

GET - PriceList

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página para paginación.
@PageNumber	No	INT	Cantidad de registros a obtener por página.
@filterListNumber	No	VARCHAR(20)	Número de la lista de precios para filtrar.

Campos devueltos

Campo	Tipo de dato	Descripción
ID_GVA10	INT	Identificador interno de la lista de precios.
NRO_DE_LIS	VARCHAR(?)	Número de la lista de precios.
NOMBRE_LIS	VARCHAR(?)	Nombre o descripción de la lista de precios.
MON_CTE	BIT / INT?	Indicador de moneda corriente (1 = sí).
INCLUY_IVA	BIT / INT?	Indica si incluye IVA (1 = sí).
INCLUY_IMP	BIT / INT?	Indica si incluye impuestos (1 = sí).
FEC_DESDE	DATETIME	Fecha desde la cual está vigente la lista.
FEC_HASTA	DATETIME	Fecha hasta la cual está vigente la lista.
HABILITADA	BIT / INT?	Indica si la lista está habilitada (1 = sí).

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Id", "PriceListNumber", "Description", "CommonCurrency", "IvaIncluded",  
"InternalTaxIncluded", "ValidityDateSince", "ValidityDateUntil", "Enabled"}
```

Observaciones

- El parámetro @filterListNumber se convierte a un tipo ENTERO_TG con NULLIF (parece un tipo definido en la base).
- La consulta filtra por NRO_DE_LIS si se pasa un valor, o devuelve todas si no.
- Los resultados se ordenan por número de lista y se paginan con OFFSET-FETCH.

GET - Price

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página para paginación.
@PageNumber	No	INT	Cantidad de registros a obtener por página.
@filterListNumber	No	VARCHAR(20)	Número de lista de precios para filtrar.
@SKUCode	No	VARCHAR(15)	Código del artículo o SKU para filtrar (opcional).
@id	No	VARCHAR(20)	Identificador específico del precio en GVA17 para filtrar (opcional).
@datePrice	No	DATETIME	Fecha mínima de modificación del precio para filtrar (opcional).

Campos devueltos

Campo	Tipo de dato	Descripción
ID_GVA17	INT	Identificador único del registro de precio.
RN	INT	Número de fila (row number) dentro de cada partición (lista y artículo).
NRO_DE_LIS	VARCHAR(?)	Número de lista de precios.
COD_ARTICU	VARCHAR(?)	Código del artículo o SKU.
PRECIO	DECIMAL / FLOAT?	Precio del artículo.
FECHA_MODI	DATETIME	Fecha de última modificación del precio.
FEC_DESDE	DATETIME	Fecha desde la cual la lista de precios está habilitada.
FEC_HASTA	DATETIME	Fecha hasta la cual la lista de precios está habilitada.

Los campos devueltos se renombran para devolver un Json de esta manera: {"Id", "PriceListNumber", "SKUCode", "Price", "DatePrice", "ValidityDateSince", "ValidityDateUntil"}

*RN se selecciona por el SP pero no se devuelve, revisar observaciones.

Observaciones

- La consulta filtra precios válidos y vigentes según reglas específicas sobre perfiles y promociones.

- Se usa paginación con OFFSET y FETCH NEXT.
- Solo devuelve el registro más reciente (RN = 1) por combinación de lista y artículo.
- La unión con GVA10 asegura que la lista de precios esté habilitada en el rango de fechas.

GET - PriceByCustomer

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página para paginación.
@PageNumber	No	INT	Cantidad de registros a obtener por página.
@filterCustomerCode	No	VARCHAR(20)	Código del cliente para filtrar los precios.
@SKUCode	No	VARCHAR(15)	Código del artículo o SKU para filtrar (opcional).
@priceListNumber	No	VARCHAR(10)	Número de lista de precios para filtrar (opcional).

Campos devueltos

Campo	Tipo de dato	Descripción
COD_ARTICU	VARCHAR(?)	Código del artículo o SKU.
COD_CLIENT	VARCHAR(?)	Código del cliente.
PRECIO	DECIMAL / FLOAT?	Precio asignado al cliente por artículo.
NRO_LISTA	VARCHAR(?)	Número de la lista de precios.

Los campos devueltos se renombran para devolver un Json de esta manera: {"SKUCode", "CustomerCode", "Price", "PriceListNumber"}

Observaciones

- El procedimiento devuelve precios específicos asignados a un cliente y un artículo dado.
- Se filtra que el perfil del artículo sea distinto de 'N'.
- La lista de precios debe estar habilitada (L.HABILITADA = 1).
- Se asegura que el artículo no sea parte de un perfil 'N' a menos que tenga una promoción diferente de 'P'.
- Utiliza paginación con OFFSET y FETCH NEXT.

GET - Store

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterStoreNumber	No	VARCHAR(20)	Número de sucursal para filtrar. Si se envía vacío, no se filtra.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_SUCURSAL	INT	Identificador único de la sucursal
NRO_SUCURSAL	INT	Número de la sucursal
DESC_SUCURSAL	VARCHAR	Descripción o nombre de la sucursal
CALLE	VARCHAR	Calle de la sucursal
NRO_DOMIC	VARCHAR	Número de domicilio
PISO	VARCHAR	Piso del domicilio
DEPARTAMENTO_DOMIC	VARCHAR	Departamento o unidad
TORRE	VARCHAR	Torre del edificio
BLOQUE	VARCHAR	Bloque del edificio
LOCALIDAD	VARCHAR	Localidad o ciudad
CODIGO_POSTAL	VARCHAR	Código postal
ID_PROVINCIA	INT	Identificador de la provincia
EMAIL	VARCHAR	Email de contacto
WEB_SITE	VARCHAR	Página web
CONTACTO	VARCHAR	Persona de contacto
TELEFONO	VARCHAR	Teléfono principal
TELEFONO2	VARCHAR	Teléfono secundario

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Id","StoreNumber","Description","Street","Number","Floor","Apartment","Tower","Block",  
,"City","PostalCode","ProvinceCode","Email","WebPage","Contact","PhoneNumber1","Ph  
oneNumber2"}
```

Observaciones

- El procedimiento aplica paginación con OFFSET y FETCH NEXT.
- Si no se informa un número de sucursal en @filterStoreNumber, se devuelven todas las sucursales.
- La tabla de origen es **SUCURSAL**.

