



Kara Effector 3.2:

Effector Book

Vol. II [Tomo XXIX]


Kara Effector 3.2:

En este **Tomo XXIX** continuaremos viendo las funciones de la **librería effector**. Esta librería contiene una serie de funciones interesantes que espero que con la ayuda de esta documentación, le puedan sacar el máximo provecho a la hora de llevar a cabo sus proyectos, no solo karaokes, sino también en la edición de los subtítulos.

Librería Effector [KE]:

1. **effector.pos**
2. **effector.knj**
3. **effector.offset_pos**
4. **effector.import**
5. **effector.addfx**
6. **effector.savefx**
7. **effector.modify_pos**
8. **effector.new_pos**
9. **effector.default_val**
10. **effector.effect_offset**
11. **effector.decide**
12. **effector.print_error**
13. **effector.run_fx**
14. **effector.preprocesses_styles**
15. **effector.preprocesses_macro**
16. **effector.preprocesses_lines**
17. **effector.macro_fx**

De las anteriores funciones, las que están en negro son las funciones administrativas, y las que están en azul son las que de una u otra manera le podemos sacar provecho.

 **effector.import(file_fx)**

Esta función nos permite importar un archivo **.lua** al **Kara Effector** para poder usar las funciones y variables que vienen en dicho archivo, al hacer uno o más efectos.

Esta función se usa en la celda de texto **"Variables"** y tiene una forma abreviada de llamarla:

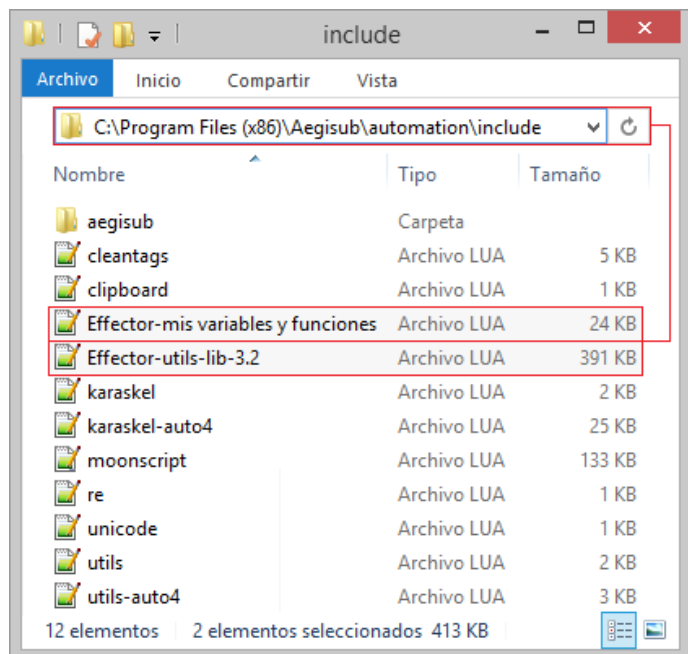
import(file_lua)

Hay dos formas de importar un archivo **.lua** al **KE**, una de ellas es guardar el archivo que queremos importar en la misma carpeta en donde esté guardado el archivo de la librería principal del **KE**: **Effector-utils-lib-3.2.lua**

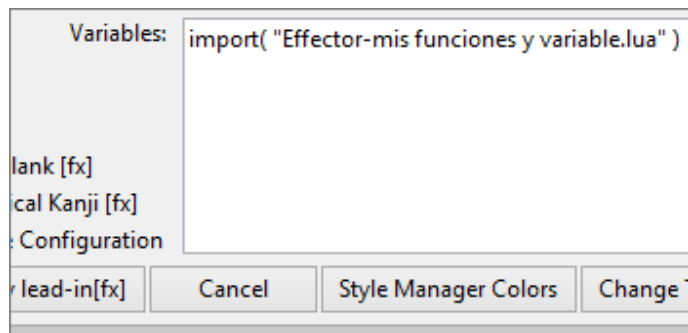
• **Método 1:**

El archivo **Effector-utils-lib-3.2.lua** del KE, la mayoría lo tenemos en la carpeta **"include"** del **Aegisub**, y es ahí en donde debemos guardar el archivo **.lua** que queremos importar al **Kara Effector**.

