

# Hoja de medidas Dax

## Tabla “info\_completa\_items”

### “ChechMostrarGrafico”

```
CheckMostrarGrafico =  
IF(  
    ISFILTERED(info_completa_items[name]), 1, 0  
)
```

Devuelve 1 si [name] está filtrando, 0 si no. Usado para múltiples trucos en el dashboard, principalmente para que ciertos elementos gráficos, que no tienen sentido si no son filtrados, no muestren nada.

### “Msg”

```
Msg =  
IF(ISFILTERED(info_completa_items[name]), "", "Selecciona un Item")
```

Es utilizada para un truco visual. Cuando hay un item seleccionado no muestra nada, cuando no hay nada seleccionado muestra el mensaje, dando la sensación de que está dentro de una tarjeta. Usado en la tarjeta falsa “Icono” en “Análisis Mercado - Detalles”.

### “Ranking Global”

```
Ranking Global =  
RANKX(ALL(info_completa_items[name]), CALCULATE(AVERAGE(lista_subasta[Precio])),  
, DESC, Dense)
```

Rankea cada [name] en función de su precio medio. Dense para que si hay empate rankee igual ambos valores. Usada como referencia en otras medidas.

### “Filtered RGlobal”

```
Filtered RGlobal =  
VAR ranking = [Ranking Global]  
VAR items = CALCULATE(DISTINCTCOUNT(info_completa_items[name]),  
ALL(info_completa_items[name]))  
RETURN IF(ISFILTERED(info_completa_items[name]), ranking & " / " & items,  
"Selecciona un Item")
```

Basada en “Ranking Global”. A mayores cuenta la cantidad única total de [name]. Finalmente se formatea para que, si hay un item filtrado, muestre “ranking / items total”.

Si no hay un item filtrado muestra “Selecciona un Item”. Utilizado en las tarjetas de la página “Análisis Mercado - Detalles”.

### “Ranking Global (Hoy)”

```
Ranking Global (Hoy) =  
RANKX(ALL(info_completa_items[name]),CALCULATE(AVERAGE(lista_subasta[Precio]),  
'Calendar'[Date] = MAX('Calendar'[Date])), ,DESC,Dense)
```

Aplica a “Ranking Global” el contexto del último día de precios registrados. Usada como referencia en otras medidas.

### “Filtered RGlobal (Hoy)”

```
Filtered RGlobal (Hoy) =  
VAR ranking = [Ranking Global (Hoy)]  
VAR items = CALCULATE(DISTINCTCOUNT(info_completa_items[name]),  
ALL(info_completa_items[name]))  
RETURN IF(ISFILTERED(info_completa_items[name]), ranking & " / " & items,  
"Selecciona un Item")
```

Similar a “Filtered Global”. Utilizada también para una tarjeta en “Análisis Mercado - Detalles”.

## Tabla “lista\_subasta”

### “PrecioMedioHoy”

```
PrecioMedioHoy =  
CALCULATE(AVERAGE(lista_subasta[Precio]), lista_subasta[date] =  
MAX(lista_subasta[date]))
```

Calcula el precio medio en el contexto temporal del último día disponible. Utilizado como columna en la tabla de estadística descriptiva, pero también como referencia en otras medidas.

### “PrecioMedioAyer”

```
PrecioMedioAyer =  
VAR fechaHoy = MAX(lista_subasta[date])  
VAR fechaAyer = fechaHoy - 1  
RETURN CALCULATE(AVERAGE(lista_subasta[Precio]), lista_subasta[date] = fechaAyer)
```

Calcula el precio medio en el contexto temporal del penúltimo día disponible. Utilizado como referencia en otras medidas.

### “Check Precios”

```
Check Precios =  
-- Devuelve 1 si el precio de hoy es mayor que el de ayer, 0 si es menor, o 2 si  
sigue igual  
VAR hoy = [PrecioMedioHoy]  
VAR ayer = [PrecioMedioAyer]  
RETURN  
IF( ISBLANK(hoy) || ISBLANK(ayer), BLANK(),  
    IF(hoy = ayer, 2,  
        IF(hoy < ayer, 0, 1)  
    )  
)
```

Explicado en el comentario de la propia medida. Utilizado para formato condicional en la tabla de estadística descriptiva. Usando iconos, muestra si el precio medio de hoy es mayor que el de ayer, menor o igual.

### “Check Venta”

```
Check Venta =  
-- Devuelve 1 si el precio medio de hoy es superior a la media + stdeev, 0 si es  
menor, 2 si no cumple ninguna de las otras  
VAR desviacion = STDEV.P(lista_subasta[Precio])  
VAR mediaHoy = [PrecioMedioHoy]  
VAR media = AVERAGE(lista_subasta[Precio])  
VAR capSuperior = media + desviacion  
VAR capInferior = media - desviacion  
  
RETURN IF(ISBLANK(media) || media = 0, BLANK(),  
    IF(mediaHoy > capSuperior, 1,  
        IF(mediaHoy < capInferior, 0, 2)  
    )  
)
```

Basándose en el precio medio general y la desviación estándar, esta medida sirve para clasificar el precio medio más reciente en múltiples categorías. Utilizado para formato condicional en la tabla de estadística descriptiva. Usando iconos, muestra recomendaciones de si un ítem debe ser comprado ( $\text{mediaHoy} < \text{capInferior}$ ), vendido ( $\text{mediaHoy} > \text{capSuperior}$ ), o atesorado (resto de escenarios).

