# Desafío Alegra BI DEV



En Alegra, nos gustaría conocer mucho más acerca de tus habilidades cómo **BI Dev**, por eso, hemos preparado este desafío exclusivamente para vos.

Como verás a lo largo de este documento, la idea es que respondas de manera breve y concisa, a unas preguntas relacionadas con la experiencia de los productos que Alegra ofrece hoy en día. Por favor, ten en cuenta que la resolución de las consignas presentadas en este desafío no implica compromiso comercial alguno con Alegra, ya que este ejercicio se trata de un proceso de selección.

**Contexto**: DeliveryApp es una empresa canadiense que distribuye productos tecnológicos por las diferentes provincias del país, después de sus primeros 3 años de funcionamiento quieren medir sus esfuerzos con el fin de tomar decisiones basadas en datos, y te han contratado a ti para que les ayude en esa misión.

Para ello cuentas con las tablas que se encuentran en el archivo <u>BD - Test Business</u> <u>Intelligence Developer</u> (cada hoja es una tabla, en la hoja Leer se describe cada una), puede cargarlas en cualquier herramienta de gestión de bases de datos en donde pueda **utilizar lenguaje SQL** para resolver las siguientes preguntas.

**Nota**: realice una copia de este documento agregando al título del archivo su nombre, así puede anexar las consultas SQL y los resultados después de cada punto. También nos gustaría conocer la herramienta y comandos utilizados para cargar la información en el sistema de gestión de bases de datos, si le es posible realice un vídeo o tome capturas de pantalla.

# Consideraciones

Para este desafío se tuvo en cuenta:

- Las tablas de Excel son los medios de almacenamiento de información que utilizan los clientes, por los que las bases de datos en SQL se deben actualizar conforme a estas.
- La herramienta de gestión de bases de datos utilizada es MySQL y se crea una data base local llamada DeliveryApp.
- Utilizando un script de Python se realiza la carga de datos en MySQL acordes a la información extraída del documento de Excel por parte del cliente. Este script podría utilizarse como un servicio en la nube para automatizar este proceso.
- En MySQL se crean las vistas que representan cada uno de los requerimientos de cada numeral.
- Utilizando Power BI se hace una conexión con la base de datos de MySQL, se cargan los datos de acuerdo con las vistas y se crean los gráficos que ayuden a la comprensión de la información para los clientes.

#### Carga de Excel a SQL

Para este proceso se utilizaron librerías de Python llamadas sqlalchemy, pandas y mysql.connector. La creación de la base de datos se realizó en el entorno de MySQL, pero las tablas fueron generadas desde Python. A continuación, se anexa una sección importante del código implementado:

```
engine = create_engine('mysql://root:123456789@localhost/deliveryapp')
    excel_name = 'data_base.xlsx'
60 df = pd.read_excel(excel_name, sheet_name='Cliente')
   df.to_sql('cliente', con=engine, if_exists='append', index=False)
63 df = pd.read_excel(excel_name, sheet_name='Descuento')
64 df.to_sql('descuento', con=engine, if_exists='append', index=False)
65 print(df.head())
67  df = pd.read_excel(excel_name, sheet_name='Ventas')
68 print(df.head())
    df.rename(columns= {
                            'Fecha de Venta': 'fecha_venta',
                             'ClienteID': 'cliente_id',
                             'Precio Unitario': 'precio_unitario',
                             'DescuentoID': 'descuento_id',
                            'Prioridad': 'prioridad',
                            'Medio de Envío': 'medio_envio',
                            'Segmento': 'segmento',
                            'Provincia': 'provincia'
                            }, inplace= True)
    df.to_sql('venta', con=engine, if_exists='append', index=False)
```

Nota: en el siguiente repositorio se encuentra el script completo utilizado **excel\_to\_sql.py**, así como otros archivos relevantes en la realización de este desafío.

dhallito/desafio Alegra (github.com)

# Resultados

1. ¿Cuál ha sido el porcentaje de crecimiento de las ventas por año? Agrupe por año las ventas brutas (precio unitario\* cantidad) y las ventas netas (restando los descuentos) por año, obtenga la variación en números absolutos y en tasa de crecimiento respecto al año anterior. También incluya el número total de productos vendidos (suma cantidades) y número de pedidos (ventas únicas) por año. Nota: La tabla ventas no cuenta un con id, incluya un índice que le permita usar como identificador para contar las ventas únicas.