GET - Currency

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código de moneda a filtrar. Si está vacío, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_MONEDA	INT	Identificador único de la moneda (clave primaria).
COD_MONEDA	VARCHAR	Código de la moneda.
DESC_MONEDA	VARCHAR	Descripción o nombre de la moneda.
SIMBOLO_MONEDA	VARCHAR	Símbolo asociado a la moneda (por ejemplo, \$, €).
TIPO_MONEDA	VARCHAR	Tipo de moneda (puede indicar si es nacional o extranjera, etc.).
COD_AFIP	VARCHAR	Código de la moneda según AFIP.
COD_AFIP_EX	VARCHAR	Código AFIP alternativo o extendido.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Id","CurrencyCode","Description","Symbol","Type","RG1547Code","AFIPCode"}
```

Observaciones:

- La información se obtiene desde la tabla MONEDA.
- Permite filtrar opcionalmente por COD_MONEDA.
- Aplica paginación con OFFSET y FETCH NEXT.

GET - Measure

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código de la medida a filtrar. Si está vacío, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_MEDIDA	INT	Identificador único de la medida.
COD_MEDIDA	VARCHAR	Código de la medida.
DESC_MEDIDA	VARCHAR	Descripción o nombre de la medida.
SIGLA_MEDIDA	VARCHAR	Sigla o abreviatura de la medida.
CANT_DECIMAL_MEDIDA	INT	Cantidad de decimales permitidos para la medida.
TIPO_UNIDAD	VARCHAR	Tipo de unidad (por ejemplo, peso, longitud, volumen, etc.).
COD_MEDIDA_AFIP	VARCHAR	Código de la medida según AFIP (Administración Federal de Ingresos Públicos).
EQUIVALENCIA_AFIP	DECIMAL	Equivalencia de la medida según AFIP.
COD_MEDIDA_TRANSPORTE	VARCHAR	Código de la medida para transporte.
EQUIVALENCIA_TRANSPORTE	DECIMAL	Equivalencia para transporte.

Los campos devueltos se renombran para devolver un Json de esta manera: {"Id", "Code", "Description", "Initials", "Quantity", "Unit Type", "Afip Code", "Afip Equivalence", "Transport Code", "Transport Equivalence"}

Observaciones:

- Los datos provienen de la tabla MEDIDA.

- El filtro por código de medida es opcional.
- Se utiliza paginación con OFFSET y FETCH NEXT.
- El orden es ascendente por código de medida (COD_MEDIDA).

GET - Scale

Parámetros de Entrada

Parámetro	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Índice de página para paginación (offset).
@PageNumber	No	INT	Número de registros a devolver (límite).
@filterCode	No	VARCHAR(20)	Código para filtrar la escala; si es NULL o cadena vacía no filtra.

Campos Devueltos

Campo	Tipo de dato	Descripción
ID_STA32	INT	Identificador único de la escala
COD_STA32	VARCHAR	Código de la escala
DESCRIP_ES	VARCHAR	Descripción en español de la escala
NRO_ESCALA	INT	Número o nivel de la escala

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Id","Code","DescriptionScale","NumberScale"}
```

Descripción

Este procedimiento obtiene registros de la tabla STA32 (escalas), filtrando por un código opcional. Los resultados se ordenan por el campo COD_ESCALA y se soporta paginación mediante OFFSET y FETCH NEXT.

GET - Transport

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código del transportista para filtrar. Si es vacío, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_GVA24	INT	Identificador único del transportista
COD_TRANSP	VARCHAR	Código del transportista
NOMBRE_TRA	VARCHAR	Nombre del transportista
CATEG_TRAN	VARCHAR	Categoría del transportista
CUIT_TRANS	VARCHAR	CUIT del transportista
PORC_RECAR	DECIMAL	Porcentaje de recargo
DOM_TRANS	VARCHAR	Domicilio del transportista
COD_POSTAL	VARCHAR	Código postal
LOCALIDAD	VARCHAR	Localidad o ciudad
COD_PROVIN	VARCHAR	Código de provincia
TELEFONO	VARCHAR	Teléfono de contacto
E_MAIL	VARCHAR	Correo electrónico
WEB	VARCHAR	Página web
OBSERVACIONES	VARCHAR/TEXT	Observaciones adicionales

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Id","Code","Name","IVACategory","Cuit","SurchargePercentage","Address","PostalCode","City","ProvinceCode","PhoneNumbers","Email","WebPage"}
```

*Revisar no devuelve observaciones?

Observaciones

- El procedimiento aplica paginación con OFFSET y FETCH NEXT.

- Si no se informa un código en `@filterCode`, se devuelven todos los transportistas.
- La tabla de origen es **GVA24**.

GET - Warehouse

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código del depósito para filtrar. Si es vacío o NULL, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_STA22	INT	Identificador del depósito
COD_STA22	VARCHAR	Código del depósito
NOMBRE_SUC	VARCHAR	Nombre o descripción del depósito
DIR_SUCURS	VARCHAR	Dirección del depósito
INHABILITA	BIT / INT	Indica si el depósito está inhabilitado (1 = Sí, 0 = No)
CENTRALIZA	BOOL	Indica si el depósito es centralizado

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Id","Code","Description","Address","Disabled","CentralizedStock"}
```

Observaciones

- El procedimiento aplica paginación con OFFSET y FETCH NEXT.
- Si no se informa un código en @filterCode o es NULL o vacío, se devuelven todos los registros.
- La tabla de origen es **STA22**.

GET - ValueScale

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código de escala para filtrar. Si es vacío o NULL, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_STA33	INT	Identificador del valor escala
ID_STA32	INT	Identificador de la escala
COD_ESCALA	VARCHAR	Código de la escala
COD_VALOR	VARCHAR	Código del valor
DESC_VALOR	VARCHAR	Descripción del valor
HABILITADO	BIT / INT	Indica si el valor está habilitado (1 = Sí, 0 = No)

Los campos devueltos se renombran para devolver un Json de esta manera:

{"Id","Code","CodeValue","DescriptionValue"}.

