

Marketing Analyst Technical Assignment

Candidate: Minal Pawar

Role: Senior Marketing Analyst (Technical Assignment)

Tools Used: Google BigQuery, Looker Studio

Dashboard Link: <https://lookerstudio.google.com/reporting/341564c1-87b8-4181-b2ab-f11942e4f856>

1) Objective

The objective of this assignment is to unify multi-channel advertising data from Facebook Ads, Google Ads, and TikTok Ads into a standardized analytics model and build a one-page dashboard for cross-channel performance analysis. The goal is to enable consistent filtering, aggregation, and KPI reporting by platform, campaign, ad group, and date.

2) Data Sources

Datasets used (provided in assignment repository):

- 01_facebook_ads.csv
- 02_google_ads.csv
- 03_tiktok_ads.csv

Source: Marketing Analytics Assignments GitHub

<https://github.com/ej29-r3d/Marketing-Analytics-Assignments/tree/main/marketing-analyst-assignment>

3) BigQuery Setup

Step 1 — Access BigQuery

To access BigQuery:

1. Open: <https://console.cloud.google.com/bigquery>
2. Sign in using the Google account used for the assignment.
3. If prompted, select an existing project or create a new project.
4. If BigQuery isn't visible, navigate:  Menu → BigQuery → Studio
5. If prompted, enable:
 - BigQuery API (required)
 - Billing account (if required by your GCP configuration)

The screenshot shows the Google Cloud BigQuery Studio interface. On the left, there's a sidebar with navigation links like Overview, Studio, Agents, Pipelines and integration, Data transfers, Dataform, Scheduled queries, Scheduling, Governance, Sharing (Analytics Hub), Partner Centre, Settings, and Release notes. The 'Studio' link is currently selected. The main area has a search bar at the top right. Below it, there's a 'Create new' section with buttons for SQL query, Notebook, Notebook with Spark, ML model, Data canvas, Pipeline, Data preparation, Conversation, and Table. A 'Recent' section shows a table with columns for Display name, Type, Last modified time, and Project, which is currently empty ('No rows to display'). Below that is a 'Try with templates' section featuring three cards: 'Try the Google Trends demo query' (with a chart icon), 'Try the Colab demo notebook' (with a 'CO' logo), and 'Try the Spark demo notebook' (with a 'Spark' logo). Each card includes a brief description and a 'View notebook gallery' link.

This screenshot is identical to the one above, showing the Google Cloud BigQuery Studio interface with the 'marketing-analytics-487401' project selected. The left sidebar, main search bar, 'Create new' section, 'Recent' table, and 'Try with templates' section are all the same as in the first screenshot.

Step 2 — Create Dataset

Created dataset inside the project:

- Dataset ID: marketing_analytics_assignment
- Location: US (default)

Steps:

1. In BigQuery left panel, locate project: marketing-analytics-487401

2. Click : (three dots) next to project → Create dataset
3. Enter dataset id and click Create dataset

The screenshot shows the Google Cloud BigQuery Studio interface. On the left, there's a sidebar with various Google Cloud services. The main area shows the 'marketing_analytics_assignment' dataset under the 'marketing-analytics-487401' project. The 'Tables' tab is selected, and it shows a single table named 'marketing_analytics_assignment'. There are tabs for 'Overview', 'Details', 'Insights', and 'Preview'.

The screenshot shows the 'Create table' dialog in Google Cloud BigQuery Studio. The 'Source' section is set to 'Upload' and 'File format' is 'CSV'. A file named '01_facebook_ads.csv' is selected. The 'Destination' section shows the 'Project' as 'marketing-analytics-487401', 'Dataset' as 'marketing_analytics_assignment', and 'Table' as 'facebook_ads_raw'. The 'Schema' section has 'Auto-detect' enabled. At the bottom are 'Create table' and 'Cancel' buttons.

Step 3 — Upload CSVs into BigQuery Raw Tables

Uploaded each CSV into BigQuery with:

- Source: Upload
- File format: CSV
- Schema: Auto-detect enabled
- Header rows to skip: 1

Created 3 raw tables:

1. facebook_ads_raw (from 01_facebook_ads.csv)
2. google_ads_raw (from 02_google_ads.csv)
3. tiktok_ads_raw (from 03_tiktok_ads.csv)

The screenshot shows the Google BigQuery Studio interface. On the left, the sidebar includes sections for Overview, Studio, Agents, Pipelines and integration, Data transfers, Dataform, Scheduled queries, Scheduling, Governance, Sharing (Analytics Hub), Partner Centre, Settings, and Release notes. The main area displays a dataset named 'marketing_analytics_assignment'. Under this dataset, there are three tables: 'facebook_ads_raw', 'google_ads_raw', and 'tiktok_ads_raw', all of which were created on '13 Feb 2020' at '14 Apr 2020'. The 'facebook_ads_raw' table has a type of 'Table' and an 'Expiry time' of 'None'. The 'google_ads_raw' and 'tiktok_ads_raw' tables also have types of 'Table' and expiry times of 'None'.