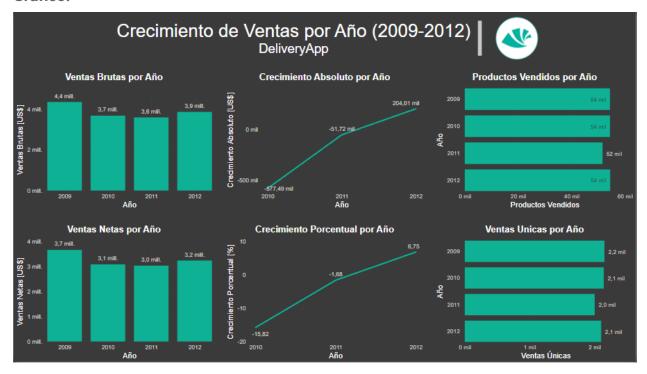
#### Vista:

```
9 •
        CREATE VIEW venta_año AS
     WITH crecimiento(año, ventas_brutas, ventas_netas, productos_vendidos, ventas_unicas) AS (
            SELECT YEAR(v.fecha venta) AS año,
                    sum(v.precio_unitario*v.cantidad) AS ventas_brutas,
12
                    sum(v.precio_unitario*v.cantidad*(1-d.descuento)) AS ventas_netas,
13
                    sum(v.cantidad) AS productos_vendidos,
14
                    count(v.id) AS ventas_unicas
15
16
            FROM venta v
            LEFT JOIN descuento d
17
            ON v.descuento_id = d.id
            GROUP BY YEAR(v.fecha_venta)
19
20
21
        SELECT c.año,
            c.ventas_brutas,
22
            c.ventas_netas,
            c.productos_vendidos,
24
25
            c.ventas_unicas,
            c.ventas_netas - c2.ventas_netas AS crecimiento_absoluto,
            100*(c.ventas_netas - c2.ventas_netas)/c2.ventas_netas AS crecimiento_porcentual
27
        FROM crecimiento c
        LEFT JOIN crecimiento c2
29
        ON c.año-1 = c2.año
30
        ORDER BY c.año;
```

#### Resultado:

1	Result Grid	Filter Rows:	Export:	Wrap Cell Content	<u>1A</u>		
	año	ventas_brutas	ventas_netas	productos_vendidos	ventas_unicas crecimiento_absoluto		
	2009	4355149.595220923	3650890.31785614	54380	2153	NULL	NULL
	2010	3685916.1579779387	3073397.823554992	54384	2142	-577492.4943011478	-15.817853839012628
	2011	3597484.1504118443	3021678.021991917	51564	2002	-51719.80156307528	-1.682821571834495
	2012	3868625.048956871	3225687.1603864753	54449	2102	204009.13839455834	6.751518093912392
	Result 5 x						

## Gráfico:



2. Se está pensando darles mayor atención a los medios de envío con mayor prioridad. Identifique el medio de envío con mayor cantidad de pedidos con Prioridad High o Critical, y compare la cantidad de pedidos y valores totales de este medio de envío respecto al resto de medios, obteniendo los resultados en porcentajes. Este análisis se desea hacer para la provincia con mayor de número de ventas (cantidad de ventas únicas).

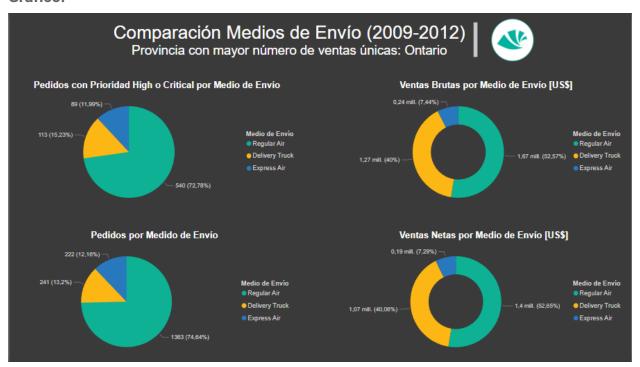
#### Vista:

```
CREATE VIEW prioridad_medio_envio AS
46
       SELECT v.medio envio.
          count(CASE WHEN v.prioridad IN ('High','Critical') THEN 1 ELSE NULL END) AS prioridad_high_critical,
          count(v.id) AS ventas_unicas,
          sum(v.precio_unitario*v.cantidad) AS ventas_brutas,
          sum(v.precio unitario*v.cantidad*(1-d.descuento)) A5 ventas netas,
           count(v.id)*100/(SELECT count(v.id) FROM venta v GROUP BY v.provincia ORDER BY count(v.id) DESC LIMIT 1) AS ventas_unicas_porcentuales,
52 ⊝ sum(v.precio_unitario*v.cantidad)*100/(
             SELECT sum(v.precio_unitario*v.cantidad)
53
             FROM venta v
             GROUP BY v.provincia
56
              ORDER BY sum(v.precio_unitario*v.cantidad) DESC LIMIT 1) AS ventas_brutas_porcentuales,
 57 \quad \ominus \qquad \text{sum(v.precio\_unitario*v.cantidad*(1-d.descuento))*100/(} 
              SELECT sum(v.precio_unitario*v.cantidad*(1-d.descuento))
59
              FROM venta v
              LEFT JOTN descuento d
60
             ON v.descuento_id = d.id
61
63
              ORDER BY sum(v.precio_unitario*v.cantidad*(1-d.descuento)) DESC LIMIT 1) AS ventas_netas_porcentuales,
         (SELECT v.provincia FROM venta v GROUP BY v.provincia ORDER BY count(v.id) DESC LIMIT 1) AS provincia
64
       FROM venta v
       LEFT JOIN descuento d
       ON v.descuento id = d.id
68 ⊝ LEFT JOIN (
         SELECT v.provincia, count(v.id)
70
          FROM venta v
71
          GROUP BY v.provincia
          ORDER BY count(v.id) DESC
          LIMIT 1) AS prov
     ON v.provincia = prov.provincia
75
     WHERE prov.provincia IS NOT NULL
    GROUP BY v.medio_envio
77 ORDER BY prioridad_high_critical desc;
```

