



*Instituto Politécnico Nacional*

*Escuela Superior de Cómputo*

*Grupo 3CV19 Data Mining*

*Profesor Zagal Flores Roberto Eswart*

*2do Parcial Practica #6: Diseño estructural del datawarehouse y desarrollo del ETL del proyecto semestral*

*Alumno García González Aarón Antonio*

*Miércoles 19 de mayo de 2021*



# Análisis de algunos factores de competitividad para AirBnB en CDMX

## Objetivo:

Dados los alcances de la unidad de aprendizaje, se busca aplicar la mayor cantidad de conocimientos adquiridos a lo largo del curso, centrado en datos abiertos de AirBnB y relacionarlo con datos demográficos, territoriales y de transporte en CDMX, donde al final del ejercicio se busca conocer que dimensiones tienen mayor peso o importancia en el éxito o fracaso de una oferta de AirBnB.

## Entendimiento del negocio:

Se busca conocer si el número de estaciones de metro o metrobús por delegación es una medida que hace más atractivo a un AirBnB, que servicio de transporte metro o metrobús es más atractivo en relación con un AirBnB, si el número y la categoría de hoteles por delegación condiciona la presencia de AirBnB en el mercado, se busca conocer las delegaciones que más exitosas son respecto a AirBnB, si el número de noches mínimas a alquilar el servicio es un factor importante, el precio de acuerdo a la delegación, que tipo de renta de AirBnB es más comercial, a manera superficial si el número de reviews es una dimensión que fortalece la competitividad, etc.

Se toman datos de 3 fuentes diferentes:

1. AirBnB  
Datos por AirBnB ofertado en CDMX, la temática de este data set menciona los detalles por AirBnB que son útiles para un usuario que decide o compara entre varios de estos, datos de resumen en el periodo que mide el ejercicio mensual y anual.
2. Inegi  
Datos recopilados sobre población y territorio seccionado por delegación en CDMX.
3. Portal de datos abiertos de CDMX
  - Datos de hotelería basado en dimensión espacial y por categorizada por estrellas en hoteles de CDMX.
  - Datos de transporte metro y metrobús basado en dimensión espacial por delegación de CDMX.

## Entendimiento de los datos:

1. AirBnB

La fuente principal de datos se obtiene de <http://insideairbnb.com/get-the-data.html>, donde Las dimensiones temáticas que incluye son la delegación, las coordenadas geográficas, el tipo de oferta, el precio, el mínimo de noches a rentar, el número de calificaciones, la fecha de la última calificación o reseña, el número de veces que se ha rentado en el mes, el total de días que se ofrece al año el servicio, así como el mes y año, hay un atributo el cual es el nombre del oferente, dado que un oferente puede ofertar más de un airbnb pues igual podría ser llamado dimensión.

Campo	Tipo	Calculado	Descripción
id	integer		Identificador único de Airbnb para el anuncio
name	string		
host_id	integer		
host_name	string		
neighbourhood_group	text	y	El grupo de vecindarios como geocodificado usando la latitud y longitud contra los vecindarios definidos por shapefiles digitales abiertos o públicos.
neighbourhood	text	y	El vecindario geocodificado utilizando la latitud y la longitud frente a los vecindarios definidos por shapefiles digitales públicos o abiertos, referido a alcaldías o delegaciones.
latitude	numeric		Utiliza la proyección del Sistema Geodésico Mundial (WGS84) para latitud y longitud.
longitude			Utiliza la proyección del Sistema Geodésico Mundial (WGS84) para latitud y

			longitud.
room_type	string		
price	currency		precio diario en moneda local. Tenga en cuenta que el signo \$ se puede utilizar a pesar de la configuración regional
minimum_nights	integer		cantidad mínima de estadía por noche para el anuncio (las reglas del calendario pueden ser diferentes)
number_of_reviews	integer		El número de reseñas que tiene la ficha
number_of_reviews_ltm	integer	y	El número de reseñas que tiene el anuncio (en los últimos 12 meses)
last_review	date	y	La fecha de la última / más reciente revisión
calculated_host_listings_count	integer	y	El número de listados que tiene el anfitrión en el scrape actual, en la geografía de la ciudad / región.
availability_365	integer	y	availability_x. La disponibilidad del listado x días en el futuro según lo determine el calendario. Tenga en cuenta que un anuncio puede estar disponible porque ha sido reservado por un invitado o bloqueado por el anfitrión.

Tabla 1. Metadatos AirBnB

Los datos disponibles en este sitio se encuentran por mes, donde vamos a descargar cada mes de interés y posterior a ello a unirlos.

## 2. Inegi

De igual manera se realiza un catalogo de delegaciones el cual incluye el id de delegación, el nombre, el numero de habitantes en 2020, la superficie en kilómetros cuadrados por delegación y la densidad de población en habitantes por kilometro cuadrado, este fue realizado a mano.

Campo	Tipo	Calculado	Descripción
id	integer		Identificador único de delegación.
nombre	string		Nombre de la delegación.
habitantes	integer		Número de habitantes por delegación en 2020.
Superficie (KM^2)	integer		Superficie por delegación en kilómetros cuadrados.
Densidad Hab/KM^2	float	y	Densidad de habitantes por kilometro cuadrado.

Tabla 2. Metadatos Delegación

## 3. Portal de datos abiertos de CDMX

Fuente de datos complementarios de hoteles en ciudad de México, el cual incluye id de hotel, tipo, nombre, categoría (por estrellas) y la alcaldía donde radica, de igual manera las estaciones de metro y metrobús en CDMX, que incluyen id de estación, nombre, línea y delegación donde se encuentra.

Se descargan los archivos CSV del sitio, donde se va a desechar lo no necesario y complementar con datos oficiales.

Campo	Tipo	Calculado	Descripción
id	integer		Identificador único de hotel.
tipo	string		Nombre de la delegación.
nombre	string		Nombre del hotel.
Categoría	integer		Categoría en estrellas del 1 al 5.
calle_y_numero	string		Calle y numero donde radica el hotel.

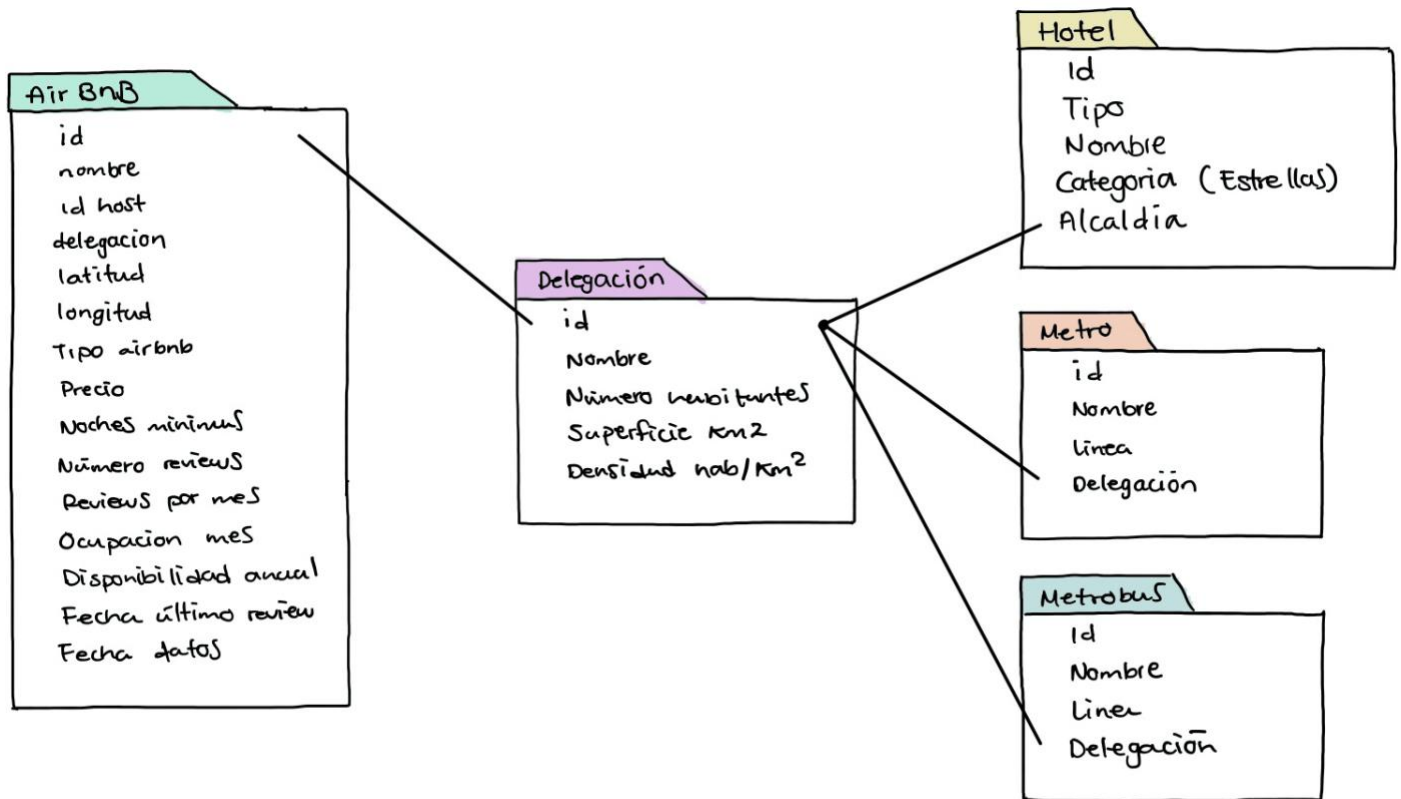
colonia	string		Colonia donde radica el hotel.
cp	integer		Código postal.
Alcaldía	string		Nombre de la delegación.

