# Documentación Superstore

## Ejercicio 1: Obtener los gastos estimados con los datos

Ya que son gastos a los que se les hará una operación fila contra fila existe una iteración dentro de la tabla virtual que luego se agrupará y es necesario utilizar una función de agrupación. El caso es de manera que requiere la función SUMX desde DAX.

## Ejercicio 2: Obtener las ganancias totales

Las ganancias se pueden calcular con la columna ya existente de la tabla del modelo, ya que no requiere operación fila a fila se puede usar una función básica de sumatoria para la columna y obtener el resultado deseado desde una medida en DAX.

## Ejercicio 3: Calcular las unidades vendidas totales

Las unidades se pueden calcular con la columna ya existente de la tabla del modelo, ya que no requiere operación fila a fila se puede usar una función básica de sumatoria para la columna y obtener el resultado deseado desde una medida en DAX.

## Ejercicio 4: Calcular el monto de venta total

El monto se puede calcular con la columna ya existente de la tabla del modelo, ya que no requiere operación fila a fila se puede usar una función básica de sumatoria para la columna y obtener el resultado deseado desde una medida en DAX.

## Ejercicio 5: Obtener el Total de clientes diferentes que compraron

Los clientes se pueden calcular con la columna ya existente de la tabla fctsales del modelo, ya que no requiere operación fila a fila se puede usar una función básica que cuente distintos para la columna y obtener el resultado deseado desde una medida en DAX. Aunque no es la mejor forma de hacerlo ya que no se le podrían aplicar nuevos filtros si se requieren en el futuro.

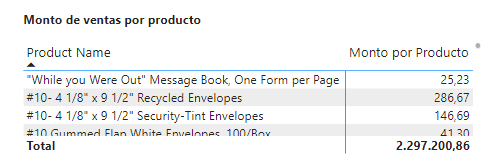
## Ejercicio 6: Obtener el total de productos diferentes vendidos

Los productos diferentes tampoco requieren una iteración por lo tanto se puede usar la función básica de distintos de DAX para obtener el resultado usando la columna de la tabla del modelo. Aunque tampoco será la opción más optima a futuro cuándo se quieran aplicar filtros.

## E**jercicio 7: Obtener el monto de venta por Producto**

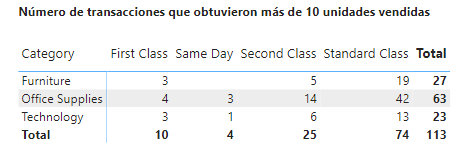
El monto de venta por producto se puede calcular usando la fórmula simple “” en una matriz, pero es más optimo utilizar una función que haga una transición de contexto y aplique los elementos seleccionados como filtro de la tabla y haga una iteración fila a fila. Este es el comportamiento de *CALCULATE* que obtendrá cada producto como filtro y hará su cálculo respectivo sumando todas las ventas obtenidas para ese producto y luego los presentará en la matriz final, hará esa iteración hasta que estén todos los productos con sus respectivas ventas.

El resultado puede representarse en una matriz con las ventas por cada producto:



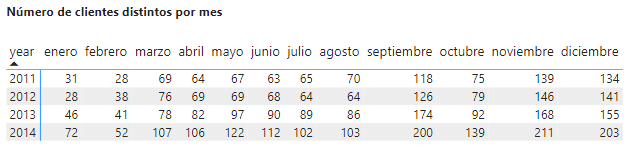
## Ejercicio 8: Obtener las categorías que obtuvieron más de 10 unidades vendidas

Esta medida tiene contexto de filtro y contexto de fila a la vez. Porque si colocamos una matriz con las categorías y el método de envió, ambos (toda la fila y columna de la tabla) se pueden convertir en contexto de filtro y realizar una iteración fila a fila, es decir, a través de un contexto de fila. Esto se puede lograr combinando las funciones FILTER para activar el contexto de fila y COUNTROWS para el contexto de filtro y que devuelven el resultado esperado.



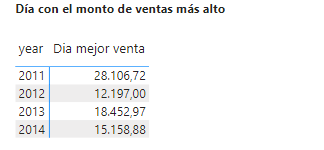
## Ejercicio 9: Obtener el número de clientes por mes

Esta medida se puede calcular a través de una operación fila a fila usando una función de agregación para realizar el calculo por mes. Podemos activar una transición de contexto y usar una medida ya creada. Podemos combinarla con VALUES porque esta función genera contexto de filtro en las medidas de acuerdo a los parámetros que se le coloquen. De manera que a VALUES le podemos pasar la columna de deseada para que use como filtro los valores distintos de esa columna y pasar la medida existente clientes diferentes totales para que calcule el número total del filtro aplicado con VALUES y hacer la agregación por cada valor distinto con SUMX. Luego de esos cálculos virtuales podemos presentar en una matriz los cálculos esperados.



