

SPRINT 5: Iniciacion al analisis de datos con Power BI

NIVELL 1

Ex1

Se han cargado los datos a Power BI a través de los botones '*Get Data – Text/CSV file*'.

Cargar los datos al programa nno tiene mas secreto, a continuacion listaremos algunas de las acciones realizadas a través de *Power Query* para adecuar la BD.

Table **companies** – first row as headers: de esta forma coje la primera línea del documento CSV i la transforma en los títulos de las columnas de la tabla.

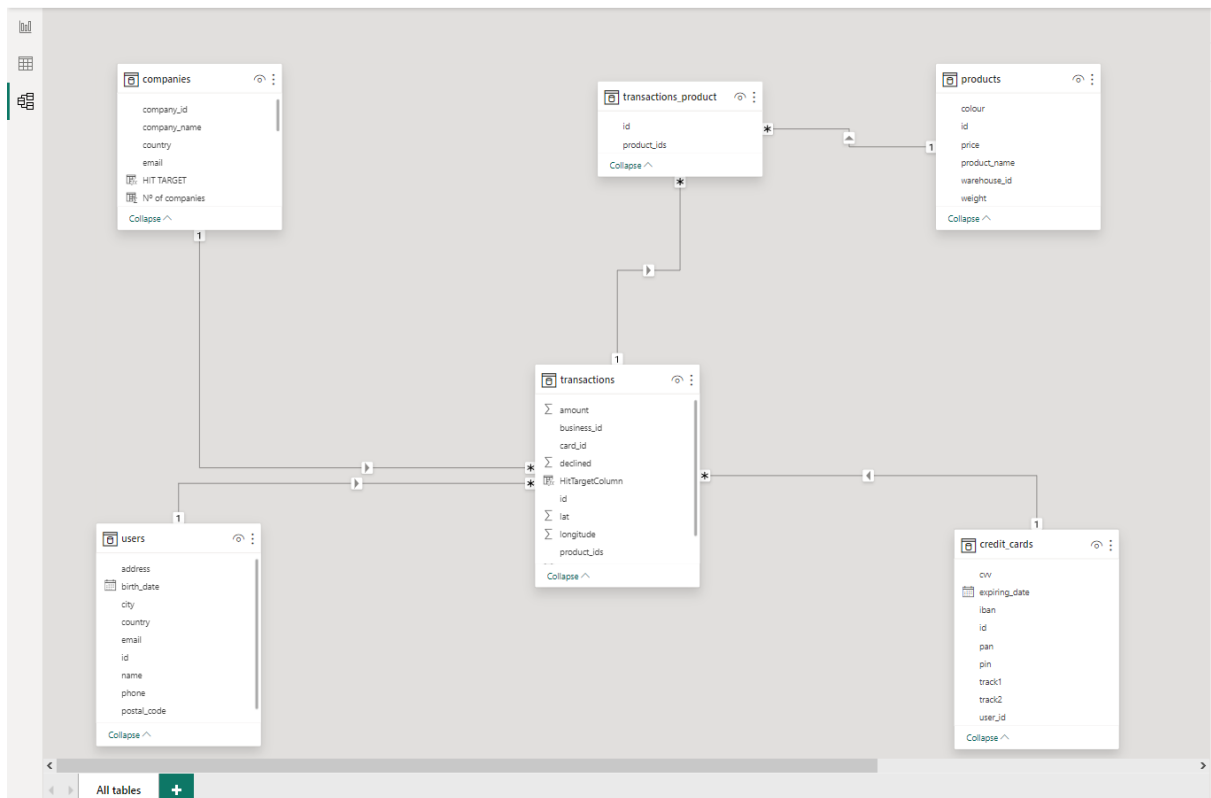
Table **credit_cards** – *expiring_date* change to *date* data type

Table **products** – replace values \$ to nothing: esto nos sirve para si luego necesitamos realizar alguna operación (AVG(), MAX(), SUM(),...)

Table **users** – append as new uk, ca, usa; move to new group don't enable load: realizamos un *append as new* para unir las tres tablas como una nueva. Luego movemos las otras tablas y desactivamos su carga en el modelo para que quede más organizado.

