


Inteligencia de tiempo – Funciones para desfases de periodos

1. SAMEPERIOLASTYEAR

= **SAMEPERIODLASTYEAR**(Fecha)

Devuelve una Tabla con una sola columna

En análisis de datos hacer comparaciones entre periodos de tiempo es pan de cada día, por ejemplo, cuando nos demandan comparar los ingresos del año actual con los ingresos del año pasado.



| Año | Ingresos Totales |
|--------------|---------------------|
| 1999 | 8.685,83 |
| 2000 | 180.680,24 |
| 2001 | 166.860,36 |
| 2002 | 184.876,83 |
| 2003 | 177.061,95 |
| 2004 | 168.059,37 |
| 2005 | 187.906,44 |
| 2006 | 198.797,54 |
| 2007 | 170.712,79 |
| 2008 | 175.936,37 |
| 2009 | 198.222,01 |
| 2010 | 240.873,82 |
| 2011 | 216.088,04 |
| 2012 | 229.644,13 |
| 2013 | 232.463,65 |
| 2014 | 256.692,19 |
| 2015 | 327.323,99 |
| Total | 3.320.885,54 |

Figura 19. 25 – Matriz Ejemplo de PY Ingresos



| Año | Ingresos Totales | PY Ingresos |
|--------------|---------------------|---------------------|
| 1999 | 8.685,83 | |
| 2000 | 180.680,24 | 8.685,83 |
| 2001 | 166.860,36 | 180.680,24 |
| 2002 | 184.876,83 | 166.860,36 |
| 2003 | 177.061,95 | 184.876,83 |
| 2004 | 168.059,37 | 177.061,95 |
| 2005 | 187.906,44 | 168.059,37 |
| 2006 | 198.797,54 | 187.906,44 |
| 2007 | 170.712,79 | 198.797,54 |
| 2008 | 175.936,37 | 170.712,79 |
| 2009 | 198.222,01 | 175.936,37 |
| 2010 | 240.873,82 | 198.222,01 |
| 2011 | 216.088,04 | 240.873,82 |
| 2012 | 229.644,13 | 216.088,04 |
| 2013 | 232.463,65 | 229.644,13 |
| 2014 | 256.692,19 | 232.463,65 |
| 2015 | 327.323,99 | 256.692,19 |
| Total | 3.320.885,54 | 3.320.885,54 |

PY Ingresos

= **CALCULATE**([Ingresos], **SAMEPERIODLASTYEAR**(Calendario[Fechas]))

*** Nosotros utilizamos las siglas PY por Previous Year (año anterior)

Desafortunadamente en el lenguaje DAX no contamos con las funciones:

- SAMEPERIODLASTQUARTER
- SAMEPERIODLASTMONTH

2. DATEADD

Devuelve una Tabla

La buena noticia es que contamos con una función que nos brindará la posibilidad de hacer los desfases no solo para trimestre y meses, sino también para días e incluso movernos adelante o atrás un número de intervalos deseados, esta función es DATEADD.

Función tabular.

La función DATEADD retorna una tabla que contiene una columna de fechas desfasada adelante o atrás en el tiempo en un número específico de intervalos según el contexto de filtro actual.

= **DATEADD** (<Fechas>;<Número de Intervalos>; <Intervalos>)

Fechas: Una columna que contiene fechas, normalmente, la columna de fechas de la tabla de calendario.

Número de Intervalos: Un numero entero que representa el desfase en el tiempo, los valores negativos significan retroceder en el tiempo y los positivos por su parte hacia adelante en el tiempo.

Intervalo: Este argumento indica en que escala de tiempo se desea realizar el desfase: años, meses, trimestres o días.

