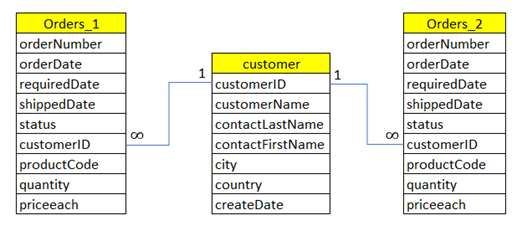
xyz.com adalah perusahan rintisan B2B yang menjual berbagai produk tidak langsung kepada end user tetapi ke bisnis/perusahaan lainnya. Sebagai data-driven company, maka setiap pengambilan keputusan di xyz.com selalu berdasarkan data. Setiap quarter xyz.com akan mengadakan townhall dimana seluruh atau perwakilan divisi akan berkumpul untuk me-review performance perusahaan selama quarter terakhir.

Sebagai seorang data analyst, kamu dimintai untuk menyediakan data dan analisa mengenai kondisi perusahaan bulan terakhir untuk dipresentasikan di townhall tersebut. (Asumsikan tahun yang sedang berjalan adalah tahun 2004).

Adapun hal yang akan direview adalah :

1. Bagaimana pertumbuhan penjualan saat ini?
2. Apakah jumlah customers xyz.com semakin bertambah ?
3. Dan seberapa banyak customers tersebut yang sudah melakukan transaksi?
4. Category produk apa saja yang paling banyak dibeli oleh customers?
5. Seberapa banyak customers yang tetap aktif bertransaksi?

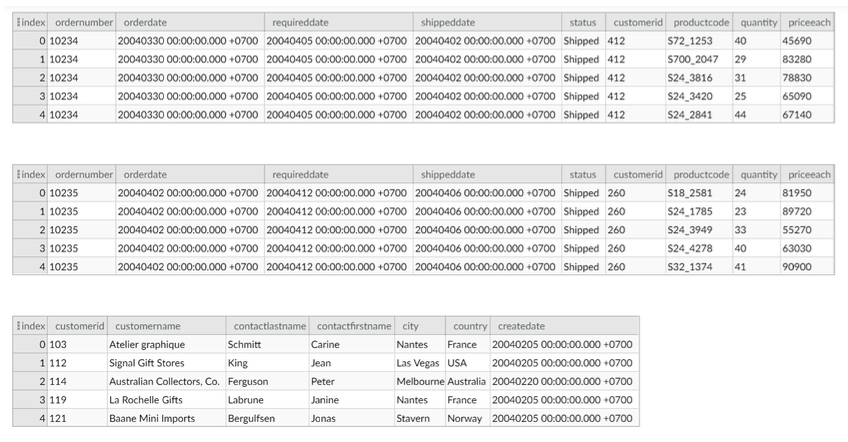
Tabel yang akan digunakan pada project kali ini adalah sebagai berikut.

1. Tabel orders\_1 : Berisi data terkait transaksi penjualan periode quarter 1 (Jan – Mar 2004)
2. Tabel Orders\_2 : Berisi data terkait transaksi penjualan periode quarter 2 (Apr – Jun 2004)
3. Tabel Customer : Berisi data profil customer yang mendaftar menjadi customer xyz.com

Sebelum memulai menyusun query SQL dan membuat Analisa dari hasil query, hal pertama yang perlu dilakukan adalah menjadi familiar dengan tabel yang akan digunakan. Hal ini akan sangat berguna dalam menentukan kolom mana sekiranya berkaitan dengan problem yang akan dianalisa, dan proses manipulasi data apa yang sekiranya perlu dilakukan untuk kolom – kolom tersebut, karena tidak semua kolom pada tabel perlu untuk digunakan.