#### Resultado:

Result Grid	Name of the Property of the Pr	Exp	ort: Wrap Cell Cont	ent: IA				
medio_envio	prioridad_high_critical	ventas_unicas	ventas_brutas	ventas_netas	ventas_unicas_porcentuales	ventas_brutas_porcentuales	ventas_netas_porcentuales	provincia
Regular Air	540	1363	1666692.1428931952	1402126.004791644	74.6440	52.565016970842336	52.651277862466394	Ontario
Delivery Truck	113	241	1268168.1147727966	1066745.25588463	13.1982	39.99615571427396	40.05738477434178	Ontario
Express Air	89	222	235864.7585890293	194171.4264437297	12.1577	7.438827314883706	7.291337363191878	Ontario

### Gráfico:



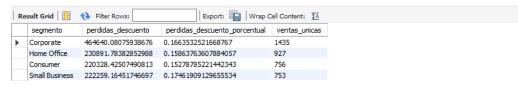
3. La empresa quiere analizar las pérdidas por descuentos realizados.

Por cada "Segmento" que atiende la empresa muestre el dinero perdido por descuentos, además el porcentaje que representa respecto al valor total de las ventas sin descuentos (valor total descuentos/valor total de ventas sin descuentos) y el número de ventas únicas que se han atendido por segmento. Por tratarse de un análisis muy específico, sólo debe considerar en la consulta los hombres (M) desde 30 años en adelante. Ordene su tabla por el número de ventas mostrando primero el segmento con mayor número de éstas. Puede utilizar subconsultas.

#### Vista:

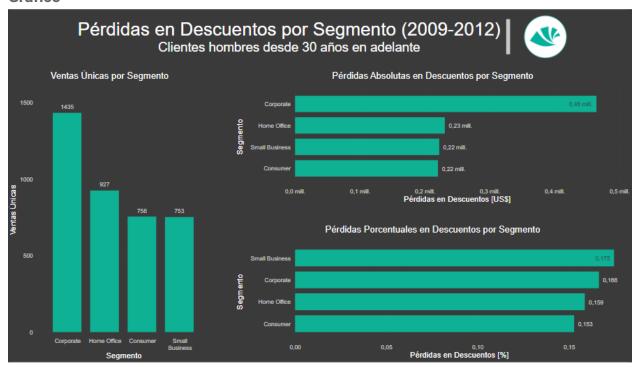
```
80 • CREATE VIEW perdida_descuento AS
        ventaM.segmento,
         sum(ventaM.cantidad*ventaM.precio_unitario*d.descuento) AS perdidas_descuento,
          sum(ventaM.cantidad*ventaM.precio_unitario*d.descuento)/sum(ventaM.cantidad*ventaM.precio_unitario) AS perdidas_descuento_porcentual,
          count(ventaM.id) AS ventas_unicas
86
87 \ominus (SELECT v.id, v.cantidad, v.precio_unitario, v.descuento_id, v.segmento
          FROM venta v
88
         LEFT JOIN cliente c
90
         ON v.cliente_id = c.id
          WHERE c.genero = 'M' AND c.edad >= 30) ventaM
91
     LEFT JOIN descuento d
92
       ON ventaM.descuento_id = d.id
      GROUP BY ventaM.segmento
    ORDER BY ventas_unicas desc;
```

#### Resultado:



Result 6 ×

### Gráfico



4. La empresa desea premiar a sus mejores clientes durante el último año, 2012, como estrategia de retención. Por ello, define los siguientes obsequios.