Tabla 3. Metadatos Hoteles

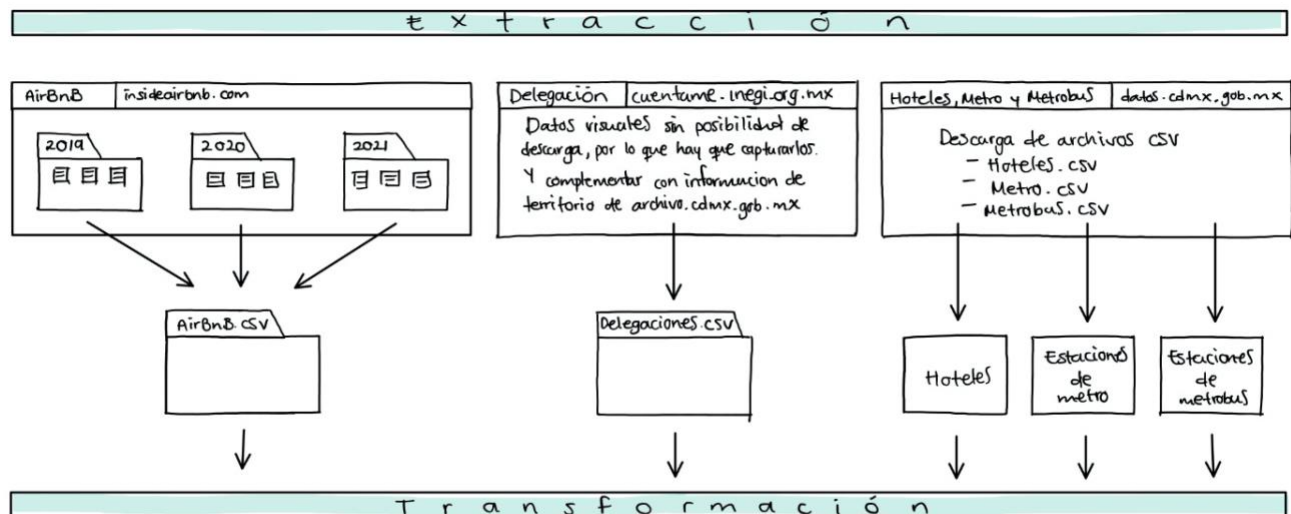
Campo	Tipo	Calculado	Descripción
id	integer		
Geometry	string		
stop_id	integer		
stop_code	string		
stop_name	string		
stop_desc	string		
stop_lat	integer		
stop_lon	integer		
trip_heads	string		
agency_id	string		
geopoint	string		

Tabla 4. Metadatos Metro y metrobus

Estructura de tabla de hechos y el modelado conceptual del data lake.



La tabla de hechos nos referimos a AirBnB, donde se complementa de dimensiones territoriales y de población por delegación o alcaldía, mediante la tabla “delegación” que relaciona las tablas “hotel”, “Metro” y “Metrobus”, mismas que serán las estaciones y hoteles dentro de la misma delegación que cada servicio AirBnB.



→ Pellenar con fecha comodín a los registros que jamás han sido evaluados.

→ Separar la fecha de último Review en día, mes y año, donde únicamente mes y año serán columnas, posterior a ello eliminamos la fecha de review entera del data set.

→ Obtener a partir de cada archivo.csv la fecha de dichos datos, donde el mes y año se agregan como columnas al data set.

→ Eliminar el campo "neighbourhood-group" del dataset

→ Cambiar el nombre de delegación o alcaldía por el id de alcaldía con base al archivo delegaciones.csv.

→ Eliminar registros que incluyan uno o más campos vacíos o no válidos

→ Combinar datos de ambas fuentes de tal manera que podamos identificar a cada delegación y además conocer su nombre, número de habitantes, extensión superficial en kilómetros cuadrados y densidad de población.

→ Para hoteles

→ Eliminar columnas: `calle_y_num`, `colonia` y `CP`

→ Cambiar el nombre de delegación o alcaldía por el id de alcaldía con base al archivo Delegaciones.csv.

→ Para estaciones de metro y metrobus

→ Eliminar columnas: `geometry`, `stop_id`, `stop_code`, `stop_desc`, `stop_lust`, `stoplon`, `trip_heads`, `agency` y `geopoint`.

→ Agregar columna "delegacion", la cual se llena con el id de delegación donde se encuentra la estación, esto se busca estación a estación en la página del servicio.

## C a r g a

Todos los archivos resultantes en el paso de transformación, resultan archivos .csv que son cargados en SQL Server para la minería siguiente.

### Script de limpieza y transformación para datos de AirBnB

```
# Script for AirBnB data cleaning

import os
import csv
import datetime
import pandas as pd
import numpy as np

# This function recives a file name like "DD_MM_YYYY.csv"
# Return the datetime DD/MM/YYYY
def getDateFromFileName(file_name):
    date_file = pd.to_datetime(file_name.split('.')[0].replace('_', '/'))
    return date_file

def exchangeNeighbourhoodByIdNeighbourhood(dictionary, value):
    return dictionary[value]

# Main function
def main():
    start_time = datetime.datetime.now()

    path = '../DataSet/AIRBNB/'
    target_directory = '../DataSet/TARGET/'
    target_file = "airbnb.csv"
    delegaciones_file = '../DataSet/CDMX/delegaciones.csv'
    dataframes = []

    # Delete target file if this exists
    if os.path.exists(target_directory + target_file):
        os.remove(target_directory + target_file)

    # Read delegaciones directory
    data = pd.read_csv(delegaciones_file, header=0)
    d_f = pd.DataFrame(data)
    d_f = d_f.dropna()

    dictionary = dict(zip(d_f["Alcaldía"].to_list(), d_f["id"].to_list()))

    for directory in os.listdir(path):
        print("\nDirectorio " + directory + " :")

        for name_file in os.listdir(path + '/' + directory):
            print("\tProcesando " + name_file + " ...")

            date_file = getDateFromFileName(name_file)

            # Getting original data of ---
            datos = pd.read_csv(path + '/' + directory + '/' + name_file, header=0)
            df = pd.DataFrame(datos)

            # Completing empty values --- comodin 01012000
            df['reviews_per_month'] = df['reviews_per_month'].fillna(value=0)
```

```

        df['last_review'] =
df['last_review'].fillna(value=pd.to_datetime('01/01/2000'))
        day_last_review = pd.to_datetime(df['last_review']).dt.day
        month_last_review = pd.to_datetime(df['last_review']).dt.month
        year_last_review = pd.to_datetime(df['last_review']).dt.year

        # Pulling away last review date in day, month and year ---
        # df = df.assign(day_last_review=day_last_review)
        df = df.assign(month_last_review=month_last_review)
        df = df.assign(year_last_review=year_last_review)

        # Adding month and year of data file ---
        # df = df.assign(date_data=date_file)
        df = df.assign(month_data=date_file.month)
        df = df.assign(year_data=date_file.year)

        # Removing last_review, host_name and neighbourhood_group columns
        df = df.drop(['last_review'], axis=1)
        df = df.drop(['host_name'], axis=1)
        df = df.drop(['neighbourhood_group'], axis=1)

        # Exchange delegacion by delegacion id
        df["neighbourhood"] = df["neighbourhood"].apply((lambda x:
exchangeNeighbourhoodByIdNeighbourhood(dictionary, x)))

        # Adding dataframe to dataframes list
        dataframes.append(df)

# Joining all the dataframes
join = pd.concat(dataframes)

# Removing nan rows
join.dropna(how="all", inplace=True)

# updating index id
join['id'] = [x for x in range(1, len(join['id']) + 1)]

# saving the new csv file
join.to_csv(target_directory + target_file, index=False)

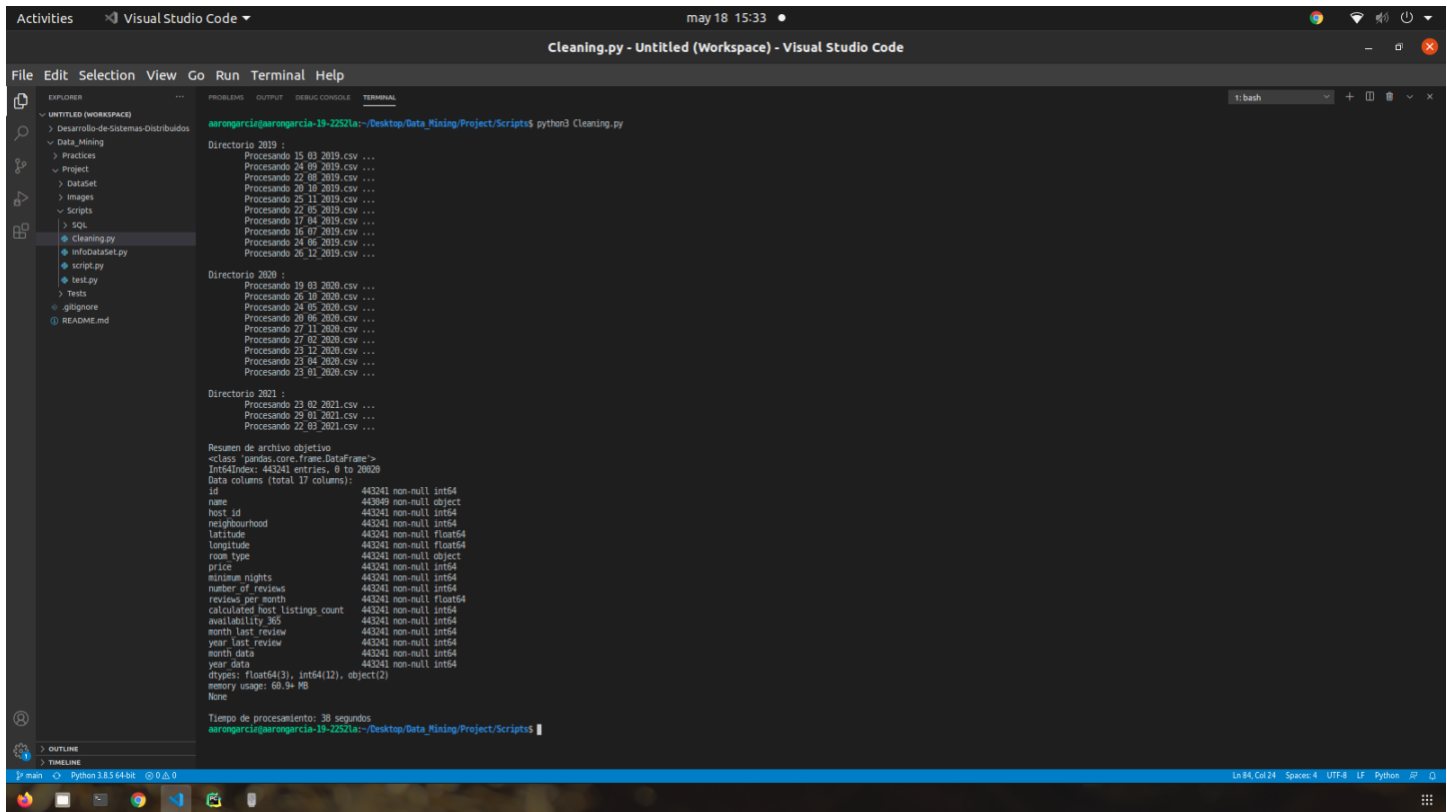
# Swowing data info
print("\nResumen de archivo objetivo")
print(join.info())

end_time = datetime.datetime.now()
diference = end_time - start_time
print("\nTiempo de procesamiento: " + str(diference.seconds) + " segundos")

if __name__ == "__main__":
    main()