- No muestra ID_STA33 ni HABILITADO

Observaciones

- El procedimiento aplica paginación con OFFSET y FETCH NEXT.
- Si no se informa un código en @filterCode o es NULL o vacío, se devuelven todos los registros.
- La tabla de origen es **STA33**.

GET - Seller

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Índice de página para paginación (offset).
@PageNumber	No	INT	Número de registros a devolver (límite).
@filterCode	No	VARCHAR(20)	Código del vendedor para filtrar; si es NULL o cadena vacía no filtra.

Campos Devueltos

Campo	Tipo de dato	Descripción
COD_VENDED	VARCHAR	Código único del vendedor
NOMBRE_VEN	VARCHAR	Nombre del vendedor
PORC_COMIS	DECIMAL	Porcentaje de comisión del vendedor
INHABILITA	BIT / CHAR	Indicador si el vendedor está inhabilitado (activo/inactivo)

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Code","Name","CommissionPercentage","Disabled"}
```

Descripción

Este procedimiento obtiene la lista de vendedores desde la tabla GVA23, permitiendo filtrar por código de vendedor. Los resultados están ordenados por código de vendedor y soportan paginación mediante los parámetros @PageSize y @PageNumber.

GET - CurrencyExchangeRate

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterIdRenglon	No	VARCHAR(20)	ID del renglón de cotización a filtrar. Si está vacío, no se aplica filtro.
@lastUpdate	No	DATETIME	Fecha mínima de actualización. Si se envía, filtra los datos a partir de esa fecha/hora.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_REGLON_COTIZACION_MONEDA	INT	Identificador del renglón de cotización.
COTIZACION	DECIMAL	Valor de la cotización correspondiente.
FECHA_HORA	DATETIME	Fecha y hora en que se registró la cotización.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"IdRenglon","Value","DateTime"}
```

Observaciones:

- Los datos provienen de la tabla RENGLON_COTIZACION_MONEDA.
- Aplica filtros opcionales por ID_REGLON_COTIZACION_MONEDA y por fecha (@lastUpdate).
- Se utiliza paginación con OFFSET y FETCH NEXT.

GET - ProductsFolderClassifier

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página (inicio) para paginar.
@PageNumber	No	INT	Cantidad de registros a devolver.
@filterIdFolder	No	VARCHAR(20)	ID de la carpeta para filtrar; si es cadena vacía se considera NULL.

Campos Devueltos

Campo	Tipo de dato	Descripción
IDFOLDER	VARCHAR	Identificador de la carpeta
DESCRIP	VARCHAR	Descripción o nombre de la carpeta
IDPARENT	VARCHAR	ID de la carpeta padre (jerarquía)

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"IdFolder","Name","IdParent"}
```

Descripción

Este procedimiento obtiene una lista paginada de carpetas de productos desde la tabla STA11FLD.

- Si el parámetro @filterIdFolder es cadena vacía, se reemplaza por NULL, lo que permite traer todas las carpetas.
- El filtro usa ISNULL(@filterIdFolder, IdFolder), por lo que si el filtro es NULL, no se aplica.
- Ordena los resultados por IDFOLDER ascendente.
- Utiliza paginación con OFFSET-FETCH.

GET - CustomersFolderClassifier

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterIdFolder	No	VARCHAR(20)	ID de la carpeta a filtrar. Si está vacío o NULL, se devuelven todas.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
IDFOLDER	VARCHAR	Identificador de la carpeta (folder).
DESCRIP	VARCHAR	Descripción o nombre de la carpeta.
IDPARENT	VARCHAR	ID de la carpeta padre (estructura jerárquica).

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"IdFolder","Name","IdParent"}
```

Observaciones:

- La información proviene de la tabla GVA14FLD.
- El filtro por IDFOLDER es opcional.
- Se usa paginación con OFFSET y FETCH NEXT.
- Permite representar jerarquías mediante el campo IDPARENT.

GET - SaleCondition

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Índice de página para paginación (offset).
@PageNumber	No	INT	Número de registros a devolver (límite de filas).
@filterCode	No	VARCHAR(20)	Código de condición de venta para filtrar. Si es cadena vacía o NULL, no se filtra.

Campos Devueltos

Campo	Tipo	Descripción
COND_VTA	INT	Código de la condición de venta
DESC_COND	VARCHAR	Descripción de la condición de venta
CONTADO	BIT	Indicador si la condición es contado (1 = Sí, 0 = No)
GENERA_FECHAS_ALTERNATIVAS	BIT	Indicador si genera fechas alternativas (1 = Sí, 0 = No)
APLICA_MORA	BIT	Indicador si aplica mora (1 = Sí, 0 = No)