## Ejercicio 10: Obtener el día con el monto de venta más alto (o menos alto)

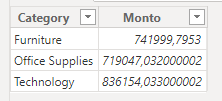
Para este caso podemos replicar la formula anterior, a excepción de que se utilizará MAXX o MINX para obtener el resultado esperado. Este tipo de formulas pueden ser replicadas en otros casos fácilmente y se le llama *reducción de granularidad*.



## Ejercicio 11: Mostrar una tabla con las categorías diferentes

El ejercicio se puede realizar usando una tabla calculada obteniendo los valores y agrupándolos a través de la función SUMMARIZE que trae toda la tabla de manera virtual solo para el cálculo (**concepto llamado tabla expandida**) y luego con la función ADDCOLUMNS agregar una columna basada en una medida existente, en este caso el monto de ventas totales. De esa forma tendremos una tabla final con las columnas deseadas.

El resultado final es una nueva tabla solo con las columnas categoría y el monto correspondiente a la venta:



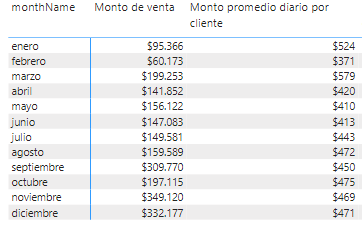
## Ejercicio 12: Obtener la última fecha de orden

Podemos hacer uso de la tabla expandida con SUMMARIZE debido para obtener solo las fechas donde hubo una orden. Luego a esa tabla virtual expandida se le aplica una agregación con MAXX para que muestre la última fecha de los resultados de la iteración.

## Ejercicio 13: Obtener el monto de venta diario promedio por cliente

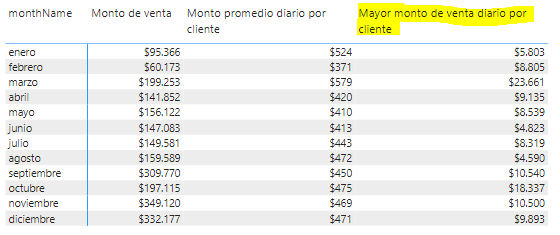
El contexto de filtro será el mes o los meses correspondientes que se mostraran de acuerdo al objeto de matriz que se quiera. Se aplicará el filtro de mes y será necesario que una función SUMMARIZE reciba una tabla virtual expandida y genere una agrupación de acuerdo al mes (columna brindada). Luego la expresión se evaluará en contexto de fila, haciendo una transición de contexto, y la fila de la tabla virtual se convertirá en el contexto de filtro nuevo, para al final llenar la tabla virtual. Para obtener el valor promedio AVERAGEX recibirá la tabla virtual creada con SUMMARIZE y se evaluará fila a fila para obtener el promedio de esos datos y mostrar el resultado deseado.

El resultado será un promedio diario por mes:



## Ejercicio 14: Obtener el monto de venta máximo por cliente

Para esto podemos replicar el ejercicio 13 y sustituir la función de agregación AVERAGEX por la función MAXX para iterar.



## Ejercicio 15: Obtener el mejor cliente diario

Se puede realizar utilizando variables y SUMMARIZE para generar una tabla virtual que obtenga los clientes y con FILTER y la medida anterior del ejercicio 14 obtener el resultado.

El resultado es: Sean Miller.

## Ejercicio 16: Crear una tabla con las categorías y el monto de ventas

El uso de SUMMARIZECOLUMNS es especialmente útil para nuevas tablas ya que se pueden crear expresiones dentro de la declaración sin ningún riesgo de que la tabla se haga expandida y agregue columnas indeseadas. El único inconveniente es que no se puede aplicar en medidas, únicamente en nuevas tablas calculadas.

## Ejercicio 17: Crear una tabla con los distintos de categoría y subcategoría

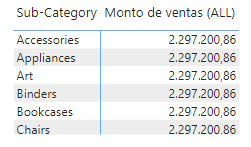
Se puede utilizar la función ALL para traer todos los valores distintos, ya que esta función ignora los filtros aplicados a la tabla si los hay.