### “PrecioAvgIsPercent”

```
PrecioAvgIsPercent =  
VAR avgPrecio = AVERAGE(lista_subasta[Precio])  
VAR maxPrecio = MAX(lista_subasta[Precio])  
VAR minPrecio = MIN(lista_subasta[Precio])  
  
RETURN (avgPrecio - minPrecio) / (maxPrecio - minPrecio) * 100
```

Tomando como referencia el precio mínimo y el máximo de un ítem, calcula, en un porcentaje de 0 (precio mínimo) a 100 (precio máximo), donde se encuentra el precio medio. Utilizado para formato condicional en la tabla de estadística descriptiva. Aplica un color de fondo, degradado de rojo a verde, en función del número obtenido.

### “Precio Current Is Percent”

```
PrecioCurrentIsPercent =  
VAR avgPrecio = [PrecioMedioHoy]  
VAR maxPrecio = MAX(lista_subasta[Precio])  
VAR minPrecio = MIN(lista_subasta[Precio])  
  
RETURN (avgPrecio - minPrecio) / (maxPrecio - minPrecio) * 100
```

Similar a la medida anterior, pero calculada sobre el precio medio del día más reciente (la medida “PrecioMedioHoy” nombrada anteriormente). Utilizado para formato condicional en la tabla de estadística descriptiva. Aplica un color de fondo, degradado de rojo a verde, en función del número obtenido.

### “stdev%”

```
stdev% =  
-- Calcula el % que representa la stdv con respecto al precio medio  
VAR desviacion = ROUND(STDEV.P(lista_subasta[Precio]), 2)  
VAR media = AVERAGE(lista_subasta[Precio])  
VAR resultado = ROUND((desviacion * 100)/media, 2)  
RETURN IF(ISBLANK(media) || media = 0, BLANK(), resultado)
```

Explicado en el comentario de la medida. Utilizado para formato condicional en la tabla de estadística descriptiva. Aplica un color de fondo, degradado de verde a rojo, en función del número obtenido.

## “stdev%Tabla”

```
stdev%Tabla =  
-- Calcula el % que representa la stdv con respecto al precio medio  
VAR desviacion = ROUND(STDEV.P(lista_subasta[Precio]), 2)  
VAR media = AVERAGE(lista_subasta[Precio])  
VAR resultado = ROUND((desviacion * 100)/media, 2)  
RETURN IF(ISBLANK(media) || media = 0, BLANK(), desviacion & " | " & resultado &  
"%")
```

Similar a la anterior medida. Formatea el resultado para mostrar la desviación estándar junto al porcentaje que representa con respecto al precio medio. Utilizado como campo en la tabla de estadística descriptiva.

## “TooltipCantidadOfertada”

```
TooltipCantidadOfertada =  
VAR cantidad = FORMAT(SUM(lista_subasta[Precio]), "#,##0.00")  
VAR texto = "Cantidad Ofertada: "  
RETURN CONCATENATE(texto, cantidad)
```

Utilizado en los tooltips relativos a las cantidades ofertadas. Muestra el total de items ofertados, formateado para que cuadre con el estilo del resto de cifras.

## “TooltipPromedioTotal”

```
TooltipPromedioTotal =  
VAR promedio = ROUND(AVERAGE(lista_subasta[Precio]), 2)  
VAR texto = "Promedio General: "  
RETURN CONCATENATE(texto, promedio)
```

Utilizado en los tooltips relativos al promedio de los precios. Muestra el promedio total formateado.

## Tabla “precio\_ficha”

### “Precio Ficha Hoy”

```
Precio Ficha Hoy =  
CALCULATE(SUM(precio_ficha[precio_ficha]), precio_ficha[fecha] =  
MAX(precio_ficha[fecha]))
```

Calcula el precio del WoW Token en el contexto del último día disponible. Utilizado como tarjeta en la página “Transacciones & KPIs - Resultados & KPIs”.

## “Precio Ficha Medio”

```
Precio Ficha Medio =  
  
VAR PromedioFecha = CALCULATE(AVERAGE(precio_ficha[precio_ficha]),  
ALLSELECTED('Calendar'[Date]))  
VAR PromedioTotal = CALCULATE(AVERAGE(precio_ficha[precio_ficha]),  
ALL(precio_ficha))  
RETURN IF( ISBLANK(PromedioFecha), PromedioTotal, PromedioFecha)
```

Calcula el promedio del precio de la ficha en distintos contextos: “PromedioFecha” calcula el promedio en el rango de fecha seleccionado, “PromedioTotal” lo calcula en general (sin filtros). Si “PromedioFecha” no arroja resultados (porque no hay datos durante el período seleccionado), utiliza “PromedioTotal”. Utilizado como valor en múltiples elementos gráficos de la página “Transacciones & KPIs - Resultados & KPIs” y como referencia en otras medidas.

