBox:

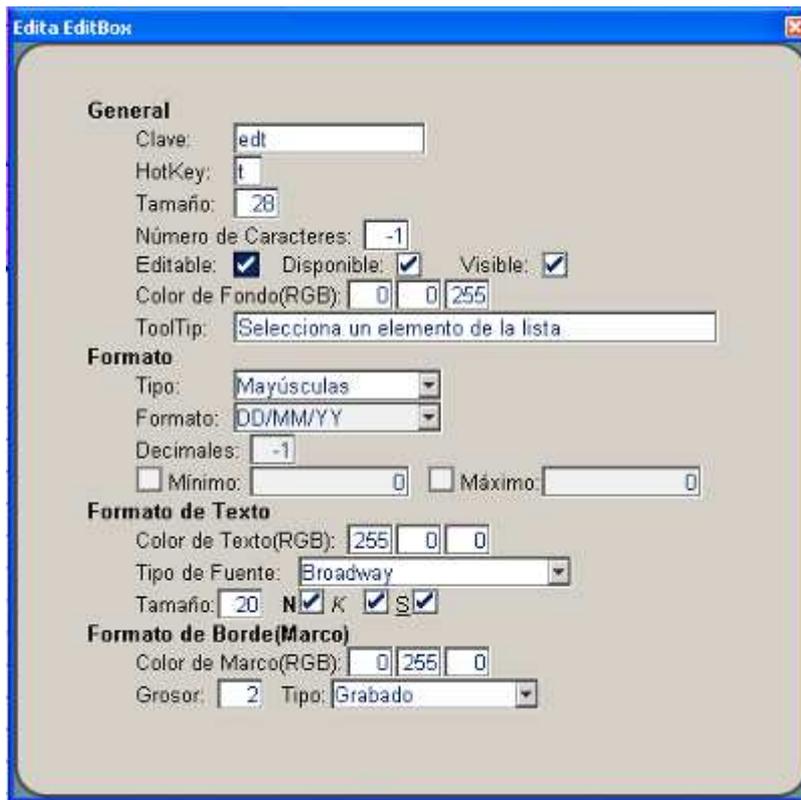


Sin embargo, ninguna de las opciones mostradas en el EditBox corresponde a la opción que el usuario desea. Entonces, el usuario podrá teclear alguna otra opción y escribirla en la caja...



Posteriormente, cuando se analice el valor en el edit (con id="edt") se podrá recuperar el valor ahí escrito.

Al realizar la edición de este elemento usando el menú de la edición gráfica, se muestra una pantalla muy similar a la siguiente:



```
<edt id="edt" dim="28,1" brdr="2,1" bclr="0,0,255" type="capital" hk="t" modify="Y"
tooltip="Selecciona un elemento de la lista" clr="255,0,0" fnt="Broadway,20bis"
clrBrdr="0,255,0" available="Y" visible="Y"/><br/><tab n="2"/>
```

Características y Atributos XML:

General

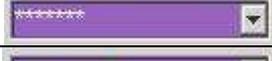
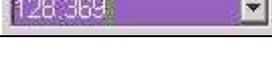
- **Clave:** Esta característica permite identificar al editbox para ser llamado posteriormente. Su equivalente en XML es:
id: Si el editbox se llamara "edt" el código sería:
`<edt id="edt">`
- **Acceso Directo:** Esta característica permite posicionar el cursor sobre el editbox, después de haberle asignado una letra en particular. Su equivalencia XML es:
hk: Por ejemplo si el (hk) HotKey fuera la letra t, entonces se tendría que presionar Alt+t para obtener un posicionamiento inmediato sobre editbox.
`<edt hk="t">`
- **Tamaño:** Permite definir el ancho del editbox. Su equivalente en XML es:
dim: Indica las dimensiones (ancho) en pixeles del objeto. Por ejemplo, para crear un editbox con un ancho de 28 la equivalencia XML sería:
`<edt dim="28">`