* Mengecek tabel orders\_1 :SELECT \* FROM orders\_1 limit 5;
* Mengecek tabel orders\_2 :SELECT \* FROM orders\_2 limit 5;
* Mengecek tabel customer :SELECT \* FROM customer limit 5;

Lakukan pengecekan data di query editor dan jika berhasil hasil sebagai berikut



select \* from orders\_1 limit 5;

select \* from orders\_2 limit 5;

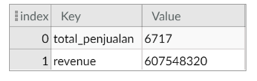
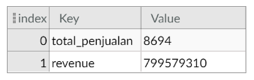
select \* from customer limit 5;

Total Penjualan dan Revenue pada Quarter-1 (Jan, Feb, Mar) dan Quarter-2 (Apr,Mei,Jun)

1. Dari tabel orders\_1 lakukan penjumlahan pada kolom quantity dengan fungsi aggregate sum() dan beri nama “total\_penjualan”, kalikan kolom quantity dengan kolom priceEach kemudian jumlahkan hasil perkalian kedua kolom tersebut dan beri nama “revenue”
2. Perusahaan hanya ingin menghitung penjualan dari produk yang terkirim saja, jadi kita perlu mem-filter kolom ‘status’ sehingga hanya menampilkan order dengan status “Shipped”.
3. Lakukan Langkah 1 & 2, untuk tabel orders\_2.

Notes: Jangan lupa untuk mengakhiri setiap statement dengan titik koma sehingga kedua statement dapat dijalankan bersamaan.(;)

Hasilnya sebagai berikut :



select sum(quantity) total\_penjualan, sum(quantity\*priceEach) revenue from orders\_1;

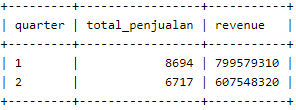
select sum(quantity) total\_penjualan, sum(quantity\*priceeach) revenue from orders\_2 where status = 'shipped';

## **Menghitung persentasi keseluruhan penjualan**

Kedua tabel orders\_1 dan orders\_2 masih terpisah, untuk menghitung persentasi keseluruhan penjualan dari kedua tabel tersebut perlu digabungkan :

1. Pilihlah kolom “orderNumber”, “status”, “quantity”, “priceEach” pada tabel orders\_1, dan tambahkan kolom baru dengan nama “quarter” dan isi dengan value “1”. Lakukan yang sama dengan tabel orders\_2, dan isi dengan value “2”, kemudian gabungkan kedua tabel tersebut.
2. Gunakan statement dari Langkah 1 sebagai subquery dan beri alias “tabel\_a”.
3. Dari “tabel\_a”, lakukan penjumlahan pada kolom “quantity” dengan fungsi aggregate sum() dan beri nama “total\_penjualan”, dan kalikan kolom quantity dengan kolom priceEach kemudian jumlahkan hasil perkalian kedua kolom tersebut dan beri nama “revenue”
4. Filter kolom ‘status’ sehingga hanya menampilkan order dengan status “Shipped”.
5. Kelompokkan total\_penjualan berdasarkan kolom “quarter”, dan jangan lupa menambahkan kolom ini pada bagian select.

Hasil kolom sebagai berikut :

****

select quarter, sum(quantity) total\_penjualan, sum(quantity\*priceeach) revenue from (

select orderNumber,status,quantity,priceeach, '1' quarter

from orders\_1

union

select orderNumber,status,quantity,priceeach, '2' quarter

from orders\_2

)as a

where status='shipped'

group by quarter;

## **Perhitungan Growth Penjualan dan Revenue**

Untuk project ini, perhitungan pertumbuhan penjualan akan dilakukan secara manual menggunakan formula yang disediakan di bawah. Adapun perhitungan pertumbuhan penjualan dengan SQL dapat dilakukan menggunakan “window function” yang akan dibahas di materi DQLab berikutnya.

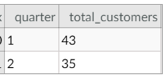
%Growth Penjualan = (6717 – 8694)/8694 = -22%

%Growth Revenue = (607548320 – 799579310)/ 799579310 = -24%

**jumlah customers xyz.com semakin bertambah?**

Penambahan jumlah customers dapat diukur dengan membandingkan total jumlah customers yang registrasi di periode saat ini dengan total jumlah customers yang registrasi diakhir periode sebelumnya.

