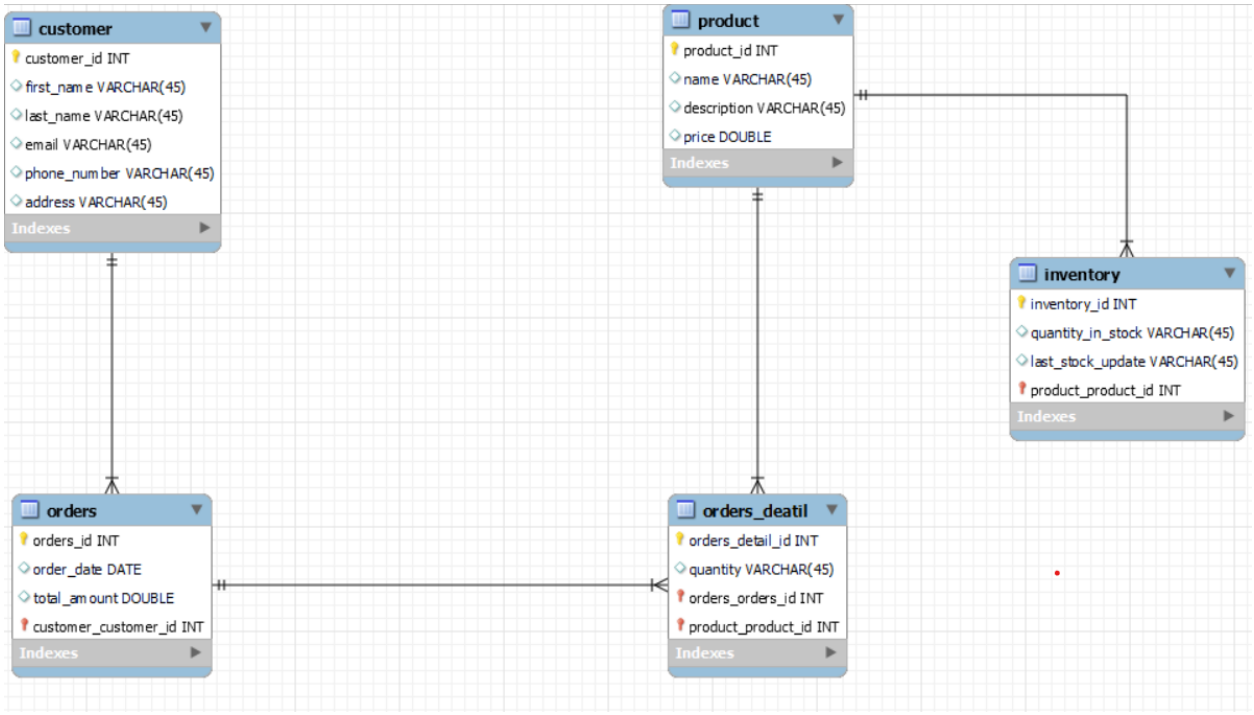
ASSIGNMENT 5
ELECTRONISC GADGET

ER DIAGRAM:



QUERIES:

```
use hexfeb2024;
show tables;

insert into customer(first_name,last_name,email,phone_number,address)
values('aadhya','parvin','123@gmail.com','123456','coimbatore'),
('goyal',null,'123@gmail.com','123456','pune'),
('aakriti','pankaj','123@gmail.com','123456','chennai'),
('zoya','pulkit','123@gmail.com','123456','mumbai'),
('rachel','sebastin','123@gmail.com','123456','tuticod');
insert into product(name,description,price)
values('panasonic tv','full screen high quality',978000),
('blender','3 inone jar',25000),
('vacuum cleaner','full speed',20790),
('mac book','high quality',850890),
('apple iphone15','flexing',990989),
('dell laptop','hd picture quality',760000),
('ipad','portable',50480);
insert into inventory(product_product_id,quantity_in_stock,last_stock_update)
values(4,'59','59'),
(6,'59','59'),
```

```
(3,'39','100'),
(2,'150','90'),
(1,'200','58'),
(7,'250','59'),
(5,'70','59');
insert into orders(customer_customer_id,order_date,total_amount)
values(2,'2024-02-28',54090),
(1,'2024-01-17',60730),
(3,'2024-01-02',98000),
(4,'2024-03-01',78450),
(5,'2024-03-05',30500);
insert into orders_deatil(orders_orders_id,product_product_id,quantity)
values(1,2,'1'),
(2,1,'2'),
(3,4,'2'),
(4,5,'1'),
(5,7,'2');

-- Tasks 2: Select, Where, Between, AND, LIKE:
/*1. Write an SQL query to retrieve the names and emails of all customers. */
select first_name,email
from customer;
/*2. Write an SQL query to list all orders with their order dates and corresponding customer
names.*/
select c.first_name,o.order_date
from customer c,orders o
where c.customer_id=o.customer_customer_id;
/*3. Write an SQL query to insert a new customer record into the "Customers" table. Include
customer information such as name, email, and address.*/
insert into customer(first_name,last_name,email,phone_number,address)
values('rayan',null,'123@gmail.com','123456','coimbatore');
/*4. Write an SQL query to update the prices of all electronic gadgets in the "Products" table by
increasing them by 10%*/
update product set price=price+(10/100)
where product_id in(1,2,3,4,5,6,7);
/*5. Write an SQL query to delete a specific order and its associated order details from the