- **Editable:** Esta característica define si se podrá editar el contenido del editbox. Al palomear esta característica el campo en cuestión será editable. La equivalencia en XML es:
modify: En el siguiente ejemplo se podría editar el contenido del editbox.
`<edt modify="Y">`
- **Disponible:** Al palomear el checkbox de disponible, el editbox estará disponible para posicionarse sobre él. De otro modo, no se podrá dar clic en el campo del editbox. Esta característica es equivalente a la siguiente etiqueta XML.
available: Por ejemplo si quisiéramos que el editbox no estuviera disponible (o pareciera desactivado) para posicionar, sería:
`<edt available="N">`
- **Visible:** Al palomear el checkbox de visible, se podrá visualizar el editbox en pantalla. En XML esta característica se conoce como:
visible: Por ejemplo si quisiéramos que el editbox fuera invisible:
`<edt visible="N">`
- **Color de Fondo (RGB):** A través de esta característica se establece el color de fondo del editbox. Su equivalencia XML es a través del atributo:
bclr: Para definir un editbox cuyo color de fondo sea azul rey se generaría el siguiente código XML.
`<edt bclr="20,65,190">`
- **ToolTip:** A través de esta característica se establece un mensaje de ayuda emergente al elemento. Se activa al dejar posicionado el puntero del ratón sobre el elemento por un instante. Su equivalencia XML es a través del atributo:
tooltip: Indica el mensaje de tooltip (ayuda emergente) del elemento.
Por ejemplo: para definir un editbox cuyo tooltip sea: "Teclea la clave".
`<edt tooltip=" Teclea la clave">`

Formato

- **Tipo:** Mediante el tipo se especifica que tipo de máscara que se guardará en el editbox. Es importante enfatizar que al igual que en el caso del Input, la máscara se refiere al tipo que se aceptara al capturar un valor ingresado por el usuario. En XML:
type: A continuación una tabla donde se muestran los posibles valores que aceptan los inputs.
Por ejemplo, si el editbox recibiera mayúsculas sería así:
`<edt type="capital">`

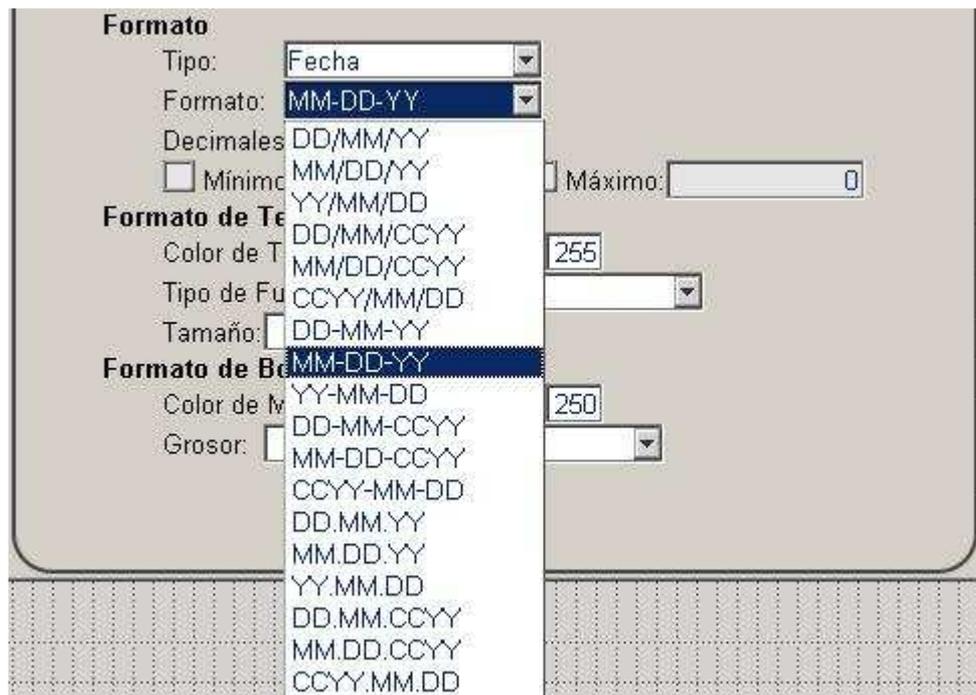
Tipos de Máscara que soporta un EditBox

Texto (String)	
Mayúsculas	
Password	
Número	
Fecha (formato default: MM-DD-YY)* Otros: DD-MM-YY,DD-MM-CCYY)	
Hora (formato Horas:Minutos)	
Folio (Completa el string introducido a un número fijo de caracteres, rellenando con ceros)	H00078

* Los formatos de presentación de la fecha son 6:

- 1) día, mes, año (con 2 dígitos)
- 2) mes, día, año (con 2 dígitos)
- 3) año (con 2 dígitos), mes, día
- 4) día, mes, año (con 4 dígitos)
- 5) mes, día, año (con 4 dígitos)
- 6) año (con 4 dígitos), mes, día

