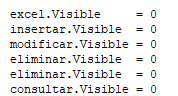
APLICAR PATTERN 🡪 TRABAJAR CON (ENTITY SERVICES) Y TRN FORM (POR WW)

ATAJOS:

* Control + shift + a 🡪 selecciona objetos
* Control + shift + j 🡪 atributos
* Control + shift + w 🡪 buscar
* Control + shift + v 🡪 variable
* Shift + tab 🡪 acomodar codigo

1. Primero trabajar con (aplicamos)
2. Vemos los stardart actions (ABM); observamos en GX90 mediante eventos si están o no

esto indica que va todo en false , si seria un 1 es al revés, si no aparecen quiere decir que tambien

1. Aplicamos la grilla

* Ordenar los atributos que se usan
* Cambiamos su descripción
* Los que no se usan van en NO VISIBLE
* Si son variables, poner en que se basan y su descripcion

1. Agregar parámetros (si hay)

* Nombre con &
* En que está basado (siempre sin el &) o indicar su dominio (si no es atributo)

1. Agregamos filtros (si hay)

* Si tenemos generados automáticamente; LOS BORRAMOS TODOS Y LO HACEMOS NOSOTROS
* Condition de gx90; los & son las variables que usa
* Del otro lado son los que usa filtros (ID)



* Si vemos la siguiente imagen quiere decir “desde – hasta”



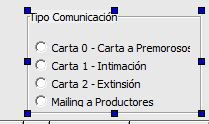
* Esto nos indica que va twofields en propiedades (manual imput type)
* Antes de crearlo de 0 como filtro, podemos hacerlo directamente desde la grilla (add filter for this attribute) y en ID le ponemos “\_filter” (agregar descripción)
* En este caso al ser fecha; data range (display in: column) [si no tiene valores nullvalue(ID)] [si los tiene serverdate() y servernow() ]
* Para cerrar en conditions 🡪 vemos el to y from (desde – hasta)
* Si es numérico con este formato (desde – hasta); filter type: numeric range y default & from : ponemos 0 a 999999 (como en este caso según el evento start)



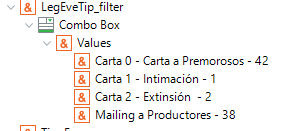
* Si es standart (filter type); display in: column (si no tira error xd) (puede tener, segun el evento start) values –> default: &today

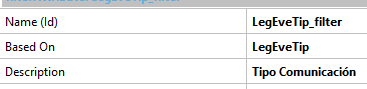


1. POR COMBO BOX:



* Agarramos el filtro 🡪 control info 🡪 edit 🡪 combo box
* Luego add values 🡪 value 🡪 agregamos ID, based on y descripcion (igual en GX90)
* SI ESTA EN EVENTOS Y NO POR CONDITIONS 🡪 creamos el filtro con el nombre de la variable (ID) y lo basamos en la misma (en este caso &tipcar = legevetip)





1. Agregar orders (si hay)

* Agregamos las variables que están en rules



* En este caso es ascendente ((xxx) , xxx) 🡪 esto nos indica que el primero va descendente
* Los que no se usan se borran

1. Si tenemos variables de filtro que estén por parámetro

* Workwith 🡪 grid conditions (copio y pego la Condition de GX90)

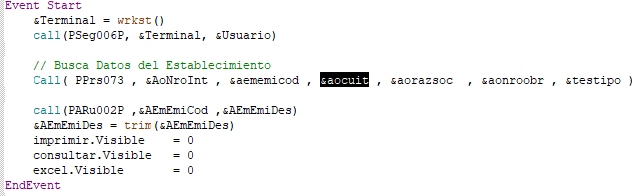


AHORA SI TENEMOS ALGO ASI:



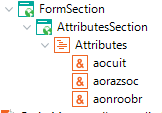
Si no está en Read Only 🡪 quiere decir que se pueden editar (si fuera read Only, seria cabecera )

Y VEMOS ALGO ASI:

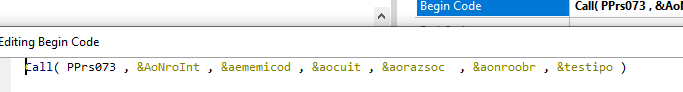


NOS ESTA INDICANDO QUE DENTRO DEL EVENTO START EN UN CALL (DENTRO DE UN PROCEDIMIENTO ESTA LA VARIABLE QUE ACTUA DE FILTRO), hacemos lo siguiente:

* Creamos dentro del workwith (WW) 🡪 FormSection, AttributesSection (ponemos en propiedades 🡪 Top ya que está arriba de la grilla), Attributes y en este caso Variable 🡪 van las que están arriba de todo (ignoramos si en otro prodecimiento estén con otro nombre y otro basado)



* Abrimos el procedimiento (en este caso PPrs073)
* Y nos fijamos por el orden (esta 3ro en este caso) si se llama igual o es con otro nombre dicha variable
*  (en el procedimiento)
* esto quiere decir que &tescuit está usando al &aocuit (por ende, debo confirmar si existe el procedimiento en gx17, si no lo importo)
* y luego crear el evento start 🡪 porque se encuentra ahí el prodecimiento (en este caso)
* meto solo en begin code (en este caso porque es lo primero que aparece) el call de la foto de arriba (en este caso)