Ejemplo:

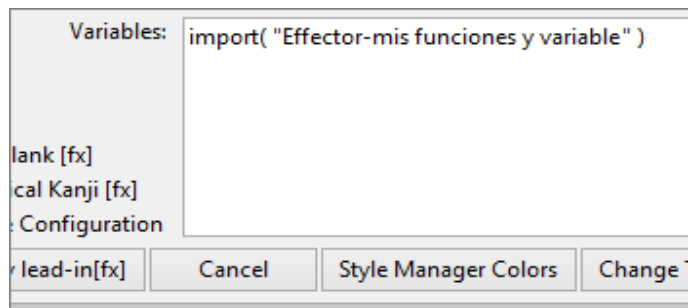


Y entonces, en la celda de textos **"Variables"** debemos poner así:



Obviamente no es necesario un nombre tan largo para el archivo **.lua** que vamos a importar, es solo un ejemplo.

Podemos omitir la extensión **.lua** del nombre del archivo y aun así poderlo importar satisfactoriamente:



• **Método 2:**

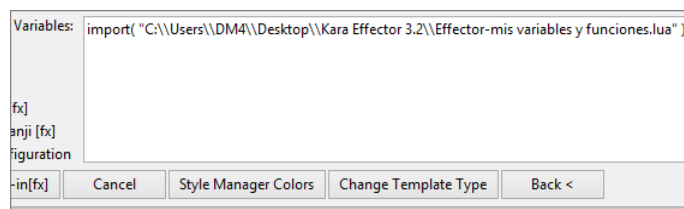
Con este método, no hay necesidad de guardar el archivo en la misma carpeta de la librería principal del **KE**, pero se debe poner la ruta completa de la posición del archivo dentro de la función:

Recordemos que las rutas deben ir con doble **"\"**.

Ejemplo:

import("C:\\Users\\DM4\\Desktop\\Kara Effector 3.2\\Effector-mis variables y funciones.lua")

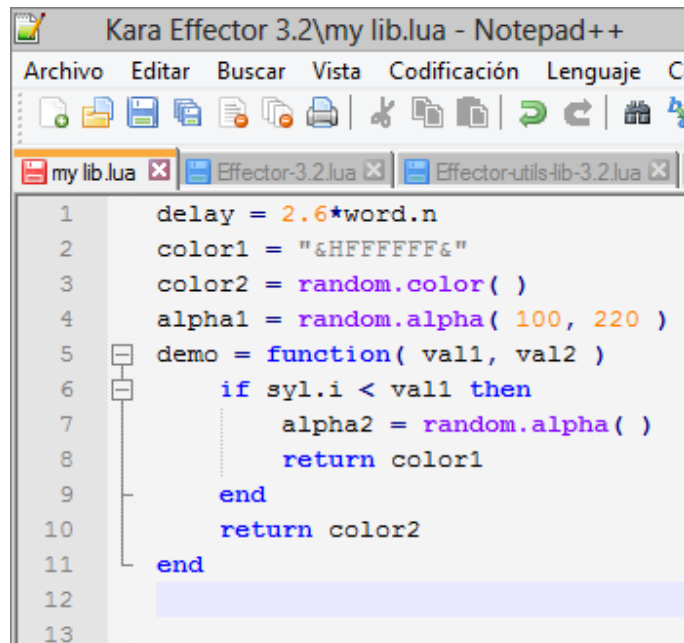
Y todo lo anterior debe ir en la celda de texto **"Variables"**:



Una vez comprendido la forma de importar un archivo **.lua** al **KE**, ya podemos ver un ejemplo de cómo puede ser dicho archivo y las ventajas que tiene el poder importarlo.

