

# wListView

```

CLASS ZListView FROM WControl

    PROPERTY cClassId      INIT "lvw"
    PROPERTY aItems        INIT {}           // Array de WItemListView
    PROPERTY cType         INIT xc_Default   // xc_Avatar, xc_Collapsible,
xc_Expandable, xc_Popout
    PROPERTY lTitleItem    INIT .F.         // Se presenta el cTitle de los
WItemListView

    PROPERTY oIconEdit     AS WIcon

    PROPERTY cIconEdit     INIT ""          WRITE INLINE ::oIconEdit:cIcon :=
Value READ INLINE ::oIconEdit:cIcon // Icono para la función de editar
    PROPERTY cOnClickEdit  INIT ""         // Funcion del CGI a llamar
cuando se pulse el Edit

    PROPERTY oIconDelete   AS WIcon
    PROPERTY cIconDelete   INIT ""          WRITE INLINE ::oIconDelete:cIcon :=
Value READ INLINE ::oIconDelete:cIcon // Icono para la función de editar
    PROPERTY cOnClickDelete INIT ""         // Funcion del CGI a llamar
cuando se pulse el Borrar

    PROPERTY cAjaxBevel    INIT ""          // Bevel donde se incrusta el
resultado de la llamada al Ajax
    PROPERTY oAvatarStyle  AS WStyle       // Stylos para la imagen del
Avatar

    PROPERTY cAjaxBevelEdit INIT ""
    PROPERTY cAjaxBevelDelete INIT ""

    PROPERTY lAjaxWaitON   INIT .T.        // Incluye un indicador de
actividad mientras se ejecuta la función Ajax
// PROPERTY lModalResult  INIT .F.        // Indicamos que al pulsar Edit
la ejecución va a salir por un Bevel Modal
    PROPERTY aParams       INIT {}         // Parametros extras a enviar al
CGI {"Param1","Valor1"},"Param2","Valor2"}
    PROPERTY cClrPaneDetail INIT ""        WRITE INLINE ::FcClrPaneDetail
:= ::CheckClr(Value)

// Color del fondo del detalle
    PROPERTY cClrText      INIT ""        WRITE INLINE ::FcClrText :=
::CheckClr(Value)

// Color del Texto
    PROPERTY cClrTextDetail INIT ""        WRITE INLINE ::FcClrTextDetail
:= ::CheckClr(Value)

// Color del Texto del Detalle

// Propiedades para la Paginación
// Hay que alimentar aItems con todos los elementos, el envia a la web solo

```

los de la página a la vista.

```
// Hay que crear un Bevel que contendra a todo el ListView para poder
actualizarlo al paginar
PROPERTY nItemInPage    INIT 0           // Si el valor es > 0 se realiza
paginación

    property nPage        INIT 1           // Página a presentar
    PROPERTY cOnChangePage INIT ""         // Función del CGI cuando
cambiamos de página
    PROPERTY cAjaxPages   INIT ""         // Bevel al que enviarle el
ListView por Ajax
    PROPERTY lAutoPage    INIT .T.        // Recibimos todos los datos y
es el ListView el que selecciona lo que va en la página
    PROPERTY nMaxItems    INIT 0           // Solo se utiliza si lAutoPage
== .F.
    PROPERTY lMultiPagesFirstLast INIT .t.
    PROPERTY nMultiPagesDots    INIT 0

// -----

METHOD Create()           // Creación del Control
METHOD AddItem(nRowId,cImage,cTitle,aText,lEdit,lDelete,aParams,cIcon)
METHOD AddArrayItems( aItems )           // Para crear los Items desde un
array {{nRowId,cImage,cTitle,aText,lEdit,lDelete,aParams,cIcon,cClrPane}}

/*
    TODO: La parte de DELETE no esta optimizada, por defecto uso ajax y el
cambio lo aplica sobre el propio listView
    TODO: Preparar una versión que no pague automáticamente, sino que le
enviamos directamente la página en aItems
*/

PROPERTY cIconOpen    INIT "" // "keyboard_arrow_down"
PROPERTY cIconClose   INIT "" // "keyboard_arrow_right"

METHOD New( oParent )
METHOD CambiaIcono
METHOD CreateItem ( oItem )

DATA nFinItem // Eran locales, pasadas a Data por createitem
DATA nIniItem
ENDCLASS
```

From:

<https://nefele.dev/wiki/> - Nefele Project

Permanent link:

<https://nefele.dev/wiki/controles/wlistview?rev=1604690447>

Last update: **06/11/2020 19:20**



