

PROMO EFFECTIVENESS REVIEW

(Jubelio x Rakamin Academy)

CASE :

Setiap 6 Bulan sekali (Akhir Semester) Omnichannel Jubelio mengadakan *Promo Effectiveness Review*, untuk melihat efektivitas dari PROMO CODE terhadap total penjualan dari salah satu marketplace.

Sebagai reporting engineer ada beberapa Task yang perlu anda lakukan.

1. Mengaplikasikan Promo Code dari Tim Budgeting.
2. Membuat Table Laporan Khusus Quartal ke 3 dan 4.
3. Melakukan Summary dan Visualisasi dari Hasil Data Table Quartal ke 3 dan 4.
4. Membuat Shipping Label Khusus Bulan December.

Untuk menjawab pertanyaan 1 sampai 3.

1. Buat database jubelio di RDBMS postgresSQL.
2. Import file Task5_DB yang berisi buyer_table, marketplace_table, sales_table, seller_table, shipping_table ke dalam database jubelio.
3. Import promo_code.csv ke dalam database yang sama.

Query:

```
CREATE TABLE promo_code(  
    promo_id INT PRIMARY KEY,  
    promo_name VARCHAR,  
    price_deduction INT,  
    Description VARCHAR,  
    Duration INT  
);  
COPY promo_code (promo_id, promo_name, price_deduction, Description, Duration)  
FROM 'C:\Data Skill 2023\Reporting Engineer Rakamin\final task\promo_code.csv'  
DELIMITER ',' CSV HEADER;
```

4. Buat table Q3_Q4_Review menggunakan 'CTE' dan 'INSERT with SELECT STATEMENT'. Table Q3_Q4_Review berisi kolom purchase_date, total_price, promo_code, discount, sales_after_promo.

Note:

total_price = quantity (sales_tabel) x price (marketplace_table)

sales_after_promo = (quantity (sales_tabel) x price (marketplace_table)) -(price_deduction as discount (promo_code))

kolom discount dan promo_code IS NOT NULL

Query:

```
CREATE TABLE Q3_Q4_Review(  
  purchase_date DATE,  
  total_price INT,  
  promo_code VARCHAR,  
  discount INT,  
  sales_after_promo INT)  
WITH A AS (SELECT sales_table.purchase_date,  
  sales_table.quantity,  
  marketplace_table.price,  
  CASE WHEN promo_code.promo_name IS NULL THEN 'TANPA_PROMO' ELSE promo_code.promo_name END AS promo_code,  
  CASE WHEN promo_code.price_deduction IS NULL THEN 0 ELSE promo_code.price_deduction END AS discount  
FROM sales_table  
LEFT JOIN marketplace_table ON sales_table.item_id = marketplace_table.item_id  
LEFT JOIN promo_code ON sales_table.promo_id = promo_code.promo_id  
WHERE sales_table.purchase_date >= '2022-07-01' AND sales_table.purchase_date <= '2022-12-31')  
INSERT INTO Q3_Q4_Review (purchase_date, total_price, promo_code, discount, sales_after_promo)  
SELECT purchase_date,  
  (quantity * price) AS total_price,  
  promo_code,  
  discount,  
  (quantity * price - discount) AS sales_after_promo  
FROM A
```

5. Export table Q3_Q4_Review dan buat visualisasi menggunakan Microsoft Excel.

Result table Q3_Q4_Review :

	purchase_date date	total_price integer	promo_code character varying	discount integer	sales_after_promo integer
1	2022-07-21	75000	GRATIS_ONGKIR	5000	70000
2	2022-07-15	117000	GRATIS_ONGKIR	5000	112000
3	2022-07-10	440000	GRATIS_ONGKIR	5000	435000
4	2022-07-09	229000	TANPA_PROMO	0	229000
5	2022-07-19	450000	TANPA_PROMO	0	450000
6	2022-07-28	299999	GRATIS_ONGKIR	5000	294999
7	2022-07-23	217500	TANPA_PROMO	0	217500
8	2022-07-07	260000	GRATIS_ONGKIR	5000	255000

Untuk melihat ratio pengguna yang memakai kode promo dan tidak memakai kode promo, Tambahkan kolom ratio pada tabel Q3_Q4_Review dengan mengubah kode promo yang bukan 'TANPA_PROMO' menjadi 'APPLY_PROMO'.

