Ajustes y Diseño de paquetes SLV V2.0

--obtiene el código de la sucursal desde donde corre el paquete.

getvlparametro('cdSucursal', 'General');

* Todos los parámetros de la aplicación pos

select \* from parametrossistema

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \* %v 21/01/2020 - ChM Versión inicial buscar comisionista*

*\**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/*

* *Preguntar a Alejandra si se puede filtrar solo los comisionistas que tengan algún pedido pendiente y no solo por ser miembro de la sucursal.*

PROCEDURE GetComisionistas (p\_cdrol char, r\_cursor OUT CURSOR\_TYPE)

AS

v\_modulo varchar2(100) := 'PKG\_SLV\_Comisionista.GetComisionistas';

BEGIN

OPEN r\_cursor FOR

SELECT DISTINCT e.identidad,

e.dsrazonsocial,

e.cdestadooperativo,

e.cdcuit

FROM entidades e,rolesentidades r

WHERE e.cdestadooperativo = 'A'

*--AND TRIM (e.cdmaincanal) = 'VC'*

*--AND trim(E.CDMAINSUCURSAl)=trim(p\_sucursal)*

AND R.IDENTIDAD=E.IDENTIDAD

and r.CDROL = p\_cdrol

order by e.dsrazonsocial ;

EXCEPTION

WHEN OTHERS THEN

PKG\_slv\_common.LOGWRITE ( 'Modulo: '||v\_modulo, ' Error: ' || SQLERRM);

END GetComisionistas;

SELECT e.IDENTIDAD,

e.CDCUIT,

e.dsrazonsocial CLIENTE,

trunc(do.dtdocumento) FECHA,

trunc(pe.dtentrega) FECHA\_ENTREGA,

pe.SQDIRECCION,

pe.CDTIPODIRECCION,

round(SUM(pe.ammonto), 2) MONTO,

*-- Formato Direccion*

de.dscalle || ' ' || de.dsnumero || ' (' ||

trim(de.cdcodigopostal) || ') ' || lo.dslocalidad || ' - ' ||

pro.dsprovincia Direccion,

pe.ID\_CANAL,

su.dssucursal sucursal,

su.cdsucursal,

op.dsobservacion observacion

FROM pedidos pe,

documentos do,

direccionesentidades de,

localidades lo,

provincias pro,

sucursales su,

entidades e,

observacionespedido op

WHERE pe.iddoctrx = do.iddoctrx

and do.identidadreal = e.identidad

AND do.identidadreal = de.identidad

AND pe.cdtipodireccion = de.cdtipodireccion

AND pe.sqdireccion = de.sqdireccion

and de.cdpais = pro.cdpais

and de.cdprovincia = pro.cdprovincia

and de.cdpais = lo.cdpais

and de.cdprovincia = lo.cdprovincia

AND de.cdlocalidad = lo.cdlocalidad

AND do.cdsucursal = su.cdsucursal

and do.cdcomprobante = 'PEDI'

and do.dtdocumento >= v\_fecha

AND pe.icestadosistema = c\_pedi\_liberado

and pe.id\_canal <> 'CO'

and nvl(pe.iczonafranca, 0) = 0

and pe.idcnpedido is null

and pe.idpedido = op.idpedido(+)

and do.cdsucursal = g\_cdSucursal

GROUP BY e.IDENTIDAD,

e.CDCUIT,

e.dsrazonsocial,

trunc(do.dtdocumento),

trunc(pe.dtentrega),

pe.SQDIRECCION,

pe.CDTIPODIRECCION,

de.dscalle,

de.dsnumero,

trim(de.cdcodigopostal),

lo.dslocalidad,

pro.dsprovincia,

pe.ID\_CANAL,

su.dssucursal,

su.cdsucursal,

op.dsobservacion

ORDER BY trunc(pe.dtentrega),

su.dssucursal,

pro.dsprovincia,

lo.dslocalidad;

*--esto es una técnica para transformar un string separado por comas a tabla*

SELECT TRIM(SUBSTR(txt,INSTR(txt, ',', 1, level ) + 1,

INSTR(txt, ',', 1, level + 1) - INSTR(txt, ',', 1, level)-1)) AS u

FROM(SELECT replace(','||v\_idcomi||',','''','') AS txt FROM dual)

CONNECT BY level <= LENGTH(txt)-LENGTH(REPLACE(txt,',',''))-1;

*/\*fin técnica \*/*