Ejemplo:

He creado un archivo llamado **"my lib.lua"** que contiene una serie de variables y una función sencilla, que he guardado en la misma carpeta en donde se encuentra la librería principal del KE:



Obviamente al ser un archivo **.lua**, el lenguaje a usar para declarar las variables y las funciones debe ser el **LUA**, lo que hasta este punto ya no debería ser un obstáculo tan difícil de superar como lo era al principio del aprendizaje.

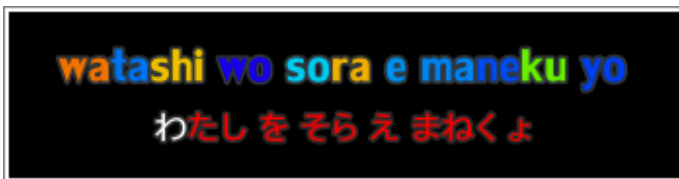
Guardado el archivo, lo importamos al **KE** en la celda de texto "**Variables**" como acabamos de aprender:

```
>> Variables:
import( "my lib" )
```

Una vez importado el archivo **.lua**, ya podemos disponer de cualquiera de sus variables o funciones declaradas en él, en cualquier celda de texto de la Ventana de Modificación del **KE**, incluso en la misma celda "**Variables**". Ejemplo:

```
Add Tags Language: >> Lua
"\1c" .. color2
```

Y al aplicar el efecto, el **KE** reconocerá la variable "**color2**" que está en el archivo **.lua** creado, porque previamente lo ha importado:



El poder declarar variables y funciones en un archivo **.lua** independiente del **KE** nos da la ventaja de tener más espacio y comodidad para hacerlo. Otra de las ventajas es la siguiente:

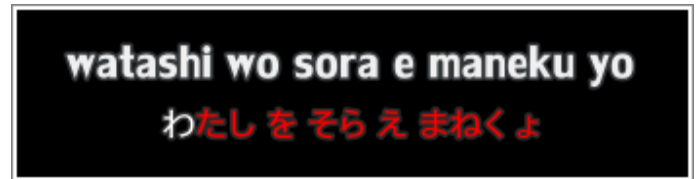
- Se puede modificar el archivo **.lua**, y la modificación se asume en tiempo real en el efecto en que importemos el archivo, sin la necesidad de tener que recargar el **KE**.

Ejemplo:

Modificamos la variable "**color2**":

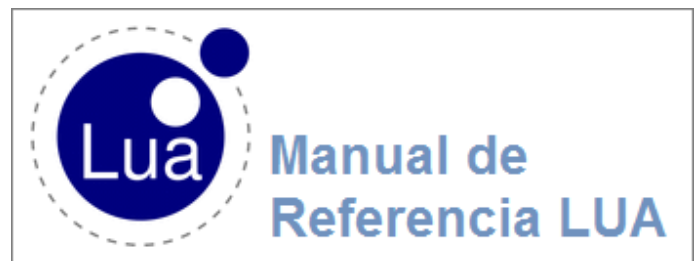
```
1 delay = 2.6*word.n
2 color1 = "&HFFFFFF&"
3 color2 = "&HFFFFFF&"
4 alpha1 = random.alpha( 100, 220 )
5 demo = function( val1, val2 )
6     if syl.i < val1 then
7         alpha2 = random.alpha( )
8         return color1
9     end
10    return color2
11 end
12
13
```

Y sin tener que recargar el **KE**, aplicamos el efecto como en el ejemplo anterior, y la modificación se verá reflejada de forma inmediata:



Saber y/o dominar el lenguaje de programación **LUA** no es un requisito para poder hacer efectos, ya que la razón del ser del **KE** es hacer sencillo el proceso de creación, edición y aplicación de los mismos. Para los que estén interesados en ampliar sus conocimientos en este lenguaje, recomiendo la siguiente web en español:

www.lua.org



O pueden descargar alguna de las ediciones en **PDF** del manual completo **LUA** en inglés:

- [Programando en LUA 2da Edición](#)
- [Programando en LUA 3ra Edición](#)



Al igual les dejaré ambos links en caso de que no funcionen los hipervínculos anteriores:

- 2da Edición: www.mediafire.com/?9aajra8ro6oukfa
- 3ra Edición: www.mediafire.com/?qf54ed3642rrn8o

Como les comentaba, este material es para aquellos que quieren profundizar en el lenguaje de programación **LUA** y aplicarlo en la creación de efectos, pero tengo pensado más adelante mostrarle unos cortos consejos de programación en **LUA** para hacer efectos en formato **.ass**

```
effector.addfx( library_fx, name_fx )
```

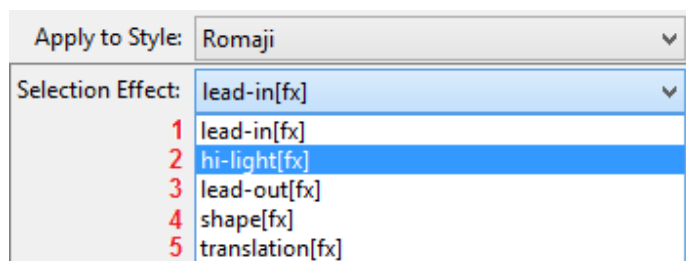
Esta función nos permite aplicar dos o más efectos al mismo tiempo, con la posibilidad de poder usar las variables y funciones declaradas en el primer efecto, en los demás adicionados. Su forma abreviada es:

```
addfx( library_fx, name_fx )
```

El parámetro **library_fx** es el nombre de alguna de las cinco librerías de efectos del **KE**:

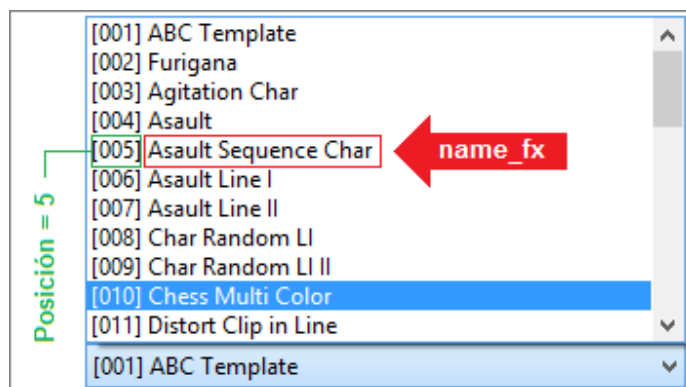
1. "lead-in"
2. "hi-light"
3. "lead-out"
4. "shape"
5. "translation"

Estas cinco librerías de efectos están en el mismo orden y hacen referencia a las opciones de la Ventana de Inicio del **Kara Effector**:



El parámetro **library_fx** puede ser el string del nombre de la librería de efectos o el número que indique su posición.

El parámetro **name_fx** es el string del nombre del efecto que queremos adicionar. Ejemplo:



Y de forma similar al parámetro **library_fx**, podemos poner el string del nombre del efecto a agregar o el número de la oposición que ocupa en su respectiva librería. Entonces podemos utilizar la función con cualquiera de las siguientes combinaciones:

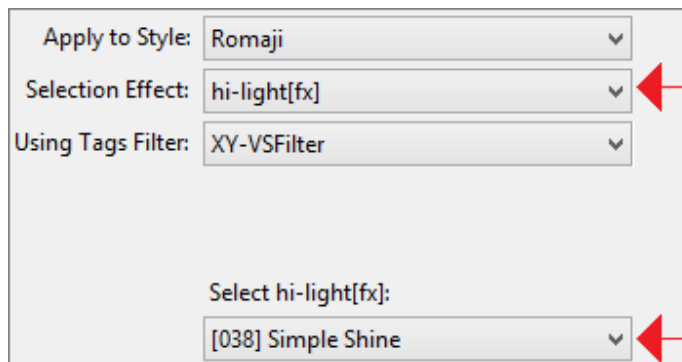
- addfx(nombre_librería, nombre_efecto)
- addfx(nombre_librería, posición_efecto)
- addfx(posición_librería, nombre_efecto)
- addfx(posición_librería, posición_efecto)

Esta función se usa en la celda de texto "**Variables**" y puede ser usada la cantidad de veces que así lo dispongamos.

