



# **Análisis de Reviews y Satisfacción de AirBnB Brasil y México de junio 2010 a febrero 2021**

**Entrega final Data Analytics**

**Comisión 49100**

**Alumno:**

**José Chirif Molina**

**Marzo 2024**

## Índice

<b>I. Introducción.....</b>	<b>5</b>
1. Introducción.....	5
2. Tabla de versiones: .....	5
<b>II. Dataset y generalidades del proyecto .....</b>	<b>5</b>
3. Dataset inicial .....	5
4. Alcance del proyecto .....	5
5. Hipótesis / Objetivos .....	5
5.1. Hipótesis 1 .....	5
5.2. Hipótesis 2 .....	6
6. Herramientas utilizadas.....	6
<b>III. Limpieza Dataset inicial (Ms. Excel) y relaciones .....</b>	<b>6</b>
7. Limpieza Dataset inicial .....	6
7.1. Columnas requeridas .....	6
7.2. Relación entre tablas (diagrama EDR).....	8
7.3. Caracteres especiales y otras consideraciones.....	8
7.4. Tablas finales.....	8
<b>IV. SQL .....</b>	<b>10</b>
8. Procesos de carga en sql .....	10
8.1. Proceso para la creación de la base de datos .....	10
8.2. Proceso para la creación de tablas.....	10
8.3. Carga de datos y Base de datos final.....	11
9. Relación de tablas en SQL.....	11
10. Exportación de sql a csv .....	13

<b>V. Análisis de datos (Power BI)</b>	<b>15</b>
11. Carga de datos en Power BI	15
12. Transformaciones y limpieza en Power Query	15
12.1. ETL en Power Query	15
12.2. ETL en Power BI	17
13. Estructura del reporte Power BI	17
14. Creación de tabla calendario	18
14.1. En Power Query (tabla calendario)	18
14.2. En Power BI (tabla calendario)	19
15. Relaciones y tablas puentes	20
15.1. Tablas complementarias de Calendario	21
15.2. Complementando categorías de Puntajes en Reviews	21
15.3. Relaciones con tablas complementarias	22
16. DAX Columnas Calculadas	23
16.1. Tabla Propiedades	23
16.2. Tabla Reviews	25
16.3. Tabla Reviews _satisfaccion total	26
17. DAX Medidas calculadas	26
17.1. Tabla medidas	26
17.2. Medidas	27
18. Otros preparativos	29
18.1. Ordenamiento de meses y días en tablas complementarias calendario	29
18.2. Ordenamiento de categorías de Satisfacción	30
18.3. Tooltips	30
19. Estructura de las solapas	31
20. Reporte	32

20.1.	Portada .....	32
20.2.	Análisis de Reviews .....	32
20.3.	Análisis de Satisfacción .....	33
20.4.	Análisis de precios .....	33
<b>VI.</b>	<b>Conclusiones y recomendaciones.....</b>	<b>34</b>
<b>21.</b>	<b>Confirmación de hipótesis.....</b>	<b>34</b>
21.1.	Hipótesis 1 .....	34
21.2.	Hipótesis 2 .....	34
<b>22.</b>	<b>Conclusiones y recomendaciones .....</b>	<b>35</b>

## I. Introducción

### 1. Introducción

Airbnb es una plataforma de hotelería online, donde los clientes pueden reservar, pagar y dejar reservas en su plataforma online, y distintos propietarios pueden poner sus propiedades a disposición para el servicio de estadía.

### 2. Tabla de versiones:

Revisión	Fecha de presentación
0	10 de marzo del 2024

## II. Dataset y generalidades del proyecto

### 3. Dataset inicial

Para el presente trabajo se escogió el Dataset de Airbnb <https://drive.google.com/drive/folders/1ViJ-dgxlTnecN4JodyvAf2DI6z3HtuJ6>. En el cual se encuentra la data de Airbnb de Rio de Janeiro (Brasil) y Mexico city (Ciudad de México) entre el 06 de junio del 2010 al 25 de febrero del 2021. Dichas ciudades y periodo serán el alcance del proyecto.

El dataset se encuentra constituido de 5 tablas: Host, Propiedades, Reviews, Ciudades y Países.



[Dataset inicial](https://drive.google.com/drive/folders/1ViJ-dgxlTnecN4JodyvAf2DI6z3HtuJ6)

<https://drive.google.com/drive/folders/1ViJ-dgxlTnecN4JodyvAf2DI6z3HtuJ6>

### 4. Alcance del proyecto

El proyecto abarcara los análisis de Airbnb de Rio de Janeiro (Brasil) y Mexico city (Ciudad de México) realizados entre el 06 de junio del 2010 al 25 de febrero del 2021.

El proyecto solo abarca análisis de la data existente. Proyecciones u otros métodos se encuentran fuera del alcance del mismo.

### 5. Hipótesis / Objetivos

Para el presente trabajo se plantean 2 hipótesis:

#### 5.1. Hipótesis 1

Entre diciembre a febrero (Feriados y vacaciones de verano) hay mayor cantidad de reviews por la disponibilidad de las personas.

### 5.2. Hipótesis 2

Porcentualmente hablando, las propiedades de mayor costo tienen un mejor promedio de calificaciones de clientes.

Nuestro objetivo será verificar si se cumplen ambas hipótesis.

## 6. Herramientas utilizadas

A lo largo del proyecto se utilizaron los siguientes softwares:

- Microsoft Excel
- SQL server
- Power BI

## III. Limpieza Dataset inicial (Ms. Excel) y relaciones

### 7. Limpieza Dataset inicial

#### 7.1. Columnas requeridas

En base a nuestras Hipótesis / Objetivos, eliminaremos las columnas que no consideremos necesarias y renombraremos las columnas que mantendremos.

Por ende, las tablas tendrán los siguientes cambios:

Tabla Host:

Nombre	Mantener/eliminar	Nuevo nombre
host_id	Mantener	Host_ID
host_since	Mantener	Fecha_inicio_host
host_location	Eliminar (tenemos el mismo dato en la tabla propiedades)	
name	Mantener	Nombre_host
host_is_superhost	Eliminar	
host_response_time	Eliminar	
host_response_rate	Mantener	Ratio_respuestas_host
host_identity:verified	Eliminar	
host_total_listings_count	Eliminar	

Tabla Propiedades:

Nombre	Mantener/eliminar	Nuevo nombre
listing_id	Mantener	Listing_id
host_id	Mantener	Host_id

Nombre	Mantener/eliminar	Nuevo nombre
city	Mantener	ID_ciudad
apartment_type	Eliminar	
room_type	Eliminar	
bedrooms	Eliminar	
acomodates	Eliminar	
neighborhood	Eliminar	
price	Mantener	Precio

Tabla Reviews:

Nombre	Mantener /eliminar	Nuevo nombre
date	Mantener	Fecha
review_id	Mantener	Review_id
reviewer_id	Mantener	Huesped_id
listing_id	Mantener	Listing_id
review_scores_rating	Mantener	Review_puntaje_total
review_scores_accuracy	Mantener	Review_puntaje_exactituddescripcion
review_scores_cleanliness	Mantener	Review_puntaje_limpieza
review_scores_value	Mantener	Review_puntaje_valorcliente
review_scores_communication	Mantener	Review_puntaje_comunicacion
review_scores_location	Mantener	Review_puntaje_ubicacion

Tabla Ciudades:

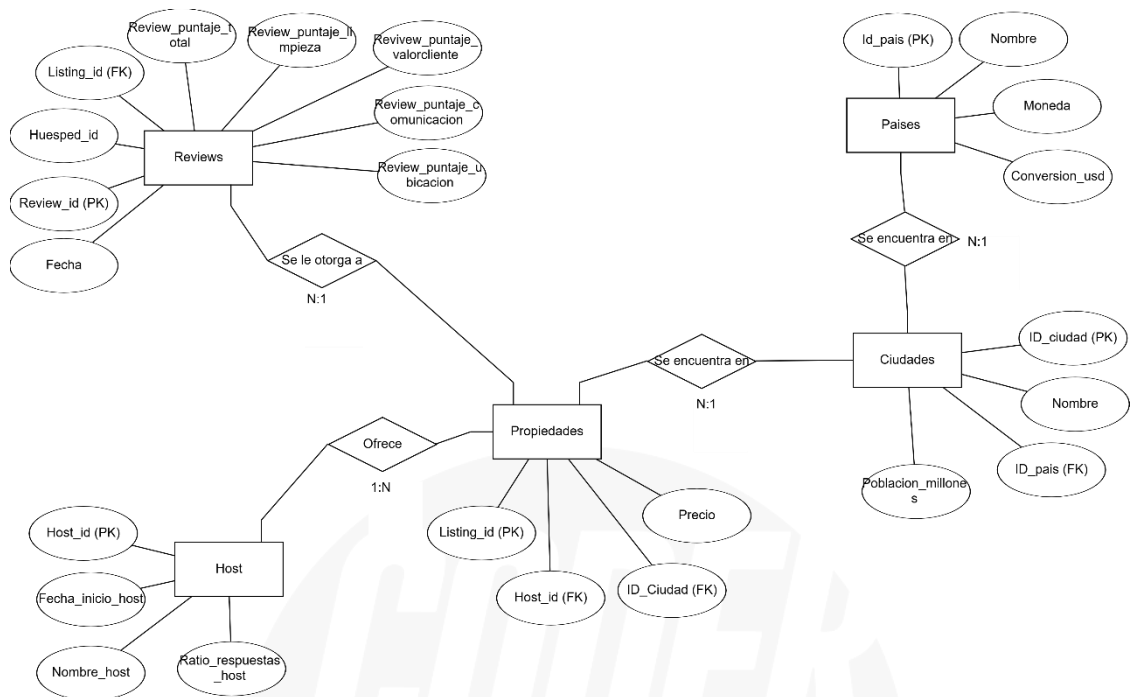
Nombre	Mantener/eliminar	Nuevo nombre
id_city	Mantener	ID_ciudad
name	Mantener	Nombre
id_pais	Mantener	ID_Pais
population_millions	Mantener	Poblacion_Millones

Tabla Paises:

Nombre	Mantener/eliminar	Nuevo nombre
id_country	Mantener	ID_Pais
name	Mantener	Nombre
currency	Mantener	Moneda
conversion_usd	Mantener	Conversion_usd
population_millions	Eliminar (el dato ya se encuentra en la tabla ciudades)	

## 7.2. Relación entre tablas (diagrama EDR)

Para verificar que las tablas cumplan con los campos necesarios para relacionarlos, se realizó el siguiente diagrama EDR.