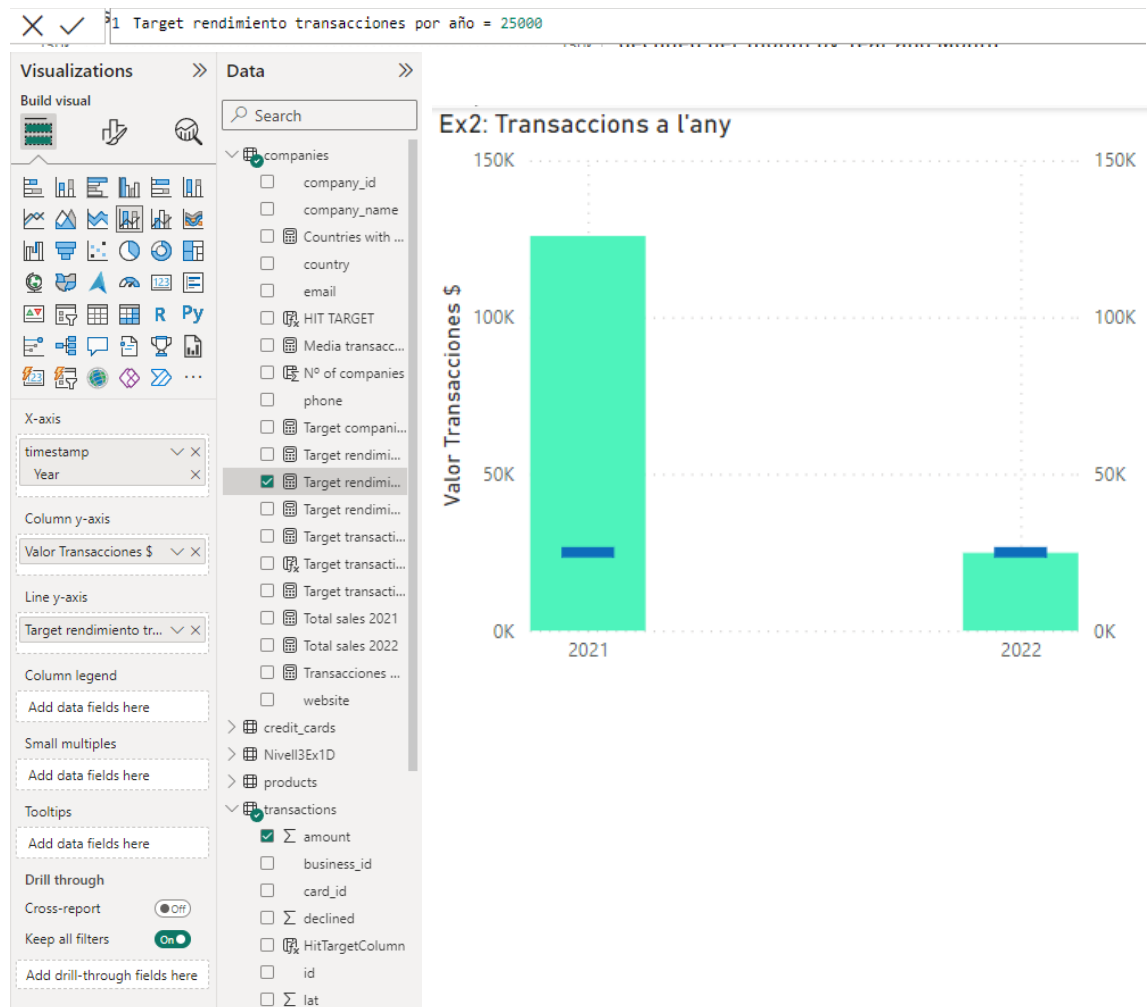
Create **transaction_product** to eliminate relation **N to N** – split column by delimiter into rows: como en MySQL creamos una table intermedia con relación **N to 1** con ambas a la cual llamaremos *transaction_product*. En este caso realizar esto es mucho más sencillo con *Power Query*, duplicando la tabla *transactions*, separando los valores de *producto_ids* en *rows* y luego eliminando las columnas que no nos interesan.

Finalmente establecemos las relaciones en la pestaña de *Model View* para crear el modelo en estrella, que queda de la siguiente forma.



Ex2

Para mostrar esta información utilizaremos un *'Line and stacked columns chart'*, donde la línea nos marcará el target de la empresa, en este caso 25.000\$ en transacciones al año. Para ello crearemos una *Quick measure* que será igual a este valor. En el *'eje y'* pondremos el **SUM()** de el amount y en el *'eje x'* pondremos los años. El resultado es el siguiente:



Ex3 y 4

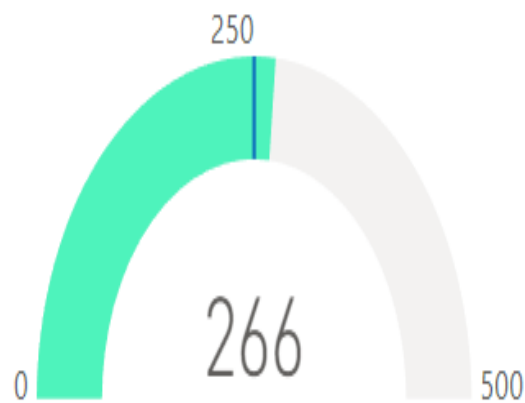
Para mostrar esta información usaremos un *Gauge*; la información la obtendremos a través de una *quick measure* que será un **CALCULATE()** de un **AVERAGE()** con un filtro para el año (2021 y 2022 respectivamente). Crearemos otro *quick measure* que sea el target de 250 (aunque en este caso no es necesario porque si el target no ha de ser un calculo se puede introducir el valor directamente en este tipo de gráfico, que proporciona la opción en *format your visual*).

1	Target transactions per year = 250
2	Total sales 2021 = CALCULATE(AVERAGE(transactions[amount]), YEAR(transactions[timestamp]) = 2021, transactions[declined] = 0)

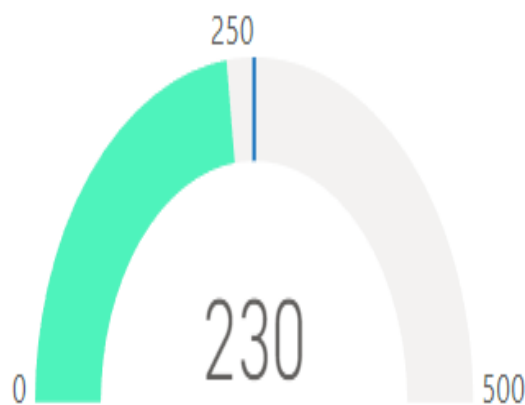
Visualizations
Build visual
Value
Total sales 2021
Minimum value
Add data fields here
Maximum value
Add data fields here
Target value
Target transactions pe...
Tooltips
Add data fields here
Drill through
Cross-report
Keep all filters
Add drill-through fields here