Ejemplo:

Seleccionamos el primer efecto de la librería **lead-in[fx]** que es el efecto: **[001] ABC Template**

Y supongamos que queremos añadir a este lead-in el efecto "**Simple Shine**" de la librería **hi-light[fx]**:



Entonces tenemos la siguiente información del efecto que vamos a añadir:

- Librería: "**hi-light**"; Posición: 2
- Efecto: "**Simple Shine**"; Posición: 38

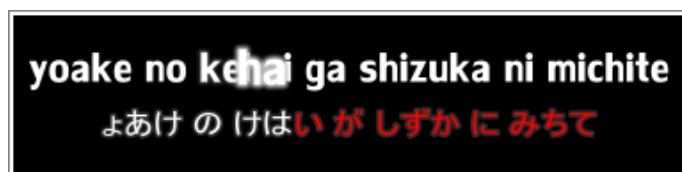
Y para añadirlo al lead-in[fx] "**ABC Template**", podemos hacerlo con algunas de las siguientes cuatro opciones en la celda de texto "**Variables**":

- addfx("hi-light", "Simple Shine")
- addfx("hi-light", 38)
- addfx(2, "Simple Shine")
- addfx(2, 38)

Ya sabemos que no importa cuál de las cuatro anteriores formas de hacerlo, la función añadirá el efecto al que ya hemos elegido, pero para el ejemplo usaré la primera, que es poniendo en nombre, tanto como el de la librería, como el del efecto:



Y al aplicar el lead-in, se aplicarán los dos efectos de forma simultánea, uno seguido del otro. Siempre se aplica primero el efecto de base, y luego todos los demás en el mismo orden en que los hayamos añadido:



Si queremos añadir más de un efecto a la vez, los debemos poner de la siguiente forma. Ejemplo:

```
>> Variables:
addfx( 2, 38 ); addfx( 4, 14 ); addfx( 3, 26 )
```

Lo que añadirá 3 efectos diferentes al que seleccionamos previamente en la Ventana de Inicio del KE.

El método para que una o más variables, al igual que las funciones declaradas en el primer efecto seleccionado en la Ventana de Inicio del KE, las podamos usar en los efectos a agregar, sin necesidad de tener que declararlas en ellos también, es el siguiente:

```
> Ejemplo:
>> Variables:
delay = 420; addfx( 2, 38 ); addfx( 4, 14 )
```

Al declarar la variable “**delay**” en este ejemplo, al inicio de la celda de texto “**Variables**”, nos aseguramos de que esta también pueda ser usada en todos los efectos añadidos que así lo dispongamos.

El aplicar más de un efecto al mismo tiempo es una ventaja cuando apenas estamos desarrollando el efecto total de un karaoke, cartel o logo; ya que no es evitará estar aplicando uno a uno a cada uno de los efecto individuales que lo componen.

```
>> effector.savefx( mode )
```

Es una función interna del KE que exporta una serie de información, tanto del script .ass, como de los efectos a aplicar. Consta de 12 modos:

1. "export config"
2. "export times ms"
3. "export times"
4. "export lines"
5. "export text"
6. "export hira"
7. "export kata"
8. "export roma"
9. "export text_stripped"
10. "export hira_stripped"
11. "export kata_stripped"
12. "export roma_stripped"

Toda la información que exporta esta función, lo hace en un archivo nuevo en formato .txt, y los modos de la función se deben ingresar en la celda de texto “**New [fx] Name**”:

The screenshot shows the Kara Effector interface. At the top, there are fields for 'Add Tags' and 'Add Tags Language' (set to 'Lua'). Below these is a large text area. At the bottom, there is a checkbox for 'Print Config [fx]' and a dropdown for 'Template Folder [fx]' (set to 'lead-in[fx]'). A red arrow points to the 'New [fx] Name' field, which is currently empty.

