wForm

El control wForm es el contenedor estándar para el tratamiento de formularios, equivale a <form> de HTML.

Puede contener cualquier control de edición de datos, para una vez realizado el submit enviar el valor de todos los controles a nuestro CGI para poder tratarlos.

Para realizar en envió (submit) sera necesario añadir un wButton con la propiedad **lSubmit** a .T., solo puede existir un botón submit en cada formulario, también podemos asignarle la funcionalidad submit a un wEdit que este en el formulario.

Normalmente en los formularios HTML cuando en un campo de texto pulsamos **[Intro]** se ejecuta el submit de dicho formulario. En Néfele, al pulsar **[Intro]** pasamos el foco al siguiente control, excepto que dicho wEdit tenga su propiedad **lSubmit** a .T., asignándola directamente o utilizando el método **:SetSubmit()**, en ese caso se ejecutaría el submit del formulario al pulsar **[Intro]** en dicho control.

```
El ultimo control de un Formulario provoca el submit al pulsar [Intro] tenga o no la propiedad lSubmit.
```

El Submit enviara al CGI un parámetro por cada control, siempre que tengan algún valor, por ejemplo los wCheckBox si están a False no se envía ningún parámetro, por eso es muy practico utilizar le método oCGI:GetCGIValue con el parámetro **xdefault** con el valor que nos debe retornar en caso de que no venga.

Los controles que permitan múltiples valores, retornaran un array por lo que es más apropiado tratarlos con oCGI:GetCGIArray.

En HTML los controles que estén **disabled**, por defecto no se envían en el submit, tengan o no valor. En Néfele si tiene valor se tratan como al resto de controles.

Por defecto se le asigna el foco al primer control del Formulario que permita edición, si queremos que sea otro deberemos indicarlo en la propiedad **lAutoFocus** de dicho control, y habilitar la propiedad **lNoAutoFocus** del Formulario.

Cuando en el formulario se utilicen controles que necesiten enviar algo más que texto, sera necesario habilitar la propiedad **lFormData** del formulario, si los controles que lo necesitan son de Néfele no sera necesario hacerlo ya que ellos mismo lo asignaran si es necesario.

Ejemplo de uso

```
WITH OBJECT WBevel():New(:WO)
  :cTitle      := "Introduzca Usuario y Contraseña"
  :cTitleAlign := xc_Center
  WITH OBJECT WForm():New(:WO)
    :cFunction := "visparam"
```

```
      :aParams   := {{"control","cadena de control"}}
      WITH OBJECT WEdit():New(:WO)
        :cId   := "usuario"
        :cHint := "Usuario"
        :cIcon := "person"
        :SetRequired()
        :Create()
      END WITH
      WITH OBJECT WEdit():New(:WO)
        :cId   := "password"
        :cHint := "Password"
        :cIcon := "vpn_key"
        :SetPassword()
        :SetRequired()
        :Create()
      END WITH
      WITH OBJECT WButton():New(:WO)
        :lSubmit := .T.
        :cText   := "Enviar"
        :Create()
      END WITH
      :Create()
    END WITH
    :Create()
  END WITH
```

Control de reenvió de formularios

Para evitar que el usuario provoque el reenvío del formulario, porque de más de una vez al summit o porque de al botón atrás del navegador, hemos de habilitar la propiedad ::lUniqueId de forma que cada wForm se generará con un identificador único por cada llamada que se realice al CGI.

Luego en el Método de nuestro CGI que ha de procesar el summit del formulario controlaremos primero la validez del identificador del wForm, para lo que utilizaremos la función nfl_ChkFormUID que nos retornara .T. o .F. dependiendo de si el identificador es valido o no.

Propiedades

Nombre	Init	Descripción	Versión
cClassId	frm		0.1
cFunction		Función del CGI que sera llamada al hacer Submit del formulario	0.1
aParams	{}	aParams que se añadirán al submit aparte del los input visibles	0.1
lAutoComplete	.t.	Habilita o no el Autocompletado de los campos por el Navegador	0.1
cMethod	POST	Método de envió de los datos del formulario (POST o GET)	0.1
lFormData	.f.	Habilita el envió de ficheros adjuntos	0.1

Nombre	Init	Descripción	Versión
cFormAction		URL a la que se envían los datos del formulario, si es vació se envían al mismo CGI	0.1
lNoValidate	.f.	Evita que se validen los campos del formulario	0.1
lNoAutoFocus	.f.	Deshabilita la asignación automática del Foco al primer control editable	0.1
lUniqueId	.f.	Genera un código único para la llamada actual a este formulario para evitar la rellamada	0.4

Hereda de...

wControl

Propiedades

Nombre	Init	Descripción	Versión
cClassId		Identificador del control para generar el Id	0.1
cId	Nil	Identificador único para cada Control	0.1
oParent		Control contenedor	0.1
oHtml		Objeto de la clase wHtml con el código para su renderizado	0.1
oStyle		Objeto de la clase wStyle con las variantes de estilo	0.1
lEnabled	.t.	El control aparece Habilitado o Inhabilitado	0.1
lHide	.f.	El control aparece Oculto o Visible	0.1
aWidth	{12,,,}	Ancho del control según el tamaño de la pantalla	0.1
aOffset	{,,,}	Margen izquierdo según el tamaño de la pantalla	0.1
cClrText	black	Color de Texto	0.1
cClrPane	white	Color del Fondo	0.1

14/11/2019 14:37 · fredy

Métodos

Nombre	Descripción	Versión
Create	Este es el método que crea cada uno de los controles, esta desarrollado en cada Control	0.1
FullHtml	Nos devuelve una cadena con todo el código HTML y JS necesario para crear el control	0.1
LogDebug	Nos Añade un mensaje a la consola del navegador cuando estamos en modo Debugger	0.1
Assert	Nos Añade un comentario en el HTML cuando estamos en modo Debugger	0.1
AddHTML	Nos Añade código HTML dentro del control	0.1
AddParam	Añade un parámetro a la propiedad aParams, hay que pasarle un array {“nombre”,“valor”}	0.2

14/11/2019 14:37 · fredy

From:
https://nefele.dev/wiki/ - **Nefele Project**

Permanent link:
https://nefele.dev/wiki/controles/wform?rev=1621779761

Last update: **23/05/2021 14:22**