1. Dari tabel customer, pilihlah kolom customerID, createDate dan tambahkan kolom baru dengan menggunakan fungsi QUARTER(…) untuk mengekstrak nilai quarter dari CreateDate dan beri nama “quarter”
2. Filter kolom “createDate” sehingga hanya menampilkan baris dengan createDate antara 1 Januari 2004 dan 30Juni 2004
3. Gunakan statement Langkah 1 & 2 sebagai subquery dengan alias tabel\_b
4. Hitunglah jumlah unik customers à tidak ada duplikasi customers dan beri nama “total\_customers”
5. Kelompokkan total\_customer berdasarkan kolom “quarter”, dan jangan lupa menambahkan kolom ini pada bagian select.



select quarter, count(distinct customerid) total\_customers from (select customerID, createDate, QUARTER(createDate) quarter

from customer

where createDate between '2004-01-01' and '2004-06-30'

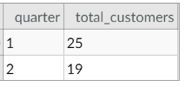
order by createDate) as b

group by 1

## **Seberapa banyak customers tersebut yang sudah melakukan transaksi?**

Problem ini merupakan kelanjutan dari problem sebelumnya yaitu dari sejumlah customer yang registrasi di periode quarter-1 dan quarter-2, berapa banyak yang sudah melakukan transaksi

1. Dari tabel customer, pilihlah kolom customerID, createDate dan tambahkan kolom baru dengan menggunakan fungsi QUARTER(…) untuk mengekstrak nilai quarter dari CreateDate dan beri nama “quarter”
2. Filter kolom “createDate” sehingga hanya menampilkan baris dengan createDate antara 1 Januari 2004 dan 30 Juni 2004
3. Gunakan statement Langkah A&B sebagai subquery dengan alias tabel\_b
4. Dari tabel orders\_1 dan orders\_2, pilihlah kolom customerID, gunakan DISTINCT untuk menghilangkan duplikasi, kemudian gabungkan dengan kedua tabel tersebut dengan UNION.
5. Filter tabel\_b dengan operator IN() menggunakan 'Select statement langkah 4' , sehingga hanya customerID yang pernah bertransaksi (customerID tercatat di tabel orders) yang diperhitungkan.
6. Hitunglah jumlah unik customers (tidak ada duplikasi customers) di statement SELECT dan beri nama “total\_customers”
7. Kelompokkan total\_customer berdasarkan kolom “quarter”, dan jangan lupa menambahkan kolom ini pada bagian select.



select quarter, count(distinct customerid) total\_customers from(

select customerID,createdate, quarter(createdate) quarter from customer

where createdate between '2004-01-01' and '2004-06-30') table\_b

where customerID IN (

select distinct customerID from orders\_1

union

select distinct customerid from orders\_2)

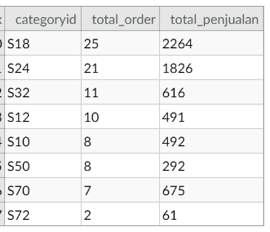
group by 1

## **Category produk apa saja yang paling banyak di-order oleh customers di Quarter-2?**

Untuk mengetahui kategori produk yang paling banyak dibeli, maka dapat dilakukan dengan menghitung total order dan jumlah penjualan dari setiap kategori produk.

1. Dari kolom orders\_2, pilih productCode, orderNumber, quantity, status
2. Tambahkan kolom baru dengan mengekstrak 3 karakter awal dari productCode yang merupakan ID untuk kategori produk; dan beri nama categoryID
3. Filter kolom “status” sehingga hanya produk dengan status “Shipped” yang diperhitungkan
4. Gunakan statement Langkah 1, 2, dan 3 sebagai subquery dengan alias tabel\_c
5. Hitunglah total order dari kolom “orderNumber” dan beri nama “total\_order”, dan jumlah penjualan dari kolom “quantity” dan beri nama “total\_penjualan”
6. Kelompokkan berdasarkan categoryID, dan jangan lupa menambahkan kolom ini pada bagian select.
7. Urutkan berdasarkan “total\_order” dari terbesar ke terkecil