### 7.3. Caracteres especiales y otras consideraciones

Para asegurar la carga correcta de los datos, primero se estandarizaron los caracteres de los textos. Reemplazando los caracteres que no pertenecen al español al equivalente de nuestra lengua.

name	name
Departamento con vista a la BasÃ¡fica de Guadalupe	Departamento con vista a la Basafica-de Guadalupe
Copacabana   Ã¡fÃ©timo Apto Proximo da Praia e MetrÃ¡	Copacabana - afacoetimo Apto Proximo da Praia e Metrafa
QUARTO PRÃ¡fÃ©XIMO A BARRA DA TIJUCA	QUARTO PRafacoeXIMO A BARRA DA TIJUCA
Cuarto privado en amplia casa cerca BasÃ¡fica Gpe	Cuarto privado en amplia casa cerca Basafsa-lica Gpe
CondomÃ¡nio Praia de Copacabana	Condomafa-nio Praia de Copacabana
Ã¡fÃ© como estar em casa	afac como estar em casa
nice & cool apt. in the best location - Condesa :)	nice & cool apt. in the best location - Condesa
nice and comfortable suite with GYM	nice and comfortable suite with GYM
Cool apt. in classic building. Roma	Cool apt. in classic building. Roma
nice and clean private apt. - Roma	nice and clean private apt. - Roma
comfortable and clean suite in Polanco w/ GYM :)	comfortable and clean suite in Polanco w/ GYM
Nice Apt! (GYM	Nice Apt GYM
Depto extracomodo y totalmente equipado	Depto extracomodo y totalmente equipado

Además, se aseguró que el ID de los hosts sean únicos.

## 7.4. Tablas finales

Finalmente, las tablas quedan con las siguientes estructuras:

Host		
Campo	Tipo de dato	Llaves
Host_id	Identity	PK



Campo	Tipo de dato	Llaves
Fecha_inicio_host	DATE	
Nombre_host	CHAR(260)	
Ratio_respuestas_host	FLOAT	

#### Propiedades

Campo	Tipo de dato	Llaves
Listing_id	Identity	PK
Host_id	INT	FK: Host.Host_id
ID_ciudad	INT	FK: Ciudades.id_ciudad
Precio	INT	

#### Reviews

Campo	Tipo de dato	Llaves
Fecha	DATE	
Review_id	Identity	PK
Huesped_id	INT	
Listing_id	INT	FK: Propiedades.listing_id
Review_puntaje_total	INT	
Review_puntaje_exactituddescripcion	INT	
Review_puntaje_limpieza	INT	
Review_puntaje_valorcliente	INT	
Review_puntaje_comunicacion	INT	
Review_puntaje_ubicacion	INT	

#### Ciudades

Campo	Tipo de dato	Llaves
ID_ciudad	Identity	PK
Nombre	CHAR(30)	
ID_Pais	INT	FK: Países.ID_Pais
Poblacion_millones	FLOAT	

#### Países

Campo	Tipo de dato	Llaves
ID_Pais	Identity	PK
Nombre	CHAR(30)	
Moneda	CHAR(3)	
Conversion_usd	FLOAT	

Las tablas listas para cargar se encuentran en la siguiente carpeta:



[Dataset Limpio](https://drive.google.com/drive/folders/1TxUzcPLlqviG-WHNI80UtcbwRGnvbn74?usp=drive_link)

[https://drive.google.com/drive/folders/1TxUzcPLlqviG-WHNI80UtcbwRGnvbn74?usp=drive\\_link](https://drive.google.com/drive/folders/1TxUzcPLlqviG-WHNI80UtcbwRGnvbn74?usp=drive_link)

## IV. SQL

Se creó una base de datos en SQL con las mismas 5 tablas y se realizó la carga y se aseguró que los tipos de datos sean correctos y no haya ID duplicados.

### 8. Procesos de carga en sql

#### 8.1. Proceso para la creación de la base de datos

La base de datos se creó con el siguiente Query:

```
CREATE DATABASE AnalisisAIRBNB;
```

#### 8.2. Proceso para la creación de tablas

Se crearon las 5 tablas con la misma estructura que dejamos nuestros archivos csv en el capítulo “7.4 Tablas finales”.

Los Queries son los siguientes:

```
USE AnalisisAIRBNB;

--Tabla Host:
CREATE TABLE Host(
Host_ID INT PRIMARY KEY NOT NULL,
Fecha_inicio_host DATE,
Ubicacion_del_host CHAR(50),
Nombre_host CHAR(30),
Ratio_respuestas_host FLOAT
);
ALTER TABLE Host
DROP COLUMN Ubicacion_del_host;

ALTER TABLE Host
ALTER COLUMN Nombre_host CHAR(260);
```

```
--Tabla Reviews:
CREATE TABLE Reviews(
Fecha DATE,
Review_id INT PRIMARY KEY NOT NULL,
Huesped_id INT NOT NULL,
Listing_id INT NOT NULL,
Review_puntaje_total INT,
Review_puntaje_exactituddescripcion INT,
Review_puntaje_limpieza INT,
Review_puntaje_valorcliente INT,
Review_puntaje_comunicacion INT,
Review_puntaje_ubicacion INT
);

--Tabla Ciudades:
CREATE TABLE Ciudades(
ID_ciudad INT PRIMARY KEY NOT NULL,
Nombre CHAR(30) NOT NULL,
ID_Pais INT NOT NULL,
Poblacion_millones FLOAT
);

--Tabla Paises:
CREATE TABLE Paises(
ID_Pais INT PRIMARY KEY NOT NULL,
Nombre CHAR(30) NOT NULL,
Moneda CHAR(3) NOT NULL,
Conversion_usd FLOAT
);
```

### 8.3. Carga de datos y Base de datos final

Desde los archivos csv, con Excel, se crearon los queries para cargar las tablas. Se aseguro que todos los IDs sean valores únicos y fechas de se reemplazaron por celdas vacías.

Todos los queries y la Base de datos final se encuentran en la siguiente carpeta:



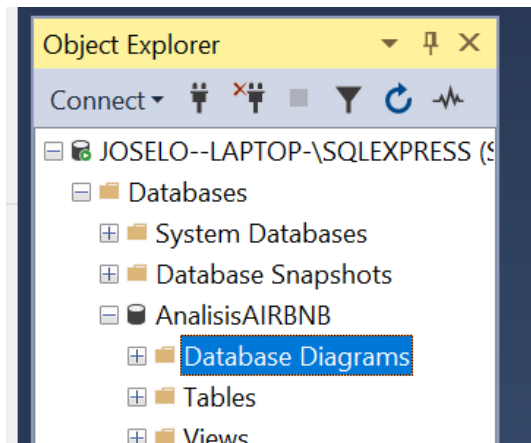
[SQL](https://drive.google.com/drive/folders/1UAP6U8xhMSjbB4XG3aX83cHTlQDekUuZ?usp=drive_link)

[https://drive.google.com/drive/folders/1UAP6U8xhMSjbB4XG3aX83cHTlQDekUuZ?usp=drive\\_link](https://drive.google.com/drive/folders/1UAP6U8xhMSjbB4XG3aX83cHTlQDekUuZ?usp=drive_link)

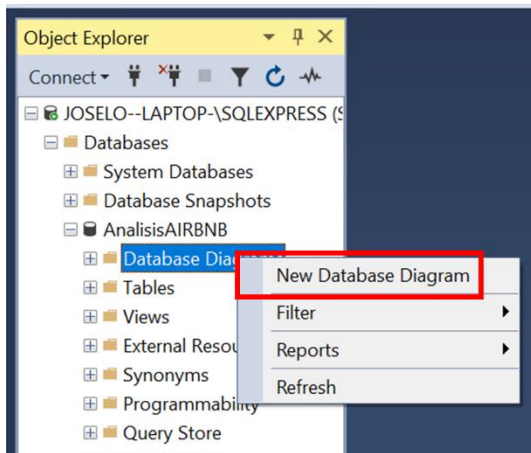
## 9. Relación de tablas en SQL

Respetando la Relación entre tablas (diagrama EDR) detallado más atrás se relacionaron las tablas creadas en SQL.

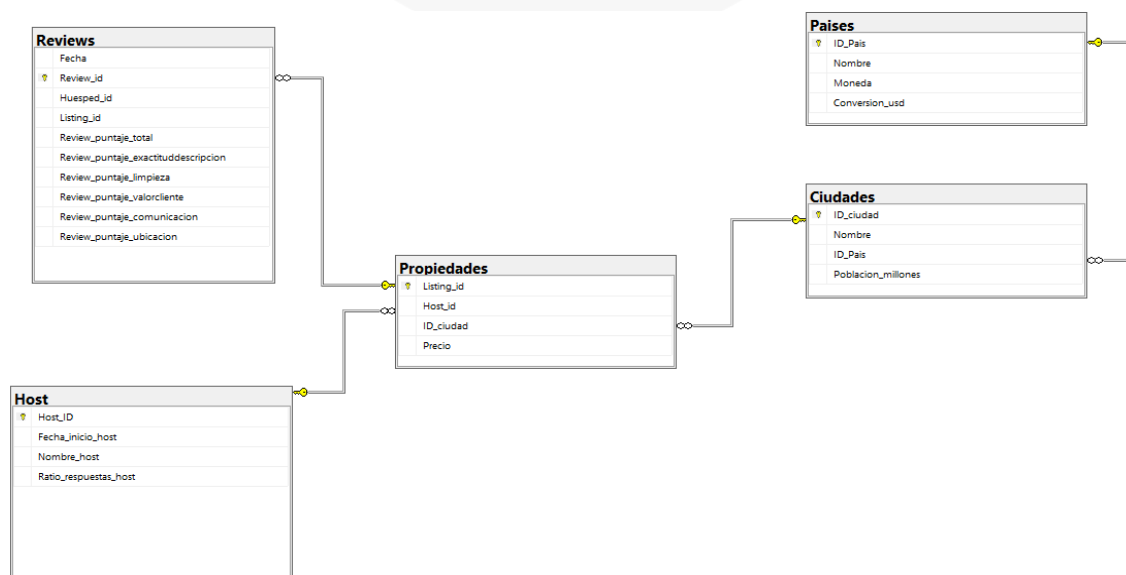
- a. En el “Object Explorer” expando “Databases”, “ analisisAIRBNB”.