Data
Search
companies
company_id
company_name
Countries with ...
country
email
HIT TARGET
Media transacc...
Nº of companies
phone
Target compani...
Target rendimi...
Target rendimi...
Target rendimi...
Target transacti...
Target transacti...
Target transacti...
Total sales 2021
Total sales 2022
Transacciones ...
website
credit_cards
Nivell3Ex1D
products
transactions
amount
business_id
card_id
declined
HitTargetColumn
id
lat

Ex3: Total sales 2021 and Target transactions per year



Ex4: Total sales 2022 and Target transactions per year

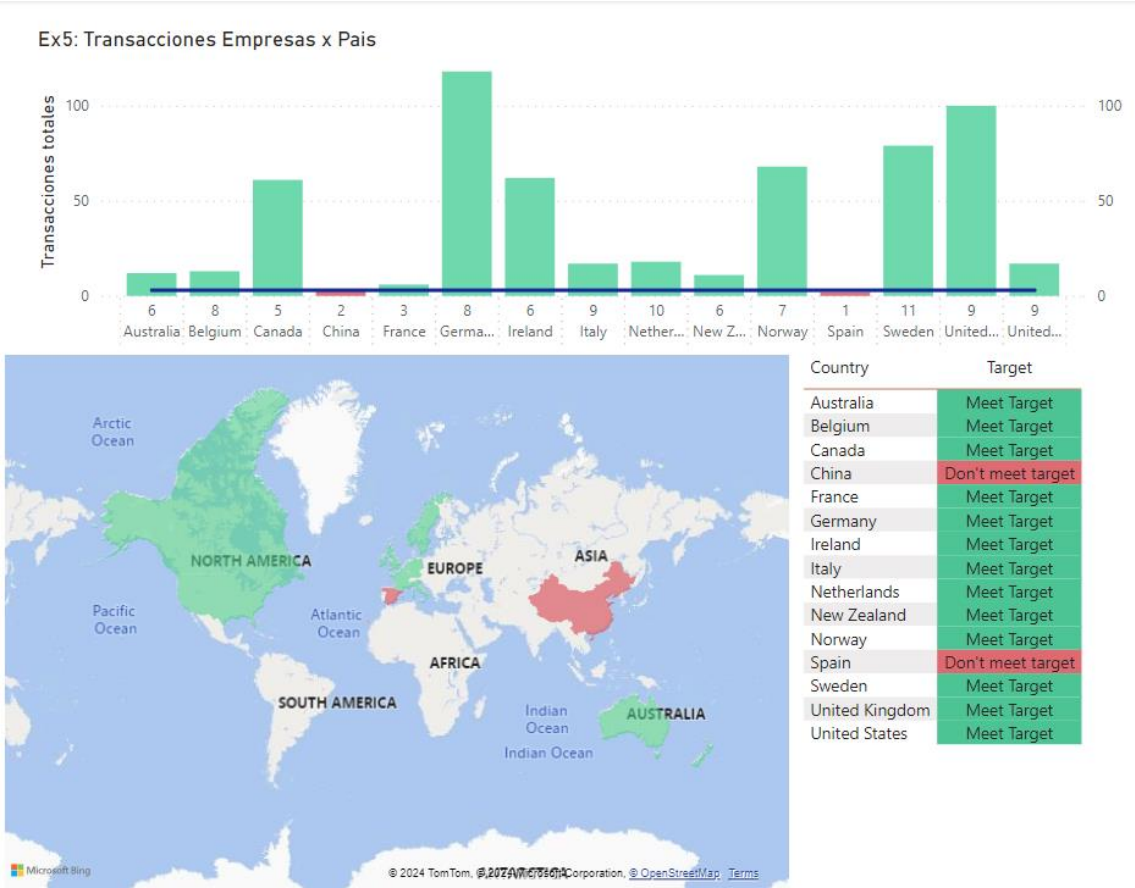


Ex5

Para saber si un país tiene mas de 3 empresas activas usaremos la siguiente formula DAX:

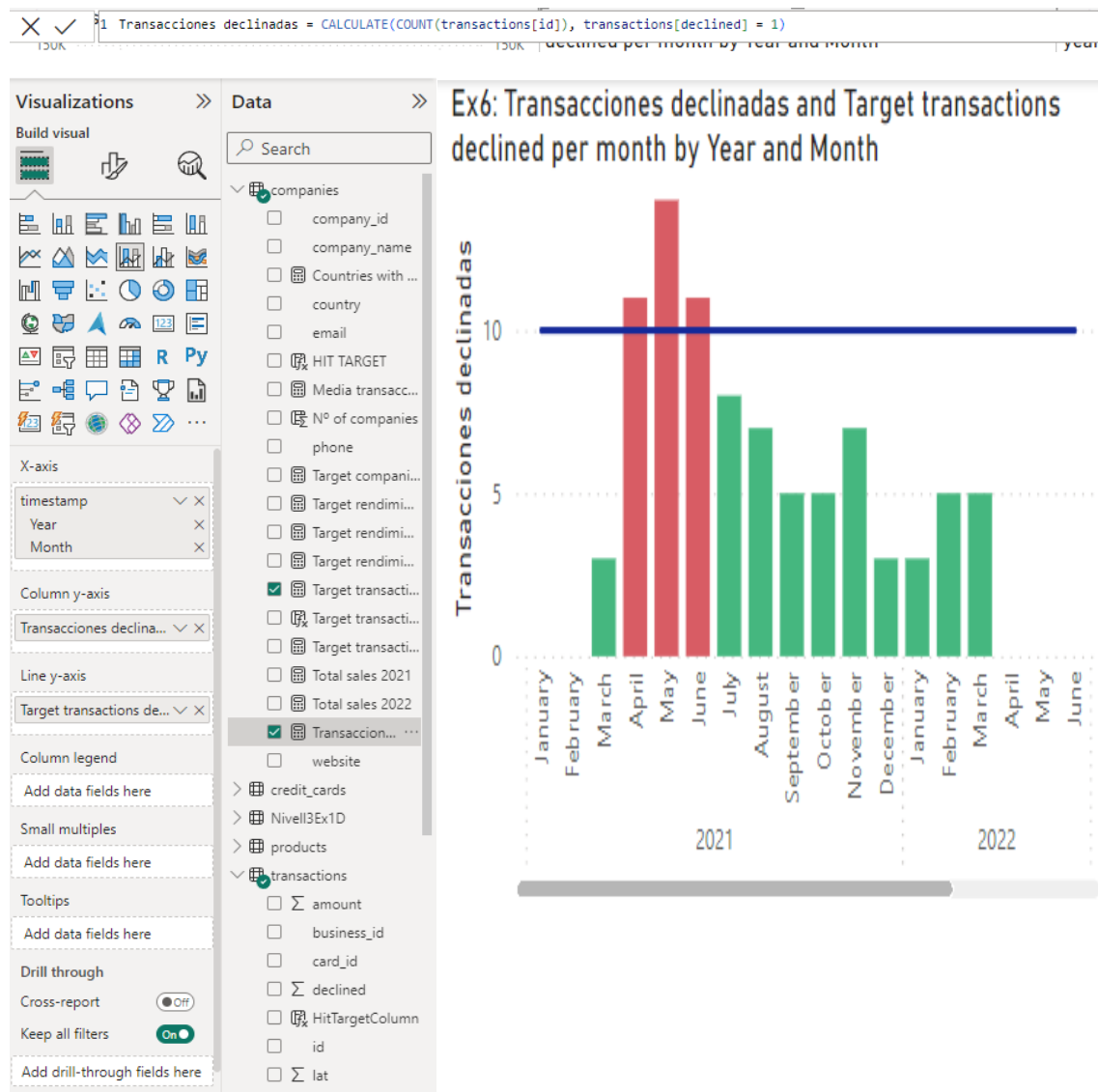
```
1 HIT TARGET =
2 IF(
3     CALCULATE(
4         COUNTROWS(VALUES (transactions[business_id])),
5         ALLEXCEPT(transactions, transactions[business_id], companies[country])
6     ) >= 3,
7     "Meet Target",
8     "Don't meet target"
9 )
```

Este código crea una **columna** en la tabla *companies*, con una condición **IF**, que devuelve un valor u otro en función de si la condición se cumple (en este caso que el país tenga 3 empresas activas). Utilizaremos un **CALCULATE()** con la función **COUNTROWS(VALUES())**, que contare los *business_id* distintos en la tabla *transactions*, i la función **ALLEXCEPT()** que eliminara todos los filtros excepto *country* y *bussiness_id*. La información la mostraremos en una tabla, un mapa y un 'Line and stacked column chart':