"Orders" and "OrderDetails" tables. Allow users to input the order ID as a parameter.*/
delete from orders
where order_id=5;
delete from orders_deatil
```

```
where order_order_id=5;
/*6. Write an SQL query to insert a new order into the "Orders" table. Include the customer ID,
order date, and any other necessary information.*/
insert into orders(customer_customer_id,order_date,total_amount)
values(2,'2024-01-28',74090);
/*7. Write an SQL query to update the contact information (e.g., email and address) of a specific
customer in the "Customers" table. Allow users to input the customer ID and new contact
information.*/
update customer set email='abc@gmail.com',address='kulu'
where customer_id=2;
/*8. Write an SQL query to recalculate and update the total cost of each order in the "Orders"
table based on the prices and quantities in the "OrderDetails" table.*/
/*9. Write an SQL query to delete all orders and their associated order details for a
specific customer from the "Orders" and "OrderDetails" tables. Allow users to input
the customer ID as a parameter.*/
delete from orders
where customer_customer_id=5;
/*10. Write an SQL query to insert a new electronic gadget product into the "Products" table,
including product name, category, price, and any other relevant details.*/
insert into product(name,description,price)
values('bluetooth spaeker','high frequency',15000);
/*11. Write an SQL query to update the status of a specific order in the "Orders" table (e.g., from
"Pending" to "Shipped"). Allow users to input the order ID and the new status.*/
/*12. Write an SQL query to calculate and update the number of orders placed by
each customer in the "Customers" table based on the data in the "Orders" table.*/
select c.first_name,count(o.orders_id)
from customer c join orders o
on c.customer_id=o.customer_customer_id
group by c.customer_id;

-- Task 3. Aggregate functions, Having, Order By, GroupBy and Joins:
/*1. Write an SQL query to retrieve a list of all orders along with customer information (e.g.,
customer name) for each order.*/
select c.first_name,c.email,o.orders_id,o.order_date,o.total_amount
from customer c join orders o
on c.customer_id=o.customer_customer_id;
/*2. Write an SQL query to find the total revenue generated by each electronic gadget product.