- b. Click derecho en “Database Diagrams” y seleccionar “New Database Diagram”

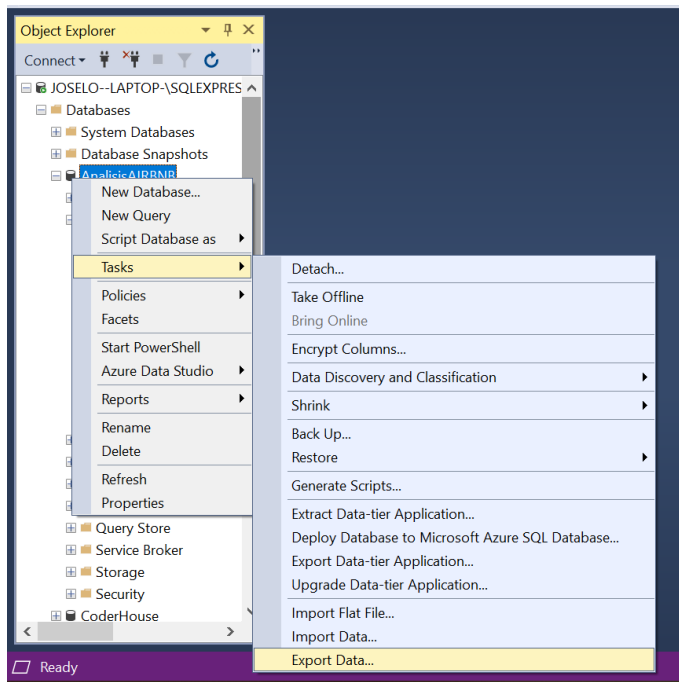


- c. Se seleccionan todas las tablas y click en “Add”.
- d. En cada tabla seleccionar la PK y arrastrar hacia los FK de las demás tablas, quedando las relaciones en SQL como se muestra en la siguiente figura:

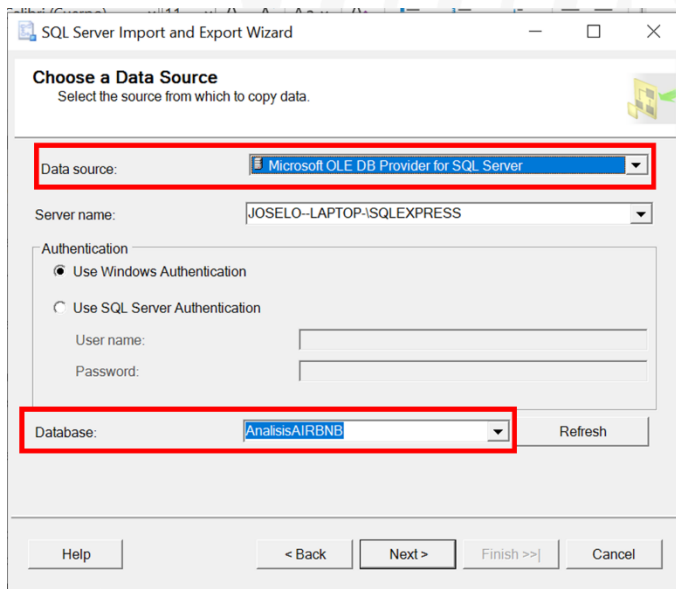


## 10. Exportación de sql a csv

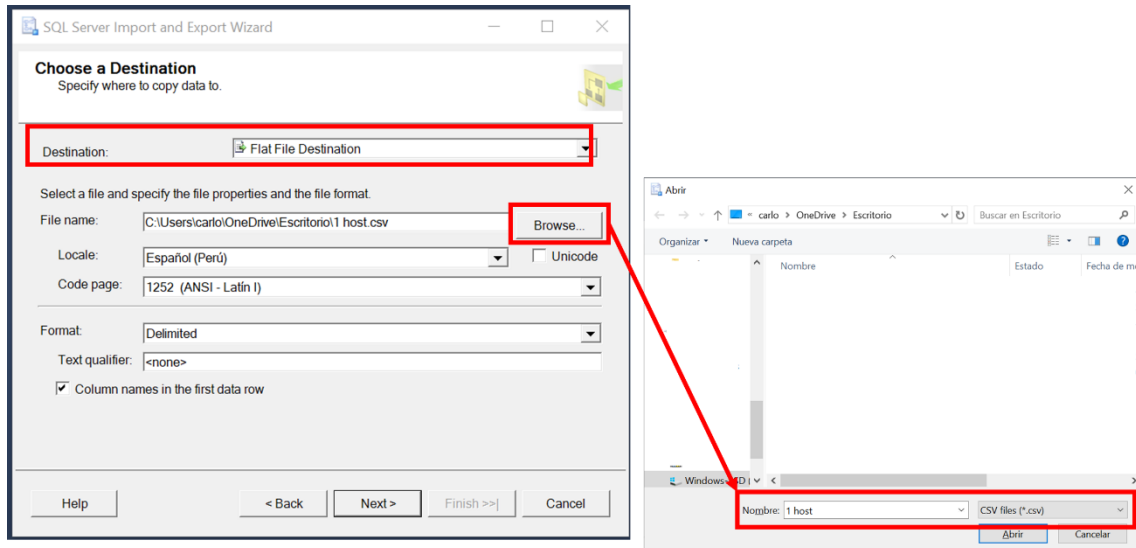
- a. En el Object Explorer se hace click derecho sobre la base de datos deseada.
- b. Se selecciona "Task" y "export data".



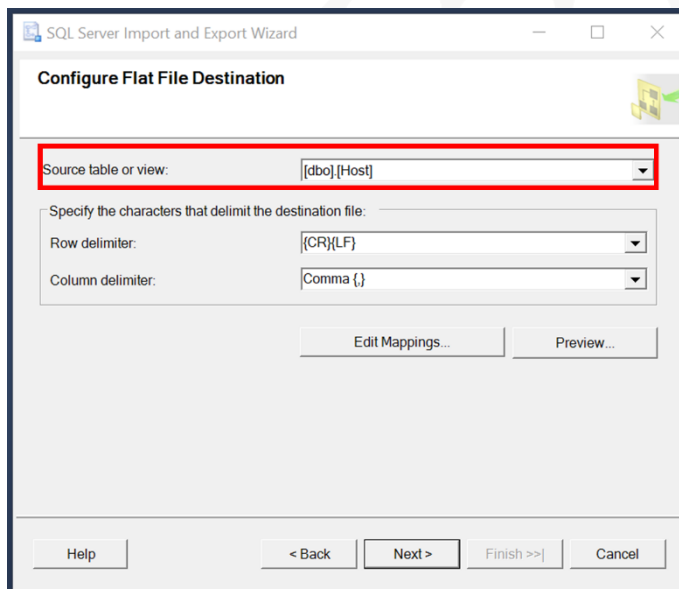
- c. En la ventana emergente se debe poner "next" y a continuación seleccionar en "Data source" "Microsoft OLE DB Provider for SQL Server" y en "Database" la base de datos deseada:



- d. En el siguiente paso seleccionar en "destination" "Flat File Destination" y en File Name seleccionar "Browse", "Archivo csv" y ponerle nombre.



- e. En el siguiente paso seleccionar “Copy data from one or more tables or views”.
- f. Seleccionar la tabla que se desea exportar.



- g. Repetir el proceso con todas las tablas

Los csv exportados desde sql se encuentran en la siguiente carpeta:



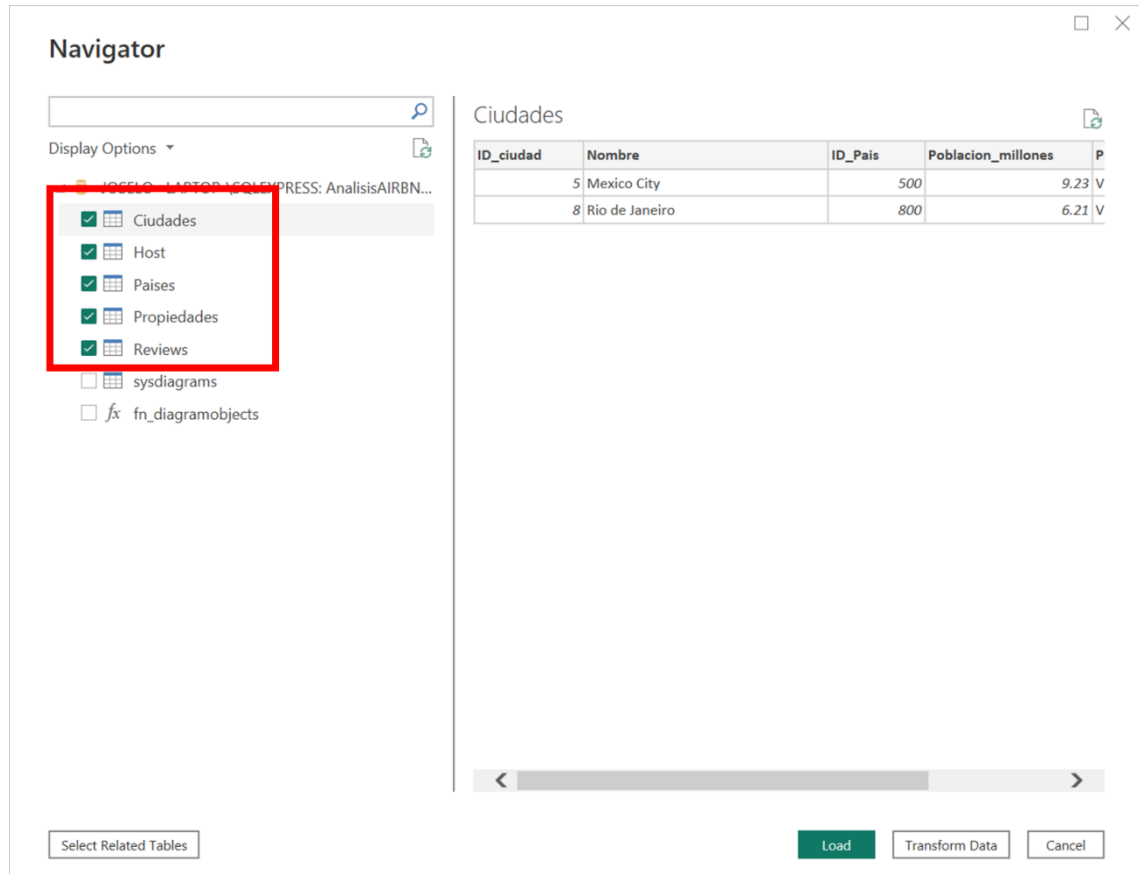
[Archivos csv exportados desde sql](https://drive.google.com/drive/folders/1UBFYKH9_J2-zn4XSAYP_-WeNt9Bb60PH?usp=drive_link)

[https://drive.google.com/drive/folders/1UBFYKH9\\_J2-zn4XSAYP\\_-WeNt9Bb60PH?usp=drive\\_link](https://drive.google.com/drive/folders/1UBFYKH9_J2-zn4XSAYP_-WeNt9Bb60PH?usp=drive_link)

## V. Análisis de datos (Power BI)

### 11. Carga de datos en Power BI

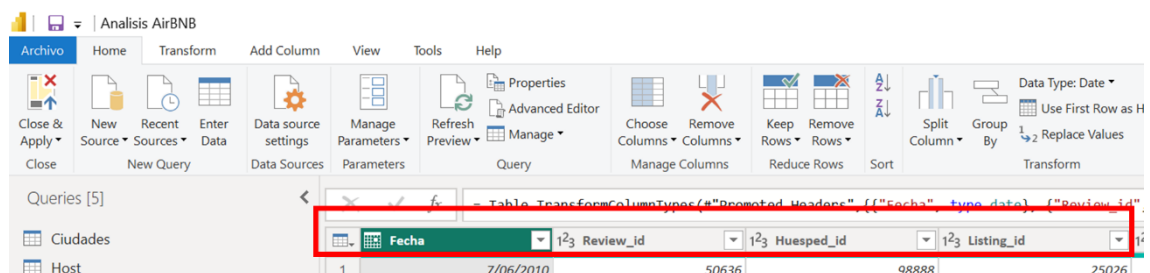
- Se realiza la carga desde SQL Server (con las credenciales de SQL Management).
- Se procede a seleccionar y cargar las tablas. Los diagramas se obvian ya que se relacionarán las tablas más adelante en Power BI.