Estas 6 presentaciones se presentan con 3 tipos de separadores: '/', '-', '.'. A continuación se muestran las 18 variantes disponibles:



*Nota: Los siguientes tres atributos aplican cuando el **Tipo** es un número.*

- **Número de Decimales:** A través de este atributo se define el número de decimales que podrá tener un número en ese editbox. Su equivalencia en el código XML es:

nd: Automáticamente, en el input aparecerán ceros con el punto de decimal y *n* ceros indicando los decimales.

```
<edt nd="5">
```

- **Mínimo:** A través de esta característica se define el número mínimo que acepta ese editbox. Su equivalencia en el código XML es:

min: Indica la cifra mínima que se guardará en el editbox. Para comprender mejor esta característica, observe el ejemplo donde el atributo de mínimo se le asigna un valor de 1.

```
<edt min="1">
```

- **Máximo:** Esta característica nos permite definir el número máximo que aceptará el editbox. Su equivalencia en el modo XML es:

max: Este sería el ejemplo donde el atributo de máximo se le asigna un valor de 1.

```
<edt max="999">
```

Formato del Texto

- **Color del Texto (RGB):** A través de esta característica se establece el color del texto del editbox. Su equivalencia XML es a través del atributo:

clr: Por ejemplo, para que el color del texto fuera azul agua sería:

```
<edt clr="0,255,255">
```

- **Fuente:** Mediante esta característica se determina la tipografía default del texto que aparezca en el editbox. En el glosario se ahonda como definir las características de texto**. Su equivalencia en XML es:

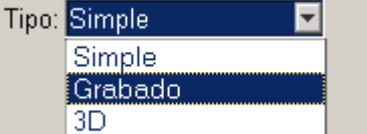
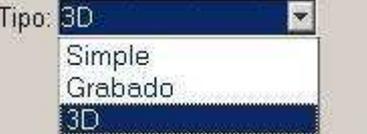
fnt: Por ejemplo, para definir un botón con letra Broadway tamaño 20, con negrita, itálica y subrayada sería:

`<bot fnt="Broadway,20bis" >`

Formato de Borde (Marco)

- **Color Marco (RGB):** Indica el color del borde, en RGB, del objeto (en este caso el editbox). Su equivalencia XML es a través del atributo: **clrBrdr**: Por ejemplo: para definir un botón con un marco color gris fuerte sería:
- **Tipo de Borde (Grosor y Tipo):** Indica el tipo de borde del editbox. Existen tres tipos de borde para el editbox:

Clasificación de Tipos de Borde de EditBox

Tipo de Borde (para todos los casos donde grueso=2)	Selección en el Menú (Formato>> Objeto)
 <p>EditBox Tipo: Simple</p>	
 <p>EditBox Tipo: Grabado</p>	
 <p>EditBox Tipo: 3D</p>	

Menús



Así como las páginas pueden modificar todos los objetos gráficos que la componen, los menús no son la excepción. Cada página puede definir sus propios menús, así como la funcionalidad que tendrá cada elemento del menú.

Es importante mencionar que no es necesario que cada página incluya todas las opciones de menús generales pues se tienen hasta 3 niveles de jerarquía. Es decir, en la página "**_ks_frame.kpl**" que define la distribución de la pantalla, es posible especificar ciertos menús que serán accesibles desde cualquier página. Luego, las páginas que contienen árbol de opciones como "**index.kpl**", pueden definir ciertos menús adicionales que serán visibles cuando se utilice dicho árbol (Cada usuario podría utilizar un árbol diferente y por lo tanto menús diferentes). Por último cada opción en específico puede definir sus propios menús adicionales.

Los elementos de menú especificados en cada nivel, se unen para mostrar un menú unificado. Cuando son menús con un nombre distinto, simplemente se agregan al final de los elementos anteriores. En caso de que alguno de los menús del nivel 2 y 3 tenga el mismo nombre que uno del nivel 1, se sustituye la funcionalidad de ese elemento y no cambia de lugar.