```





## Script de limpieza y transformación para datos de hoteles

```

import os
import csv
import datetime
import pandas as pd
import numpy as np
import itertools

def exchangeNeighbourhoodByIdNeighbourhood(dictionary, value):
    return dictionary[value]

def getCategoryOk(category):
    aux = category.split()

    if(category == 'Sin Clasificar'):
        return 0
    else:
        return aux[0]

def main():
    start_time = datetime.datetime.now()

```

```

path = '../DataSet/CDMX/'
target_directory = '../DataSet/TARGET/'
hotels = 'hoteles.csv'
delegaciones = "delegaciones.csv"
target_file = "hotels.csv"

# Delete target file if this exists
if os.path.exists(target_directory + target_file):
    os.remove(target_directory + target_file)

datos_1 = pd.read_csv(path + delegaciones, header=0)
df_1 = pd.DataFrame(datos_1)
df_1 = df_1.dropna()

dictionary = dict(zip(df_1["Alcaldía"].to_list(), df_1["id"].to_list()))

datos_2 = pd.read_csv(path + hotels, header=0)
df_2 = pd.DataFrame(datos_2)
df_2 = df_2.dropna()

# Exchanging delegacion by delegacion id
print("Sustituyendo nombre de delegacion por id de delegacion ...")
df_2['alcaldia'] = df_2['alcaldia'].apply((lambda x:
exchangeNeighbourhoodByIdNeighbourhood(dictionary, x)))

# Removing last_review, host_name and neighbourhood_group columns
print("Eliminando dimensiones: calle_y_numero, cp, colonia ...")
df_2 = df_2.drop(['calle_y_numero'], axis=1)
df_2 = df_2.drop(['colonia'], axis=1)
df_2 = df_2.drop(['cp'], axis=1)

# Setting category format
# Exchange delegacion by delegacion id
df_2["categoria"] = df_2["categoria"].apply((lambda x: getCategoryOk(x)))

# saving the new csv file
print("Obteniendo formato de categoria correcto por estrellas ...")
df_2.to_csv(target_directory + target_file, index=False)

# Showing data info
print("\nResumen de archivo objetivo")
print(df_2.info())

end_time = datetime.datetime.now()
diference = end_time - start_time
print("\nTiempo de procesamiento: " + str(diference.seconds) + " segundos")

if __name__ == "__main__":
    main()

```

Activities Visual Studio Code may 18 16:14

Hotel.py - Untitled (Workspace) - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER

- UNTITLED (WORKSPACE)
- Desarrollo-de-Sistemas-Distribuidos
- Data\_Mining
  - Practices
  - Project
    - DataSet
    - Images
    - Scripts
      - SQL
      - AirBnB.py
      - Hotel.py
      - InfoDataSet.py
      - test.py
      - Tests
    - .gitignore
    - README.md

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: bash

```
aarongarcia@aarongarcia-19-22521a:~/Desktop/Data_Mining/Project/Scripts$ python3 Hotel.py
Sustituyendo nombre de delegacion por id de delegacion ...
Eliminando dimensiones: calle y numero, cp, colonia ...
Obteniendo formato de categoria correcto por estrellas ...

Resumen de archivo objetivo
<class 'pandas.core.frame.DataFrame'>
Int64Index: 625 entries, 0 to 628
Data columns (total 5 columns):
id                625 non-null int64
tipo              625 non-null object
nombre            625 non-null object
categoria         625 non-null object
alcaldia          625 non-null int64
dtypes: int64(2), object(3)
memory usage: 29.3+ KB
None

Tiempo de procesamiento: 0 segundos
aarongarcia@aarongarcia-19-22521a:~/Desktop/Data_Mining/Project/Scripts$
```

Ln 18, Col 21 Spaces: 4 UTF-8 LF Python

## Archivos resultantes de la transformación



airbnb.csv



hotels.csv



metro.csv



metrobus.csv



delegaciones.csv

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
name	nvarchar(512)	<input checked="" type="checkbox"/>
host_id	int	<input type="checkbox"/>
neighbourhood	int	<input type="checkbox"/>
latitude	float	<input type="checkbox"/>
longitude	float	<input type="checkbox"/>
room_type	nvarchar(256)	<input type="checkbox"/>
price	int	<input type="checkbox"/>
minimum_nights	int	<input type="checkbox"/>
number_of_reviews	int	<input type="checkbox"/>
reviews_per_month	float	<input type="checkbox"/>
calculated_host_listings_...	int	<input type="checkbox"/>
availability_365	int	<input type="checkbox"/>
month_last_review	int	<input type="checkbox"/>
year_last_review	int	<input type="checkbox"/>
month_data	int	<input type="checkbox"/>
year_data	int	<input type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
Alcaldia	nvarchar(128)	<input type="checkbox"/>
Población	int	<input type="checkbox"/>
Superficie_km2	float	<input type="checkbox"/>
Densidad_hab_km2	nvarchar(128)	<input type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
tipo	int	<input type="checkbox"/>
nombre	nvarchar(512)	<input checked="" type="checkbox"/>
categoria	int	<input type="checkbox"/>
alcaldia	int	<input type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
nombre	nvarchar(128)	<input type="checkbox"/>
linea	varchar(50)	<input checked="" type="checkbox"/>
delegación	int	<input type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
nombre	nvarchar(128)	<input type="checkbox"/>
linea	int	<input type="checkbox"/>
delegacion	int	<input type="checkbox"/>

A continuación, se muestran datos de los datos cargados por cada tabla:

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the 'airbnb' database structure, including tables like 'airbnb', 'delegaciones', 'hotels', 'metro', and 'metrobus'. The right pane shows a SQL query executed in the 'airbnb' database, which selects the top 1000 rows from the 'airbnb' table. The query result is displayed in a table with 14 columns: id, name, host\_id, neighbourhood, latitude, longitude, room\_type, price, minimum\_nights, number\_of\_reviews, reviews\_per\_month, calculated\_host\_listings\_count, availability\_365, month\_last\_review, year\_last\_review, month\_data, and year\_data. The first 14 rows of data are shown.

id	name	host_id	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings_count	availability_365	month_last_review	year_last_review	month_data	year_data
1	Private room c/ bathroom. Colonia Juárez.	57785	6	19.43035	-99.15511	Private room	483	2	0	0	1					
2	Sunny suite w/ queen size bed, inside boutique B&B	87973	6	19.44076	-99.16324	Private room	1990	1	6	0.06	9					
3	Couple of Rooms	145672	14	19.27215	-99.21848	Private room	1758	1	0	0	1					
4	Villa Dante	153786	5	19.38399	-99.27335	Entire home/apt	3863	1	0	0	2					
5	CONDESA HAUS BOUTIQUE B&B	196253	6	19.41006	-99.17645	Private room	1893	1	38	0.42	5					
6	Great space in historical San Rafael	265650	6	19.43937	-99.15614	Entire home/apt	715	4	56	0.58	2					
7	Entire beautiful duplex in la Roma	282620	6	19.42292	-99.15775	Entire home/apt	1642	3	37	0.42	1					
8	Spacious Clean Quiet room (own bath) in la Condesa	299558	6	19.41259	-99.17959	Private room	966	2	48	2.88	3					
9	2 bedroom apt. deco bldg, Condesa	334451	6	19.41375	-99.17028	Entire home/apt	1835	2	39	0.44	3					
10	Beautiful light Studio Coyoacan-full equipped !	212109	4	19.35601	-99.16167	Entire home/apt	1062	3	90	1.04	4					
11	Great studio Coyoacan!	212109	4	19.35466	-99.16304	Entire home/apt	1140	3	31	0.32	4					
12	Villa Dante	153786	5	19.36585	-99.27911	Private room	3863	1	0	0	2					
13	Beautiful house Downtown Coyoacan	458775	4	19.34772	-99.16701	Entire home/apt	2897	5	0	0	1					
14	Seize our Holy Week / Easter \$247USD/in promotion!	519159	11	19.40236	-99.18163	Entire home/apt	4617	2	123	1.39	18					