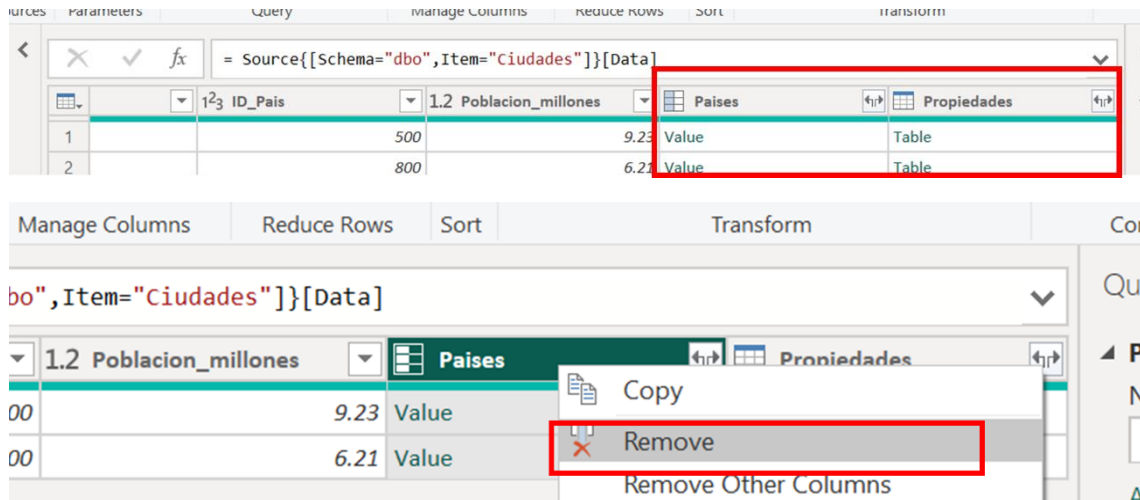
### 12. Transformaciones y limpieza en Power Query

#### 12.1. ETL en Power Query

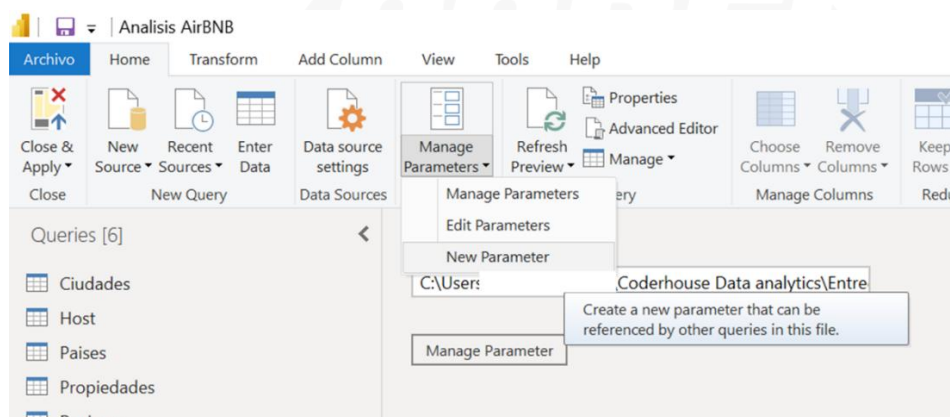
- Tras cargar los datos se ingresó a Power Query.
- Revisamos que los tipos de datos sean correctos (en todas las tablas).



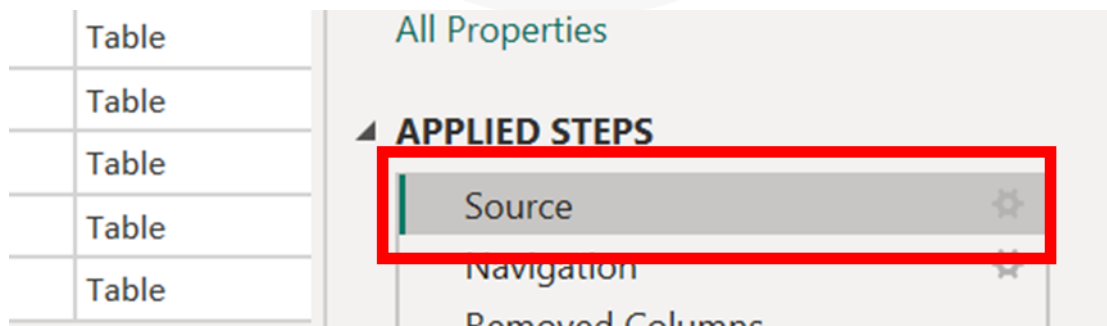
- e. Las tablas relacionadas en SQL se encuentran comprimidas al final de cada una (Similar a combinar tablas). Procederemos a eliminar esas columnas.



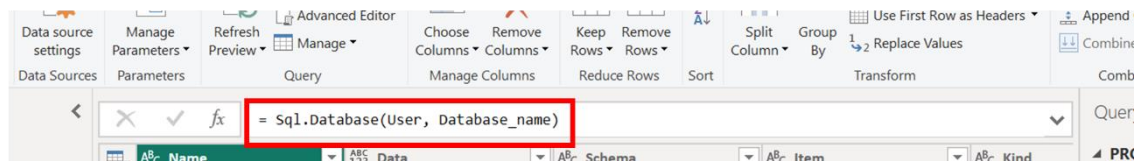
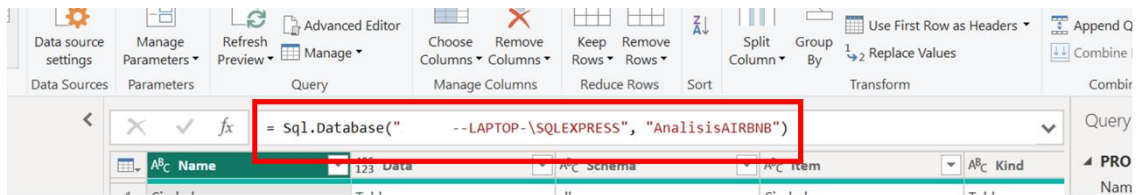
- f. Creamos el parámetro "user" y "database\_name" donde pondremos la credencial de SQL y el nombre de la base de datos respectivamente.



- g. En cada tabla, nos dirigimos al primer paso (Origen) y modificamos las credenciales con los parámetros creados.



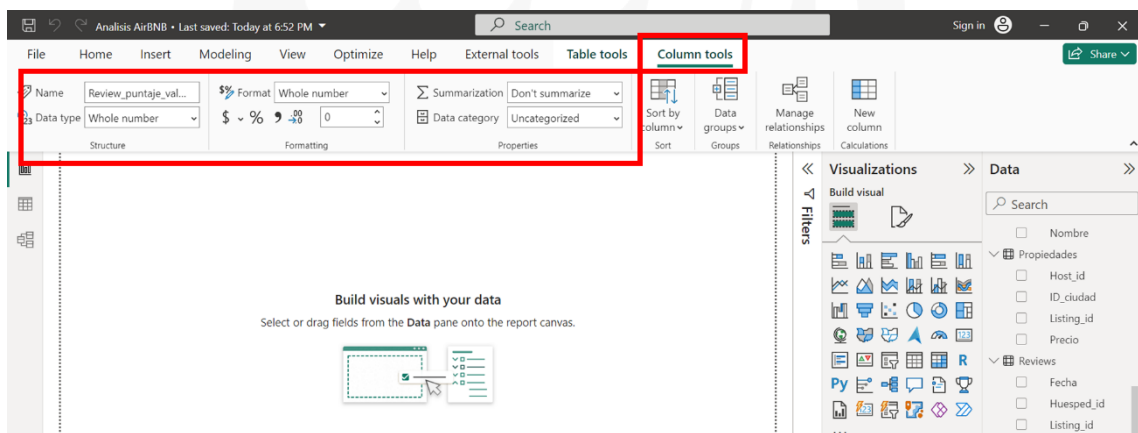




- h. Si se modifica la credencial de SQL (o se cambia de pc) solo se deberá modificar el parámetro “user” con la nueva credencial. De la misma forma, si se modifica el nombre de la base de datos, solo se deberá ingresar el nuevo nombre al parámetro “Database\_name”.

## 12.2. ETL en Power BI

- i. Una vez verificado, en el tablero de Power BI revisamos que todos los campos sean correctos en Power BI.



## 13. Estructura del reporte Power BI

Al estructurar el reporte, priorizare la comprobación de las hipótesis planteadas. Luego se cubrirán otros elementos de interés que surgen durante el análisis.

Las hipótesis se formularon en el capítulo 5 Hipótesis / Objetivos.

Recapitulando, las hipótesis son:

- “Entre diciembre a febrero (Feriados y vacaciones de verano) hay mayor cantidad de reviews por la disponibilidad de las personas.”
- “Porcentualmente hablando, las propiedades de mayor costo tienen un mejor promedio de calificaciones de clientes.”

Entonces, como mínimo debo analizar los reviews (incluyendo las cantidades por mes), y relacionar precio con satisfacción del cliente.

Por ende, el reporte se dividirá en:

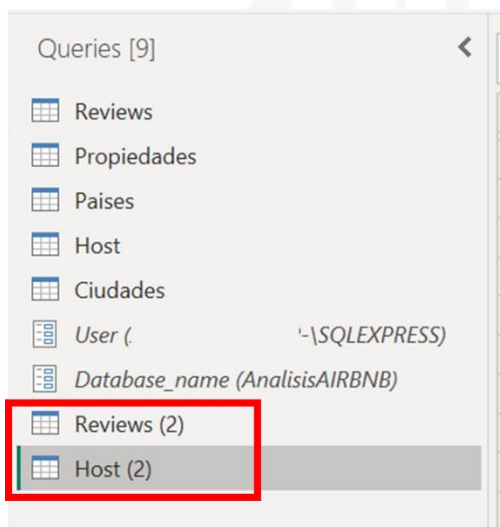
- Portada.
- Análisis de reviews.
- Análisis de satisfacción.
- Análisis de precios.

## 14. Creación de tabla calendario

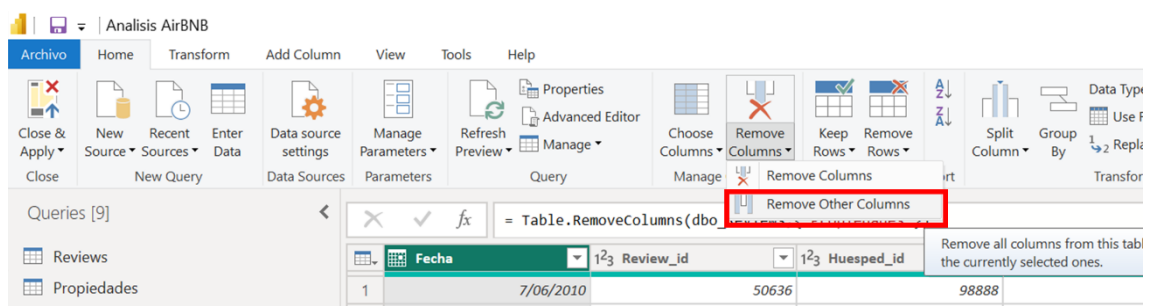
### 14.1. En Power Query (tabla calendario)

Las únicas tablas con fechas son “Reviews” y “Host”.