Así emulamos la medida *PY Ingresos* con DATEADD:

PY Ingresos

= **CALCULATE**([Ingresos],**DATEADD**(Calendario[Fechas],-1,YEAR))

Así creamos la medida de los mismos ingresos del trimestre pasado o *PQ Ingresos*:

PQ Ingresos

= **CALCULATE**([Ingresos],**DATEADD**(Calendario[Fechas],-1,QUARTER))

Así creamos la medida de los mismos ingresos del mes pasado o *PM Ingresos*:

PM Ingresos

= **CALCULATE**([Ingresos],**DATEADD**(Calendario[Fechas],-1,MONTH))

| Año | Ingresos Totales | PY Ingresos | PQ Ingresos | PM Ingresos |
|------------|-------------------|-------------------|-------------------|-------------------|
| 2001 | 166.860,36 | 180.680,24 | 172.189,31 | 166.796,08 |
| Tr. 1 | 42.316,61 | 49.429,52 | 43.557,78 | 40.541,27 |
| Enero | 14.558,15 | 13.508,96 | 14.678,72 | 14.016,11 |
| Febrero | 11.967,01 | 17.592,04 | 14.862,94 | 14.558,15 |
| Marzo | 15.791,46 | 18.328,51 | 14.016,11 | 11.967,01 |
| Tr. 2 | 44.759,01 | 45.700,64 | 42.316,61 | 43.450,04 |
| Abril | 14.528,67 | 14.726,52 | 14.558,15 | 15.791,46 |
| Mayo | 13.129,91 | 13.507,17 | 11.967,01 | 14.528,67 |
| Junio | 17.100,43 | 17.466,96 | 15.791,46 | 13.129,91 |
| Tr. 3 | 41.555,91 | 41.992,30 | 44.759,01 | 44.667,36 |
| Julio | 14.712,75 | 15.729,11 | 14.528,67 | 17.100,43 |
| Agosto | 12.854,18 | 15.056,15 | 13.129,91 | 14.712,75 |
| Septiembre | 13.988,98 | 11.207,04 | 17.100,43 | 12.854,18 |

Figura 19. 27 – Medidas PY, PQ y PM Ingresos con DATEADD

3. PARALLELPERIOD

Otra situación frecuente es retornar el total del año pasado, especialmente útil para comparar con el acumulado hasta la fecha (YTD), esto para hacer un seguimiento si se ha cumplido el objetivo en contraste con el año anterior o no.

La función time Intelligence para hacer referencia al total de un periodo pasado o futuro si llega ser necesario es: **PARALLELPERIOD**.

= **PARALLELPERIOD** (<Fechas>;<Número de Intervalos>; <Intervalos>)

- **Fechas:** Una columna que contiene fechas, normalmente, la columna de fechas de la tabla de *Calendario*.
- **Número de Intervalos:** Un numero entero que representa el desfase en el tiempo, los valores negativos significan retroceder en el tiempo y los positivos por su parte hacia adelante en el tiempo.
- **Intervalo:** Este argumento indica en que escala de tiempo se desea realizar el desfase: años, meses, trimestres o días.

La creación de la medida *PY Total Ingresos* se muestra a continuación:

PY Ingresos

= **CALCULATE**([Ingresos],**PARALLELPERIOD**(Calendario[Fechas], -1, YEAR))



Nota La diferencia entre DATEADD y PARALLELPERIOD se encuentra en el tercer argumento, el intervalo, pues mientras que DATEADD devuelve el periodo desfasado, PARALLELPERIOD retorna la lista desde el principio del año pasado hasta el final del año pasado.

| Año | Ingresos Totales | PY Total Ingresos |
|------------|------------------|-------------------|
| 2000 | 180.680,24 | 8.685,83 |
| Enero | 13.508,96 | 8.685,83 |
| Febrero | 17.592,04 | 8.685,83 |
| Marzo | 18.328,51 | 8.685,83 |
| Abril | 14.726,52 | 8.685,83 |
| Mayo | 13.507,17 | 8.685,83 |
| Junio | 17.466,96 | 8.685,83 |
| Julio | 15.729,11 | 8.685,83 |
| Agosto | 15.056,15 | 8.685,83 |
| Septiembre | 11.207,04 | 8.685,83 |
| Octubre | 14.678,72 | 8.685,83 |
| Noviembre | 14.862,94 | 8.685,83 |
| Diciembre | 14.016,11 | 8.685,83 |
| 2001 | 166.860,36 | 180.680,24 |
| Enero | 14.558,15 | 180.680,24 |
| Febrero | 11.967,01 | 180.680,24 |
| Marzo | 15.791,46 | 180.680,24 |
| Abril | 14.578,67 | 180.680,24 |
| Total | 3.320.885,54 | 3.320.885,54 |