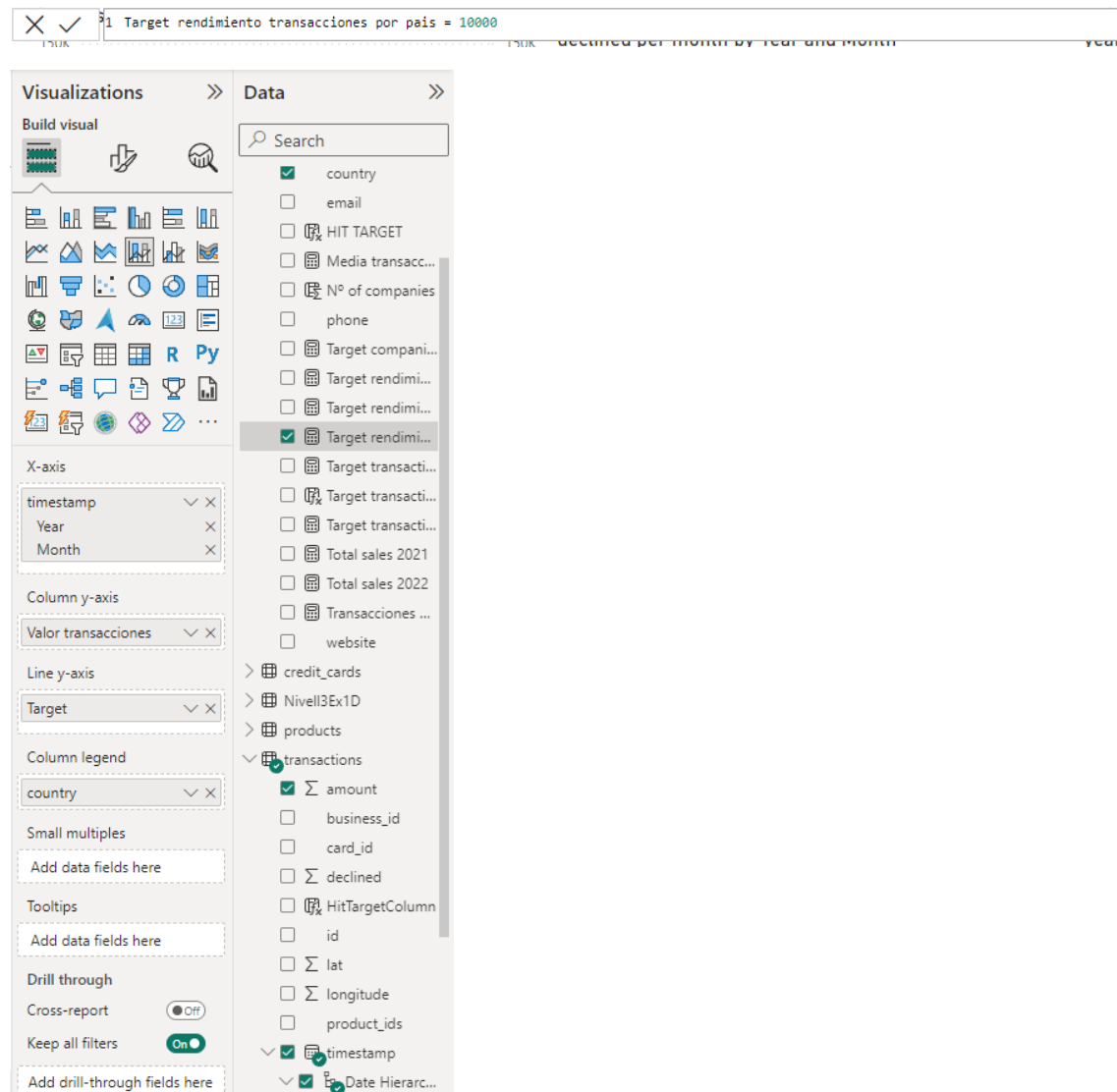
Ex6

Esta información se mostrará a través de un 'Line and stacked column chart'. Crearemos una *quick measure* que será igual a 10 (nuestra target) y otra que será un **CALCULATE(COUNT())** las id de transactions con el filtro de que *declined* = 1. Mostraremos esta info a lo largo de los meses y años.

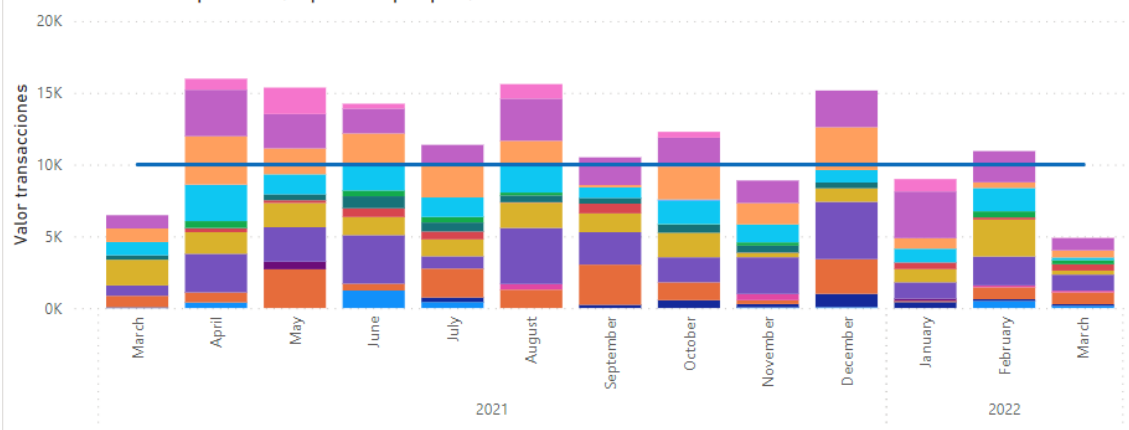


Ex7

Volveremos a utilizar un 'Line and stacked column chart' para mostrar esta info, que mostraremos a lo largo de los meses/años, pero donde las columnas serán agregados de los países. Crearemos una *quick measure* que será igual a 10000, nuestro target.



Ex7: Transacciones por mes (Separadas por país)



Ex8

Para crear esta tabla utilizaremos *Power Query*. Haremos un *Merge Queries as new* con las tablas *user* y *transactions* y llamaremos a la nueva tabla *transaction_users*. Eliminaremos las columnas que no nos interesan, mayoritariamente de *transactions*. Juntaremos las columnas *name* y *surname* para tener el nombre completo, extraeremos la edad de la fecha de nacimiento creando una columna (que vendrá en *duration DATA TYPE*) con la fórmula `Age = Date.From(DateTime.LocalNow()) - [birth_date]`. Luego haremos el **AVERAGE()** de la columna *amount* y lo agruparemos con el nombre de la persona y la edad. Para finalizar crearemos una columna condicional que nos indicara si esa persona gasta mas o menos que el target especificado(320\$/300€).

The screenshot shows the Power Query Editor interface. The main table has the following columns: **Full name**, **birth_date**, **Age**, **Avg transaction \$**, **Avg transaction €**, **Spender Type**, and **150Spender**. The table contains 38 rows of data, representing individual users and their transaction statistics. The right-hand pane shows the 'Query Settings' for 'transaction_users', including properties like 'Name', 'Source', and 'Applied Steps'.