* luego en workwith 🡪 open WW 🡪 eventos y vemos si esta ese call y agregamos sus variables (parámetros) en workwith 🡪 variables (ver que estén basados en el mismo al agregarlos) 🡪 solo los que no están arriba de todo

confirmamos que esta todo correcto

si en propiedades sale que están basados, nos indica que están agregados

1. Detalles necesarios a poner en Entity Services

* Borrar los subworkwith
* Workwith 🡪 allow grid view persistance 🡪 true
* Cambiar los nombres de workwith (plural) y entity services (singular) 🡪 form caption mismo nombre
* Revisar si el object name es WW

1. Al trabajar con WW y no un web panel a mano (en mi caso lmoliterno)

* Abrimos el web panel ligado a la pantalla de berkley (nos lo da toto)
* Nos vamos al event 🡪 borramos el ultimo call y ponemos el ww.setempty()
* En la trn y parámetro/s (tenemos que crearlos seguramente, los ponemos arriba del call)
* Creamos variables (evento start gx90 🡪 wrkst NO VA)

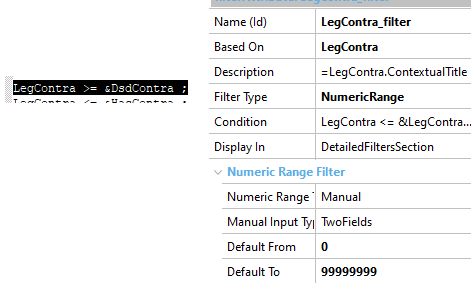
1. El primer call tiene variables
2. Creamos, ponemos su nombre y vemos su base on (click derecho 🡪 edit) 🡪 recomendado basarlo en su atributo si tiene
3. Creamos un evento



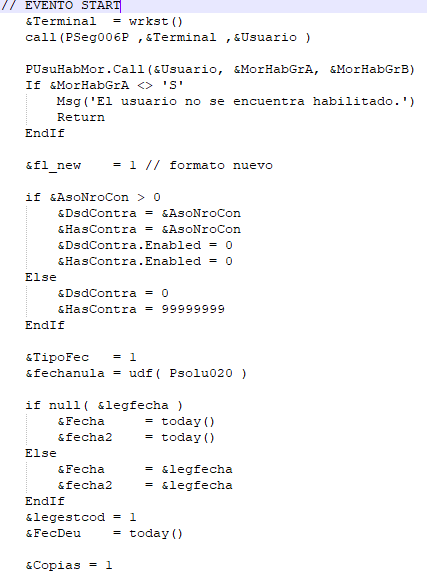
* Su nombre : start
* Begin code: copiamos (gx90) el primer call y pegamos (gx17)
* El objeto que llama al call debamos emigrarlo (import) y cambiamos su carpeta por el web panel
* Mensaje: K2BToolsMsg('', K2BToolsMessageType.Error)

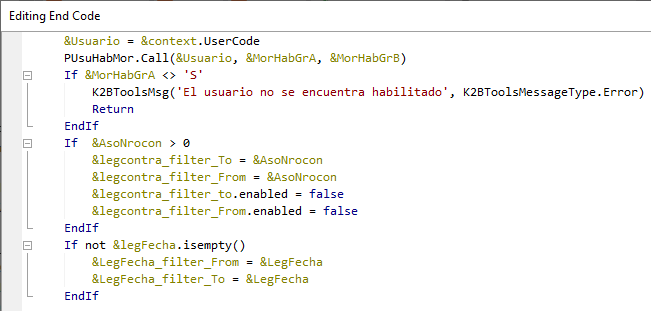
.error / .warning /.access /.info 🡪 cambia de color el msj

* Los otros 2 IFS seria casi lo mismo, &dscontra (es un filtro) del cual seguro nos pone from y to 

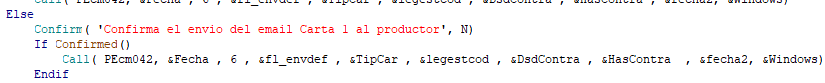


* Los IF: begin code
* Mensaje: edit code
* Legfecha\_filter 🡪 serverdate() y servernow() 🡪 date range default date from / to
* Legcontra\_filter 🡪 0 y 99999999 🡪 default from / to
* Fecdeu 🡪 default 🡪 &today

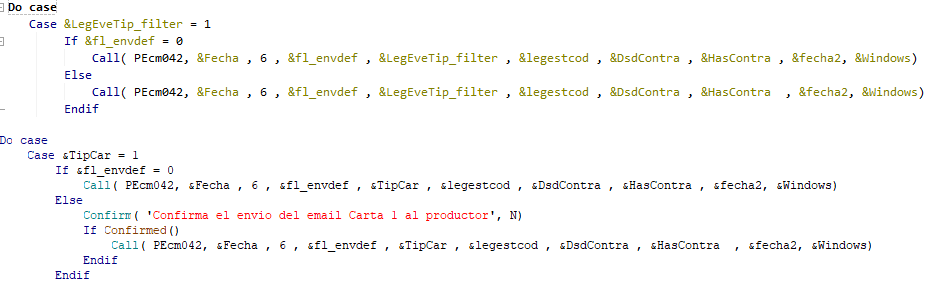




1. Si en el evento de un botón tiene el IF Confirmed() 🡪 confirm 🡪 true (acción del botón)



1. Confirm message expression 🡪 como en nuestro código siempre esta el mensaje confirma el envio del mail… 🡪 ‘¿confirma el envio del mail?’
2. En event end code 🡪 va lo siguiente:



En este caso &tipcar se cambia porque es &legevetip\_filter (ya que es el filtro del combo box y esto nos damos cuenta por el &tipcar = 1 , etc)

1. Los If Confirmed() se borran 🡪 se le deja el call (el Web Designer trabaja distinto al WW)