Define Problem Statement – Daily Product Revenue

Let us try to get daily product revenue using retail tables.

- daily is derived from orders.order_date.
- product has to be derived from products.product_name.
- revenue has to be derived from order_items.order_item_subtotal.
- We need to join all the 3 tables, then group by order_date, product_id as well as product_name to get revenue using order_item_subtotal.
- Get Daily Product Revenue using products, orders and order_items data set.
- We have following fields in **orders**.
  - order_id
  - order_date
  - order_customer_id
  - order_status
- We have following fields in **order_items**.
  - order_item_id
  - order_item_order_id
  - order_item_product_id
  - order_item_quantity
  - order_item_subtotal
  - order_item_product_price
- We have following fields in **products**
  - product_id
  - product_category_id
  - product_name
  - product_description
  - product_price
  - product_image
- We have one to many relationship between orders and order_items.
- **orders.order_id** is **primary key** and **order_items.order_item_order_id** is foreign key to **orders.order_id**.
- We have one to many relationship between products and order_items.
- **products.product_id** is **primary key** and **order_items.order_item_product_id** is foreign key to **products.product_id**
- By the end of this module we will explore all standard transformations and get daily product revenue using following fields.
  - **orders.order_date**
  - **order_items.order_item_product_id**
  - **products.product_name**
  - **order_items.order_item_subtotal** (aggregated using date and product_id).
- We will consider only **COMPLETE** or **CLOSED** orders.
- As there can be more than one product names with different ids, we have to include product_id as part of the key using which we will group the data.

By Durga Gadiraju