Figura 1. Datos cargados en tabla AirBnB

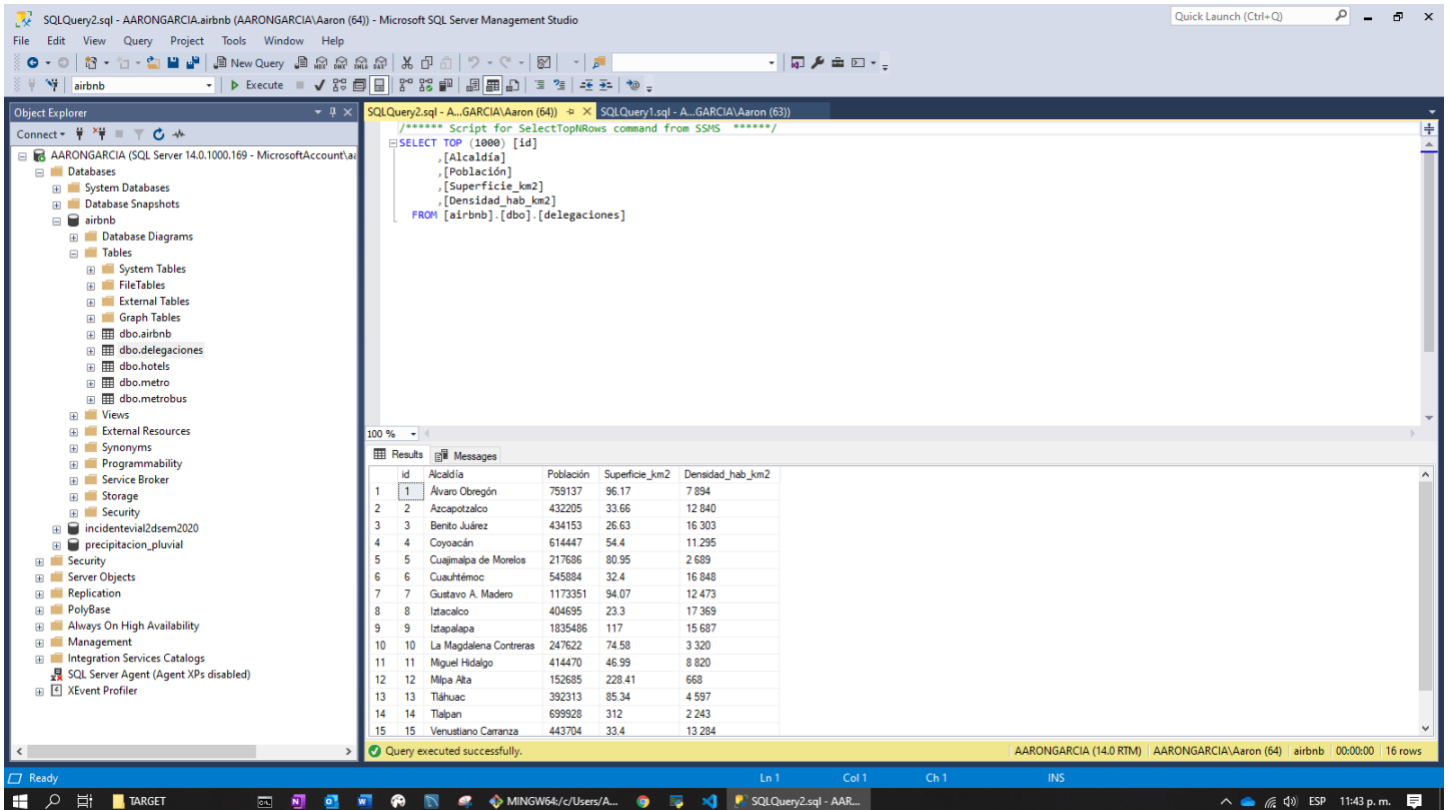


Figura 3. Datos cargados en tabla Delegaciones

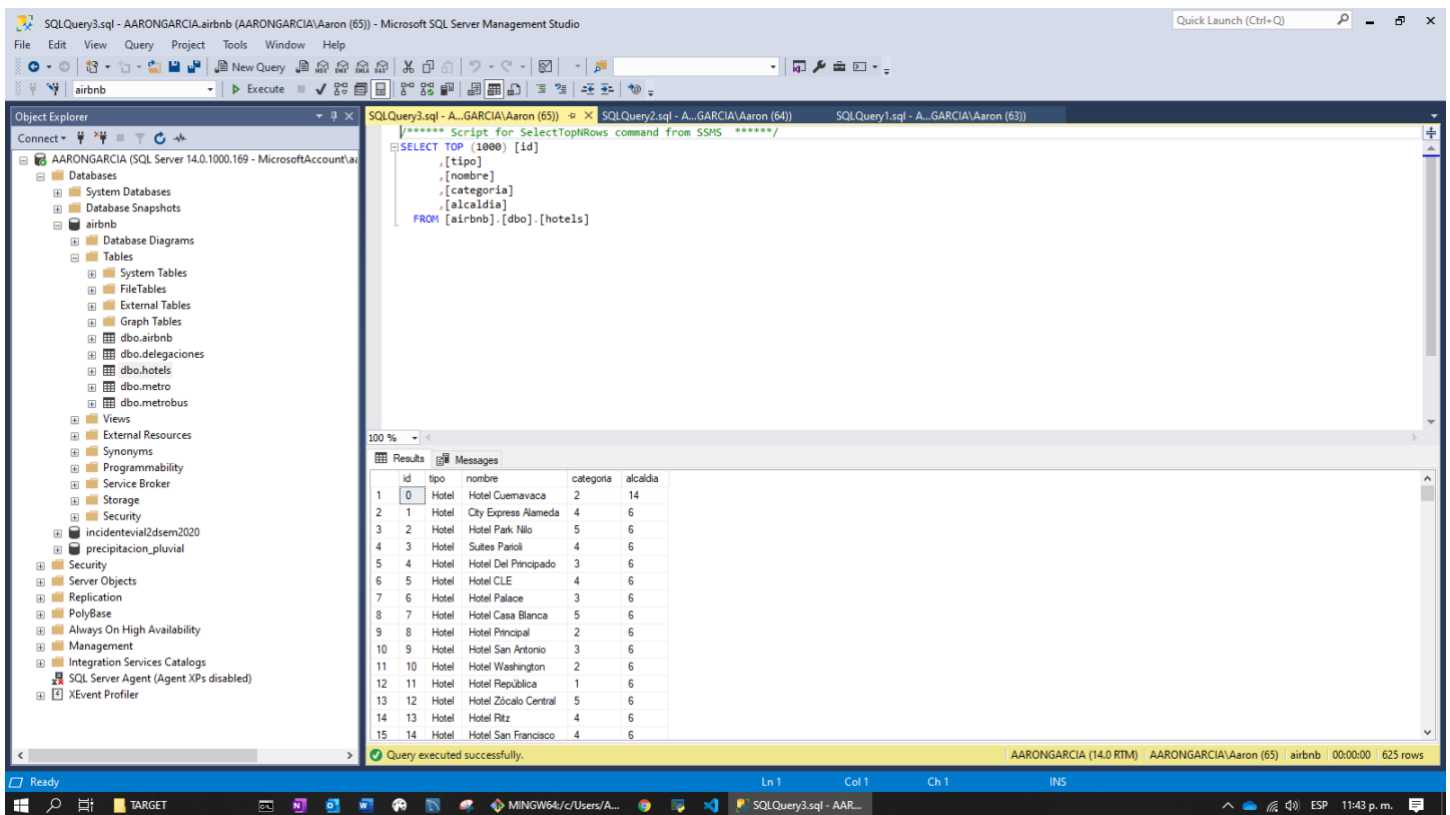


Figura 2. Datos cargados en tabla Hoteles

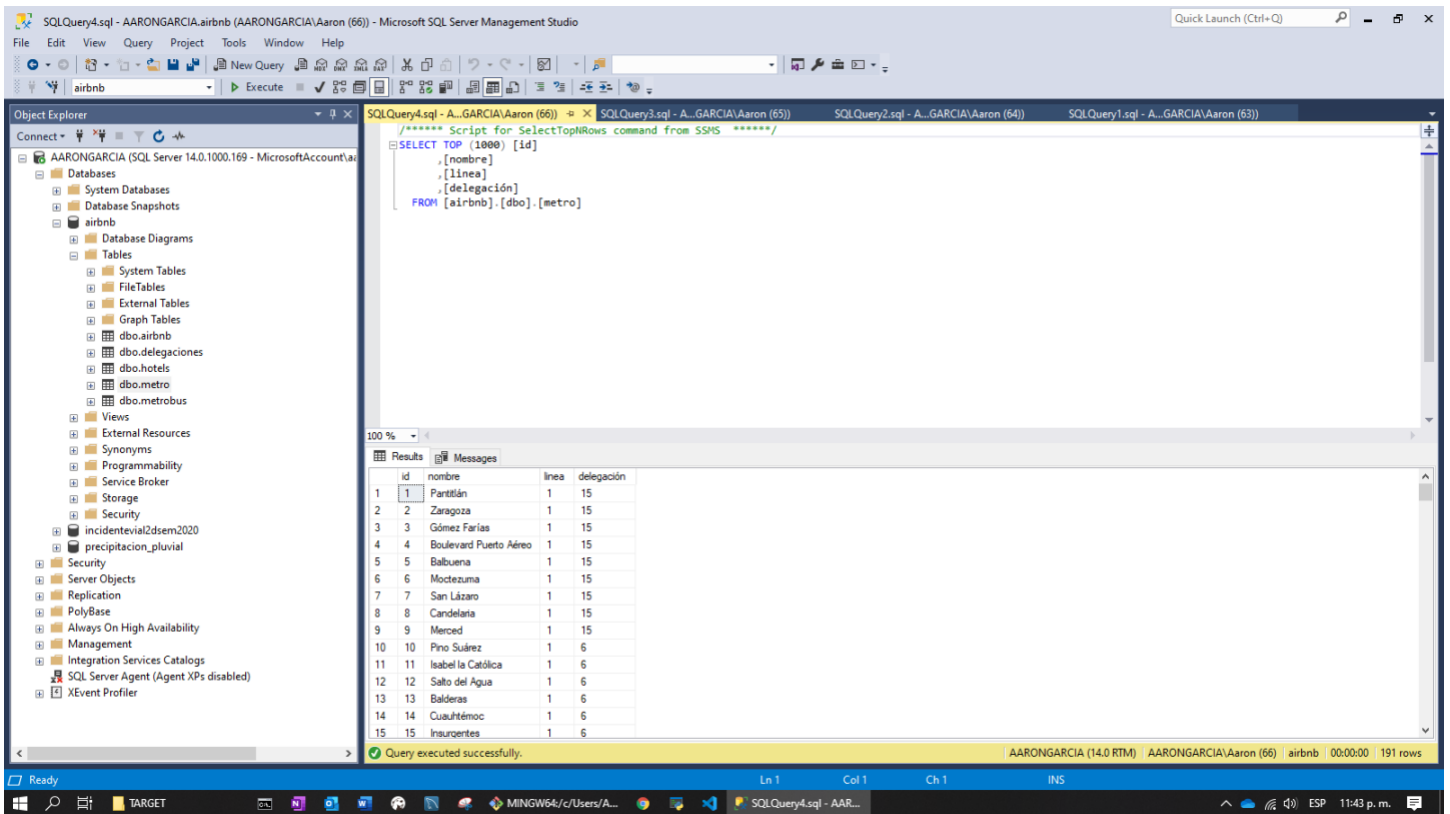


Figura 5. Datos cargados en tabla Metrobus

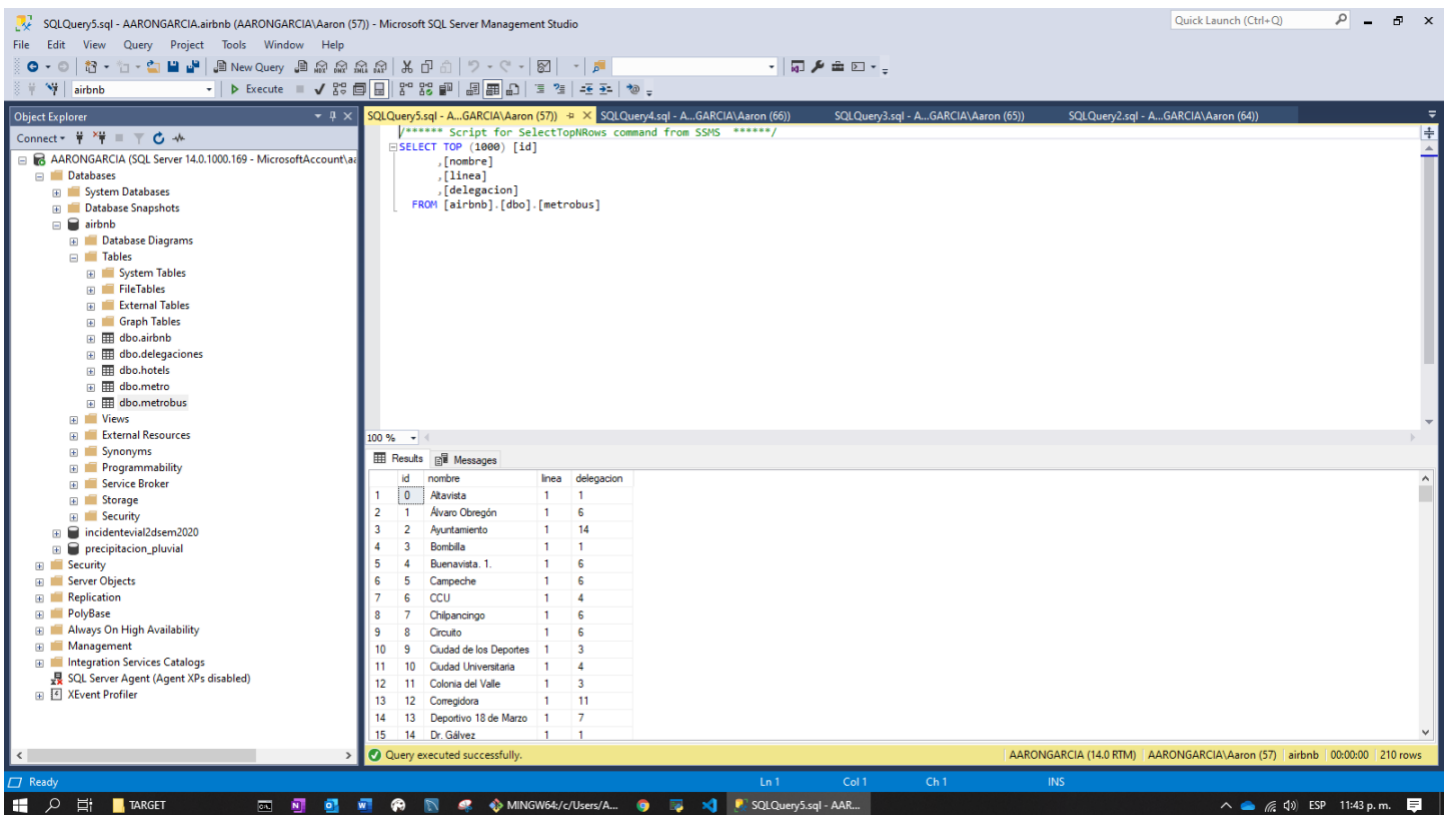
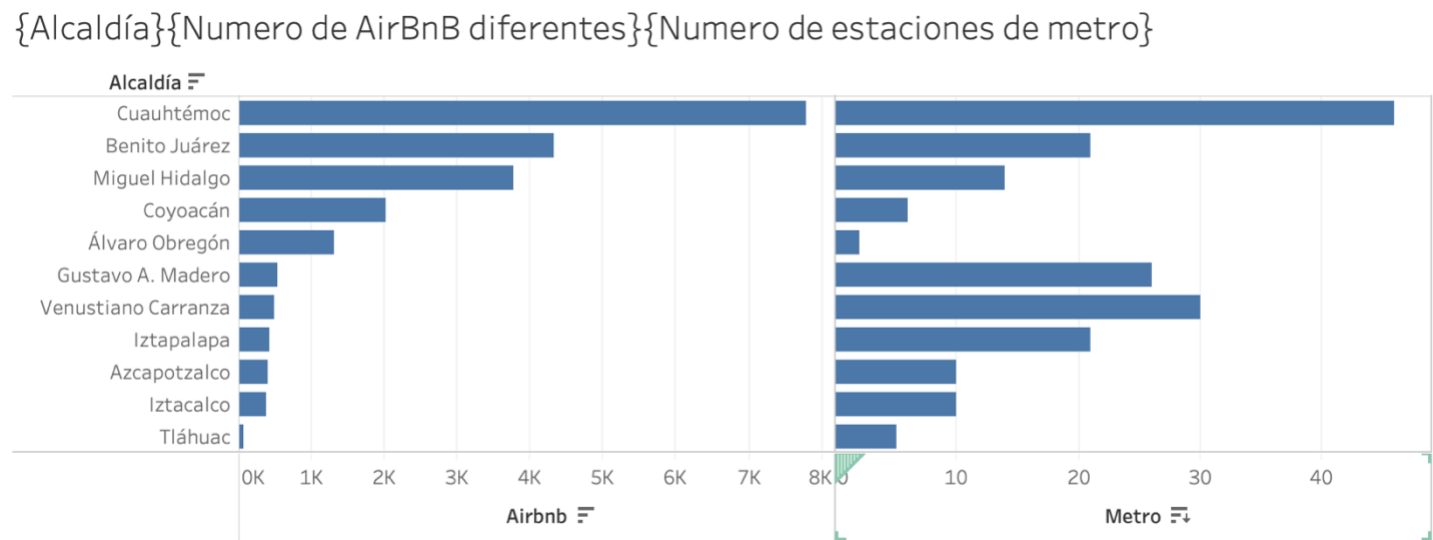


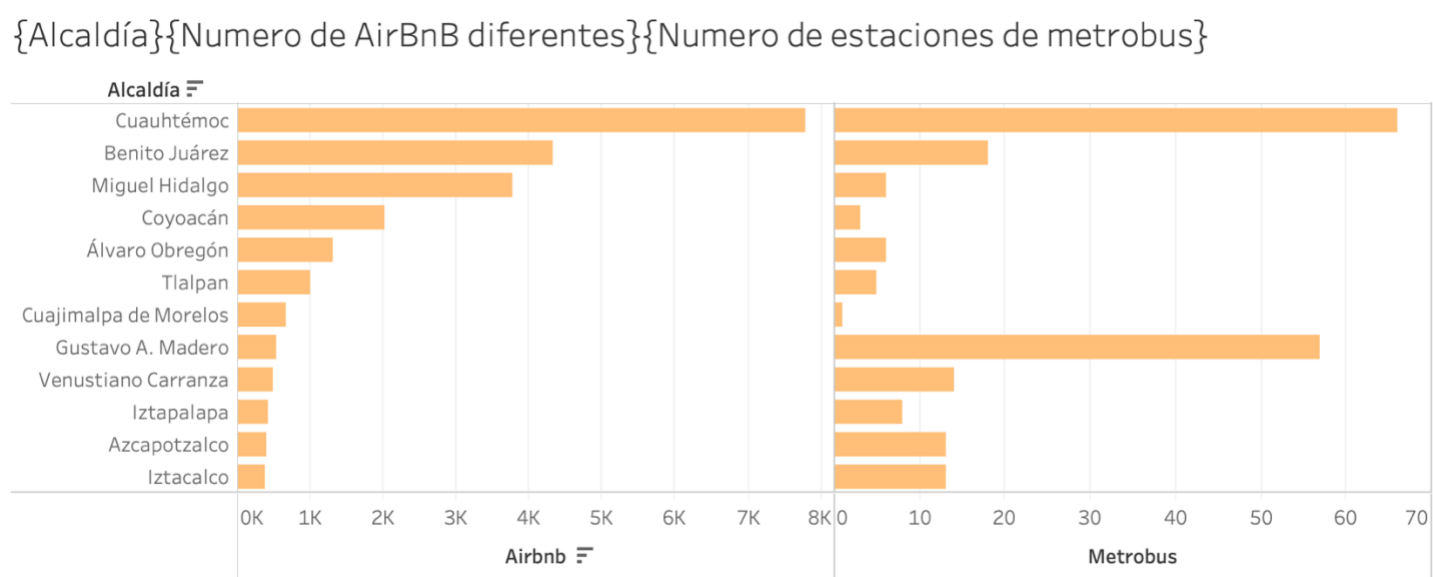
Figura 4. Datos cargados en tabla Metro

Los cubos de datos realizados fueron los siguientes:

Numero de AirBnB diferentes y numero de estaciones de metro por alcaldía



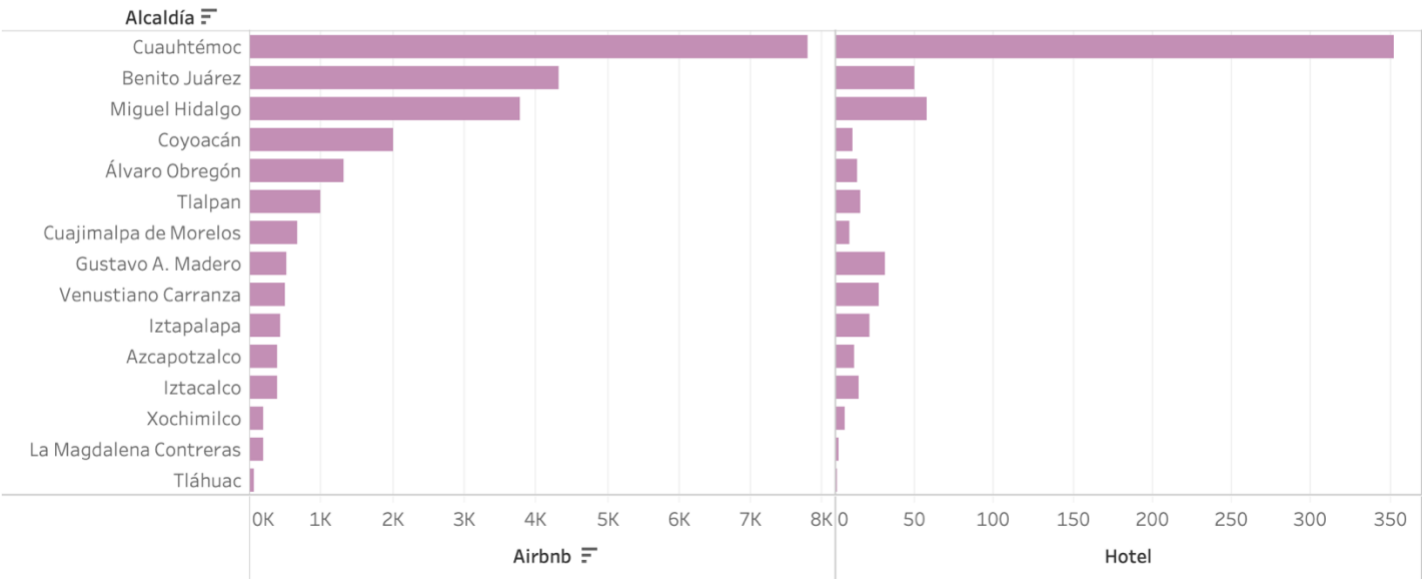
Numero de AirBnB diferentes y numero de estaciones de metrobus por alcaldía



A este nivel de granularidad parece que el número de estaciones de metro o metrobus no son proporcionales al número de AirBnB que hay por delegación, únicamente en las 3 delegaciones donde más hay AirBnB's pareciera tener relación, pero delegaciones como Gustavo A. Madero, Álvaro Obregón e Iztapalapa que concentran estaciones de metro, hay muy pocos AirBnB's, esto tiene lógica debido a que estas ultimas delegaciones se ubican en la periferia de la ciudad y el target de AirBnB radica en zonas cercanas a servicios o lugares importantes, que en su mayoría están en la parte interna de la ciudad.