Por ejemplo si tenemos:

_ks_frame.kpl	index.kpl	Pagina1.kpl
Menú	Menú	Menú
-Archivo - Cerrar	-Archivo -Info	-Edición -Copiar

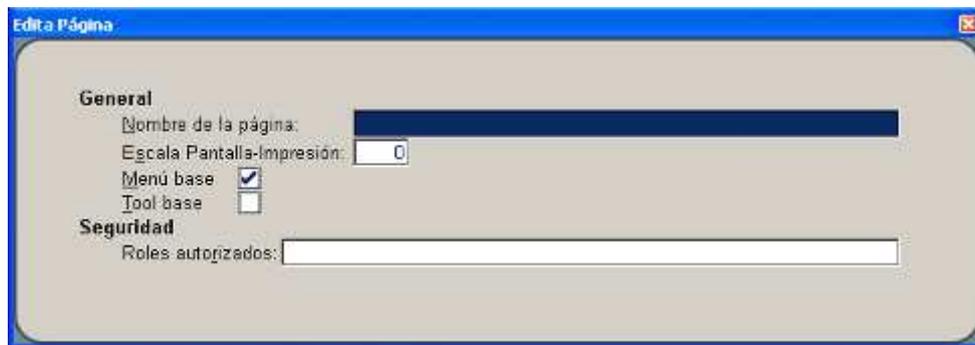
La página "**_ks_frame**" como ya se había comentado es una página no visible que indica la distribución general de la pantalla principal, pero si puede contener menús que serán visibles en cualquier momento. En este caso el menú "Archivo→Cerrar", se encontrará visible en todas las opciones

La página "**index.kpl**", es una página que contiene un árbol de opciones, entre las cuales se encuentra una liga a la "**Página1.kpl**". Cualquier página que esté indicada en el árbol contendrá los menús especificados en la página "**index.kpl**", además de los definidos en "**_ks_frame.kpl**". Por último cada página individual puede contener más elementos de menús que se agregarán a los ya definidos por "**index.kpl**" y por "**_ks_frame.kpl**".

Finalmente siguiendo el ejemplo, el usuario solamente verá el siguiente menú unificado al estar trabajando sobre la página "Página1.kpl":

-Archivo -Edición
--Cerrar --Copiar
--Info

NOTA: Es posible eliminar la funcionalidad de herencia de menús para una página en especial, especificando el atributo **menu="base"** al elemento **<pag>** o bien, se puede cambiar con la edición gráfica de una página en el menú **Formato> Página** en los campos **Menu Base** o **Tool Base**, como se muestra en la siguiente figura:



Ej. `<pag xmlns="www.kepler.com.mx/2003/KSchema" menu="base">`

Para indicar los menús de cada página es necesario ir a la edición XML y agregar el elemento **<menú>**, al elemento raíz **<pag>**. Ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<pag xmlns="www.kepler.com.mx/2003/KSchema" menu="base">
  <menu>
    <menu name="Archivo" hk="A">
      <menu name="Abrir..." hk="A" scut="Ctrl+A" dialog="_ks_open"/>
      <menu name="Cerrar" hk="C" scut="Ctrl+R" macro="2"/>
      <menu/>
      <menu name="Preferencias..." hk="P" dialog="_ks_properties"/>
      <menu/>
      <menu name="Cerrar Sesión" hk="e" macro="14"/>
      <menu name="Salir" hk="S" macro="1"/>
    </menu>
    <menu name="Edición" hk="E">
      <menu name="Deshacer" hk="D" scut="Ctrl+Z" macro="5"/>
      <menu name="Rehacer" hk="R" scut="Ctrl+Y" macro="6"/>
      <menu/>
      <menu name="Cortar" hk="a" scut="Ctrl+X" macro="9"/>
      <menu name="Copiar" hk="C" scut="Ctrl+C" macro="7"/>
      <menu name="Pegar" hk="P" scut="Ctrl+V" macro="8"/>
      <menu name="Borrar" hk="B" macro="10"/>
      <menu/>
      <menu name="Edición Gráfica" hk="f" scut="Ctrl+F" macro="16"/>
      <menu name="Edición XML" hk="M" scut="Ctrl+M" macro="19"/>
    </menu>
    <menu name="Ver" hk="V">
      <menu name="Split de Pantalla" hk="S" macro="15"/>
    </menu>
    <menu name="Ayuda" hk="y">
      <menu name="Acerca de..." hk="A" dialog="_about.kpl"/>
    </menu>
  </menu>
</pag>
```

Como se puede observar el elemento **<menu>** a su vez se compone de elementos hijo tipo **<menu>**, y de esta forma se define de forma jerárquica los menús de la página.

Cabe mencionar que un elemento menú vacío (**<menu/>**) indica una separador.

Características y Atributos XML:

General

- **name:** Identifica de forma única a cada elemento del menú. El código para un elemento llamado menu1 sería:

```
<menu name="menu1"/>
```

- **hk:** (HotKey) Esta característica permite posicionar el cursor sobre dicho elemento cuando el menú se encuentra desplegado. Adicionalmente, si la letra indicada forma parte del nombre subraya esa letra. Por ejemplo si el (hk) HotKey fuera la letra u, entonces al presionar "u" estando el menú desplegado se posicionaría directamente en dicho elemento, y si es el único elemento con dicho hot key, ejecuta la acción del elemento del menú.

```
<menu name="menu1" hk="u"/>
```

- **scut:** (Shortcut) Esta característica permite ejecutar la acción del elemento del menú como si se hubiera seleccionado con el Mouse, pero indicando una combinación de letras de teclado. Por ejemplo si el Shortcut de un elemento

de menú fuera “Ctrl+R”, bastaría con presionar en el teclado “Ctrl” y “R” para que se ejecute la acción de dicho elemento de menú.

```
<menu name="menu1" hk="u" scut="Ctrl+R"/>
```

- **dialog:** Esta característica indica qué página kepler se abrirá como ventana de diálogo al presionar dicho elemento de menú. Por ejemplo si se desea que al presionar el elemento “menu1” se abra la página kepler llamada “Pag2.kpl” se tendría”:

```
<menu name="menu1" hk="u" scut="Ctrl+R" dialog="Pag2.kpl"/>
```

- **macro:** Esta característica indica qué acción predeterminada se ejecutará al presionar dicho elemento de menú. Normalmente dichas acciones son muy específicas para determinadas situaciones, y se utilizan para dar la funcionalidad a la aplicación. Por ejemplo si se especifica que se ejecute la macro “9”, se indica que se “Cortará” algún elemento en la edición gráfica que esté seleccionado ó el texto del input actual:

```
<menu name="menu1" hk="u" scut="Ctrl+R" macro="9"/>
```

Para ver más detalle de las diferentes macros, consultar la documentación del Manual de Comandos.

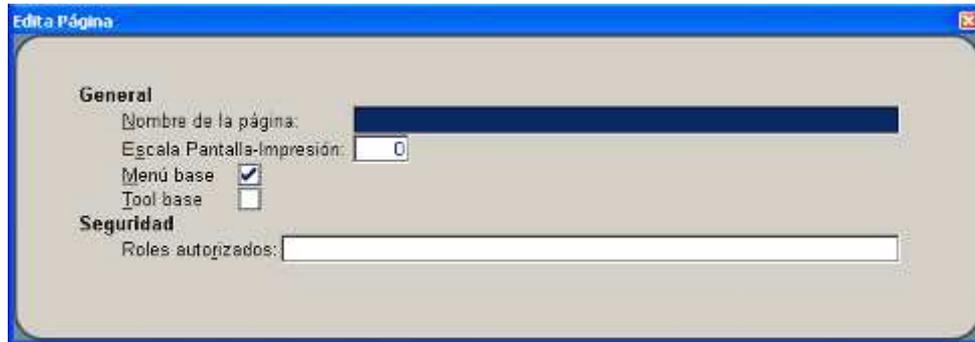
ToolBar



La barra de toolbar, es un conjunto de íconos con ciertas características y acciones. Siguen una lógica muy similar a los menús, en cuanto a que tienen 3 niveles de jerarquía y es necesario especificar el funcionamiento de cada elemento mediante la edición XML: Ejemplo:

```
<tools>
  <tool name=""/>
  <tool name="open" dialog="_ks_open" img="_Open.png" tooltip="Abrir archivo..."/>
  <tool name="line"/>
  <tool name="cut" macro="9" img="_Cut.png" tooltip="Cortar"/>
  <tool name="copy" macro="7" img="_Copy.png" tooltip="Copiar"/>
  <tool name="paste" macro="8" img="_Paste.png" tooltip="Pegar"/>
  <tool name="line"/>
  <tool name="home" macro="14" img="_Home.png" tooltip="Cerrar Sesión e ir a Home"/>
  <tool name="line"/>
  <tool name="top" macro="35" img="_Up.png"/>
  <tool name="bottom" macro="36" img="_Down.png"/>
  <tool name="line"/>
  <tool name="endleft" macro="31" img="_First.png"/>
  <tool name="left" macro="32" img="_Previous.png"/>
  <tool name="right" macro="33" img="_Next.png"/>
  <tool name="endright" macro="34" img="_Last.png"/>
  <tool name="line"/>
  <tool name="help" img="_Help.png"/>
  <tool name="line"/>
  <tool name="address" dim="" txt="Dirección" hk="D" tooltip="Dirección del Servidor"/>
</tools>
</pag>
```

NOTA: Es posible eliminar la funcionalidad de herencia de **toolbar** para una página en especial, especificando el atributo **tool="base"** al elemento **<pag>**, o bien, se puede cambiar con la edición gráfica de una página en el menú **Formato>Página** en los campos **Menú Base o Tool Base**, como se muestra en la siguiente figura:



Ej: `<pag xmlns="www.kepler.com.mx/2003/KSchema" tool="base">`

Como se puede observar el elemento `<tools>` se compone de elementos hijo tipo `<tool>`. Cabe mencionar que existen ciertos nombres reservados de tool tales como el de "address" que indica la barra de direcciones; "line" que indica un separador vertical, y algunos otros como "open", "cut", "copy", "paste", que no tienen acción predefinida pero si tienen imágenes predefinidas en caso de que no se especifique alguna otra..

Características y Atributos XML:

General

- **name:** Identifica de forma única a cada elemento del toolbar. El código para un elemento llamado tool1 sería:

`<tool name="tool1"/>`

- **hk:** (HotKey) Esta característica permite posicionar el cursor sobre dicho elemento. Solo sirve para elementos especiales que contienen una etiqueta, como "address".

`<tool name="address" hk="D"/>`

- **scut:** (Shortcut) Esta característica permite ejecutar la acción del elemento del toolbar como si se hubiera seleccionado con el Mouse, pero indicando una combinación de letras de teclado. Por ejemplo si el Shortcut de un elemento de toolbar fuera "Ctrl+O", bastaría con presionar en el teclado "Ctrl" y "O" para que se ejecute la acción de dicho elemento de toolbar.

`<tool name="open" scut="Ctrl+O"/>`

- **dialog:** Esta característica indica qué página kepler se abrirá como ventana de diálogo al presionar dicho elemento del toolbar. Por ejemplo si se desea que al presionar el elemento “tool1” se abra la página kepler llamada “Pag2.kpl se tendría”:

```
<tool name="tool1" dialog="Pag2.kpl"/>
```

- **macro:** Esta característica indica qué acción predeterminada se ejecutará al presionar dicho elemento del toolbar. Normalmente dichas acciones son muy específicas para determinadas situaciones, y se utilizan para dar la funcionalidad a la aplicación. Por ejemplo si se especifica que se ejecute la macro “9”, se indica que se “Cortará” algún elemento en la edición gráfica que esté seleccionado ó el texto del input actual:

```
<tool name="tool1" scut="Ctrl+R" macro="9"/>
```

- **img:** Esta característica indica qué imagen mostrará el ícono del toolbar. Por ejemplo si se desea que se muestre una imagen que se encuentra en la carpeta local llamada “imag1.jpg” se tendría:

```
<tool name="tool1" img=" imag1.jpg"/>
```

Nota: Se pueden utilizar los mismos formatos de imágenes que el elemento img. Automáticamente se ajusta el tamaño de la imagen.

- **tooltip:** Esta característica indica qué leyenda se mostrará al posicionarse sobre el ícono del toolbar. Por ejemplo si se desea que se muestre la leyenda “Abrir archivo...” al posicionarse sobre el elemento tool1 se tendría:

```
<tool name="tool1" tooltip=" Abrir archivo..."/>
```

Nota: Se pueden utilizar los mismos formato de imágenes que el elemento img. Automáticamente se ajusta el tamaño de la imagen.

Eventos. (A qué acciones responde el elemento)

- **OnMouseL:** Este evento se dispara al dar clic con el botón izquierdo del Mouse. Es necesario especificar la función de la misma página que se deberá ejecutar. Ej: Para ejecutar la función “fun1” al dar clic sobre el elemento de toolbar se tendría:

```
<tool name="tool1" OnMouseL="fun1()"/>
```

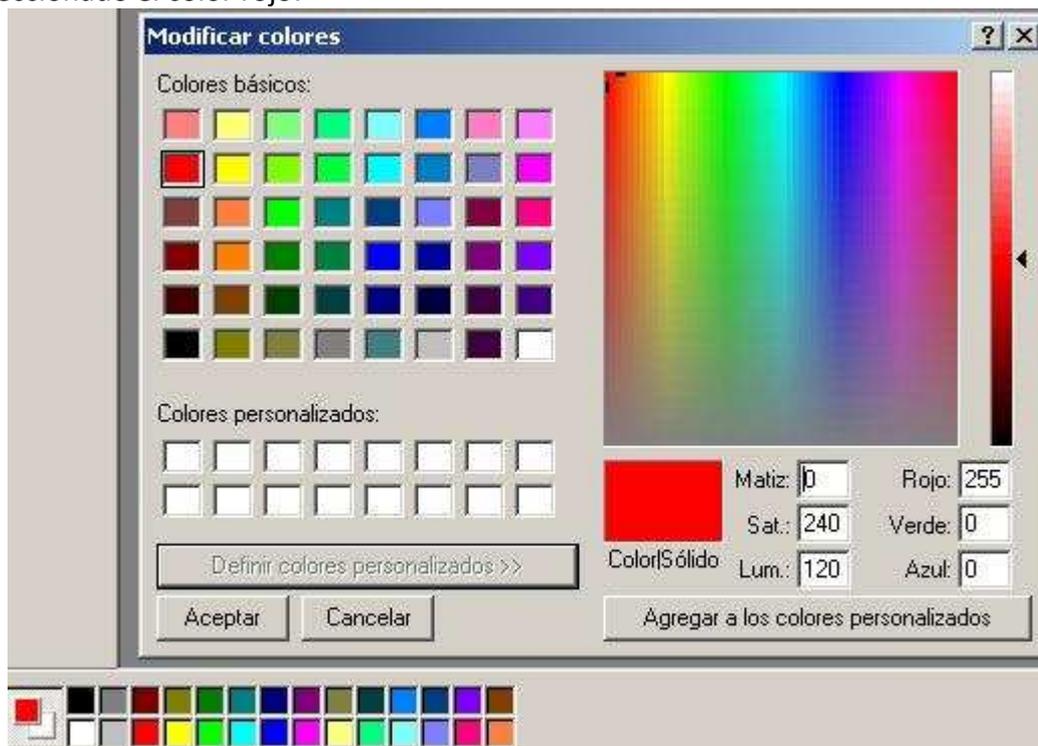
Evaluación

Una vez terminado el curso, favor de solicitar la evaluación a Casa Matriz, para poder acreditar los conocimientos adquiridos.

Glosario

* **RGB**: Se refiere a los componentes rojo, verde y azul en una escala de 0 a 255 en cada uno. Se forma un string en el siguiente orden: **Red, Green, Blue**
Por ejemplo, para indicar el color rojo, su equivalencia en notación RGB sería: “255,0,0”.

Para saber como expresar un color con el que se desea trabajar, se recomienda abrir el programa Paint (Inicio>>Programas>>Accesorios>>Paint). Después dar doble clic en la paleta de colores que se encuentra en la barra inferior y se abrirá una ventana con una paleta de colores mas extensa. Después dar clic en “Definir Colores Personalizados” y se ampliará la ventana mostrando las equivalencias en RGB, al seleccionar algún color. En la pantalla que se muestra a continuación se había seleccionado el color rojo:



También es posible especificar el color en notación hexadecimal anteponiendo el símbolo “#” y sin dividir los componentes con comas. Por ejemplo: “#FF0000” indica un color rojo.

** El formato de texto se compone del tipo de letra, tamaño y características (negritas, cursiva, subrayado). Se forma un string en el siguiente orden: “tipo_de_letra,tamaño[b][i][s]”

donde b = **negritas**, i= *cursiva*, s= subrayado.

Por ejemplo, el siguiente string: "Arial,12is", indica letra Arial de tamaño 12, cursiva, y subrayada.

TAB

Es equivalente a las tabulaciones se pueden agregar con cualquier procesador de palabras. La principal ventaja es que se puede definir el número de tabulaciones que se desee dar. El código es:

```
<tab n="2"/>
```

BR

Es equivalente a dar un "enter" con la tecla return. Esto ocasiona que el siguiente objeto u oración se verán en el siguiente renglón de la pantalla. El código para esta acción es:

```
<br/>
```