Include the product name and the total revenue.*/
select product_id,name,description,sum(price) as revenue
from product
```

```
group by product_id;
/*3. Write an SQL query to list all customers who have made at least one purchase. Include their
names and contact information.*/
select c.first_name,c.phone_number,count(c.customer_id)
from customer c join orders o
on c.customer_id=o.customer_customer_id
group by c.customer_id
having count(c.customer_id)>=1;
/*4. Write an SQL query to find the most popular electronic gadget, which is the one with the
highest
total quantity ordered. Include the product name and the total quantity ordered.*/
select p.name,p.description,sum(o.quantity)
from product p,orders_deatil o
where p.product_id=o.product_product_id
group by p.product_id
order by sum(o.quantity) desc
limit 1;
/*5. Write an SQL query to retrieve a list of electronic gadgets along with their corresponding
categories.*/
select name,description,price
from product;
/*6. Write an SQL query to calculate the average order value for each customer. Include the
customer's name and their average order value.*/
/*7. Write an SQL query to find the order with the highest total revenue. Include the order ID,
customer information, and the total revenue.*/
select sum(od.quantity*p.price) as revenue
from orders_deatil od,product p
where p.product_id=od.product_product_id
group by orders_detail_id
order by revenue desc
limit 1;
/*8. Write an SQL query to list electronic gadgets and the number of times each product has been
ordered.*/
select p.name,p.description,o.quantity
from product p join orders_deatil o
on p.product_id=o.product_product_id;
/*9. Write an SQL query to find customers who have purchased a specific electronic gadget
product.
Allow users to input the product name as a parameter.*/
select c.first_name,p.name
```

```
from customer c,product p,orders o,orders_deatil od
where c.customer_id=o.customer_customer_id and
o.orders_id=od.orders_orders_id and
p.product_id=od.product_product_id;
/*10. Write an SQL query to calculate the total revenue generated by all orders placed within a
specific time period. Allow users to input the start and end dates as parameters.*/
select sum(od.quantity*p.price) as revenue
from orders_deatil od,product p,orders o
where p.product_id=od.product_product_id and o.orders_id=od.orders_orders_id
and order_date between '2024-02-02' and '2024-03-01';

-- Task 4. Subquery and its type:
/*1. Write an SQL query to find out which customers have not placed any orders.*/
select first_name,email
from customer
where customer_id not in(select customer_customer_id from orders);
/*2. Write an SQL query to find the total number of products available for sale. */
select count(*) as total
from product;

/*3. Write an SQL query to calculate the total revenue generated by TechShop.*/
select sum(od.quantity*p.price) as revenue
from orders_deatil od,product p
where p.product_id=od.product_product_id
group by orders_detail_id;
/*4. Write an SQL query to calculate the average quantity ordered for products in a specific
category.
Allow users to input the category name as a parameter.*/
select avg(quantity) as avg
from orders_deatil od join product p
on od.product_product_id=p.product_id
where p.product_id=2;
/*5. Write an SQL query to calculate the total revenue generated by a specific customer. Allow
users
to input the customer ID as a parameter.*/
select sum(od.quantity*p.price) as revenue
from orders_deatil od,product p
where p.product_id=od.product_product_id
group by orders_detail_id;
/*6. Write an SQL query to find the customers who have placed the most orders. List their names
```

```
and the number of orders they've placed.*/
select count(o.orders_id) as total,c.first_name
from orders o join customer c
on c.customer_id=o.customer_customer_id
group by(c.customer_id)
order by total desc
limit 0,1;

/*7. Write an SQL query to find the most popular product category, which is the one with the
highest
total quantity ordered across all orders.*/
select p.product_id,sum(od.quantity) as total
from product p join orders_deatil od
on p.product_id=od.product_product_id
group by p.product_id
order by total desc
limit 1;
/*8. Write an SQL query to find the customer who has spent the most money (highest total
revenue)
on electronic gadgets. List their name and total spending.*/

/*9. Write an SQL query to calculate the average order value (total revenue divided by the
number of
orders) for all customers.*/
/*10. Write an SQL query to find the total number of orders placed by each customer and list
their
names along with the order count*/
select c.customer_id,sum(o.orders_id) as total
from orders o join customer c
on c.customer_id=o.customer_customer_id
group by (c.customer_id);
```