Hasilnya sebagai berikut :



select \* from (select categoryID, count( distinct orderNumber) total\_order, sum(quantity) total\_penjualan from (select productCode, orderNumber, quantity, status, Left(productCode, 3) categoryID from orders\_2

where status = 'shipped') table\_c

group by categoryID) a

order by total\_order desc;

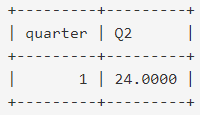
## **Seberapa banyak customers yang tetap aktif bertransaksi setelah transaksi pertamanya?**

Mengetahui seberapa banyak customers yang tetap aktif menunjukkan apakah xyz.com tetap digemari oleh customers untuk memesan kebutuhan bisnis mereka. Hal ini juga dapat menjadi dasar bagi tim product dan business untuk pengembangan product dan business kedepannya. Adapun metrik yang digunakan disebut retention cohort. Untuk project ini, kita akan menghitung retention dengan query SQL sederhana, sedangkan cara lain yaitu JOIN dan SELF JOIN akan dibahas dimateri selanjutnya :

Oleh karena baru terdapat 2 periode yang Quarter 1 dan Quarter 2, maka retention yang dapat dihitung adalah retention dari customers yang berbelanja di Quarter 1 dan kembali berbelanja di Quarter 2, sedangkan untuk customers yang berbelanja di Quarter 2 baru bisa dihitung retentionnya di Quarter 3.

1. Dari tabel orders\_1, tambahkan kolom baru dengan value “1” dan beri nama “quarter”
2. Dari tabel orders\_2, pilihlah kolom customerID, gunakan distinct untuk menghilangkan duplikasi
3. Filter tabel orders\_1 dengan operator IN() menggunakan 'Select statement langkah 2', sehingga hanya customerID yang pernah bertransaksi di quarter 2 (customerID tercatat di tabel orders\_2) yang diperhitungkan.
4. Hitunglah jumlah unik customers (tidak ada duplikasi customers) dibagi dengan total\_ customers dalam percentage, pada Select statement dan beri nama “Q2”

Hasilnya sebagai berikut :



#Menghitung total unik customers yang transaksi di quarter\_1

SELECT COUNT(DISTINCT customerID) as total\_customers FROM orders\_1;

#output = 25

select 1 quarter, (count(distinct customerid)\*100)/25 as q2

from orders\_1

where customerid in (

select distinct customerid

from orders\_2 );

Berdasarkan data yang telah kita peroleh melalui query SQL, Kita dapat menarik kesimpulan bahwa :

1. Performance xyz.com menurun signifikan di quarter ke-2, terlihat dari nilai penjualan dan revenue yang drop hingga 20% dan 24%,
2. perolehan customer baru juga tidak terlalu baik, dan sedikit menurun dibandingkan quarter sebelumnya.
3. Ketertarikan customer baru untuk berbelanja di xyz.com masih kurang, hanya sekitar 56% saja yang sudah bertransaksi. Disarankan tim Produk untuk perlu mempelajari behaviour customer dan melakukan product improvement, sehingga conversion rate (register to transaction) dapat meningkat.
4. Produk kategori S18 dan S24 berkontribusi sekitar 50% dari total order dan 60% dari total penjualan, sehingga xyz.com sebaiknya fokus untuk pengembangan category S18 dan S24.
5. Retention rate customer xyz.com juga sangat rendah yaitu hanya 24%, artinya banyak customer yang sudah bertransaksi di quarter-1 tidak kembali melakukan order di quarter ke-2 (no repeat order).
6. xyz.com mengalami pertumbuhan negatif di quarter ke-2 dan perlu melakukan banyak improvement baik itu di sisi produk dan bisnis marketing, jika ingin mencapai target dan positif growth di quarter ke-3. Rendahnya retention rate dan conversion rate bisa menjadi diagnosa awal bahwa customer tidak tertarik/kurang puas/kecewa berbelanja di xyz.com.