



Enext

Consultoria E-commerce e Marketing Digital

Débora Nunes Ferreira

DESAFIO 1: SQL

BRASÍLIA, DF
2024

Introdução

A empresa Impulso Digital Retailers é uma organização fictícia que atua no setor de varejo e tem investido fortemente em marketing digital para impulsionar suas vendas online. Utilizando três principais veículos de publicidade - Meta Ads (Facebook), Google Ads e TikTok Ads - a empresa busca alcançar seus clientes de maneira eficaz através de campanhas direcionadas.

A Impulso Digital Retailers enfrenta desafios significativos na análise de dados de suas campanhas publicitárias. Atualmente, os dados são coletados manualmente de cada plataforma e consolidados em uma planilha do Excel. Esse processo é moroso, propenso a erros e não proporciona uma visão unificada dos resultados das campanhas. Além disso, a equipe de marketing enfrenta dificuldades para comparar o desempenho entre as diferentes plataformas.

Objetivo

Para superar esses obstáculos, a Impulso Digital Retailers decidiu implementar um dashboard de Business Intelligence (BI) utilizando o Looker Studio. Este dashboard integrará automaticamente os dados das três plataformas de publicidade utilizando o Google BigQuery. Para alcançar nosso objetivo, fizemos a consolidação das tabelas Google Ads, Meta Ads e TikTok Ads em duas principais entidades: campanhas e veículos. Isso nos permitiu integrar e unificar os dados de maneira eficiente, proporcionando uma análise consolidada e facilitando a tomada de decisões estratégicas pela equipe de marketing.

Na seguinte seção, vamos mostrar como essas tabelas foram consolidadas em SQL, além das views criadas e selects utilizados para preparar os dados para consumo.

Consolidação de Dados Usando UNION ALL

A consolidação de dados é um passo crucial em nosso projeto de integração de campanhas publicitárias. Utilizamos a cláusula UNION ALL para combinar dados provenientes de diferentes plataformas, como Meta Ads (Facebook), Google Ads e TikTok Ads.

Criando uma Nova Tabela Usando UNION ALL

Criamos uma nova tabela chamada Campanhas, que consolida dados de todas as plataformas de anúncios, facilitando a consulta e a análise subsequente dos dados. Além disso, criamos outra tabela para consolidar os dados dos veículos de anúncios.

Campanhas Consolidadas

EXECUTAR

SALVAR CONSULTA ▾

```
-- Criando tabela consolidada de campanhas

CREATE TABLE desafio_sql_time03.campanhas_consolidadas AS
SELECT
  'Google Ads' AS plataforma,
  campanha AS campanha_id,
  funil
FROM
  desafio_sql_time03.googleads_campanhas

UNION ALL

SELECT
  'Meta Ads' AS plataforma,
  campanha AS campanha_id,
  funil
FROM
  desafio_sql_time03.metaads_campanhas

UNION ALL

SELECT
  'TikTok Ads' AS plataforma,
  campanha AS campanha_id,
  funil
FROM
  desafio_sql_time03.tiktok_campanhas;
```

Veículos Consolidados

[EXECUTAR](#)[SALVAR CONSULTA](#) ▼

```
-- Criando tabela consolidada de veículos

CREATE TABLE desafio_sql_time03.veiculos_consolidados AS
SELECT
    data,
    campanha AS campanha_id,
    custo,
    impressoes,
    cliques,
    'Google Ads' AS plataforma
FROM
    desafio_sql_time03.googleads_resultados

UNION ALL

SELECT
    data,
    campanha AS campanha_id,
    custo,
    impressoes,
    cliques,
    'Meta Ads' AS plataforma
FROM
    desafio_sql_time03.metaads_resultados

UNION ALL

SELECT
    data,
    campanha AS campanha_id,
    custo,
    impressoes,
    cliques,
    'TikTok Ads' AS plataforma
FROM
    desafio_sql_time03.tiktok_resultados;
```