Lo que debemos hacer es escribir el modo en esta celda de texto, sin las comillas, y aplicar el efecto como siempre lo hacemos de forma normal.

The screenshot shows the Kara Effector interface. The 'New [fx] Name' field now contains the text 'export config'.

Exporta en un archivo .txt a todas las configuraciones del efecto, que posteriormente puede ser compartido con otros usuarios del KE.

```
> Ejemplo:
```

Al usar cualquiera de los doce modos de la función, nos debe salir un cuadro de dialogo como el de la siguiente imagen, en donde decidiremos el lugar de destino del archivo y el nombre del mismo:

The screenshot shows a file explorer window titled 'Kara Effector[fx]: Export File'. It displays a list of files and folders in the 'Documentos' directory. The 'Nombre' field is highlighted with a red box, and the 'Tipo' is set to 'Text files (.txt)'. The 'Guardar' (Save) button is visible at the bottom right.

Este es el método rápido de exportar las configuraciones del efecto en un archivo .txt, y una vez hallamos decidido en lugar de destino, le ponemos un nombre a dicho archivo:

The screenshot shows the 'Nombre' field in the export dialog box. It now contains the text 'mi nuevo efecto'.

Al abrir el archivo exportado, veremos algo como esto:

```
--[[lead-in fx      10/17/14 13:57]]
leadin_fx_436222 = table.duplicate(PfxM_Box);
table.inbox(leadin_fx_436222, "lead-in[fx]:
leadin fx
436222", "Syl", true, false, "#F6F3F3", "#ADC4D6", "#6A
8DD6", "0", "0", "0", "1.start_time", "1.end_time", "",
"", "", "", "syl.center", "syl.middle", "", "", "5", "0",
"fx.pos_x", "fx.pos_y", "", "1", "", "syl.text", "", "",
"Lua", false); table.insert(leadin_fx_library,
leadin_fx_436222); table.insert(leadin_fx,
"leadin fx 436222")
```

De la anterior imagen del archivo exportado se resaltan tres rectángulos de diferentes colores:

- En verde, el nombre del archivo .txt
- En rojo vemos 2 datos, primero la librería de efectos en la que se guardará el efecto una vez que hallamos pegado estas configuraciones en el KE, en el archivo **Effector-newfx-3.2.lua**. y segundo, la fecha y hora de creación del archivo. La fecha en formato Mes/Día/Año y la hora en HH:MM
- En azul, el nombre que por default el KE le asigna a las configuraciones de un efecto al cual no le hallamos puesto uno.

El método para asignarle un nombre al efecto exportado en un archivo .txt es el siguiente:

New [fx] Name: **export config, Nombre del efecto**

O sea, escribimos el modo **"export config"** y el nombre que le pondremos al efecto, separados por una coma (,). Ambos sin comillas.

Modo 2:

New [fx] Name: **export times ms**

Exporta un archivo .txt con los tiempos de inicio y final en ms de las líneas seleccionadas para aplicar un efecto:

```
2430,8160
8330,13190
13540,18390
18670,27220
28520,34300
34300,39310
40700,45790
45920,54560
```

Modo 3:

New [fx] Name: **export times**

Exporta un archivo .txt con los tiempos de inicio y final en formato HH:MM:SS.ms de las líneas seleccionadas para aplicar un efecto:

```
0:00:02.430,0:00:08.160
0:00:08.330,0:00:13.190
0:00:13.540,0:00:18.390
0:00:18.670,0:00:27.220
0:00:28.520,0:00:34.300
0:00:34.300,0:00:39.310
0:00:40.700,0:00:45.790
0:00:45.920,0:00:54.560
```

Modo 4:

New [fx] Name: **export lines**

Exporta un archivo .txt con la información completa de las líneas seleccionadas para aplicar un efecto, con el mismo formato de un script .ass:

```
Dialogue: 0,0:00:02.43,0:00:08.16,Romaji,,0,0,,{\k17}ko{\k16}do{\k37}ku {\k32+fx}na {\k34}ho{\k28
-fx}ho {\k68}wo {\k32}nu{\k22}ra{\k74}su {\k34}nu{\k19}ra{\k46}su {\k32}ke{\k82}do
Dialogue: 0,0:00:08.33,0:00:13.19,Romaji,,0,0,,{\k18}yo{\k14}a{\k37}ke {\k28}no {\k36}ke{\k29}ha
{\k34}i {\k25}ga {\k36}shi{\k35}zu{\k31}ka {\k30}ni {\k54}mi{\k13}chi{\k66}te
Dialogue: 0,0:00:13.54,0:00:18.39,Romaji,,0,0,,{\k32}wa{\k20-fx}ta{\k45}shi {\k24}wo {\k40}so
{\k34}ra {\k32}e {\k64}ma{\k64}ne{\k65}ku {\k65}yo
Dialogue: 0,0:00:18.67,0:00:27.22,Romaji,,0,0,,{\k20}ki{\k21}bo{\k32}u {\k31}ga {\k35}ka{\k33}na
{\k32}ta {\k31}de {\k18}ma{\k13}t{\k21}te{\k33}ru {\k16}so{\k28}u {\k18}da {\k49}yo {\k30}i{\k18}ku
{\k37}yo
Dialogue: 0,0:00:28.52,0:00:34.30,Romaji,,0,0,,{\k19}ma{\k18}yo{\k28}i {\k32}na{\k31}ga{\k35}ra
{\k59}mo {\k38}ki{\k32}mi {\k55}wo {\k44}sa{\k19}ga{\k46}su {\k31}ta{\k91}bi
Dialogue: 0,0:00:34.30,0:00:39.31,Romaji,,0,0,,{\k21}su{\k16}re{\k37}chi{\k32}ga{\k31}u {\k23}i{\k44}shi{\k33}ki
{\k21}tsu{\k15}ka{\k18}ma{\k33}e{\k17}ru {\k167}yo {\k43}shi{\k31}
{\k21}mo{\k15}to{\k18}me {\k37}a{\k25}u {\k67}ko{\k68}ko{\k56}ro {
```

Modo 5:

New [fx] Name: **export text**

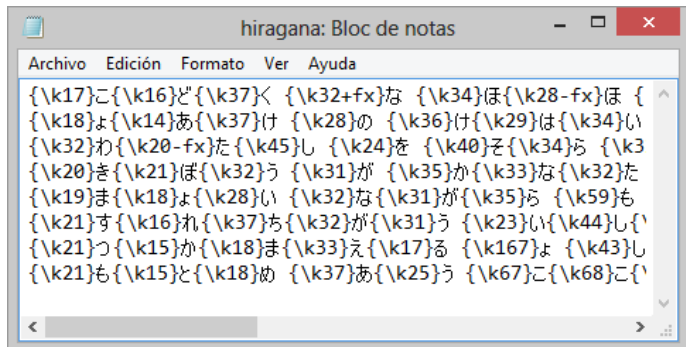
Exporta un archivo .txt con el contenido completo de las líneas seleccionadas para aplicar un efecto, incluidos los tags (etiquetas):

```
{\k17}ko{\k16}do{\k37}ku {\k32+fx}na {\k34}ho{\k28-fx}ho {\k68}wo
{\k18}yo{\k14}a{\k37}ke {\k28}no {\k36}ke{\k29}ha{\k34}i {\k25}ga
{\k32}wa{\k20-fx}ta{\k45}shi {\k24}wo {\k40}so{\k34}ra {\k32}e {\k
{\k20}ki{\k21}bo{\k32}u {\k31}ga {\k35}ka{\k33}na{\k32}ta {\k31}de
{\k19}ma{\k18}yo{\k28}i {\k32}na{\k31}ga{\k35}ra {\k59}mo {\k38}ki
{\k21}su{\k16}re{\k37}chi{\k32}ga{\k31}u {\k23}i{\k44}shi{\k33}ki
{\k21}tsu{\k15}ka{\k18}ma{\k33}e{\k17}ru {\k167}yo {\k43}shi{\k31}
{\k21}mo{\k15}to{\k18}me {\k37}a{\k25}u {\k67}ko{\k68}ko{\k56}ro {
```