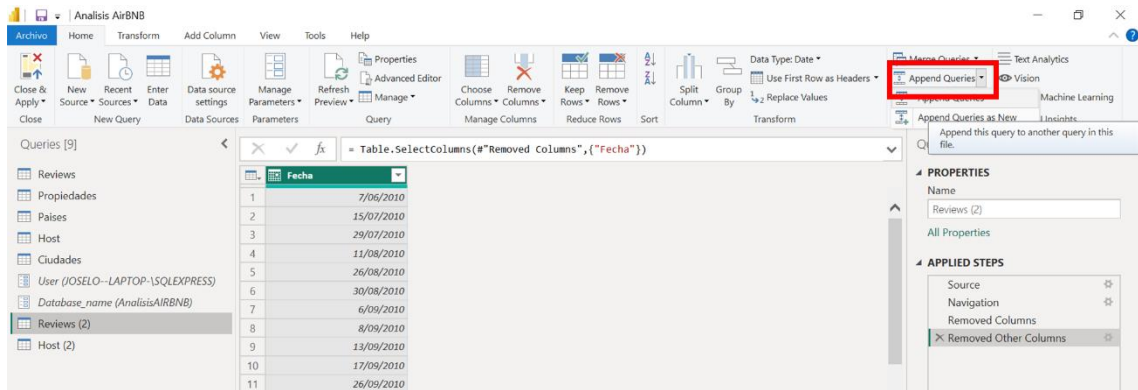
- Primero duplicare ambas tablas (a la tabla, click derecho, duplicar).



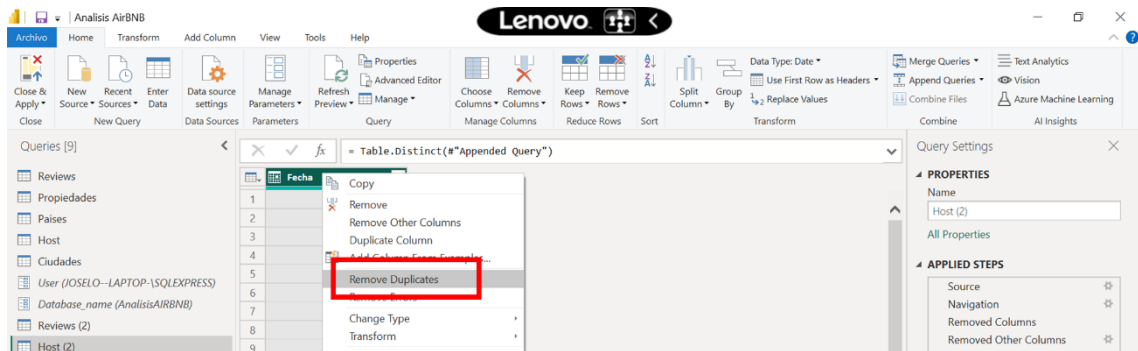
- Eliminaré todos los campos de las tablas duplicadas, excepto la columna de fecha.



- c. Anexare ambas tablas (para eso ambos campos deben tener el mismo nombre, de forma exacta)



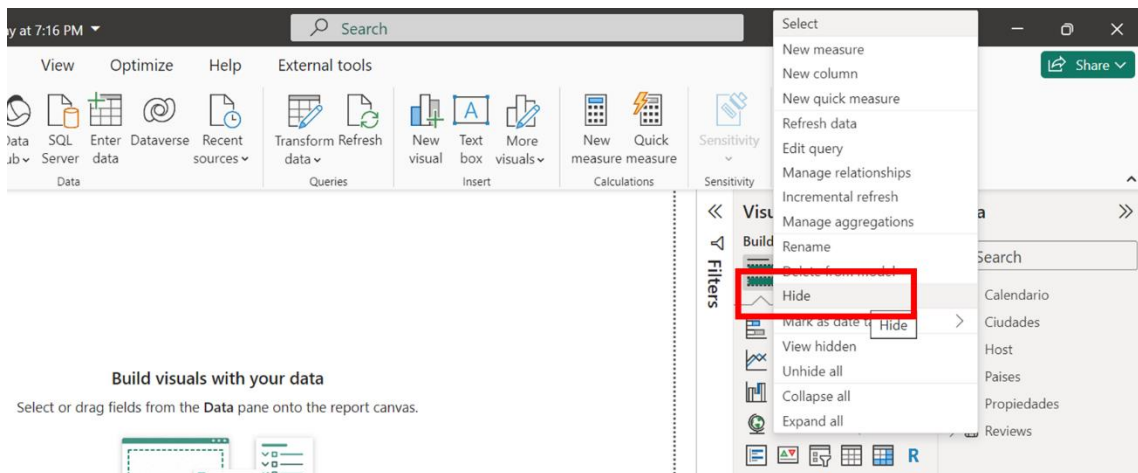
- d. Eliminare duplicados; quedándome con las fechas necesarias.



- e. Ordenar las fechas de forma ascendente.  
f. Renombrar la tabla como calendario.

#### 14.2. En Power BI (tabla calendario)

- g. En Power BI, primero, eliminar del modelo la otra tabla duplicada para evitar confusiones.



- h. Finalmente, en la vista de tabla, añadiré al calendario las siguientes columnas de mayor detalle con DAX:

Año = YEAR(Calendario[Fecha])

Mes = MONTH(Calendario[Fecha])

Trimestre = QUARTER(Calendario[Fecha])

Día de la semana \_Número = WEEKDAY(Calendario[Fecha],2)

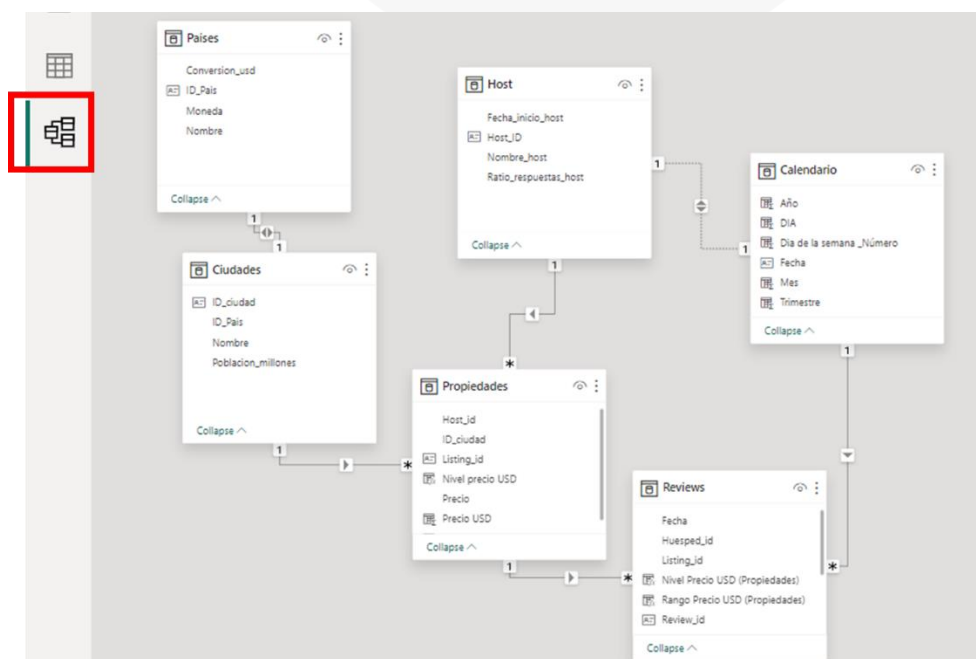
DIA = DAY(Calendario[Fecha])

Fecha	Año	Mes	Trimestre	Día de la semana _Número	DIA
lunes, 13 de octubre de 2008	2008	10	4	1	13
martes, 3 de febrero de 2009	2009	2	1	2	3
domingo, 29 de marzo de 2009	2009	3	1	7	29
jueves, 28 de mayo de 2009	2009	5	2	4	28
martes, 14 de julio de 2009	2009	7	3	2	14
domingo, 23 de agosto de 2009	2009	8	3	7	23
miércoles, 26 de agosto de 2009	2009	8	3	3	26

## 15. Relaciones y tablas puentes

En la vista de relación de tablas se arrastran las PK hacia los FK de las demás tablas, respetando el EDR detallado en el capítulo “7.2 Relación entre tablas (diagrama EDR)”.

A dicha relación se le debe añadir la tabla calendario creado para el análisis, quedando de la siguiente manera:



**Nota:** La relación de la tabla calendario a Host esta como relación inactiva. La principal es Calendario a Reviews, puesto que es uno de los objetivos a analizar.

A continuación, se crearán más tablas complementarias las cuales también en el diagrama.

### 15.1. Tablas complementarias de Calendario

En Power Query, se crean las siguientes tablas:

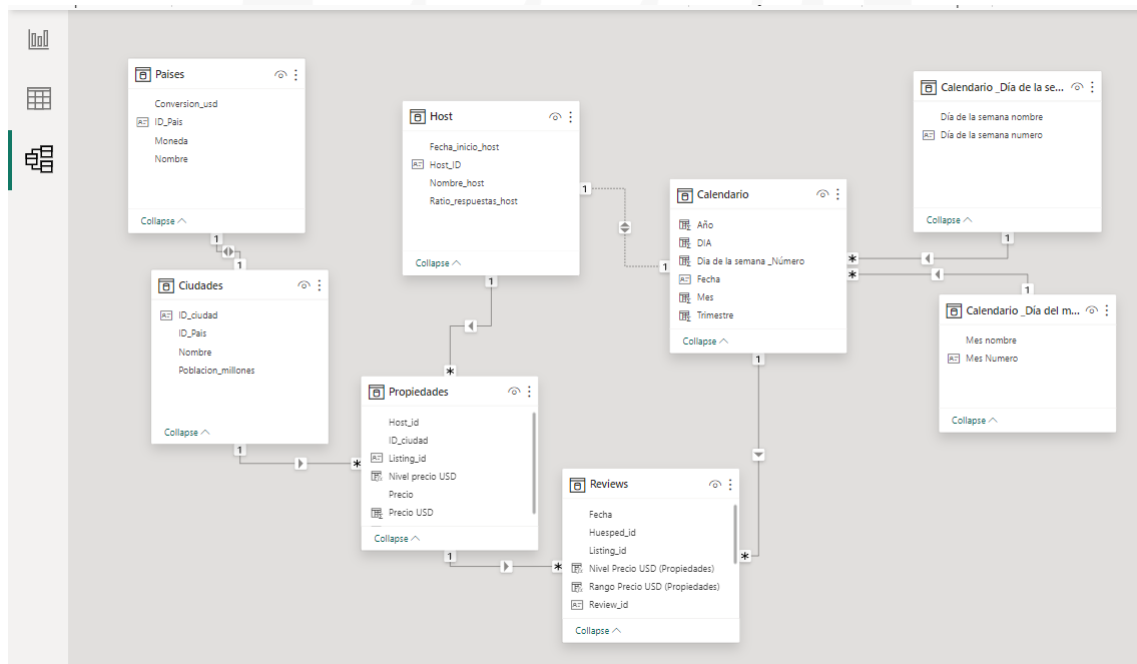
123 Mes Numero	A8C Mes nombre
1	1 Enero
2	2 Febrero
3	3 Marzo
4	4 Abril
5	5 Mayo
6	6 Junio
7	7 Julio
8	8 Agosto
9	9 Setiembre
10	10 Octubre
11	11 Noviembre
12	12 Diciembre

*Calendario Mes con nombre*

123 Día de la semana numero	A8C Día de la semana nombre
1	1 Lunes
2	2 Martes
3	3 Miércoles
4	4 Jueves
5	5 Viernes
6	6 Sábado
7	7 Domingo

*Calendario Día de la semana con nombre*

Las cuales se relacionan con Calendario (mes número y día de la semana número) para brindar mayor detalle.