Uso de Views para Consolidação de Dados

Outra forma de consolidação é através das views, que são tabelas virtuais baseadas no resultado de uma consulta SQL. Utilizando views, podemos simplificar a gestão e a consolidação de dados, mantendo a lógica de união encapsulada e separada das operações diárias. Veja a seguir:

View Campanhas Consolidadas

[EXECUTAR](#)[SALVAR CONSULTA](#) ▾

```
-- Criando uma view de campanhas consolidadas

CREATE VIEW desafio_sql_time03.view_campanhas_consolidadas AS
SELECT
    'Google Ads' AS plataforma,
    campanha AS campanha_id,
    funil
FROM
    desafio_sql_time03.googleads_campanhas

UNION ALL

SELECT
    'Meta Ads' AS plataforma,
    campanha AS campanha_id,
    funil
FROM
    desafio_sql_time03.metaads_campanhas

UNION ALL

SELECT
    'TikTok Ads' AS plataforma,
    campanha AS campanha_id,
    funil
FROM
    desafio_sql_time03.tiktok_campanhas;

-- Consultando a view de campanhas consolidadas

SELECT *
FROM desafio_sql_time03.view_campanhas_consolidadas;
```

Veículos Consolidados

[EXECUTAR](#)[SALVAR CONSULTA](#) ▾

```
-- Criando tabela consolidada de veículos

CREATE TABLE desafio_sql_time03.veiculos_consolidados AS
SELECT
    data,
    campanha AS campanha_id,
    custo,
    impressoes,
    cliques,
    'Google Ads' AS plataforma
FROM
    desafio_sql_time03.googleads_resultados

UNION ALL

SELECT
    data,
    campanha AS campanha_id,
    custo,
    impressoes,
    cliques,
    'Meta Ads' AS plataforma
FROM
    desafio_sql_time03.metaads_resultados

UNION ALL

SELECT
    data,
    campanha AS campanha_id,
    custo,
    impressoes,
    cliques,
    'TikTok Ads' AS plataforma
FROM
    desafio_sql_time03.tiktok_resultados;
```

Com as tabelas consolidadas, é mais fácil e rápido realizar consultas (SELECT). O SELECT é uma instrução SQL usada para buscar dados em uma tabela ou view. Ele permite selecionar colunas específicas e aplicar filtros para obter as informações desejadas, como ilustrado nas imagens a seguir.

Search Selects

EXECUTAR

SALVAR CONSULTA

FAZER O DOWNLOAD

```
1
2  -- Consulta para somar cliques por plataforma na tabela veiculos_consolidados
3
4  SELECT
5      plataforma,
6      SUM(cliques) AS total_cliques
7  FROM
8      `prj-enext-bi-capacitacao.desafio_sql_time03.veiculos_consolidados`
9  WHERE
10     plataforma IN ('Meta Ads', 'TikTok Ads', 'Google Ads')
11  GROUP BY
12     plataforma;
13
```

Resultados da consulta

INFORMAÇÕES DO JOB	RESULTADOS	GRÁFICO	JSON	DETALHES DA EX
Linha	plataforma	total_cliques		
1	TikTok Ads	163693		
2	Meta Ads	294349		
3	Google Ads	2204694		

59

60

61

62

63

64

65

66

67

68

69

```
-- Calcular o custo de cada plataforma na tabela veiculos_consolidados
SELECT
    plataforma,
    SUM(custo) AS total_custo
FROM
    `prj-enext-bi-capacitacao.desafio_sql_time03.veiculos_consolidados`
WHERE
    plataforma IN ('Meta Ads', 'TikTok Ads', 'Google Ads')
GROUP BY
    plataforma;
```

Resultados da consulta

INFORMAÇÕES DO JOB	RESULTADOS	GRÁFICO	JSON	DETALHES DA EX
Linha	plataforma	total_custo		
1	TikTok Ads	34341.8		
2	Meta Ads	129462.89999999...		
3	Google Ads	655852.68999999...		