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"Code","Description","Cash","GenerateAlternativeDate","GenerateDebitLatePayment"}
```

Descripción

Este procedimiento recupera condiciones de venta de la tabla GVA01 aplicando filtros opcionales por código de condición. Además:

- Convierte campos que almacenan 'S' o 'N' en valores BIT (1 o 0).
- Ordena los resultados por código de condición (COND_VTA).
- Implementa paginación usando OFFSET y FETCH NEXT.

GET - StockGroupByProduct

Parámetros de entrada

Campo	Req.	Tipo	Descripción
@PageSize	No	INT	Página a saltar (para paginación)
@PageNumber	No	INT	Número de registros a devolver
@filterProductCode	No	VARCHAR(20)	Código de producto para filtrar (igual o NULL/empty para no filtrar)
@discountPendingOrders	No	BIT	Si se descuentan cantidades pedidas en pedidos recibidos (1 = sí, 0 = no)
@centraliza	No	BIT	Si usar lógica de stock centralizado (1) o stock normal (0)

Lógica interna

1. Cantidad Pedida en pedidos recibidos:

- Si @discountPendingOrders = 1, se crea una tabla temporal @ArticuloCantidadPedida con la suma de cantidades pedidas por artículo de pedidos con estado 'RECIBIDA', filtrando por producto si se indica.

2. Filtrado y agrupación según @centraliza:

- Si @centraliza = 1:
 - Consulta desde tablas CTA_SALDO_ARTICULO_DEPOSITO (stock centralizado).
 - Se filtra y agrupa por código de artículo.
 - Se suman cantidades de stock, pendiente y comprometida.
 - Se suma también la cantidad pedida en pedidos a la cantidad comprometida.
 - Se puede filtrar por depósitos y sucursales si estas tablas temporales tuvieran datos (están declaradas pero no usadas en el código que diste).
 - Se hace paginación con OFFSET y FETCH NEXT.
- Si @centraliza = 0:
 - Consulta desde tablas STA19, STA22, y STA11 (stock normal).
 - Se filtra y agrupa por código de artículo.
 - También suma cantidades de stock, pendiente y comprometida (incluyendo la cantidad pedida).

- iv. Aplica filtros para depósitos y sucursales análogos a los del stock centralizado.
- v. Pagina resultados.

Campos resultantes (ambos casos)

Campo	Tipo	Descripción
COD_ARTICULO	VARCHAR	Código del artículo
CANTIDAD_STOCK	Decimal	Suma total de stock disponible
CANTIDAD_PENDIENTE	Decimal	Suma total de cantidad pendiente
CANTIDAD_COMPROMETIDA	Decimal	Suma total de cantidad comprometida + cantidad pedida

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"SKUCode","Quantity","PendingQuantity","EngagedQuantity"}
```

Manejo de errores

- TRY...CATCH para capturar errores y relanzarlos con RAISERROR.

Notas importantes

- Las tablas temporales @Depositos y @Sucursales se declaran pero no reciben valores en el procedimiento tal como está, por lo que sus filtros solo aplican si se insertan datos manualmente antes de la ejecución o si se modifica el procedimiento.
- El procedimiento hace uso de la tabla temporal para las cantidades pedidas solo si @discountPendingOrders es 1.
- Los filtros sobre producto pueden ser exactos o parciales según el parámetro, aunque aquí está usando solo igualdad.

GET - Stock

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Índice para paginación (número de páginas a saltar)
@PageNumber	No	INT	Cantidad de registros a devolver
@filterProductCode	No	VARCHAR(20)	Código del producto para filtrar (puede usarse parcial o igual según @UseEqual)
@storeNumber	No	VARCHAR(10)	Número de sucursal para filtrar (igual o LIKE según @UseEqual)
@warehouseCode	No	VARCHAR(10)	Código del depósito para filtrar (igual o LIKE según @UseEqual)
@discountPendingOrders	No	BIT	Indica si se descuentan cantidades pendientes en pedidos (1 = sí, 0 = no)
@id	No	VARCHAR(20)	ID específico para filtrar (convierte a BIGINT para comparación)
@UseEqual	No	BIT (default=0)	Indica si la búsqueda de filtros es exacta (=) o por LIKE (0 = LIKE, 1 = exacta)
@centraliza	No	BIT (default=0)	Indica qué lógica de stock usar: 1 para stock centralizado, 0 para stock normal

Funcionamiento general

- Si @discountPendingOrders = 1, primero calcula la cantidad pedida por artículo en pedidos recibidos y guarda ese dato en una tabla temporal.
- Luego, según @centraliza, se ejecuta una consulta u otra:
 1. **Si centraliza = 1:**

Se usa la tabla CTA_SALDO_ARTICULO_DEPOSITO (stock centralizado) y se filtra por los parámetros recibidos.

Se suma la cantidad comprometida con la cantidad pedida de la tabla temporal para calcular la cantidad comprometida final.
 2. **Si centraliza = 0:**

Se usa la tabla STA19 (stock normal) y se filtra por los parámetros recibidos.

También suma la cantidad comprometida con la cantidad pedida para calcular la cantidad comprometida final.

- En ambos casos se realiza paginación y ordenamiento por código de artículo.

Campos devueltos

Campo	Req.	Tipo	Descripción
CodigoArticulo	No	VARCHAR	Código del artículo
CodigoDeposito	No	VARCHAR	Código del depósito (almacén o sucursal)
NumeroSucursal	No	INT	Número de sucursal (0 si no aplica)
CantidadStock	No	DECIMAL	Cantidad disponible en stock
CantidadStock2	No	DECIMAL	Segunda unidad de medida del stock
CantidadPendiente	No	DECIMAL	Cantidad pendiente (no disponible aún)
CantidadPendiente2	No	DECIMAL	Segunda unidad de medida para cantidad pendiente
CantidadComprometida	No	DECIMAL	Cantidad comprometida + cantidad pedida en pedidos
CantidadComprometida2	No	DECIMAL	Segunda unidad para cantidad comprometida

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"SKUCode","WarehouseCode","StoreNumber","Quantity","SecondQuantity","PendingQuantity","SecondPendingQuantity","EngagedQuantity","SecondEngagedQuantity"}
```

Manejo de errores

- Usa TRY...CATCH para capturar y relanzar cualquier error ocurrido durante la ejecución.

GET - Counterfoil

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código del talonario a filtrar. Si está vacío, no se aplica filtro.
@voucher	No	VARCHAR(3)	Código del comprobante a filtrar. Si está vacío, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
TALONARIO	INT	Código del talonario.
DESCRIP	VARCHAR	Descripción del talonario.
TIPO	VARCHAR	Tipo del talonario o comprobante.
COMPROB	VARCHAR	Código del comprobante asociado.
FECHA_VTO	DATETIME	Fecha de vencimiento del talonario/comprobante.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"CounterfoilCode","Description","CounterfoilType","Voucher","CounterfoilExpiration"}
```

Observaciones:

- La información se obtiene desde la tabla GVA43.
- Aplica filtros opcionales por TALONARIO y COMPROB.
- Usa paginación con OFFSET y FETCH NEXT.

