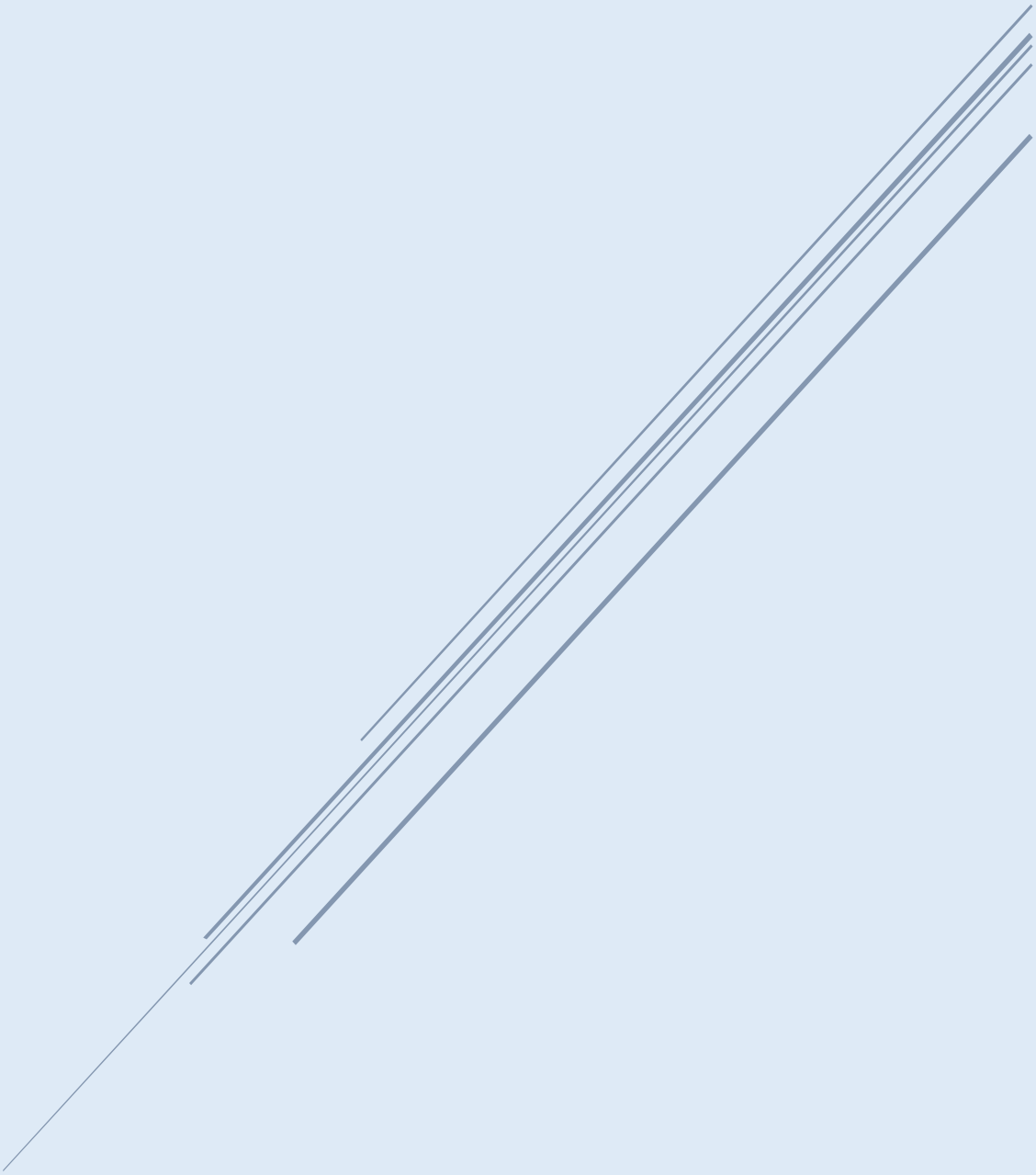
MODELADO Y TRANSFORMACION DE DATOS

Estos 2 procesos forman parte del proyecto de exploración y análisis de datos de un negocio ecommerce de productos electrónicos.

Link del proyecto



Paul Chipana Muñiz

Modelado de Datos (Data Modeling):

El Data Modeling es el proceso de diseñar una estructura organizada para los datos. Es equiparable a crear un plano para una casa antes de construirla. Los data analysts utilizan modelos de datos para definir cómo se verán y funcionarán los datos en una base de datos.