Table ID	Type	Create time	Expiry time	Label
facebook_ads_raw	Table	13 Feb 20...	14 Apr 20...	None
google_ads_raw	Table	13 Feb 20...	14 Apr 20...	None
tiktok_ads_raw	Table	13 Feb 20...	14 Apr 20...	None

4) Unified Data Model (SQL Transformations)

4.1 Goal of the transformation

To enable cross-channel analysis, I standardized Facebook, Google Ads, and TikTok datasets into a single unified schema with consistent column names and data types. This allows seamless aggregation and filtering by platform, campaign, ad group, and date.

4.2 Platform Clean Views

Created 3 “clean” views, one per platform:

- facebook_ads_clean
- google_ads_clean
- tiktok_ads_clean

Each clean view:

- Adds a platform field (facebook, google, tiktok)
- Renames platform-specific identifiers to shared fields (adgroup_id, adgroup_name)
- Casts metrics into correct types using SAFE_CAST
 - impressions/clicks/conversions → INT64
 - spend/cost → NUMERIC
- Sets fields unavailable for that platform to NULL to maintain schema consistency

This ensures each platform outputs the same columns, making it safe to union.

The screenshot shows the Google Cloud BigQuery interface. On the left, the navigation pane is open with sections like Overview, Studio, Pipelines and integration, Data transfers, Dataform, Scheduled queries, Scheduling, Governance, Sharing (Analytics Hub), Partner Centre, Settings, and Release notes. The Studio section is currently selected.

In the center, there's a search bar and a query editor titled "Untitled query". The query is:

```
SELECT FROM marketing-analytics:487401.marketing_analytics_assignment.facebook_ads_raw LIMIT 1000
```

A red error message at the bottom of the query editor says: "Syntax error: SELECT list must not be empty at [1:9]".

To the right of the query editor is a "Reference" panel for the table "facebook_ads_raw". It shows the following schema:

Field name	Type
date	DATE
campaign_id	STRING
campaign_name	STRING
ad_set_id	STRING
ad_set_name	STRING
impressions	INTEGER
clicks	INTEGER
spend	FLOAT

The screenshot shows the Google Cloud BigQuery interface. On the left, the navigation sidebar includes sections like Overview, Preview, Studio, Agents, Pipelines and integration, Data transfers, Dataform, Scheduled queries, Scheduling, Governance, Sharing (Analytics Hub), Partner Centre, Settings, and Release notes. The main workspace displays an 'Explorer' view with a search bar and a list of resources. A specific query titled 'Untitled query' is running, showing SQL code for selecting data from a source and creating views for 'marketing'. The results table shows five rows of data with columns Status, End time, SQL, and Action. To the right, a 'Reference' panel lists field names and their types: date (DATE), platform (STRING), campaign_id (STRING), campaign_name (STRING), adgroup_id (STRING), adgroup_name (STRING), impressions (INTEGER), clicks (INTEGER), spend (NUMERIC), conversions (INTEGER), conversion_value (NUMERIC), video_views (INTEGER), engagement_rate (NUMERIC), and reach (INTEGER). The status of all five rows is 'Success'.

Table ID	Type	Create time	Expiry time	Label
ads_unified	View	14 Feb 20...	15 Apr 20...	None
ads_unified_metrics	View	14 Feb 20...	15 Apr 20...	None
facebook_ads_clean	View	14 Feb 20...	15 Apr 20...	None
google_ads_clean	Table	13 Feb 20...	14 Apr 20...	None
google_ads_raw	Table	13 Feb 20...	14 Apr 20...	None
tiktok_ads_clean	View	14 Feb 20...	15 Apr 20...	None
tiktok_ads_raw	Table	13 Feb 20...	14 Apr 20...	None

4.3 Unified Dataset Layer — ads_unified

Created a unified cross-channel view called ads_unified using UNION ALL across the 3 clean views.

Why UNION ALL:

- Advertising metrics are additive
- We must retain all records without deduplicating rows

This produces one cross-channel fact table where each row represents:
platform + campaign + ad group + date performance.

Field name	Type	Mode	Description	Key	Collation	Default value	Policy tags
date	DATE	NULLABLE	-	-	-	-	-
platform	STRING	NULLABLE	-	-	-	-	-
campaign_id	STRING	NULLABLE	-	-	-	-	-
campaign_name	STRING	NULLABLE	-	-	-	-	-
adgroup_id	STRING	NULLABLE	-	-	-	-	-
adgroup_name	STRING	NULLABLE	-	-	-	-	-
impressions	INTEGER	NULLABLE	-	-	-	-	-
clicks	INTEGER	NULLABLE	-	-	-	-	-
spend	NUMERIC	NULLABLE	-	-	-	-	-
conversions	INTEGER	NULLABLE	-	-	-	-	-
conversion_value	NUMERIC	NULLABLE	-	-	-	-	-

4.4 Metrics Layer — ads_unified_metrics

Created ads_unified_metrics as the dashboard-ready layer. It includes calculated KPIs:

- CTR = clicks / impressions
- CPC = spend / clicks
- CPA = spend / conversions
- CVR = conversions / clicks
- ROAS = conversion_value / spend (*when conversion_value exists; primarily Google*)

Also retains platform-specific fields:

- Facebook: engagement_rate, reach, frequency
- Google: quality_score, search_impression_share
- TikTok: video_watch_*, likes, shares, comments

This enables a single Looker Studio dashboard to show both standardized KPIs and channel-specific insights.

KPI Definitions (How metrics are calculated)

Base metrics (from unified model)

These fields come directly from the unified dataset and are aggregated in the dashboard (typically SUM):

- Spend = SUM(spend)
- Impressions = SUM(impressions)
- Clicks = SUM(clicks)
- Conversions = SUM(conversions)

Calculated KPIs (platform-neutral)

1) CTR (Click-Through Rate)

Definition: % of impressions that resulted in a click

Formula (recommended ratio-of-totals):

- CTR = SUM(clicks) / SUM(impressions)

Looker Studio calculated field used:

IF(SUM(impressions)=0, 0, SUM(clicks)/SUM(impressions))

Format: Percent

2) CPC (Cost per Click)

Definition: average cost paid per click

Formula:

- $CPC = \text{SUM}(\text{spend}) / \text{SUM}(\text{clicks})$

Looker Studio calculated field used:

$\text{IF}(\text{SUM}(\text{clicks})=0, 0, \text{SUM}(\text{spend})/\text{SUM}(\text{clicks}))$

Format: Currency

3) CPA (Cost per Acquisition / Cost per Conversion)

Definition: average cost paid per conversion

Formula:

- $CPA = \text{SUM}(\text{spend}) / \text{SUM}(\text{conversions})$

Looker Studio calculated field used:

$\text{IF}(\text{SUM}(\text{conversions})=0, 0, \text{SUM}(\text{spend})/\text{SUM}(\text{conversions}))$

Format: Currency

4) CVR (Conversion Rate)

Definition: % of clicks that resulted in a conversion

Formula:

- $CVR = \text{SUM}(\text{conversions}) / \text{SUM}(\text{clicks})$

Looker Studio calculated field used:

$\text{IF}(\text{SUM}(\text{clicks})=0, 0, \text{SUM}(\text{conversions})/\text{SUM}(\text{clicks}))$

Format: Percent

5) ROAS (Return on Ad Spend) (*available where conversion_value exists — mainly Google Ads*)

Definition: revenue/value returned per \$1 of ad spend

Formula:

- $ROAS = \text{SUM}(\text{conversion_value}) / \text{SUM}(\text{spend})$

In BigQuery metrics view (`ads_unified_metrics`):

`SAFE_DIVIDE(conversion_value, spend) AS calc_roas`

Why KPI calculations were done this way

KPIs were calculated as **ratio-of-totals** (e.g., $\text{SUM}(\text{clicks})/\text{SUM}(\text{impressions})$) to avoid distortion that can happen if averaging daily rates. This makes metrics consistent across different date ranges and filters.

BigQuery uses **SAFE_DIVIDE** to avoid errors from divide-by-zero, and Looker Studio uses **IF()** logic for the same reason in dashboard-level calculated fields.

The screenshot shows the Google BigQuery interface. On the left, the navigation pane includes sections like Overview, Studio, Agents, Pipelines and integration, Data transfers, Dataform, Scheduled queries, Scheduling, Governance, Sharing (Analytics Hub), Partner Centre, Settings, Release notes, and Pipelines. The Studio section is selected. In the center, the schema of the 'ads_unified_metrics' table is displayed under the 'marketing-analytics-487401' dataset. The table has 12 columns: conversion_value, video_views, calc_ctr, calc_cpc, calc_cpa, calc_cvr, calc_ros, engagement_rate, reach, frequency, google_ctr, and google_avg_cpc. The 'quality_score' column is highlighted. At the bottom of the schema view, there are 'Edit schema' and 'Describe data' buttons.

5) Data Quality & Validation Checks

5.1 Row counts by platform

Validated completeness of the unified dataset:

```
SELECT platform, COUNT(*) AS row_count
FROM `marketing_analytics_assignment.ads_unified`
GROUP BY platform
ORDER BY row_count DESC;
```

The screenshot shows the Google Cloud Platform interface with the 'Cloud Hub' selected in the sidebar. The main area displays a query titled 'Row counts' with the following SQL code:

```
1 SELECT platform,
2    COUNT(*) AS row_count
3 FROM `marketing_analytics_assignment.ads_unified`
4 GROUP BY platform
5 ORDER BY row_count DESC;
```

The results of the query are shown in a table:

platform	row_count
google	110
facebook	110
tiktok	110

On the right side, there is a 'Reference' panel showing the schema of the 'ads_unified_metrics' table:

Field name	Type
date	DATE
platform	STRING
campaign_id	STRING
campaign_name	STRING
adgroup_id	STRING
adgroup_name	STRING
impressions	INTEGER
clicks	INTEGER
spend	NUMERIC

5.2 Totals by platform

Validated key totals to ensure aggregations work correctly:

```
SELECT
  platform,
  SUM(spend) AS total_spend,
  SUM(impressions) AS total_impressions,
  SUM(clicks) AS total_clicks,
  SUM(conversions) AS total_conversions
FROM `marketing_analytics_assignment.ads_unified`
GROUP BY platform;
```

The screenshot shows the Google Cloud Marketing Analytics interface. On the left, there's a navigation sidebar with various options like Home, Deployments, Health and troubleshooting, Optimisation, Quotas and reservations, Maintenance, and Support. The main area is titled 'Totals by platform' and contains a code editor with the SQL query provided above. Below the code editor is a note: 'This query will process 10.31 KB when run.' and 'Using on-demand processing quota'. To the right of the code editor is a 'Reference' pane showing the schema for 'ads_unified_metrics' from the table 'marketing-analytics-487401.marketing_analytics_assignment.ads_unified_metrics'. The schema includes fields: date (DATE), platform (STRING), campaign_id (STRING), campaign_name (STRING), adgroup_id (STRING), adgroup_name (STRING), impressions (INTEGER), clicks (INTEGER), and spend (NUMERIC). At the bottom of the reference pane, it says '0 rows • 0 B • Last modified at 14 Feb 2026, 13:09'. The 'Results' tab is selected in the 'Query results' section, which displays the following data:

Row	platform	total_spend	total_impressions	total_clicks	total_conversions
1	facebook	18292	4541474	88899	2395
2	google	37686.2	7223544	137590	4218
3	tiktok	74266.7	28708167	461844	6750

5.3 Date range validation

Confirmed min/max dates for correct reporting and dashboard filtering:

```
SELECT MIN(date) AS min_date, MAX(date) AS max_date
FROM `marketing_analytics_assignment.ads_unified`;
```

The screenshot shows the Google Cloud BigQuery interface. On the left, there's a navigation sidebar with options like Home, Deployments, Health and troubleshooting, Optimisation, Quotas and reservations, Maintenance, and Support. The main area has a search bar at the top. Below it, a query editor window displays a SQL query:

```
1 SELECT MIN(date) AS min_date, MAX(date) AS max_date
2 FROM `marketing_analytics_assignment.ads_unified`;
```

The status bar indicates "Query completed". To the right, a "Reference" panel shows the schema for the table "ads_unified_metrics" from the dataset "marketing-analytics-487401". The schema includes fields: date (DATE), platform (STRING), campaign_id (STRING), campaign_name (STRING), adgroup_id (STRING), adgroup_name (STRING), impressions (INTEGER), clicks (INTEGER), and spend (NUMERIC). A "Results" tab is selected, showing one row of data: min_date (2024-01-01) and max_date (2024-01-30).

6) Dashboard Build (Looker Studio)

6.1 Connect Looker Studio to BigQuery

Steps:

1. Open Looker Studio
2. Create → Report
3. Connector: BigQuery
4. Select:
 - Project: your GCP project
 - Dataset: marketing_analytics_assignment
 - Table/View: ads_unified_metrics
5. Click Add to report

The screenshot shows the Looker Studio report creation interface. At the top, there's a browser header with the URL "lookerstudio.google.com/u/0/reporting/bfb8765a-727f-456a-8ac8-336a42accd56/page/mRjofF/edit". The main area is titled "Untitled Report". The "Add data to report" section is open, showing a "BigQuery" connector. The "Project" dropdown is set to "marketing-analytics-487401" and "My First Project". The "Data set" dropdown is empty, showing the message "There are no datasets available in this project." There are "Cancel" and "Add" buttons at the bottom right.

6.2 Dashboard Layout (One Page)

Dashboard includes filters, KPIs, trends, comparisons, and campaign drill-down.

Top Row: KPI Scorecards

- Total Spend (SUM spend)
- Total Impressions (SUM impressions)
- Total Clicks (SUM clicks)
- Total Conversions (SUM conversions)
- CTR (calculated field in Looker Studio)
- CPC (calculated field in Looker Studio)
- CPA (calculated field in Looker Studio)
- CVR (calculated field in Looker Studio)

Middle: Trend + Channel Comparison

- Trend chart (Combo): Spend over time + Conversions over time
- Platform performance bar chart (Spend and conversions)
- CPA by platform

Bottom: Campaign Table

- campaign_name performance table with metrics:
 - spend, clicks, conversions, CTR, CPC, CPA, CVR
- Sorted by spend descending for quick identification of top campaigns

Filters

- Date range control
- Platform dropdown
- Campaign dropdown

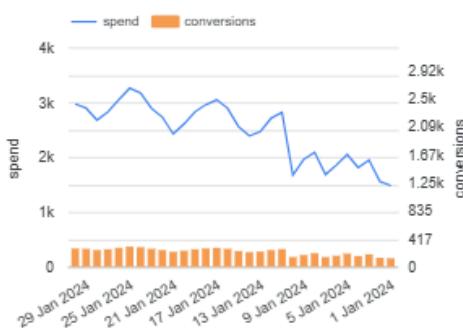
Cross-Channel Ads Performance Dashboard (Spend, Conversions, CPA & CTR)

Track multi platforms, ad spend, conversions, CPA and CTR performance across Tik Tok, Facebook and Google

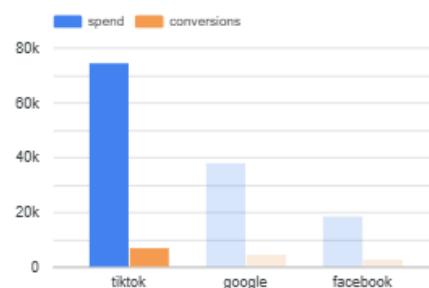
1 Jan 2024 - 31 Jan 2024 - platform campaign

Total Spend	Impressions	Clicks	Conversions	CTR	CPC	CPA	CVR
74.3k	28.7m	461.8k	6.8k	1.6%	\$0.16	\$11.00	1.46%

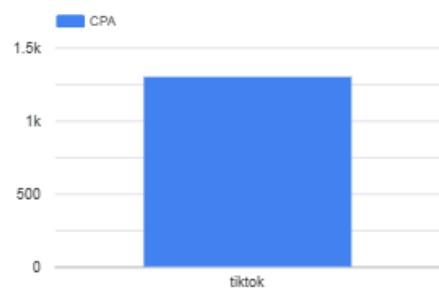
Trend chart (Spend + Conversions over time)



Platform performance (Spend + Conversions)



CPA by platform



Top campaigns (with KPIs)

	campaign_name	platform	Spend	impressions	clicks	CPA	CTR
1.	Influencer_Collab	tiktok	\$28,312.3	10,513,148	170,684	\$257.85	42.22%
2.	Conversion_Focus	tiktok	\$20,805.9	3,970,406	75,914	\$270.1	51.59%
3.	Awareness_GenZ	tiktok	\$15,640.0	8,059,087	118,539	\$390.59	44.12%
4.	Traffic_Campaign	tiktok	\$11,708.5	6,165,526	98,707	\$380.71	42.34%

1 - 4 / 4 < >

7) Key Insights (3–5)

- Efficiency varies by platform (CPA differences): CPA differs across channels, indicating opportunities to optimize budget allocation and improve underperforming channels.
- Spend and conversions move together over time: Trend analysis highlights periods where increased spend aligns with conversion lift and periods where it doesn't, signaling efficiency opportunities.
- Performance is concentrated in top campaigns: A subset of campaigns contributes most of the spend and conversions, allowing prioritization of winners and optimization of the long tail.
- CTR and CVR provide different funnel signals: CTR reflects engagement/traffic quality, while CVR reflects post-click effectiveness—both are needed for end-to-end performance evaluation.
- Unified reporting improves consistency: Standardized KPIs across platforms allow accurate cross-channel comparisons at campaign and ad group levels.