## Ejercicio 18: Crear una tabla solo con la columna subcategoría y producto\_id

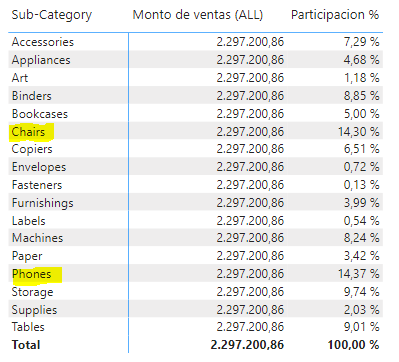
La tabla se puede crear indicando las columnas que queremos ignorar, como categoría y el nombre del producto y dejar solo las necesarias. Eso se puede lograr con ALLEXCEPT y obtener solo las dos columnas deseadas.

## Ejercicio 19: Mostrar las participaciones porcentuales por subcategoría

Para lograr esto debemos tener un total para dividir cada categoría, de modo que necesitamos el total de las ventas brutas. Como es un total fijo debemos usar ALLSELECTED para que se deshaga de todos los **filtros internos** aplicados y tome solo los **filtros externos** (filtros informes), de modo que sume la columna con filtros externos para obtener el total de ventas, dando como resultado una suma igual para todas las subcategorías.

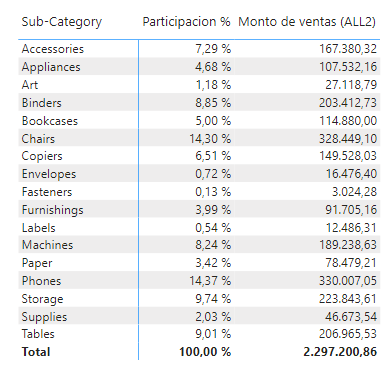


Ahora se procede a dividir la venta individual de cada categoría tomando también los filtros de informe con el total de ventas calculado. Esto se puede ejecutar con la función DIVIDE y obtener los porcentajes deseados. Se le puede colocar cualquier filtro externo más y funcionará.



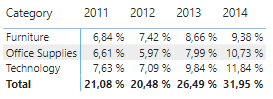
## Ejercicio 20: Calcular el monto de venta por subcategoría

Podemos hacerlo arreglando el ejercicio anterior y usando CALCULATE y ALLSELECTED para hacer una transición de contexto y aplicar solo filtros externos.

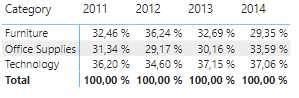


## Ejercicio 21: Hacer que la columna participación sume 100%

Este es requerido en situaciones donde PBI no hace la suma correctamente, por ejemplo, cuándos hay tres variables comparadas, filas, columnas y valores. Para ello podemos utilizar las funciones ALL y variables. Además, podemos usar CALCULATE y pasar ALL como modificador para que se evalué primero y luego pasar una tabla creada con VALUES. De esa manera podemos modificar el resultado, ya que como se observa las categorías por año no suman 100%.



Resultado final:



## Ejercicio 22: Obtener el número de registros por cliente

Podemos obtener agregando una columna calculada directamente a la tabla existente de dimensión cliente. Podemos usar CALCULATE para hacer una transición de contexto y usar ALLEXCEPT para obtener solo la columna deseada.

# Documentación SCLP

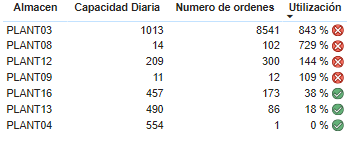
## Ejercicio 1: Porcentaje de capacidad máxima usado por almacén

Para obtener la capacidad porcentual debemos dividir las ordenes por planta por la capacidad total de la planta. Para ello podemos utilizar la función DIVIDE, pero para ello debemos crear dos medidas que compongan numerador y denominador. La primera medida es el número de ordenes por planta, esto se puede calcular con la función COUNT usando el contexto de filtro el tipo de almacén:

Luego es necesario calcular la capacidad total por planta, para ello usamos la función SELECTEDVALUE que devuelve el valor único de la fila (contexto de fila), es decir, la capacidad por cada planta.

Por último, ingresamos las dos medidas en la función DIVIDE para obtener el resultado de la utilización total por planta. El contexto de filtro será el almacén de la visual.

El Resultado final será un nivel porcentual de la utilización por planta que se puede diseñar a gusto o necesidad del proyecto.



## Ejercicio 2: Costo total por almacén

Para obtener el costo por almacén es necesario multiplicar el número de ordenes asignadas al almacen por el costo unitario del almacén, en otras palabras: , para ello podemos hacer uso de la función SUMX.