1. **Understand Requirements**
    - **List the total revenue of each store everyday.**
      sum(amount) [on store + day level]
    - **List the total revenue of totally everyday.**
      sum(amount) [on day level]
    - **List the top 5 stores according to their weekly revenue every week.**
      sum(amount) [on week level].
    - **List top 5 sales clerks who have the most sales each day/week/month.**
      sum(amount) [on staff_id + day (week/month) level]
    - **Which film is the most popular each week/month in each store/totally?**
      count(fjlm_id) [on store + day (week/month) level]
    - **Who are our top 10 customers each month/year? How much do they spend accordingly each month/year?**
      sum(amount) [on customer_id + month/year level]
    - **Is there any store the sales is in a decline trend (within the recent 4 weeks the avg sales of each week is declining)**
        - *sum(amount) [on store + week level]*
        - ***Is_decline*** *Flag that indicates if the store is in the decline trend. The flag needs to be calculated in further steps based on the sum(amount) per store per week. You will find the details from the ETL script. The reason why we put this flag in the fact table instead of calculating it on the fly is because we need to make user queries as easy as possible.*

2. **Identify Grain and dimensions**

From the above requirements, we used customer_id, film_id, store_id, staff_id therefore our dimensions will be **customer, film, store, staff**. Also, we used the day/week/month for time, therefore we will include the **calendar** dimension.

The gain of dimensions are: customer on customer_id level, store on store_id level, staff on staff_id level, calendar on day level.

- Customer dimension
  The new customer dimension needs to be joined by the original customer, address, city, country tables.

- Film dimension
  The new film dimension needs to be joined by the original film, film_actor, actor, category, film_category , language tables. In this project, we joined the language table although it was not used in the current requirements. The reason for this is because we need to prepare for the future possible requirements.

- Store dimension
  The new store dimension needs to be joined by the original store, staff, address, city, country tables.

- Staff dimension
  The new staff dimension needs to be joined by the staff address, city, country tables.

- Calendar dimension
  Used for the time dimension include day/week/month/year. It will be populated only once and used all the time.

3. Fact table
   The fact table are build on the follow atomic row:
   date + customer_id + store_id + film_id + staff_id

   The fact table is created by joining the original tables **payment** + **rental** + **inventory**.
   The measures are:
   amount (from the payment table)
   is_decline flag (from ETL calculation)