Véase que los valores se repiten a nivel de meses puesto que siempre se está haciendo referencia al total del año pasado.

En este caso no existe diferencia entre DATEADD y PARALLELPERIOD

1 Cantidad Mes Anterior = CALCULATE([Cantidad], DATEADD(Dates[Fechas], -1, MONTH))

1 Cantidad Mes Anterior 2 = CALCULATE([Cantidad], PREVIOUSMONTH(Dates[Fechas]))

| Año | Cantidad | Cantidad Año Anterior | Cantidad Anterior 2 | Cantidad Mes Anterior | Cantidad Mes Anterior 2 | Cantidad Mes Anterior 3 | Producción Acumulada en el Año | Cantidad Total |
|------------|----------|-----------------------|---------------------|-----------------------|-------------------------|-------------------------|--------------------------------|----------------|
| 2020 | 65.216 | | | 60.303 | | 60.303 | 65.216 | 65.216 |
| Enero | 5.196 | | | | | | 5.196 | 65.216 |
| Febrero | 5.157 | | | 5.196 | 5.196 | 5.196 | 10.353 | 65.216 |
| Marzo | 5.323 | | | 5.157 | 5.157 | 5.157 | 15.676 | 65.216 |
| Abril | 5.883 | | | 5.323 | 5.323 | 5.323 | 21.559 | 65.216 |
| Mayo | 5.099 | | | 5.883 | 5.883 | 5.883 | 26.658 | 65.216 |
| Junio | 5.703 | | | 5.099 | 5.099 | 5.099 | 32.361 | 65.216 |
| Julio | 5.579 | | | 5.703 | 5.703 | 5.703 | 37.940 | 65.216 |
| Agosto | 5.603 | | | 5.579 | 5.579 | 5.579 | 43.543 | 65.216 |
| Septiembre | 5.857 | | | 5.603 | 5.603 | 5.603 | 49.400 | 65.216 |
| Octubre | 5.668 | | | 5.857 | 5.857 | 5.857 | 55.068 | 65.216 |
| Noviembre | 5.235 | | | 5.668 | 5.668 | 5.668 | 60.303 | 65.216 |
| Diciembre | 4.913 | | | 5.235 | 5.235 | 5.235 | 65.216 | 65.216 |
| 2021 | 64.565 | 65.216 | 65.216 | 64.132 | 4.913 | 64.132 | 64.565 | 64.565 |
| Enero | 6.307 | 5.196 | 5.196 | 4.913 | 4.913 | 4.913 | 6.307 | 64.565 |
| Febrero | 4.179 | 5.157 | 5.157 | 6.307 | 6.307 | 6.307 | 10.486 | 64.565 |
| Marzo | 5.378 | 5.323 | 5.323 | 4.179 | 4.179 | 4.179 | 15.864 | 64.565 |
| Abril | 5.117 | 5.883 | 5.883 | 5.378 | 5.378 | 5.378 | 20.981 | 64.565 |
| Mayo | 4.964 | 5.099 | 5.099 | 5.117 | 5.117 | 5.117 | 25.945 | 64.565 |
| Junio | 5.528 | 5.703 | 5.703 | 4.964 | 4.964 | 4.964 | 31.473 | 64.565 |
| Julio | 5.525 | 5.579 | 5.579 | 5.528 | 5.528 | 5.528 | 36.998 | 64.565 |
| Agosto | 5.694 | 5.603 | 5.603 | 5.525 | 5.525 | 5.525 | 42.692 | 64.565 |
| Septiembre | 5.051 | 5.857 | 5.857 | 5.694 | 5.694 | 5.694 | 47.743 | 64.565 |
| Octubre | 5.777 | 5.668 | 5.668 | 5.051 | 5.051 | 5.051 | 53.520 | 64.565 |
| Total | 450.561 | 386.996 | 386.996 | 444.723 | | 444.723 | 63.565 | 450.561 |

1 Cantidad Año Anterior = CALCULATE([Cantidad], SAMEPERIODLASTYEAR(Dates[Fechas]))

1 Cantidad Año Anterior 2 = CALCULATE([Cantidad], DATEADD(Dates[Fechas], -1, YEAR))

1 Cantidad Año Anterior 3 = CALCULATE([Cantidad], PARALLELPERIOD(Dates[Fechas], -1, YEAR))

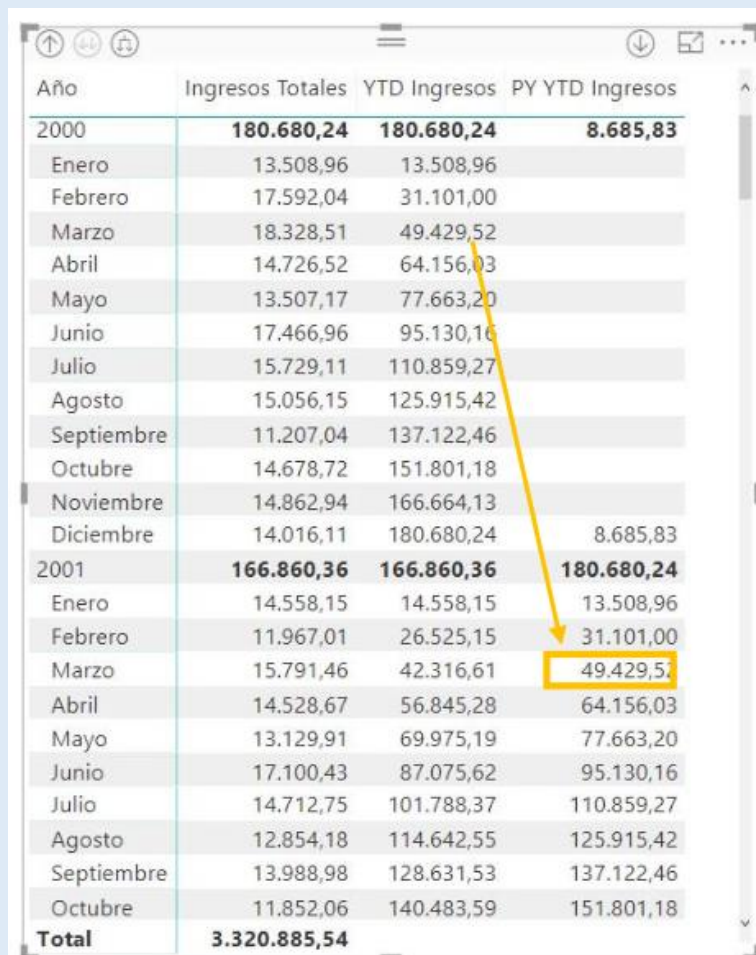
| Año | Cantidad | Cantidad Año Anterior | Cantidad Año Anterior 2 | Cantidad Año Anterior 3 | Cantidad Mes Anterior | Cantidad Mes Anterior 2 | Cantidad Mes Anterior 3 | Producción Acumulada en el Año | Cantidad Total |
|--------------|----------------|--------------------------|----------------------------|----------------------------|--------------------------|----------------------------|----------------------------|-----------------------------------|-------------------|
| 2020 | 65.216 | | | | 60.303 | | 60.303 | 65.216 | 65.216 |
| Enero | 5.196 | | | | | | | 5.196 | 65.216 |
| Febrero | 5.157 | | | | 5.196 | 5.196 | 5.196 | 10.353 | 65.216 |
| Marzo | 5.323 | | | | 5.157 | 5.157 | 5.157 | 15.676 | 65.216 |
| Abril | 5.883 | | | | 5.323 | 5.323 | 5.323 | 21.559 | 65.216 |
| Mayo | 5.099 | | | | 5.883 | 5.883 | 5.883 | 26.658 | 65.216 |
| Junio | 5.703 | | | | 5.099 | 5.099 | 5.099 | 32.361 | 65.216 |
| Julio | 5.579 | | | | 5.703 | 5.703 | 5.703 | 37.940 | 65.216 |
| Agosto | 5.603 | | | | 5.579 | 5.579 | 5.579 | 43.543 | 65.216 |
| Septiembre | 5.857 | | | | 5.603 | 5.603 | 5.603 | 49.400 | 65.216 |
| Octubre | 5.668 | | | | 5.857 | 5.857 | 5.857 | 55.068 | 65.216 |
| Noviembre | 5.235 | | | | 5.668 | 5.668 | 5.668 | 60.303 | 65.216 |
| Diciembre | 4.913 | | | | 5.235 | 5.235 | 5.235 | 65.216 | 65.216 |
| 2021 | 64.565 | 65.216 | 65.216 | 65.216 | 64.132 | 4.913 | 64.132 | 64.565 | 64.565 |
| Enero | 6.307 | 5.196 | 5.196 | 65.216 | 4.913 | 4.913 | 4.913 | 6.307 | 64.565 |
| Febrero | 4.179 | 5.157 | 5.157 | 65.216 | 6.307 | 6.307 | 6.307 | 10.486 | 64.565 |
| Marzo | 5.378 | 5.323 | 5.323 | 65.216 | 4.179 | 4.179 | 4.179 | 15.864 | 64.565 |
| Abril | 5.117 | 5.883 | 5.883 | 65.216 | 5.378 | 5.378 | 5.378 | 20.981 | 64.565 |
| Mayo | 4.964 | 5.099 | 5.099 | 65.216 | 5.117 | 5.117 | 5.117 | 25.945 | 64.565 |
| Junio | 5.528 | 5.703 | 5.703 | 65.216 | 4.964 | 4.964 | 4.964 | 31.473 | 64.565 |
| Julio | 5.525 | 5.579 | 5.579 | 65.216 | 5.528 | 5.528 | 5.528 | 36.998 | 64.565 |
| Agosto | 5.694 | 5.603 | 5.603 | 65.216 | 5.525 | 5.525 | 5.525 | 42.692 | 64.565 |
| Septiembre | 5.051 | 5.857 | 5.857 | 65.216 | 5.694 | 5.694 | 5.694 | 47.743 | 64.565 |
| Octubre | 5.777 | 5.668 | 5.668 | 65.216 | 5.051 | 5.051 | 5.051 | 53.520 | 64.565 |
| Total | 450.561 | 386.996 | 386.996 | 386.996 | 444.723 | | 444.723 | 63.565 | 450.561 |