GET - ProductsRelation

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página (inicio) para paginar.
@PageNumber	No	INT	Cantidad de registros a devolver.
@filterIdFolder	No	VARCHAR(20)	ID de la carpeta para filtrar productos; cadena vacía se trata como NULL (sin filtro).
@skuCode	No	VARCHAR(15)	Código SKU del producto; cadena vacía se trata como NULL.
@relationName	No	VARCHAR(40)	Nombre de la relación para filtrar; cadena vacía se trata como NULL.

Campos Devueltos

Campo	Tipo de dato	Descripción
CODE	VARCHAR	Código del producto
IDFOLDER	VARCHAR	ID de la carpeta a la que pertenece el producto
DESCRIP	VARCHAR	Descripción del tipo de relación
SHOW	BOOL	Campo indicador, probable flag para mostrar

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"SkuCode","IdFolder","RelationName","ShowRelation"}
```

Descripción

Este procedimiento devuelve una lista paginada de relaciones entre productos y carpetas desde las tablas STA11RIF y STA11REL:

- Filtra por IDFOLDER, CODE y DESCRIP según los parámetros.
- Si los parámetros vienen como cadena vacía, se consideran NULL y no aplican filtro para ese campo.
- Ordena por IDFOLDER ascendente.
- Aplica paginación con OFFSET-FETCH.

GET - ProductsFolder

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Página de inicio para la paginación.
@PageNumber	No	INT	Cantidad de registros a devolver por página.
@filterIdFolder	No	VARCHAR(20)	ID de la carpeta para filtrar.
@skuCode	No	VARCHAR(15)	Código SKU del producto para filtrar.

Campos devueltos

Campo	Tipo de dato	Descripción
CODE	VARCHAR	Código del producto SKU
IDFOLDER	VARCHAR	ID de la carpeta asociada

Los campos devueltos se renombran para devolver un Json de esta manera: {"SkuCode", "IdFolder"}

Descripción

Este procedimiento devuelve una lista paginada de productos (CODE) asociados a una carpeta específica (IDFOLDER), permitiendo filtrar tanto por la carpeta como por el código SKU del producto.

Los filtros permiten usar valores vacíos (' '), en cuyo caso no filtra por ese campo.

El resultado está ordenado por código de producto (CODE) ascendentemente.

GET - CustomersFolder

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterId	No	VARCHAR(20)	ID de la carpeta a filtrar. Si está vacío, no se aplica filtro.
@customerCode	No	VARCHAR(6)	Código del cliente a filtrar. Si está vacío, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
CODE	VARCHAR	Código del cliente.
IDFOLDER	VARCHAR	Identificador de la carpeta asociada.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"CustomerCode", "IdFolder"}
```

Observaciones:

- La información se obtiene desde la tabla GVA14ITC.
- Permite aplicar filtros opcionales por ID de carpeta y código de cliente.
- Se utiliza paginación con OFFSET y FETCH NEXT.
- El orden está dado por ID_GVA14ITC ASC (aunque no se devuelve ese campo en el resultado).

GET - CustomersRelation

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterId	No	VARCHAR(20)	ID de la carpeta (folder) a filtrar. Si está vacío, no se aplica filtro.
@customerCode	No	VARCHAR(6)	Código del cliente a filtrar. Si es NULL o vacío, no se aplica filtro.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
CODE	VARCHAR	Código del cliente asociado.
IDFOLDER	VARCHAR	Identificador de la carpeta (folder) relacionada.
DESCRIP	VARCHAR	Descripción de la relación (desde tabla GVA14REL).
SHOW	BIT / INT	Indicador para mostrar la relación (booleano o entero).

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"CustomerCode","IdFolder","RelationName","ShowRelation"}
```

Observaciones:

- Los datos se obtienen de la tabla GVA14RIF con join a GVA14REL para detalles de la relación.
- Se permiten filtros opcionales por carpeta y código de cliente.
- Implementa paginación con OFFSET y FETCH NEXT.
- Ordena resultados por IDFOLDER ascendente.

GET - Invoices

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página para paginación.
@PageNumber	No	INT	Cantidad de registros a obtener por página.
@orderID	No	VARCHAR(220)	Identificador de la orden en la tienda externa (filtro opcional).
@orderNumber	No	VARCHAR(220)	Número de orden en la tienda externa (filtro opcional).
@fromDate	No	DATETIME	Fecha mínima para filtrar órdenes (inclusive).
@toDate	No	DATETIME	Fecha máxima para filtrar órdenes (exclusiva, hasta el día anterior al siguiente día).

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_NEXO_PEDIDOS_ORDEN	VARCHAR	Identificador único de la orden.
ORDER_NRO_TIENDA	VARCHAR	Número de la orden en la tienda externa.
FECHA_ORDEN	DATETIME	Fecha en que se realizó la orden.
T_COMP	VARCHAR	Tipo de comprobante relacionado.
N_COMP	INT	Número de comprobante relacionado.
ARCHIVO_BINARIO	VARCHAR	Archivo binario asociado a la factura (no nulo ni vacío).
ARCHIVO_BINARIO_EXPIRATION	VARCHAR	¿?????

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"OrderID","OrderNumber","Date","InvoiceType","InvoiceNumber","InvoiceFileUrl","InvoiceFileExpiration"}
```

*No coincide DTO <-> Resultado / Campos devueltos

Observaciones:

- Filtra órdenes por ID, número, rango de fechas y sólo aquellas que tienen un archivo binario asociado.
- Ordena resultados por fecha de orden ascendente.
- Paginación aplicada con OFFSET y FETCH NEXT.

