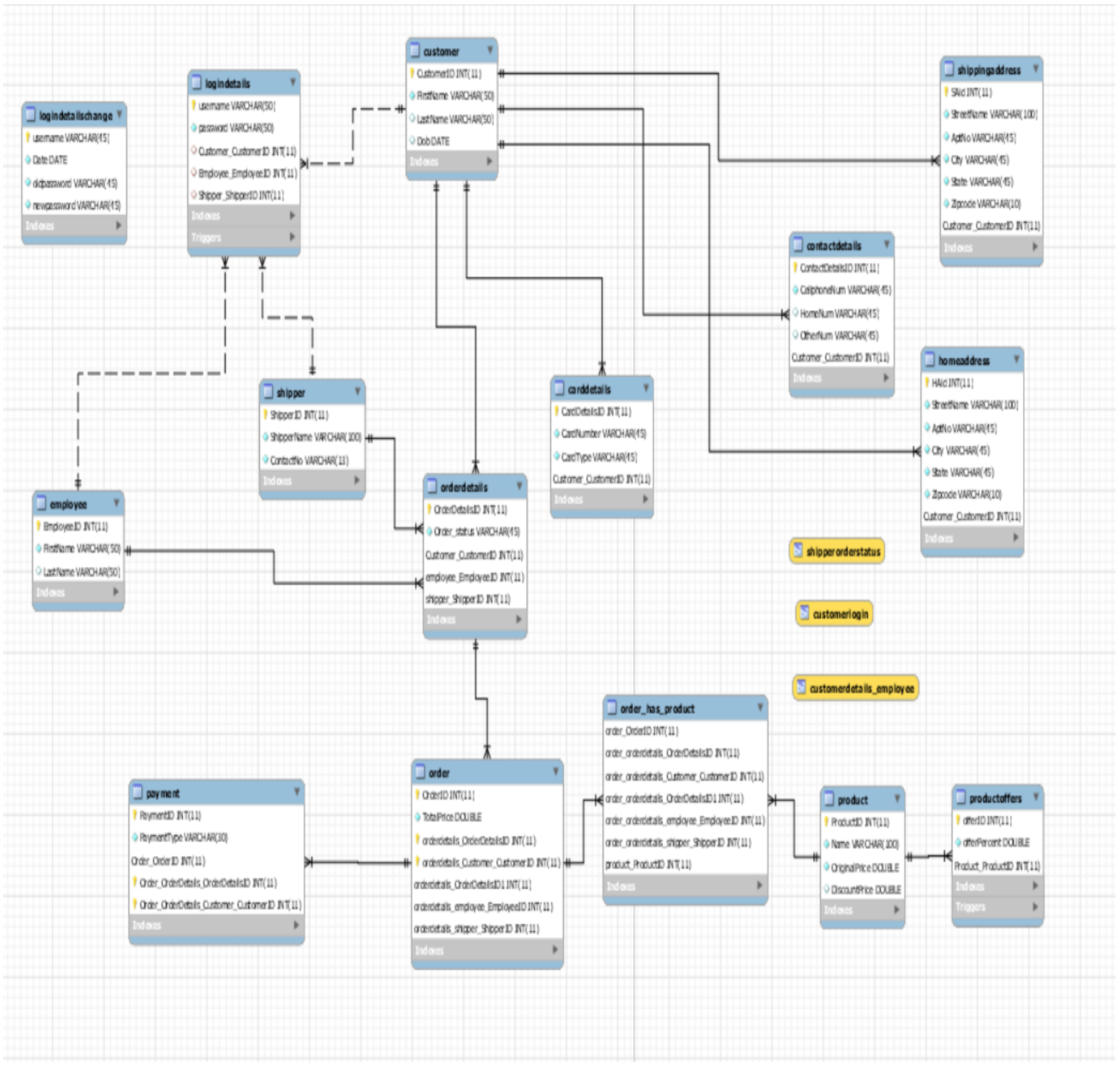
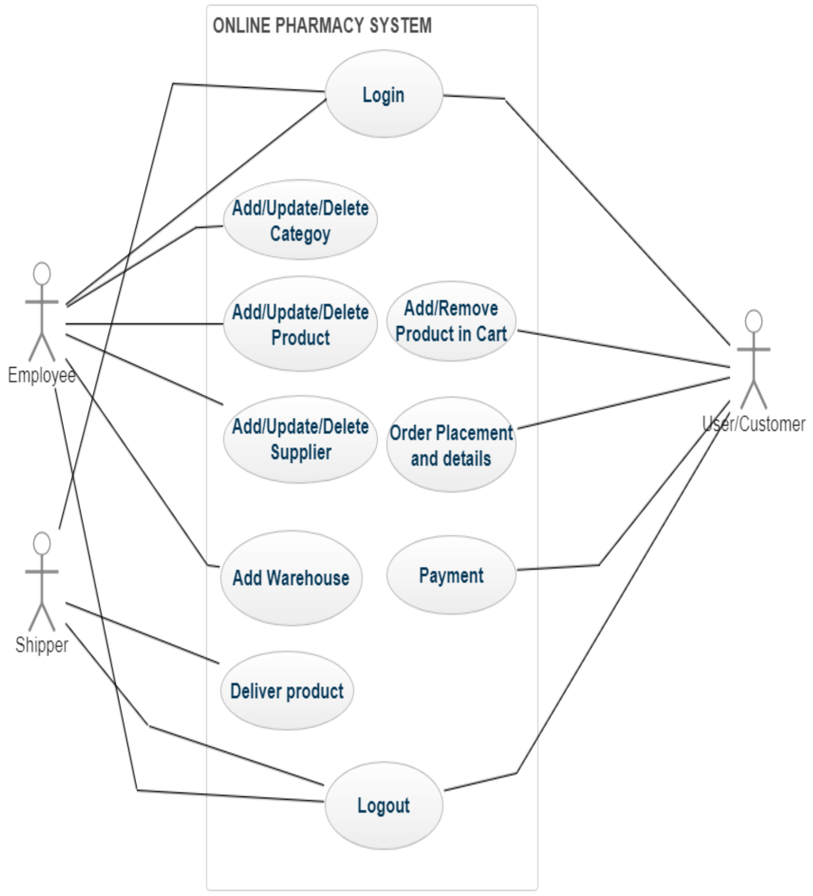
TOPIC: ONLINE MEDICAL STORE DHAVAL SUTHAR