4. SAMEPERIODLASTYEAR \ DATEADD

Acumulado del año pasado.

¿Cómo crear una medida que realice el acumulado de los mismos en estudio en el contexto actual, pero para el año anterior?

= **CALCULATE**([Ingresos], **SAMEPERIODLASTYEAR**(**DATESYTD**(Calendario[Fechas])))



| Año | Ingresos Totales | YTD Ingresos | PY YTD Ingresos |
|------------|------------------|--------------|-----------------|
| 2000 | 180.680,24 | 180.680,24 | 8.685,83 |
| Enero | 13.508,96 | 13.508,96 | |
| Febrero | 17.592,04 | 31.101,00 | |
| Marzo | 18.328,51 | 49.429,52 | |
| Abril | 14.726,52 | 64.156,03 | |
| Mayo | 13.507,17 | 77.663,20 | |
| Junio | 17.466,96 | 95.130,16 | |
| Julio | 15.729,11 | 110.859,27 | |
| Agosto | 15.056,15 | 125.915,42 | |
| Septiembre | 11.207,04 | 137.122,46 | |
| Octubre | 14.678,72 | 151.801,18 | |
| Noviembre | 14.862,94 | 166.664,13 | |
| Diciembre | 14.016,11 | 180.680,24 | 8.685,83 |
| 2001 | 166.860,36 | 166.860,36 | 180.680,24 |
| Enero | 14.558,15 | 14.558,15 | 13.508,96 |
| Febrero | 11.967,01 | 26.525,15 | 31.101,00 |
| Marzo | 15.791,46 | 42.316,61 | 49.429,52 |
| Abril | 14.528,67 | 56.845,28 | 64.156,03 |
| Mayo | 13.129,91 | 69.975,19 | 77.663,20 |
| Junio | 17.100,43 | 87.075,62 | 95.130,16 |
| Julio | 14.712,75 | 101.788,37 | 110.859,27 |
| Agosto | 12.854,18 | 114.642,55 | 125.915,42 |
| Septiembre | 13.988,98 | 128.631,53 | 137.122,46 |
| Octubre | 11.852,06 | 140.483,59 | 151.801,18 |
| Total | 3.320.885,54 | | |

Figura 19. 30 – Matriz con Medida PY YTD Ingresos

5. PREVIOUSMONTH

Devuelve una tabla

Funciona de manera similar que la función DATEADD para los meses, con pequeñas diferencias:

= **CALCULATE**([Ingresos],**DATEADD**(Calendario[Fechas], -1,MONTH))

Ejemplo de comparación:

| 1 Cantidad Mes Anterior = CALCULATE([Cantidad], DATEADD(Dates[Fechas], -1, MONTH)) | | | | | | | |
|--|----------|-----------------------|-------------------------|-----------------------|-------------------------|--------------------------------|----------------|
| Año | Cantidad | Cantidad Año Anterior | Cantidad Año Anterior 2 | Cantidad Mes Anterior | Cantidad Mes Anterior 2 | Producción Acumulada en el Año | Cantidad Total |
| 2020 | 65.216 | | | 60.303 | | 65.216 | 65.216 |
| Enero | 5.196 | | | | | 5.196 | 65.216 |
| Febrero | 5.157 | | | 5.196 | 5.196 | 10.353 | 65.216 |
| Marzo | 5.323 | | | 5.157 | 5.157 | 15.676 | 65.216 |
| Abril | 5.883 | | | 5.323 | 5.323 | 21.559 | 65.216 |
| Mayo | 5.099 | | | 5.883 | 5.883 | 26.658 | 65.216 |
| Junio | 5.703 | | | 5.099 | 5.099 | 32.361 | 65.216 |
| Julio | 5.579 | | | 5.703 | 5.703 | 37.940 | 65.216 |
| Agosto | 5.603 | | | 5.579 | 5.579 | 43.543 | 65.216 |
| Septiembre | 5.857 | | | 5.603 | 5.603 | 49.400 | 65.216 |
| Octubre | 5.668 | | | 5.857 | 5.857 | 55.068 | 65.216 |
| Noviembre | 5.235 | | | 5.668 | 5.668 | 60.303 | 65.216 |
| Diciembre | 4.913 | | | 5.235 | 5.235 | 65.216 | 65.216 |
| 2021 | 64.565 | 65.216 | 65.216 | 64.132 | 4.913 | 64.565 | 64.565 |
| Enero | 6.307 | 5.196 | 5.196 | 4.913 | 4.913 | 6.307 | 64.565 |
| Febrero | 4.179 | 5.157 | 5.157 | 6.307 | 6.307 | 10.486 | 64.565 |
| Marzo | 5.378 | 5.323 | 5.323 | 4.179 | 4.179 | 15.864 | 64.565 |
| Abril | 5.117 | 5.883 | 5.883 | 5.378 | 5.378 | 20.981 | 64.565 |
| Mayo | 4.964 | 5.099 | 5.099 | 5.117 | 5.117 | 25.945 | 64.565 |
| Junio | 5.528 | 5.703 | 5.703 | 4.964 | 4.964 | 31.473 | 64.565 |
| Julio | 5.525 | 5.579 | 5.579 | 5.528 | 5.528 | 36.998 | 64.565 |
| Agosto | 5.694 | 5.603 | 5.603 | 5.525 | 5.525 | 42.692 | 64.565 |
| Septiembre | 5.051 | 5.857 | 5.857 | 5.694 | 5.694 | 47.743 | 64.565 |
| Octubre | 5.777 | 5.668 | 5.668 | 5.051 | 5.051 | 53.520 | 64.565 |
| Total | 450.561 | 386.996 | 386.996 | 444.723 | | 63.565 | 450.561 |