GET - DiscountByCustomer

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCustomerCode	No	VARCHAR(20)	Código del cliente a filtrar. Puede ser comparación exacta o parcial según @UseEqual.
@UseEqual	No	BIT	Indica tipo de búsqueda: 1 = comparación exacta, 0 = búsqueda con LIKE (parcial).
@SKUCode	No	VARCHAR(15)	Código del artículo (SKU) a filtrar. También aplica según @UseEqual para búsqueda exacta o LIKE.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
COD_ARTICU	VARCHAR	Código del artículo (SKU).
COD_CLIENT	VARCHAR	Código del cliente.
BONIF	NUMÉRICO	Bonificación o descuento asignado.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"SKUCode","CustomerCode","Discount"}
```

Observaciones:

- La búsqueda permite filtrar cliente y artículo con comparación exacta o parcial (LIKE) según @UseEqual.
- Se usa una CTE TablaIdsEncontrados para aplicar paginación en la consulta de clientes/artículos.
- El resultado devuelve descuentos asociados a los clientes y artículos encontrados.

- El orden es por cliente y artículo ascendente.

GET - Publications

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Página o desplazamiento para la paginación (offset).
@PageNumber	No	INT	Cantidad de registros a devolver (límite de filas).
@productCode	No	VARCHAR(50)	Código del artículo en la tienda. Si es cadena vacía, no se filtra por este campo.
@skuCode	No	VARCHAR(20)	Código SKU. Si es cadena vacía, no se filtra por este campo.
@variantCode	No	VARCHAR(50)	Código de variante o escala del artículo. Si es cadena vacía, no se filtra por este campo.

Campos Devueltos

Campo	Tipo de dato	Descripción
CODIGO_ARTICULO_TIENDA	VARCHAR	Código del artículo en la tienda
DESCRIPCION_ARTICULO_TIENDA	VARCHAR	Descripción del artículo en la tienda
CODIGO_ARTICULO_ESCALA_TIENDA	VARCHAR	Código de variante o escala del artículo
DESCRIPCION_ARTICULO_ESCALA_TIENDA	VARCHAR	Descripción de la variante o escala
SKU	VARCHAR	Código SKU del artículo

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"ProductCode","Description","VariantCode","VariantDescription","SkuCode"}
```

Descripción

El procedimiento recupera una lista paginada de publicaciones de artículos en tienda basándose en los filtros opcionales:

- Filtra por código del artículo, código SKU y código de variante si se proporcionan (no vacíos).
- Ordena por el identificador ID_NEXO_RELACION_ARTICULO_TIENDA de forma ascendente.
- Aplica paginación usando OFFSET y FETCH NEXT.

GET - Product

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página para paginación.
@PageNumber	No	INT	Cantidad de registros a devolver por página.
@filterProductCode	No	VARCHAR(20)	Código del producto para filtrar.
@synonym	No	VARCHAR(15)	Sinónimo del producto para filtrar.
@barCode	No	VARCHAR(40)	Código de barras para filtrar.
@generic	No	VARCHAR(MAX)	Filtro adicional para aplicar condiciones personalizadas (cadena SQL dinámica).
@onlyKit	No	BIT (0 o 1)	Si es 1, filtra solo productos tipo kit (PROMO_MENU = 'P').
@onlyEnabled	No	BIT (0 o 1)	Si es 1, filtra solo productos habilitados (PERFIL distinto de 'N').
@updatedAtDate	No	DATETIME	Filtra productos modificados desde esta fecha en adelante.

Campos devueltos

Campo	Tipo de dato	Descripción
ID_STA11	INT	Identificador interno del producto.
COD_ARTICU	VARCHAR	Código del artículo/producto.
DESCRIPCIO	VARCHAR	Descripción principal del producto.
DESC_ADIC	VARCHAR	Descripción adicional del producto.
SINONIMO	VARCHAR	Sinónimo o código alternativo.
COD_BARRA	VARCHAR	Código de barras.
COMISION_V	FLOAT/DECIMAL	Comisión de venta.
DESCUENTO	FLOAT/DECIMAL	Descuento aplicado.
COD_MEDIDA	VARCHAR	Código de la unidad de medida.
COD_MEDIDA_AFIP	VARCHAR	Código de medida según AFIP (para reportes fiscales).
EQUIVALENCIA_STOCK_2	NUMÉRICO	Equivalencia para stock en segunda unidad.
COD_MEDIDA_VENTAS	VARCHAR	Código de unidad de medida para ventas.