Numero de AirBnB diferentes y numero de hoteles por alcaldía

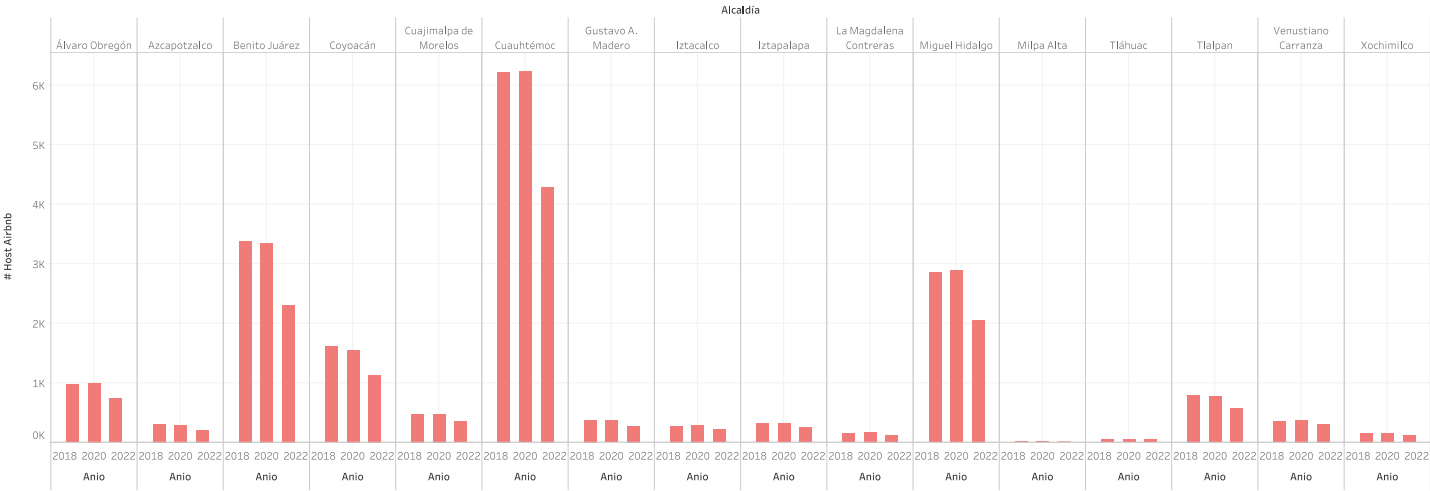
{Alcaldía}{Numero de AirBnB diferentes}{Numero de hoteles}



Definitivamente la delegación Cuauhtémoc es donde mayor oferta de hospedaje hay, donde mayor cantidad de estaciones de metro y metrobús hay dentro de CDMX, de igual manera la competencia por el sector hotelero es increíblemente grande, por cada hotel se ofertan 22 servicios AirBnB, en la alcaldía de Benito Juárez por cada hotel se ofertan 86 servicios AirBnB, en Miguel Hidalgo por cada hotel se ofrecen 65 servicios AirBnB, mientras que en delegaciones más grandes como Coyoacan, por cada hotel se ofertan 182 servicios AirBnB.

Numero de AirBnB diferentes por alcaldía y año

{Alcaldía}{Numero de AirBnB diferentes}{Año}



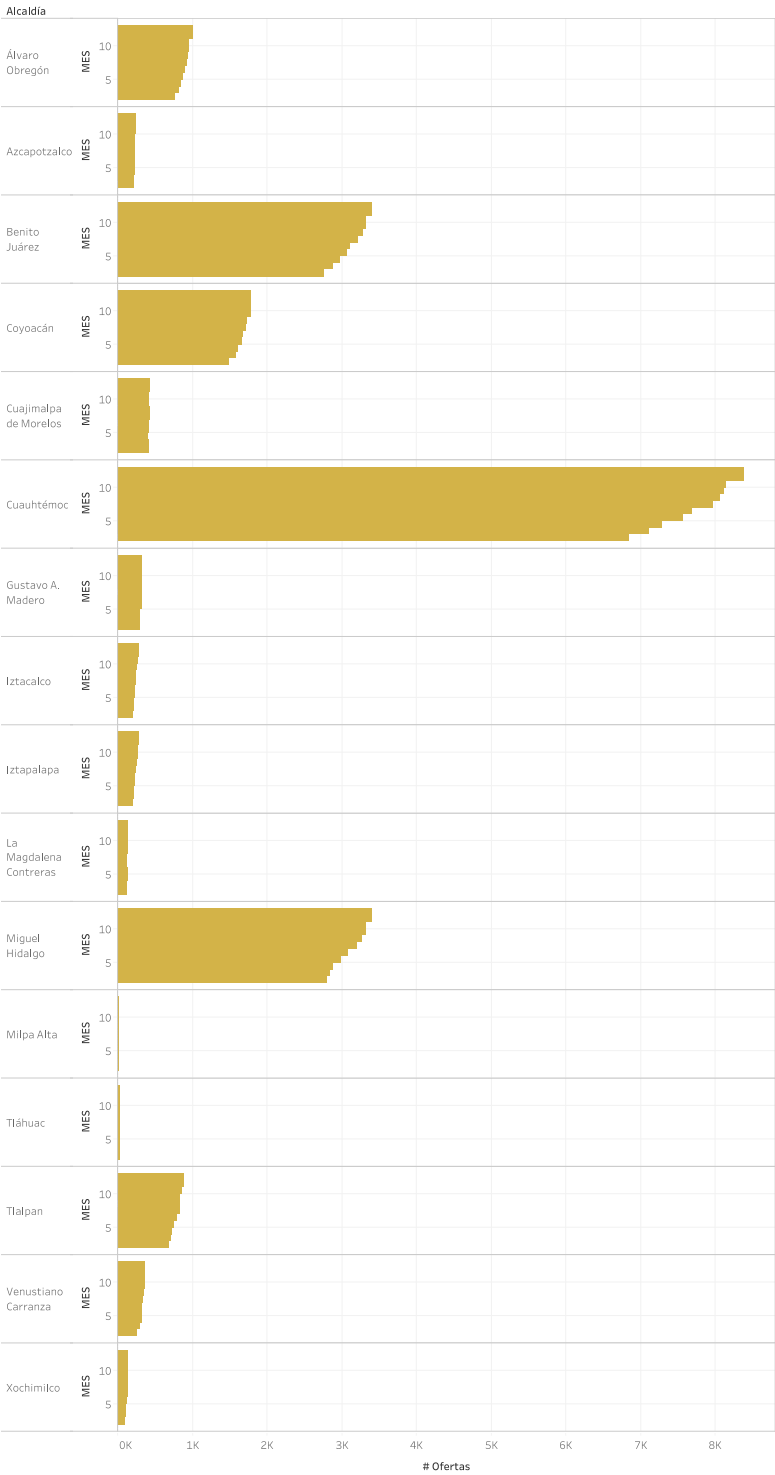
The plot of sum of # Host Airbnb for Año broken down by Alcaldía.

Como era de esperarse, el número de servicios diferentes de AirBnB disminuye de 2019 al 2020 debido a la pandemia covid 19.



Numero de AirBnB diferentes por mes y por alcaldía en 2019

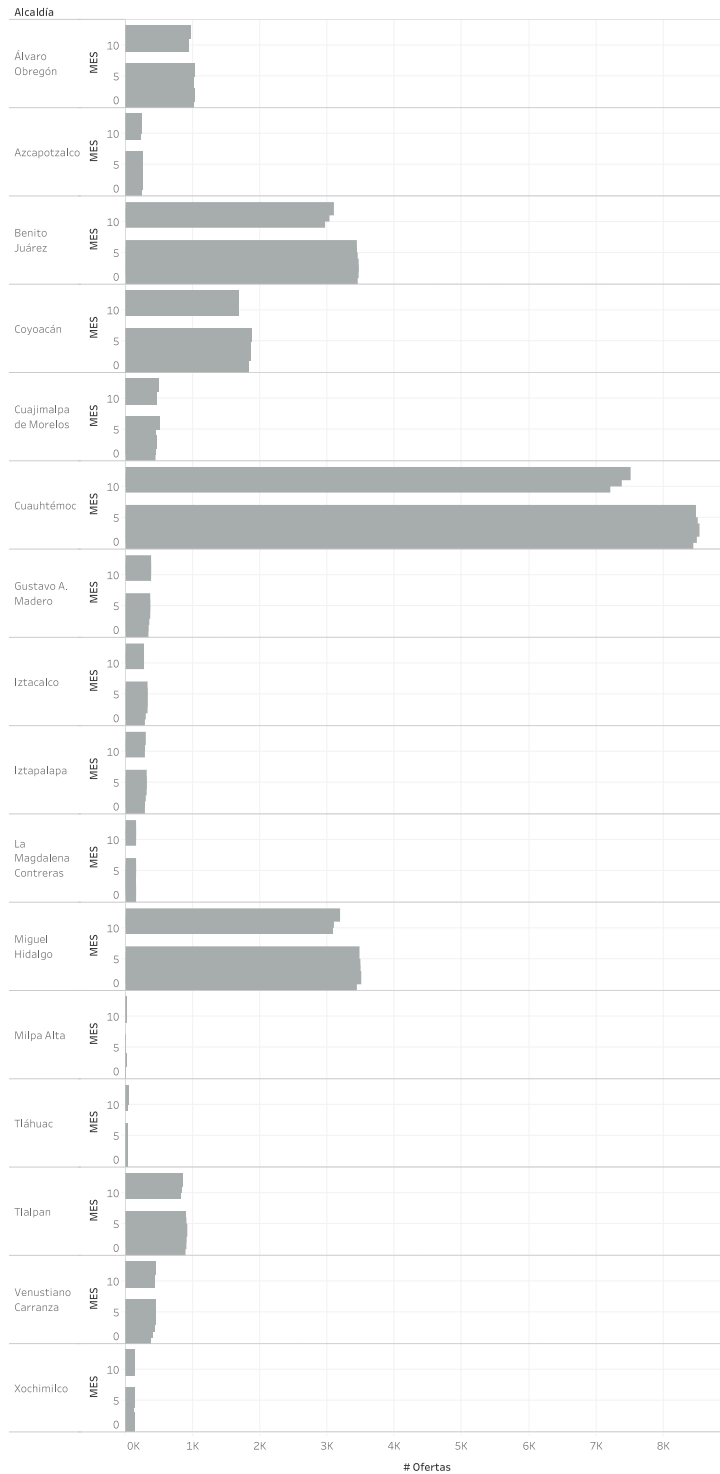
{Alcaldía}{Numero de AirBnB diferentes}{Mes} 2019



Es perceptible la relación entre avanzar los meses del año, se ofrecen más servicios de AirBnB en cualquiera de las alcaldías de CDMX.