Ex8: Transacciones x Usuario

Full name	Age	Avg transaction \$	Avg transaction €	Spender Type
Acton Gallegos	35	283.15	262.1759259	Small Spender
Aiko Chaney	37	278.36	257.7361111	Small Spender
Ainsley Herrera	27	105.51	97.69444444	Small Spender
Alan Vazquez	27	257.86	238.7546296	Small Spender
Alika Kinney	28	394.59	365.3611111	Big Spender
Allen Calhoun	36	286.60	265.3703704	Small Spender
Amal Kennedy	37	411.64	381.1481481	Big Spender
Amber Blevins	44	193.33	179.0092593	Small Spender
Amelia Valenzuela	38	321.39	297.5833333	Big Spender
Andrew Strong	26	375.48	347.662037	Big Spender
Astra Baldwin	24	472.18	437.2037037	Big Spender
Athena Malone	32	162.56	150.5185185	Small Spender
Avye Key	37	396.04	366.6990741	Big Spender
Bert Juarez	35	381.17	352.9351852	Big Spender
Bertha Sloan	27	58.16	53.85185185	Small Spender
Beverly Burt	27	82.43	76.32407407	Small Spender
Blake Strickland	40	149.06	138.0185185	Small Spender
Blaze Daniel	25	267.76	247.9212963	Small Spender
Blaze Duke	32	399.62	370.0185185	Big Spender
Brennan Wynn	35	264.63	245.0314815	Small Spender
Brent Bates	27	451.66	418.2037037	Big Spender
Brett Kirby	35	325.18	301.0925926	Big Spender
Brock Doyle	38	28.95	26.80555556	Small Spender
Brody Goodwin	41	478.54	443.0925926	Big Spender

Ex9

De las visualizaciones presentados podemos extraer la conclusión de que los targets de la empresa se están cumpliendo en mayor parte. El rendimiento de las transacciones anual se supera en 2021 y seguramente también en 2022 (ya que solo hay datos hasta marzo de ese año), el valor medio de cada transacción también ronda los 250 en ambos años y las transacciones declinadas, así como las transacciones por mes no se cumplen siempre, pero si en mayor parte.

NIVEL 2

Ex1

Se utiliza un 'Line stacked column chart' con un *quick measure* que equivaldrá al target de 12.500. De esta forma se visualiza claramente que meses llegan o no al target.



Visualizations

Build visual

Data

Search

- Target rendimi...
- Target transaci...
- Target transaci...
- Target transaci...
- Total sales 2021
- Total sales 2022
- Transacciones ...
- website
- credit_cards
- Nivel3Ex1D
- products
- transactions
 - amount
 - business_id
 - card_id
 - declined
 - HitTargetColumn
 - id
 - lat
 - longitude
 - product_ids
 - timestamp
 - Date Hierarc...
 - Year
 - Quarter
 - Month
 - Day
 - user_id
- transactions_product
- transactions_users
 - 150Spender
 - ...

X-axis

timestamp

Year

Month

Column y-axis

Sum of amount

Line y-axis

Target rendimiento m...