Modo 6:

New [fx] Name: **export hira**

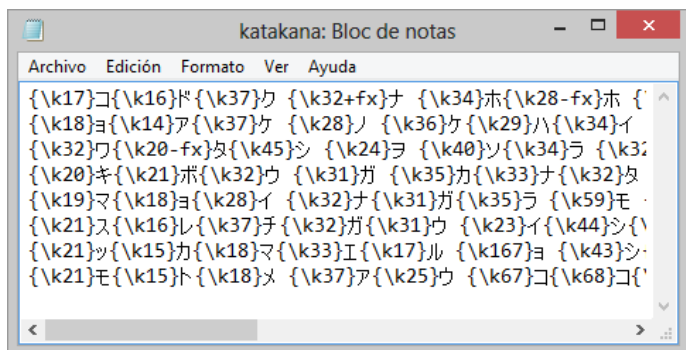
Exporta un archivo .txt con el **romaji** o el **katakana** de las líneas karaoke seleccionas, convertidos en **hiraganas**:



Modo 7:

New [fx] Name: **export kata**

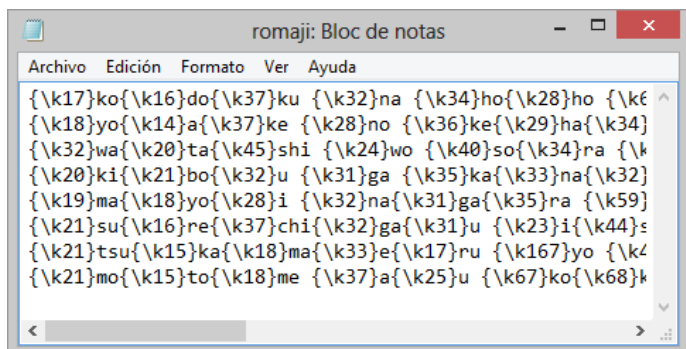
Exporta un archivo .txt con el **romaji** o el **hiragana** de las líneas karaoke seleccionas, convertidos en **katakanas**:



Modo 8:

New [fx] Name: **export roma**

Exporta un archivo .txt con el **hiragana** o el **katakana** de las líneas karaoke seleccionas, convertidos en **romajis**:



Modo 9:

New [fx] Name: **export text_stripped**

Este modo es similar al modo 5, pero exporta el contenido de las líneas de texto seleccionadas sin los tags (etiquetas).

Modo 10:

New [fx] Name: **export hira_stripped**

Este modo es similar al modo 6, exporta el contenido de las líneas de texto seleccionadas convertidas a **hiraganas**, pero sin los tags (etiquetas).

Modo 11:

New [fx] Name: **export kata_stripped**

Este modo es similar al modo 6, exporta el contenido de las líneas de texto seleccionadas convertidas a **katakanas**, pero sin los tags (etiquetas).

Modo 12:

New [fx] Name: **export roma_stripped**

Este modo es similar al modo 6, exporta el contenido de las líneas de texto seleccionadas convertidas a **romajis**, pero sin los tags (etiquetas).

No está demás aclarar que los modos que exportan a los **romajis**, **katanas** y **hiraganas** están pensados para las líneas de karaoke que tienen las sílabas separadas, es decir que no aplican a las líneas de traducción normales.

Es todo por ahora para el **Tomo XXIX**. Intenten poner en práctica todos los ejemplos vistos y no olviden descargar la última actualización disponible del **Kara Effector 3.2** y visitarnos en el **Blog Oficial**, lo mismo que en los canales de **YouTube** para descargar los nuevos Efectos o dejar algún comentario. Pueden visitarnos y dejar su comentario en nuestra página de **Facebook**:

- www.karaeffector.blogspot.com
- www.facebook.com/karaeffector
- www.youtube.com/user/victor8607
- www.youtube.com/user/NatsuoKE
- www.youtube.com/user/karalaura2012