### 15.2. Complementando categorías de Puntajes en Reviews

#### *Categorías en Puntaje total*

El puntaje total va de 0 a 100. Entonces, en Power Query se duplico la tabla reviews, se eliminaron todas las columnas excepto puntaje total, y se creó la siguiente columna condicional:

## Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name

Nivel de satisfacción

	Column Name	Operator	Value		Output
If	Review_puntaje_t...	is greater than or...	90	Then	Muy bueno
Else If	Review_puntaje_t...	is greater than or...	70	Then	Bueno
Else If	Review_puntaje_t...	is greater than or...	50	Then	Regular
Else If	Review_puntaje_t...	is greater than or...	30	Then	Malo

Add Clause

Else

Muy malo

OK

Cancel

### Categorías en Otros Puntajes

Los demás puntajes van de 0 a 10. Entonces siguiendo los niveles de satisfacción creados en puntaje total, se creará una tabla en Power Query las mismas categorías de la siguiente manera:

	Review_puntaje_categoria	Nivel de satisfacción_categoria
1	0	Muy malo
2	1	Muy malo
3	2	Muy malo
4	3	Malo
5	4	Malo
6	5	Regular
7	6	Regular
8	7	Bueno
9	8	Bueno
10	9	Muy bueno
11	10	Muy bueno

Si el puntaje se multiplica por 10, seguirá la misma categoría de Puntaje total.

### Tabla para ordenar las categorías de los puntajes

Con la finalidad de que las categorías de los puntajes tengan una lógica, se crea la siguiente tabla:

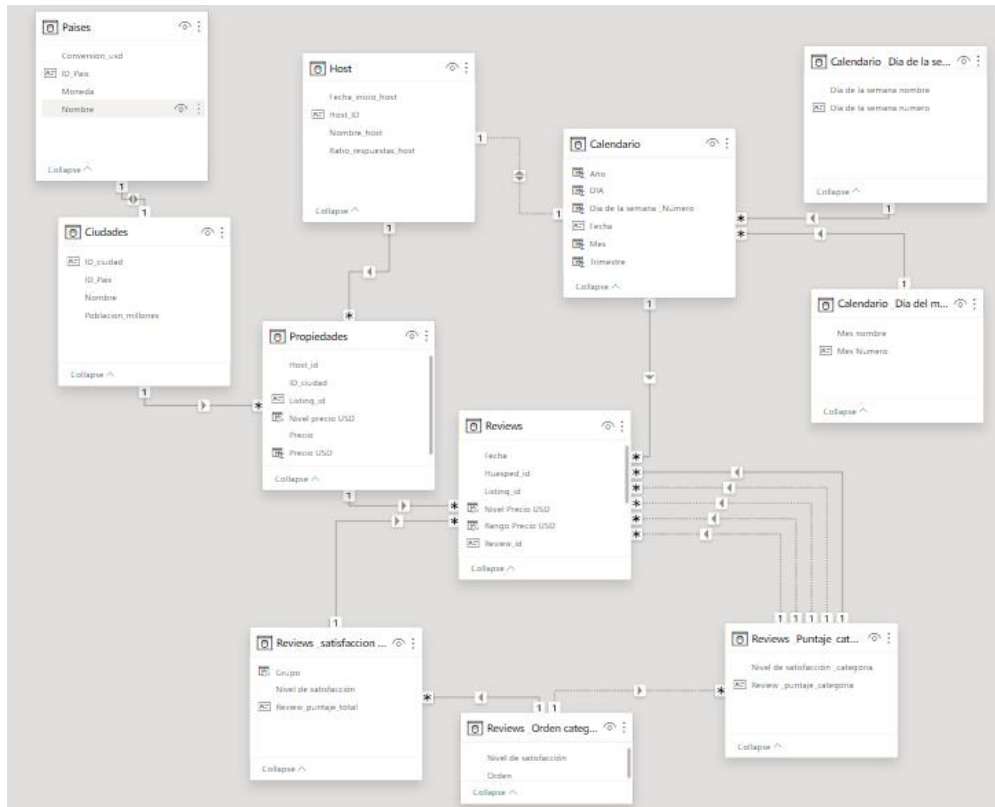
Orden	Nivel de satisfacción
-1	Muy malo
-2	Malo
-3	Regular
-4	Bueno
-5	Muy bueno

Se hace el **Ordenamiento de categorías de Satisfacción** (detallado más adelante en el título 18.2).

Y finalmente, se conecta con las tablas "Reviews \_Puntaje\_categorías" y "Reviews \_satisfaccion total". Todas las Primary key son "Nivel de satisfacción".

### 15.3. Relaciones con tablas complementarias

Finalmente, las relaciones quedan de la siguiente manera:



## 16. DAX Columnas Calculadas

### 16.1. Tabla Propiedades

#### Precio en USD

Estandariza los precios a dólares americanos. Los pasos son los siguientes:

- Valor\_precio: Con las relaciones copia el valor de conversión desde la tabla Países a Propiedades y se multiplica por el Precio local
- Condición: Si el Valor\_precio es mayor a 40000 se considera 0. De esta forma eliminamos los datos anómalos que no dejan hacer un buen análisis.
- Blancos: Convertimos en celdas en Blanco los valores igual a cero. De esta forma se hacen las estadísticas con aquellos que solo tienen valores numéricos.

En el Dataset original ya habían valores igual a 0, a ello se le suman los valores mayores de 40000 por el paso "condición"

- Resultado = blancos.

Precio USD =

```
VAR Valor_precio = RELATED(Paises[Conversion_usd]) * Propiedades[Precio]
VAR condicion = IF(Valor_precio > 40000, 0, Valor_precio)
VAR Blancos = IF(condicion=0,BLANK(),condicion)
RETURN Blancos
```

### Rango de precio en USD

Crea rangos a partir del precio en USD para poder hacer un análisis correcto.

- a. Distancia\_rango: Determina el rango de cada intervalo.
- b. Multiplo\_rango: Redondea el Precio USD en Distancia\_rango.
- c. Rango Inferior:
  - a. Si Multiplo\_rango es menor a Precio USD, entonces se tomará ese multiplo + 1 centavo de dólar.
  - b. De lo contrario, ese multiplo se restará Distancia\_rango para ir al rango inferior y se sumara el centavo de dólar.
- d. Rango Superior:
  - a. Si Multiplo\_rango es mayor o igual al Precio USD, se tomará ese multiplo como el rango superior.
  - b. De lo contrario se le sumara la distancia rango.
- e. Concatenado: Concatena el rango inferior, un guion y el rango superior.
- f. Resultado = concatenado.

```
Rango Precio USD =  
VAR Distancia_rango = 40  
VAR Multiplo_rango = MROUND(Propiedades[Precio USD], Distancia_rango)  
VAR Rango_inferior = IF(Multiplo_rango < Propiedades[Precio USD],  
Multiplo_rango + 0.01, Multiplo_rango - Distancia_rango + 0.01)  
VAR Rango_superior = IF (Multiplo_rango >= Propiedades[Precio USD], Multiplo_rango,  
Multiplo_rango + Distancia_rango)  
VAR Concatenado = Rango_inferior & " - " & Rango_superior  
VAR Resultado = IF(Propiedades[Precio USD]=BLANK(), BLANK(), Concatenado)  
RETURN Resultado
```

### Nivel de precio en USD

Divide el precio en 5 categorías.

- a. Precio USD: Es el precio en USD.
- b. Precio\_maximo USD: Es el mayor precio de toda la tabla en USD.
- c. Rango1: Son los precios que llegan máximo a la 5 parte del Precio\_maximo.
- d. Rango2: Son los precios que llegan máximo a los 2/5 del Precio\_maximo.
- e. Rango3: Son los precios que llegan máximo a los 3/5 del Precio\_maximo.
- f. Rango4: Son los precios que llegan máximo a los 4/5 del Precio\_maximo.
- g. Nivel: es la condicional donde el rango1 equivale a "Muy barato", de lo contrario, si no supera el rango2 equivale a "Barato", siguiendo la lógica, si no supera rango3 es "Precio regular", si no supera rango4 es "Precio elevado", y lo demás son "Precio muy elevado".
- h. Resultado = nivel.



```
Nivel precio USD =
VAR Precio_USD = Propiedades[Precio USD]
VAR Precio_maximo = MAX(Propiedades[Precio USD])
VAR Rango1 = Precio_maximo/5
VAR Rango2 = Precio_maximo/5*2
VAR Rango3 = Precio_maximo/5*3
VAR Rango4 = Precio_maximo/5*4
VAR Nivel = IF ( Precio_USD <= Rango1 , "Muy barato",
                IF ( Precio_USD <= Rango2 , "Barato",
                    IF ( Precio_USD <= Rango3 , "Precio Regular",
                        IF ( Precio_USD <= Rango4 , "Precio elevado",
                            "Precio muy elevado" ) ) ) ) )
RETURN Nivel
```

Listing_id	Host_id	ID_ciudad	Precio	Precio USD	Rango Precio USD	Nivel precio USD
101182	530471	8	200	40	0.01 - 40	Muy barato
145966	706303	8	200	40	0.01 - 40	Muy barato
242214	530471	8	200	40	0.01 - 40	Muy barato
286792	1429181	8	200	40	0.01 - 40	Muy barato
324324	1659441	8	200	40	0.01 - 40	Muy barato

## 16.2. Tabla Reviews

### Rango de precio en USD

Copia los Rangos de Precio en USD de la Tabla Propiedades a la tabla Reviews.

```
Rango Precio USD (Propiedades) =
LOOKUPVALUE(
    Propiedades[Rango Precio USD],
    Propiedades[Listing_id],
    Reviews[Listing_id])
```

### Nivel de precio en USD

Copia los Niveles (categorías) de Precio en USD de la Tabla Propiedades a la tabla Reviews.

```
Nivel Precio USD (Propiedades) =
LOOKUPVALUE(
    Propiedades[Nivel precio USD],
    Propiedades[Listing_id],
    Reviews[Listing_id])
```

Review_puntaje_valorcliente	Review_puntaje_comunicacion	Review_puntaje_ubicacion	Rango Precio USD (Propiedades)	Nivel Precio USD (Propiedades)
10	10	10	40.01 - 80	Muy barato
10	10	10	40.01 - 80	Muy barato
10	10	10	40.01 - 80	Muy barato
10	10	10	40.01 - 80	Muy barato
10	10	10	40.01 - 80	Muy barato
10	10	10	40.01 - 80	Muy barato
10	10	10	40.01 - 80	Muy barato

### 16.3. Tabla Reviews \_satisfaccion total

Columna calculada sobre la tabla creada en el capítulo 15.2 Complementando categorías de Puntajes en Reviews.