## Numero de AirBnB diferentes por mes y por alcaldía en 2020

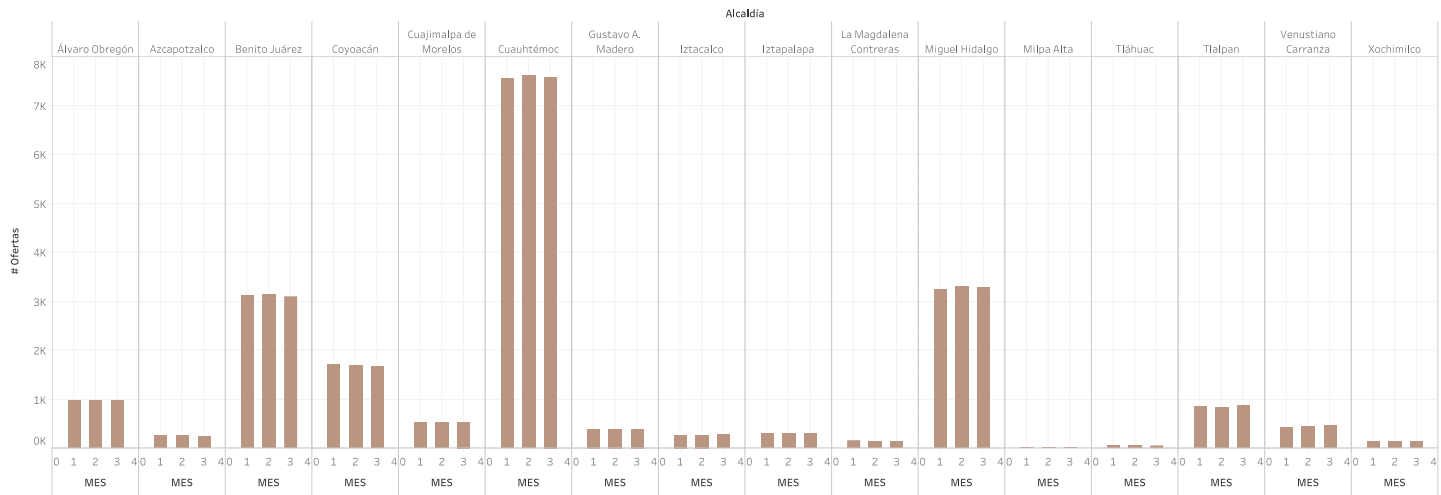
{Alcaldía}{Numero de AirBnB diferentes}{Mes} 2020



Como era de esperarse dado el avance de la pandemia en 2020, los primeros meses del año fue cuando mayor oferta de AirBnB hubo.

Numero de AirBnB diferentes por mes y por alcaldía en 2021

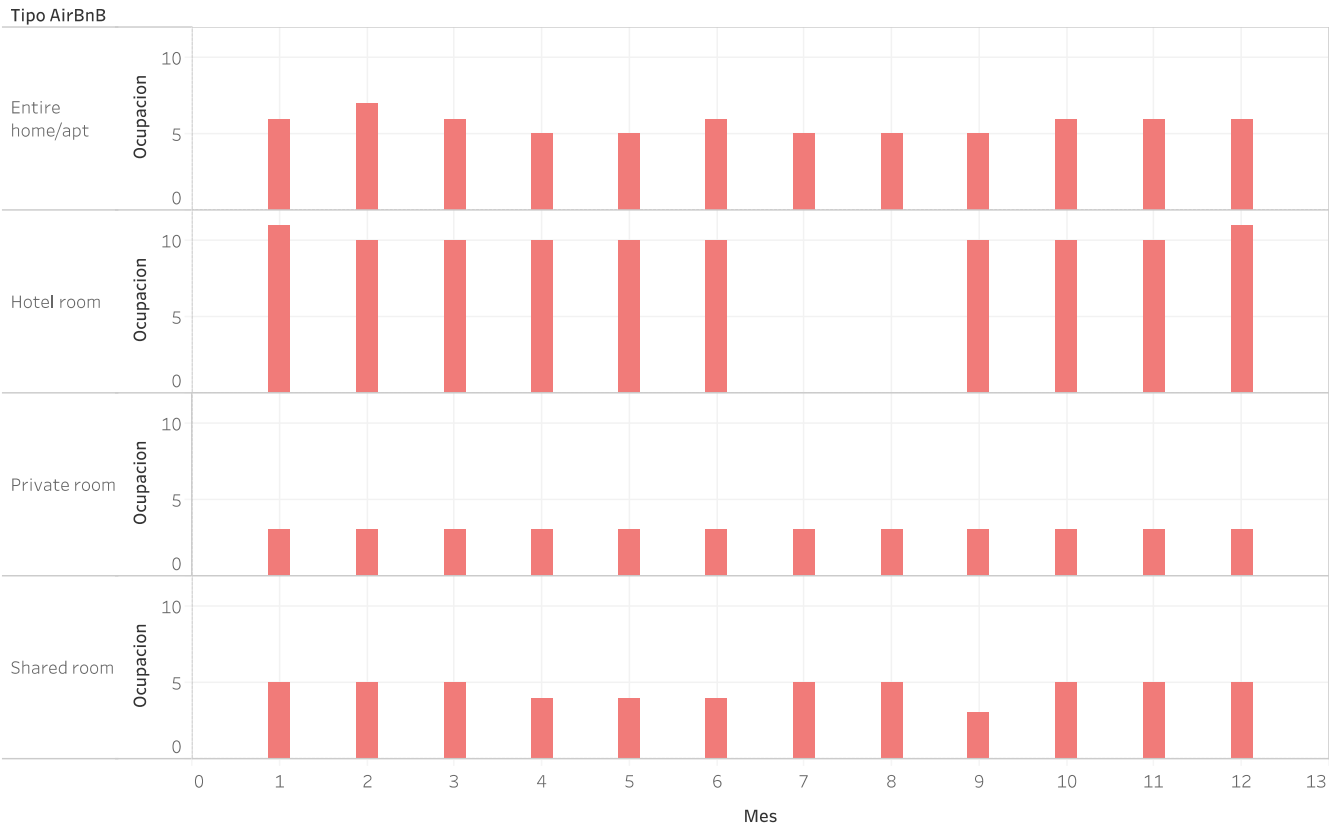
{Alcaldía}{Numero de AirBnB diferentes}{Mes} 2021



Para los meses reportados el 2021, no hay cambios significativos entre los 3 primeros meses del año que indiquen algún patrón o dato a mencionar.

Número de días promedio al mes por tipo de AirBnB

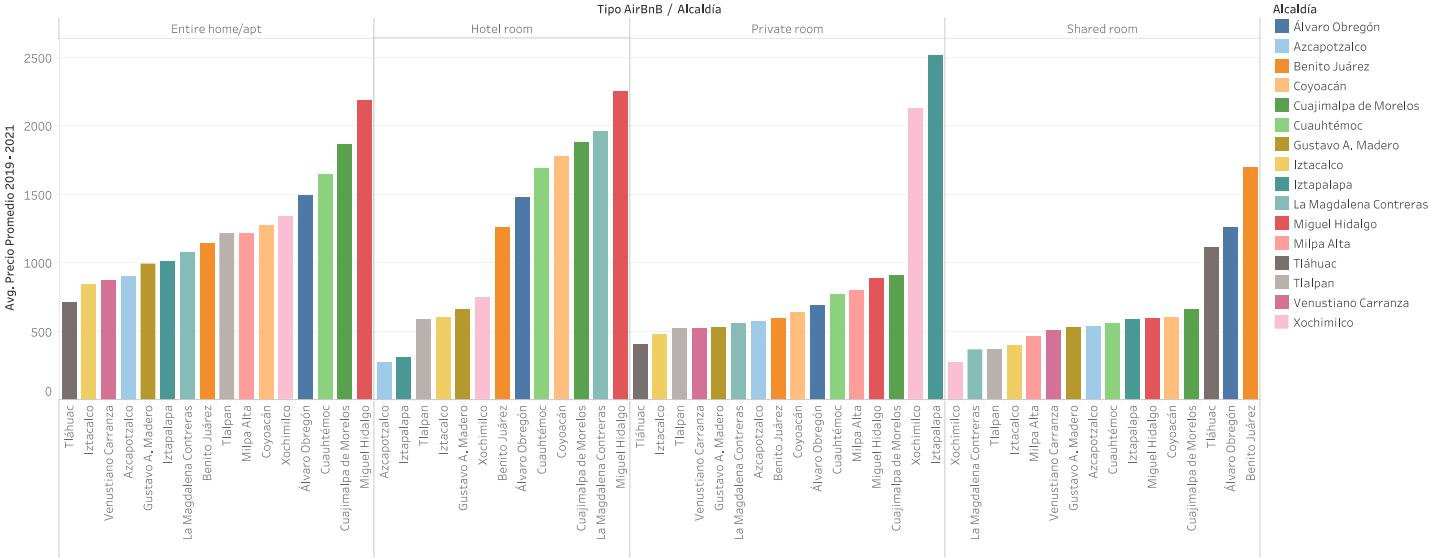
{Mes}{Tipo de AirBnB}{Ocupación promedio en días}