Una vez creada la conexión con MySql server, procedemos a "***Crear la base de datos relacional"***:

```
------------------------
----  Data Modeling  ----
------------------------

-- Creando la base de datos

CREATE DATABASE electronics_store_project;
USE electronics_store_project;
CREATE TABLE ecommerce_events_history_in_electronic_store (
    event_time VARCHAR(45) NULL,
    event_type VARCHAR(20) NULL,
    product_id INT NULL,
    category_id BIGINT NULL,
    category_code VARCHAR(90) NULL,
    brand VARCHAR(45) NULL,
    price DOUBLE NULL,
    user_id BIGINT NULL,
    user_session VARCHAR(90) NULL
) ENGINE=InnoDB;

-- Comprobando las características de los campos de la tabla
USE electronics_store_project ;
DESCRIBE ecommerce_events_history_in_electronic_store;
```

Field	Type	Null	Key	Default	Extra
event_time	varchar(45)	YES		NULL	
event_type	varchar(20)	YES		NULL	
product_id	int	YES		NULL	
category_id	bigint	YES		NULL	
category_code	varchar(90)	YES		NULL	
brand	varchar(45)	YES		NULL	
price	double	YES		NULL	
user_id	bigint	YES		NULL	
user_session	varchar(90)	YES		NULL	

```
-- Leeremos el siguiente Script que contiene 885 129 INSERT INTO statements para insertar registros del dataset de kaggle en la tabla recientemente creada
source C:/Users/GIGABYTE/Documents/Amazon_s3_storage/insert_sql_query_output.sql;
        -- Nota: el recurso "insert_sql_query_output.sql" debe estar ubicado en archivos locales para que mysql server pueda leerlo
        Link de descarga de "insert_sql_query_output.sql"

        -- si desea ver cómo transformamos cada registro (885 129) del dataset .csv de kaggle a formato INSERT INTO STATEMENT para facilitar a Mysql server la carga de registros a la base de datos recientemente creada, click aquí .
```

```sql
-- Comprobando la carga de registros (885129)
SELECT COUNT(*)
FROM ecommerce_events_history_in_electronic_store;
```

COUNT(*)
885129

```sql
-- Vista General de los 10 primeros registros de la tabla creada en MySql server
SELECT *
FROM ecommerce_events_history_in_electronic_store
LIMIT 10;
```

event_time	event_type	product_id	category_id	category_code	brand	price	user_id	user_session
2020-09-24 11:57:06 UTC	view	1996170	2144415922528452715	electronics.telephone		31.9	1515915625519388267	LJuJVLEjPT
2020-09-24 11:57:26 UTC	view	139905	2144415926932472027	computers.components.cooler	zalman	17.16	1515915625519380411	tdicluNnRY
2020-09-24 11:57:27 UTC	view	215454	2144415927158964449			9.81	1515915625513238515	4TMArHtXQy
2020-09-24 11:57:33 UTC	view	635807	2144415923107266682	computers.peripherals.printer	pantum	113.81	1515915625519014356	aGFYrNgC08
2020-09-24 11:57:36 UTC	view	3658723	2144415921169498184		cameronsino	15.87	1515915625510743344	aa4mmk0kwQ
2020-09-24 11:57:59 UTC	view	664325	2144415951611757447	construction.tools.saw	carver	52.33	1515915625519388062	vnkdP81DDW
2020-09-24 11:58:23 UTC	view	3791349	2144415935086199225	computers.desktop		215.41	1515915625519388877	J1t6sIYXiV
2020-09-24 11:58:24 UTC	view	716611	2144415923694469257	computers.network.router	d-link	53.14	1515915625519388882	kVBeYDPcBw
2020-09-24 11:58:25 UTC	view	657859	2144415939431498289			34.17	1515915625519320570	HEl15U7JVy
2020-09-24 11:58:31 UTC	view	716611	2144415923694469257	computers.network.router	d-link	53.14	1515915625519388929	F3VB9LYp39

Transformación de Datos (Data Transformation):

La Transformación de Datos implica tomar información en su forma original y cambiarla o adaptarla para que sea más útil o comprensible. Al realizar un proyecto de análisis de datos se utilizan técnicas de transformación para limpiar, combinar o reformatear datos, permitiendo que los datos se utilicen de manera efectiva en posteriores análisis y reportes.

En este caso, vamos a cambiar el formato de los datos de la columna event_time, que se encuentran en formato VARCHAR por DATETIME, esto nos permitirá realizar cálculos de hora y fecha, operaciones y comparaciones que impliquen intervalos de tiempo.

Asimismo, realizar una investigación más profunda de los valores atípicos (outliers) encontrados en el análisis exploratorio y tomar una decisión respecto a estos.

```sql
------------------------
-- Data Transformation --
------------------------

-- Corregiendo Formato de campos
-- 1. Cambiando de VARCHAR a DATETIME en columna event_time
-- 1.1. Quitar UTC al final de los datos event_time para obtener formato(YYYY-MM-DD %H:%i:%s)
UPDATE ecommerce_events_history_in_electronic_store
SET event_time = SUBSTRING_INDEX(event_time, " UTC", 1);

-- Verificando el cambio
SELECT event_time from ecommerce_events_history_in_electronic_store where event_time like "%UTC%";
```

event_time

```sql
SELECT event_time from ecommerce_events_history_in_electronic_store
LIMIT 10;
```

event_time
2020-09-24 11:57:06
2020-09-24 11:57:26
2020-09-24 11:57:27
2020-09-24 11:57:33
2020-09-24 11:57:36
2020-09-24 11:57:59
2020-09-24 11:58:23
2020-09-24 11:58:24
2020-09-24 11:58:25
2020-09-24 11:58:31

```sql
-- 1.2. Cambiar el datatype de la columna "event_time" de "varchar" a "datetime"
ALTER TABLE ecommerce_events_history_in_electronic_store
MODIFY COLUMN event_time DATETIME;

-- Verificando el cambio
DESCRIBE ecommerce_events_history_in_electronic_store;
```