Column legend

Add data fields here

Small multiples

Add data fields here

Tooltips

Add data fields here

Drill through

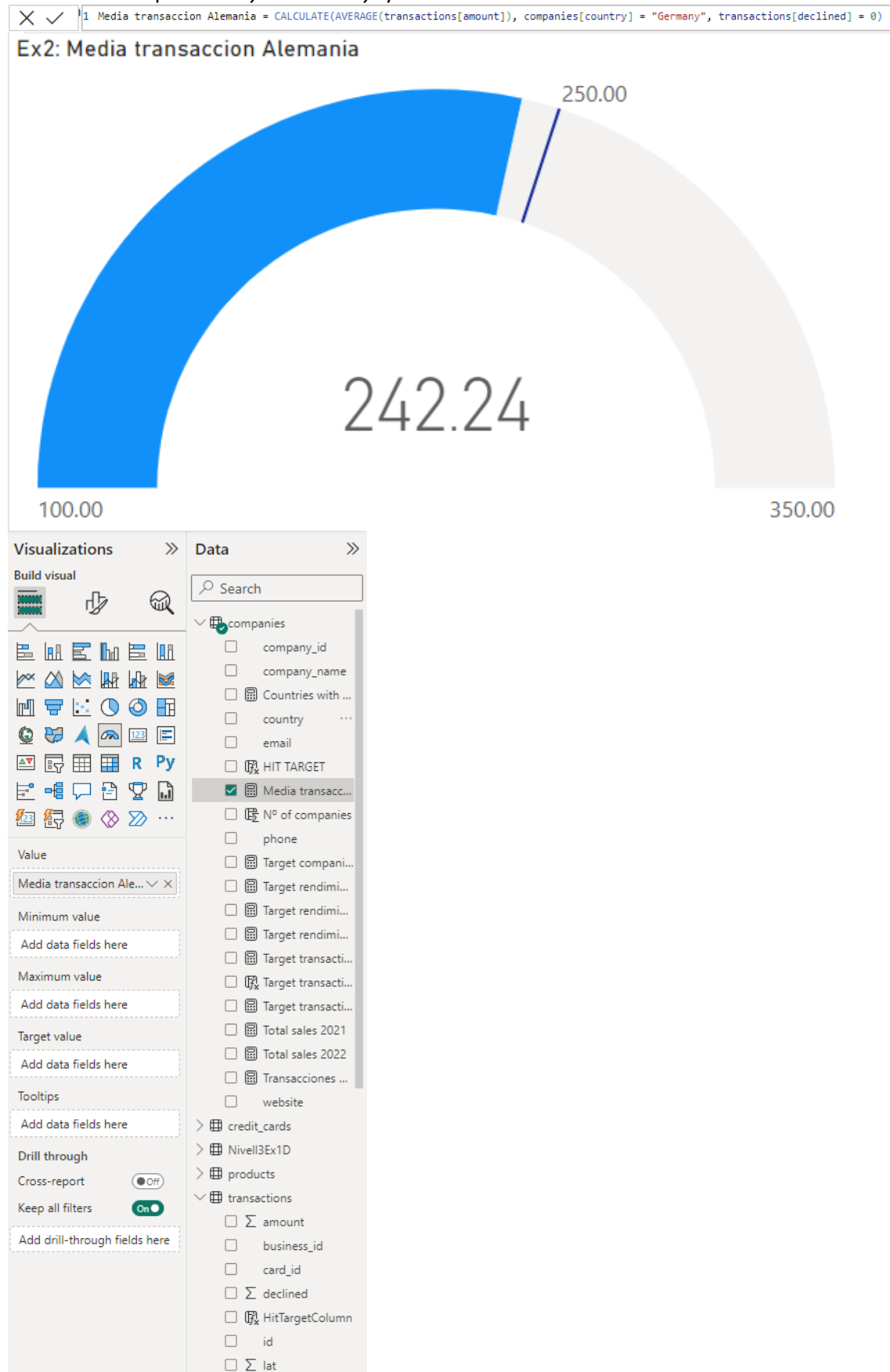
Cross-report

Keep all filters

Add drill-through fields here

Ex2

Se utilizará un *quick measure* que será un **CALCULATE(AVERAGE())** del *amount* de *transactions* con el filtro de que *country* = 'Germany' y *declined* = 0.



Ex3

Los meses en que no se cumplió el objetivo fueron Marzo, Julio, Septiembre, Octubre y Noviembre en 2021; y Enero, Febrero y Marzo en 2022. Es decir, en 8 de 13 meses.

NIVEL 3

Ex1

1.B - Productos por usuario: se valora que la mejor forma de visualizar esta información es en modo de tabla.

Ex1.B: Productos comprados

name	surname	Nº de productos
Ainsley	Herrera	1
Alan	Vazquez	2
Alika	Kinney	3
Allen	Calhoun	3
Amal	Kennedy	1
Amber	Blevins	4
Amelia	Valenzuela	1
Andrew	Strong	4
Astra	Baldwin	3
Athena	Malone	5
Avey	Key	7
Bert	Juarez	4
Bertha	Sloan	4
Beverly	Burt	1
Blake	Strickland	3
Blaze	Daniel	5
Blaze	Duke	4
Brennan	Wynn	25
Brent	Bates	3
Brett	Kirby	5
Brock	Doyle	1
Brody	Goodwin	4
Brody	Talley	3
Brooke	Jensen	2
Burke	Graham	5
Camden	Carpenter	2
Camilla	Roach	5
Camilla	Zimmerman	3
Carly	Mathews	4
Cassandra	Ferguson	1
Celeste	Ellis	2
Chase	Ellis	4
Chester	Haynes	4

Visualizations

Build visual

Columns

name surname Nº de productos

Drill through

Cross-report Off

Keep all filters On

Add drill-through fields here

Data

Search

- companies
- credit_cards
- Nivel3Ex1D
- products
- transactions
- transactions_product
 - id
 - product_ids
- transactions_users
 - 150Spender
 - Σ Age
 - Σ Avg transaction \$
 - Avg transaction €
- birth_date
- Full name
- Spender Type
- users
 - address
 - birth_date
 - city
 - country
 - email
 - id
 - name
 - phone
 - postal_code
 - surname

1.C - Media de venta por usuario: se valora que la mejor forma de visualizar esta información es en modo de tabla. Se utilizará *PowerQuery* para crear una columna condicional que muestre si un usuario gasta mas o menos de 150\$ de media por transacción. Podemos usar la tabla *transaction_user* que hemos creado anteriormente para esto.

Full name	Avg Transaction	150Spender
Acton Gallegos	283.15	150Spender
Aiko Chaney	278.36	150Spender
Ainsley Herrera	105.51	Not150Spender
Alan Vazquez	257.86	150Spender
Alika Kinney	394.59	150Spender
Allen Calhoun	286.60	150Spender
Amal Kennedy	411.64	150Spender
Amber Blevins	193.33	150Spender
Amelia Valenzuela	321.39	150Spender
Andrew Strong	375.48	150Spender
Astra Baldwin	472.18	150Spender
Athena Malone	162.56	150Spender
Avye Key	396.04	150Spender
Bert Juarez	381.17	150Spender
Bertha Sloan	58.16	Not150Spender
Beverly Burt	82.43	Not150Spender
Blake Strickland	149.06	Not150Spender
Blaze Daniel	267.76	150Spender
Blaze Duke	399.62	150Spender
Brennan Wynn	264.63	150Spender
Brent Bates	451.66	150Spender
Brett Kirby	325.18	150Spender
Brock Doyle	28.95	Not150Spender
Brody Goodwin	478.54	150Spender
Brody Talley	458.66	150Spender
Brooke Jensen	271.27	150Spender
Burke Graham	414.58	150Spender
Camden Carpenter	391.38	150Spender
Camilla Roach	445.48	150Spender
Camilla Zimmerman	424.18	150Spender
Carly Mathews	211.58	150Spender
Cassandra Ferguson	274.28	150Spender
Celeste Ellis	399.71	150Spender