- **PROBLEM DEFINITION:**
  - Pharmacists are integral entity in the healthcare system as they provide the drugs and medicines to cure the disease. With increase in demands of drugs it is important to provide the required items in the proper timeframe to accommodate the cure. As technology is evolving at rapid pace, we can use the Online Medical store system to abridge the chain between healthcare and pharmacist.
- **APPROACH TO PROBLEM:**
  - The online medical store aims to provide patients a convenient way to order medicines at home in a shorter time and with optimal price. With an efficient communication between the Customer, Organization and Shipper, the drugs can be delivered to customers in the shortest time frame and making it hassle-free for customer to buy it physically.
- **ER DIAGRAM:**



- **Users of the System:**
  - *Customer*: Who can login into the website and buy drugs.
  - *Employees of the organization*: Who will manage the orders from the customer and assign the orders to shipper.
  - *Shipper:* Who will ship the orders to customer.
- **Use Case Diagram:**


ONLINE PHARMACY SYSTEM
Login
Add/Update/Delete Categoy
Add/Update/Delete Product
Add/Remove Product in Cart
Add/Update/Delete Supplier
Order Placement and details
Add Warehouse
Payment
Deliver product
Logout
Employee
Shipper
User/Customer

- **Normalization:**
  - *Customer TABLE*:
    1NF: it has repeated group as address and contact details. so, I have split it into contact details and home address and shipping address.
    2NF: customer has customerid and username as candidate key but as password depends username it violates the 2nf properties. so, I have split username and password in another table called LoginDetails. (Similar with Shipper and Employee table)

  - *Order Details TABLE*:

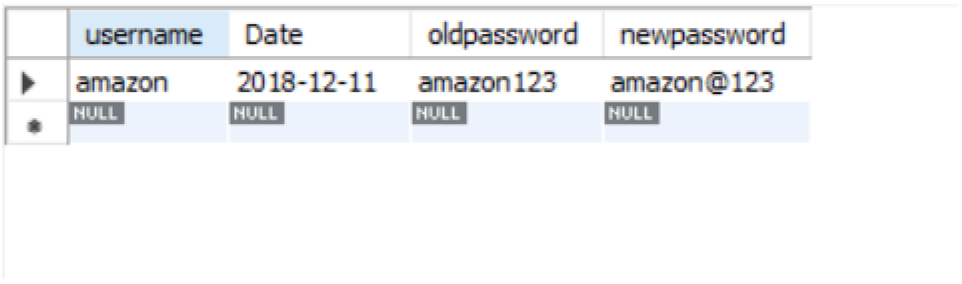
    3NF: Order Details has list of orders belongs to particular customer, employee and shipper. Although it is in 2NF, Product depends on order where order depends on the OrderDetails ID so there is a transitivity dependency.

I have split it into Order Table and there also product related information depends on ProductName that is why I have split it into Another table called Product.

- Triggers:
  - ***Trigger which maintains record for the username and its updated password***

```
DELIMITER //
Create trigger LoginDetailsRecord
AFTER UPDATE ON LoginDetails
FOR EACH ROW
BEGIN
        INSERT INTO LoginDetailsChange VALUES
        (NEW.username,now(),OLD.password,NEW.password);
END;//
```

OUTPUT:

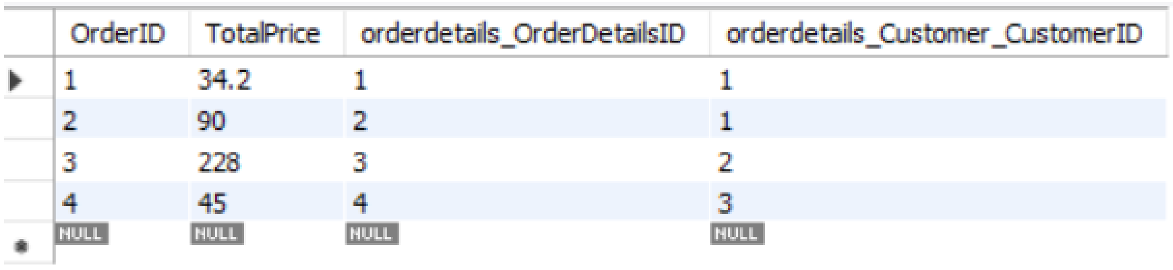


	username	Date	oldpassword	newpassword
▶	amazon	2018-12-11	amazon123	amazon@123
*	NULL	NULL	NULL	NULL

  - ***Trigger which update the order total price***

```
DELIMITER //
Create trigger UpdateOrderPrice
AFTER INSERT on order_has_product
For each row
BEGIN
        SET @quantity = new.quantity;
        SET @productid = new.Product_ProductID;
        SET @price = (select discountprice from product where P
        roductID=@productid);
        UPDATE onlinemedicalstore.order
        SET Totalprice= @quantity*@price
        WHERE onlinemedicalstore.order.OrderID =
        new.Order_OrderDetails_OrderDetailsID;
END;//
```

OUTPUT:

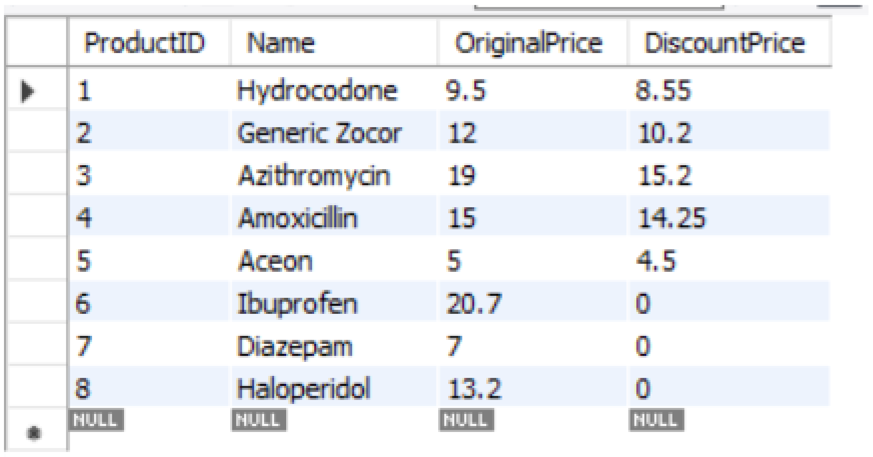


OrderID	TotalPrice	orderdetails_OrderDetailsID	orderdetails_Customer_CustomerID
1	34.2	1	1
2	90	2	1
3	228	3	2
4	45	4	3
NULL	NULL	NULL	NULL

- ***Trigger which updates the product discount price***

```
DELIMITER //
Create trigger Productpriceupdate
AFTER INSERT ON productoffers
FOR EACH ROW
BEGIN
        SET @offer = new.offerPercent;
        UPDATE product
        SET DiscountPrice = round(originalprice*((100-@offer)/100),2)
        WHERE product.ProductID = new.Product_ProductID;
END;//
```

OUTPUT:

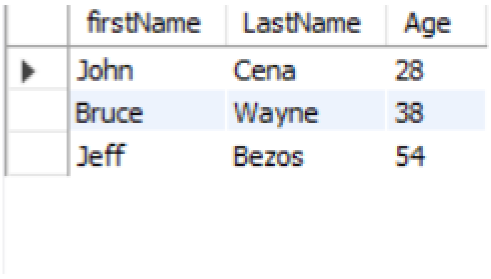


ProductID	Name	OriginalPrice	DiscountPrice
1	Hydrocodone	9.5	8.55
2	Generic Zocor	12	10.2
3	Azithromycin	19	15.2
4	Amoxicillin	15	14.25
5	Aceon	5	4.5
6	Ibuprofen	20.7	0
7	Diazepam	7	0
8	Haloperidol	13.2	0
NULL	NULL	NULL	NULL

- **USER DEFINED FUNCTION**:
  - ***To calculate age of the customer***

```
create function findage(birthdate DATE)
returns int DETERMINISTIC
return year(curdate())-year(birthdate);
```

OUTPUT:

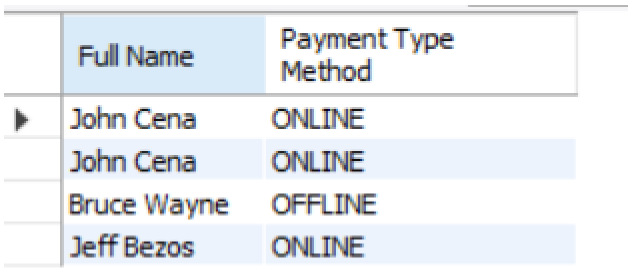


firstName	LastName	Age
John	Cena	28
Bruce	Wayne	38
Jeff	Bezos	54

  - ***To find of payment type of the customer***
    online: credit/debit card
    offline: cash on delivery

```
create function findpaymenttype(paymentType varchar(50))
returns varchar (50) DETERMINISTIC
return IF (paymentType in ("Credit card","Debit card"),"ONLINE","OFFLINE");
```

OUTPUT:



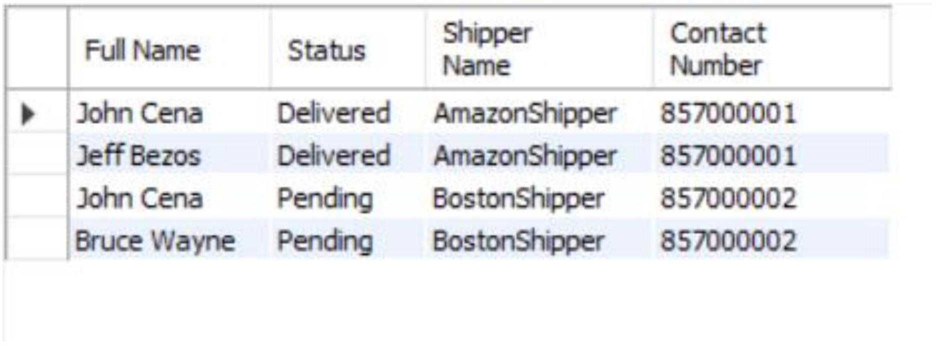
Full Name	Payment Type Method
John Cena	ONLINE
John Cena	ONLINE
Bruce Wayne	OFFLINE
Jeff Bezos	ONLINE