Query :

```
SELECT purchase_date,  
  promo_code,  
  CASE WHEN promo_code != 'TANPA_PROMO' THEN 'APPLY_PROMO' ELSE promo_code END as promo_ratio,  
  total_price,  
  sales_after_promo  
FROM Q3_Q4_Review  
ORDER BY purchase_date;
```

Result :

	purchase_date date	promo_code character varying	promo_ratio character varying	total_price integer	sales_after_promo integer
1	2022-07-01	GRATIS_ONGKIR	APPLY_PROMO	9998000	9993000
2	2022-07-01	GRATIS_ONGKIR	APPLY_PROMO	144000	139000
3	2022-07-01	AWAL_GAJIAN	APPLY_PROMO	315000	300000
4	2022-07-01	GRATIS_ONGKIR	APPLY_PROMO	154440	149440
5	2022-07-01	GRATIS_ONGKIR	APPLY_PROMO	1500000	1405000

Hasil visualisasi data:



promo_kode	Jul	Aug	Sep	Oct	Nov	Dec	Grand Total
10_10				7			7
AWAL_GAJIAN	8	9	14	10	7	15	63
CHRISTMAS						146	146
END_YEAR						399	399
GRATIS_ONGKIR	396	462	389	400	435	371	2453
HARBOLNAS						55	55
TANPA_PROMO	403	436	389	423	445	550	2646
TUJUHBELASAN		8					8
Grand Total	807	915	792	840	887	1536	5777

Interpretasi hasil visualisasi:

- Kode promo bulanan yang ditawarkan AWAL_GAJIAN dan GRATIS_ONGKIR.
- Penjualan di Q3 mengalami penurunan dan Penjualan di Q4 tidak menentu.
- Penjualan di akhir Q4 mengalami peningkatan sangat tinggi, hal tersebut mungkin disebabkan karena beragamnya kode promo yang ditawarkan.

Rekomendasi:

- Meningkatkan jumlah kode promo yang ditawarkan setiap bulannya.
- Memberikan pengingat notifikasi pada setiap pengguna untuk belanja menggunakan kode_promo.
- Membuat campaign atau bekerja sama dengan KOL specialist dan influencer untuk mengajak belanja dengan menggunakan kode promo.

Untuk membuat shipping label pada bulan desember:

1. Buat table shipping_summary yang terjadi di bulan desember didatabase yang sama menggunakan 'INSERT with SELECT STATEMENT'. Table berisi kolom shipping_date, seller_name, buyer_name, buyer_address, buyer_city, buyer_zipcode dan kode resi.
Noted :
Kode resi didapatkan dengan menggabungkan kolom shipping_id - purchase_date - shipping_date ("YYYYMMDD") - buyer_id - seller_id).
Menghubungkan shipping_table dengan seller_table dan buyer_table.

Query :

```
CREATE TABLE shipping_summary(shipping_date DATE,
    seller_name VARCHAR,
    buyer_name VARCHAR,
    buyer_address VARCHAR,
    buyer_city VARCHAR,
    buyer_zipcode VARCHAR,
    kode_resi VARCHAR
);
INSERT INTO shipping_summary (shipping_date, seller_name, buyer_name,
    buyer_address, buyer_city, buyer_zipcode, kode_resi)
SELECT sh.shipping_date,
    se.seller_name,
    bu.buyer_name,
    bu.address,
    bu.city,
    bu.zipcode,
    concat(sh.shipping_id, to_char(sh.purchase_date, 'yyyymmdd'), to_char(sh.shipping_date, 'yyyymmdd'),
        bu.buyer_id, se.seller_id) as kode_resi
FROM shipping_table sh
LEFT JOIN seller_table se ON sh.seller_id=se.seller_id
LEFT JOIN buyer_table bu ON sh.buyer_id=bu.buyer_id
WHERE sh.shipping_date BETWEEN '2022-12-01' AND '2022-12-31';
```

2. Export dan save hasil table.
Link table : https://bit.ly/Shipping_Summary
3. Selanjutnya buat label di Microsoft word dengan layout 2 Across dan 5 Down serta format buyer_name, buyer_address, buyer_city, buyer_zipcode, dan kode resi.

Result label_shipping:

AutoSave

label • Saved to this PC

Search

ideapad slim 3i

FileHomeInsertDrawDesignLayoutReferencesMailingsReviewViewHelpNitro ProTable DesignLayout

Comments

Editing

Share

Clipboard

Font

Paragraph

Styles

Normal

No Spacing

Heading 1

Heading 2

Christiana Halim	Bobby Andhika
Jl Mampang Prapatan Raya 18	Jl Panti Asuhan 2
Jakarta Pusat	Jakarta Pusat
17224	15502
8871202211292022120163151	8826202211302022120154302
Evi Wikarsa	Effendy Husin
Jl Dewi Sartika 357 C	Jl Ampera Raya Kav I/5 Ged Satmarindo Lt 2
Jakarta Pusat	Jakarta Pusat
15415	16277
92702022113020221201103125	9353202211292022120187305
Ben Morice	Elvira Pudjiwati
Jl Kembangan Indah Utama Bl J-2 A/16-17	Jl Raya Pasar Minggu Bl B/12

Page 1 of 14816645 words

English (Indonesia)

Accessibility: Investigate

Focus

100%