Field	Type	Null	Key	Default	Extra
event_time	datetime	YES		NULL	
event_type	varchar(20)	YES		NULL	
product_id	int	YES		NULL	
category_id	bigint	YES		NULL	
category_code	varchar(90)	YES		NULL	
brand	varchar(45)	YES		NULL	
price	double	YES		NULL	
user_id	bigint	YES		NULL	
user_session	varchar(90)	YES		NULL	

```sql
-- OUTLIERS
-- Análisis de outliers: valores atípicos en la columna "price"

-- Ordenando según Precios mas altos y su frecuencia
USE electronics_store_project;
SELECT
category_id,
category_code as categoria_de_producto,
product_id,
price,
count(user_id) as Frecuencia
FROM ecommerce_events_history_in_electronic_store
group by price,category_id,category_code,product_id
HAVING price > 5000
order by price DESC;
```

category_id	categoria_de_producto	product_id	price	Frecuencia
2144415922402623591	computers.peripherals.monitor	4170916	64771.06	4
2144415927049912542	electronics.video.tv	4078837	42590.13	4
2144415927049912542	electronics.video.tv	4078834	27775.87	5
2144415929012846868		3830190	26985.35	2
2144415923107266682	computers.peripherals.printer	4101181	26909.62	1
2144415929012846868		3830191	23242.11	3
2144415929012846868		3790935	14233.22	1
2144415927049912542	electronics.video.tv	4078835	13627.02	3
2144415922402623591	computers.peripherals.monitor	1633600	12601.57	2
2144415927049912542	electronics.video.tv	875455	12559.95	4
2144415927049912542	electronics.video.tv	1029371	12149.24	2
2144415927049912542	electronics.video.tv	1011219	11886.76	11
2144415929012846868		3790942	11389.24	1
2144415923107266682	computers.peripherals.printer	1248537	11036.3	1
2144415927049912542	electronics.video.tv	4078836	10644.92	1
2144415929012846868		4086535	10284.33	1
2144415927049912542	electronics.video.tv	4078819	9338.17	3
2144415929012846868		1838848	7774.05	2
2144415922402623591	computers.peripherals.monitor	933606	7556.51	1

category_id	categoria_de_producto	product_id	price	Frecuencia
2144415922402623591	computers.peripherals.monitor	4170487	7497.32	1
2144415929012846868		1203151	7228.54	1
2144415923610583175		400238	7092.24	3
2144415923107266682	computers.peripherals.printer	669034	7052.08	2
2144415923694469257	computers.network.router	457077	6953.75	1
2144415923107266682	computers.peripherals.printer	1248535	6410.54	1
2144415922402623591	computers.peripherals.monitor	1848405	6242.56	1
2144415927049912542	electronics.video.tv	3735066	6036.83	1
2144415927049912542	electronics.video.tv	3735104	6036.83	3
2144415923107266682	computers.peripherals.printer	4159838	5937	11
2144415929012846868		1838847	5792.83	1
2144415923610583175		400237	5784.97	7
2144415922427789416	computers.components.videocards	1821826	5596.38	2
2144415929012846868		4076556	5512.65	1
2144415923107266682	computers.peripherals.printer	247475	5507.35	4
2144415926060056772	electronics.video.projector	4009528	5444.65	1
2144415923107266682	computers.peripherals.printer	11056	5149.38	1
2144415923610583175		288611	5148.13	3
2144415922402623591	computers.peripherals.monitor	1429013	5128.3	2
2144415923610583175		400236	5022.49	14
2144415927049912542	electronics.video.tv	4078833	5006.41	2

-- Conclusión:

-- Como se puede observar en la recuperación de datos de la consulta anterior, los registros con precios más altos tienen una frecuencia muy baja, los que tienen frecuencia de 1 son menos de 20 registros dentro del rango de precios de 5000 a 65000
-- Asimismo, se descarta que los registros con productos de precios más altos sean un error de ingreso de datos, ya que estos registros se encuentran dentro de las CATEGORÍAS DE PRODUCTO DE PRECIOS MÁS ALTOS como por ejemplo las siguientes categorías de producto: electronics.video.tv, computers.peripherals.monitor y computers.peripherals.printer.
-- Por lo tanto, al guardar relación estos registros de productos con precios más altos con categorías de producto, también de precios altos. Asimismo, al no contar estos registros con datos nulos o vacíos en campos relevantes como event_time, event_type, product_id, user_id, user_session, no se encuentra razones suficientes para eliminar estos registros de la base de datos.