- **Views**:
  - ***View for Employee to find out the status of customer order***

```
create view ShipperOrderStatus as
select concat_ws(" ",c.Firstname,c.Lastname) as "Full Name",o.Order_status as
"Status" ,s.shippername as "Shipper Name",s.ContactNo as "Contact Number"
from customer c inner join orderDetails o
on c.CustomerID =o.Customer_CustomerID
```

inner join shipper s
on s.ShipperID = o.shipper_ShipperID;

OUTPUT:

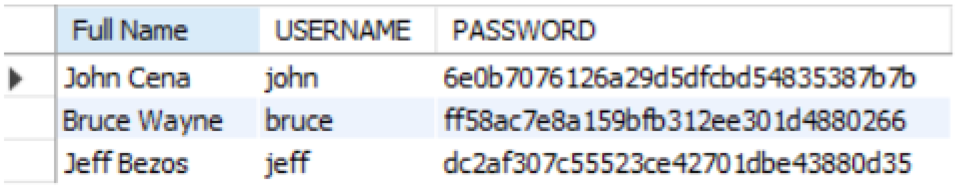


Full Name	Status	Shipper Name	Contact Number
John Cena	Delivered	AmazonShipper	857000001
Jeff Bezos	Delivered	AmazonShipper	857000001
John Cena	Pending	BostonShipper	857000002
Bruce Wayne	Pending	BostonShipper	857000002

- ***View for Employee to find out the customer username and encrypted password***

  create view CustomerLogin as
  select concat_ws(" ",c.Firstname,c.Lastname) as "Full Name", l.username as "USERNAME",MD5(l.password) as "PASSWORD"
  from customer c
  inner join logindetails l
  on c.CustomerID=l.Customer_CustomerID;

  OUTPUT:

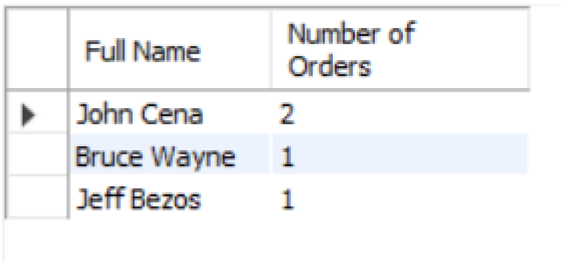


Full Name	USERNAME	PASSWORD
John Cena	john	6e0b7076126a29d5dfcbd54835387b7b
Bruce Wayne	bruce	ff58ac7e8a159bfb312ee301d4880266
Jeff Bezos	jeff	dc2af307c55523ce42701dbe43880d35

- ***View for Employee to find out the how many orders customer has placed***

  create view CustomerDetails_Employee as
  select concat_ws(" ",c.Firstname,c.Lastname) as "Full Name",count(o.Customer_CustomerID) as "Number of Orders"
  from customer c inner join orderDetails o
  on c.CustomerID=o.Customer_CustomerID
  group by c.Firstname,c.Lastname

OUTPUT:



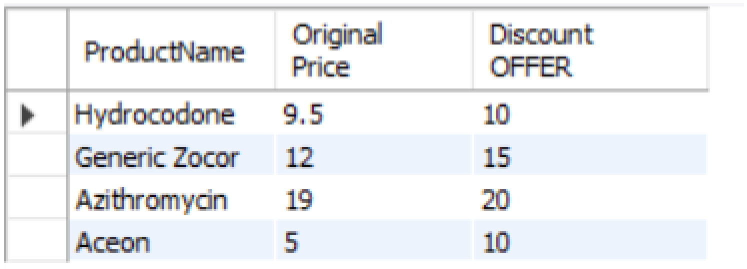
Full Name	Number of Orders
John Cena	2
Bruce Wayne	1
Jeff Bezos	1