Visualizations
Build visual
Columns
Full name
Avg Transaction
150Spender
Drill through
Cross-report
Keep all filters
Add drill-through fields here

Data
Search
companies
credit_cards
Nivel3Ex1D
products
transactions
transactions_product
transactions_users
150Spender
Age
Avg transaction \$
Avg transaction €
birth_date
Full name
Spender Type
users

1.D - Precio de producto más caro por usuario: Se utilizará *PowerQuery* para crear una tabla que contenga esta información. El proceso será similar al del ejercicio 8 del nivel 1. Se utilizará el comando *merge query* (tablas *user*, *transactions* separando los *producto_ids* en columnas y tabla *products*), se eliminarán columnas no deseados y luego se hará un *group by* del **MAX()** del precio de producto con los nombres de usuario.

users.name	users.surname	products.price
Griffith	Golden	195.94
Guinevere	Kemp	195.94
Hedwig	Gilbert	195.94
Jael	Robles	195.94
Jameson	Hunt	195.94
Jarrold	Fields	195.94
Joshua	Russell	195.94
Keane	Parks	195.94
Kenyon	Hartman	195.94
Kim	Mooney	195.94
Kimberley	Avila	195.94
Lane	Paul	195.94
Linda	Gaines	195.94
Lynn	Riddle	195.94
Medge	Nieves	195.94
Ocean	Nelson	195.94
Philip	Carey	195.94
Phoebe	Roth	195.94
Porter	Francis	195.94
Preston	Hubbard	195.94
Remedios	Case	195.94
Roth	Cook	195.94
Sasha	Emerson	195.94
Slade	Poole	195.94
Solomon	Blake	195.94
Stuart	Small	195.94
Ursula	Stewart	195.94
Victor	Valencia	195.94
Amber	Blevins	172.78
Avye	Key	172.78
Brennan	Wynn	172.78
Camilla	Roach	172.78
Chloe	Keith	172.78

Ex1.D: Producto mas caro por €

users.name	users.surname	products.price
Griffith	Golden	195.94
Guinevere	Kemp	195.94
Hedwig	Gilbert	195.94
Jael	Robles	195.94
Jameson	Hunt	195.94
Jarrold	Fields	195.94
Joshua	Russell	195.94
Keane	Parks	195.94
Kenyon	Hartman	195.94
Kim	Mooney	195.94
Kimberley	Avila	195.94
Lane	Paul	195.94
Linda	Gaines	195.94
Lynn	Riddle	195.94
Medge	Nieves	195.94
Ocean	Nelson	195.94
Philip	Carey	195.94
Phoebe	Roth	195.94
Porter	Francis	195.94
Preston	Hubbard	195.94
Remedios	Case	195.94
Roth	Cook	195.94
Sasha	Emerson	195.94
Slade	Poole	195.94
Solomon	Blake	195.94
Stuart	Small	195.94
Ursula	Stewart	195.94
Victor	Valencia	195.94
Amber	Blevins	172.78
Avye	Key	172.78
Brennan	Wynn	172.78
Camilla	Roach	172.78
Chloe	Keith	172.78

1.A - Número de transacciones totales por cliente: se ha considerado que esta podría ser una medida interesante para ver que clientes son los que más transacciones realizan. Se ha volcado un **COUNT()** de los *transaction_id* y los nombres de los usuarios en una tabla.

Ex1.A: Transacciones por cliente

Count of id	name	surname
198	Hedwig	Gilbert
124	Ocean	Nelson
116	Kenyon	Hartman
89	Lynn	Riddle
65	Brennan	Wynn
53	Slade	Poole
41	Kim	Mooney
12	Hilary	Ferguson
11	Jameson	Hunt
11	Neil	Powers
10	Clark	Olson
10	Haley	Fitzpatrick
10	Leandra	Cherry
8	Elijah	Stone
7	Aiko	Chaney
7	Avye	Key
7	Chloe	Keith
7	Gemma	Calderon
7	Hilda	Levy
7	Jarrold	Fields
7	Lane	Paul
7	Lewis	Lynn
7	Rosalyn	Blake
6	Blaze	Daniel
6	Cleo	George
6	Dean	Donovan
6	Elton	Roberson
6	Heather	Burks
6	Keegan	Watson
6	Leonard	Davenport
6	Maxwell	Holden
6	Medge	Nieves
6	Minerva	Wilkins

Visualizations

Build visual

Columns

Count of id

name

surname

Drill through

Cross-report Off

Keep all filters On

Add drill-through fields here

Data

Search

- companies
- credit_cards
- Nivell3Ex1D
- products
- transactions
- transactions_product
 - ☒ id
 - ☐ product_ids
- transactions_users
- users
 - ☐ address
 - ☐ birth_date
 - ☐ city
 - ☐ country
 - ☐ email
 - ☐ id
 - ☒ name
 - ☐ phone
 - ☐ postal_code
 - ☒ surname