- 1. Update stg_order_details, add the following columns:
 - customer_phone
 - normalized_customer_phone (use macro to normalize the phone number)
 - country (based on normalized_customer_phone)
 - If the phone number is started with 62, the country should be Indonesia
 - If the phone number is started with 91, the country should be India

```
my_project > models > store > == stg_order_details.sql
        orders.order_date
        , details.quantity
        , details.price
        , {{normalize_phone_number('orders.customer_phone')}} as customer_phone
         , {{get_country('orders.customer_phone')}} as country
         , brands.name as brand_name
          , products.name as product_name
    from {{source('store','order_details')}} as details
      left join {{source('store', 'products')}} as products
      on details.product_id = products.product_id
 12 left join {{source('store', 'brands')}} as brands
    on brands.brand_id = products.brand_id
    left join {{source('store', 'orders')}} as orders
      on orders.order_id = details.order_id
 15
```

Kita mengupdate stg_order_details dengan menambahkan customer_phone dan country, cutomer_phone dan country ditambahkan dengan menggunakan macro, macro membantu kita untuk membuat logic yang dapat digunakan berulang-ulang

Query diatas adalah macro dari phone_number, macro tersebut berfungsi untuk menghapus tanda + dari phone_number

Query diatas adalah macro dari country, macro diatas berdasarkan dari kolom phone number, jika phone number yang diawali angka 62 maka country nya adalah indonesia dan jika phone number diawali dengan angka 91 maka country adalah india

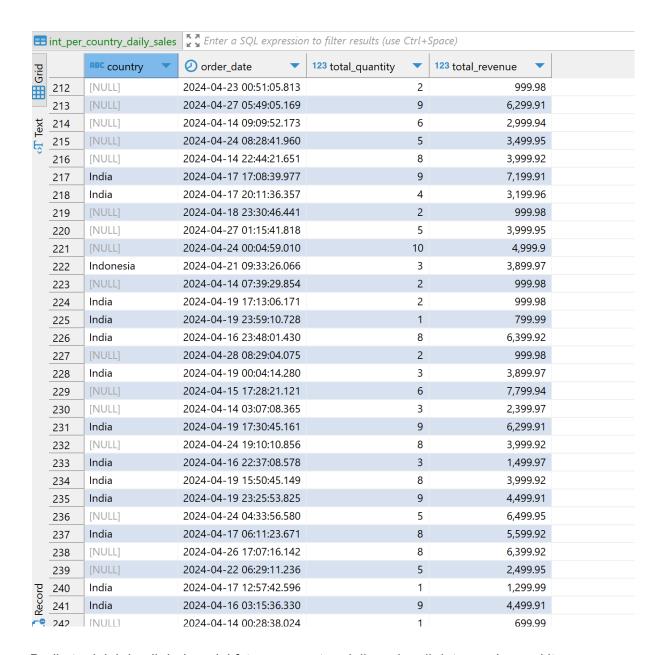
- 2. Base on stg_order_details, make another model named fct_per_country_daily_sales containing per country daily sales:
 - country
 - order date
 - total_quantity
 - total revenue

```
my_project > models > store_analytics >  int_per_country_daily_sales.sql

1    select
2    details.country
3    , details.order_date
4    , sum(details.quantity) as total_quantity
5    , sum(details.price) as total_revenue
6    from {{ref('stg_order_details')}} as details
7    group by
8    details.country
9    , details.order_date
```

Sesuai dengan instruksi mentor pada task sebelumnya, saya terlebih dahulu membuat model int_per_country_daily_sales

Kemudian saya baru membuat model fct_per_country_daily_sales



Berikut adalah hasil dari model fct_per_country_daily_sales di datawarehouse kita