- **STORED PROCEDURE:**

  - ***To find out all the product which has discount >5***

```
Delimiter //
create procedure productofferdiscount()
BEGIN
        Select p.Name as "ProductName" ,p.OriginalPrice as "Original
        Price",f.offerPercent as "Discount OFFER"
        from product p
        inner join productoffers f
        on p.ProductID = f.Product_ProductID
        where f.offerPercent>5;
END //
```

OUTPUT:



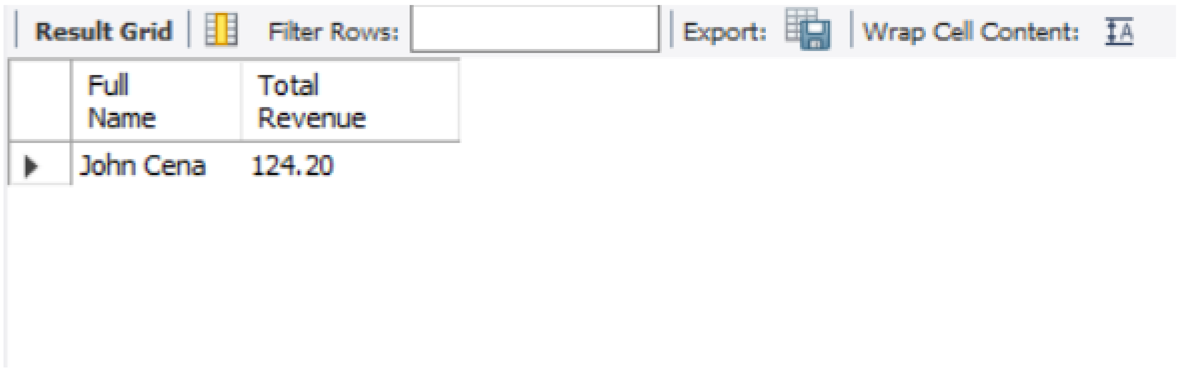
ProductName	Original Price	Discount OFFER
Hydrocodone	9.5	10
Generic Zocor	12	15
Azithromycin	19	20
Aceon	5	10

- ***Procedure which will return total revenue generated from customer***

```
Delimiter //
create procedure RevenuefromCustomer(in firstname varchar(50))
BEGIN
        select concat_ws(" ",c.Firstname,c.Lastname) as "Full
        Name",sum(round(ord.TotalPrice,2))as "Total Revenue"
        from customer c
        inner join onlinemedicalstore.order ord
        on c.CustomerID = ord.orderdetails_Customer_CustomerID
        where c.firstName = firstname
        group by concat_ws(" ",c.Firstname,c.Lastname);
END //

call RevenuefromCustomer('John');
```

OUTPUT:



Full Name	Total Revenue
John Cena	124.20

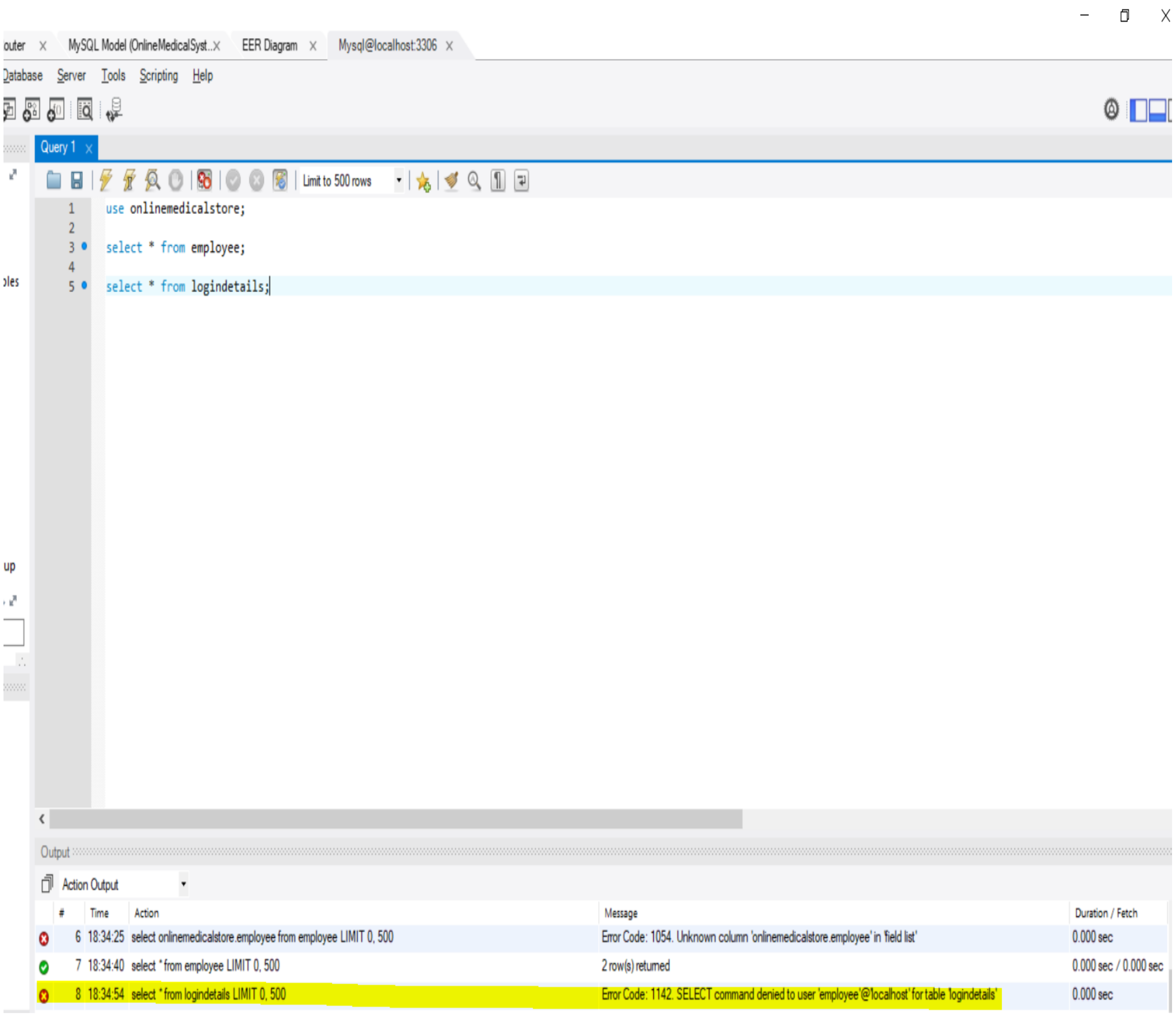
- ***Create User Employee which have access to all table except login details and login change table.***

```
create user 'employee'@'localhost' identified by  'employee';

GRANT ALL ON onlinemedicalstore.employee to 'employee'@'localhost'
GRANT ALL ON onlinemedicalstore.order to 'employee'@'localhost'
```

→ When employee group logins they are only able to view the details related to them where all the sensitive table information will be hidden.

OUTPUT:



```
use onlinemedicalstore;

select * from employee;

select * from logindetails;
```

```
#  Time      Action                                                         Message                                                                                   Duration / Fetch
6  18:34:25  select onlinemedicalstore.employee from employee LIMIT 0, 500  Error Code: 1054. Unknown column 'onlinemedicalstore.employee' in 'field list'              0.000 sec
7  18:34:40  select * from employee LIMIT 0, 500                            2 row(s) returned                                                                         0.000 sec / 0.000 sec
8  18:34:54  select * from logindetails LIMIT 0, 500                        Error Code: 1142. SELECT command denied to user 'employee'@'localhost' for table 'logindetails'  0.000 sec
```