EQUIVALE_V	NUMÉRICO	Equivalencia para unidad de venta.
STOCK_MAXI	INT	Stock máximo permitido.
STOCK_MINI	INT	Stock mínimo permitido.
PTO_PEDIDO	INT	Punto de pedido para reposición.
OBSERVACIONES	VARCHAR	Observaciones adicionales.
EsKit	BIT	1 = es kit (PROMO_MENU = 'P'), 0 = no.
PROMODESDE	DATETIME	Fecha de inicio de promoción.
PROMOHASTA	DATETIME	Fecha de fin de promoción.
FECHA_ULTIMA_MODIFICACION	DATETIME	Fecha de la última modificación del producto.
USA_ESC	?	(Campo no documentado, posible flag o categoría)
ESCALA_1	VARCHAR	Código escala 1 (relacionado con medidas o categorías).
ESCALA_2	VARCHAR	Código escala 2.
BASE	?	(Campo no documentado, posible base para cálculo)
VALOR1	?	(Campo no documentado)
VALOR2	?	(Campo no documentado)
DESCRIP_ES	VARCHAR	Descripción en español relacionada con ESCALA_1 (desde tabla STA32).
DESCRIP_ES2	VARCHAR	Descripción en español relacionada con ESCALA_2 (desde tabla STA32).
DESC_VALOR	VARCHAR	Descripción valor 1 (desde STA33, relacionada con ESCALA_1 y VALOR1).
DESC_VALOR2	VARCHAR	Descripción valor 2 (desde STA33, relacionada con ESCALA_2 y VALOR2).
Deshabilitado	BIT	1 = producto deshabilitado (PERFIL = 'N'), 0 = habilitado.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{
  "Id",
  "SKUCode",
  "Description",
  "AdditionalDescription",
  "AlternativeCode",
  "BarCode",
  "Commission",
  "Discount",
  "MeasureUnitCode",
  "SecondMeasureUnitCode",
  "StockEquivalence",
  "StockControlUnit",
  "SalesMeasureUnitCode",
  "SalesEquivalence",
  "MaximumStock",
  "MinimumStock",
  "RestockPoint",
  "Observations",
  "Kit",
  "KitValidityDateSince",
  "KitValidityDate"
}
```

Until","LastUpdateUtc","UseScale","Scale1","Scale2","BaseArticle","ScaleValue1","Scale Value2","DescriptionScale1","DescriptionScale2","DescriptionValueScale1","Description ValueScale2","Disabled"}

Comentarios adicionales

- El filtro dinámico en @generic se agrega directamente a la cláusula WHERE, permitiendo filtros complejos.
- El campo EsKit determina si el producto es un kit (PROMO_MENU = 'P').
- El procedimiento usa sp_executesql para ejecutar la consulta dinámica con parámetros para evitar SQL Injection.
- Los productos pueden filtrarse por código, sinónimo, código de barras, y fechas de actualización.
- El flag @onlyEnabled filtra productos habilitados (no 'N' en PERFIL).
- Campos USA_ESC, BASE, VALOR1, VALOR2 no están documentados en el procedimiento, probablemente relacionados a escalas o parámetros internos.

GET - Customer

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página desde donde se comienza a paginar.
@PageNumber	No	INT	Cantidad de registros que se desea obtener (paginación).
@filterCode	No	VARCHAR(20)	Código del cliente a filtrar. Si está vacío, no se aplica filtro.
@ivaCategoyCode	No	VARCHAR(3)	Código de categoría de IVA. Si está vacío o NULL, no se aplica filtro.
@documentType	No	VARCHAR(2)	Tipo de documento (por ejemplo, DNI, CUIT, etc.). Si está vacío o NULL, no se aplica filtro.
@documentNumber	No	VARCHAR(20)	Número de documento o CUIT del cliente. Si está vacío o NULL, no se aplica filtro.
@updateDate	No	DATETIME	Fecha mínima de modificación (FECHA_MODI). Devuelve clientes modificados desde esa fecha.

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
COD_CLIENT	VARCHAR	Código del cliente.
RAZON_SOCI	VARCHAR	Razón social del cliente.
NOM_COM	VARCHAR	Nombre comercial del cliente.
DOMICILIO	VARCHAR	Domicilio principal.
C_POSTAL	VARCHAR	Código postal.
LOCALIDAD	VARCHAR	Localidad del cliente.
COD_PROVIN	VARCHAR	Código de la provincia.
DIR_COM	VARCHAR	Dirección comercial.
TELEFONO_1	VARCHAR	Teléfono principal.
E_MAIL	VARCHAR	Correo electrónico.
TELEFONO_MOVIL	VARCHAR	Teléfono móvil.
WEB	VARCHAR	Sitio web.

COD_CATEGORIA_IVA	VARCHAR	Código de la categoría de IVA (desde tabla CATEGORIA_IVA).
TIPO_DOC	VARCHAR	Tipo de documento.
CUIT	VARCHAR	CUIT del cliente.
NRO_LISTA	INT	Número de lista de precios asociada.
PORC_DESC	DECIMAL	Porcentaje de descuento.
OBSERVACIONES	VARCHAR / TEXT	Observaciones internas.
FECHA_INHA	DATETIME	Fecha de inhabilitación.
FECHA_ALTA	DATETIME	Fecha de alta del cliente.
FECHA_MODI	DATETIME	Fecha de última modificación.
COD_VENDED	VARCHAR	Código del vendedor asignado.
CUPO_CREDI	DECIMAL	Cupo de crédito asignado.
SALDO_CC	DECIMAL	Saldo en cuenta corriente.
SALDO_CC_U	DECIMAL	Saldo en cuenta corriente en moneda extranjera.
CLAUSULA	VARCHAR	Cláusula de venta.
MON_CTE	BIT	Indicador de cuenta corriente (1 = cliente con cuenta corriente).
COND_VTA	VARCHAR	Condición de venta.
COD_TRANSP	VARCHAR	Código del transportista.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{
  "Code": "BusinessName",
  "TradeName": "Address",
  "PostalCode": "City",
  "ProvinceCode": "TradeAddress",
  "PhoneNumbers": "Email",
  "MobilePhoneNumber": "WebPage",
  "IvaCategoryCode": "DocumentType",
  "DocumentNumber": "PriceListNumber",
  "Discount": "Observations",
  "DisabledDate": "UpdateDatetime",
  "LastUpdateUtc": "ShippingAddresses",
  "CustomerComments": "SellerCode",
  "CreditQuota": "LocalAccountBalance",
  "ForeignAccountBalance": "ForeignCurrencyClause",
  "CreditQuotaCurrencyCode": "SaleConditionCode",
  "TransportCode"
}
```

Observaciones:

- Los datos provienen de la tabla GVA14 y se une con CATEGORIA_IVA para obtener el código de IVA.
- Se puede filtrar por múltiples campos opcionales, incluidos CUIT, tipo de documento, categoría IVA, etc.
- Se incluye una lógica para devolver registros modificados desde una fecha (@updateDate).

- La consulta implementa paginación con OFFSET y FETCH NEXT.

GET - OrderStatus

Parámetros de Entrada

Campo	Req.	Tipo de dato	Descripción
@PageSize	No	INT	Número de página para paginación.
@PageNumber	No	INT	Cantidad de registros por página a obtener.
@orderID	No	VARCHAR(MAX)	ID(s) de orden(es) a filtrar. Puede ser un JSON array o un valor único.
@fromDate	No	DATETIME	Fecha mínima para filtrar las órdenes (inclusive).
@toDate	No	DATETIME	Fecha máxima para filtrar las órdenes (exclusiva, filtra hasta el día anterior al siguiente día).
@state	No	VARCHAR(600)	Estados de órdenes separados por coma para filtrar.
@includeInvoices	No	BIT	Indica si incluir información de facturas relacionadas (1 = sí, 0 = no).

Resultado / Campos devueltos

Campo	Tipo de dato	Descripción
ID_NEXO_PEDIDOS_ORDEN	INT	Identificador único de la orden.
FECHA_ORDEN	DATETIME	Fecha en que se realizó la orden.
ESTADO_ORDEN	VARCHAR	Estado actual de la orden.
T_COMP	VARCHAR	Tipo de comprobante relacionado (solo si @includeInvoices = 1).
N_COMP	VARCHAR	Número de comprobante relacionado (solo si @includeInvoices = 1).
ARCHIVO_BINARIO	VARCHAR	Archivo binario asociado a la factura (solo si @includeInvoices = 1).
ARCHIVO_BINARIO_EXPIRATION	VARCHAR	Identificador de la orden en la tienda externa.