Obsequio A: El mejor cliente. El cliente con la compra más alta (quien realizó la compra neta con mayor valor).

Obsequio B: Cliente más fiel. El cliente con mayor cantidad de compras únicas.

Obsequio C: Clientes estrellas. Ranking con los 3 clientes que dejaron más ganancias.

El obsequio consiste en un bonus para comprar cualquier producto de la tienda, en función del total de sus compras (valor neto) durante el 2012.

Obsequio	Porcentaje de descuento
А	1%
В	2%
С	4%

¿Cuál será el valor del bonus por cada cliente?

#### Vista:

```
CREATE VIEW obsequio_cliente AS
     SELECT
100
               c.nombre,
102
               max(v.cantidad*v.precio_unitario) AS mayor_compra,
                count(v.id) AS ventas_unicas,
                sum(v.precio_unitario*v.cantidad*(1-d.descuento)) AS ventas_netas
104
            FROM venta v
105
            LEFT JOIN descuento d
106
            ON v.descuento_id = d.id
107
108
           LEFT JOIN cliente c
109
           ON v.cliente_id = c.id
            WHERE year(v.fecha_venta)=2012
110
111
            GROUP BY c.nombre
      ( با
112
113
      SELECT p.nombre,
114
            p.mayor_compra,
           p.ventas_unicas,
115
116
            p.ventas_netas,
            CASE WHEN p.mayor_compra = (SELECT max(p.mayor_compra) FROM premios p) THEN 0.01 ELSE 0 END AS obsequio_A,
117
118
            CASE WHEN p.ventas_unicas = (SELECT max(p.ventas_unicas) FROM premios p) THEN 0.02 ELSE 0 END AS obsequio_B,
            CASE WHEN p2.nombre IS NOT NULL THEN 0.04 ELSE 0 END AS obsequio_C,
119
120
            ((CASE WHEN p.mayor_compra = (SELECT max(p.mayor_compra) FROM premios p) THEN 0.01 ELSE 0 END)+
            (CASE WHEN p.ventas_unicas = (SELECT max(p.ventas_unicas) FROM premios p) THEN 0.02 ELSE 0 END)+
121
122
            (CASE WHEN p2.nombre IS NOT NULL THEN 0.04 ELSE 0 END))*p.ventas_netas AS valor_bonus
123
      FROM premios p
124

    □ LEFT JOIN (
            SELECT p.ventas_netas, p.nombre
125
126
            FROM premios p
            ORDER BY p.ventas_netas DESC
127
128
            LIMIT 3) AS p2
       ON p.nombre = p2.nombre
129
        ORDER BY valor_bonus desc;
```

#### Resultado:

	Result Grid			Export: Wrap Cell Content: 🚹					
	nombre	mayor_compra	ventas_unicas	ventas_netas	obsequio_A	obsequio_B	obsequio_C	valor_bonus	
•	Jim Radford	31499.909912109375	62	150423.10132952535	0.00	0.00	0.04	6016.924053181014	
	Sylvia Foulston	22049.509521484375	55	128599.15549743675	0.00	0.00	0.04	5143.96621989747	
	Don Jones	24151.48992919922	54	114781.09051820353	0.00	0.00	0.04	4591.243620728142	
	Grant Carroll	21542.14047241211	65	100429.07685638223	0.00	0.02	0.00	2008.5815371276447	
	Annie Cyprus	54264.16015625	62	108565.80638276001	0.01	0.00	0.00	1085.6580638276002	
	Clay Rozendal	14325.4794921875	52	100631.7973177876	0.00	0.00	0.00	0	
	Carl Jackson	7302.500152587891	52	63779.44521358108	0.00	0.00	0.00	0	
	Joy Bell	8126.460296630859	53	58333.79355129832	0.00	0.00	0.00	0	
	Frank Price	11384.5595703125	61	79468.62656370855	0.00	0.00	0.00	0	
	Michelle Lonsdale	10630.000305175781	60	58835.21545184326	0.00	0.00	0.00	0	
	Edward Hooks	10199.84017944336	52	55564.593994861185	0.00	0.00	0.00	0	

#### Gráfico:

