

Social Media Marketing Dashboard

1.0 Introduction

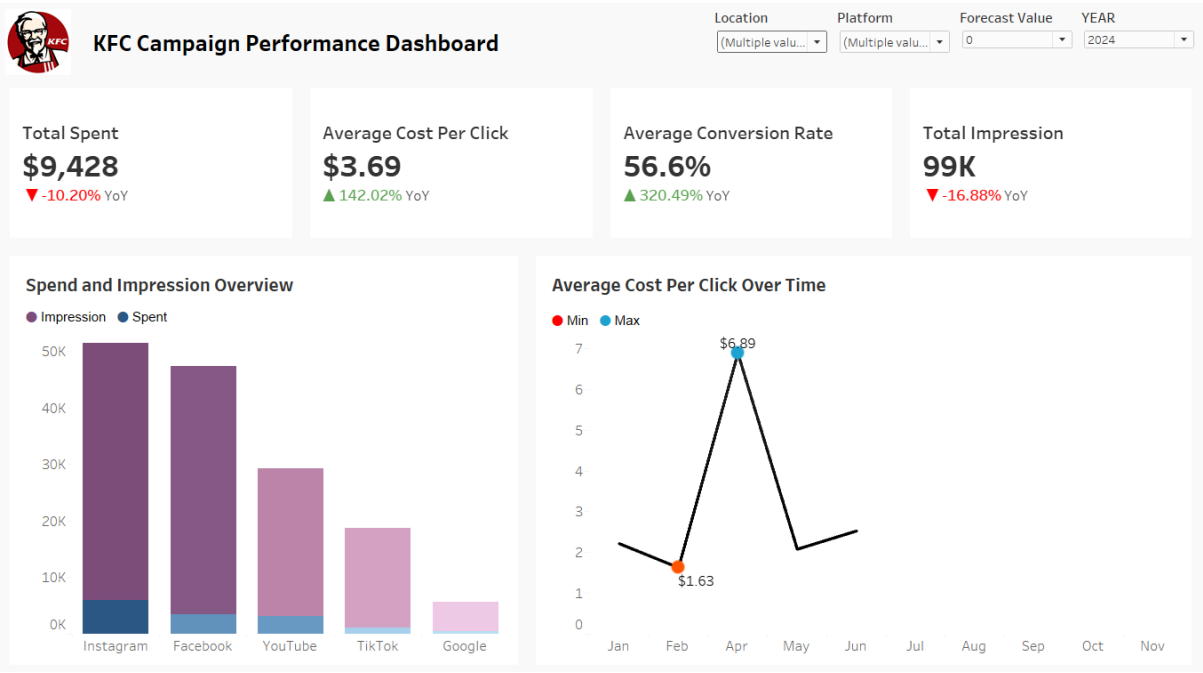
Title: KFC Campaign Performance Dashboard

Purpose: Provide Insights on Marketing campaign by KFC

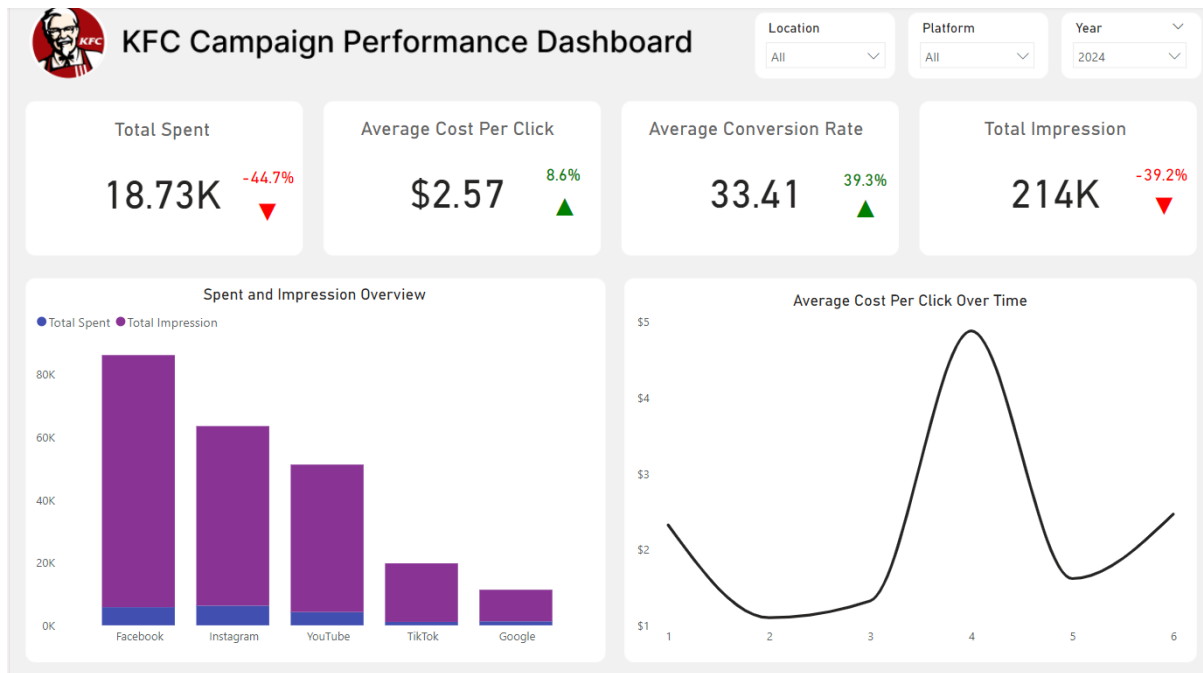
Audience: KFC Managers, Stakeholders

2.0 Dashboard Overview

2.0.1 Tableau Dashboard



2.0.2 PowerBI Dashboard



3.0 Data Sources

Origin: SQL Server with fact and dimension tables

4.0 Data Fields and Calculations

For Tableau and Power BI datafields are same.

Tables Description:

1.0 Fact_camp

- Date
- Location_id
- Platform_id
- Ads_run
- Impressions
- Clicks
- Conversions
- Total_spent
- Cost_per_click

- j. Click_through_rate
- k. Conversion_rate
- l. Cost_per_conversion

2.0 dim_product

- a) Platform_id
- b) Platform

3.0 dim_location

- a) Location_id
- b) Location

4.0 dim_date

- a) Date
- b) Weekday
- c) Day
- d) Month
- e) Year

4.0.1 Tableau Calculations

1. KPI Total Spent

Total Spent

\$9,428

▼ -10.20% YoY

Positive Spent

```
IF [%Diff Total Spent] > 0 THEN [%Diff Total Spent] END
```

Negative Spent

```
IF [%Diff Total Spent] < 0 THEN [%Diff Total Spent] END
```

CY Total Spent

```
IF [Forecast Value] > 1 THEN
    IF YEAR([Date]) = [Selected Year] THEN
        [Total Spent] * (1 + [Forecast Value] / 100)
    END
ELSE
    IF YEAR([Date]) = [Selected Year] THEN [Total Spent] END
END
```

PY Total Spent

```
IF [Year] = [Selected Year] - 1 THEN [Total Spent] END
```

%Diff Total Spent

```
(SUM([CY Total Spent]) - SUM([PY Total Spent])) /
SUM([PY Total Spent])
```

2. KPI Avg Cost Per Click

Average Cost Per Click

\$3.69

▲ 142.02% YoY

Avg CY CPC

AVG([CY CPC])

CY CPC

```
IF [Forecast Value] > 0 THEN
  IF [Year] = [Selected Year] THEN
    [Cost Per Click] *
    (1 + [Forecast Value] / 100 *
    {FIXED : CORR([Total Spent], [Cost Per Click])})
  END
ELSE
  IF [Year] = [Selected Year] THEN [Cost Per Click] END
END
```

Positive Avg CPC

```
IF [%Diff Avg CPC] > 0 THEN [%Diff Avg CPC] END
```

Negative Avg CPC

```
IF [%Diff Avg CPC] < 0 THEN [%Diff Avg CPC] END
```

PY CPC

```
IF [Year] = [Selected Year] - 1 THEN [Cost Per Click] END
```

%Diff Avg CPC

([Avg CY CPC] - [Avg PY CPC]) / [Avg PY CPC]

3. KPI Conversion Rate

Average Conversion Rate

56.6%

▲ 320.49% YoY

Avg CY Conversion Rate

`AVG([CY Conversion Rate])`

CY Conversion Rate

```
IF [Forecast Value] > 0 THEN
  IF [Year] = [Selected Year] THEN
    [Conversion Rate] *
    (1 + [Forecast Value] / 100 *
    {FIXED : CORR([Total Spent], [Conversion Rate])})
  END
ELSE
  IF [Year] = [Selected Year] THEN [Conversion Rate] END
END
```

Positive Conversion Rate

`[%Diff Avg Conver. Rate] > 0 THEN [%Diff Avg Conver. Rate] END`

Negative Conversion Rate

`IF [%Diff Avg Conver. Rate] < 0 THEN [%Diff Avg Conver. Rate] END`

PY Conversion Rate

`IF [Year] = [Selected Year] - 1 THEN [Conversion Rate] END`

%Diff Avg Conver. Rate

`(([Avg CY Conversion Rate] - [Avg PY Conversion Rate])
/ [Avg PY Conversion Rate])`

4. KPI Total Impression

Total Impression

99K

▼ -16.88% YoY

CY Impression

```
IF [Forecast Value] > 0 THEN
  IF [Year] = [Selected Year] THEN
    [Impressions] *
    (1 + [Forecast Value] / 100 *
    {FIXED : CORR([Total Spent], [Impressions])})
  END
ELSE
  IF [Year] = [Selected Year] THEN [Impressions] END
END
```

Positive Impression

```
IF [%Diff Impression] > 0 THEN [%Diff Impression] END
```

Negative Impression

```
IF [%Diff Impression] < 0 THEN [%Diff Impression] END
```

PY Impression

```
IF [Year] = [Selected Year] - 1 THEN [Impressions] END
```

%Diff Impression

```
(SUM([CY Impression]) - SUM([PY Impression])) / SUM([PY Impression])
```

5. Avg CPC Line Chart

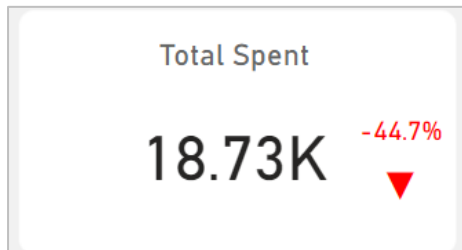
Min/Max CPC

Results are computed along Table (across).

```
IF AVG([CY CPC]) = WINDOW_MIN([Avg CY CPC])
THEN [Avg CY CPC]
ELSEIF [Avg CY CPC] = WINDOW_MAX([Avg CY CPC])
THEN [Avg CY CPC]
END
```

4.0.2 Power BI Measures

I'm only keeping in docs for one kpi as its too long



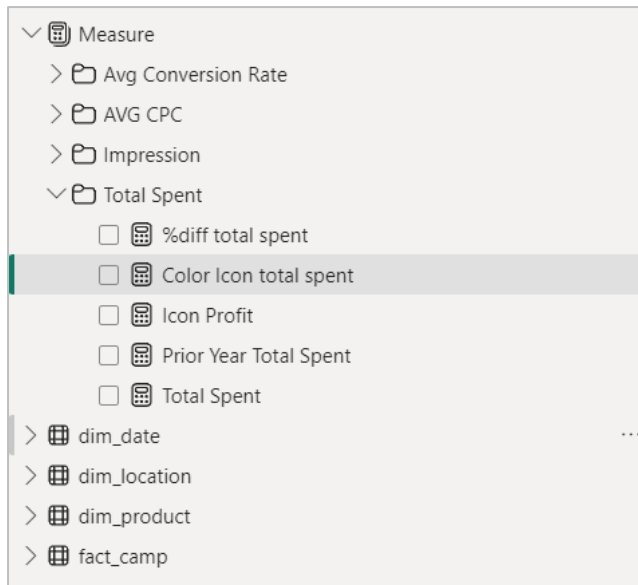
```
Total Spent = SUM('fact_camp'[total_spent])
```

```
1 Prior Year Total Spent =
2 VAR selected_year = SELECTEDVALUE('dim_date'[year])
3
4 RETURN
5 CALCULATE('Measure'[Total Spent],
6     FILTER(
7         ALL('dim_date'),
8         'dim_date'[year] = selected_year - 1
9     )
10 )
11
```

```
1 %diff total spent = DIVIDE('Measure'[Total Spent] - 'Measure'[Prior Year Total Spent], 'Measure'[Prior Year Total Spent], 0)
```

```
1 Icon Profit =
2 VAR positive_icon = UNICHAR(9650)
3 VAR negative_icon = UNICHAR(9660)
4 VAR neutral_icon = UNICHAR(10134)
5
6 VAR result =
7     IF('Measure'[%diff total spent] > 0, positive_icon,
8     IF('Measure'[%diff total spent] < 0, negative_icon, neutral_icon))
9
10 return result
```

```
1 Color Icon total spent = IF('Measure'[%diff total spent] > 0, "Green",
2 IF('Measure'[%diff total spent] < 0, "Red", "Orange"))
```

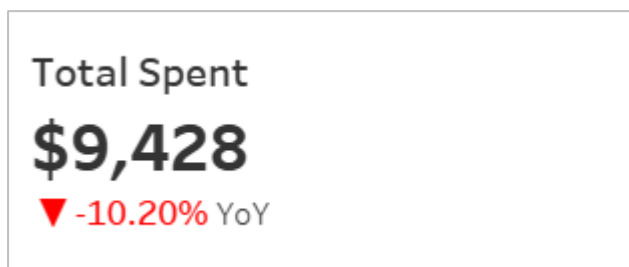
LESSON – NEVER KEEP STORAGE MODE TO MIXED

5.0 SQL Validations

NOTE: In tableau in filters

- a. **Location filter** (*Bhaktapur, Kathmandu is only selected*)
- b. **Platform Filter** (*Facebook, Instagram is only selected*)

1. Total Spent KPI (doesn't include YoY)



```

1  -- Sp Total Spent
2
3  CREATE OR ALTER PROC spTotalSpent
4      @year int = 2024,
5      @platforms nvarchar(max) = NULL,
6      @locations nvarchar(max) = NULL
7  AS
8  BEGIN
9
10     DECLARE @sql nvarchar(max);
11     DECLARE @platformsFilter nvarchar(max);
12     DECLARE @locationsFilter nvarchar(max);
13
14     -- handle platform is null or not provided
15     IF @platforms IS NULL OR @platforms = ''
16     BEGIN
17         -- get all platforms if @platforms is null or empty
18         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ''), ''
19         FROM dim_product
20     END
21     ELSE
22     BEGIN
23         SET @platformsFilter = @platforms;
24     END
25
26
27     -- handle locations
28     IF @locations IS NULL OR @locations = ''
29     BEGIN
30         -- get all locations if @locations is null or empty
31         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ''), ''
32         FROM dim_location
33     END
34     ELSE
35     BEGIN
36         SET @locationsFilter = @locations
37     END
38
39
40     -- base query
41     SET @sql = '
42
43     select SUM(total_spent) as total_spent
44     from fact_camp fc
45     INNER JOIN dim_product dp
46     ON dp.platform_id = fc.platform_id
47     INNER JOIN dim_location dl
48     ON dl.location_id = fc.location_id
49     WHERE YEAR(date) = @years
50         AND dp.platform IN (' + @platformsFilter + ')
51         AND dl.location IN (' + @locationsFilter + ')
52     '
53
54     EXEC sp_executesql @sql, N'@years INT', @years = @year;
55
56 END
57
58 exec spTotalSpent @year = 2024,
59     @platforms = ''facebook'', 'instagram'',
60     @locations = ''bhaktapur'', 'kathmandu'';

```

	total_spent	
1	9428.42	

KPI Total Spent YoY

```
1  -- KPI Total Spent
2  CREATE OR ALTER PROC spKPITotalSpent
3  @year int = 2024,
4  @platforms nvarchar(max) = NULL,
5  @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ''
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ''
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations;
35     END
36
37     -- base query
38     SET @sql = '
39
40     WITH cy_total_spent_cte
41     AS
42     (
43         SELECT
44             SUM(f.total_spent) as cy_total_spent
45         FROM fact_camp f
46         INNER JOIN dim_product dp
47             ON f.platform_id = dp.platform_id
48         INNER JOIN dim_location dl
49             ON f.location_id = dl.location_id
50         WHERE YEAR(date) = @years
51             AND dp.platform IN (' + @platformsFilter + ')
52             AND dl.location IN (' + @locationsFilter + ')
53     ),
54     py_total_spent_cte
55     AS
56     (
57         SELECT
58             SUM(total_spent) as py_total_spent
59         FROM fact_camp f
60         INNER JOIN dim_product dp
61             ON f.platform_id = dp.platform_id
62         INNER JOIN dim_location dl
63             ON f.location_id = dl.location_id
64         WHERE YEAR(date) = @years - 1
65             AND dp.platform IN (' + @platformsFilter + ')
66             AND dl.location IN (' + @locationsFilter + ')
67     )
68
69     SELECT
70         ((cy_total_spent - py_total_spent) / py_total_spent)*100 as yoy_spent
71     FROM cy_total_spent_cte,py_total_spent_cte
72
73     '
74
75     EXEC sp_executesql @sql, N'@years INT', @years = @year;
76
77 END
78
79
80
81 exec spKPITotalSpent @year = 2024,
82 @platforms = ''facebook'', 'instagram'',
83 @locations = ''bhaktapur'', 'kathmandu'';
```

	yoy_spent
1	-10.203300

2. KPI Average Cost Per Click

Average Cost Per Click

\$3.69

▲ 142.02% YoY

```
1  -- SP Average Cost Per Click
2
3  CREATE OR ALTER PROC spAvgCostPerClick
4      @year int = 2024,
5      @platforms nvarchar(max) = NULL,
6      @locations nvarchar(max) = NULL
7  AS
8  BEGIN
9
10     DECLARE @sql nvarchar(max);
11     DECLARE @platformsFilter nvarchar(max);
12     DECLARE @locationsFilter nvarchar(max);
13
14     -- handle platform is null or not provided
15     IF @platforms IS NULL OR @platforms = ' '
16     BEGIN
17         -- get all platforms if @platforms is null or empty
18         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
19         FROM dim_product
20     END
21     ELSE
22     BEGIN
23         SET @platformsFilter = @platforms;
24     END
25
26     -- handle locations
27     IF @locations IS NULL OR @locations = ' '
28     BEGIN
29         -- get all locations if @locations is null or empty
30         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
31         FROM dim_location
32     END
33     ELSE
34     BEGIN
35         SET @locationsFilter = @locations
36     END
37
38     -- base query
39     SET @sql = '
40
41     SELECT ROUND(AVG(cost_per_click),2) as avg_cpc
42     from fact_camp fc
43     INNER JOIN dim_product dp
44     ON dp.platform_id = fc.platform_id
45     INNER JOIN dim_location dl
46     ON dl.location_id = fc.location_id
47     WHERE YEAR(date) = @years
48         AND dp.platform IN (' + @platformsFilter + ')
49         AND dl.location IN (' + @locationsFilter + ')
50     ,
51
52     EXEC sp_executesql @sql, N'@years INT', @years = @year;
53
54
55
56
57 END
58
59 exec spAvgCostPerClick @year = 2024,
60     @platforms = ''facebook'', 'instagram'',
61     @locations = ''bhaktapur'', 'kathmandu'';
```

	avg_cpc
1	3.690000

Avg Cost Per Click YoY

```
1  -- KPI Average Cost Per Click
2  CREATE OR ALTER PROC spKPIAvgCostPerClick
3  @year int = 2024,
4  @platforms nvarchar(max) = NULL,
5  @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ''
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ''
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations
35     END
36
37     -- base query
38     SET @sql = '
39
40     WITH cy_cpc_cte
41     AS
42     (
43         SELECT
44             avg(cost_per_click) as cy_cpc
45         FROM fact_camp f
46         INNER JOIN dim_product dp
47             ON f.platform_id = dp.platform_id
48         INNER JOIN dim_location dl
49             ON f.location_id = dl.location_id
50         WHERE YEAR(date) = @years
51             AND dp.platform IN (' + @platformsFilter + ')
52             AND dl.location IN (' + @locationsFilter + ')
53     ),
54     py_cpc_cte
55     AS
56     (
57         SELECT
58             avg(cost_per_click) as py_cpc
59         FROM fact_camp f
60         INNER JOIN dim_product dp
61             ON f.platform_id = dp.platform_id
62         INNER JOIN dim_location dl
63             ON f.location_id = dl.location_id
64         WHERE YEAR(date) = @years - 1
65             AND dp.platform IN (' + @platformsFilter + ')
66             AND dl.location IN (' + @locationsFilter + ')
67     )
68
69     SELECT
70         ((cy_cpc - py_cpc) / py_cpc)*100 as yoy_spent
71     FROM cy_cpc_cte, py_cpc_cte
72
73     '
74
75     EXEC sp_executesql @sql, N'@years INT', @years = @year;
76
77 END
78
79 exec spKPIAvgCostPerClick @year = 2024,
80 @platforms = ''facebook'', 'instagram'',
81 @locations = ''bhaktapur'', 'kathmandu'';
```

	yoy_spent
1	142.016600

3. KPI Average Conversion Rate

Average Conversion Rate

56.6%

▲ 320.49% YoY

```
1  -- SP Average Conversion Rate
2  CREATE OR ALTER PROC spAvgConversionRate
3      @year int = 2024,
4      @platforms nvarchar(max) = NULL,
5      @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ' '
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ' '
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations
35     END
36
37
38     -- base query
39     SET @sql = '
40
41     SELECT ROUND(AVG(conversion_rate),0) as conversion_rate
42     FROM fact_camp fc
43     INNER JOIN dim_product dp
44     ON dp.platform_id = fc.platform_id
45     INNER JOIN dim_location dl
46     ON dl.location_id = fc.location_id
47     WHERE YEAR(date) = @years
48         AND dp.platform IN (' + @platformsFilter + ')
49         AND dl.location IN (' + @locationsFilter + ')
50
51     '
52
53     EXEC sp_executesql @sql, N'@years INT', @years = @year;
54
55 END
56
57 exec spAvgConversionRate @year = 2024,
58     @platforms = ''facebook'', 'instagram'',
59     @locations = ''bhaktapur'', 'kathmandu'';
```

	conversion_rate	
1	57.000000	

KPI Conversion Rate YoY

```
1  -- KPI Average Conversion Rate
2  CREATE OR ALTER PROC spKPIConversionRate
3  @year int = 2024,
4  @platforms nvarchar(max) = NULL,
5  @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ''
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ''
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations;
35     END
36
37     -- base query
38     SET @sql = '
39
40     WITH cy_avg_conv_rate_cte
41     AS
42     (
43         SELECT
44             avg(conversion_rate) as cy_conv_rate
45         FROM fact_camp f
46         INNER JOIN dim_product dp
47             ON f.platform_id = dp.platform_id
48         INNER JOIN dim_location dl
49             ON f.location_id = dl.location_id
50         WHERE YEAR(date) = @years
51             AND dp.platform IN (' + @platformsFilter + ')
52             AND dl.location IN (' + @locationsFilter + ')
53     ),
54     py_avg_conv_rate_cte
55     AS
56     (
57         SELECT
58             avg(conversion_rate) as py_conv_rate
59         FROM fact_camp f
60         INNER JOIN dim_product dp
61             ON f.platform_id = dp.platform_id
62         INNER JOIN dim_location dl
63             ON f.location_id = dl.location_id
64         WHERE YEAR(date) = @years - 1
65             AND dp.platform IN (' + @platformsFilter + ')
66             AND dl.location IN (' + @locationsFilter + ')
67     )
68
69     SELECT
70         ((cy_conv_rate - py_conv_rate) / py_conv_rate)*100 as yoy_spent
71     FROM cy_avg_conv_rate_cte, py_avg_conv_rate_cte
72
73     '
74
75     EXEC sp_executesql @sql, N'@years INT', @years = @year;
76
77 END
78
79 exec spKPIConversionRate @year = 2024,
80 @platforms = ''facebook'', 'instagram'',
81 @locations = ''bhaktapur'', 'kathmandu'';
```

	yoy_spent	
1	320.488500	

4. KPI Total Impression

Total Impression

99K

▼-16.88% YoY

```
1  -- SP Total Impressions
2  CREATE OR ALTER PROC spTotalImpression
3      @year int = 2024,
4      @platforms nvarchar(max) = NULL,
5      @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ' '
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ' '
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations;
35     END
36
37     -- base query
38     SET @sql = '
39
40     SELECT SUM(impressions) as total_impression
41     FROM fact_camp fc
42     INNER JOIN dim_product dp
43     ON dp.platform_id = fc.platform_id
44     INNER JOIN dim_location dl
45     ON dl.location_id = fc.location_id
46     WHERE YEAR(date) = @years
47         AND dp.platform IN (' + @platformsFilter + ')
48         AND dl.location IN (' + @locationsFilter + ')
49     '
50
51     EXEC sp_executesql @sql, N'@years INT', @years = @year;
52
53 END
54
55 exec spTotalImpression @year = 2024,
56     @platforms = ''facebook'', 'instagram'',
57     @locations = ''bhaktapur'', 'kathmandu'';
```

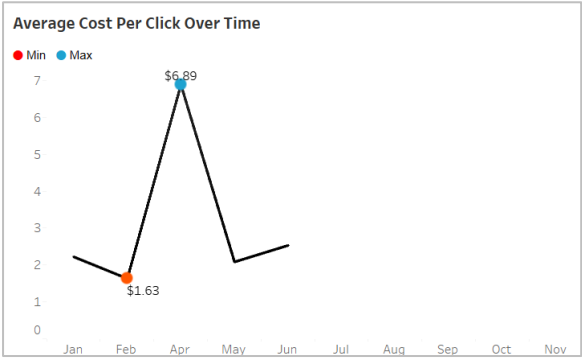
	total_impression	
1	98825	

KPI impression YoY

```
1  -- KPI Total Impression YoY
2  CREATE OR ALTER PROCEDURE spKPITotalImpression
3  @year int = 2024,
4  @platforms nvarchar(max) = NULL,
5  @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ' '
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ' '
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations;
35     END
36
37     -- base query
38     SET @sql = '
39     WITH cy_total_impression_cte AS (
40         SELECT
41             SUM(CAST(impressions AS DECIMAL(10,2))) AS cy_impression
42         FROM fact_camp f
43         INNER JOIN dim_product dp ON f.platform_id = dp.platform_id
44         INNER JOIN dim_location dl ON f.location_id = dl.location_id
45         WHERE YEAR(date) = @year
46             AND dp.platform IN (' + @platformsFilter + ')
47             AND dl.location IN (' + @locationsFilter + ')
48     ),
49     py_total_impression_cte AS (
50         SELECT
51             SUM(CAST(impressions AS DECIMAL(10,2))) AS py_impression
52         FROM fact_camp f
53         INNER JOIN dim_product dp ON f.platform_id = dp.platform_id
54         INNER JOIN dim_location dl ON f.location_id = dl.location_id
55         WHERE YEAR(date) = @year - 1
56             AND dp.platform IN (' + @platformsFilter + ')
57             AND dl.location IN (' + @locationsFilter + ')
58     )
59     SELECT
60         ((CAST(cy_impression AS DECIMAL(10,2)) - py_impression) / NULLIF(py_impression, 0))*100 AS yoy_spent
61     FROM cy_total_impression_cte
62     CROSS JOIN py_total_impression_cte
63     '
64
65     EXEC sp_executesql @sql, N'@year INT', @year = @year;
66
67 END
68
69
70
71 exec spKPITotalImpression @year = 2024,
72     @platforms = ''facebook'', ''instagram'',
73     @locations = ''bhaktapur'', ''kathmandu'';
```

	yoy_spent	
1	-16.878300	

Line Chart avg cpc over month of particular selected year



```
1  -- SP Average Cost Per Click Over Time
2  CREATE OR ALTER PROC spAvgCostPerClickByMonth
3  @year int = 2024,
4  @platforms nvarchar(max) = NULL,
5  @locations nvarchar(max) = NULL
6  AS
7  BEGIN
8
9      DECLARE @sql nvarchar(max);
10     DECLARE @platformsFilter nvarchar(max);
11     DECLARE @locationsFilter nvarchar(max);
12
13     -- handle platform is null or not provided
14     IF @platforms IS NULL OR @platforms = ''
15     BEGIN
16         -- get all platforms if @platforms is null or empty
17         SELECT @platformsFilter = STRING_AGG(QUOTENAME(platform, '''), ',')
18         FROM dim_product
19     END
20     ELSE
21     BEGIN
22         SET @platformsFilter = @platforms;
23     END
24
25     -- handle locations
26     IF @locations IS NULL OR @locations = ''
27     BEGIN
28         -- get all locations if @locations is null or empty
29         SELECT @locationsFilter = STRING_AGG(QUOTENAME(location, '''), ',')
30         FROM dim_location
31     END
32     ELSE
33     BEGIN
34         SET @locationsFilter = @locations;
35     END
36
37     -- base query
38     SET @sql = '
39
40     SELECT
41         MONTH(date) as month,
42         AVG(cost_per_click) as avg_cpc
43     FROM fact_camp f
44     INNER JOIN dim_product dp
45     ON f.platform_id = dp.platform_id
46     INNER JOIN dim_location dl
47     ON f.location_id = dl.location_id
48     WHERE YEAR(date) = @years
49         AND dp.platform IN ('' + @platformsFilter + '')
50         AND dl.location IN ('' + @locationsFilter + '')
51     GROUP BY MONTH(date)
52
53 '
54
55     EXEC sp_executesql @sql, N'@years INT', @years = @year;
56
57 END
58
59 exec spAvgCostPerClickByMonth @year = 2024,
60 @platforms = ''facebook'', ''instagram'',
61 @locations = ''bhaktapur'', ''kathmandu'';
```

	month	avg_cpc
1	1	2.220000
2	2	1.630000
3	4	6.885000
4	5	2.082000
5	6	2.530000