#### Grupo Puntaje

Agrupa los puntajes totales en grupos de 10.

```

Grupo Puntaje =
VAR Puntos = 'Reviews _satisfaccion total'[Review_puntaje_total]
VAR Grupos =
    IF(Puntos <= 10, "0 - 10",
    IF(Puntos <= 20, "11 - 20",
    IF(Puntos <= 30, "21 - 130",
    IF(Puntos <= 40, "31 - 40",
    IF(Puntos <= 50, "41 - 50",
    IF(Puntos <= 60, "51 - 60",
    IF(Puntos <= 70, "61 - 70",
    IF(Puntos <= 80, "71 - 80",
    IF(Puntos <= 90, "81 - 90",
    "91 - 100"
    )))))))
RETURN Grupos
    
```

Review_puntaje_total	Nivel de satisfacción	Grupo Puntaje
0	Muy malo	0 - 10
20	Muy malo	11 - 20
30	Malo	21 - 130
35	Malo	31 - 40
40	Malo	31 - 40
45	Malo	41 - 50
47	Malo	41 - 50
48	Malo	41 - 50
50	Regular	41 - 50
52	Regular	51 - 60
53	Regular	51 - 60
55	Regular	51 - 60

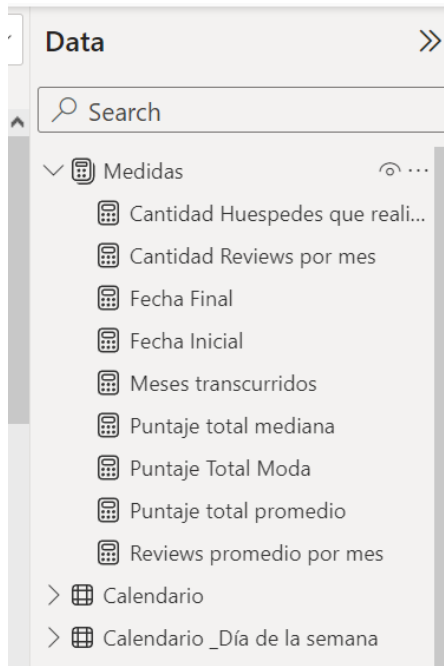
## 17. DAX Medidas calculadas

### 17.1. Tabla medidas

Se creo la tabla “Medidas”, donde se ordenarán todas las medidas que creemos. Para ello se realizaron los siguientes pasos:

- Crear una tabla vacía llamada “Medidas”.
- Crear la primera medida en dicha tabla.
- Borrar la columna vacía dejando solo la medida.

Ello situara la tabla en la parte superior de nuestro tablero. Las demás medidas se deben crear en esta tabla, si se crea en una tabla aparte, se puede arrastrar desde la vista de Modelo.



## 17.2. Medidas

### Fecha inicial

Es la menor fecha del calendario, sirve para verificar los filtros y poder crear nuevas medidas con mayor facilidad.

```
Fecha Inicial = MIN(Calendario[Fecha])
```

### Fecha final

Es la mayor fecha del calendario, sirve para verificar los filtros y poder crear nuevas medidas con mayor facilidad.

```
Fecha Final = MAX(Calendario[Fecha])
```

### Cantidad Huespedes que realizaron Reviews por mes

Calcula cuantos huéspedes (con diferente ID) realizaron Reviews por mes.

```
Cantidad Huespedes que realizaron Reviews por mes =  
CALCULATE(DISTINCTCOUNT(Reviews[Huesped_id]),Calendario[Mes])
```

### Cantidad de Reviews por mes

Calcula cuantos Reviews se realizaron por mes.

```
Cantidad Reviews por mes =  
CALCULATE(DISTINCTCOUNT(Reviews[Review_id]),Calendario[Mes])
```

#### Meses transcurridos

Calcula la cantidad de meses transcurridos desde la fecha inicial a la fecha final.

```
Meses transcurridos = DATEDIFF([Fecha Inicial], [Fecha Final], MONTH)
```

#### Puntaje total mediana

Es la mediana del Puntaje total.

```
Puntaje total mediana = MEDIAN(Reviews[Review_puntaje_total])
```

#### Puntaje total Promedio

Es el promedio del puntaje total

```
Puntaje total promedio = AVERAGE(Reviews[Review_puntaje_total])
```

#### Reviews promedio por mes

Es la cantidad promedio de reviews que realizaron mensualmente desde la fecha de inicio hasta la fecha final.

```
Reviews promedio por mes =  
VAR nro_reviews_total = DISTINCTCOUNT(Reviews[Review_id])  
VAR division = DIVIDE(nro_reviews_total, [Meses transcurridos])  
VAR resultado = IF(division=BLANK(),nro_reviews_total,division)  
RETURN resultado
```

#### Puntaje total moda

Es la moda mensual del puntaje total. El proceso es el siguiente:

- Filtro\_no\_blanks: Filtra la tabla de reviews con todos los puntajes que tengan algún valor. Los puntajes que están en blanco (null) son obviados.
- Tabla\_resumen: Crea internamente una tabla resumen de Filtro\_no\_blanks, con las columnas de puntaje total y el conteo de cada puntaje.
- Maximo: Es el valor máximo de conteo de la tabla\_resumen.
- Moda: Calcula el promedio de las modas (puntajes con mayor cantidad de conteo).
- Resultado = Cantidad.

```
Puntaje Total Moda =
VAR Filtro_no_blanks = FILTER(Reviews, Reviews[Review_puntaje_total]<>BLANK())
VAR conteo = SUMMARIZE(
    Filtro_no_blanks,
    [Review_puntaje_total],
    "total",
    COUNT(Reviews[Review_puntaje_total])
)
VAR maximo = MAXX(conteo, [total])

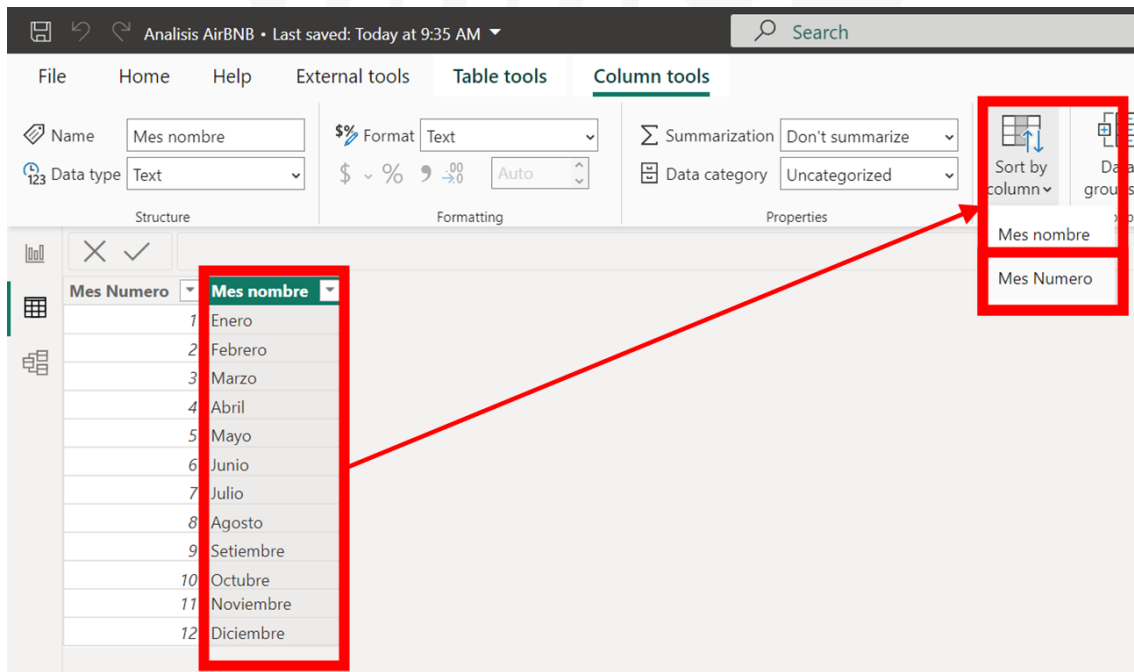
VAR moda =
    CALCULATE(
        AVERAGE(Reviews[Review_puntaje_total]),
        FILTER(conteo, [total] = maximo)
    )

RETURN moda
```

## 18. Otros preparativos

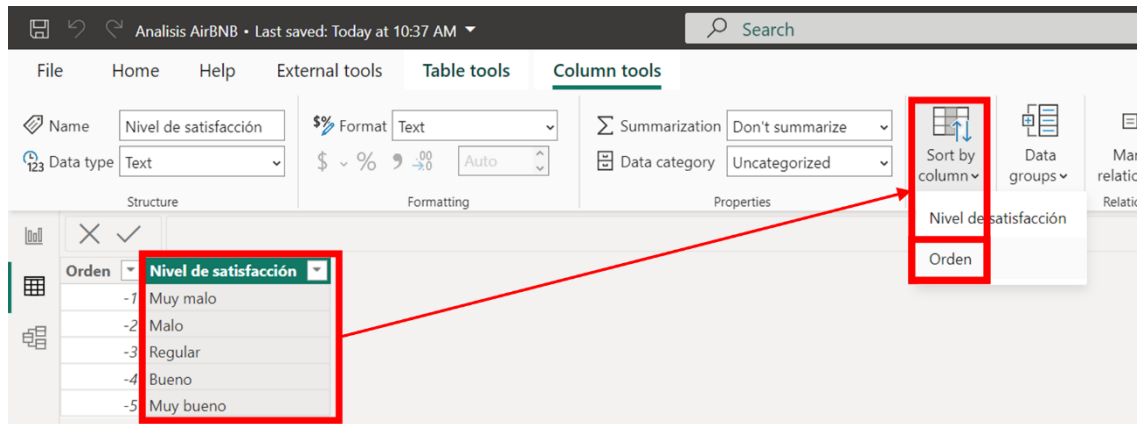
### 18.1. Ordenamiento de meses y días en tablas complementarias calendario

- Se selecciona la columna la columna que se desea ordenar (mes nombre o día de la semana nombre, dependiendo de la tabla)
- En la pestaña herramienta de columnas se selecciona “ordenar por” y la categoría numérica del mes o día de la semana.



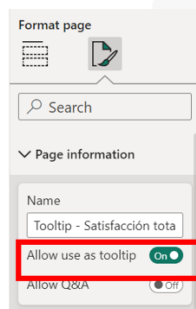
## 18.2. Ordenamiento de categorías de Satisfacción

Se sigue el mismo proceso de Ordenamiento de meses y días en tablas complementarias calendario con la tabla "Reviews \_Orden categorías satisfacción"

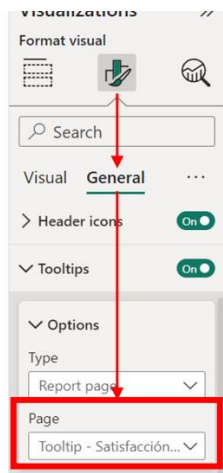


## 18.3. Tooltips

- Se crea la página para el Tooltip.
- En Configuración de Página, Información de Página se habilita "Permitir usar como tooltip".