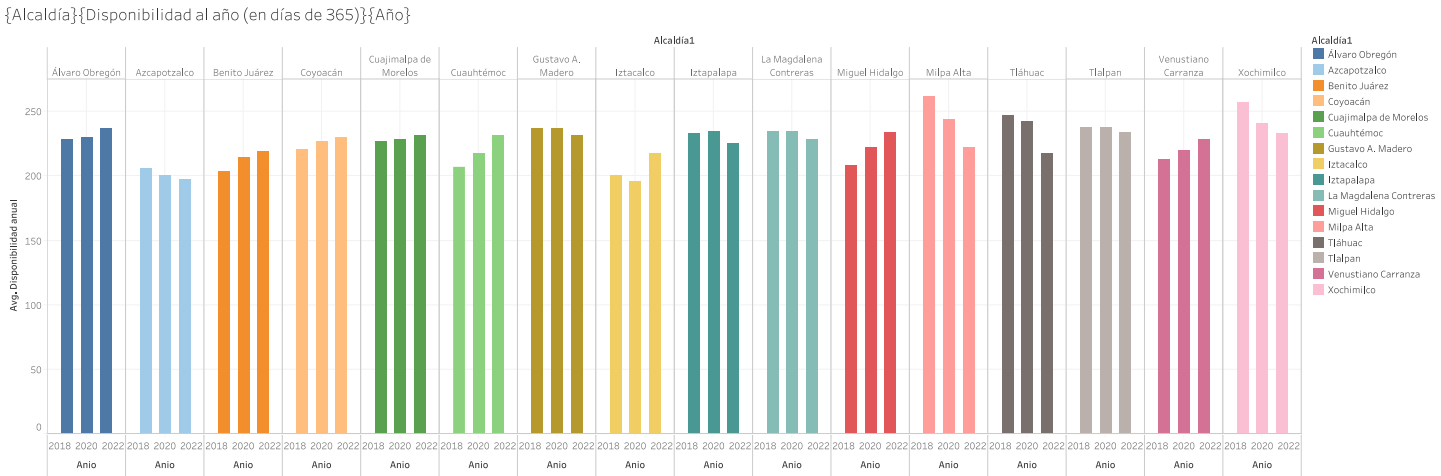
Dato curioso es que en promedio se renta 10 días de 30 que tiene el mes los AirBnB por habitación de hotel, en promedio 5 días al mes departamentos enteros y 4 días habitaciones compartidas.

Precio promedio por tipo de Airbnb por cada delegación

{Alcaldía}{Tipo de Airbnb}{Precio promedio}

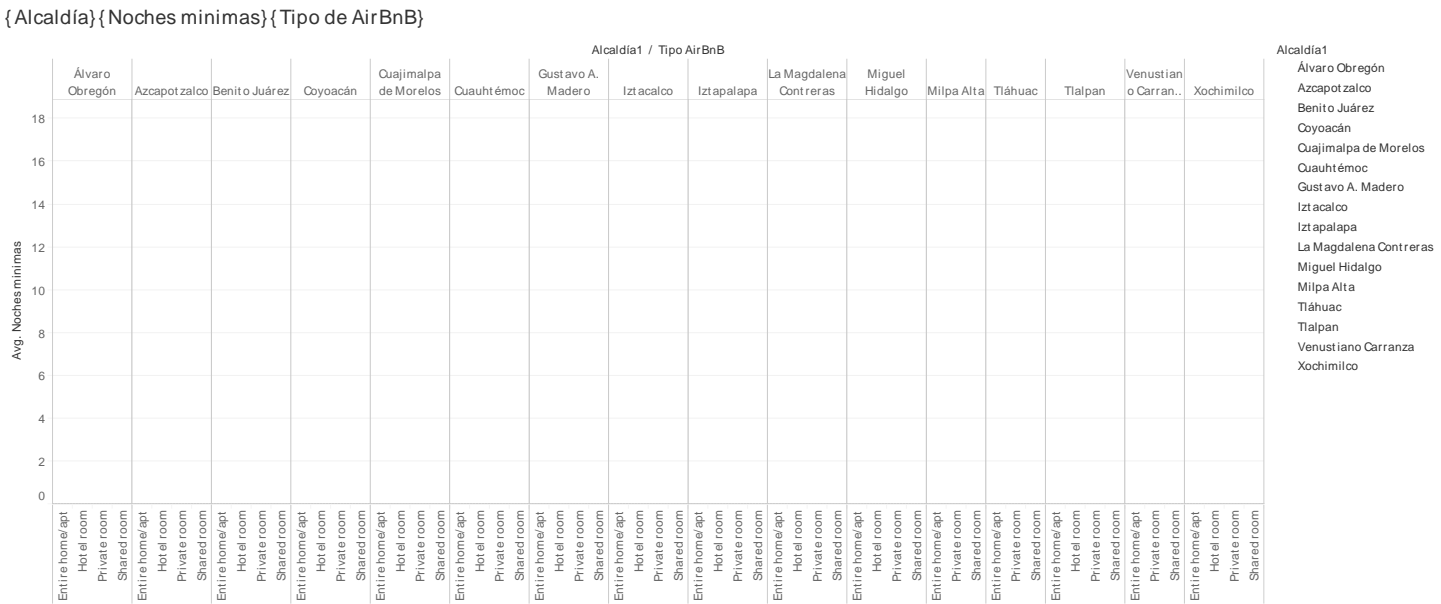


Número de días en promedio ofertados al año por cada alcaldía



Mencionaremos que la alcaldía Azcapotzalco e Iztacalco son las que menor numero de días al año ofertan los AirBnB’s disponibles, mientras que Milpa Alta, Xochimilco y Tlalpan son las que mas lo hacen.

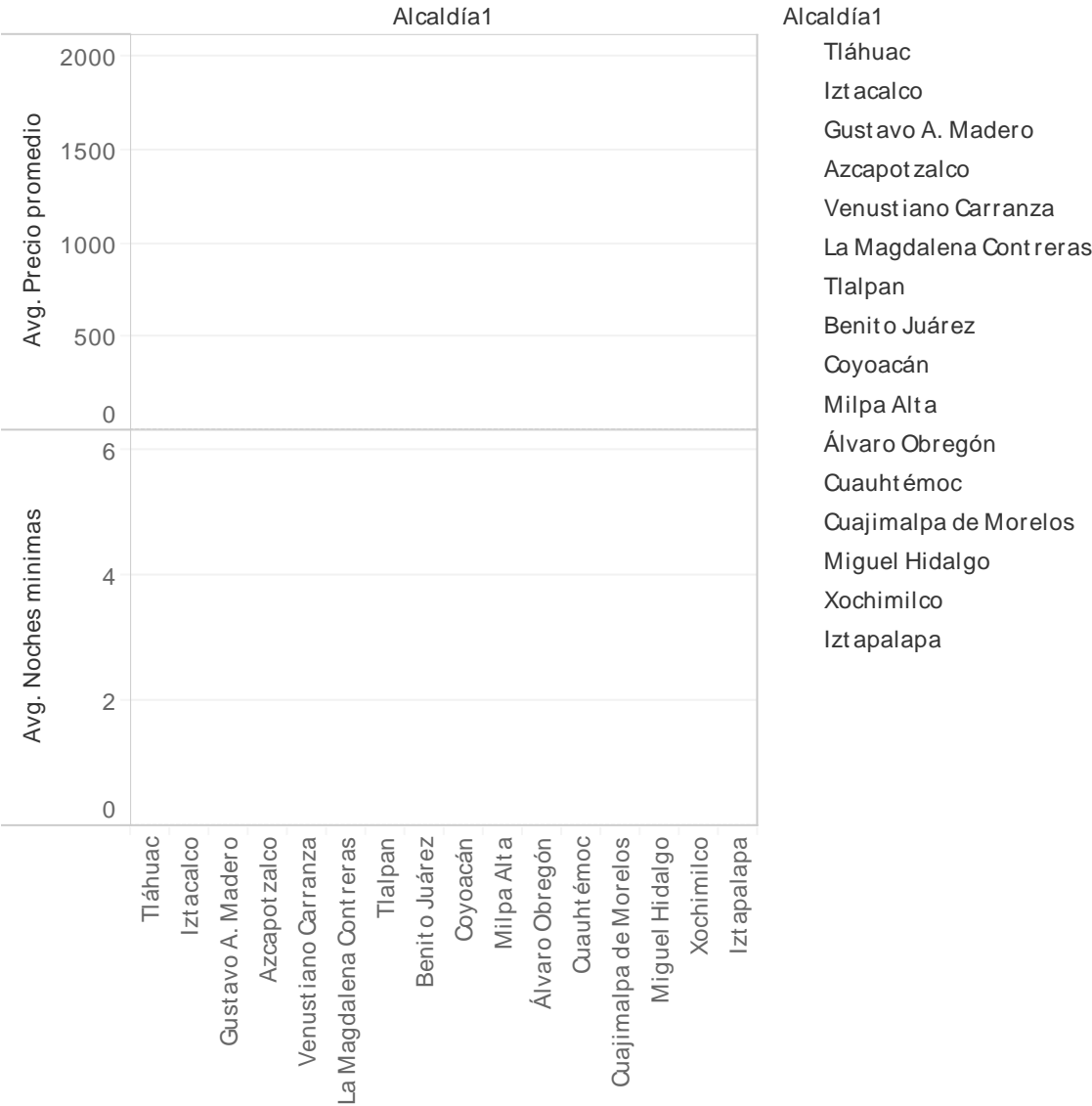
Número de noches mínimas promedio por tipo de AirBnB en cada alcaldía



Regularmente el numero de noches mínimas por tipo de servicio tiende a ser mayor si se oferta por renta completa de departamento, la alcaldía Tlalpan es la que tiene en promedio el numero de días mínimos mas alto de toda la ciudad, seguido por Álvaro Obregón, Azcapotzalco, Benito Juárez y Coyoacán, en contraste las que piden menor numero de días mínimos para departamento completo son; Milpa Alta, Tláhuac, Iztapalapa, Iztacalco y Cuauhtémoc.

Esto ultimo es un detalle muy importante, la alcaldía Cuauhtémoc siendo la mas atractiva y con mayor oferta, es de las que menor numero de días mínimos de estadía pide, mientras que las otras dos alcaldías mas exitosas en AirBnB (Benito Juárez y Miguel Hidalgo), son de las que mayor numero de días mínimos de estadía piden, un contraste muy interesante.

{ Alcaldía } { Noches minimas } { Precio promedio }



Pareciera no haber una relación apreciable entre el numero de días mínimos de estadía con los precios promedios por alcaldía.

## Conclusiones

Fue un ejercicio bastante interesante, reuní varias fuentes de datos para complementar la tabla de hechos siguiendo la arquitectura en estrella, debidamente el mayor tiempo que me tomo desarrollar esta practica fue en la limpieza e integración de los datos, el crear los cubos de datos es hasta cierto punto un arte si no se hace de manera automatizada ya que hay grandes posibilidades de hacer una y otra, claro, no todas tienen sentido pero la posibilidad ahí esta, es por ello entender los datos para saber que se puede lograr con respecto a lo solicitado.