## Tabla “Transacciones”

### “Average Revenue per Day”

```
Average Revenue per Day =  
SUM(Transacciones[Oro]) / DISTINCTCOUNT('Calendar'[Date])
```

Calcula el beneficio promedio por día. Utilizado como valor para la tarjeta doble en “Transacciones & KPIs - Detalles”.

### “Average Revenue per Trading Day”

```
Average Revenue per Trading Day =  
SUM(Transacciones[Oro]) / DISTINCTCOUNT(Transacciones[Fecha])
```

Similar a la medida anterior, pero con la sutil diferencia que solo cuenta los días en los que ha habido transacciones (en vez de todos). Utilizado como valor para la tarjeta doble en “Transacciones & KPIs - Detalles”.

### “Promedio Mercado if Exists”

```
Promedio Mercado if Exists =  
IF(CALCULATE(COUNTROWS(Transacciones)) > 0, AVERAGE(lista_subasta[Precio]))
```

Calcula el precio medio de un item si existe en la tabla de transacciones. Necesario para que no cree líneas vacías en la tabla donde se usa. Utilizado como columna en la tabla “Transacciones & KPIs - Detalles” y como referencia en otras medidas.

### “Check Avg Venta Vs Avg Mercado”

```
Check Avg Venta Vs Avg Mercado =  
VAR avgVenta = AVERAGE(Transacciones[Precio Unitario])  
VAR avgMercado = [Promedio Mercado if Exists]  
RETURN IF(avgVenta = avgMercado, 2,  
    IF(avgVenta < avgMercado, 0, 1)  
)
```

Contrasta el precio promedio de venta con el precio medio general de cada ítem. Utilizado como formato condicional para colores de fondo. Devuelve verde si el precio promedio de venta es mayor que el de mercado (1), amarillo si es igual (2), rojo si es inferior (0).

### “Dinero Real Ganado”

```
Dinero Real Ganado =  
VAR totalOro = SUM(Transacciones[Oro])  
VAR valorFicha = 13  
VAR fichaMedia = precio_ficha[Precio Ficha Medio]  
  
RETURN (totalOro * valorFicha)/fichaMedia
```

Es una sencilla regla de 3. Conociendo el valor del WoW Token, en oro y €, se puede saber a cuánto dinero real equivale la cantidad de oro generada hasta la fecha. Utilizado como valor para una tarjeta doble en “Transacciones & KPIs - Resultados & KPIs”.

### “Fichas conseguidas”

```
Fichas conseguidas =  
SUM(Transacciones[Oro]) / precio_ficha[Precio Ficha Medio]
```

Divide el oro total generado entre el precio medio de la ficha. Devuelve la cantidad de fichas obtenidas. Utilizado como valor para una tarjeta doble en “Transacciones & KPIs - Resultados & KPIs”.

### “KPI Brutosaurio”

```
KPI Brutosaurio =  
(precio_ficha[Precio Ficha Medio] * 80) / 13
```

Es otra regla de 3, esta vez para pasar el precio en € que cuesta el “Brutosaurio Dorado de Comerciante” a oro. Utilizado como valor de destino en un KPI de “Transacciones & KPIs - Resultados & KPIs”.

## “KPI Dinero”

```
KPI Dinero =  
VAR meses = DISTINCTCOUNT('Calendar'[MesAño])  
VAR precioFicha = precio_ficha[Precio Ficha Medio]  
var cantidadGenerada = SUM(Transacciones[Oro])  
  
RETURN cantidadGenerada - (meses * precioFicha)
```

Calcula la cantidad de oro sobrante después de haber pagado las fichas equivalentes a los meses transcurridos. Se utiliza como valor para el KPI del “Brutosaurio Dorado de Comerciante”, un KPI de “Transacciones & KPIs - Resultados & KPIs”.

## “Oro Mensual”

```
Oro Mensual =  
VAR mesActual = MONTH(MAX('Calendar'[Date]))  
  
RETURN CALCULATE(SUM(Transacciones[Oro]), MONTH(Transacciones[Fecha]) = mesActual)
```

Calcula la cantidad de oro generado en el contexto del último mes. Se utiliza como valor para el KPI del “WoW Token”, un KPI de “Transacciones & KPIs - Resultados & KPIs”

## “Precio Brutosaurio”

```
Precio Brutosaurio =  
80
```

Esta medida es únicamente para mostrar (en formato €) el precio de la montura “Brutosaurio Dorado de Comerciante” en una tarjeta de la página “Transacciones & KPIs - Resultados & KPIs”.