- Seleccionar el objeto visual al que se desee añadir ese Tooltip.
- En Formato de objeto visual, General, Tooltips, seleccionar la página que se creó en el paso a.



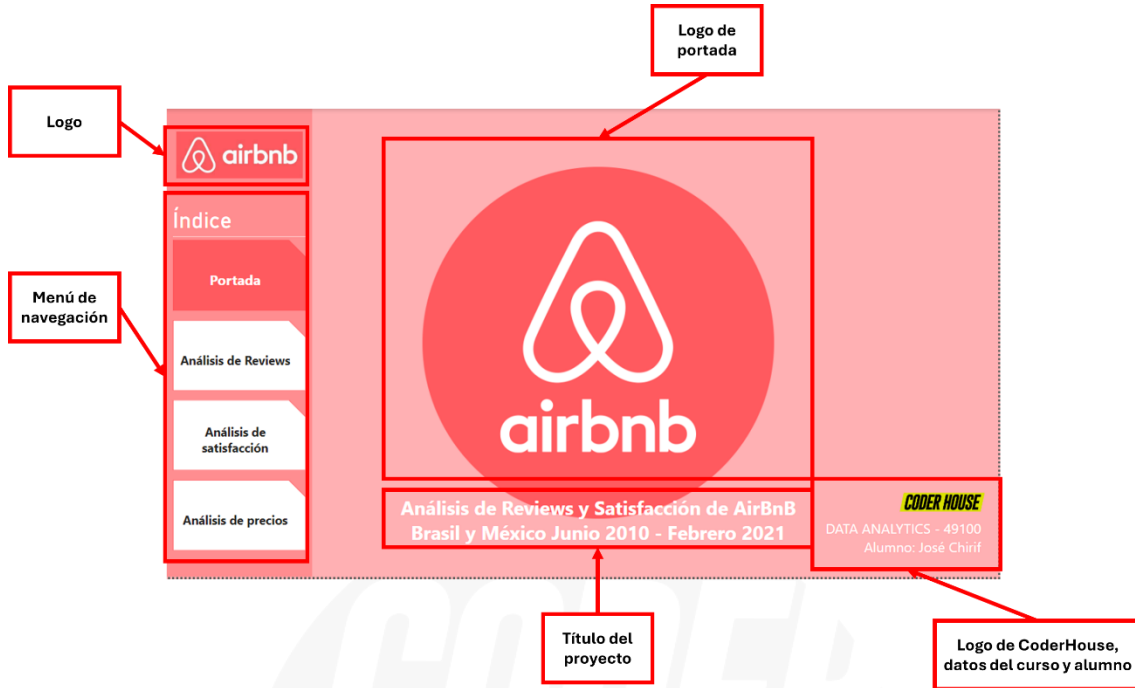
## 19. Estructura de las solapas

Las solapas tendrán la siguiente estructura:

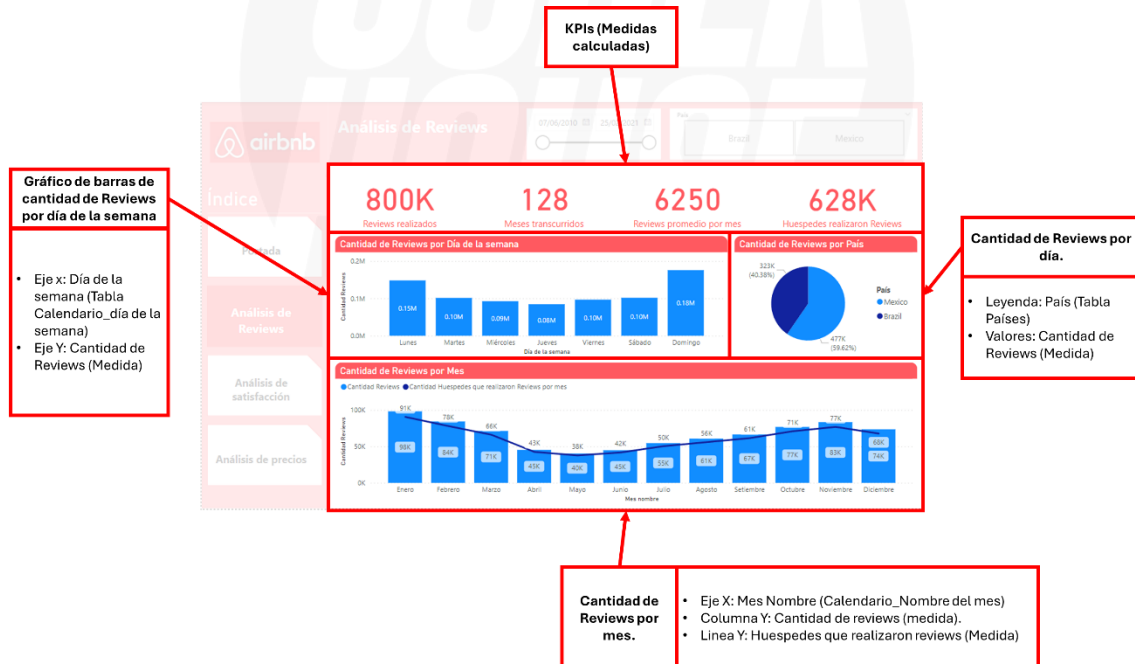


## 20. Reporte

### 20.1. Portada



### 20.2. Análisis de Reviews



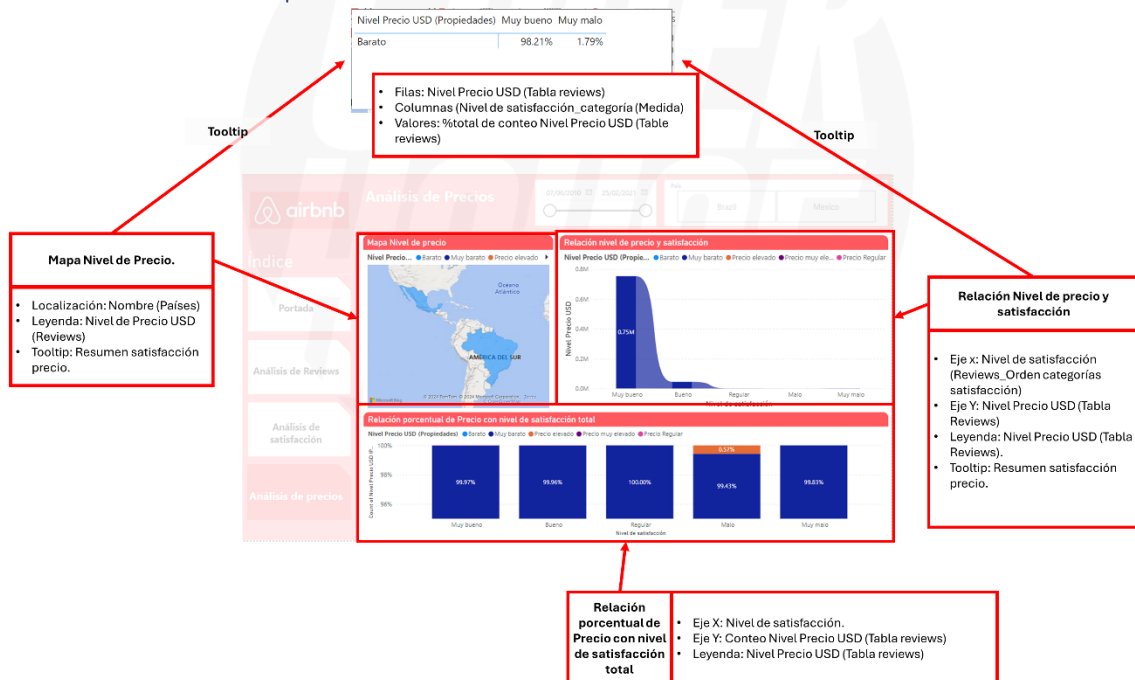
**Nota:** A esta solapa se le aplicó el filtro de Fecha es posterior al 6 de junio del 2010, puesto que no existes Reviews de fechas anteriores a esa.



### 20.3. Análisis de Satisfacción



### 20.4. Análisis de precios



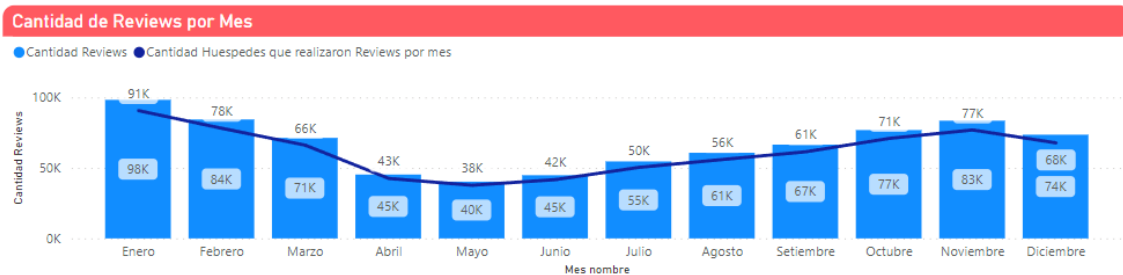
## VI. Conclusiones y recomendaciones

### 21. Confirmación de hipótesis

#### 21.1. Hipótesis 1

La hipótesis indica “Entre diciembre a febrero (Feriados y vacaciones de verano) hay mayor cantidad de reviews por la disponibilidad de las personas”.

En la solapa “Análisis de Reviews” tenemos el siguiente gráfico.



En este gráfico se puede observar en enero empieza la tendencia a decaer la cantidad de reviews y en mayo a crecer paulatinamente. Sin embargo, de noviembre a diciembre hay una pequeña caída en cantidad de reviews.

Por tanto, es **hipótesis falsa**. Los meses con mayor cantidad de reviews son enero, febrero y noviembre respectivamente.

#### 21.2. Hipótesis 2

La hipótesis indica “Porcentualmente hablando, las propiedades de mayor costo tienen un mejor promedio de calificaciones de clientes”.

En la solapa “Análisis de Precios” tenemos la siguiente gráfica.



En ella podemos observar que independientemente del precio (la mayoría de los alquileres son “muy baratos”), los usuarios tienen calificaciones mixtas de las propiedades. Por ende, es **hipótesis falsa**.

## 22. Conclusiones y recomendaciones

- a. Existe la estacionalidad (de cantidad de reviews) al inicio de año el pico más alto, de ahí descendiendo hasta mayo y empieza a crecer gradualmente hasta noviembre donde hay un pequeño decrecimiento a diciembre. La cantidad de reviews puede estar sujeto a cantidad de reservas.

**Recomendación:** Alinear las estrategias de marketing con la estacionalidad de reservas. Recomiendo realizar proyecciones, lo que sería motivo de otro estudio más a fondo, puesto que no se encuentra en los alcances del proyecto.

- b. Si bien es cierto, los niveles de satisfacción de cliente se distribuyen a cualquier precio, se hizo notable la relación de “Precio elevado” con “Satisfacción mala”.

**Recomendación:** Verificar si los precios de ese grupo de propiedades están de acorde a lo que ofrecen, y posteriormente velar por mejorar la experiencia del usuario final independientemente del precio.