Los campos devueltos se renombran para devolver un Json de esta manera:

```
{"OrderID","Date","Status","InvoiceType","InvoiceNumber","InvoiceFileUrl","InvoiceFileExpiration"}
```

- Por defecto, se devuelve {"OrderID","Date","Status"} y en caso de “incluir invoices”, se devuelven también {"InvoiceType","InvoiceNumber","InvoiceFileUrl","InvoiceFileExpiration"}.

Observaciones:

- Filtrado por IDs de órdenes, fechas y estados (con posibilidad de múltiples estados separados por coma).
- Cuando `@includeInvoices = 1` se devuelve información completa con datos de facturas asociadas.
- Ordena resultados por fecha de orden ascendente.
- Paginación aplicada mediante OFFSET y FETCH NEXT.

POST– Order

Este es un ejemplo de Body para realizar un Post Order y subir una única Orden.

Parámetros de Entrada - Body

```
{
  "Date": "2025-08-21T00:00:00",
  "Total": 8523.0,
  "TotalDiscount": 77.0,
  "PaidTotal": 8523.0,
  "FinancialSurcharge": 200.0,
  "WarehouseCode": "2",
  "SellerCode": "2",
  "TransportCode": "01",
  "SaleConditionCode": 1,
  "InvoiceCounterfoil": 1,
  "OrderCounterfoil": 0,
  "PriceListNumber": 0,
  "AgreedWithSeller": false,
  "IvalIncluded": false,
  "InternalTaxIncluded": false,
  "OrderID": "10003",
  "OrderNumber": "7",
  "ValidateTotalWithPaidTotal": true,
  "ValidateTotalWithItems": true,
  "Comment": null,
  "Customer": {
    "CustomerID": 1,
    "DocumentType": "80",
    "DocumentNumber": "20-11111111-2",
    "IVACategoryCode": "CF",
    "User": "ADMIN",
    "Email": "api@axoft.com",
    "FirstName": "Juan",
    "LastName": "Perez",
    "BusinessName": "Empresa",
    "Street": "Cerrito",
    "HouseNumber": "1186",
    "Floor": "2",
    "Apartment": "1",
    "City": "CABA",
    "ProvinceCode": "1",
```

```
"PostalCode": "1122",
"PhoneNumber1": "12459856",
"PhoneNumber2": "42563698",
"Bonus": 0.0,
"MobilePhoneNumber": "165952141",
"WebPage": null,
"BusinessAddress": "Cerrito 1186",
"Comments": "Comentario",
"NumberListPrice": 0,
"Removed": false,
"DateUpdate": "0001-01-01T00:00:00",
"Disable": "0001-01-01T00:00:00"
},
"CancelOrder": false,
"OrderItems": [
  {
    "ProductCode": "203",
    "SKUCode": "0100200659",
    "VariantCode": null,
    "Description": "LAVARROPAS AUTOM. MOD.BLUE ",
    "VariantDescription": null,
    "Quantity": 1.0,
    "UnitPrice": 7700.0,
    "DiscountPercentage": 0.0
  },
  {
    "ProductCode": "104",
    "SKUCode": "0100100269",
    "VariantCode": null,
    "Description": "CÁMARA DIGITAL 4X MARCA TCL",
    "VariantDescription": null,
    "Quantity": 1.0,
    "UnitPrice": 300.0,
    "DiscountPercentage": 0.0
  }
],
"Shipping": {
  "ShippingID": "71906",
  "ShippingCode": "abc",
  "Street": "9 de Julio",
  "HouseNumber": "1186",
  "Floor": "1",
  "Apartment": "1",
  "City": "CABA",
```

```
"ProvinceCode": "2",
"PostalCode": "1122",
"PhoneNumber1": "125165151",
"PhoneNumber2": "12345678",
"ShippingCost": 400.0,
"DeliversMonday": "S",
"DeliversTuesday": "S",
"DeliversWednesday": "S",
"DeliversThursday": "S",
"DeliversFriday": "S",
"DeliversSaturday": "S",
"DeliversSunday": "S",
"DeliveryHours": "8",
"DeliveryDate": null
},
"CashPayments": null,
"Payments": [
  {
    "PaymentId": 385663,
    "TransactionDate": "2025-12-16T00:00:00",
    "AuthorizationCode": "52",
    "TransactionNumber": "998595",
    "Installments": 1,
    "InstallmentAmount": 8100.0,
    "Total": 8100.0,
    "CardCode": "01",
    "CardPlanCode": "1",
    "VoucherNo": 48,
    "CardPromotionCode": "2"
  },
  {
    "PaymentId": 385614,
    "TransactionDate": "2025-12-16T00:00:00",
    "AuthorizationCode": "53",
    "TransactionNumber": "5849849",
    "Installments": 2,
    "InstallmentAmount": 250.0,
    "Total": 300.0,
    "CardCode": "DI",
    "CardPlanCode": "2",
    "VoucherNo": 49,
    "CardPromotionCode": "1"
  }
],
```

```
"CancelReason": null,  
"CancelDate": null  
}
```

POST – Order/Batch

Este es un ejemplo de body para realizar un Post Order/Batch y subir varias órdenes. Se debe tomar el ejemplo de una orden individual para el body en el ejemplo documentado para PostOrder (Individual).

- Un Order/Batch es una única solicitud para la carga de varias ordenes de manera simultánea. Por lo tanto, podemos escribirlo como un arreglo de ordenes.
- Orders = [O₁, O₂, O₃, ... , O_n] donde “O” representa a una orden y “n” es el número de esa orden.
- El siguiente ejemplo muestra la manera correcta de escribir el body de Order/Batch, donde cada O_n debe ser reemplazado por un body completo para una orden (explicado en el ejemplo de PostOrder).

Parámetros de Entrada

```
{  
  "Orders": [ {O1} , {O2} ]  
}
```