| X ✓ | | 1 Cantidad Mes Anterior 2 = CALCULATE([Cantidad], PREVIOUSMONTH(Dates[Fechas])) | | | | | |
|------------|----------|---|-------------------------|-----------------------|-------------------------|--------------------------------|----------------|
| Año | Cantidad | Cantidad Año Anterior | Cantidad Año Anterior 2 | Cantidad Mes Anterior | Cantidad Mes Anterior 2 | Producción Acumulada en el Año | Cantidad Total |
| 2020 | 65.216 | | | 60.303 | | 65.216 | 65.216 |
| Enero | 5.196 | | | | | 5.196 | 65.216 |
| Febrero | 5.157 | | | 5.196 | 5.196 | 10.353 | 65.216 |
| Marzo | 5.323 | | | 5.157 | 5.157 | 15.676 | 65.216 |
| Abril | 5.883 | | | 5.323 | 5.323 | 21.559 | 65.216 |
| Mayo | 5.099 | | | 5.883 | 5.883 | 26.658 | 65.216 |
| Junio | 5.703 | | | 5.099 | 5.099 | 32.361 | 65.216 |
| Julio | 5.579 | | | 5.703 | 5.703 | 37.940 | 65.216 |
| Agosto | 5.603 | | | 5.579 | 5.579 | 43.543 | 65.216 |
| Septiembre | 5.857 | | | 5.603 | 5.603 | 49.400 | 65.216 |
| Octubre | 5.668 | | | 5.857 | 5.857 | 55.068 | 65.216 |
| Noviembre | 5.235 | | | 5.668 | 5.668 | 60.303 | 65.216 |
| Diciembre | 4.913 | | | 5.235 | 5.235 | 65.216 | 65.216 |
| 2021 | 64.565 | 65.216 | 65.216 | 64.132 | 4.913 | 64.565 | 64.565 |
| Enero | 6.307 | 5.196 | 5.196 | 4.913 | 4.913 | 6.307 | 64.565 |
| Febrero | 4.179 | 5.157 | 5.157 | 6.307 | 6.307 | 10.486 | 64.565 |
| Marzo | 5.378 | 5.323 | 5.323 | 4.179 | 4.179 | 15.864 | 64.565 |
| Abril | 5.117 | 5.883 | 5.883 | 5.378 | 5.378 | 20.981 | 64.565 |
| Mayo | 4.964 | 5.099 | 5.099 | 5.117 | 5.117 | 25.945 | 64.565 |
| Junio | 5.528 | 5.703 | 5.703 | 4.964 | 4.964 | 31.473 | 64.565 |
| Julio | 5.525 | 5.579 | 5.579 | 5.528 | 5.528 | 36.998 | 64.565 |
| Agosto | 5.694 | 5.603 | 5.603 | 5.525 | 5.525 | 42.692 | 64.565 |
| Septiembre | 5.051 | 5.857 | 5.857 | 5.694 | 5.694 | 47.743 | 64.565 |
| Octubre | 5.777 | 5.668 | 5.668 | 5.051 | 5.051 | 53.520 | 64.565 |
| Total | 450.561 | 386.996 | 386.996 | 444.723 | | 63.565 | 450.561 |