We have the most fundamental basic table that is the online market table. Like all tables, the online market table has its own attributes such as shop id, shop name, shop phone. Perhaps you will say why we need a shop phone number, not all business processes of the transaction can be successfully completed and at this moment you need to call the technical part, at such moments you need a shop phone number. But such as shop name, shop id are all individual. As you can see, there are three important tables directly connected to our root online market table. These are customer table , staff table , supplier table. The online market table ratio for them is one to many. There is a reason for this, because each one online market can have many customers and many partners and sellers. The same goes for the stuff table, because one person cannot build a big business. And for this we need a team, or rather a stuff table. Let's take a closer look at the attributes of these tables. As you can see, the supplier table has its own attributes such as supplier email, supplier id, supplier name, supplier surname, supplier address. And to be sure, we have complete information about our partner sellers. For us, this is a huge advantage because we will know with what person we will cooperate. Now let's look at the attributes of the customer table. The customer table has such attributes as first name, last name, country, customer id, customer address, customer email. Perhaps you will say why we need the country attribute. We can have clients not only from another city, but also from another country, at such a moment we must know all the information about our client. Now let's move on to the staff table. This table contains such attributes as staff id, staff name, staff phone , staff email. Our staff may include marketers, programmers, targetologists, and in order to track them, we need this table. Moreover, detailed information about our staff can be found in the staff detail table. The staff details table has attributes such as staff salary, staff id, staff specialty. I wanted to mention that there is an equally important table that is closely connected with the customer table. This is the login table. This table has attributes such as login username, user id, user password. With this table, we can keep track of the number of customers. If there are more successful transactions, then there are more users. And with that, we can track customers. There is another table directly connected to the customer table. This is a table payment when a transaction with a client ends successfully, he pays for the goods. Then the table payment will be needed. The attributes of the payment table are payment method , payment amount, payment amount , payment no. payment know it goes like a check of the goods and payment date confirms the payment for the goods. If a customer has made a purchase and wants to cancel then at that point these attributes will come into play. There is one more additional table which is the website table. After all, any successful online business application has a website. The site will help you find out more information about us. 80% of the app has a website. The table site is closely related to the staff table. After all, they work with the site to optimize and track the staff of the team. This table also has its own website link and shop ID attributes. Now let's look at our main table - this is the product. The table is our product. and it has its own important attributes like product id, product name, product price, category id. With this, we can distinguish between the product itself or the product itself. After all, we may have toys for sale or jewelry or household appliances. and At this point, to parse everything, we need a category table. There are attributes like category name, category id. This is where the goods are separated. With the table, the category product is closely related to the sales table. With the help of the sales table. We will be able to track the bestseller of the season or hot items. This gives a great advantage when choosing a product for sellers. There is one more that I have to consider is the table that is related to the product table which is the cart table. The cart table has its own customer id, product id, product name attributes. In this table, when a customer selects an item, it automatically goes to the cart table. There, he can remove this selected product or proceed with the payment transaction. Now let's move on to the order table, this is also the main important part of our diagramm. The order table will be responsible for the beginning transaction of the goods. After payment, the goods are prepared for shipment and go to the order table. The order table has its own attributes like customer id, order amount, order date , order no. After that, the goods go to the second part of the transaction. This is a delivery table where the customer can cancel the delivery of the goods if he wants to return the goods. In this table, we again call the client to clarify the delivery of the goods. When the product passes the third part of the table this is the tracking detail. In this part, the goods go to the courier and the goods are delivered directly. As you can see the tracking detail attributes there we can find out completely about our courier. After a successful transaction, as you can see, the payment transaction passes through the online market, because there the platform itself takes a commission of 10%, the rest of the amount goes to the partner, or rather the seller. This is how our logic is built in ER diagramm.

There are codes :

1) Add user-defined exception which disallows to enter title of item (e.g. book) to be less than 5 characters

DECLARE

product\_name VARCHAR2(225) := 'ball';

min\_length CONSTANT INTEGER := 5;

length\_error EXCEPTION;

BEGIN

IF LENGTH(product\_name) < min\_length THEN

RAISE length\_error;

END IF;

EXCEPTION

WHEN length\_error THEN

DBMS\_OUTPUT.PUT\_LINE('Error: product name length must be at least ' min\_length ' characters.');

END;

2)Procedure which does group by information

CREATE OR REPLACE PROCEDURE group\_by\_tako1

IS

c SYS\_REFCURSOR;

BEGIN

OPEN c FOR

SELECT staff.staff\_id, staff.staff\_name, staff.staff\_phone

FROM staff

GROUP BY staff.staff\_id, staff.staff\_name, staff.staff\_phone;

DBMS\_SQL.RETURN\_RESULT(c);

END;

3) Function which counts the number of records

CREATE OR REPLACE FUNCTION totalStaff

RETURN number IS

total number(3):=0;

BEGIN

SELECT count(\*) into total

FROM STAFF;

return total;

END;

DECLARE

c number(3);

BEGIN

c:=totalStaff();

dbms\_output.put\_line('Total number of staffs: '|| c);

END;

4)Create a trigger before insert on any entity which will show the current number of rows in the table

CREATE OR REPLACE TRIGGER tako\_count

BEFORE INSERT ON product

FOR EACH ROW

DECLARE

product\_id NUMBER;

BEGIN

SELECT COUNT(\*) INTO product\_id FROM product;

DBMS\_OUTPUT.PUT\_LINE('Number of rows before insert: ' || product\_id);

END;

5)Procedure which uses SQL%ROWCOUNT to determine the number of rows affected

CREATE OR REPLACE PROCEDURE affect AS

total\_rows number(2);

BEGIN

UPDATE staff\_datail2

SET salary = salary + 1000

WHERE staff\_speciality='marketer';

IF SQL%ROWCOUNT = 0 THEN

dbms\_output.put\_line('no rows selected');

ELSIF SQL%ROWCOUNT > 0 THEN

total\_rows := SQL%ROWCOUNT;

dbms\_output.put\_line(total\_rows || ' rows updated');

END IF;

END;

This codes are described in the video(You can find an youtube link in the google form). Also, the fifteen tables in the video(you can find the tables in the google drives).